Efficient Speaker Change Detection Using Adapted Gaussian Mixture Models

Amit S. Malegaonkar, Aladdin M. Ariyaeeinia, and Perasiriyan Sivakumaran

***Abstract*—A new approach to speaker change detection is proposed and investigated. The method, which is based on a probabilistic framework, provides an effective means for tackling the problem posed by phonetic variation in high-resolution speaker change detection. Additionally, the approach incorporates the capability for dealing with undesired effects of variations in speech characteristics. Using the experimental investigations conduced with clean and broadcast news audio, it is shown that the proposed method is significantly more effective than the currently popular techniques for speaker change detection. To enhance the computational efficiency of the proposed method, modified implementation algorithms are introduced which are based on the exploitation of the redundant operations and a fast scoring procedure. It is shown that, through the use of the proposed fast algorithm, the computational efficiency of the approach can be increased by over 77% without significant reduction in its accuracy. The paper discusses the principles and characteristics of the proposed speaker change detection method, and provides a detailed description of its efficient implementation. The experiments, investigating the performance of the proposed method and its effectiveness in relation to other approaches, are described and an analysis of the results is presented.**

***Index Terms*—Bilateral scoring, phonetic heterogeneity, probabilistic approach.**

I. Introduction

SPEAKER change detection (SCD) can be defined as the process of determining the time indices of the points of speaker change in a given conversational audio stream. SCD has a range of applications in different areas including speaker tracking, improving the accuracy of speech recognition systems (via speaker normalization/adaptation), indexing audio recordings, and providing cues for scene/topic/program changes in multimedia applications. When there is no prior information about the identities of speakers present in the stream, the process is called unsupervised SCD.

The initial approach to this process involves sliding an analysis window through the audio stream and measuring the similarity between the adjacent subsets of the data within it at each window positioning. This is based on representing the data subsets with single density Gaussian models [1]. If the level of similarity is below a threshold, then a speaker change is registered. In that work, the generalized log-likelihood ratio is used as the similarity measure. Since then, various other measures have been investigated. These include the Kullback–Leibler symmetrical measure (KL-2) [2], Bhattacharyya measure [3], divergence measure [2], and distances derived from second-order statistics [3].

An alternative to the above approach is that of using the Bayesian information criterion (BIC) [4], [5]. This involves a statistical hypothesis test between the null hypothesis (no speaker change in the analysis window) and the alternative hypothesis (a speaker change in the analysis window). These hypotheses are tested based on modeling the data with single Gaussian densities. For the null hypothesis, a single Gaussian model is fitted to the data in the entire analysis window. For the alternative hypothesis, two single Gaussian models are fitted to the data subsets which share adjacency at the hypothesized point of speaker change. These hypotheses are then evaluated using a penalized likelihood. This has been the most dominant approach for speaker change detection in recent years. Its popularity is mainly due to its superior ability to detect various acoustic changes including speaker changes [2]–[5]. Further attempts to enhance the performance of this approach have involved using a combination of distance measures and BIC [2], deploying Gaussian mixture models (GMMs) with BIC [6] and using adapted single Gaussian models in the BIC framework [7].

A departure from the above statistical-based approaches to speaker changes detection is that of using support vector machines (SVMs) [8]. SVM is a nonlinear classifier which is based on the principle of structural risk minimization (SRM). The use of SVM involves first training the classifier with a number of training examples from two types of representative patterns. The first pattern is assigned a positive label and consists of a segment of data covering a speaker change. The second pattern is assigned a negative label and consists of a segment of data without any speaker changes. When trained with such example data sequences, the SVM finds the nonlinear multidimensional surface (boundary) in the hyper-space that can best distinguish between the two training patterns. For detecting speaker changes, a sliding window of fixed length is used to scan the speech data. The pretrained SVM assigns a label to the sliding window according to the pattern of data present in the window. It is expected that this classifier assigns a positive label when an actual speaker change is present. This approach is reported to work reasonably well for detecting speaker changes [8]. The main concerns in this approach are the high computational cost involved in the training and testing processes, and the require-

Manuscript received May 12, 2006; revised February 5, 2007. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Mary P. Harper.

A. S. Malegaonkar was with the School of Electronic, Communication and Electrical Engineering, University of Hertfordshire, Hatfield AL10 9AB, U.K. He is now with the Trinity Convergence India Pvt., Ltd., Pune, India (e-mail: A.Malegaonkar@herts.ac.uk; amalegaonkar@trinityconvergence.com).

A. M. Ariyaeeinia and P. Sivakumaran are with the School of Electronic, Communication, and Electrical Engineering, University of Hertfordshire, Hatfield AL10 9AB, U.K. (e-mail: A.M.Ariyaeeinia@herts.ac.uk; apb@herts.ac.uk).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TASL.2007.896665

1558-7916/$25.00 © 2007 IEEE

ment for a large amount of training examples to ensure of the reliability of the training process. The methods such as this, which fall under the broad category of discrimination-based approaches, are not considered further as the focus of this paper is on the statistical-based approaches.

In recent years, there have been further advances in the statistical-based approaches to speaker change detection. These include the introduction of the XBIC measure which is derived from comparing BIC with a distance measure for hidden Markov models (HMMs) [9]. A previous study in the field by the authors has resulted in the development of an effective method which is termed bilateral scoring-based speaker change detection (BLS-SCD) [10]. This involves the use of a probabilistic pattern matching approach that has been shown to be more effective than both BIC and XBIC [10]. This superior performance of BLS-SCD is due to its incorporation of a mechanism for providing robustness against time-localized speech anomalies. Such anomalies can range from variations in the communication channel and background noise to uncharacteristic sounds generated by the speakers.

All the statistical methods mentioned above involve comparing the voice patterns in the two parts of a data segment divided by a hypothesized speaker change point. The typical duration of these data sub-segments can range from 1 to 5 s [2], [3]. The voice patterns in short subsegments of data across a hypothesized speaker change point can become quite diverse due to the differences in their acoustic contents. Therefore, when the spoken material is from the same speaker, the data subsegments being compared can appear quite dissimilar. This, indeed, can lead to an increased number of false alarms and hence to a reduced accuracy in the speaker change detection process.

This paper proposes an enhancement to the BLS-SCD approach in order to tackle the specific problem mentioned above. The fundamental to the proposed enhancement is changing the statistical speaker representation from a single Gaussian model to a Gaussian mixture model (GMM) obtained using a single-step Bayesian adaptation of a universal background model (UBM). One of the issues which are required to be addressed in adopting this approach is that of computational complexity. The study describes the proposed approach and its effectiveness in detail, and proposes methods for reducing the computational cost. The analysis of the performance of the proposed method relative to that of the original BLS-SCD is also included.

The remainder of the paper is organized in the following manner. Section II provides a detailed account of the motivation behind the present study. Section III starts with a full description of the proposed method and then discusses the mechanism by which the problem associated with speaker change detection in short data segments is alleviated. Section IV details the experimental investigations, and Section V gives the overall conclusions.

II. Motivation

Currently, the statistical approaches to SCD [1]–[10] are based on modeling each of the two test subsegments, associated with a hypothesized speaker change point, by using a single Gaussian density. As a result, these approaches inherently rely on averaging out the phonetic differences between the subsegments in question in order to focus on their speaker homogeneity. The extent of this averaging depends on the length of the subsegments and their phonetic contents. Fig. 1 shows an example of the dissimilarity of the successive subsegments of speech originated from the same speaker when their length is simultaneously increased from 1 to 30 s. Each of the subsegments in this example is modeled using a single Gaussian density and the dissimilarity is measured in terms of the Bhattacharya distance. Moreover, the speech data used is of telephone quality without background distortion.

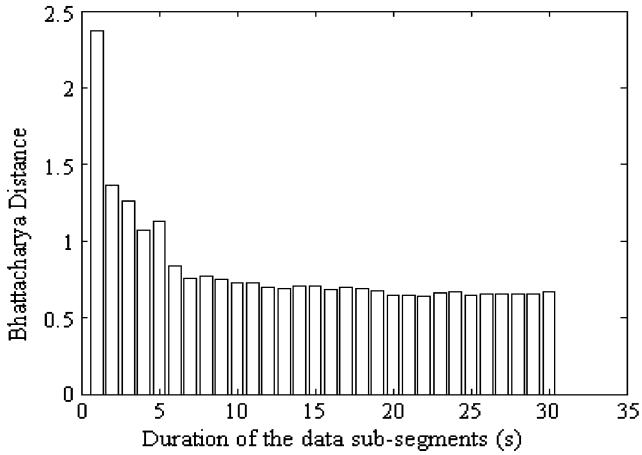

Fig. 1. Dissimilarity between single Gaussian models generated using successive subsegments of speech produced by the same speaker, as a function of the subsegment length.

Fig. 1 implies that, in order for the phonetic variations to be averaged out, the test segments need to be at least 7 s long. However, for speaker change detection, the length of the test segments should be much shorter than this (typically 1–5 s) to account for closely spaced speaker changes. In such cases, the phonetic differences between the speech subsegments from the same speaker may become significant. The use of single Gaussian modeling in such instances can result in obtaining a larger dissimilarity for two subsegments from the same speaker than that for two subsegments with less phonetic diversity, produced by different speakers. Fig. 2 illustrate this point further. The results shown in this figure are obtained by sliding an analysis window of 4 s in length through an audio stream in which the actual speaker change is present at the 24th s. For each positioning of the window, a speaker change is hypothesized at the center of the window and the dissimilarity of the subsegments is measured in terms of KL-2 and BIC. The traces of these measures are normalized to be in the range 0 to 1 for the purpose of comparison. As observed in this figure, in either of the two cases, it may not be possible to detect the actual speaker change point without generating several false alarms.

It should be noted that in practical applications, the SCD stage is normally followed by a clustering stage [5] for the purpose of resolving the false alarms generated in the above manner. However, too many false alarms in the SCD stage would certainly strain the clustering stage and it may even force the clustering process to suppress some of the actual speaker changes. It would be highly useful if such false alarms could be prevented in the

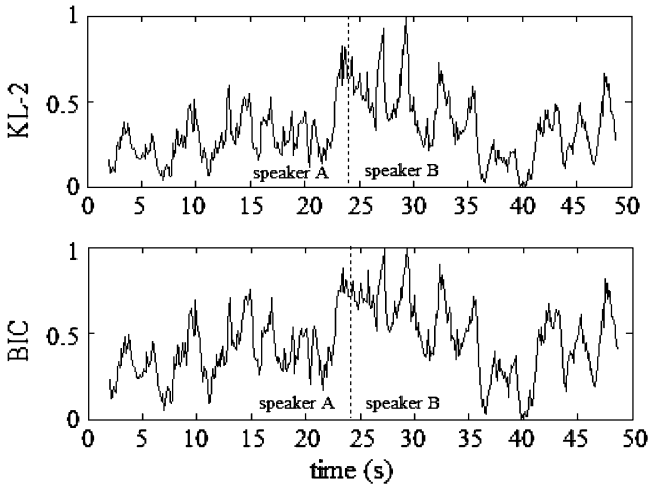

Fig. 2. Illustration of the false alarms primarily due to phonetic variations.

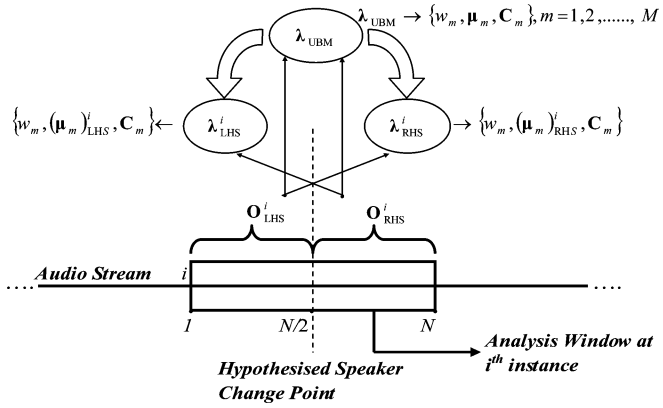

Fig. 3. BLS-SCD with adapted models.

SCD stage without missing the points of speaker change. This is, in fact, the aim of this study. The method proposed is based on the probabilistic framework introduced by the authors in a previous study [10]. The reason for choosing this framework is that it has already proven to provide an effective mechanism for tackling the time-localized speech distortions which arise due to various factors such as the environmental and channel conditions, and the speaker generated speech variations. The details of the proposed method are given in the next section.

III. Proposed Approach

In this method, a fixed-size analysis window is slid through the given audio stream at a predetermined rate. At each instance, a speaker change is hypothesized at the midpoint of the window. This results in the following two speech subsegments:

$$\mathbf{O}^i_{\mathrm{LHS}} = \left\{\mathbf{o}^i_1, \mathbf{o}^i_2, \ldots, \mathbf{o}^i_{N/2}\right\} \tag{1}$$

$$\mathbf{O}^i_{\mathrm{RHS}} = \left\{\mathbf{o}^i_{1+N/2}, \mathbf{o}^i_{2+N/2}, \ldots, \mathbf{o}^i_{N}\right\} \tag{2}$$

where $\mathbf{o}^i_n$ is the $(i+n)$th feature vector of the audio stream, and N is the size of the analysis window. The subscripts LHS and RHS are used to indicate whether the speech subsegment is on the left-hand side or the right-hand side of the hypothesized speaker change point. These subsegments are then used to obtain the speaker models, $\boldsymbol{\lambda}^i_{\mathrm{LHS}}$, and $\boldsymbol{\lambda}^i_{\mathrm{RHS}}$, respectively, by adapting an independent universal background model $\boldsymbol{\lambda}_{\mathrm{UBM}}$. This process is shown in Fig. 3.

While it is possible to generate single Gaussian models through the adaptation process, GMMs are preferred here for two reasons. First, they provide a better representation of the speaker information according to the previous studies in the field of speaker recognition [11]. Second, as described in Section III-A, the use of GMMs helps suppress the adverse effects of speech heterogeneity on speaker change detection. The adaptation method considered here is the single step Bayesian which has already been shown to be useful in speaker verification [12]. With this setup, and by using the probabilistic framework proposed for BLS-SCD, a measure for SCD can be derived as follows:

$$S^i_{\mathrm{SCD}} = p\left(\boldsymbol{\lambda}^i_{\mathrm{LHS}}|\mathbf{O}^i_{\mathrm{RHS}}\right) \times p\left(\boldsymbol{\lambda}^i_{\mathrm{RHS}}|\mathbf{O}^i_{\mathrm{LHS}}\right) \tag{3}$$

$$= \left\{\frac{p\left(\mathbf{O}^i_{\mathrm{RHS}}|\boldsymbol{\lambda}^i_{\mathrm{LHS}}\right)p\left(\boldsymbol{\lambda}^i_{\mathrm{LHS}}\right)}{p\left(\mathbf{O}^i_{\mathrm{RHS}}\right)}\right\} \times \left\{\frac{p\left(\mathbf{O}^i_{\mathrm{LHS}}|\boldsymbol{\lambda}^i_{\mathrm{RHS}}\right)p\left(\boldsymbol{\lambda}^i_{\mathrm{RHS}}\right)}{p\left(\mathbf{O}^i_{\mathrm{LHS}}\right)}\right\}. \tag{4}$$

In the log likelihood domain, the measure is given as

$$\rho^i_{\mathrm{SCD}} = \log\left(S^i_{\mathrm{SCD}}\right) = \left\{L\left(\mathbf{O}^i_{\mathrm{RHS}}|\boldsymbol{\lambda}^i_{\mathrm{LHS}}\right) - L\left(\mathbf{O}^i_{\mathrm{RHS}}\right)\right\} + \left\{L\left(\mathbf{O}^i_{\mathrm{LHS}}|\boldsymbol{\lambda}^i_{\mathrm{RHS}}\right) - L\left(\mathbf{O}^i_{\mathrm{LHS}}\right)\right\} \tag{5}$$

where $L(\cdot) = \log p(\cdot)$. In this equation, the prior probabilities of the speaker models, $p(\boldsymbol{\lambda}^i_{\mathrm{LHS}})$ and $p(\boldsymbol{\lambda}^i_{\mathrm{RHS}})$, are dropped as they can be considered equal for all the instances of the analysis window. When the speaker models are adapted from a UBM, the UBM itself can be used to approximate the terms $L(\mathbf{O}^i_{\mathrm{RHS}})$ and $L(\mathbf{O}^i_{\mathrm{LHS}})$. Hence, (5) in this case can be written as

$$\rho^i_{\mathrm{SCD}} = \left\{L\left(\mathbf{O}^i_{\mathrm{RHS}}|\boldsymbol{\lambda}^i_{\mathrm{LHS}}\right) - L\left(\mathbf{O}^i_{\mathrm{RHS}}|\boldsymbol{\lambda}_{\mathrm{UBM}}\right)\right\} + \left\{L\left(\mathbf{O}^i_{\mathrm{LHS}}|\boldsymbol{\lambda}^i_{\mathrm{RHS}}\right) - L\left(\mathbf{O}^i_{\mathrm{LHS}}|\boldsymbol{\lambda}_{\mathrm{UBM}}\right)\right\}. \tag{6}$$

A speaker change is assumed to be at the instance i (more precisely, at the point $i+N/2$), if $\rho^i_{\mathrm{SCD}} \geq \theta$, where θ is the decision threshold which should be determined *a priori* by using a set of experiments. Such an estimation of θ should be reasonably easy and reliable because of the property of ρ^i_{SCD} in terms of robustness against both variations in speech characteristics and phonetic heterogeneity.

It should be pointed out that (6) presents a framework similar to that of the cross likelihood ratio (CRL) which is extensively used in speaker clustering stage in the task of speaker diarization [13]–[16].

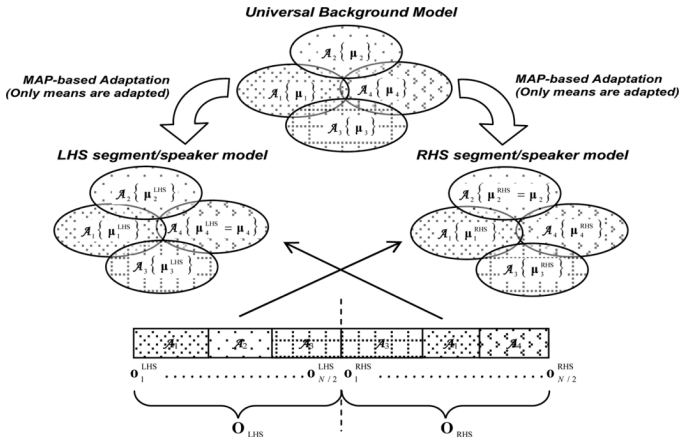

Fig. 4. Example used for demonstrating the effectiveness of the proposed approach. $\mathcal{A}_1$, $\mathcal{A}_2$, $\mathcal{A}_3$, and $\mathcal{A}_4$ denote four different acoustic classes. In each model, the mean vector of each mixture density is shown in the curly brackets. MAP: maximum *a posteriori*.

For the purpose of this paper, the above method is referred to as *adapted model-based BLS-SCD* (ABLS-SCD), as its fundamental difference to the original form of BLS-SCD is the use of adapted-GMMs. It should be noted that it is also possible to use only one of the speaker models (i.e., $\boldsymbol{\lambda}^i_{\mathrm{LHS}}$ or $\boldsymbol{\lambda}^i_{\mathrm{RHS}}$) for scoring instead of both as described in (3). In this case, the resulting approach would effectively be an enhancement to *unilateral scoring-based speaker change detection* (ULS-SCD) [10]. This method is not considered in the present study since it has already been shown that, due to its ability to exploit the nonreciprocity between different speakers, BLS-SCD is more effective than USL-SCD [10], [17].

A. Benefits of the Adapted Models in the BLS-SCD Framework

When a speaker model is generated by adapting a universal background model (UBM) using some training data, the adapted component densities are those associated with the acoustic units which are strongly represented in the training data. The densities of the UBM that correspond to the weaker or missing acoustic units in the training data remain unadapted in the speaker model [12]. In other words, the resultant speaker model shares unadapted component densities with the UBM. This can be considerably helpful for alleviating false alarms with the proposed approach. To highlight this point further, the situation given in Fig. 4 can be considered as an example.

The figure shows the LHS and RHS speaker models obtained by adapting a UBM with four mixture densities. Each of the densities $(\mathcal{A}_1 - \mathcal{A}_4)$ represents a very different acoustic class, and the LHS and RHS segments contain data which belong to these acoustic classes as indicated in the figure. Here, it is assumed that the MAP-based adaptation is applied only to the mean vectors, and each vector in a segment scores negligibly low with all the densities of the UBM except for the one it belongs to. It can be observed that due to the absence of the acoustic classes $\mathcal{A}_4$ and $\mathcal{A}_2$ in the LHS and RHS segments, respectively, one of the mixture densities in each of the LHS and RHS speaker models is an exact copy of that in the UBM. In this case, the first two terms in (6) can be expanded as follows:

$$\begin{aligned}\{L(\mathbf{O}_{\mathrm{RHS}}|\boldsymbol{\lambda}_{\mathrm{LHS}}) - L(\mathbf{O}_{\mathrm{RHS}}|\boldsymbol{\lambda}_{\mathrm{UBM}})\} \\ = \left\{\sum_{n=1}^{N/2}\left(L\left(\sum_{m=1}^{4} w_m \mathcal{N}\left(\boldsymbol{\mu}_m^{\mathrm{LHS}}, \mathbf{C}_m, \mathbf{o}_n^{\mathrm{RHS}}\right)\right)\right.\right. \\ \left.\left. - L\left(\sum_{m=1}^{4} w_m \mathcal{N}\left(\boldsymbol{\mu}_m, \mathbf{C}_m, \mathbf{o}_n^{\mathrm{RHS}}\right)\right)\right)\right\} \quad (7)\end{aligned}$$

where the index of instance i is left out for convenience, w_m and $\mathbf{C}_m$ are the weight and the covariance matrix associated with the mth mixture density in the UBM, and $\mathcal{N}(\boldsymbol{\mu}, \mathbf{C}, \cdot)$ is the multivariate Gaussian probability density function with mean $\boldsymbol{\mu}$ and covariance $\mathbf{C}$. It is clear from (7) that this approach leads to the elimination of the contribution of $\mathbf{o}_n^{\mathrm{RHS}} \in \mathcal{A}_4$ to the final score, ρ_{SCD}. Similarly, the combination of the third and fourth terms in (6) will result in the elimination of the contribution of $\mathbf{o}_n^{\mathrm{LHS}} \in \mathcal{A}_2$ to the final score. These indicate that the final score will only contain contributions from the segment vectors belonging to $\mathcal{A}_1$ and $\mathcal{A}_3$, as these two are the only acoustic classes shared by the data across the hypothesized point of speaker change. More precisely, the final score will have the following form:

$$\begin{aligned}\rho_{\mathrm{SCD}} = &\sum_{\mathbf{o}_n^{\mathrm{RHS}} \in \mathcal{A}_1} \rho_1^{\mathrm{LHS}}\left(\mathbf{o}_n^{\mathrm{RHS}}\right) + \sum_{\mathbf{o}_n^{\mathrm{RHS}} \in \mathcal{A}_3} \rho_3^{\mathrm{LHS}}\left(\mathbf{o}_n^{\mathrm{RHS}}\right) \\ &+ \sum_{\mathbf{o}_n^{\mathrm{LHS}} \in \mathcal{A}_1} \rho_1^{\mathrm{RHS}}\left(\mathbf{o}_n^{\mathrm{LHS}}\right) + \sum_{\mathbf{o}_n^{\mathrm{LHS}} \in \mathcal{A}_3} \rho_3^{\mathrm{RHS}}\left(\mathbf{o}_n^{\mathrm{LHS}}\right) \quad (8)\end{aligned}$$

where

$$\rho_m^{\mathrm{X}}(\mathbf{o}) = L\left(\mathcal{N}\left(\boldsymbol{\mu}_m^{\mathrm{X}}, \mathbf{C}_m, \mathbf{o}\right)\right) - L\left(\mathcal{N}\left(\boldsymbol{\mu}_m, \mathbf{C}_m, \mathbf{o}\right)\right). \quad (9)$$

From the above equations, it is evident that the proposed method inherits the two key features of its predecessor: the exploitation of the nonreciprocity between different speakers and the use of score normalization to tackle the effects of undesired variation in speech. Additionally, these equations indicate that the final score is based only on the similarity of the two groups of vectors belonging to $\mathcal{A}_1/\mathcal{A}_3$, in either side of the hypothesized speaker change point. This comparison over the same acoustic classes, indeed, helps focus on speaker specific features of the subsegments in question. As a result, the false alarms are expected to reduce without increasing the possibility of missing speaker change points.

Forming the final score based only on comparing the acoustic classes which are common in the speaker subsegments in question is undoubtedly a main attraction of the proposed method. The final score is virtually free from the contributions of highly dissimilar acoustic classes in the said subsegments. It should be noted that, here, the acoustic classes are defined by the mixture densities of the UBM. Hence, as the number of UBM mixture densities is increased, the associated acoustic classes become finer (they may even reach subphonetic levels). This increases the possibility of measuring the speaker differences between the subsegments in question with a greater accuracy. As a result, the effectiveness of the proposed method is expected to improve further by increasing the number of mixture densities in the UBM.

B. Implementation Issues

It should be pointed out that the primary drawback of the proposed method is the computational load. At each analysis instance, two adapted models have to be generated and the probability of each of these models generating the sequence of vectors used for adapting the other model must be computed. Furthermore, the likelihood of the UBM producing each of the two vector sequences in that analysis window has to be estimated. As the number of mixture densities in the UBM is increased, the computational load becomes more intense. This is nontrivial in practice even with the modern processors. Therefore, the computational saving is an important consideration in the proposed method. This section is dedicated to addressing this particular issue.

In order to improve the detection of closely spaced speaker changes, the interval at which the analysis window is shifted should be a fraction of the length of the analysis window. As this interval becomes smaller, the number of analysis instances increases and therefore the overall computational load increases even further. However, in such cases, due to the overlap between adjacent frames, there exist a large number of redundant computations. It is, therefore, possible to reduce the computational load significantly by effectively exploiting these redundant computations. In particular, if the shift interval is set to 50% of length of the analysis window, the RHS speaker model of the previous instance can be used as the LHS speaker model of the current instance. In this case, at each analysis instance, only the RHS speaker model is required to be generated. The main problem with this approach is the rigidity of the shift interval which may not be satisfactory for all practical cases. A scheme which exploits the said redundant operations while allowing the analysis window to shift at any rate required is proposed below. The only assumption in this method is that N is an even number which is divisible by δ (where N is the length of the analysis window, and δ is the rate at which the analysis window is shifted). This method is inspired by the authors' previous work on improving the computational efficiency of BIC [18].

It should be pointed out that building individual speaker models based only on the adaptation of the mean of the mixture densities of UBM has been proven to be the most effective approach in some previous studies on speaker recognition [12]. Obviously, this form of adaptation is also computationally more efficient than adapting all the parameters of Gaussian densities in the UBM. It is, therefore, believed that adapting only the mean vectors is the ideal choice for the proposed method. With this approach, at the ith analysis instance, the mean vector components for the LHS model are obtained by evaluating the following equation for $m = 1, \ldots, M$, where M is the number of mixtures in the UBM:

$$\boldsymbol{\mu}_{i,m}^{\mathrm{LHS}} = \frac{\left\{\sum_{t=i}^{i+\frac{N}{2}-1} \mathcal{P}_m\left(\mathbf{o}(t)\right)\mathbf{o}(t)\right\} + R\boldsymbol{\mu}_m}{\left\{\sum_{t=i}^{i+\frac{N}{2}-1} \mathcal{P}_m\left(\mathbf{o}(t)\right)\right\} + R} \tag{10}$$

where $\boldsymbol{\mu}_{i,m}^{\mathrm{LHS}}$ is the mean vector associated with the mth mixture density of the LHS speaker model at the ith analysis instance, R is the relevance factor for the mean statistic [12], and $\mathcal{P}_m(\cdot)$ is defined as

$$\mathcal{P}_m\left(\mathbf{o}(t)\right) = \frac{w_m \mathcal{N}\left(\boldsymbol{\mu}_m, \mathbf{C}_m, \mathbf{o}(t)\right)}{\sum_{j=1}^{M} w_j \mathcal{N}\left(\boldsymbol{\mu}_j, \mathbf{C}_j, \mathbf{o}(t)\right)} \tag{11}$$

where w_m is the weight of the mth mixture density of the UBM and all the other symbols in (10) and (11) have the same meaning as before. Equation (10) can be re-expressed in the following form:

$$\begin{aligned}\boldsymbol{\mu}_{i,m}^{\mathrm{LHS}} &= \frac{\mathbf{E}_{i,m}^{0\to\frac{N}{2}} + R\boldsymbol{\mu}_m}{\eta_{i,m}^{0\to\frac{N}{2}} + R} \\ &= \frac{\mathbf{E}_{i-1,m}^{0\to\frac{N}{2}} - \mathbf{E}_{i,m}^{0\to\delta} + \mathbf{E}_{i+\frac{N}{2},m}^{0\to\delta} + R\boldsymbol{\mu}_m}{\eta_{i-1,m}^{0\to\frac{N}{2}} - \eta_{i,m}^{0\to\delta} + \eta_{i+\frac{N}{2},m}^{0\to\delta} + R}\end{aligned} \tag{12}$$

where

$$\mathbf{E}_{x,y}^{a\to b} = \sum_{t=\delta x+a}^{\delta x+b-1} \mathcal{P}_y\left(\mathbf{o}(t)\right)\mathbf{o}(t) \tag{13}$$

and

$$\eta_{x,y}^{a\to b} = \sum_{t=\delta x+a}^{\delta x+b-1} \mathcal{P}_y\left(\mathbf{o}(t)\right). \tag{14}$$

Similarly, the equation for obtaining the components of the mean vector for the RHS model at instance i can be expressed as follows:

$$\begin{aligned}\boldsymbol{\mu}_{i,m}^{\mathrm{RHS}} &= \frac{\mathbf{E}_{i,m}^{\frac{N}{2}\to N} + R\boldsymbol{\mu}_m}{\eta_{i,m}^{\frac{N}{2}\to N} + R} \\ &= \frac{\mathbf{E}_{i-1,m}^{\frac{N}{2}\to N} - \mathbf{E}_{i+\frac{N}{2},m}^{0\to\delta} + \mathbf{E}_{i+N,m}^{0\to\delta} + R\boldsymbol{\mu}_m}{\eta_{i-1,m}^{\frac{N}{2}\to N} - \eta_{i+\frac{N}{2},m}^{0\to\delta} + \eta_{i+N,m}^{0\to\delta} + R}.\end{aligned} \tag{15}$$

Furthermore, the formulas for $L(\mathbf{O}_{\mathrm{LHS}}^{i}|\boldsymbol{\lambda}_{\mathrm{UBM}})$ and $L(\mathbf{O}_{\mathrm{RHS}}^{i}|\boldsymbol{\lambda}_{\mathrm{UBM}})$, which need to be calculated for each analysis window positioning for the purpose of score normalization, can be written as

$$L\left(\mathbf{O}_{\mathrm{LHS}}^{i}|\boldsymbol{\lambda}_{\mathrm{UBM}}\right) = \beta_i^{0\to\frac{\mathrm{N}}{2}} = \beta_{i-1}^{0\to\frac{\mathrm{N}}{2}} - \beta_i^{0\to\delta} + \beta_{i+\frac{N}{2}}^{0\to\delta} \tag{16}$$

$$L\left(\mathbf{O}_{\mathrm{RHS}}^{i}|\boldsymbol{\lambda}_{\mathrm{UBM}}\right) = \beta_i^{\frac{\mathrm{N}}{2}\to N} = \beta_{i-1}^{\frac{\mathrm{N}}{2}\to N} - \beta_{i+\frac{N}{2}}^{0\to\delta} + \beta_{i+N}^{0\to\delta} \tag{17}$$

where

$$\beta_x^{a\to b} = \sum_{n=\delta x+a}^{\delta x+b-1} \left(L\left(\sum_{m=1}^{M} w_m \mathcal{N}\left(\boldsymbol{\mu}_m^{\mathrm{LHS}}, \mathbf{C}_m, \mathbf{o}_n^{\mathrm{RHS}}\right)\right)\right) \tag{18}$$

Equation (12) and (15)–(17) imply that the redundant operations due to the overlap between the adjacent window positions will become negligible if the audio stream is encoded into triplets of $\{\mathbf{E}_i^{0\to\delta}, \boldsymbol{\eta}_i^{0\to\delta}, \beta_i^{0\to\delta}\}_{i=1,2,\ldots,I}$, where

$$\mathbf{E}_x^{a\to b} = \left\{\mathbf{E}_{x,1}^{a\to b}, \mathbf{E}_{x,2}^{a\to b}, \ldots\ldots, \mathbf{E}_{x,M}^{a\to b}\right\} \tag{19}$$

$$\boldsymbol{\eta}_x^{a\to b} = \left\{\eta_{x,1}^{a\to b}, \eta_{x,2}^{a\to b}, \ldots\ldots, \eta_{x,M}^{a\to b}\right\} \tag{20}$$

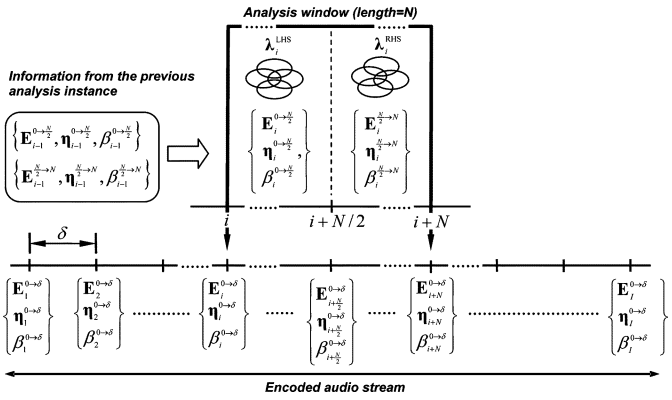

Fig. 5. Scheme for reducing the computational cost of the proposed method without compromising its effectiveness.

and I is the last possible analysis instance in the given audio stream. The logic behind this is that, at each analysis instance, the set of parameters computed for the previous analysis instance can be used together with the parameters in the encoded stream to determine the required mean vectors (an illustration of this is given in Fig. 5).

It should be noted that applying the above procedure requires neither all the sets of the encoded parameters to be stored nor the end of audio stream to be known. It is sufficient to memorise N/δ sets of encoded parameter at any given analysis instance. For example, at the ith analysis instance, the encoded parameter sets to be memorized are: $\{\mathbf{E}_i^{0\to\delta}, \boldsymbol{\eta}_i^{0\to\delta}, \beta_i^{0\to\delta}\}$, $\{\mathbf{E}_{i+1}^{0\to\delta}, \boldsymbol{\eta}_{i+1}^{0\to\delta}, \beta_{i+1}^{0\to\delta}\}, \ldots\ldots, \{\mathbf{E}_{i+N}^{0\to\delta}, \boldsymbol{\eta}_{i+N}^{0\to\delta}, \beta_{i+N}^{0\to\delta}\}$. It is now easy to see that subtracting $\{\mathbf{E}_i^{0\to\delta}, \boldsymbol{\eta}_i^{0\to\delta}, \beta_i^{0\to\delta}\}$, and adding $\{\mathbf{E}_{i+N+1}^{0\to\delta}, \boldsymbol{\eta}_{i+N+1}^{0\to\delta}, \beta_{i+N+1}^{0\to\delta}\}$ would give the required encoded parameter sets for the $(i+1)$th analysis instance. This implies that this encoding can be efficiently implemented as an in-place algorithm by using such structures as the circular buffer for the purpose of live audio analysis.

It can also be noticed that (13), (14), and (18), which are used for computing the encoding parameters, $\{\mathbf{E}_i^{0\to\delta}, \boldsymbol{\eta}_i^{0\to\delta}, \beta_i^{0\to\delta}\}$, themselves share a significant amount of common operations. Fig. 6 shows a possible implementation for eliminating these redundancies.

Along with the use of the encoding scheme described above, it is possible to deploy another approach for further improving the computational efficiency of the proposed method. The idea is based on two observed effects initially reported in [12].

1) When evaluating the likelihood of a large GMM generating a given vector, only a small group of mixture densities contributes significantly to the final score. This is because, such a GMM is expected to represent wide varieties of acoustic events and, therefore, its mixture densities span over a large space. As a result, an acoustic event captured by a single vector is covered by a small number of mixture densities.
2) Due to the single-step Bayesian adaptation, the mixture densities of the adapted GMM retain a correspondence with the mixtures of the UBM, so that vectors close to a particular mixture in the UBM will also be close to the corresponding mixture in the adapted GMM.

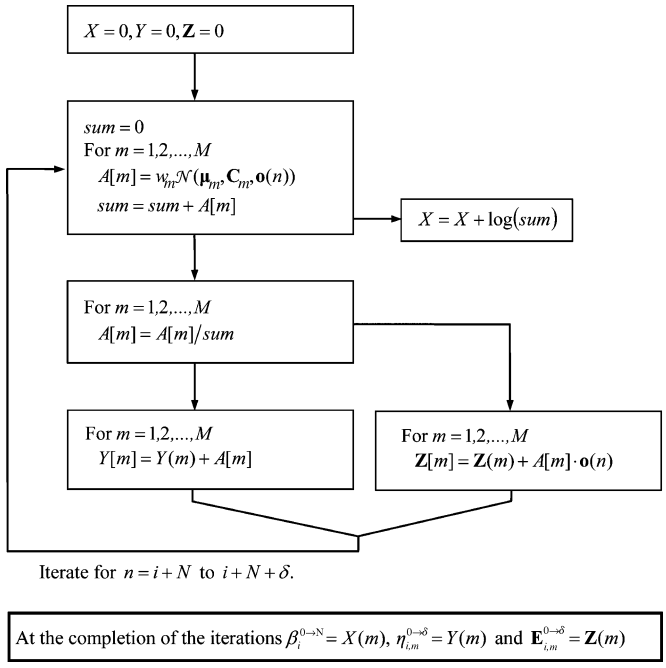

Fig. 6. Algorithm for eliminating the redundant operations in the computation of the encoding parameters.

This implies that with the algorithm described in Fig. 6 if the UBM mixture densities achieving significant scores are identified in each main iteration (which invokes different $\mathbf{o}(n)$) for computing $A[\cdot]$ parameters, this information can be used in the subsequent operations to save computation. More importantly, this information can be used later at the cross evaluation stage (where $L(\mathbf{O}_{\mathrm{RHS}}|\boldsymbol{\lambda}_{\mathrm{LHS}})$ and $L(\mathbf{O}_{\mathrm{RHS}}|\boldsymbol{\lambda}_{\mathrm{UBM}})$ are determined) to save a significant amount of computation. This involves adding a fourth parameter, $\boldsymbol{\gamma}_i^{0\to\delta}$, to the encoding parameter set, where $\boldsymbol{\gamma}_i^{0\to\delta}$, is a $\delta \times K$ matrix in which each column represents indices of K mixture densities of the UBM, achieving the top K scores for $\mathbf{o}(n)$, $n \in \{1, \ldots, \delta\}$. It should be noted that by making K a variable which is dependant on $\mathbf{o}(n)$, it may be possible to save some additional computation and/or storage. However, for the sake of simplicity it is kept constant here.

This method, unlike the two efficiency enhancements described earlier, could affect the effectiveness of the proposed method if the value of K is not chosen appropriately. In the speaker verification experiments which first exploited the said observed effects, it has been concluded that for a 2048-mixture UBM, it is sufficient to only consider the mixtures that achieves the top five scores [12]. In a recent study for open set speaker identification, it has been shown that considering the best scoring mixture is the most appropriate for that task [19]. The nature of problem here is somewhat different from those in [12] and [19]. Therefore, it may not be possible to select the best value for K based on the information available in the literature. As a result, it is decided to obtain the value of K by carrying out an experimental investigation. Details of this are presented in Section IV-E. For the purpose of this paper, the version of ABLS-SCD which incorporates all the efficiency enhancements described in this section is referred to as *Fast*-ABLS-SCD.

IV. Experimental Investigation

A. Speech Data

The experiments are conducted using the speech data obtained in clean audio conditions as well as in broadcast news audio conditions. The test data from clean audio consists of two artificial recordings created using a subset of the TIMIT database. These recordings are similar to those adopted in [2] and have 100 and 1000 speaker turns, respectively. In this paper, these datasets are referred to as TIM_100 and TIM_1000 and they constitute test speech data of 0.5 and 4.5 h in length, respectively. The duration of speaker specific segments in these recordings varies from about 2.5 to around 33 s.

For the experiments with broadcast news audio, the News Shows in the HUB-4 database are considered and used to obtain two sets of test data. The first set is obtained from seven recordings in the CNN Prime News. This consists of around 2.5 h of speech data with 500 speaker change points. This set is termed HUB-4_500. The second test dataset consists of five parts, each based on four recordings from a different News Show in HUB-4. These are 1) "PRI The World," 2) "CNN Prime News," 3) "CNN The World Today," 4) "C-SPAN Public Policy," 5) "C-SPAN Washington Journal." The CNN Prime News recordings used in this case are different from the ones used in HUB-4_500. The overall duration of this second test data is about 20 h, and it includes 3000 speaker changes. This dataset is termed HUB-4_3000. The duration of speaker-specific segments in HUB-4_500 is between around 2.8 to 115 s, and varies from around 3.1 to about 125 s in HUB-4_3000. It should be pointed out that both these HUB-4 test datasets are based on excluding the regions in the original audio recordings containing overlapped speech, music, commercials, and other nonspeech artefacts as such phenomena are outside the scope of this study.

The above-mentioned data characteristics in terms of large variation in the duration of speaker specific segments help conduct experiments in conditions similar to those expected in real broadcast applications.

B. Feature Representation

For the purpose of this study, the tth frame of the input speech data is represented as $\mathbf{c}_t \equiv \{c_t(1), c_t(2), \ldots, c_t(20)\}$, where $c(i)$ is the ith, linear predictive coding-derived cepstral (LPCC) parameter. The extraction of LPCC parameters is based on first pre-emphasising the input speech data using a first order digital filter and then segmenting it into 20 ms frames at the intervals of 10 ms using a Hamming window.

C. Speaker and Background Modeling

For the purpose of the experiments, four UBMs are built: one using the clean data and the remaining three using the broadcast audio material. For the UBM based on clean audio, a dataset is extracted from the TIMIT database. This dataset consists of one hour of speech material gathered from 90 speakers, and it does not share any speakers with either TIMIT_100 or TIMIT_1000.

The UBMs based on broadcast audio are built using speech material from three News Shows in HUB-4, and by excluding the undesired audio regions as described in Section IV-A. The News Shows considered are "CNN Prime News," "ABC World News Tonight," and "CNN Headline News." In each case, three recordings from the relevant News Show are used. The recordings used for each individual UBM provide about one hour of speech material from 40 background speakers. In the case of UBM based on CNN Prime News, the recordings adopted are different from those used in HUB-4_500 and HUB-4_3000. It should also be noted that all the UBMs are gender balanced, i.e., they are based on equal numbers of male and female speakers.

The representation of the speakers in the proposed method is based on GMMs. These are obtained by the Bayesian adaptation of the relevant UBM in each case. The relevance factor used for the adaptation of speaker models in all the cases is 15, which is similar to that used in [12].

D. Audio Scanning and Testing Procedure

The tests in this experimental investigation are conducted by sliding a window of 4-s duration through the recording at a rate of 0.1 s between two successive instances of the window. The length of the sliding window and the sliding rate are decided *a priori* using pilot experiments. It should be pointed out that, in a given application, the choice of the analysis window length depends on the duration of the shortest speaker-specific segments in the data. As indicated earlier, in the present study, the shortest speaker-specific segments are around 2.5–3 s long. Therefore, the duration of 4 s considered for the analysis window, leading to a data length of 2 s on either side of the hypothesized speaker change point, appears to be appropriate.

For the purpose of evaluating the SCD performance, a detected speaker change point is declared to be correct if it is within a 0.25-s margin on either side of the actual speaker change point. The error rates are calculated in terms of the percentage of correct speaker change points that are missed, i.e., the missed detection rate (MDR), and the percentage of test points wrongly identified as speaker change points, i.e., the false alarm rate (FAR). This error calculation is the same as that given in [7]. The equal error rate (EER %) is then calculated by adjusting the decision threshold such that MDR and FAR are equal. This is then used as the measure of performance in this work.

E. Experimental Conditions, Results, and Discussions

The first set of experiments in this study is carried out to determine the optimum value for the key parameter K of the *Fast*-ABLS-SCD approach (Section III-B). For this purpose, the following criterion is used:

$$\hat{K}_M = \underset{1 \le K \le M}{\arg\min} \left[\frac{\sum_{n=1}^{N_E} L\left(\sum_{k=1}^{K} w_{f(k)} \mathcal{N}\left(\boldsymbol{\mu}_{f(k)}, \mathbf{C}_{f(k)}, \mathbf{o}_n \right) \right)}{\sum_{n=1}^{N_E} L\left(\sum_{m=1}^{M} w_m \mathcal{N}(\boldsymbol{\mu}_m, \mathbf{C}_m, \mathbf{o}_n) \right)} \ge 0.99 \right] \tag{21}$$

where $\hat{K}_M$ is the optimum value of K for an Mth order UBM (i.e., a UBM with M mixtures), N_E is the total number of vectors in the considered evaluation data set (TIM_100 in this case), $f(k)$ is the function that gives the index of the kth top scoring mixture in the UBM, and all the other symbols have the same meaning as before. In other words, the optimum value for K

TABLE I
RESULTS OF THE EXPERIMENTS WITH *Fast*-ABLS-SCD IN TERMS OF THE NUMBER ($\hat{K}_M$) OF THE BEST SCORING MIXTURES OBTAINED FOR EACH UBM OF A PARTICULAR ORDER. THE TABLE ALSO SHOWS THE PERCENTAGE OF THE UBM MIXTURES REPRESENTED BY $\hat{K}_M$ IN EACH CASE

UBM Order : M	64	128	256	512	1024
$\hat{K}_M$	25	40	35	40	20
$\frac{100\times\hat{K}_M}{M}(\%)$	39%	31%	14%	8%	2%

TABLE II
EFFECTIVENESS AND EFFICIENCY OF *Fast*-ABLS-SCD FOR VARIOUS UBM ORDERS (THE DURATION OF THE AUDIO STREAM USED IS 0.5 h)

UBM Order	64	128	256	512	1024
EER (%)	1.87	1.76	1.70	1.68	1.53
Processing Time (hrs)	0.37	0.45	1.2	2.24	3.88
Processing speed in relation to real-time (×RT)	1.35	1.11	0.47	0.22	0.13

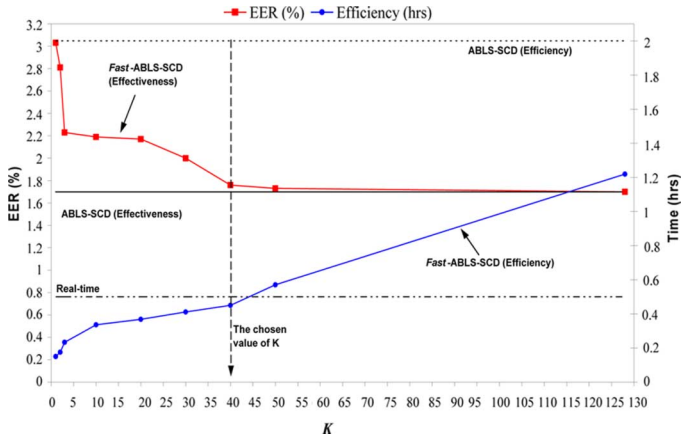

Fig. 7. Effectiveness and efficiency of ABLS-SCD with and without efficiency enhancement in the case of 128-mixture UBM.

TABLE III
COMPARISON OF ABLS-SCD AND *Fast*-ABLS-SCD WITH $K = 40$

	ABLS-SCD	Fast-ABLS-SCD
EER (%)	1.7	1.76
Processing Time (hrs)	2.0	0.45

is set to be the minimum number of top-scoring UBM mixtures that are needed to cover, at least, 99% of the accumulated log-likelihood score yield by the entire mixtures in the UBM. Table I shows the results of this study in terms of the number as well as the percentage of the best scoring mixtures that are chosen for each UBM of a particular order. It can be observed that the lower the UBM order is, the higher the proportion of the mixtures required to satisfy the said criterion becomes.

The next set of experiments is designed to study both the effectiveness and efficiency of *Fast*-ABLS-SCD for the considered UBM orders with the corresponding optimum values of K. Again, the TIM_100 dataset, which is 0.5 h in duration, is used. The experiments are run on the Windows XP platform powered by a Pentium 4, 2-GHz processor. The results obtained are given in Table II. Here, the effectiveness is expressed in terms of EER (%), while the efficiency is provided in terms of the time taken to process the entire audio material. This table also shows the efficiency in terms of the real time (RT) norm, shown as $(\times\mathrm{RT})$.

It can be observed from Table II that, as expected, the effectiveness of *Fast*-ABLS-SCD increases with the model order. However, the tradeoff here is the computational efficiency. It appears that the largest model order that could keep the processing speed faster than the real-time speed is 128. The error rate for this case can be reduced by about 13%, if the model order is increased to 1024. The problem is that with such a large model, the processing speed reduces to about 0.13 times the real-time speed. It is felt that keeping the processing speed at least at the real-time speed is important, as it would enable the proposed method to process live audio within the time constraint. As a result, it is concluded that a model order of 128 is the best choice for carrying-out the rest of the experimental studies. Fig. 7 provides additional information on the effectiveness and efficiency of this chosen case as the value of K is increased from 1 to 128. This figure also includes the corresponding values yielded by the original form of the ABLS-SCD approach.

It can be observed from these results that even when $K = 128$, i.e., all the mixtures are considered for scoring, the computational efficiency of *Fast*-ABLS-SCD is considerably ($\approx$40%) better than that of ABLS-SCD. This difference is due to the elimination of the redundant computations as described in Section III-B. However, the results also indicate that this efficiency enhancement alone is not sufficient to operate *Fast*-ABLS-SCD in real-time. As expected, the maximum computational efficiency is obtained when $K = 1$. In this case, *Fast*-ABLS-SCD can be operated at several times faster than the real-time speed. However, in terms of effectiveness, this is the worst case, as observed in Fig. 7. According to this figure, when K is increased from 1 to 3, a steep improvement occurs in the effectiveness of *Fast*-ABLS-SCD. This improvement seems to slow down significantly when K is increased from 3 to 20. Another sharp improvement occurs when K is increased from 20 to 40. At $K = 40$, *Fast*-ABLS-SCD become almost as effective as ABLS-SCD. A very small improvement is seen when K is increased beyond this point. It can also be observed that at $K = 40$, it is possible to operate *Fast*-ABLS-SCD in real-time. Therefore, a value of 40 seems to be the best choice for K. This is, in fact, the conclusion of the experimental study summarized in Table I. For convenience, Table III presents a comparison of the effectiveness/efficiency of ABLS-SCD and *Fast*-ABLS-SCD with $K = 40$. The results in this table indicate that the proposed efficiency enhancement approach improves the efficiency of the ABLS-SCD by 77.5% at the cost of 3.5% degradation in effectiveness.

The next set of experiments examines the variation in the performance of *Fast*-ABLS-SCD when the test dataset and the

TABLE IV
EFFECTIVENESS OF *Fast*-ABLS-SCD IN DIFFERENT DATA CONDITIONS

UBM Data Source	EER (%) for HUB-4_500 test data (based on CNN Prime News)
CNN Prime News	13.0
ABC World News Tonight	13.8
CNN Headline News	14.3
TIMIT	14.6

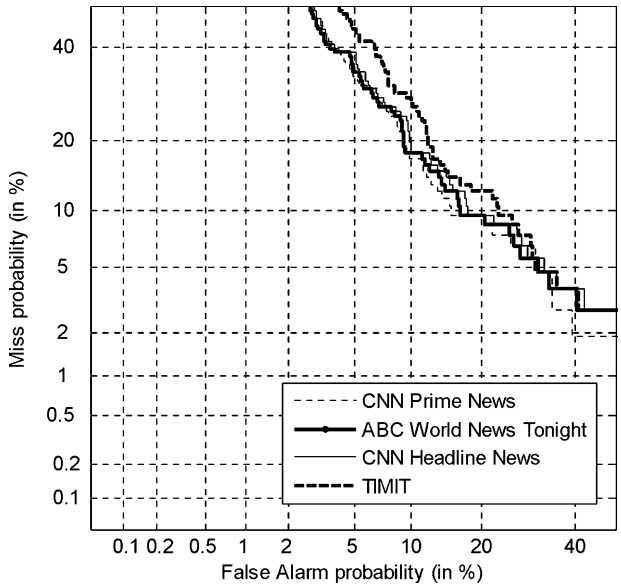

Fig 8. Performance of the proposed method operating in different data conditions.

TABLE V
RELATIVE EFFECTIVENESS OF VARIOUS SCD APPROACHES IN TERMS OF EER (%)

	KL2	BIC	XBIC	BLS-SCD	*Fast*-ABLS-SCD
TIMIT_1000	5.5	4.2	4.3	2.8	1.7
HUB-4_3000	25.6	22.9	22.5	19.5	16.1

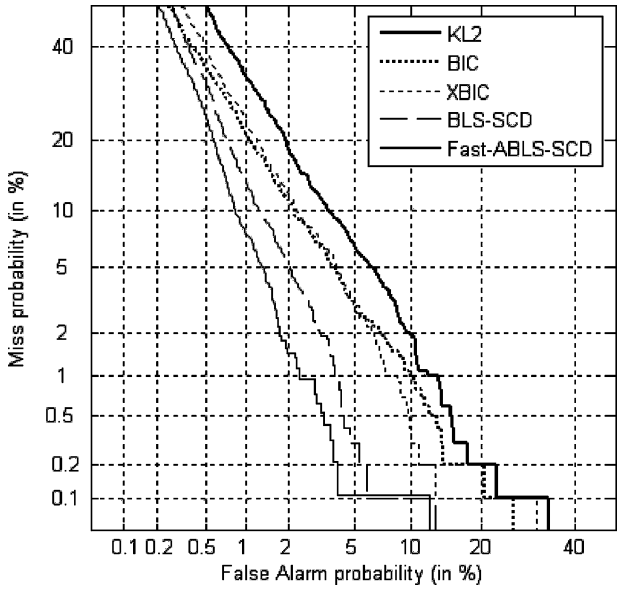

Fig. 9. DET plots of various SCD methods, based on the TIMIT_1000 test dataset.

dataset used for building the UBM are obtained from the same source and when these are from different sources. This is considered a required measure of performance as the application of the proposed SCD method to some given audio material, in practice, may involve the deployment of a UBM built previously using some other (different) source of data. For this study, HUB-4_500, which is based on CNN Prime News, is used as the test data. The investigation involves conducting four independent experiments, each based on a different UBM. For this purpose, the UBMs built using CNN Prime News, ABC World News Tonight, CNN Headline News, and TIMIT are deployed. The results for these experiments are presented in Table IV in terms of EER (%), and also given in Fig. 8 as DET plots.

Based on the results, it is evident that, as expected, the lowest EER with the proposed method is obtained when the data for building the UBM is extracted from the same source as that of the test data, i.e., CNN Prime News. However, it is also noted (especially from Fig. 8) that the effectiveness of *Fast*-ABLS-SCD does not vary significantly when operating in a cross-data condition. This is especially the case when the UBM is built using data from one of the News Shows considered for this purpose. Nevertheless, it is observed that the EER obtained through the use of the TIMIT-based UBM is also only marginally worse than those when the UBM is based on "CNN Headline News" or "ABC World News Tonight."

The results in Table IV and Fig. 8 give an indication of the performance of the proposed method in isolation. In order to obtain a more comprehensive evaluation of the effectiveness of *Fast*-ABLS-SCD, it is decided to compare its performance with those of other well-known SCD methods using two large test datasets. The SCD methods considered for the purpose of comparison are BLS-SCD, XBIC, BIC, and KL-2, i.e., the classical approach described in [1] with KL-2 measure). The first set of experiments is based on the use of TIMIT_1000. For this investigation, the UBM built using a subset of the TIMIT database (Section IV-C) is used with the proposed method. The second set of experiments is intended to examine the relative usefulness of the proposed method under a condition which is closer to those in practice. For this set of experiments, HUB-4_3000 is used as the test data. The UBM used with the proposed *Fast*-ABLS-SCD method in this case is that based on "ABC World News Tonight" (Section IV-C). As indicated in Section IV-A, this News Show is not part the data used in HUB-4_3000.

The results for these experiments are presented in terms of EER (%) in Table V, and also given as DET plots in Figs. 9 and 10.

These results show that among the considered methods, KL-2 is the worst performer. This is followed by XBIC and BIC which

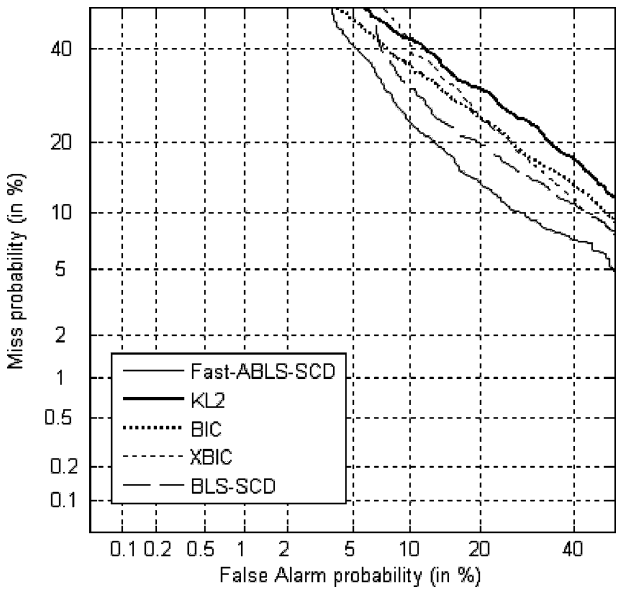

Fig. 10. DET plots for various SCD methods, obtained on the HUB-4_3000 test dataset.

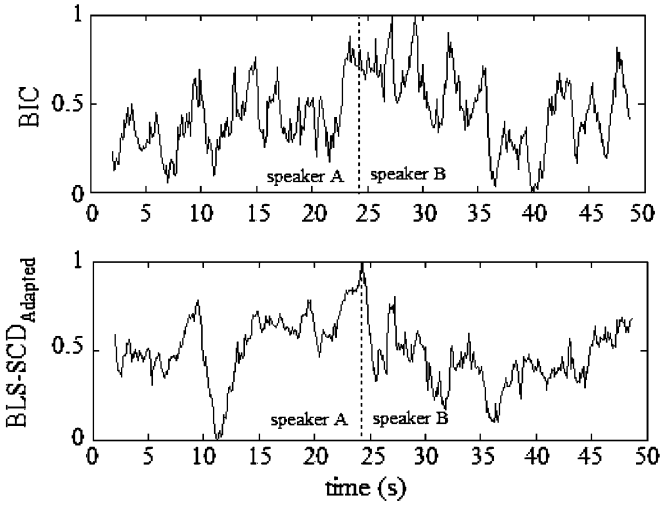

Fig. 11. Suppression of false alarms with the proposed approach.

both achieve a similar level of performance. Moreover, Table V indicates that the use of the BLS-SCD approach, which is introduced in [10] by the authors, leads to considerably higher accuracy in SCD than that offered by either of BIC or XBIC. The improvements in accuracy offered by BLS-SCD is seen to be at least by 33% and 13% in the cases of clean and broadcast test data conditions, respectively. Table V also shows that the effectiveness offered by *Fast*-ABLS-SCD is at least 17% better than that obtained with BLS-SCD. This makes the *Fast*-ABLS-SCD approach, which can also operate in real-time, the most effective among all the methods considered. As noted in Section III-A, the uniqueness of this method is its ability to form the final score based only on the comparison of the common acoustic classes in the two subsegments across the hypothesized speaker change point. As a result, the final score is virtually free from the contributions of highly dissimilar acoustic classes in the said subsegments. It should be reiterated that this important characteristic of *Fast*-ABLS-SCD is in addition to two valuable properties which the method shares with BLS-SCD. These are the ability to exploit the nonreciprocity between different speakers, and the incorporation of an effective means for tackling various forms of speech distortions. Moreover, the proposed method provides the possibility for measuring the speaker differences over 128 acoustic classes which are internally defined by the associated UBM. None of the other considered methods attempt to use the speaker differences within the acoustic classes. They simply rely on single-Gaussian densities to model the overall acoustic structure of the subsegments in question. Consequently, the proposed method is capable of measuring the speaker differences with higher accuracy than that offered by any of the other methods.

As noted in Section III-A, the primary benefit of the proposed method is believed to be the reduction of the false alarm without increasing the misdetection rate. This is evident from the DET plots shown in Figs. 9 and 10. In order to illustrate this point further, the experiments leading to the results given in Fig. 2 are rerun with *Fast*-ABLS-SCD. The outcome is shown in Fig. 11 together with the original result for BIC. From this example, it can be observed that the peaks which could potentially lead to false alarms in the case of BIC are suppressed effectively in the case of the proposed approach.

V. Conclusion

The conventional, computationally efficient, approaches to speaker change detection are based on the use of single Gaussian modeling for speaker representation. A main drawback of these approaches is that they can lead to a high rate of false alarms. A main cause of this problem has been shown to be the phonetic heterogeneity of the speech material being tested. This is of particular concern when successive speech sections used for speaker change detection are of short duration, e.g., 1–5 s.

Incorporating Gaussian mixture models, using a single-step adaptation procedure, in the framework of probabilistic pattern matching has been shown to be highly effective for tackling the above-mentioned problem. The main attraction of the proposed approach is that the comparison of two successive parts of a given segment (the subsegments in the analysis window) of speech is based on their common phonetic contents. This, which is achieved by using score normalization together with the said modeling procedure, has shown to provide an effective means for reducing false alarms.

The superior effectiveness of the proposed approach over the conventional methods has been demonstrated through a set of experimental investigations. According to the experimental results, in broadcast audio conditions, the speaker change detection accuracy obtainable with the currently popular method of BIC, can be enhanced by about 30% through the use of the proposed method.

It has been found that a drawback with the proposed method is the high computational cost due to the requirement for the adaptation of Gaussian mixture models. In order to enhance the computational efficiency of the approach, a fast algorithm has been developed which allows the exploitation of the redundant operations. The algorithm is based on a combination of a fast scoring procedure and encoding the sufficient statistics in the

Gaussian mixture model. It has been shown that, using an appropriate model order for speaker representation, it may be possible to achieve considerable enhancement in the computational efficiency while maintaining the performance accuracy to a large extent. In particular, through a set of experiments it has been demonstrated that, using GMMs of order 128 for speaker modeling, the fast algorithm allows operating in real-time through enhancing the computational efficiency by about 77%. The drop in accuracy in this case is not found to be significant and is shown to be by about 3.5%.

ACKNOWLEDGMENT

The authors would like to thank Dr. J. Fortuna for his valuable comments and discussions during the course of this work.

REFERENCES

[1] H. Gish, M.-H. Siu, and R. Rohlicek, "Segregation of speakers for speech recognition and speaker identification," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process., (ICASSP'91)*, 1991, vol. 2, pp. 873–876.

[2] P. Delacourt and C. J. Wellekens, "DISTBIC: A speaker-based segmentation for audio data indexing," *Speech Commun.*, vol. 32, no. 1–2, pp. 111–126, 2000.

[3] S. Johnson, "Speaker tracking," M.Phil. thesis, C.U.E.D., Univ. Cambridge, Cambridge, U.K., 1997.

[4] S. Chen and P. Gopalakrishnan, "Speaker, environment and channel change detection and clustering via the Bayesian information criterion," in *Proc. DARPA Speech Recognition Workshop*, 1998, pp. 127–132.

[5] A. Tritschler and R. Gopinath, "Improved speaker segmentation and segments clustering using the Bayesian Information Criterion," in *Proc. Eurospeech*, 1999, vol. 2, pp. 679–682.

[6] J. Ajmera *et al.*, "Robust speaker change detection," *IEEE Signal Process. Lett.*, vol. 11, no. 8, pp. 649–651, Aug. 2003.

[7] M. Roch and Y. Cheng, "Speaker segmentation using MAP-Adapted Bayesian Information Criterion," in *Proc. Speaker Lang. Recognition Workshop (Odyssey)*, 2004, pp. 349–354.

[8] V. Karthik, D. S. Satish, and C. C. Sekhar, "Speaker change detection using support vector machine," in *Proc. 3rd Int. Conf. Non-Linear Speech Process.*, Barcelona, Spain, Apr. 19–22, 2005, pp. 130–136.

[9] X. Anguera, "XBIC: Real-time cross probabilities measure for speaker segmentation," Univ. California Berkeley, ICSI—Berkeley Tech. Rep., Aug. 2005.

[10] A. Malegaonkar, A. Ariyaeeinia, P. Sivakumaran, and J. Fortuna, "Unsupervised speaker change detection using probabilistic pattern matching," *IEEE Signal Process. Lett.*, vol. 13, no. 8, pp. 509–512, Aug. 2006.

[11] D. Reynolds and R. C. Rose, "Robust text-independent speaker identification using Gaussian mixture speaker models," *IEEE Trans. Speech Audio Process.*, vol. 3, no. 1, pp. 75–83, Jan. 1995.

[12] D. A. Reynolds, T. F. Quatieri, and R. B. Dunn, "Speaker verification using adapted Gaussian mixture models," *Digital Signal Process.*, vol. 10, no. 1–3, pp. 19–41, 2000.

[13] D. Reynolds, E. Singer, B. Carlson, J. McLaughlin, G. O'Leary, and M. Zissman, "Blind clustering of speech utterances based on speaker and language characteristics," in *Proc. Int. Conf. Spoken Lang. Process. (ICSLP'98)*, 1998, pp. 610–613.

[14] D. Reynolds and P. Torres-Carrasquillo, "Approaches and applications of audio diarization," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process. (ICASSP'05)*, 2005, vol. 5, pp. 953–956.

[15] R. Sinha, S. Tranter, M. Gales, and P. Woodland, "The Cambridge University March 2005 speaker diarisation system," in *Proc. Interspeech*, 2005, pp. 2437–2440.

[16] X. Zhu, C. Barras, S. Meignier, and J.-L. Gauvain, "Combining speaker identification and BIC for speaker diarisation," in *Proc. Interspeech*, 2005, pp. 2441–2444.

[17] E. Parris and M. Carey, "Multilateral techniques for speaker recognition," in *Proc. Int. Conf. Spoken Lang. Process. (ICSLP'98)*, 1998, pp. 1343–1346.

[18] P. Sivakumaran, J. Fortuna, and A. Ariyaeeinia, "On the use of the Bayesian information criterion in multiple speaker detection," in *Proc. Eurospeech*, 2001, pp. 795–798.

[19] J. Fortuna, P. Sivakumaran, A. Ariyaeeinia, and A. Malegaonkar, "Open-set speaker identification using adapted Gaussian mixture models," in *Proc. Interspeech*, pp. 1997–2000.

Amit S. Malegaonkar received the B.Eng. degree in electronics and telecommunications from the University of Pune, Pune, India, and the M.Sc. degree in data communications and networks and the Ph.D. degree in voice biometrics from the University of Hertfordshire, Hatfield, in 2000, 2002, and 2006, respectively.

He is currently a Senior Software Developer for audio codecs at Trinity Convergence India Pvt., Ltd., Pune.

Aladdin M. Ariyaeeinia received the B.Eng. degree in communication engineering from the Institute of Communications, Tehran, Iran, the M.Sc. degree in digital signal processing from Keele University, Staffordshire, U.K., and the Ph.D. degree in active imaging Trent Polytechnic, Nottingham, U.K., in 1976, 1982, and 1986, respectively. He was awarded C.Eng. status in 1988.

In 1986, he joined the University of East Anglia as a Senior Research Fellow. Over the last 20 years, he has been working in the Faculty of Engineering and Information Sciences, University of Hertfordshire, Hatfield. U.K. During this period, he has been conducting research on various aspects of speech processing in close collaboration with industry. He is now a Reader in Signal Processing and is responsible for leading the Audio Processing and Biometrics Group. His current research interests include speaker and language recognition, speech enhancement, speaker-based audio indexing, biometric fusion, and active coordinate imaging. He has over 50 publications, and has served on various scientific committees.

Perasiriyan Sivakumaran received the first-class B.Eng. (hon.) degree in electrical and electronic engineering from the University of Herefordshire, Hatfield, U.K., in 1994. Subsequently, he was sponsored by BT Research Laboratories to carryout a program of research in the field of speaker recognition. This led to the Ph.D. degree in 1998.

From 1998 to 2001, he worked with the BBC Research Group to develop a speech-based semi-automated system for subtitling live TV programs. He joined 20/20 Speech in 2001 and continued to work there as a (speech) scientist until 2003. His research activities continued at the Canon Research Centre Europe between 2003 and 2006. Since his departure from this research center, Sivakumaran has been working for a leading investment bank while being a Visiting Industrial Fellow at the University of Herefordshire. His current research interest in the area of speech processing includes speaker and language recognition, speaker-based audio indexing, speech enhancement, and speech synthesis.