Enhanced Eigen-Audioframes for Audiovisual Scene Change Detection

Marios Kyperountas, Constantine Kotropoulos, *Senior Member, IEEE*, and Ioannis Pitas, *Senior Member, IEEE*

***Abstract*—In this paper, a novel audio-visual scene change detection algorithm is presented and evaluated experimentally. An enhanced set of eigen-audioframes is created that is related to an audio signal subspace, where audio background changes are easily discovered. An analysis is presented that justifies why this subspace favors scene change detection. Additionally, a novel process is developed in order to detect audio scene change candidates in this subspace. Visual information is used to align audio scene change indications with neighboring video shot changes and, accordingly, to reduce the false alarm rate of the audio-only scene change detection. Moreover, video fade effects are identified and used independently in order to track scene changes. The false alarm rate is reduced further by extracting acoustic features in order to verify that the scene change indications are valid. The detection methodology was tested on newscast videos provided by the TRECVID2003 video test set. The experimental results demonstrate that the proposed method achieves an $F-measure$ exceeding 0.85. Accordingly, it effectively tackles the scene change detection problem.**

***Index Terms*—Audio-visual video segmentation, eigen-audioframes, scene change detection, TRECVID2003.**

I. INTRODUCTION

THE ever-growing volume of audio-visual data has revealed the need for developing proper algorithms, typically aiming to group audio-visual data into meaningful categories, index these categories, and provide fast browsing and retrieval. The integration of information provided by various media sources in order to handle multimodal queries on large and diverse audio-visual databases is the ultimate goal. The integration efficiency depends heavily on the proper categorization of the audio-visual information into semantic classes.

Video shot and scene detection is essential to automatic content-based video segmentation. A *video shot* is a collection of video frames obtained through a continuous camera recording. Similar background and motion patterns typify the set of frames within a shot. Whereas shots can be thought of as the basic units of video grouping, they usually lead to a far too fine segmentation in terms of the semantic audio-visual data representation. In order to acquire an effective non-linear access to video information, the data are grouped into *scenes*, where scenes are defined as sequences of related shots chosen according to certain semantic rules.

Manuscript received December 23, 2005; revised October 23, 2006. This work was supported in part by the FP6 European Network of Excellence MUSCLE "Multimedia Understanding through Semantics, Computation, and Learning" (FP6-507752). The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Yoshihisa Shinagawa.

The authors are with the Department of Informatics, Aristotle University of Thessaloniki, Thessaloniki 54006, Greece (e-mail: costas@aiia.csd.auth.gr).

Digital Object Identifier 10.1109/TMM.2007.893337

However, the aforementioned definition of scenes is quite vague, as different people can use different criteria to determine the scene borders. To make matters worse, different principles are used to define scenes for TV news programs, talk shows, documentaries, or movies. Several papers try to define models for scene detection, mainly in the field of TV news, where simple and effective models can be defined [1]. The definite structure of news videos and their suitability for content analysis was one reason that explains why they have attracted the attention of the research community. The interest of broadcasters in building large digital archives of their assets has provided additional motivation [2]. News archives are consistently used in news, talk shows, documentaries, or, sometimes, even in movies.

Typically, TV news programs have a very rigid structure. The algorithms in this paper were evaluated on the news model that the U.S. channel 'ABC' uses, which contains the following basic news elements.

- Headings accompanied by anchorperson speech along with video and audio special effects.
- Station news logo animation accompanied by music.
- Anchorperson close-up and speeches; sometimes news-story related video is displayed.
- Pre-recorded journalist reports, interviews, and commentary shot at various locations. During these audio-visual segments background noise (e.g., crowd or car noise) is usually quite evident. In some cases, music is inserted.
- Remote interviews, where the anchorperson converses with a reporter or a key-figure to a story that is not present in the studio. Frequently, the second person is shown using a digital 'window' or a large screen inside the news-studio.
- Commercials that are customarily separated from one another and from the rest of the audio-visual material using video fade effects and silence.

So far, most of the developed video segmentation algorithms merely exploit visual information. However, it has been observed that the visual information alone does not satisfactorily detect semantically meaningful scenes [3]. The use of multiple camera angles or special effects further complicates this task. As a result, shot changes are misclassified as scene changes in many cases. Typically, an efficient method to separate a video scene change from a shot change is by employing the audio information. A scene change is most often accompanied by a significant change in the audio characteristics, whereas a shot change is not. A notable example is a typical TV commercial, which constitutes a single scene with similar audio characteristics, but consists of numerous shots. At this point, the definition of an audio scene semantic unit is essential. An *audio scene* is a semantically consistent audio segment that can be distinguished by the basic characteristics of the dominant sound. It can be detected when the majority of dominant sources changes [4].

It is well known that audio-visual program and movie directors use audio not only to convey critical information to the audience, such as dialogues, but also to stimulate and maintain audience's interest. To do so, different scenes are intentionally accompanied by characteristically dissimilar sounds. Moreover, highly dissimilar audio characteristics among the scenes are naturally found in news broadcasts, where the background noise in the news studio almost always differs characteristically from the background noise of the pre-recorded news story clips. Whereas audio information is critical for the detection of scene changes, the joint processing of visual and audio information could improve scene change detection. The actual integration of the two information sources, in order to acquire a final scene change detection decision, can be a challenging problem due to the versatile nature of video content. Unsynchronized visual and audio content streams further complicate the fusion at the data level.

This paper introduces and evaluates experimentally a novel audio-visual method for video scene change detection that heavily depends on the use of the audio information. The success of the audio-based scene change detection approach can largely be attributed to the frequent presence of similar background noise such as car, crowd, or room noise, throughout each individual scene and the limited presence of foreground audio signal components during scene changes. Video information is exploited to extract video shot change information, which is employed to synchronize audio scene change indications with relevant video shot changes. Moreover, by considering a time span in which both an audio scene and a video shot change should co-occur, the false alarm rate of the proposed audio-only scene change detection algorithm is reduced. Additionally, video information is used to track scene change indications, simply by identifying video fade effects that are commonly employed during scene changes. A *video fade* is a transition of gradual diminishing content to a blank frame (fade out) or the opposite (fade in). The false alarm rate is reduced further by extracting acoustic features before and after the scene change indications in order to verify that they are valid. In the following, state-of-the-art scene change detection techniques are reviewed.

A. State-of-the-Art Review

In recent years, video scene boundary detection and video structure parsing have received a lot of attention from the research community. Various multimodal approaches have been developed in order to solve this problem. Of course, in all multimodal scene change detection algorithms, a non-trivial issue is how the audio information is integrated with the visual information. Quite often, discordances between the visual and the audio data are met. For example, the visual data may suggest the insertion of a scene break, whereas the audio data may not and vice-versa. Due to the diversity in the video structure, the integration of information from both the visual and the audio channels remains very challenging. In [5], fuzzy c-means clustering is employed to detect boundaries between two different kinds of the audio signal called *audio-cuts*. After the audio-cut detection, the semantic correlation between adjacent audio shots is measured and the semantically correlated audio shots are merged into the same audio segment. Five audio classes are considered: silence, speech, music, speech with music background, and speech with noise background. The following five features are employed in order to classify each audio segment into one of the aforementioned five audio classes: the mean and variance of the audio signal power, the mean and the variance of the center of the gravity of the 0th sub-band, and the zero-crossing rate. Since a large number of audio scene changes correspond to silence segments and many pauses can be found in speech audio segments, numerous false detections can occur, if only the audio information is exploited. As a result, video shot changes are detected and a scene change is set, if a video shot change and an audio-cut c-occur within a small time interval. In [6], semantically meaningful movie events are extracted such as two-speaker dialogues, multiple-speaker dialogues, and hybrid events that accommodate less speech and more visual action. Initially, video shots are identified by employing a color histogram-based approach and each set of grouped shots is clustered and classified into one of the three categories: periodic, partly-periodic, and non-periodic. Then, every shot is classified into one of the following four audio classes: silence, speech, music, and environmental sounds. The ratio of the speech content in each shot is used to indicate if the event is a dialogue or a hybrid event, whereas face detection is employed to discriminate between two-speaker and multiple-speaker dialogues. In [7], the authors present a scene change detection method based on audio and visual features, which analyzes both the auditory and the visual sources and accounts for their inter-relation and synergy to semantically identify video scenes. The audio stream is classified into speech, music, environmental sound, or silence. Speech data are further decomposed into different subclasses representing the different speakers. Meanwhile, the visual analysis partitions the video stream into shots. By combining visual and audio features using certain temporal expectations, the scene change detection performance is enhanced and more semantic segmentations are developed. In [3], speech and non-speech segments are detected and the non-speech segments are further classified into music and environmental sound. The classification is based on the audio periodicity and audio features such as zero-crossing rate, short-time energy, spectral flux, and line spectral pairs. Audio scene changes are detected using 1 s long audioframes. Then, all the shot boundaries within a 1 s interval from an audio scene change are set as scene candidates. Subsequently, a color correlation algorithm is used for video shot clustering. In [8], a scene classification scheme is presented, which uses a Hidden Markov Model (HMM)-based classifier in order to classify pre-segmented clips into predefined scene classes such as commercials, basketball games, football games, news, and weather forecasts. Three approaches are proposed for joint audio-visual scene segmentation and classification. The first two approaches search for an optimal scene class transition based on the likelihoods computed for every short video segment belonging to a particular class. The third approach searches for the optimal path in a super HMM built by concatenating HMMs for different scene classes. All these approaches process the audio and the visual information simultaneously. Specifically,

fourteen audio features as well as color and motion features are employed. In [9], low and midlevel audio-visual features are statistically analyzed according to genre characteristics. These features are directly obtained in the MPEG compressed domain. Then a linear machine decision tree classifier is built to classify each shot into predetermined genre sets. In [10], the audio feature extraction is related to the detected video shots. Unlike most audio feature-based segmentation algorithms, the nature of the audio content (music, conversation, etc.) is no longer relevant. A scene change is indicated simply based on the differences of the audio features from the corresponding adjacent shots.

Several methods are summarized next that have been proposed particularly for segmenting TV-news programs. A two-stage scene classification scheme is usually employed for context-based indexing and retrieval in news videos. In the first stage, the video stream is segmented into video shots, whereas in the second stage, each shot is assigned to various content classes, e.g., anchorperson, report, or weather forecast. Interested readers may consult [2], [11] for studying several methodologies used to build the two-stage classifier. The two-stage methodology was applied not only to visual, but also to text, motion, and audio information. Shot classification is often assisted by methods such as 'anchorperson-spotting', since news stories are often separated by an anchorperson visual appearance [12]. Of course, news story segmentation is a different and more specific task than news video scene segmentation. The criteria of what constitutes a news story are defined in [13]. In [14], audio and video feature information is employed to automatically segment news items. Silence segments are detected in the accompanying audio and this information is integrated with shot segmentation results as well as anchor shot detection ones to determine the boundaries between the news items. In [15], the spectral properties of the audio types are analyzed and audio features based on them as well as on harmonic enhancement are employed to classify audio. A multi-model HMM is used to capture the spectral variations for each audio type. A hierarchical classification method is used to segment the audio streams into speech, commercials, environment sound, physical violence, and silence in multiple steps. To do so, the merits of a particular audio feature in detecting a specific audio type are exploited and primitive spectral features present in an audio type are modeled for audio segmentation. A hybrid approach to classify news videos is proposed in [16]. A decision strategy is built that relies on evidence stemming from text classifiers and audio-visual cues. Ten different categories are defined that cover the most species of news videos. Support Vector Machines (SVM) are exploited to compute text confidence scores from low-level textual features and Gaussian Mixture Models (GMM) are employed to compute audio-visual confidence scores from low-level audio-visual features. The hybrid decision strategies perform the classification in a text-biased way. In one of the most popular applications for news shows segmentation, namely the News-on-Demand [17], news from TV and radio sources are indexed, thus enabling users to retrieve and browse news by content. The system creates a time-aligned transcript from speech recognition and audio classification. Specifically, speech recognition creates a text stream that is segmented in correspondence with acoustic condition changes and long pauses. The different acoustic conditions are related to six generic audio classes in which the audio segments are classified to by employing GMMs.

B. Novelty and Outline of Paper

During news story reports, the use of the video information only rarely helps to find a scene change, since various sequential shots that belong to the same scene are often completely uncorrelated in terms of the visual content. This problem aggravates when pre-recorded news stories are shown, where each news video sequence contains several video shots that can be quite dissimilar to one another with respect to their visual characteristics. Therefore, a novel method is proposed that uses primarily audio information in order to detect scene changes in an audio subspace using Principal Component Analysis (PCA) that favors the detection of background noise and background audio changes during scene transitions in the audio track. In regards to the temporal correspondence between audio and video, it has been estimated that the audio and video do not match with respect to who is speaking in broadcast news in approximately 85% of the time, whereas in the remaining 15% the voice and face match [18]. Therefore, we do not depend on the simultaneous detection of an audio and a video scene change (i.e., a perfect match), in order to set a boundary. Rather than using video scene information, video shots are employed to reduce the false alarm rate of the audio-only scene change algorithm. The false alarm rate is further reduced by comparing acoustic features near a scene change point.

Contrary to most video news analysis approaches, the proposed method tracks audio scene changes in a single stage. There is no need to partition the audioframes into a predefined number of classes, as in [5]–[9], [11], [15]–[17]. The algorithm does not use any news-specific a-priori knowledge such as anchorperson spotting as in [12]. Instead, it detects certain scene transition types, as is described in the next section, that are often found in news videos as well as in other types of TV programming and films. Accordingly, it qualifies as a potentially effective tool for scene change detection in other types of audio-visual content besides news videos.

The outline of the paper is as follows. Section II shows how PCA can be used for detecting scene changes by using the audio-eigenframes. A process that is used to detect candidate audio scene changes is presented. In Section III, it is justified why different subspaces can be used to decompose the audio signal into its different semantic components and an experimental procedure is carried out in order to determine one particular subspace that can aptly be used to solve the scene change detection problem. Section IV describes the integration of the proposed audio scene change detection with the video shot boundary detection that uses the mutual information and the joint entropy. In Section V, experimental results are demonstrated and discussed in order to evaluate the performance of the proposed overall algorithm and comparisons are made with relevant state-of-the-art scene change detection techniques. Finally, conclusions are drawn and guidelines for future work are laid out in Section VI.

II. Eigen-Audioframes for Scene Change Detection

In order to gather information on how the audio content behaves during and around scene changes a number of films, documentaries, and news videos have been collected from TV broadcasting and the internet and studied. To train the scene change detection algorithm presented in Sections II and III that exploits only the audio information, the news videos found in [19] were used. In particular, 6 "CBC News Online: Canada Now" videos of daily news programming having a duration of half an hour were collected over a period of 2 weeks. The audio background noise was extracted from 2 out of the 6 "CBC News Online: Canada Now" news videos in order to create a proper template. These 2 videos are not used in any other part of the subsequent training process. Therefore, we will refer only to the 4 remaining "CBC News Online: Canada Now" news videos as the training videos onwards. In addition, the 6 "CBC News Online: Canada Now" videos are excluded from the set of news videos used in Section V to evaluate the algorithm. The audio information that accompanies the videos was encoded at 34 kHz and then down-sampled to a sampling rate of 11.25 kHz.

Having studied the aforementioned video sequences, it is concluded that scene changes can efficiently be detected by considering the audio background information or background noise. For example, we can distinguish a scene that represents a report of a soccer game from a scene where a journalist is reporting from a busy street by comparing the characteristic differences between crowd and car traffic background noise. If the variations between the various types of background noise are sufficiently large, as is the case during scene transitions, then such variations can be exploited efficiently for scene change detection.

To model the audio content let us consider the decomposition of the audio signal into the background audio (BA), the transmission noise (TN), and the foreground signal (FS). Moreover, the scene transition region is specified as the temporal neighborhood around a scene change from the point when the foreground signal disappears (or begins to disappear) until it fully re-emerges after the occurrence of the scene change. During the transition from scene i to scene $i+1$, two different sequences are found to describe consistently the majority of audio content changes

$$\begin{aligned} &Scene\_change\_sequence\_a \\ &= \{\{FS_i + BA_i + TN\}\{BA_i + TN\} \\ &\quad \times \{FS_{i+1} + BA_{i+1} + TN\}\}, \end{aligned} \tag{1}$$

$$\begin{aligned} &Scene\_change\_sequence\_b \\ &= \{\{FS_i + BA_i + TN\}\{BA_i + TN\} \\ &\quad \times \{BA_{i+1} + TN\}\{FS_{i+1} + BA_{i+1} + TN\}\}. \end{aligned} \tag{2}$$

In certain cases, an audio fade-out effect is applied at the beginning of a scene transition so that the intensity of the background noise and any foreground signal present in the first scene gradually diminishes. In addition, an audio fade-in effect is sometimes used after the scene change. The foreground signal transition from one type of audio content to the next one is carried out in either an abrupt or gradual fashion, whereas the transition of background noise from one scene to the next is always abrupt.

Let the audio signal be split into non-overlapping audio segments of equal size (i.e., duration measured in samples or ms), called *audioframes*. To maintain simplicity, let us first consider only a single feature extracted from an audioframe, namely the short-term energy of an audioframe. As is suggested by (1) and (2), only background audio and transmission noise are present at some point during a scene transition. Outside the scene transition region, the audio signal is dominated by the foreground signal (e.g., speech or music), because the variance of the short-term energy is much larger there than in the background noise region. Therefore, we can expect that the scene transition regions are located at relatively low levels of audio signal variance. In some cases, as (2) indicates, the scene change point can be detected at the location where the characteristics of background noise change rapidly, from one scene to the next. Thus, finding the point of the largest variation in the short-term energy in the background noise region can help detecting the scene change point. The proposed method anticipates detecting scene change points in sequences (1) and (2) and in cases where fade in or fade out effects are applied to either sequence.

Let us argue on the use of the subspace spanned by the eigenvectors associated to the moderate eigenvalues of the covariance matrix of the audioframes for audio scene change detection. For simplicity reasons, let us assume that two different audio classes $\Omega_i, i = 1, 2$ are present. Let P_1 and P_2 be their *a priori* probabilities. Each class is modeled by its class mean vector $\boldsymbol{\mu}_i$ and the class covariance matrix $\boldsymbol{\Sigma}_i, i = 1, 2$. To detect if the audioframe $\mathbf{x} \in \Omega_2$ using subspace analysis, we are seeking a column orthogonal matrix $\mathbf{W}$ (i.e., $\mathbf{W}^{\mathrm{T}}\mathbf{W} = \mathbf{I}$, where $\mathbf{I}$ is the identity matrix) such that

$$E\left\{\left\|\mathbf{W}\mathbf{W}^T(\mathbf{x} - \boldsymbol{\mu}_2)\right\|^2 | \mathbf{x} \in \Omega_2\right\} \rightarrow \max \tag{3}$$

$$E\left\{\left\|\mathbf{W}\mathbf{W}^T(\mathbf{x} - \boldsymbol{\mu}_2)\right\|^2 | \mathbf{x} \in \Omega_1\right\} \rightarrow \min \tag{4}$$

where ^{T} is the transposition operator, $\|\ \|$ denotes the L_2 norm, and $E\{\}$ is the expectation operator. Equivalently, $\mathbf{W}$ should simultaneously satisfy

$$\mathrm{tr}(\mathbf{W}^T\boldsymbol{\Sigma}_2\mathbf{W}) \rightarrow \max \tag{5}$$

$$\mathrm{tr}(\mathbf{W}^T\widetilde{\boldsymbol{\Sigma}}_1\mathbf{W}) \rightarrow \min \tag{6}$$

where tr() stands for the trace operator and $\widetilde{\boldsymbol{\Sigma}}_1 = \boldsymbol{\Sigma}_1 + (\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2)(\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2)^T$. The solution of the optimization problem (5) is to choose the columns of matrix $\mathbf{W}$ as the eigenvectors that are associated to the largest eigenvalues of $\boldsymbol{\Sigma}_2$, while the solution of the optimization problem (6) is to choose the columns of matrix $\mathbf{W}$ as the eigenvectors that are associated to the smallest eigenvalues of $\widetilde{\boldsymbol{\Sigma}}_1$. To accommodate both requirements, we constrain the solution to subspaces spanned by the eigenvectors of the total covariance matrix $\boldsymbol{\Sigma}$ that is related to the class covariance matrices $\boldsymbol{\Sigma}_i, i = 1, 2$ through

$$\boldsymbol{\Sigma} = P_1\boldsymbol{\Sigma}_1 + P_2\boldsymbol{\Sigma}_2 + P_1P_2(\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2)(\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2)^T. \tag{7}$$

By doing so, we circumvent dealing with the null space of $\widetilde{\boldsymbol{\Sigma}}_1$ that should somehow be approximated. Of course the constrained solution is a suboptimal one. To avoid any feature extraction, we rely on the raw audio signal intensities within an audioframe and we apply PCA to the $N \times M$ matrix $\mathbf{X}$ whose

columns are the M zero-mean non-overlapping audioframes to which an audio stream is segmented to:

$$\mathbf{X} = [\underline{\mathbf{x}}_1 \quad \underline{\mathbf{x}}_2 \quad \dots \quad \underline{\mathbf{x}}_M]. \tag{8}$$

In (8), each $\underline{\mathbf{x}}_i$, $i = 1,2,\dots,M$, has $N = 3750$ samples obtained by sampling an audioframe (i.e., audio recording) of duration 1/3 s at a sampling rate of 11.25 kHz. M is equal to 5400, since in our case, a half-hour-long news audio signal is processed. To obtain the zero-mean audioframes, the mean audioframe is subtracted from each raw audio frame. The mean audioframe is estimated by averaging the audioframes from all the training audio recordings. The PCA of the covariance matrix of the audioframes can truly help in detecting a scene change point. Indeed, the audioframes that belong to a single background noise class are located in the subspace defined by the eigenvectors that correspond to a portion of the smallest eigenvalues of the covariance matrix of the audioframes. However, as it was shown above, when several background noise classes are present (as in news sequences where several audio background noise sources are found), the variations of background noise from one scene/class to the other yield audioframes that are located in a subspace defined by eigenvectors that are associated to small eigenvalues of the covariance matrix of the audioframes, but not the smallest ones. In the following, we shall assume that a subspace defined by a set of K eigenvectors of the covariance matrix of the audioframes associated to K appropriately chosen eigenvalues is used. Let us call the eigenvectors associated to the eigenvalues of this covariance matrix *eigen-audioframes*. The eigen-audioframes define a subspace where the original audioframes are projected onto.

Let $Q = \min(M, N)$, then the $Q - 1$ non-zero eigenvalues of the $N \times N$ covariance matrix of $\mathbf{X}$ and their corresponding eigenvectors can be found by solving a symmetric eigenvalue problem that employs a $Q \times Q$ matrix and taking the linear combinations of the resulting vectors, as is described in [20]. This is particular useful when short audio recordings are processed, such that $M < N$. Let $\mathbf{E}$ be the $N \times (Q-1)$ matrix, whose columns are the distinct eigen-audioframes. In most recognition, detection, or compression applications, a much smaller number of eigen-audioframes is retained. Let $\mathbf{E_S}$ denote the $N \times K$ matrix whose columns are the retained K eigen-audioframes, with $K \ll Q$. In order to project $\mathbf{X}$ onto the eigenspace of rank K, a single matrix operation is needed:

$$\mathbf{T} = \mathbf{E}_\mathbf{S}^{\mathrm{T}}\mathbf{X}, \tag{9}$$

where $\mathbf{T} = [\underline{\boldsymbol{\tau}}_1, \underline{\boldsymbol{\tau}}_2, \dots, \underline{\boldsymbol{\tau}}_M]$ is the $K \times M$ matrix that contains all the projected audio frames. Each projected audio frame, $\underline{\boldsymbol{\tau}}_i$, can be treated as a point in the eigen-audiospace.

Let $\underline{\widetilde{\boldsymbol{\upsilon}}}$ be the sample average of several background noise frames picked up during various scene transition periods when no foreground signals were present. The audio background noise frames were extracted from 2 out of the 6 "CBC News Online: Canada Now" news videos that were not used in any other part of the training process. Let $\underline{\boldsymbol{\upsilon}}$ be the zero-mean reference background noise frame obtained by subtracting the mean audioframe of $\mathbf{X}$ from $\underline{\widetilde{\boldsymbol{\upsilon}}}$. By projecting $\underline{\boldsymbol{\upsilon}}$ onto the eigen-audiospace we get

$$\underline{\boldsymbol{\nu}} = \mathbf{E}_\mathbf{S}^{\mathrm{T}}\underline{\boldsymbol{\upsilon}} \tag{10}$$

which is used as a reference frame in the eigen-audiospace that is associated with the background noise. In order to follow the audio channel trends, the distance between $\underline{\boldsymbol{\nu}}$ and any projected audioframe $\underline{\boldsymbol{\tau}}_i$ is measured. Several distance measures can be used, e.g., the L2-norm. Hereafter, we shall rely on the L_2-norm:

$$D(i) = \|\underline{\boldsymbol{\tau}}_i - \underline{\boldsymbol{\nu}}\|, \quad i = 1,2,\dots,M \tag{11}$$

where the subscript i refers to audioframe time index. Under the condition that no two consecutive scene changes can occur close to one another in the temporal domain, the time instants of the local minima of the function $D(i)$ are set as potential scene changes. To determine the local minima, a temporal window of 12 s duration is applied to $D(i)$. The duration of the temporal window was selected based on experimental evidence about the minimum duration of a scene. To be more precise, the duration of the shortest commercial was found to be 12 s. The minimum value m_j of the distance function within the 12 s-long window, is found at each time instant

$$m_j = \min_l \{D(l), l = j - 12f_F, j - 12f_F + 1, \dots, j\} \tag{12}$$

where f_F is the number of audioframes per second (i.e., $f_F = 3$ frames per second in our case). Subsequently, a distance boundary signal, $Y(j)$, is created by assigning m_j to the temporal location of the most recent point that enters the window's coverage area. This value is updated on the fly once a lower value is found. Thus, the distance boundary signal is determined by

$$Y(j) = m_j, \quad j = 12f_F + 1, \dots, M. \tag{13}$$

Next, the points of the distance function $D(i)$ that lay on the generated distance boundary signal $Y(j)$ are set as potential scene change points. If more than one point exists within any 12 s interval, only the point with the minimum value is retained in the set of local minima points. Fig. 1 depicts a portion of $D(i)$ that corresponds to a 'commercial break' segment from 1 out of the 4 training "CBC News Online: Canada Now" news videos and shows the distance boundary signal that is created as well as the potential scene change points. Moreover, the true scene change points and a false alarm point are indicated with arrows. Finally, a global scalar is used for thresholding $D(i)$ to reduce false alarms. The latter are usually found to be further away from the reference noise frame in the eigenspace than the true scene changes are. In order to cancel the false alarm incident that occurs at $i = 751$ s, by means of thresholding, we would have to miss the true scene change at $i = 782$ s. For the example shown in Fig. 1, the K retained eigen-audioframes correspond to the eigenvalues that capture the 50–75% range of the covariance matrix trace, as is explained in Section III.

An important parameter for scene change transitions to be effectively mapped to as local minima of the distance function $D(i)$ is the length of the audio frames. During scene transitions, the audio signal characteristics experience large variations over rather large time windows. An example that illustrates the aforementioned observation is the audio fade-out effect that is often used during a scene end. The beginning of the new scene is accompanied by normal levels of audio intensity or audio intensity gradually climbs to such levels by either using an audio

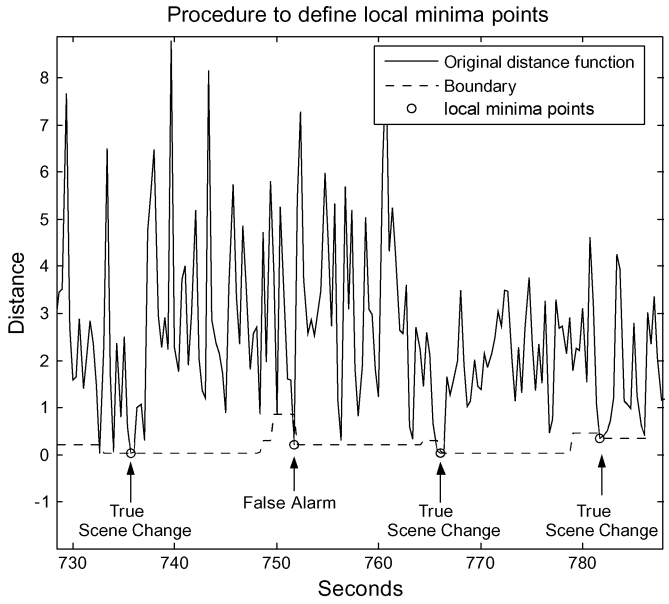

Fig. 1. Local minima of the distance function $D(i)$.

fade-in effect or by the gradual appearance of foreground signals. These transition periods are typically longer than speech pauses or other pauses in the foreground signal. With a sufficiently large audio frame size only the scene change transition regions will be mapped as local minima of $D(i)$ since the shorter pauses that are found in the foreground signal will combine with a significant amount of non-silence data in order to produce an audio frame, thus reducing the false alarm rate. On the other hand, the size of the audio frames should not be too large since the same could occur for the scene change transition regions, thus reducing the detection rate. One way to ensure that the size of the audio frame would not have to be too large and still be able to reduce the false alarm rate is to suppress the foreground signal. Therefore, in order to improve the scene change detection ability a subspace is created that reduces the representation of the foreground signal and transmission noise. This process is described next.

III. Optimum Audio Signal Subspace Selection for Scene Change Detection

Section II illustrated how scene changes can be detected by tracking the variations of background noise. Motivated by the expectation that the changes of the background noise can be sufficiently captured, a PCA-based scene change detection process is defined. This section describes which subspace $\mathbf{E_S}$ should be used in order to capture the background noise changes as much as possible. In addition, it explains how the remaining foreground and transmission noise signal components are suppressed and cannot degrade the detection performance by using PCA.

Seven experiments were carried out in order to discover which subset of eigenvalues is more suitable for solving the scene change detection problem. In each experiment, the eigenvectors are associated to various percentages of the trace of the covariance matrix of the audioframes starting with the eigenvector associated to the largest eigenvalue and proceeding toward the eigenvector that corresponds to the smallest eigenvalue. Specifically, tests were carried out using the following percentile ranges of the trace of the covariance matrix of the audioframes: full trace (0–100%); first half (0–50%), second half (50–100%); first quartile (0–25%), second quartile (25–50%), third quartile (50–75%), and fourth quartile (75–100%). These numbers were selected so as to provide a two-level decomposition of the audio signal variance. The seven experiments, one for each subspace, were repeated on 4 half-hour-long-each "CBC News Online: Canada Now" training news audio streams.

The goal of this analysis is to determine the appropriate set of eigenvectors that defines a subspace where improved audio-based scene change detection rates can be obtained. Let the ordered audio signal eigenvalues be represented by $\boldsymbol{\lambda} = [\lambda_1, \lambda_2, \ldots, \lambda_P, \ldots, \lambda_{P+K-1}, \ldots, \lambda_{Q-1}]$. Then, the K most useful eigenvalues are expected to belong to the subset $\boldsymbol{\lambda}_S = [\lambda_P, \lambda_{P+1}, \ldots, \lambda_{P+K-1}]$, where estimates for P and K are to be found. In order to compare the scene change detection ability for each one of the aforementioned seven subspaces, the following procedure was realized: the principal components from each subspace were calculated and used to project the audio frames as well as the reference noise frame onto each eigenspace. Then, the Euclidean distance between the projected audioframes and the projected reference noise frame was calculated.

In order to evaluate the detection performance of each subspace, the Recall and Precision rates (probabilities) were used [21]. Let GT denote the ground truth set and Det the set of detected (both correct and false) scene changes. Let $|A|$ define the cardinality of set A. Then we can define the following.

- The $Recall$ rate, that is also known as the true positive function or sensitivity, as

$$Recall = \frac{|Det \cap GT|}{|GT|}. \tag{14}$$

- The $Precision$ rate, that corresponds to the accuracy of the method considering the false detections, as

$$Precision = \frac{|Det \cap GT|}{|Det|}. \tag{15}$$

It is difficult to select a common threshold on the values admitted by the distance function $D(i)$ in each subspace due to the different range of values, whereas a threshold on the number of local minima of $D(i)$, as defined in Section II, can be set more easily. The local minima of $D(i)$ corresponding to the J lowest values of $D(i)$ were measured and the $Recall$ and $Precision$ rates for the set of J points were registered.

Each of the 4 "CBC News Online: Canada Now" training news videos contained on average 41 scene changes. We note that the training news videos are exclusively used to determine an approximation of the best subspace and the optimum value of J for the problem of scene change detection. The aforementioned training videos are not included in the evaluation experiments described in Section V. A detection performance analysis was carried out by collecting $Recall$ and $Precision$ rates, while J varies. By doing so, the tradeoff between the correct detections and false alarm incidents was observed and compared in all the 7 subspaces. Fig. 2(a) and (b) show how $Precision$ and $Recall$ vary as J increases, respectively. These

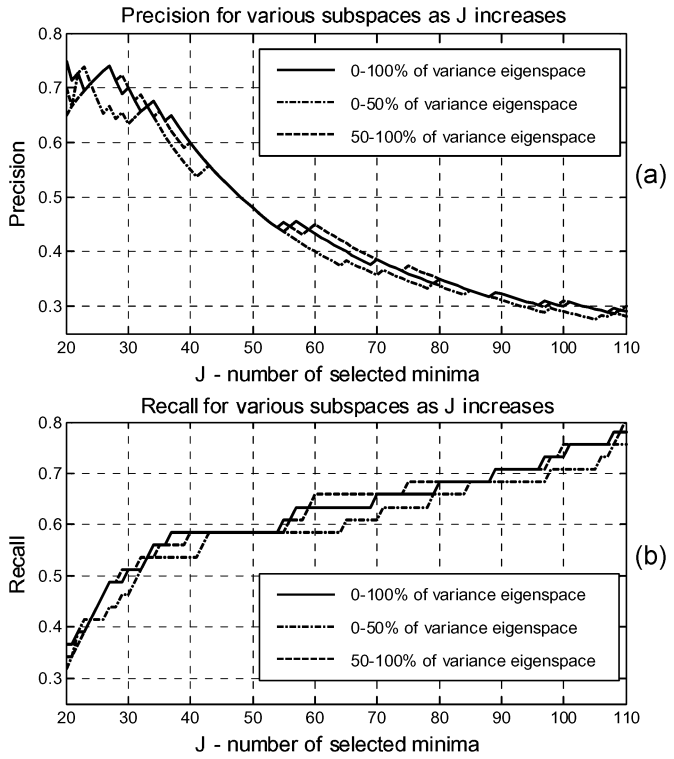

Fig. 2. $Precision$ and $Recall$ rates as the number of selected minima of the distance function, J, increases for several portions of the trace of the covariance matrix of the audioframes.

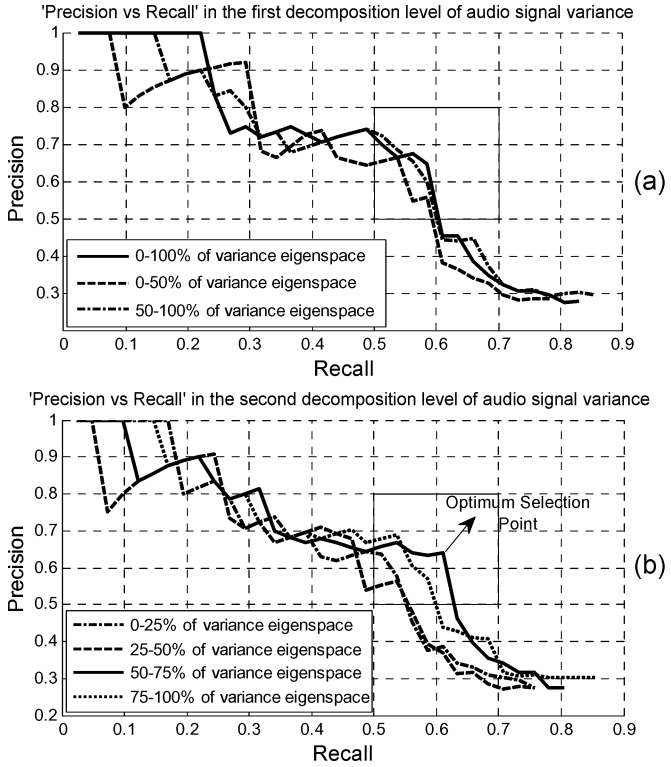

Fig. 3. $Precision$ versus $Recall$ curves in the various subspaces. (a) $Precision$ versus $Recall$ when the audio signal variance is decomposed into two subspaces. The first subspace is defined by the eigenvectors associated to the eigenvalues in the first half of the trace of the covariance matrix of the audioframes and the second one is defined by the remaining eigenvectors. (b) $Precision$ versus $Recall$ when the audio signal variance is decomposed into four subspaces, where each subspace is defined by the eigenvectors associated to the eigenvalues in successive quartiles of the trace of the covariance matrix of the audioframes.

results were drawn from an experiment that processed only a single training news video sequence. $Precision$ and $Recall$ curves, like the ones shown in Fig. 2(a) and (b), were collected for the 4 half-hour-long training news videos. In each experiment, J was assigned values from 1 to 120 with step-size 1. The results were then averaged, for fixed values of J, over all news-videos and are shown in the $Recall$ versus $Precision$ curves in Fig. 3(a) and (b). Each of these curves demonstrates detection results obtained in various subspaces. In order to determine which subspace is more suitable for scene change detection, certain criteria should be met by both the $Recall$ and the $Precision$ rates to guarantee acceptable results. Intuitively, we require that both $Recall$ and $Precision$ exceed 50%. This criterion is roughly met for $30 \leq J \leq 50$ for all subspaces. Note that exact but unique values of J in each subspace can be found that satisfy this condition, as can be seen in Fig. 2. The preferred operating region is marked by a solid-line rectangle in Fig. 3(a) and (b). Table I shows the area under the $Recall$ versus $Precision$ curve for the preferred operating region for the 7 subspaces that were evaluated. The largest area under the $Recall$ versus $Precision$ curve was obtained in the subspace associated to the 50–75% percentile range of the covariance matrix trace.

Fig. 3(a) illustrates that the subspace which captures the second half of the trace of the covariance matrix accommodates better a solution for scene change detection than the subspace that captures the first half of the trace of the covariance matrix. An important remark coming from Fig. 3(a) is that in certain cases, (e.g., for a $Recall$ rate between 0.5 and 0.55), the decision taken in the complete audio eigenspace is not as efficient as the decision taken in the subspace associated with 50–100% of the covariance matrix trace. This fortifies our decision to use subspace selection in order to improve the detection performance.

TABLE I
AREA UNDER THE RECALL VERSUS PRECISION CURVE WHEN BOTH RATES EXCEED 50%

Percentile range of the covariance matrix trace retained (%)	Area in the preferred operating region of Fig. 3
0-100	0.0166
0-50	0.0099
50-100	0.0160
0-25	0.0049
25-50	0.0026
50-75	**0.0175**
75-100	0.0128

Fig. 3(b) shows the $Recall$ versus $Precision$ curves obtained in subspaces at the second decomposition level. It is clear that the two subspaces that capture the second half of the trace of the covariance matrix of the audioframes (50–75% and 75–100%) promise a better detection performance than the remaining two subspaces. Moreover, the eigen-audioframes that capture the first and second quartiles of the trace (0–25% and 25–50%) almost always offer lower $Recall$ and $Precision$ rates than those

obtained from the subset that retains the first half of the trace. The subspace that retains the fourth quartile of the trace rarely outperforms either the subspace that contains the complete set of eigen-audioframes or the subspace that retains the second half of the trace in terms of $Recall$ and $Precision$. On the other hand, the subspace defined by the eigen-audioframes corresponding to the eigenvalues in the third quartile (50–75%) of the trace of the covariance matrix provides the best performance in many occasions.

It has been observed that most false alarm incidents that deteriorate the performance of the proposed algorithm are related to speech pauses. Speech segments contribute highly to the overall variance of the audio signal. Thus, large-eigenvalue subspaces such as those corresponding to the first half or the first and second quartile of the trace of the covariance matrix should not be used for detecting scene changes. This is verified by the results of Table I, where these eigenspaces are shown to be the least effective. Conversely, a small eigenvalue subspace such as the one associated with the fourth quartile of the trace, models properly only the background audio and the transmission noise. Additional tests have shown that this subspace could not be used to detect efficiently scene changes, since by not having any indication on the presence/absence of the foreground signal it is difficult to identify the scene change transition regions in (1) and (2). Moreover, in the subspace under discussion, scene change detection becomes susceptible to false alarms that are caused by the random transmission noise variations. Accordingly, it is evident by the measurements in Table I, that the subspace associated with the fourth quartile of the trace should not be used for detecting scene changes. On the other hand, the subspace that captures 50–75% of the trace of the covariance matrix of the audioframes provides the best compromise for efficient scene change detection. As Table I illustrates, this particular subspace yields the largest area under the $Recall$ versus $Precision$ curve for the selected operating region.

In order to determine the optimum value for J, the well-known $F-measure$ [5] is employed

$$F-measure=\frac{2\cdot Recall\cdot Precision}{Recall+Precision}. \tag{16}$$

The largest value that the $F-measure$ acquired is 0.625 and is achieved when $J=39$ in the 50–75% subspace. When J is set to 39, the $Precision$ slightly exceeds 64% and $Recall$ is found to be 61%. The optimum point (for $J=39$) is overlaid in Fig. 3(b). Both rates outperform the previously reported rates in [22]. In particular, the $Precision$ rate has improved by 7% and the $Recall$ rate by 8%.

IV. Integrating Audio and Video Cues and Verifying Scene Change Indications

In this section, a method for scene change detection that integrates audio and video cues is presented. In addition, a method that employs commonly used audio features to verify that the audiovisual scene change indications are valid is described. It is well-known that abrupt cuts and fade effects indicate video shot changes. Video abrupt cuts and video fades can be detected by means of the mutual information and the joint entropy between two video frames. Mutual information is a measure of information conveyed from one frame to another and is used for detecting abrupt video cuts, where the image intensity or color is abruptly changed. In the case of a video fade-out, where the visual intensity usually decreases to a black image, the inter-frame joint entropy decreases, whereas for a fade-in the joint entropy increases. A detailed description of this method used to detect video shot boundaries can be found in [23]. We used a simplified version of the method described in [23] for video fade detection, since, for our purpose, it is sufficient to detect the start or the end of a fade effect. If both a fade-out and a fade-in effect occur within 5 s, then only a single shot change is set at the midpoint between the corresponding time instants. Thus, a smaller number of fade effects is detected than that determined by the method in [23]. Video fade effects are used to detect scene changes as well.

After processing the audio track in order to locate the potential scene change points and the video track to find the possible shot changes and fade effects, a process is described that integrates all the aforementioned information in order to enhance scene change detection. Let T_{AU} be a vector of length n_{AU} that contains all the potential scene change points collected by applying the process described in Sections II, III to the audio information. Let T_{VD} be a vector of length n_{VD} that contains the shot change points collected by detecting video abrupt cuts and video fades. Finally, we define a temporal window W_I with duration of t_I s (typically 3 s) within which the audio and the video convey information about semantically correlated events. For example, when a shot change is indicated by processing the visual data, we expect the corresponding scene change derived by processing the audio information to occur within t_I s. The following algorithm integrates the visual and audio information clues

$k=0$

for $i=1:n$

$v=\min_j(T_{VD}[j]-T_{AU}[i])\ j=1,2,\ldots,n_{VD}$

$l=\arg\min_j(T_{VD}[j]-T_{AU}[i])$

if $|v|\le t_I$

$k=k+1$

$Scene\_candidate\_aligned(k)=T_{VD}(l)$

end

end,

where v is the minimum time mismatch between a video shot change point and the ith audio scene change point and l is the corresponding index in T_{VD}. $Scene\_candidate\_aligned$ holds the time instants of the audio-based scene change candidate points that neighbor with a video shot change within t_I. Whenever the scene change indications from the audio information do not match with a shot change indication from the video information within a time window of duration t_I, then the scene change is rejected as a false alarm. Otherwise, the audio scene change point is time-aligned to the location of the nearest video shot

change point, thus time-aligned candidate scene change points are now created. Furthermore, let T_{FD} be a vector that contains only the points that correspond to the detected fade effects. In addition to the time-aligned candidate scene changes stored in $Scene\_candidate\_aligned$, each point for which a video fade effect was detected (i.e., a point in T_{FD}) is also classified as a candidate scene change point.

This idea of requiring for audio scene change candidate points and video shot cut points to be temporally aligned in order to set a scene change has been shown to be quite effective in [5] and [24]. However, whereas this condition gives a high probability that the candidate scene change point will in fact correspond to a true scene change, it does not guarantee it. In addition, there is no guarantee that a video fade effect is always used to separate two scenes. As a result, further processing is required in order to validate if the candidate scene change points that correspond to the temporal locations in $Scene\_candidate\_aligned$ and T_{FD} are true scene changes. More specifically, two 6-s long audio segments are collected, equally split before and after each candidate scene change point. We do not extract acoustic features from the entire scene because the foreground signals can vary significantly throughout the scene, especially when its duration is long. For the task at hand, the average of these variations may not be as meaningful as the average of the features near the candidate scene change point. Since the scene change transition region usually presents a general lack of foreground signals, which are now important for our analysis, the 6-s long audio segments are selected 3 s before and 3 s after the candidate scene change point. This 3-s-long temporal distance also ensures that the selected audio segment contains audio from a single scene only since the candidate scene change points are time-aligned to video shot change points within $t_I = 3$ s. Overlapping audio frames are extracted for each audio segment, where each frame consists of 512 samples and the overlap is set to 128 samples. Subsequently, the following set of acoustic features are extracted from each audio segment [10]: volume (V), energy (E), frequency centroid (FC), and frequency bandwidth (FB), as they are defined in [25]; low shot-time energy ratio $(LSTER)$ and spectral flux (SF), as they are defined in [26]; cepstral flux (CF), as it is defined in [4]; sub-band energy $(SubE)$ for the 0-to-1/16 fs and 1/8 fs-to-1/4 fs bands, where fs is the sampling rate. In addition, we calculate the low shot-time ratio of the sub-band energy $(LSTSubER)$. Then, these features are normalized based on their maximum value and for each ith audio segment the following three values, that are introduced in [10] are calculated:

$$\begin{aligned} C_{1,i} &= \mathrm{mean}(V_i) + \mathrm{std}(V_i) + \mathrm{vdr}(V_i) + \mathrm{stdif}(V_i), \\ C_{2,i} &= \mathrm{std}(E_i) + \mathrm{std}(SubE_i) + \mathrm{stdif}(E_i) \\ &\quad + \mathrm{stdif}(SubE_i) + LSTER_i + LSTSubER_i, \\ C_{3,i} &= \mathrm{std}(FB_i) + \mathrm{std}(FC_i) + \mathrm{stdif}(SF_i) + \mathrm{stdif}(CF_i) \end{aligned}$$

where mean() is the mean of a feature, std() is the standard deviation of a feature, vdr() is the volume dynamic range of a feature, and stdif() is the standard deviation of the frame-to-frame difference of a feature in the ith segment.

Let N be the number of audio segments extracted from a video sequence, each being 6-s long, and let $j = 1, 2, 3$ correspond to the three groups of features for volume, power, and spectrum. Then, for each pair of audio segments, i.e., the ith and $(i+1)$th segments, before and after a candidate scene change point, the following three statements, corresponding to $j = 1, 2, 3$, are examined:

$$\|C_{j,i} - C_{j,i+1}\| > \frac{1}{\sqrt{2}} \cdot \left(\underset{N-1}{mean} \left(\|C_{j,k} - C_{j,k+1}\| \right) + \underset{N-1}{std} \left(\|C_{j,k} - C_{j,k+1}\| \right) \right) \quad (17)$$

where $k = 1, \ldots, N-1$

$$\underset{N-1}{mean} \left(\|C_{j,k} - C_{j,k+1}\| \right) = \frac{1}{N} \sum_{k=1}^{N-1} \left(\|C_{j,k} - C_{j,k+1}\| \right), \text{ and}$$

$$\underset{N-1}{std} \left(\|C_{j,k} - C_{j,k+1}\| \right) = \sqrt{\frac{1}{N-1}} \cdot \sqrt{\sum_{k=1}^{N-1} \left(\|C_{j,k} - C_{j,k+1}\| - \underset{N-1}{mean} \left(\|C_{j,k} - C_{j,k+1}\| \right) \right)^2}.$$

The left side of (17) represents the Euclidean distances between the features of the two neighboring audio segments for which we try to determine if they are separated by a valid scene change point. The first and second numerator terms on the right-hand-side of (17) represent the mean and the standard deviation of the Euclidean distances between the features of the $N-1$ pairs of neighboring audio segments in the video sequence.

Essentially, the right-hand-side of (17) represents three threshold values, T_j, $j = 1, 2, 3$, that are calculated based on the statistics of all 6-s-long audio segments and not just based on the two neighboring audio segments. If the distance of any one of the three values is larger than its corresponding threshold value T_j, a true scene change point is set. The three distance values are not combined into one distance measure because of the following reasons [10]: each value contains different audio semantic meanings, and simply combining them will destroy these meanings. In addition, since different audio features may have different levels of importance for different audio data it is difficult to derive proper weights in order to combine them. As a result, the three T_j values are set to be complementary to each other. For a valid scene change point to be set, it is expected that the features of the audio segments before and after this point will considerable vary. By making use of the three independent thresholds, the inequality in (17) simply examines if any one of the three feature group values changes considerably from the previous audio segment to the next. In such a case, a decision is made that the aligned candidate scene change point is indeed a valid scene change point and its time instance is stored in $Scene$.

V. Experimental Results and Discussion

The proposed technique was tested on eight "ABC News" and "CNN Headlines" videos that are included in the test set of the well-established reference video test set TRECVID2003 [13]. These news shows have a rather rigid structure, in terms

TABLE II
RECALL AND PRECISION RATES FOR VIDEO SHOT CHANGE AND FADE DETECTION

Type of processed videos	Shot change detection using Mutual Information (%)		Fade detection using Joint Entropy (%)	
	Recall	Precision	Recall	Precision
4 'ABC News' videos	97.0	94.0	93.0	90.0
4 'CNN Headlines' videos	96.0	96.0	100.0	84.0

TABLE III
DETECTION PERFORMANCE BY SCENE CATEGORY

Scene category	Ground truth Scene changes per category	Detected scene changes				
		Au*EigFrame*	Vi*Fades*	AuVi*Align*	AuVi*Fuse*	AuVi*FuseValid*
News Headings	14	11	7	11	12	12
News Logo / to studio transition	25	15	12	15	18	18
Anchorperson	91	64	49	64	81	81
Remote-distance interview	21	7	0	7	7	6
Pre-recorded news stories	82	64	0	64	64	64
Commercials	130	71	113	71	116	110
Total	**363**	**234**	**181**	**234**	**298**	**291**

of semantic content, which makes them attractive when having to determine a ground truth for scene changes. For each video sequence, a human observer determined the precise locations of the scene changes; no official scene change ground truth data are yet available for the TRECVID2003 database. Overall, it was found that the 8, roughly half-hour-long each, test videos contain 363 scene changes. The video had a frame rate of 29.97 fps and each frame was resized to half of the original resolution, at 176×132 pixels in order to speed up calculations. The audio track that we used was converted to an 8-bit mono channel with a sampling rate of 11.25 kHz. Audio frames were extracted, with each one corresponding to roughly 1/3 s, or to 10 video frames.

Let the algorithm described in Sections II and III that is used to extract scene change information from the audio data be termed as 'Au*EigFrame*'. When projecting the audio frames to the eigenspace we used the eigenvectors associated with the ordered eigenvalues related to the third quartile (50–75%) of the trace of the covariance matrix of the audioframes, as Section III advices. 'Vi*Fades*' is the process that applies the algorithm that was used in order to detect candidate scenes from the video data by tracking video fade effects, as is described in Section IV. 'AuVi*Align*' is defined as the method that was used to align the audio-based scene change indications and video shot cuts described in Section IV. The value of parameter t_I was set to 3 s. 'AuVi*Fuse*' is the process that combines the detection results of both Vi*Fades* and AuVi*Align* by taking the set union of the candidate scene change points determined by each technique independently. Finally, 'AuVi*FuseValid*' is the approach described in Section IV that is used to validate the candidate scene change points of AuVi*Fuse* by comparing distances of acoustic features before and after a candidate scene change point, as (17) indicates. The latter approach represents the scene change detection in video sequences proposed in this paper.

Table II provides the recall and precision rates when detecting video shot cuts using mutual information and video fade effects using joint entropy. It is worth mentioning that the number of detected fade-in and fade-out effects is larger than the number of fade effects that are used for scene change detection, as is explained in Section IV. For more details on detecting video shot cuts and video fades, the interested reader is referred to [27]. Table III shows the type of scenes that are manually identified, how often they actually appeared in the test news videos as well as how many of them have been detected by the algorithms under discussion. This table verifies the high efficiency of the integration process for the audio and visual information. It is clear that the audio-only scene change detection algorithm works very well for tracking the scene changes when a news story video is processed. This observation justifies our expectation for high detection rates in segments where the presence of background noise is highly evident, as is the case during the transition to a pre-recorded news video from the newsroom. The brief silence segments, lacking any foreground audio signals right after when the anchorperson finishes introducing the news-story video that follows until the beginning of the report from the journalist in that news-story, allow the formation of a number of audio frames that contain pure background noise. These frames are then projected to the selected subspace and their distance $D(i)$ from the projected reference noise frame is found at levels low enough so as to be considered local minima point candidates. On the other hand, it was observed that if the audio scene change transition period or silence segment was too short, the Au*EigFrame* algorithm cannot detect the scene change. As stated in Section II, this is because the audio frames also contain the foreground signals that reappear shortly after the scene change occurs. One way to correct these failures is to use audioframes shorter than 1/3 s. However, the number of false alarms would also increase since speaker changes or speech pauses, which are usually quite short, may be falsely identified as audio scene changes.

A closer look at Table III reveals that the AuVi*Align* and Vi*Fades* processes independently detect 64.5% and 49.9% of the total number of existing scene changes, respectively. A number of scene changes is detected by both processes whereas all false alarm incidents add up to the final decision, as Table IV indi-

TABLE IV
FALSE ALARMS OF EACH APPROACH FOR SCENE CHANGE DETECTION

Process	False Alarm Incidents
Au*EigFrame*	134
Vi*Fades*	32
AuVi*Align*	37
AuVi*Fuse*	69
AuVi*FuseValid*	**28**

TABLE V
EVALUATION MEASURES OF EACH APPROACH FOR SCENE CHANGE DETECTION

Evaluation of Scene Change Detection Capabilities	*Recall* (%)	*Precision* (%)	*F-measure*
*Au*EigFrame	64.5	63.6	0.640
Vi*Fades*	49.9	85.0	0.629
AuVi*Align*	64.5	86.3	0.738
*AuVi*Fuse	**82.1**	81.2	0.816
***AuVi*FuseValid**	80.2	**91.2**	**0.853**

cates. However, the $Precision$ reported for AuVi*Fuse* is found to be lower than the rates achieved by the two processes separately, as can be seen in Table V. This table also shows that, as expected, AuVi*FuseValid* yields a significant improvement for $Precision$. From Table III it can be seen that processing the audio only in order to detect scene changes does not perform very well in commercial breaks. Typically, most commercials are quite 'refined' TV segments that are created on a very similar formula. Usually loud or emphatic music, or various sound effects, are inserted in order to draw the viewer's attention. Some commercials maintain the same audio intensity from the beginning until the very end of the video segment. The presence of foreground signals does not provide occurrences of audio frames containing pure background noise. Also, silence segments are used to connect one commercial to the next; thus, some of these scene changes can not be detected by processing the audio information.

As Table IV indicates there is a significant improvement in the false alarm rate of the Au*EigFrame* process by exploiting the corresponding video data. The temporal window of 3 s used by the AuVi*Align* process in order to guarantee that audio scene change indications roughly coincided with video shot changes significantly reduced the number of false alarms. Table V illustrates the overall performance of the AuVi*Fuse* algorithm which combines both audio and visual data in order to determine scene changes. The benefits of the AuVi*Fuse* approach are clearly demonstrated. AuVi*Align* improves the $Precision$ rate of the audio-based process, Au*EigFrame*, by roughly 23% while maintaining the same $Recall$ rate. Moreover, the proposed AuVi*Fuse* algorithm sacrifices a modest drop of 3.8% in $Precision$ with respect to Vi*Fades* (and 5.1% with respect to AuVi*Align*) due to the scene changes that were simultaneously detected by both algorithms. Furthermore, an improvement of 32.2% and 17.6% is obtained in $Recall$ with respect to the corresponding rates of the Vi*Fades* and the AuVi*Align* processes. Finally, the proposed AuVi*FuseValid* scene change detection scheme, which in addition compares acoustic signals around the candidate scene change points, offers the best solution since the $Precision$ rate increases by 10% for a modest drop of 1.9% in the $Recall$ rate, with respect to AuVi*Fuse*.

TABLE VI
EVALUATION MEASURES OF VARIOUS SCENE CHANGE DETECTION ALGORITHMS

Comparative Study on TRECVID2003 Database	*Recall* (%)	*Precision* (%)	*F-measure*
Nitanda et al. [5, 31]	62.4	80.6	0.703
Sundaram et al. [4]	56.4	67.1	0.613
Chen et al. [10]	**84.7**	63.1	0.723
***AuVi*FuseValid**	80.2	**91.2**	**0.853**

To enable performance comparison between the proposed AuVi*FuseValid* algorithm and other methods, the $F-measure$ (16) is once again employed. For a valid comparison among scene change detection methods, one should bear in mind that the type of transition (e.g., abrupt or gradual) that is used to connect scenes with one another can greatly affect how successful an algorithm is [28]. In addition, the video genre that is used can greatly influence the performance of a scene change detection algorithm [29]. For this reason, the $Recall$ and $Precision$ rates reported for the two algorithms in [30] were used, since an extensive evaluation was done in this work using ten different videos as test data for two different algorithms. Furthermore, no two videos belonged to the same film genre, or combination of genres. Thus, the set of test videos was quite diverse in principle. Moreover, the authors did not pre-categorize scenes before dealing with scene change detection (e.g., they did not simplify a news show into a stream that contains news story, commercial, weather, and anchorperson segments) thus avoiding generalization problems. The precision rates for the two methods reported in [30] were 73.9% and 79.2% whereas the corresponding recall rates were 86.3% and 78.7%. The associated $F-measure$ values were 0.796 and 0.789, which are lower than the $F-measure$ value achieved by the AuVi*FuseValid* algorithm as can be seen in Table VI.

Undoubtedly, since the AuVi*FuseValid* algorithm was tested on the TRECVID2003 database, the ultimate comparison test should use detection results from an alternative algorithm that has was tested on the same dataset. The recently proposed algorithm in [5] and [31] was evaluated on the same TRECVID2003 test set news videos as the proposed algorithm. The only difference is that in [31] the audio track had a sampling rate of 44.1 kHz. The algorithm in [31] introduced a novel way to detect audio scene changes. The same algorithm was employed in [5], which is reviewed in Section I-A, where a methodology is laid out that exploits the visual content in order to detect scene changes. Since [5] does not provide any novelty in detecting video shot changes or editing effects, such as fades, we use the same shot and fade detection results [27] that were also used while evaluating the performance of the proposed algorithm. In order to integrate the results from the audio and the visual information, the process described in [5] is followed. A temporal window of 3 s duration is again used to indicate if an audio shot change and a video shot change co-occur in order to declare a scene change. The parameters for the algorithm that processes the audio are the ones in [31]. The corresponding $Recall$ and $Precision$ rates, as well as the associated $F-measure$ can

be seen in Table VI. Ten additional runs of this algorithm were carried out, in order to record various $F-measure$ results by tweaking the values of the parameters specified in [31]. The largest recorded $F-measure$ value is 0.717. Accordingly, the AuVi*Fuse* as well as our overall solution AuVi*FuseValid* algorithm are more efficient than those in [5], [31]. It is noted, however, that for comparing the effectiveness of the audio part of these algorithms the $F-measure$ of AuVi*Fuse* should be used since the remaining video-processing and audio-visual integration scheme is identical to the one used to evaluate [5], [31].

Moreover, we implement the audio-based scene change detection algorithm of [4]. The optimal envelope fits for the audio features are determined as well as the correlation amongst these envelopes. The 'periodicity' and 'randomness' models are not implemented since they are also not used for the experiments described in [4]. As in [4], each audio frame has a duration of 100 ms whereas the analysis window is shifted by 1 s at a time. Moreover, the attention span parameter is set to 16 s and 5 different memory lengths—corresponding to 17, 19, 23, 31, and 37 s—are used, as in [4]. Based on these parameters, the largest $F-measure$ value that is recorded for experiments on the TRECVID2003 database is shown in Table VI along with its corresponding $Recall$ and a $Precision$ rate. The $F-measure$ of the audio-only Au*EigFrame* that is presented in this paper is larger -as Table V shows- and, obviously, the performance of [4] is inferior with respect to the performance of AuVi*FuseValid* that constitutes our overall solution.

In addition, we have implemented the scene change detection algorithm that is proposed in [10] where traditional acoustic features are extracted. A scene change is indicated based on the differences of 9 audio features that accompany corresponding adjacent video shots. In order to set up a comparable processing of the 11.25 kHz-sampled audio track to the one in [10], we consider for 512 samples to comprise an audio frame and each frame to be shifted by 128 samples from the previous frame. The shot boundary information, which is a preprocessing step to the audio-based detection of [10], is once again provided by the algorithm in [27] that was also used when evaluating the performance of the proposed algorithm. The 9 different audio features of the audio signal that accompanies each shot are produced by following the audio feature extraction process in [10]. The features are divided into three groups, namely a volume, a power, and a spectrum group and different values (such as mean, standard deviation, volume dynamic range etc. [10]) for each feature in each group are calculated and added. A scene change boundary is set if the distance of any one of the three values is larger than its corresponding threshold value, as is proposed in [10]. The corresponding $Recall$ and $Precision$ rates, as well as the associated $F-measure$, for the experiments on the TRECVID2003 database are shown in Table VI. Whereas [10] shows a higher $Recall$ rate than that of the AuVi*Fuse* or the AuVi*FuseValid* algorithm, its associated low $Precision$ rate brings about a much lower $F-measure$ than that AuVi*Fuse* or AuVi*FuseValid* achieves. Thus, our algorithm provides a better overall solution.

VI. Conclusion

This paper has introduced an audio-visual scene change detection algorithm that uses a subspace created by a set of eigenaudioframes that enable an enhanced representation of the background audio variations. A novel method for identifying scene change candidate points, which is well-accommodated in this particular subspace, has been employed. Audio scene change indications were integrated with shot and scene change information extracted from the visual information. These indications were further validated by comparing various acoustic features and the overall process was evaluated on news videos taken from the well-established TRECVID2003 database in order to enable consistent comparisons with past as well as future techniques. The results are very promising as $Recall$ and $Precision$ rates exceeding 80% and 91%, respectively have been attained. Future research could aim at determining which subspaces are more suitable for detecting specific video genres or types of TV programming in order to achieve even higher detection results.

Acknowledgment

The authors would like to thank their colleague E. Benetos for providing them the audio feature extraction toolbox that was used in the experiments.

References

[1] M. De Santo, G. Percannella, C. Sansone, and M. Vento, "Dialogue scenes detection in MPEG movies: A multi-expert approach," *Lecture Notes Comput. Sci.*, vol. 2184, pp. 192–201, Sep. 2001.

[2] M. Bertini, A. Del Bimbo, and P. Pala, "Content-based indexing and retrieval of TV-news," *Pattern Recognit. Lett.*, vol. 22, no. 5, pp. 503–516, 2001.

[3] H. Jiang, T. Lin, and H. J. Zhang, "Video segmentation with the assistance of audio content analysis," in *Proc. 2000 IEEE Int. Conf. Multimedia Expo*, 30 July–30 Aug. 2000, vol. 3, pp. 1507–1510.

[4] H. Sundaram and S. F. Chang, "Audio scene segmentation using multiple features, models and time scales," in *Proc. 2000 IEEE Int. Conf. Acoustics, Speech, and Signal Processing*, Istanbul, Turkey, Jun. 5–9, 2000, vol. 4, pp. 2441–2444.

[5] N. Nitanda, M. Haseyama, and H. Kitajima, "Audio signal segmentation and classification for scene-cut detection," in *Proc. 2005 Int. Symp. on Circuits and Systems*, May 23–26, 2005, pp. 4030–4033.

[6] Y. Li, S. Narayanan, and C.-C. J. Kuo, "Content-based movie analysis and indexing based on audio-visual cues," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 14, no. 8, pp. 1073–1085, 2004.

[7] Y. Zhu and D. Zhou, "Scene change detection based on audio and video content analysis," in *Proc. 5th Int. Conf. Computational Intelligence and Multimedia Applications*, Sep. 27–30, 2003, pp. 229–234.

[8] J. Huang, Z. Liu, and Y. Wang, "Joint scene classification and segmentation based on Hidden Markov Model," *IEEE Trans. Multimedia*, vol. 7, no. 3, pp. 538–550, 2005.

[9] M. Sugano, R. Isaksson, Y. Nakajima, and H. Yanagihara, "Shot genre classification using compressed audio-visual features," in *Proc. 2003 IEEE Int. Conf. Image Processing*, Barcelona, Spain, Sep. 14–17, 2003, vol. 2, pp. 17–20.

[10] S.-C. Chen, M.-L. Shyu, W. Liao, and C. Zhang, "Scene change detection by audio and video clues," in *Proc. 2002 IEEE Int. Conf. Multimedia and Expo*, Lausanne, 2002, vol. 2, pp. 365–368.

[11] W. H.-M. Hsu and S.-F. Chang, "A statistical framework for fusing midlevel perceptual features in news story segmentation," in *Proc. 2003 IEEE Int. Conf. Multimedia and Expo*, Baltimore, MD, 2003, vol. 2, pp. 413–416.

[12] A. O'Hare, A. F. Smeaton, C. Czirjek, N. O'Connor, and N. Murphy, "A generic news story segmentation system and its evaluation," in *Proc. 2004 IEEE Int. Conf. Acoustics, Speech, and Signal Processing*, Montreal, May 17–21, 2004, vol. 3, pp. 1028–1031.

[13] *Guidelines TRECVID 2003 Evaluation*, , Nat. Inst. Stand. Technol. (NIST) [Online]. Available: http://www.nlpir.nist.gov/projects/tv2003/tv2003.html

[14] W. Weiqiang and G. Wen, "Automatic segmentation of news items based on video and audio features," *J. Comput. Sci. Technol.*, vol. 17, no. 2, pp. 189–195, 2002.

[15] T. L. Nwe and H. Li, "Broadcast news segmentation by audio type analysis," in *Proc. 2005 IEEE Int. Conf. Acoustics, Speech, and Signal Processing*, Montreal, QC, Canada, Mar. 18–23, 2005, vol. 2, pp. 1065–1068.

[16] P. Wang, R. Cai, and S.-Q. Yang, "A hybrid approach to news video classification multimodal features," in *Proc. 2003 Joint 4th Int. Conf. Information, Communications, and Signal Processing and Fourth IEEE Pacific-Rim Conf. Multimedia*, Singapore, Dec. 15–18, 2003, vol. 2, pp. 787–791.

[17] N. Bertoldi, F. Brugnara, M. Cettolo, M. Federico, D. Giuliani, E. Leeuwis, and V. Sandrini, "The ITC-irst news on demand platform," in *Proc. 25th Eur. Conf. IR Research: Advances in Information Retrieval*, Pisa, Apr. 14–16, 2003, vol. 2633, pp. 520–527.

[18] A. Albiol, L. Torres, and E. J. Delp, "Combining audio and video for video sequence indexing applications," in *Proc. 2002 IEEE Int. Conf. Multimedia and Expo*, Laussane, Switzerland, 2002, vol. 2, pp. 353–356.

[19] "CBC news online," Canada Now [Online]. Available: http://www.cbc.ca/MRL/clips/latest/canadanow.ram

[20] M. Turk and A. Pentland, "Eigenfaces for recognition," *J. Cogn. Neurosci.*, vol. 3, no. 1, pp. 71–86, 1991.

[21] K. Fukunaga, *Introduction to Statistical Pattern Recognition*. New York: Academic, 1990.

[22] M. Kyperountas, Z. Cernekova, C. Kotropoulos, M. Gavrielides, and I. Pitas, "Audio PCA in a novel multimedia scheme for scene change detection," in *Proc. 2004 IEEE Int. Conf. Acoustics, Speech, and Signal Processing*, Montreal, May 17–21, 2004, vol. 4, pp. 353–356.

[23] Z. Cernekova, C. Nikou, and I. Pitas, "Shot detection in video sequences using entropy-based metrics," in *Proc. 2002 IEEE Int. Conf. Image Processing*, 2002, vol. 3, pp. 421–424.

[24] J. Huang, Z. Liu, and W. Yao, "Integration of audio and visual information for content-based video segmentation," in *Proc. 1998 IEEE Int. Conf. Image Processing*, Oct. 4–7, 1998, vol. 3, pp. 526–529.

[25] Y. Wang, Z. Liu, and J. Huang, "Multimedia content analysis using both audio and visual clues," *Signal Process. Mag.*, vol. 17, pp. 12–36, Nov. 2000.

[26] L. Lu, H. Jiang, and H. J. Zhang, "A robust audio classification and segmentation method," in *Proc. 9th ACM Int. Conf. on Multimedia*, Ottawa, Canada, 2001, vol. 9, pp. 203–211.

[27] Z. Cernekova, I. Pitas, and C. Nikou, "Information theory-based shot cut/fade detection and video summarization," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 16, no. 1, Jan. 2006.

[28] C.-L. Huang and B.-Y. Liao, "A robust scene-change detection method for video segmentation," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 11, no. 12, pp. 1281–1288, 2001.

[29] S. Lee and M. H. Hayes, "Efficient scene segmentation for content-based indexing in the compressed domain," in *Proc. IEEE Workshop Multimedia Signal Processing*, Cannes, France, Oct. 3–5, 2001, pp. 473–478.

[30] B. T. Truong, S. Venkatesh, and C. Dorai, "Scene extraction in motion pictures," *IEEE Trans. Circuits Syst. Video Technol.: Special Issue Multimedia Content Description*, vol. 15, no. 1, pp. 5–15, 2003.

[31] N. Nitanda, M. Haseyama, and H. Kitajima, "An audio-scene cut detection method using fuzzy c-means algorithm for audio-visual indexing," in *Proc. 2004 Int. Symp. Circuits and Systems*, May 23–26, 2004, vol. 2, pp. 89–92.

Marios Kyperountas received the B.Sc. degree in electrical engineering in 2002 and the M.Sc. degree in electrical engineering in 2003, both from Florida Atlantic University, Boca Raton.

He was a Research Assistant with the FAU Imaging Technology Center from 2000 to 2003 where he worked on several high-resolution imaging R&D projects funded by NASA, DARPA, and the U.S. Navy. Currently, he is pursuing the Ph.D. degree at the Artificial Intelligence and Information Analysis Lab, Department of Informatics, Aristotle University of Thessaloniki, and is working as an Image Processing Engineer in Santa Barbara, CA. His research interests include high resolution and ultrasonic imaging, pattern recognition, DSP algorithms, and real-time video processing.

Mr. Kyperountas is a member of the Golden Key Honor Society, the Phi Kappa Phi Honor Society, and the Tau Beta Pi Engineering Honor Society.

Constantine Kotropoulos (SM'06) received the Diploma degree with honors in electrical engineering in 1988 and the Ph.D. degree in electrical and computer engineering in 1993, both from the Aristotle University of Thessaloniki.

Since 2002, he has been an Assistant Professor in the Department of Informatics, Aristotle University of Thessaloniki. From 1989 to 1993, he was a Research and Teaching Assistant in the Department of Electrical and Computer Engineering at the same university. In 1995, he joined the Department of Informatics at the Aristotle University of Thessaloniki as a Senior Researcher and was then a Lecturer from 1997 to 2001. He has also conducted research in the Signal Processing Laboratory, Tampere University of Technology, Finland, during the summer of 1993. He has authored 25 journal papers, 131 conference papers, and contributed five chapters to edited books in his areas of expertise. He is co-editor of the book *Nonlinear Model-Based Image/Video Processing and Analysis* (New York: Wiley, 2001). His current research interests include speech, audio, and language processing; signal processing; pattern recognition; multimedia information retrieval; biometric authentication techniques, and human-centered multimodal computer interaction.

Dr. Kotropoulos was a scholar of the State Scholarship Foundation of Greece and the Bodossaki Foundation. He is a member of EURASIP, IAPR, ISCA, and the Technical Chamber of Greece.

Ioannis Pitas (SM'94) received the Diploma in electrical engineering in 1980 and the Ph.D. degree in electrical engineering in 1985, both from the University of Thessaloniki, Greece.

Since 1994 he has been a Professor at the Department of Informatics, University of Thessaloniki. From 1980 to 1993, he served as Scientific Assistant, Lecturer, Assistant Professor, and Associate Professor in the Department of Electrical and Computer Engineering at the same university. He served as Visiting Professor and ASI fellow at the University of British Columbia, Canada, as Visiting Professor at Ecole Polytechnique Federale de Lausanne, at Tampere University of Technology, Finland, as Visiting Assistant Professor at the University of Toronto, Canada, and as a Visiting Research Associate at the University of Toronto and the University of Erlangen-Nuernberg, Germany. He has published over 510 papers, contributed in 20 books, and authored, co-authored, edited, or co-edited seven books in his area of interest. His current interests are in the areas of digital image processing, multimedia signal processing, multidimensional signal processing, and computer vision.

Dr. Pitas has given 24 invited lectures, was member of the program committee of more than 115 scientific conferences and workshops and was chair of more than 35 conference sessions. He is/was Associate Editor of the IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS, IEEE TRANSACTIONS ON NEURAL NETWORKS, IEEE TRANSACTIONS ON IMAGE PROCESSING, *IJIG*, *IEICE, Circuits Systems and Signal Processing (CSSP)*, co-editor of *Multidimensional Systems and Signal Processing*, member of the editorial board of six journals and guest editor in six special journal issues. He is member of the National Research Council of Greece.