Audio-assisted movie dialogue detection

Margarita Kotti, Dimitrios Ververidis, Georgios Evangelopoulos, Ioannis Panagakis, Constantine Kotropoulos *Senior Member, IEEE*, Petros Maragos *Fellow Member, IEEE*, Ioannis Pitas *Fellow Member, IEEE*

***Abstract*—An audio-assisted system is investigated that detects if a movie scene is a dialogue or not. The system is based on actor indicator functions. That is, functions which define if an actor speaks at a certain time instant. In particular, the cross-correlation and the magnitude of the corresponding the cross-power spectral density of a pair of indicator functions are input to various classifiers, such as voted perceptrons, radial basis function networks, random trees, and support vector machines for dialogue/non-dialogue detection. To boost classifier efficiency AdaBoost is also exploited. The aforementioned classifiers are trained using ground truth indicator functions determined by human annotators for 41 dialogue and another 20 non-dialogue audio instances. For testing, actual indicator functions are derived by applying audio activity detection and actor clustering to audio recordings. 23 instances are randomly chosen among the aforementioned 41 dialogue instances, 17 of which correspond to dialogue scenes and 6 to non-dialogue ones. Accuracy ranging between 0.739 and 0.826 is reported.**

***Index Terms*—Dialogue detection; Indicator functions; Audio activity detection; Speaker clustering; Cross-correlation; Cross-power spectral density.**

I. Introduction

Movies constitute a large sector of the entertainment industry as over 9.000 hours of video are released every year [1]. Semantic content-based video indexing offers a promising solution for efficient digital movie management. Event analysis in movies is of paramount importance as it aims at obtaining a structured organization of the movie content and understanding its embedded semantics as humans do. A movie has some basic scene types, such as dialogues, stories, actions, and generic. Movie dialogue detection is the task of determining whether a scene derived from a movie is a dialogue or not. Movie dialogue detection is a challenging problem within movie event analysis, since there are no limitations on the emotional state of persons, the rate at which scenes interchange, the duration of silent periods, and the volume of background noise or music. For example, the detection of dialogue scenes in a movie is more complicated than detecting changes between anchor persons in TV-news, since many different scene types are incorporated in movies depending on the movie director [2]. Dialogue detection in conjunction with face and/or speaker identification could locate the scenes, where two or more particular persons are conversing. Furthermore, the statistics of dialogue scene durations may give a rough idea about the movie genre.

Although dialogues constitute the basic sentences of a movie, there is no commonly accepted definition for them. A broad definition of a dialogue scene is a set of consecutive shots, which contain conversations of people [1]. Conversations are assumed to include significant interaction between the persons, e.g a passing "hello" between two persons does not qualify as a dialogue. It is possible some audio segments are included in a dialogue scene, although they do not contain any conversation, due to their semantic coherence. For example, when two people are talking to each other, one should tolerate for short interruptions by a third person. However, such random effects should not affect dialogue detection. According to Chen [3], the elements of a dialogue scene are: the people, the conversation, and the location, where the dialogue is taking place. Recognizable dialogue acts are [4]: (i) Statements, (ii) Questions, (iii) Backchannels, (iv) Incomplete utterances, (v) Agreements, (vi) Appreciations. Repetition and periodicity are the main characteristics of a dialogue according to [5], [6]. Lehane states that dialogue detection is feasible, since there is usually an A-B-A-B structure in a 2-person dialogue [7]. An A-B-A-B-A-B structure is also employed in [5], [8]. Motivated by the just described assumptions, we consider that 4 actor changes should occur in order to declare a dialogue between actor A and actor B in a movie scene audio channel.

To the best of the authors' knowledge, movie dialogues have been mostly treated from the visual channel perspective (e.g. [3]), whereas the audio channel has been treated either as auxiliary or it is totally ignored. Recognizing a scene as a dialogue using exclusively the audio information has not been investigated, although significant information content exists in the audio channel, as is demonstrated in this paper. Indeed, it is usually possible to understand what is taking place by just listening to the sound and not resorting to visuals [1], although the reverse is not always true [7]. Moreover, audio information is faster to process than video information. Furthermore, combined audio-visual processing is more close to human perception. Audio-based dialogue detection can be used auxiliary to video-based dialogue detection and is proven to boost dialogue detection efficiency [3], [9], [10]. Related topics to dialogue detection are face detection and tracking, speaker tracking and speaker turn detection [12]. Aural information could also be exploited in various video analysis tasks, like video segmentation [11] or video classification [8], for example.

Margarita Kotti, Dimitrios Ververidis, Ioannis Panagakis, Constantine Kotropoulos, and Ioannis Pitas are with the Department of Informatics, Aristotle University of Thessaloniki, Thessaloniki 54124, Box 451, Greece, Tel: +30-2310-996361, Fax: +30-2310-998453, e-mail:{mkotti,jimver,panagakis,costas,pitas}@aiia.csd.auth.gr. Georgios Evangelopoulos and Petros Maragos are with the School of Electrical and Computer Engineering, National Technical University of Athens, 10682 Athens, Greece, Tel: +30-210-772-2964, Fax: +30-210-772-3397 e-mail: {gevag,maragos}@cs.ntua.gr This work is supported in part by European Commission 6th Framework Program with grant number FP6-507752 (MUSCLE Network of Excellence Project). M. Kotti was supported by the Propondis Public Welfare Foundation through a scholarship.

Among the three systems developed for dialogue detection in [9], we refer to the first system, that is based on audio and color information. Low-level audio features are extracted, such as zero crossing rate, silence ratio, and energy. Audio is classified into speech, music, and silence by means of support vector machines (SVMs). A finite state machine is used to detect a dialogue with precision being equal to 0.751 at recall equal to 0.955. By combining video information, the precision for dialogue detection equals 0.813 at recall 0.955.

Dialogue detection experiments have been performed using hidden Markov models (HMMs) in [1]. The audio component is analyzed to determine if it contains speech, silence, or music based. On the one hand, silence segments contain a quasi-stationary background noise with a low energy level with respect to signals belonging to other classes, making energy thresholding is sufficient. On the other hand, music segments contain a combination of sounds exhibiting high periodicity, which is exploited for their detection. To classify a scene, the audio classification is fused with a face detector and a location scene detector. Dialogue detection accuracy ranging from 0.71 to 0.99 is reported.

A top-down approach is adopted by Chen et al. [3]. Audio cues are derived by an SVM that differentiates among speech mixed with music, speech mixed with environmental background sound, and environment sound mixed with music. The following audio features are used: the variance of zero crossing rate, the silence ratio, and the harmonic ratio. Audio classification accuracy ranges from 0.6325 to 0.8594 depending on the features. Concerning dialogue detection, a finite state machine that incorporates the aforementioned audio cues is applied. The average precision using both audio and visual information equals 0.898, while the average recall is 0.936.

In [2], a multi-expert system performs dialogue detection. Three experts are employed, namely face detection, camera-motion estimation, and audio classification. A multi-layer perceptron performs dialogue classification for each expert. Audio classification categories are speech, music, silence, noise, speech with music, speech with noise, and music with noise. Physical features and perceptual ones are used for classification. In particular, the 14 physical features are related to energy, temporal energy variability, average and variance of the number of significant bands, sub-band centroid mean and variance, pause rate, and energy sub-band ratio. The remaining two perceptual features are based on pitch. The recognition rate equals 0.79 for the audio classification expert which discriminates among silence, speech, music, noise, speech with music, speech with noise, and music with noise. The achieved miss detection rate for dialogue detection for all experts equals 0.090, while the false alarm rate is 0.070.

Detection of monologues is discussed in [13]. A monologue is considered to occur at those shots, where speech and facial movements are synchronized. The audio channel is manually annotated as speech, music, silence, explosion, and traffic sounds. A Gaussian mixture model (GMM) is trained for each audio class and HMMs generate an N-best list for each audio frame and then the scores per shot are averaged. Monologue is detected through weighting speech, face and synchrony scores. The best monologue recall equals 0.88 at 0.30 precision.

Preliminary results on audio-assisted movie dialogue detection are described in [14] that resort to actor indicator functions. An actor indicator function defines if an actor speaks at a certain time instant. Ground truth indicator functions are used both for training and for testing. They are obtained manually by human annotators, who are listening to the audio recordings and provide their judgments on actor speech activity. The cross-correlation function of a pair of ground-truth indicator functions and the magnitude of the corresponding cross-power spectral density are fed as input to neural networks for dialogue detection. The average detection accuracy achieved ranges between 84.78% and 91.43%.

In this paper, a novel system for audio-assisted dialogue detection is proposed, that is depicted in Figure 1. Two types of indicator functions are employed: ground truth indicator functions and actual ones. Actual indicator functions are derived automatically after audio activity detection (AAD), that locates the boundaries of actor's speech within a noisy background followed by actor clustering aiming at grouping speech segments based actor characteristics. Dialogue decisions are provided by several classifiers, namely voted perceptrons (VPs), radial basis function (RBF) networks, random trees, and SVMs. The classifiers are fed by the cross-correlation sequence and the corresponding magnitude of the cross-power spectral density of a pair of indicator functions. To eliminate the impact of errors committed by AAD and/or actor clustering front-end in the classifier training, ground truth indicator functions are employed during training. However, actual indicator functions are used during testing. AdaBoost is also employed in order to enhance the performance of the aforementioned classifiers in a second stage. Experiments are carried out using the audio scenes extracted from 6 different movies of the MUSCLE movie database [15]. A total of 41 dialogue instances and another 20 non-dialogue instances are extracted. A high dialogue detection accuracy ranging between 0.739 and 0.826 is achieved enabling the use the proposed system in applications like movie classification, indexing, abstraction, annotation, retrieval, summarization, browsing, or searching. Although, the proposed system is tested on movie audio recordings, it is applicable to broadcasts and meeting recordings as well.

The paper introduces several novelties. 1) The exploitation of the audio channel for dialogue detection is rarely met in the related literature. To the best of the authors' knowledge, this is one of the first attempts to exploit the audio channel exclusively. 2) In previous works, the audio channel is just segmented [1] and is not capable by itself to distinguish a dialogue. The most common segmentation is into speech, music, and silence [1], [9]. More complicated cases include speech, music, silence, music, noise, speech with music, speech with noise, and music with noise [2] or in speech mixed with music, speech mixed with environmental background sound, and environment sound mixed with music [3]. Dialogue occurs if there is pure speech or mixed speech in a scene [6]. 3) An advanced and robust AAD is used here to determine speech activity in an audio recording avoiding the need for audio segmentation and the AAD is combined with actor clustering in order to extract the actual indicator functions. 4) The actor

clustering is unsupervised. The number of actors is found automatically. 5) It is demonstrated that the cross-correlation and the magnitude of the cross-power spectral density of pairs of indicator functions are fairly robust, easily interpretable, and powerful features to conduct dialogue detection which is not always possible for low-level audio features. 6) Several classifiers with Random Trees used for the first time, and one meta-classifier (AdaBoost) are assessed for dialogue detection. AdaBoost accomplishes to improve performance of Random Trees and SVMs.

The remainder of the paper is as follows. In Section II, the approach for AAD is detailed. Actor clustering is described in Section III. Indicator functions are treated in Section IV, where the cross-correlation and cross-power spectral density, which are used as features for dialogue detection, are also described. In Section V, the database, the figures of merit, and the classification results are presented along with performance comparison and discussion. Finally, conclusions are drawn in Section VI.

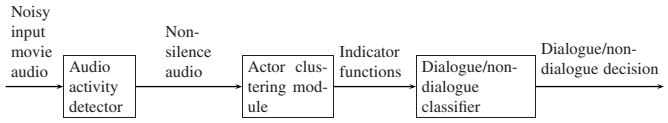


Fig. 1. The block diagram of the proposed system.

II. Audio Activity Detection

The need to differentiate between speech and noise has been recognized in previous studies [3], [9]. Voice activity detection (VAD) is a special case of the more general problem of speech segmentation and event detection. It is currently used in processing large speech databases, speech enhancement and noise reduction, frame dropping for efficient front-ends, echo cancellation, energy normalization, silence compression and selective power-reserving transmission. A VAD system performs a rough classification of input signal frames based on feature estimation in two classes: speech activity and non-speech events (pauses, silence, or background noise) [16], [17]. The interested reader is referred to [16], [17] for a discussion on recent approaches to VAD. Here, the algorithm proposed in [17] is applied for VAD in order to extract the meaningful, speech-containing movie audio segments from the input audio recording. The system is based on a modulation model for speech signals motivated by physical observations during speech production [18], the microproperties of speech signals, and a detection-theoretic optimality criterion. The features involved in the decision process have been previously used with success for speech endpoint detection in isolated word and sentences, VAD in large-scale databases and audio saliency modeling [19]. Moreover the developed VAD, based on divergence measures has been systematically compared in [17] with recent, high detection rate VAD [16], which in turn was evaluated against common standards. In the following, a system designed for speech-silence classification, that performs satisfactorily AAD, since the audio recordings may contain music, sound effects, or environmental sounds, is

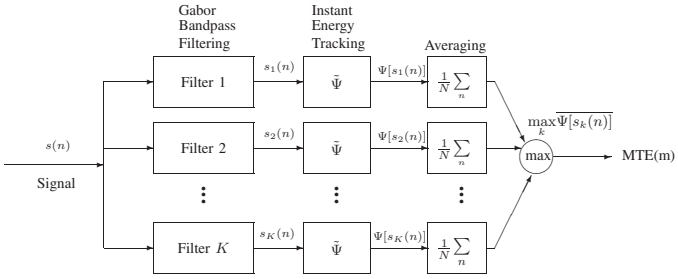


Fig. 2. Multiband filtering and modulation energy tracking for the *maximum average Teager energy* (MTE) audio representation.

described. The system provides an audio existence indicator at its output. The audio extracted after AAD is speech often mixed with music or environmental background noise [3].

According to the amplitude modulation - frequency modulation (AM-FM) model, a wideband audio signal is modeled by a sum of narrowband amplitude and frequency varying, non-stationary sinusoids $s(t) = \sum_{k=1}^{K} a_k(t)\cos\left(\varphi_k(t)\right)$, with time varying amplitude envelope $a_k(t)$ and instantaneous frequency $\omega_k(t) = d\varphi_k(t)/dt$ signals. Bandpass filtering decomposes the signal in frequency bands, each assumed to be dominated by a single AM-FM component in that frequency range [20]. This process of frequency-domain component separation is applied through a filterbank of K linearly-spaced Gabor filters $g_k\left(t\right) = \exp(-\alpha_k^2 t^2)\cos(\omega_{ck} t)$, with ω_{ck} the central filter frequency and α_k its root-mean square (rms) bandwidth. The filters globally separate modulation components assuming a priori a fixed component configuration, while simultaneously suppress the noise present in the wideband signal. To model a discrete-time audio signal $s[n] = s(nT)$, we use K discrete AM-FM components.

For discrete-time AM-FM signals $s[n]$, a direct approach is to apply the discrete-time Teager -Kaiser operator $\tilde{\Psi}[s[n]] = s^2[n] - s[n-1]s[n+1]$. The energy separation algorithm [18], can be further applied for demodulation by separating the instantaneous energy into its amplitude and frequency components. Assume $s[n]$ is a noisy, discrete time audio signal. A short-time representation in terms of a single component per analysis frame emerges by maximizing an energy criterion in the multi-dimensional filter response space [17], [20]. For each analysis frame m of N samples duration, the dominant modulation component is the one with *maximum average Teager energy* (MTE):

$$MTE[m] = \max_{1\le k\le K} \frac{1}{N} \sum_{n=(m-1)N+1}^{mN} \tilde{\Psi}((s * g_k)[n]), \qquad (1)$$

where $*$ denotes convolution and g_k the impulse response of the kth Gabor filter. The dominant component is the most salient signal modulation structure and energy. MTE may be thought of as the dominant signal *modulation energy*, capturing the joint amplitude-frequency information inherent in speech activity. The process of MTE derivation is detailed in the block diagram of Figure 2.

The algorithm for AAD is based on MTE measurements, adaptive thresholds, and noise estimation update. The signal is frame-processed and the *Multiband Teager Energy Divergence*

(MTED) estimates the divergence of MTE of an incoming frame with respect to its value for the background noise (MTEW):

$$\text{MTED}[m] = 10\log_{10}\left(\text{MTE}[m]/\text{MTEW}\right). \qquad (2)$$

Classification in speech (or audio) and silence is performed by comparing this level difference in dB from background noise to an adaptive threshold $\gamma \in [\gamma_0, \gamma_1]$: $\gamma = \gamma_0 + (\gamma_1 - \gamma_0)(E - E_0)/(E_1 - E_0)$, where E the background noise energy and the threshold interval boundaries depend on the cleanest E_0 and noisiest E_1 energies, computed during the initialization period from the database under consideration. Thus, it is assumed that the system will work in different noisy conditions.

The noise characteristics MTEW are learned during a short initialization period, assumed to be non-speech, and adapted whenever silence or pause is detected, by averaging in a small frame neighborhood. If $MTED[m] > \gamma$, then frame m is labeled as speech. A hang-over scheme is otherwise applied that delays the speech to non-speech transition in order to prevent low-energy word endings being misclassified as silence. Such a scheme considers the previous observations of a first-order Markov process modeling speech occurrences and is found to be beneficial to maintain a high accuracy detecting speech periods at low signal-to-noise ratio levels.

For the implementations herein the analysis frame is set to 20 ms, with 10 ms shifts and a 25 Gabor filterbank was used for narrowband component separation. In Figure 3, an example of the proposed AAD for a movie audio recording is shown with the resulting audio-presence indicator function superimposed. More details on the algorithm can be found in [17].

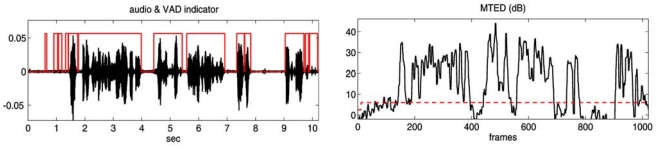


Fig. 3. Audio indicator using AAD. The audio recording from 'Jackie Brown' (left) is submitted to MTED-based (right) two-class classification in order to extract the non-silent audio segments.

III. Actor Clustering

A review on speaker clustering approaches can be found in [21]. The proposed approach is an unsupervised one. Unsupervised approaches are distance-based approaches, that rely mainly on speaker turn point detection to find if two neighboring long-segments stem from the same speaker [22], [23]. The length of the long-segment is user-defined. It should not be too short, because it causes erroneous estimation of the GMM parameters, nor too long, because it may result to a missed speaker turn point. Speaker turn point detection algorithms suffer by high false alarm rates due to their dependency on the linguistic content, because they use MFCCs. Distances or log likelihood ratios between GMMs, penalized by an information criterion such as the Bayesian one (BIC), are often used to find whether two successive frames stem from the same speaker [22], [24]. The disadvantages of such approaches are the convergence of the BIC criterion to local optima of the log likelihood ratio, and the execution delay due to GMM estimation for each long-segment of the audio recording. The proposed approach relies on the assumption that if two actors exist, then they would have significant different fundamental frequency and energy below 150 Hz regions, i.e. one actor would tend to be bass and the other will tend to be soprano. The approach is not so computationally demanding as the aforementioned approaches are. It requires about 4 s to converge for an audio recording of 1 min length in a PC at 3 GHz with 1 GB RAM at 400 MHz using Matlab 7.5.

In order to derive actual indicator functions, actor clustering is applied to the non-silence audio recordings extracted by AAD. The goal is to find whether one actor or two different actors are present in the recording. Furthermore, if the hypothesis of two actors holds, we wish to know when each actor speaks. We shall processes speech on the basis of short-term frames having duration of 20 ms, denoted as s_m. $\mathcal{S} = \{s_m\}_{m=1}^{N}$ be the set of the non-silence frames of an audio recording. Let also $h_q(s_m)$ be the probability of s_m belongs to qth actor, where $q = 1, 2, \ldots, Q$. Since the maximum number of actors in the audio recordings is 2, the maximum value allowed for Q is 2. The actor clustering module is shown in Figure 4.

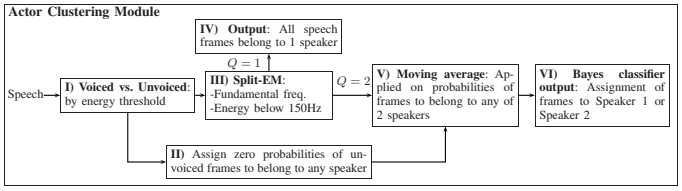


Fig. 4. The actor clustering module that gives attention to the voiced frames for speech clustering.

In Stage I, speech is classified into voiced or unvoiced frames by applying a heuristic algorithm that it is based on energy. The frame with energy content greater than 10% of the maximum energy of 200 successive frames is declared as voiced frame. The large window of 200 successive frames is shifted without overlap. This algorithm detects the voiced frames wit high precision and medium recall. This is important, because actor clustering is based on the voiced frames, as it is difficult for one to identify an actor by processing unvoiced speech. Let $\mathcal{S} = \mathcal{V} \cup \mathcal{N}$, be the division of the speech frames set to a voiced and an unvoiced set, respectively. In Stage II, $h_q(s_m \in \mathcal{N}) := 0$, i.e. the probability of unvoiced frames s_m belong either to either actor $q = 1, 2$ is set equal to zero.

Stage III resorts to a modification of the expectation-maximization algorithm [25]. The approach applies multivariate statistical tests so as to split a non-Gaussian cluster to Q Gaussian ones, where each Gaussian cluster corresponds to an actor. Throughout this paper, the clustering algorithm will be referred as Split-EM. Let $s_m = \{\mathbf{x}_m, c_m\}$ with $\mathbf{x}_m$ being a sample measurement vector extracted from s_m, and $c_m = 1, 2, \ldots, Q$ being the predicted s_m label. Two sample measurements are extracted for each speech frame s_m. The

first is the fundamental frequency found by locating the index at the cepstrum peak. The second is the energy below 150 Hz, that is estimated from the 3 spectral coefficients measuring the energy content within the first three 50 Hz bands. Bass actors have a low fundamental frequency and large energy content below 150 Hz. The opposite holds for soprano actors. The application of the Split-EM leads to Gaussian components that model the two-dimensional probability density function (pdf) of the sample measurement vectors $\mathfrak{X} = \{\mathbf{x}_m\}_{i=1}^{N_x}$. For example, in Figure 5, the voiced speech frames of an audio recording are modeled by two Gaussian components. Then, frames are assigned to a component by the Bayes classifier. The number of components, Q, is found automatically with Split-EM algorithm. Besides Q, Split-EM returns the probabilities $h_q(s_m \in \mathcal{V})$.

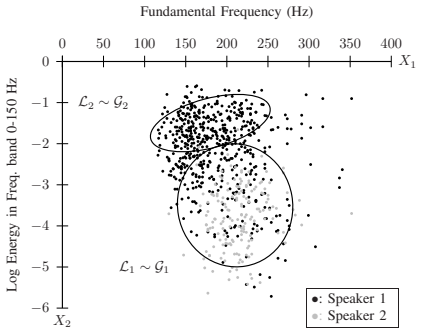


Fig. 5. Ellipses correspond to components found by Split-EM algorithm for the voiced speech frames. It can be seen that each component can be used as an actor conditional pdf. Therefore, frames can be assigned to actors by the Bayes classifier.

If Q equals 1, (e.g. Stage IV), then only one actor exists, and the algorithm stops. If $Q = 2$, then the probabilities $h_q(s_m)$ are smoothed by an average operator applied to 20 successive voiced and unvoiced frames with a shift of 1 frame. In this manner, unvoiced speech frames obtain probabilities to belong to an actor according to their neighboring voiced frames. Finally, in Stage VI, the Bayes classifier exploits probabilities $h_q(s_m)$ to assign frame s_m to qth actor.

The novel contributions of the proposed approach are 1) it is unsupervised, i.e. no training data are needed for each actor, 2) the number of actors is found by EM, and 3) the initialization of the GMM is accomplished through statistical tests in order to avoid local optima of the likelihood function during E- and M-steps.

IV. Actor indicator function processing

A. Indicator functions

Indicator functions are closely related to zero-one random variables used in the computation of expected values in order to derive the probabilities of events. Indicator functions are high-level features that can be easily compared to human annotations. Let us suppose that we know exactly when a particular actor (i.e. speaker) appears in an audio recording of N_s samples. Such information can be quantified by the indicator function of say actor A, $I_A[n]$, defined as:

$$I_A[n] = \begin{cases} 1, & \text{actor } A \text{ is present at sample } n \\ 0, & \text{otherwise.} \end{cases} \tag{3}$$

We shall confine ourselves to 2-person dialogues, without loss of generality. If the first actor is denoted by A and the second by B, their corresponding indicator functions are $I_A[n]$ and $I_B[n]$, respectively. For a dialogue scene the plot of ground indicator functions can be seen in Figure 6 (a). There are several alternatives to describe a dialogue scene. In 2-actor dialogues, the first actor rarely stops at sample n and the second actor starts at sample $n+1$. There might be audio frames corresponding to both actors. In addition, short silence periods should be tolerated. For an non-dialogue scene (i.e. a monologue), typical ground truth indicator functions are depicted in Figure 6 (b). $I_B[n]$ corresponds to short exclamations of the second actor. For comparison purposes, the actual indicator functions derived from the dialogue scene are shown in Figure 6 (c), and those for the non-dialogue scene are plotted in Figure 6 (d).

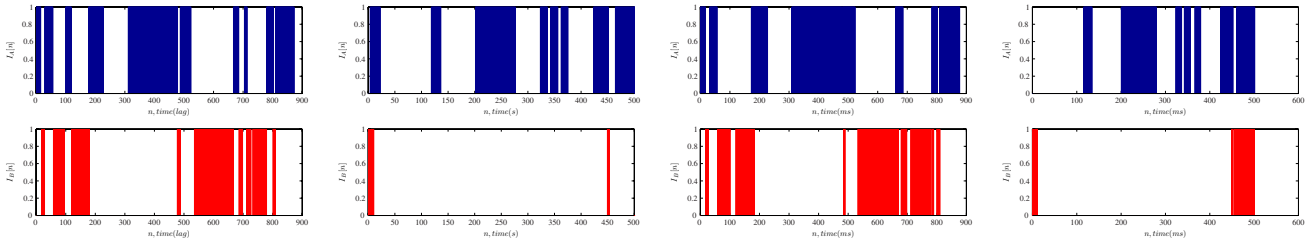
B. Cross-correlation and cross-power spectral density

The cross-correlation is widely used in pattern recognition. It is a common similarity measure between two signals [26]. It is used to find the linear relationship between two signals. The cross-correlation of a pair of indicator functions is defined by:

$$c_{AB}[l] = \begin{cases} \frac{1}{N_s} \sum_{n=1}^{N_s - l} I_A[n+l] I_B[n], & \text{when } 0 \le l \le N_s - 1 \\ c_{BA}[-l], & \text{when } -(N_s - 1) \le l \le 0 \end{cases} \tag{4}$$

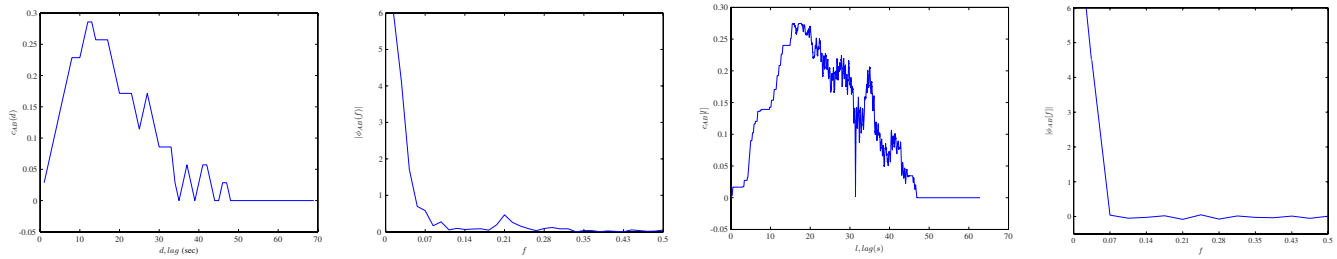
where l is the time-lag. In an ideal 2-person dialogue, the first indicator function is a train of rectangular pulses having a duration related to the average actor utterance separated by silent periods having a duration related also to average actor utterance. When the first actor is silent, the second actor speaks and accordingly between the indicator functions of two actors a shift between identical patterns is observed. Thus, dialogue is a repetitive, non-random pattern and the cross-correlation can be used detect those patterns. When the patterns of the two indicator functions match, the cross-correlation is maximized. The time-lag, where the cross-correlation of the two indicator functions is maximized is closely related the mean actor utterance duration. Significantly large values of the cross-correlation function indicate the presence of a dialogue. It can also be used to measure the overlap between two signals, because normally during a conversation there are samples where both actors speak simultaneously. Finally, the full cross-correlation sequence provides a detailed characterization of the dialogue pattern between any two actors. For the dialogue instance studied in Figure 6 (a) and 6 (c), the cross-correlation of the ground truth indicator functions is depicted in Figure 7 (a), whereas the corresponding cross-correlation of the actual indicator functions is plotted in Figure 7 (c).

Another useful notion to be exploited for dialogue detection is the discrete-time Fourier transform of the cross-correlation,



(a) (b) (c) (d)

Fig. 6. (a) Ground truth indicator functions of two actors in a dialogue scene. (b) Ground truth indicator functions of two actors in a non-dialogue scene (i.e. monologue). (c) Actual indicator functions of two actors for the dialogue scene in (a). (d) Actual indicator functions of two actors for the non-dialogue scene in (b).



(a) (b) (c) (d)

Fig. 7. (a) Cross-correlation of the ground truth indicator functions for the two actors in the dialogue scene of Figure 6 (a). (b) Magnitude of the cross-power spectral density when ground truth indicator functions for the two actors in the same dialogue scene are employed. (c) Cross-correlation in the same dialogue scene, when actual indicator functions are employed. (d) Magnitude of the cross-power spectral density in the same dialogue scene, when actual indicator functions are employed.

i.e. the cross-power spectral density [26]. The cross-power spectral density is defined as:

$$\phi_{AB}[f] = \sum_{l=-(N_s-1)}^{N_s-1} c_{AB}[l]\ \exp\left(-j2\pi\ f\ l\right) \qquad (5)$$

where $f \in [-0.5, 0.5]$ is the frequency in cycles per sampling interval. For negative frequencies, $\phi_{AB}[-f] = \phi^{\star}_{AB}[f]$, where * denotes complex conjugation. In audio processing experiments, the magnitude of the cross-power spectral density is commonly employed. The magnitude of the cross-power spectral density reveals the strength of the similarities between the two signals as a function of frequency. So, it shows which frequencies are related to strong similarities and which frequencies are related to weak similarities. When there is a dialogue, the area under $|\phi_{AB}[f]|$ is considerably large, whereas it admits a rather small value for a non-dialogue. Figure 7 (b) shows the magnitude of the cross-power spectral density derived from the dialogue instance under study, when ground truth indicator functions are used. Figure 7 (d) depicts the magnitude of the cross-power spectral density derived from the same audio recording, when actual indicator functions are used. For comparison purposes, Figure 8 (a) demonstrates the cross-correlation of ground truth indicator functions of the non-dialogue instance under study, whereas Figure 8 (b) shows the corresponding magnitude of the cross-power spectral density. Similarly, when actual indicator functions are used, the cross-correlation is plotted in Figure 8 (c) and the magnitude of the cross-power spectral density in Figure 8 (d). The differences between dialogue and non-dialogue cases are self-evident in both time and frequency domains.

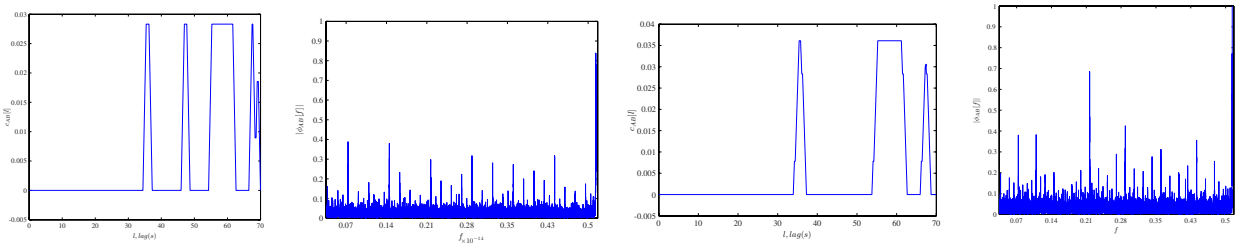
In preliminary experiments on dialogue detection, two values were only used, namely the value admitted by cross-correlation at zero lag $c_{AB}[0]$ and the cross-spectrum energy in the frequency band [0.065, 0.25] [27]. Both values were compared against properly set thresholds, derived by training, in order to detect dialogues. The interpretation of $c_{AB}[0]$ is straightforward, since it is the product of the two indicator functions. The greater the value of $c_{AB}[0]$ is, the longer time the two actors speak simultaneously. In this paper, we avoid dealing with scalar values, derived from the cross-correlation and the corresponding cross-power spectral density, allowing for a more generic approach.

V. Experimental Results

First, the database used is outlined in subsection V-A. Then, the figures of merit for performance assessment are defined in subsection V-B. Next, the classifiers are briefly described along with the corresponding experimental results in subsection V-C. Finally, performance comparison and discussion is made in subsections V-D and V-E, respectively.

A. Database

The MUSCLE movie database is used. The database contains dialogue and non-dialogue scenes for 6 movies, as indicated in Table I. There are multiple reasons justifying the choice of these movies. First of all, they are quite popular.



(a) (b) (c) (d)

Fig. 8. (a) Cross-correlation of the ground truth indicator functions for the two actors in the non-dialogue scene of Figure 6 (b). (b) Magnitude of the cross-power spectral density when ground truth indicator functions for the two actors in the same non-dialogue scene are employed. (c) Cross-correlation in the same non-dialogue scene, when actual indicator functions are employed. (d) Magnitude of the cross-power spectral density in the same non-dialogue scene, when actual indicator functions are employed.

Secondly, they cover a wide area of movie genres. For example, Analyze That is a comedy, Platoon is an action, and Cold Mountain is a drama. Finally, they have already been widely used in movie analysis experiments. The dialogue scenes refer to two-person dialogues. Examples of non-dialogue scenes include monologues, music soundtrack, songs, street noise, or instances where the first actor is talking and the second one is just making exclamations. The database is available on demand and it includes audio, visual, audiovisual, and text manifestations of dialogue and non-dialogue scenes. In addition, all scenes are fully annotated by human agents [15].

TABLE I
THE 6 MOVIES IN MUSCLE MOVIE DATABASE.

Movie name	**Dialogue scenes**	**Non-dialogue scenes**	**Total scenes**
Analyze That	4	2	6
Cold Mountain	5	1	6
Jackie Brown	3	3	6
Lord of the Rings I	5	3	8
Platoon	4	2	6
Secret Window	4	6	10
Total	25	17	42

In this paper, we explore the audio information only. In total, 42 scenes are extracted from the aforementioned movies, as can be seen in Table I. The audio track of these scenes is digitized in PCM at a sampling rate of 48 kHz and each sample is quantized in 16 bit two-channel.

To fix the number of inputs in the classifiers under study, a running time-window of 25 s duration is applied to each audio scene. The particular choice of the duration for the time window is justified in [14]. In short, after modeling the empirical distribution of the actor utterance duration, it is found that it is the Inverse Gaussian with expected value equal to 5 s. This means that actor changes are expected to occur, on average, every 5 s. We consider that four actor changes should occur within the time-window employed in our analysis on average. Accordingly, an A-B-A-B-A structure is assumed. Similar assumptions are also invoked in [3], [5]–[9]. As a result, an appropriate dialogue window should have a duration of $5\times(4+1)$ = 25 s. Non-dialogue events could exhibit A-A-A-A-A or a B-B-B-B-B structures, i.e. monologues. Another case of a non-dialogue is a scene where no actor talks, but there is background music or noise, e.g. an C-C-C-C-C structure is observed, where C stands for everything else but speech.

In the training phase, 61 instances are extracted by applying the 25 s window to the 42 audio scenes. 41 out of the 61 instances correspond to dialogue instances and the remaining 20 to non-dialogue ones. For a 25 s window and a sampling frequency of 1 Hz, 49 samples of $c_{AB}[l]$ and another 49 samples of $|\phi_{AB}[f]|$ are computed. The aforementioned 98 samples, plus the label, stating whether the instance is a dialogue or not, are fed as input to train the classifiers detailed in subsection V-C. In the test phase, 23 instances are randomly selected. 17 of them correspond to dialogues and 6 to non-dialogues. After AAD and actor clustering, 49 samples of $c_{AB}[l]$ and another 49 samples of $|\phi_{AB}[f]|$ are computed for each test instance. The aforementioned instances are used to assess the classifiers performance.

B. Figures of Merit

The most commonly used figures of merit for dialogue detection are described in this subsection, in order to enable a comparable performance assessment with other similar works. Let us call the correctly classified dialogue instances $hits_d$ and the correctly classified non-dialogue instances $hits_{nd}$. Then, $misses$ are the dialogue instances that are not classified correctly and $false\ alarms$ are non-dialogue instances classified as dialogue ones. Obviously, the total number of dialogue instances is equal to the sum of $hits_d$ plus $misses$.

Two sets of figures of merit are employed. The first set includes the rate of correctly classified instances, the rate of the incorrectly classified instances, the root mean square error, and the mean absolute error. The rate of correctly classified instances (CCI) and the rate of incorrectly found instances (ICI) is defined as [28]:

$$\begin{aligned} CCI &= \frac{hits_d+hits_{nd}}{hits_d+hits_{nd}+misses+false\ alarms}, \\ ICI &= \frac{misses+false\ alarms}{hits_d+hits_{nd}+misses+false\ alarms}. \end{aligned} \tag{6}$$

The root mean square error ($RMSE$) for the 2-class problem and the mean absolute error (MAE) are also defined as

[28]:

$$RMSE = \sqrt{ICI}, \quad MAE = \frac{hits_d + hits_{nd}}{misses + false\ alarms}. \tag{7}$$

The second set consists of precision (PRC), recall (RCL), and F_1 measure. For the dialogue instances, they are defined as [28]:

$$PRC_d = \frac{hits_d}{hits_d + false\ alarms}, \quad RCL_d = \frac{hits_d}{hits_d + misses}, \quad F_{1d} = \frac{2\ PRC_d \cdot\ RCL_d}{PRC_d + RCL_d} \tag{8}$$

For non-dialogue instances, the aforementioned figures of merit are as follows:

$$PRC_{nd} = \frac{hits_{nd}}{hits_{nd} + misses}, \quad RCL_{nd} = \frac{hits_{nd}}{hits_{nd} + false\ alarms}, \quad F_{1nd} = \frac{2\ PRC_{nd} \cdot\ RCL_{nd}}{PRC_{nd} + RCL_{nd}}. \tag{9}$$

F_1 measure admits a value between 0 and 1. The higher its value is, the better performance is obtained.

C. Classifiers

Several classifiers have been employed for audio-assisted movie dialogue detection. An ideal feature extraction method would require a trivial classifier, whereas an ideal classifier would not need a sophisticated feature extraction method. However, in practice neither an ideal feature extraction method nor an ideal classifier are available. Accordingly, a comparative study among various classifiers is necessary. The classifiers are trained on ground truth indicator functions and tested on actual indicator functions to assess their generalization ability. The following classifiers are tested: VPs, RBF networks, random trees, and SVMs. At a second stage, the AdaBoost meta-classifier is applied to improve the performance of the aforementioned classifiers.

1) Voted Perceptrons: VPs operate in a higher dimensional space using kernel functions. In VPs, the algorithm takes advantage of data that are linearly separable with large margins [29]. VP also utilizes the leave-one-out method. For the marginal case of one epoch, VP is equivalent to multilineal perceptron. The main expectation underlying VP, is that data are more likely to be linearly separable into higher dimension spaces. VP is easy to implement and also saves computation time. VP exponent is set equal to 1.0. Dialogue detection results using VPs are enlisted in the second column of Table II.

2) Radial basis function networks: In classification problems, the RBF network output layer is typically a sigmoid function of a linear combination of hidden layer values representing the posterior probability. RBF networks apply linear mapping from hidden layer to output layer, which is adjusted in the learning process. In classification problems, the fixed non-linearity introduced by the sigmoid output function, is most efficiently dealt with iterated reweighed least squares [30]. RBF networks have also shown approximation capabilities. A normalized Gaussian RBF network is used. The k-means clustering algorithm is used to provide the basis functions, while the logistic regression model is employed for learning. Symmetric multivariate Gaussians fit the data of each cluster. All features are standardized to zero mean and unit variance. Dialogue detection results using the RBF network are summarized in the third column of Table II.

TABLE II
FIGURES OF MERIT FOR DIALOGUE/NON-DIALOGUE DETECTION USING VPS, RBF NETWORKS, RANDOM TREES, AND SVMS TRAINED ON GROUND TRUTH INDICATOR FUNCTIONS AND TESTED ON ACTUAL INDICATOR FUNCTIONS.

	VPs	RBF networks	Random Trees	SVMs
CCI	0.826	0.826	0.783	0.739
$RMSE$	0.417	0.417	0.447	0.511
MAE	0.174	0.174	0.224	0.261
PRC_d	0.933	0.933	0.929	0.923
RCL_d	0.824	0.824	0.765	0.706
F_{1d}	0.875	0.875	0.839	0.8
PRC_{nd}	0.625	0.625	0.556	0.5
RCL_{nd}	0.833	0.833	0.833	0.833
F_{1nd}	0.714	0.714	0.667	0.625

TABLE III
FIGURES OF MERIT FOR DIALOGUE/NON-DIALOGUE DETECTION USING ADABOOST ON VPS, RBF NETWORKS, RANDOM TREES, AND SVMS TRAINED ON GROUND TRUTH INDICATOR FUNCTIONS AND TESTED ON ACTUAL INDICATOR FUNCTIONS.

AdaBoost				
	VPs	RBF networks	Random Trees	SVMs
CCI	0.826	0.826	0.826	0.783
$RMSE$	0.417	0.406	0.406	0.481
MAE	0.174	0.215	0.215	0.309
PRC_d	0.933	0.933	0.933	1
RCL_d	0.824	0.824	0.824	0.706
F_{1d}	0.875	0.875	0.875	0.828
PRC_{nd}	0.625	0.625	0.625	0.545
RCL_{nd}	0.833	0.833	0.833	1
F_{1nd}	0.714	0.714	0.714	0.706

3) Random Trees: Random trees mimic natural evolution [31]. They are also suitable to encode any form of information, that is successively replicated over time and transmitted with occasional errors. This attribute yields random trees suitable for the application under consideration, since dialogues contain actor changes that are replicated and sporadic errors can be attributed to erroneous indicator functions that are derived by AAD and actor clustering. In this paper, random trees with 1 random feature at each node are applied. No pruning is performed. The results using random trees are summarized in the fourth column of Table II.

4) Support Vector Machines: SVMs are supervised learning methods that can be applied either to classification or regression. SVMs take a different approach to avoid overfitting by finding the maximum-margin hyperplane. In dialogue detection experiments performed, the sequential minimal optimization algorithm is used for training the support vector classifier [32]. In this paper, we deal with a two-class problem. The linear kernel is employed. The experimental results are detailed in the fifth column of Table II.

5) AdaBoost: AdaBoost is a meta-classifier for constructing a strong classifier as linear combination of simple weak classifiers [33]. It is adaptive in the sense that subsequently built classifiers are tweaked in favor of those instances misclassified by previous classifiers. The biggest drawback of AdaBoost is its sensitivity to noisy data and outliers. Otherwise, it has a better generalization performance than most learning algorithms. In this paper, the AdaBoost algorithm is used to build a strong classifier based on VPs, the RBF network, the random trees, and the SVM classifier. Dialogue detection results using the AdaBoost algorithm for VP, RBF networks, random trees, and the SVM classifier are shown in Tables III. The results are reported for 10 iterations of AdaBoost.

D. Performance comparison

Regarding the classification performance of the aforementioned classifiers, the best results are obtained by the VPs and the RBF networks. The worst performance is achieved

by SVMs. We suspect that the number of training instances is not sufficient for SVMs to take advantage of feature statistics. However, it is worth mentioning that SVM performance is improved after applying AdaBoost. In fact, SVM is the most favored classifier from AdaBoost. The relative CCI improvement equals 6%. However, SVM performance, even after boosting remains considerably low, indicating that SVM is not suitable for this particular dialogue/non-dialogue detection problem. AdaBoost also manages to enhance the performance of random trees and boost it to the same level of VPs and RBF networks performance. Accordingly, AdaBoost is appropriate for the dialogue/non-dialogue detection problem.

E. Discussion

Since the dialogue detection system is fully automated, it is worth looking into its performance when the processed recordings are far from being ideal. Two extreme cases are considered. Scenes with high background noise/music and scenes, where one or both actors increase their volume suddenly. Let us consider first the background noise/music. The actor clustering algorithm has a mean cluster accuracy of 0.908, when there is no or little background noise/music. The corresponding accuracy is 0.905, when there is medium background noise/music, while it drops to 0.747 when the background noise/music is high. If there is a sudden increase in the volume of both actors, (e.g. when they strongly argue), the mean actor clustering accuracy is 0.732. However, when only one actor increases his/her volume, the corresponding actor clustering accuracy equals 0.888. When both of them increase their volume successively, actor clustering accuracy drops to 0.388. If the conversation is calm, (e.g. there is no increase in actors' volume), the actor clustering accuracy equals 0.910.

However, even when actor clustering is not perfect the tested classifiers manage to compensate for the resulted erroneous indicator functions. In the presence of high background noise/music, SVMs and random trees face a greater difficulty to classify dialogues correctly. About 60% of the dialogues that exhibit high background noise/music are correctly classified by both the SVMs and the random trees. When there is a sudden successive increase in both actors' volume, SVMs exhibit the poorest performance. About 45% of dialogues where both actors increase their volume successively are misclassified. Poor SVM performance can be attributed to the fact that an SVM optimizes generalization for the worst case. Random trees degraded performance is due to slight variations in the training data which can cause different attribute selections at each choice point within the tree.

The performance of dialogue detection of the proposed system is compared to the performance of a system that uses the ground truth indicator functions in both the training and the test phases [14]. In [14], two splits of the ground truth indicator functions between the training and the test set are examined, namely the 70%/30% training/test split and the 50%/50% training/test split. Concerning the RBF networks, for the 70%/30% split CCI is 0.872, while for the 50%/50% split CCI is 0.848. The relative performance drop is 5.28% and 2.57%, respectively. When AdaBoost is applied to RBF networks, CCI is 0.864 for the 70%/30% and 0.871 for the 50%/50% split. That is, a relative deterioration of 4.46% and 5.15% between the CCI reported in [14] and that of AdaBoost on RBF networks is reported in this paper. A similar deterioration is observed for VPs and SVMs. As expected, when error-free ground truth indicator functions are used, the reported performance is better than that reported here. Errors in actual indicator functions may be due to AAD errors or actor clustering deficiencies. In any case, the dialogue detection accuracy still remains high justifying its use in movie indexing, browsing, navigation, abstraction, annotation, search and retrieval.

A rough comparison between the reported performance here and that of related past works is attempted next. However, a fair comparison is not feasible due to the following reasons: 1) Aural information is used to enhance video dialogue detection results in the majority of previous works. Thus, when fusion of aural and video information is made, the results are obviously improved [3]. 2) The databases used are not always of the same nature. 3) The definition of a dialogue is not unique in the research community. 4) Researchers do not employ the same figures of merit nor the same experimental protocol, which prevents direct comparisons.

Three systems are developed by Lehane et al. for detection dialogues in movies: the first system is based on audio and color information, the second on video and color information, and the third combines results of both the first and the second system [9]. The average dialogue detection precision equals 0.751 and the average recall equals 0.955 for the first system. So the corresponding F_{1d} is 0.841. For the third system, a precision of 0.813 for dialogue detection at a corresponding recall of 0.965 is reported. Accordingly, F_{1d} is 0.882 for the third system. Our best F_{1d} equals 0.875 for VPs and RBF networks with or without AdaBoost as well as for random trees after AdaBoost. Our reported F_{1d} is higher than that of the first system, but it is inferior than the F_{1d} of the third system. However, it should be noted that video information is exploited in the third system [9].

Alatan et al. tested both circular and left-to-right topologies [1]. MPEG-7 Test dataset is used for evaluation. Mean accuracy for the left-to-right HMM is 0.963, while for the circular HMM accuracy equals 0.823. Our best achieved CCI is 0.826, that is favorably compared to circular HMM accuracy, but it is inferior to left-to-right HMM accuracy. However, one should bear in mind than the dataset in [1] consisted of two sitcoms and a movie making the nature of the dataset different than that of the MUSCLE movie database.

Chen et al. apply a finite state machine model to extract simple dialogue or action scenes from two movies [3]. The best performance is achieved when video information is coupled with audio cues. In this case, dialogue detection precision equals 0.835 at dialogue detection recall 1. The corresponding F_{1d} is 0.91. The best F_{1d} achieved by the proposed system equals 0.875 for VPs and RBF networks with or without AdaBoost as well as for random trees after AdaBoost. Nevertheless, one should keep in mind that in [3] audio and video information is fused.

De Santo et al. applied multiple experts for dialogue/non-

dialogue detection [2]. The applied database consisted of movie audio and video tracks. When aggregating the video and the audio information, the false alarm rate equals 0.090, while the miss detection rate equals 0.070. However, false alarm and miss detection rates are defined in a different way than in this paper. In [2], a dialogue/non-dialogue scene is detected correctly, when it overlaps with the true scene by 50% of the time at least.

VI. Conclusions

In this paper, a system for audio dialogue detection in movies was proposed that integrates audio activity detection based on the multiband teager energy divergence and actor clustering based on GMM modeling by a variant of the expectation-maximization algorithm to derive actual indicator functions. The cross-correlation sequence of a pair of indicator functions and the corresponding magnitude of the cross-power spectral density are fed as features to various classifiers tested for dialogue/non-dialogue detection, namely VPs, RBF networks, random trees, and SVMs. The aforementioned classifiers are trained using ground truth indicator functions. Audio scenes were extracted from 6 movies. Furthermore, a multitude of commonly employed objective figures of merit are used to assess the classifier performance in order to facilitate future comparisons. The best accuracy reported was 0.826 for VPs and RBF networks. AdaBoost has demonstrated to improve the efficiency of random trees and SVMs efficiency at a second stage.

References

[1] A. A. Alatan, A. N. Akansu, and W. Wolf, "Comparative analysis of hidden Markov models for multi-modal dialogue scene indexing", in Proc. *2000 IEEE Int. Conf. Acoustics, Speech, and Signal Processing*, vol. 4, pp. 2401-2404, 2000.

[2] M. De Santo, G Percannella, C. Sansone, and M. Vento, "Dialogue scenes detection in MPEG movies: A multi-expert approach", in *Lecture Notes in Computer Science*, vol. 2184, no. 5, pp. 192-201, September 2001.

[3] L. Chen, S. J. Rizvi, and M. T. Ozsu , "Incorporating audio cues into dialog and action scene extraction", in Proc. *Storage and Retrieval for Media Databases*, vol. 5021, pp. 252-263, January 2003.

[4] P. Král, C. Cerisara, and J. Kleckova, "Combination of classifiers for automatic recognition of dialogue acts", in Proc. *9th European Conf. Speech Communication and Technology*, pp. 825-828, 2005.

[5] H. Sundaram and S.-F. Chang, "Computable scenes and structures in films", *IEEE Trans. Multimedia*, vol. 4, no. 4, pp. 482- 491, 2002.

[6] Y. Li, S. Narayanan, and C. -C. J. Kuo, "Content-based movie analysis and indexing based on audiovisual cues", *IEEE Trans. Circuits and Systems for Video Technology*, vol. 14, no. 8, pp. 1073- 1085, 2004.

[7] B. Lehane, N. O' Connor, and N. Murphy, "Dialogue scene detection in movies using low and mid-level visual features", in Proc. *Int. Conf. Image and Video Retrieval*, pp. 286-296, 2005.

[8] Y. Wang, Z. Liu and J. C. Huang, "Multimedia content analysis-using both audio and visual clues", *IEEE Signal Processing Magazine*, vol. 17, no. 6, pp. 12-36, 2000.

[9] B. Lehane, N. O' Connor, and N. Murphy, "Dialogue sequence detection in movies", in Proc. *4th Int. Conf. Image And Video Retrieval*, vol. 3568, pp. 286-296, July 2005.

[10] E. Benetos, S. Siatras, C. Kotropoulos, N. Nikolaidis, and I. Pitas, "Movie analysis with emphasis to dialogue and action scene detection", in P. Maragos, A. Potamianos, and P. Gros (Eds.), *Multimodal Processing and Interaction Audio, Video, Text*, N. Y.: Springer, 2008.

[11] M. Kyperountas, C. Kotropoulos, and I. Pitas , "Enhanced eigen-audioframes for audiovisual scene change detection", *IEEE Trans. Multimedia*, vol. 9, no. 4, pp. 785-797, 2007.

[12] M. Kotti, E. Benetos, and C. Kotropoulos, "Automatic speaker change detection with the Bayesian information criterion using MPEG-7 features and a fusion scheme", in Proc. *2006 IEEE Int. Symp. Circuits and Systems*, pp. 1856-1859, May 2006.

[13] G. Iyengal, H. J. Nock, and C. Neti, "Audio-visual synchrony for detection of monologues in video archives", in Proc. *2003 IEEE Int. Conf. Acoustics, Speech, and Signal Processing,* vol. 1, pp. 329-332, April 2003.

[14] M. Kotti, E. Benetos, C. Kotropoulos, and I. Pitas, "A neural network approach to audio-assisted movie dialogue detection", *Neurocomputing, Special Issue: Advances in Neural Networks*, vol. 71, no. 1-3, pp. 157-166, December 2007.

[15] D. Spachos, A. Zlatintsi, V. Moschou, P. Antonopoulos, E. Benetos, M. Kotti, K. Tzimouli, C. Kotropoulos, N. Nikolaidis, P. Maragos, and I. Pitas, "MUSCLE movie database: A multimodal corpus with rich annotation for dialogue and saliency detection", in Proc. *6th Int. Conf. Language Resources and Evaluation*, Marrakech, Morocco, May 2008. Available on-line http://poseidon.csd.auth.gr/EN/MUSCLE_moviedb/index.php

[16] J. Ramirez, J. C. Segura, C. Benitez, A. de la Torre, and A.J. Rubio, "Efficient voice activity detection algorithms using long-term speech information", *Speech Communication*, vol. 42, no. 3-4, pp. 271-287, April 2004.

[17] G. Evangelopoulos and P. Maragos, "Multiband modulation energy tracking for noisy speech detection", *IEEE Trans. Audio, Speech, and Language Proccessing*, vol. 14, no. 6, pp. 2024-2038, July 2006.

[18] P. Maragos, J. F. Kaiser, and T. F. Quatieri, "Energy separation in signal modulations with application to speech analysis", *IEEE Trans. Signal Proccessing*, vol. 41, no. 10, pp. 3024-3051, October 1993.

[19] G. Evangelopoulos, K. Rapantzikos, P. Maragos, Y. Avrithis, and A. Potamianos, "Multimodal processing and interaction: audio, video, text", in n P. Maragos, A. Potamianos, and P. Gros (Eds.), *Audiovisual attention modeling and salient event detection* N. Y.: Springer, 2008.

[20] A. C. Bovik, P. Maragos, and T. F. Quatieri, "AM-FM energy detection and separation in noise using multiband energy operators", *IEEE Trans. Signal Proccessing*, vol. 41, no. 12, pp. 3245-3265, December 1993.

[21] M. Kotti, V. Moschou, and C. Kotropoulos, "Speaker segmentation and clustering", *Signal Processing*, vol. 88, no. 5, pp. 1091-1124, May 2008.

[22] C. Barras, X. Zhu, S. Meignier, and J. L. Gauvain, "Multistage speaker diarization of broadcast news", *IEEE Trans. Audio, Speech and Language Processing*, vol. 14, no. 5, pp. 1505-1512, September 2006.

[23] B. Fergani, M. Davy, and A. Houacine, "Speaker diarization using one-class support vector machines", *Speech Communication*, vol. 50, no. 5, pp. 355-365, May 2008.

[24] J. Ajmera, I. McCowan, and H. Bourlard, "Robust speaker change detection", *IEEE Signal Processing Letters*, vol. 11, no. 8, pp. 649-651, August 2004.

[25] D. Ververidis and C. Kotropoulos, "Gaussian mixture modeling by exploiting the Mahalanobis distance", *IEEE Trans. Signal Proccessing*, vol. 56, no. 7, pp. 2797 - 2811, July 2008.

[26] P. Stoica and R. L. Moses, *Introduction to Spectral Analysis,* Upper Saddle River, N. J.: Prentice Hall, 1997.

[27] M. Kotti, C. Kotropoulos, B. Ziólko, I. Pitas, and V. Moschou, "A framework for dialogue detection in movies", in *Lecture Notes in Computer Science*, vol. 4105, pp. 371-378, Istanbul, September 2006.

[28] I. H. Witten and E. Frank, *Data Mining-Pracitcal Machine Learning Tools and Techniques* 2/*e*., MA: Morgan Kaufmann, 2000.

[29] Y. Freund and R. E. Schapire, "Large margin classification using the perceptron algorithm", *Machine Learning*, vol. 37, no. 3, pp. 277-296, 1999.

[30] T. Poggio and F. Girosi, "Networks for approximation and learning", *Proceedings of the IEEE*, vol. 78, no. 9, pp. 1481-1497, September 1990.

[31] L. Breiman, J. H. Friedman, R. A. Olshen, and C. J. Stone, *Classification and Regression Tress*, Wadsworth and Brooks, Pacific Grove, California, 1984.

[32] J. Platt, "Fast training of support vector machines using sequential minimal optimization", in B. Schoelkopf, C. Burges, and A. Smola, (Eds.), *Advances in Kernel Methods - Support Learning*, MIT Press, 1999.

[33] Y. Freund and R. E. Schapire, "A short introduction to boosting", *J. Japanese Society for Artificial Intelligence,* vol. 14, no. 5, pp. 771-780, September 1999.

Authors' photos and biographies are not available at the time of publication.