IDIAP RESEARCH REPORT

IMPROVING REAL TIME FACTOR OF INFORMATION BOTTLENECK-BASED SPEAKER DIARIZATION SYSTEM

Srikanth Madikeri David Imseng Hervé Bourlard

Idiap-RR-18-2015

JUNE 2015

Centre du Parc, Rue Marconi 19, P.O. Box 592, CH - 1920 Martigny
T +41 27 721 77 11 F +41 27 721 77 12 info@idiap.ch www.idiap.ch

Improving Real Time Factor of Information Bottleneck-based Speaker Diarization System

Srikanth Madikeri[1]*, David Imseng*[1] *and Herve Bourlard*[1,2]

[1]Idiap Research Institute, Martigny, Switzerland
[2]Ecole Polytechnique Fédérale de Lausanne, Lausanne, Switzerland
srikanth.madikeri@idiap.ch, herve.bourlard@idiap.ch

Abstract

In this paper, we discuss a fast implementation of the Information Bottleneck (IB) approach based speaker diarization system. The IB system has low real time factors (RTF) compared to other conventional approaches to speaker diarization. The IB diarization system has three modules: (1) a posterior probability estimation module, (2) a clustering module and (3) a realignment module. In this paper, we propose techniques to further improve RTFs of the first and the last module without affecting the diarization performance. The techniques avoid the estimation of redundant terms and reduce the complexity of the underlying models. Exhaustive evaluations on the NIST Rich Transcription datasets show relative RTF reductions of 15% while maintaining the performance. Particularly, the posterior extraction stage is optimized to obtain a relative RTF reduction of 56%. Optimization on the different realignment strategies are shown to provide a relative RTF reduction of 67% and 58%.
Index Terms: speaker diarization, information bottleneck approach

1. Introduction

Speaker diarization systems address the problem of *who spoke when* in a given audio recording. The problem is approached in an unsupervised fashion using techniques such as HMM/GMM (Hidden Markov Model/Gaussian Mixture Model) [1] and Information Bottleneck approach based diarization systems [2]. The IB framework based speaker diarization system has been shown to be a faster approach in terms of real time factors (RTF) [3]. The RTF is defined as the ratio of the time taken by the algorithm to the duration of the input. It indicates the suitability of a system for real-time applications. In the context of speaker diarization, in which the input audio is significantly longer than those used in typical ASR systems or speaker verification systems, obtaining systems that have faster than real-time run-times with good performance (in terms of the error rate) is beneficial. The output of diarization systems are also often used as input to other speech analysis systems. In [4], the diarization output is used as an input to the ASR system. Speaker linking is yet another task in which the output of speaker diarization is useful [5]. Such uses motivate the need for fast speaker diarization systems. Several attempts, including [6, 7], can be found in the literature emphasizing the need for fast diarization systems [1].

Thanks to the non-iterative nature of the IB based speaker diarization systems, they are significantly faster than conventional HMM/GMM based diarization systems while yielding similar performance. As a result, the IB based speaker diarization systems have not been optimized so far. In this work, we propose optimization techniques that significantly decrease the run-time of IB based diarization systems.

More specifically, during posterior feature estimation, the covariance of the Gaussians in a GMM are shared and certain computations can therefore be pre-computed and others avoided, thereby yielding 10% relative RTF improvement. Furthermore, two different methods can be applied for the realignment: HMM/GMM based realignment and KL-HMM based realignment. The run-time of both methods can be significantly reduced. Our experiments show that the model complexity as well as the number of iterations for the HMM/GMM realignment can be reduced without noticeable effect on the diarization performance, resulting in a faster realignment. The KL-HMM based realignment method involves computationally expensive logarithm computations. Careful algorithm analysis reveals that many logarithm computations are indeed redundant and can be omitted, leading to a speed-up of 56% relative. Altogether, these optimizations significantly improve the RTF of the diarization system without compromising its diarization performance.

The rest of the paper is organized as follows: Section 2 gives an overview of the IB-based approach to speaker diarization. The computational complexities of the modules involved are discussed. In Section 3, an optimization to posterior computation is proposed and its effects studied. In Section 4, observations on improving the run-time HMM/GMM system are discussed. In Section 5, an efficient optimization to the KL-HMM algorithm is introduced. The combined results are reported in Section 6.

2. Information Bottleneck Approach

The Information Bottleneck (IB) method performs diarization by optimizing the clusters with respect to a set of relevance variables [2]. The optimization criterion is given below:

$$\mathcal{F} = I(Y;C) - \frac{1}{\beta} I(C;X) \quad (1)$$

where X is the feature set, Y is the set of relevance variables and C is the set of clusters. The Lagrangian multiplier β controls the trade-off between information preserved in the clusters and the cluster size. A block diagram of the IB approach is given in Figure 1. The system comprises of many stages: the feature extraction stage is typical to those available in any speech analysis system. A voice activity detector (VAD) segments the audio into speech and non-speech regions. The speech segments are further split into shorter segments, if possible, of length 2.5s. The next three stages are exclusive to the IB approach. In the posterior extraction module, Gaussian parameters are estimated for every segment in the audio and combined

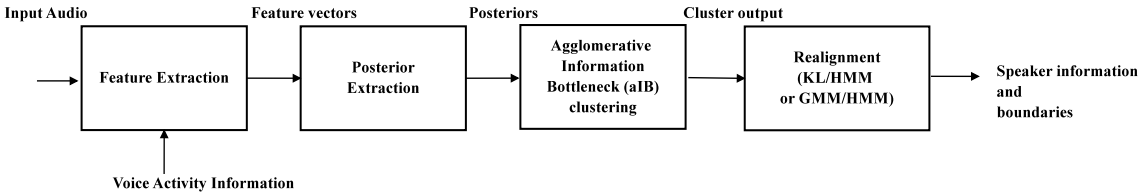

Figure 1: Block diagram representing of the information bottleneck based system

to form a mixture of Gaussians. The GMM thus estimated is used to obtain a posterior vector for every speech frame in the audio file. The posteriors within a segment are combined and used as input to the clustering stage.

The agglomerative IB (aIB) algorithm is a clustering algorithm that uses the IB principle on the input segments represented by the posteriors. The segments, which initially have a maximum length of 2.5 seconds, are clustered together with respect to the difference in Eq. 1. Two candidates are chosen by the clustering algorithm if it results in the decrease in $\mathcal{F}$. The number of clusters at the end of the algorithm are chosen using the Normalized Mutual Information (NMI) criterion, which is a function of the Minimum Description Length (MDL) of the clustering output. The result of the clustering is refined by the realignment module.

The realignment module modifies the boundaries given by the VAD. Two approaches to the realignment may be employed - HMM/GMM approach and KL-HMM approach. The two approaches are compared in [8]. It is shown that KL-HMM is much more beneficial when multiple features are used. Moreover, the KL-HMM approach is scalable with respect to the number of features or feature dimensions. The computational complexity of the algorithm is determined by the number of segments produced by the VAD. Thus, it increases, in general, only with the length of the input audio.

The HMM/GMM based realignment is also a suitable alternative as a realignment module. The clusters provided by the aIB algorithm are used as states in an ergodic HMM as opposed to using uniform segmentation by the conventional HMM/GMM based speaker diarization system. Moreover, in the realignment step in the IB framework, there is no merging of the states. However, GMM-estimation and Viterbi realignment may be run multiple times. In the systems reported in [9, 2], typically 70 mixture GMMs are used with 3 iterations of GMM-estimation and realignment. This is computationally more intensive than the KL-HMM method.

The RTFs of the IB based system using MFCC (Mel Frequency Cepstral Co-efficients) ([10]) features are presented in Table 1. The implementation of the baseline IB based system presented here is described in [11]. The diarization systems use 19-dimensional MFCC vectors extracted from beamformed audio. The results are presented on the benchmark NIST RT 05, RT 06, RT 07 and RT 09 datasets. The run-times are compared with the conventional HMM/GMM based diarization system ([12]) to re-iterate the differences in RTFs. The experiments are run on a machine with Intel(R) Core(TM) i7-3770K CPU @ 3.50GHz, 16 GB RAM and 1 TB SATA HDD. All experiments in this paper are conducted on the same machine. The RTFs values (reported in a scale of 10^{-2}) exclude the time taken to extract feature vectors and speech/non-speech boundaries from a VAD. All RTF values reported are averaged over 10 runs. [13, 14].

Table 2 shows the time taken by each of the blocks specific

Table 1: *Baseline Real Time Factors (RTFs) of HMM/GMM and Information Bottleneck (IB) based system on the NIST RT datasets. The IB system is observed to be faster than the HMM/GMM system. RTFs are in the scale of* 10^{-2}.

System	RTFs				
	RT05	RT06	RT 07	RT 09	Overall
HMM/GMM	7.9	9.4	9.2	8.9	8.8
IB (HMM/GMM)	12.5	18.4	19.0	19.9	17.2
IB (KL-HMM)	2.1	5.3	6.3	7.5	5.0

Table 2: *Baseline Real Time Factors (RTFs) Information Bottleneck (IB) based system on the NIST RT datasets for different modules. RTFs are in the scale of* 10^{-2}.

Module	RTFs				
	RT05	RT06	RT 07	RT 09	Overall
Posterior extraction	0.3	0.6	0.7	0.8	0.6
aIB clustering	1.3	3.7	4.6	5.6	3.6
Realignment strategies					
KL-HMM	0.4	0.9	0.9	1.1	0.8
HMM/GMM	9.8	14.0	13.7	13.5	12.6

to the IB system in Figure 1. The clustering step takes a major amount of the time in the system. This is due to the quadratic nature of the initialization step in the clustering algorithm and the cubic number of comparisons performed before each merge [15].

3. Posterior Computation

As described in the earlier section, the posterior computation procedure involves estimating Gaussian distribution parameters for each speech segment and computing posterior vector with respect to the Gaussians for every feature vector extracted from the audio file. Let n_{seg} be the number of segments in the audio file. Then, n_{seg} Gaussian mean and co-variance matrix parameters are estimated. Let the set of parameters be $\boldsymbol{\theta}_j = \{\pi_i \boldsymbol{\mu}_j, \boldsymbol{\Sigma}\}$ for $j = 1, 2, \ldots, n_{\text{seg}}$, where π_j is the weight, $\boldsymbol{\mu}_j$ is the mean vector and $\boldsymbol{\Sigma}$ is the diagonal covariance matrix of the i^{th} segment (and mixture). The audio file can be represented as a sequence of feature vectors $\mathbf{X} = \{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_N\}$. If the feature dimension is D, then the diagonal entries of Σ can be represented as $\{\sigma_1^2, \sigma_2^2, \ldots, \sigma_D^2\}$ and $\boldsymbol{\mu}_j = (\mu_{j,1}, \mu_{j,2}, \ldots, \mu_{j,D})^t$ and $\mathbf{x}_i = (x_{i,1}, x_{i,2}, \ldots, x_{i,D})^t$, Notice that the covariance matrix is shared between the Gaussians.

A posterior vector is computed from every feature vector. The log probabilities are computed and normalized for every feature vector. The log probability of a feature vector $\mathbf{x}_i$ with respect to the j^{th} mixture is given as follows

$$\ln p(\mathbf{x}_i|\boldsymbol{\theta}_j) = \ln \pi_j \mathcal{N}\left(\boldsymbol{\mu}_j, \boldsymbol{\Sigma}\right) \quad (2)$$

Table 3: *Comparison of the number of floating-point additions and multiplications to compute log posteriors in the baseline and proposed posterior extraction algorithms*

System	Additions	Multiplications
Baseline	$2Nn_{\text{seg}}(D+1)$	$2NDn_{\text{seg}}$
Proposed	$Nn_{\text{seg}}(D+1)$	NDn_{seg}

Table 4: *Real Time Factors (RTFs) of baseline and proposed posterior extraction procedures. RTFs are in the scale of* 10^{-2}.

System	RTFs				
	RT05	RT06	RT 07	RT 09	Overall
Baseline	0.3	0.6	0.7	0.8	0.6
Proposed	0.2	0.5	0.6	0.7	0.5

which can be expanded to

$$\ln p(\mathbf{x}_i|\boldsymbol{\theta}_j) = \ln \pi_j - \frac{D}{2} ln|\boldsymbol{\Sigma}| - \frac{1}{2}\left(\mathbf{x}_i - \boldsymbol{\mu}_j\right)^t \boldsymbol{\Sigma}^{-1}\left(\mathbf{x}_i - \boldsymbol{\mu}_j\right) \tag{3}$$

where $|\boldsymbol{\Sigma}|$ is the determinant of the matrix $\boldsymbol{\Sigma}$.

The log posterior for every feature vector is normalized by subtracting the maximum log posterior value among all mixtures and computing their exponent. The values are then normalized to sum to 1 for a feature vector.

The terms in the log posterior computation can be segregated to mixture-dependent, feature-dependent and co-dependent terms.

$$\ln p(\mathbf{x}_i|\boldsymbol{\theta}_j) = T_j + T_i - T_{i,j} \tag{4}$$

where

$$T_j = \ln \pi_j - \frac{D}{2}|\boldsymbol{\Sigma}| + \frac{1}{2}||\boldsymbol{\Sigma}^{-1}\boldsymbol{\mu}||_2^2 \tag{5}$$

$$T_i = \frac{1}{2}||\boldsymbol{\Sigma}^{-1}\mathbf{x}||_2^2 \tag{6}$$

$$T_{i,j} = \sum_d x_{i,d}\, \mu_{j,d}\, \sigma_d^{-2} \tag{7}$$

$||.||_2$ signifies the L_2 norm. Since posteriors are eventually normalized, T_i can be ignored. Thus, the number of additions and multiplications are significantly reduced. These are compared in Table 3. The baseline, which is the direct implementation of Eq. 3, requires $2N(D+1)n_{\text{seg}}$ additions: for each of N feature vectors n_{seg} posteriors are required. For each of these posteriors, D additions of $(x-\mu)$ and D additions of Mahalanobis distances for each mixture are performed. And there are Nn_{seg} additions of the mixture-specific constant term T_i. There are $2ND$ multiplications: for each of Nn_{seg} posterior values, D multiplications to compute the square of $x-\mu$ and D further multiplications between the results and the precision σ^{-2} are performed. In the optimized approach, only a dot product needs to be computed, which has for each of Nn_{seg} posterior values, D additions and D multiplications. Further Nn_{seg} additions are required to add T_i to the posteriors appropriately.

Table 4 presents the baseline and improved RTFs as a result of the simplification. The RTF improves by approximately 10% in relative terms without any changes to the output.

4. HMM/GMM realignment

In IB based systems, two types of realignment procedure can be used. The HMM/GMM realignment procedure uses the clustering output from the aIB algorithm and represents each cluster by one of the states of a HMM. In [9] and [11], each state is modelled by a 70-mixture GMM (optimized for performance). The procedure includes 3 iterations of GMM estimations and Viterbi segmentation. It has been shown that the GMM/HMM realignment procedure can be beneficial when using only one feature (as opposed to using multiple features) [8].

Table 5: *Comparison of Speaker Error Rates (SER) and RTFs for different number of mixtures in HMM/GMM realignment.RTFs are in the scale of* 10^{-2}.

System	SER/RTF				Overall
(No. of mixtures)	RT05	RT06	RT 07	RT 09	RTF
70 (baseline)	19.7/7.0	17/9.0	11.7/8.8	21.2/8.6	8.2
32	19.6/2.3	17.1/3.1	12.0/3.0	21.1/3.0	2.9
16	19.6/1.2	17.8/1.7	13.1/1.6	21.4/1.6	1.5
8	19.8/0.7	18.2/0.9	14.7/0.9	22.1/0.9	0.8

Results presented in this section show that, when using the HMM/GMM realignment approach, the number of mixtures of the GMMs representing a state can be smaller than 70 mixtures. The number of speakers in meeting recordings do not warrant 70 mixtures and the results suggest that the confusability is not entirely avoided either. Thus, using fewer mixtures is to reduce the run-time is proposed. Furthermore, it is often observed that multiple iterations of estimation-and-segmentation procedure propagate the error and thus can be avoided. These observations are also corroborated in [16].

4.1. Number of iterations

First, the case for reducing the number of iterations is presented. When the HMM-states are represented with 70 mixtures, the increase in error rate is observed to be 0.2% per iteration across all NIST RT datasets per iteration (varied from 1 to 3). As the number of mixtures are reduced to 16, the error rate increases by 0.3% per iteration across the same dataset. Thus, keeping the number of iterations to a minimum is beneficial.

4.2. Number of mixtures

The average number of speakers participating in the meetings considered is 5. Also, when the goal is to reduce the runtime of the system while still maintaining the diarization performance, studying the effect of varying the number of mixtures of the GMM can be useful. Experiments are conducted on NIST RT datasets to reduce the number of mixtures while maintaining the performance of the system. Table 5 discusses these results. Only speaker error rate (SER) is reported as miss and false alarm speech are the same across the systems being compared. It can be seen that even the number of mixtures used is only 16, there is mostly negligible increase in error rate (a worst case of 1%). Thus, it is certainly beneficial to reduce the number of mixtures when using the HMM/GMM realignment procedure. The corresponding RTFs when using different number of mixtures is also given in the table. Significant reductions can be observed.

5. KL-HMM realignment

The KL-HMM realignment method, introduced in [8], is observed to be particularly useful when using multiple features, for instance when combining MFCC and TDOA features. The realignment method reuses the posteriors produced in the first stage of the diarization process. Each cluster is represented by the average of the posteriors of the features in that cluster. The audio is now segmented through Viterbi decoding with respect to the KL-divergence between the posterior of the feature vectors and the mean posterior of the clusters. Let $\mathbf{z}_n$ be the

Table 6: *Real Time Factors (RTFs) of baseline and proposed KL-HMM procedures. RTFs are in the scale of* 10^{-2}.

System	RTFs				
	RT05	RT06	RT 07	RT 09	Overall
Baseline	0.4	0.9	0.9	1.1	0.8
Proposed	0.2	0.4	0.4	0.4	0.35

posterior vector for the n^{th} feature vector in the audio. Let $\mathbf{y}_1, \mathbf{y}_2, \ldots, \mathbf{y}_C$ be the set of mean posteriors for C clusters obtained at the output of aIB clustering. The KL-divergence measure between $\mathbf{z}_n$ and $\mathbf{y}_c$ is given by

$$KL(\mathbf{z}_n, \mathbf{y}_c) = -\sum_{l=1}^{n_{\text{seg}}} z_{n,l} \log \frac{z_{n,l}}{y_{c,l}} \tag{8}$$

Viterbi decoding is applied on the measures obtained in the above equation.

A direct implementation of the above strategy involves multiple calls to logarithmic function. For N feature vectors and n_{seg} segments, Nn_{seg} calls to the log function are made even if $\log y_{c,l}$ is pre-computed to avoid floating point division. The speed of this method is therefore dependent on the number of segments produced by the VAD module. Therefore, an optimization of the KL-divergence computation is proposed that avoids almost all calls to the log function without any change to the end-result thereby giving a tremendous speed-up of the realignment process. This in turn results in an overall speed-up of the entire diarization process.

Proposed optimization: In Equation 8, the KL-divergence measure can be further simplified by observing that the term $z_{n,l} \log z_{n,l}$ is a constant across all clusters. The output of the Viterbi decoding process will not change when this constant removed from the measure. Thus, it suffices to reduce the measure to

$$\hat{KL}(\mathbf{z}_n, \mathbf{y}_c) = \sum_{l=1}^{C} z_{n,l} \log y_{c,l} \tag{9}$$

In the above simplification, the NC calls to logarithms on the posteriors are entirely avoided (ignoring the pre-computing of $\mathbf{y}_c \forall c$). This simplification is applicable to the system that uses KL-HMM as its component. For instance, it can be used in KL-HMM based ASR systems as well [17, 18].

Table 6 compares the change in the RTFs between the baseline (direct implementation) and proposed approach. A significant improvement in the RTFs are observed across all datasets and the averaged RTF improves by 56%. There is no change in SER. Thus, the improvement is extremely beneficial.

An important observation is that the run-time of the KL-HMM algorithm will not change when multiple features are used by the system. This is because the posteriors are computed at the initial stage of the diarization process. Therefore, when using multiple features the KL-HMM is much more suitable for fast speaker diarization compared to the HMM/GMM based realignment both in terms of speed and accuracy.

6. Results

The proposed optimizations are combined and the overall improvements to the system are presented in Table 7. The overall RTF and SER across all datasets are reported. The overall SER refers to the error rate across all the 34 files in the NIST RT 05 to 09 datasets. The KL-HMM based IB system shows an improvement of approximately 14% in RTF with no deterioration in performance. The optimizations proposed for the HMM/GMM based IB system provides a relative improvement of approximately 67% in the case where 16 mixture GMMs are used to represent each state. The performance drops by 0.6% in absolute terms. In the case of 32 mixture GMMs, the RTF improves by approximately 58% in relative terms with a deterioration in performance by only 0.2% (absolute).

Table 7: *Average RTFs and SERs of all the approaches presented in the paper. RTFs are in the scale of* 10^{-2}. *The systems are identified by the realignment method used.*

System	Average RTF/SER	
	MFCC	MFCC + TDOA
Baseline		
KL-HMM	5.0/17.4	5.3/8.4
HMM-GMM	17.0/17.2	17.3/8.4
Proposed		
KL-HMM realignment	4.3/17.4	4.5/8.4
HMM/GMM (16 mixtures)	5.6/17.8	5.8/8.9
HMM/GMM (32 mixtures)	7.0/17.4	7.3/8.8

The scalability of the system is tested with the inclusion of TDOA based features. The TDOA features have been shown to be adding sufficient complementary information [9]. A weight of 0.8 on MFCC and 0.2 on TDOA are utilized while combining the posteriors before aIB clustering. In the case of HMM/GMM realignment, the states of the HMM for the TDOA features are represented by 3 mixture GMMs. The results of the experiments are included in Table 7 alongside the MFCC results. For the KL-HMM system, a relative improvement of approximately 15% is observed for the RTF, which is close to the improvement observed in the case of MFCC-based system. This shows that the optimization is scalable. The IB system using HMM/GMM realignment with fewer mixtures (16 and 32) of GMMs are also observed to be scalable when using multiple features. The results, however, show that there are sufficient advantages to use the KL-HMM over the HMM/GMM for realignment even after applying the multiple optimizations proposed. The IB system with KL-HMM realignment is faster by 23% relative when compared to using the HMM/GMM realignment with only 16 mixtures.

7. Summary

Techniques to improve the run-time of the IB based speaker diarization system are presented. Optimizations to the posterior extraction stage exploit the redundancy in the computation of log posteriors results in the RTF relatively improving by approximately 10%. The HMM/GMM system's run-time is optimized by significantly reducing the number of mixtures of the GMM and avoiding re-iterations of the decoding process. Improvements of 58% and 67% are observed for RTFs in the cases of 32-mixture and 16-mixture GMMs. The KL-HMM based segmentation algorithm is optimized by reducing the redundancy in the computation of the KL-divergence measure. This is shown to improve the RTF of the algorithm relatively by 56%.

8. Acknowledgements

The authors thank the Swiss National Science Foundation for their financial support for this project through the National center of Competence in Research on "Interactive Multimodal Information Management", and more specifically through the sub-project DIMHA (Diarizing Massive Amounts of Heterogeneous Audio).

9. References

[1] X. Anguera Miro, S. Bozonnet, N. Evans, C. Fredouille, G. Friedland, and O. Vinyals, "Speaker diarization: A review of recent research," *IEEE Tran. on Audio, Speech, and Language Processing*, vol. 20, no. 2, pp. 356–370, 2012.

[2] D. Vijayasenan, F. Valente, and H. Bourlard, "An information theoretic approach to speaker diarization of meeting data," *IEEE Tran. on Audio, Speech, and Language Processing*, vol. 17, no. 7, pp. 1382–1393, 2009.

[3] D. Vijayasenan, "An information theoretic approach to speaker diarization of meeting recordings," Ph.D. dissertation, ÉCOLE POLYTECHNIQUE FÉDÉRALE DE LAUSANNE, 2010.

[4] A. Stolcke, G. Friedland, and D. Imseng, "Leveraging speaker diarization for meeting recognition from distant microphones," in *Acoustics Speech and Signal Processing (ICASSP), 2010 IEEE International Conference on*. IEEE, 2010, pp. 4390–4393.

[5] M. Ferras and H. Bourlard, "Speaker Diarization and Linking of Large Corpora," 2012.

[6] G. Friedland, J. Chong, and A. Janin, "A parallel meeting diarist," in *Proceedings of the 2010 International workshop on Searching spontaneous conversational speech*. ACM, 2010, pp. 57–60.

[7] G. Friedland *et al.*, "Parallelizing speaker-attributed speech recognition for meeting browsing," in *IEEE International Symposium on Multimedia (ISM)*, 2010, pp. 121–128.

[8] D. Vijayasenan *et al.*, "KL realignment for speaker diarization with multiple feature streams," in *INTERSPEECH*, 2009, pp. 1059–1062.

[9] D. Vijayasenan, F. Valente, and H. Bourlard, "An information theoretic combination of mfcc and tdoa features for speaker diarization," *IEEE Tran. on Audio, Speech, and Language Processing*, vol. 19, no. 2, pp. 431–438, 2011.

[10] Davis, S. and Mermelstein, P., "Comparison of parametric representations for monosyllabic word recognition," vol. 28(4). IEEE Trans. Acoust., Speech, Signal Processing, 1980, pp. 357–366.

[11] D. Vijayasenan and F. Valente, "Diartk: An open source toolkit for research in multistream speaker diarization and its application to meetings recordings." in *INTERSPEECH*, 2012.

[12] C. Wooters and M. Huijbregts, "The icsi rt07s speaker diarization system," in *Multimodal Technologies for Perception of Humans*. Springer, 2008, pp. 509–519.

[13] J. M. Pardo, X. Anguera, and C. Wooters, "Speaker diarization for multiple distant microphone meetings: mixing acoustic features and inter-channel time differences." in *In Proc. of INTERSPEECH*, 2006.

[14] X. Anguera, "Beamformit (the fast and robust acoustic beamformer)," http://www.xavieranguera.com/beamformit/.

[15] D. Vijayasenan, F. Valente, and H. Bourlard, "Agglomerative information bottleneck for speaker diarization of meetings data," in *IEEE Workshop on Automatic Speech Recognition & Understanding, 2007*. IEEE, 2007, pp. 250–255.

[16] M. Sinclair and S. King, "Where are the challenges in speaker diarization?" in *IEEE Int. Conf. on Acoustics, Speech and Signal Processing (ICASSP)*, 2013, pp. 7741–7745.

[17] D. Imseng, R. Rasipuram, and M. Magimai-Doss, "Fast and flexible kullback-leibler divergence based acoustic modeling for non-native speech recognition," in *Proceedings of the IEEE workshop on Automatic Speech Recognition and Understanding*, Dec. 2011, pp. 348–353.

[18] G. Aradilla, J. Vepa, and H. Bourlard, "An acoustic model based on kullback-leibler divergence for posterior features," in *IEEE Int. Conf. on Acoustics, Speech, and Signal Processing (ICASSP)*, vol. 4, 2007, pp. 657–660.