UBM Fused Total Variability Modeling for Language Identification

Maarten Van Segbroeck, Ruchir Travadi and Shrikanth S. Narayanan

Signal Analysis and Interpretation Lab,
University of Southern California, Los Angeles, USA
{maarten, shri}@sipi.usc.edu, travadi@usc.edu

Abstract

This paper proposes Universal Background Model (UBM) fusion in the framework of total variability or i-vector modeling with the application to language identification (LID). The total variability subspace which is typically exploited to discriminate between the language classes of different speech recordings, is trained by combining the normalized Baum-Welch statistics of multiple UBMs. When the UBMs model a diverse set of feature representations, the method yields an i-vector representation which is more discriminant between the classes of interest. This approach is particularly useful when applied to short-duration utterances, and is a computationally less complex alternative to performance boosting as compared to system level fusion. We assess the performance of UBM fused total variability modeling on the task of robust language identification on short-duration utterances, as part of Phase-III of the DARPA RATS (Robust Automatic Transcription of Speech) program.

Index Terms: language identification, i-vector representation, short-duration, noise robustness, RATS

1. Introduction

Language identification (LID) is defined as the task of identifying the spoken language from audio recordings. Over the years, several machine learning approaches have been proposed in the development of automatic language identification systems. Phonotactic systems perform LID by employing a phone recognizer that converts speech signals into a sequence of phone symbols or tokens, followed by a language model (PRLM) to extract phonotactic information from the token strings statistics [1–4]. Alternative approaches to LID attempt to classify languages by using Gaussian mixture models (GMMs) which capture the acoustic properties of speech. The potential of GMM-based language identification was shown in [5, 6] and significant progress in LID performance was made by employing supervector modeling [7] and the introduction of Joint Factor Analysis (JFA) [8, 9]. JFA attempts to reduce the variability caused by different channels and sessions in order to retain a subspace that captures the variability of the desired factor of interest, e.g. the spoken language. Although originally applied to the problem of speaker verification, the factor analysis formulation can be generalized to other application domains such as language identification [10].

The method of JFA has led to its successful variant, namely total variability or i-vector modeling, which was introduced in [11] and became popular due to its excellent performance, reduced complexity and small model size. The success of performing LID in the i-vector framework, has been shown in [12–15]. In the work of [15, 16], the i-vector approach has been extended towards the simplified and supervised i-vector framework by allowing label-regularized i-vector training and by pre-normalizing the first order Baum-Welch statistics during the factor analysis.

This paper builds on top of the simplified i-vector framework by the introduction of UBM-fused total variability modeling, with the main goal to improve the systems' performance. We will show that, when the UBMs are trained on diverse feature representations, the extracted Baum-Welch statistics are more equally distributed along the Gaussian components of the UBM. Moreover, the UBM-fused statistics will exhibit less redundancy compared to a single UBM of the same size trained on a single representation. This leads to an improved i-vector extraction, particularly in situations where the available data is limited, such as is the case for short-duration sentences.

Although generally applicable, the proposed method will be applied as part of the Phase-III LID Evaluation in the DARPA Robust Automatic Transcription of Speech (RATS) program. In the RATS program, the main goal is to accurately separate the target speech from interfering background sources, to identify the language and the speaker, and to apply keyword detection on a data corpus that consists of highly degraded speech recordings that were transmitted over noisy radio communication channels [17]. For the task of LID, the major challenge is the development of a system that is robust under various noisy conditions when applied on utterances of different length. In this paper, we focus on the short-duration sentence task for which the test set contains utterances with a duration of 3 and 10 seconds. The challenging task of accurate i-vector based classification on short-duration sentences has also been studied in [13, 18–21] and in [22] where the problem was addressed differently by modifying the prior distribution of the i-vectors.

The remainder of the paper is organized as follows: Section 2 restates the framework of total variability modeling, the simplified variant with prenormalized first order Baum-Welch statistics and will introduce the UBM-fused total variability modeling. The application of the latter on the RATS LID corpus will be discussed in Section 3. Experimental results for the short-duration test sets are given in Section 4. Finally, Section 5 concludes.

2. UBM-fused Total Variability Modeling

Total variability or i-vector modeling, originally proposed in [11] is based on and motivated by the technique of Joint Factor Analysis (JFA) [9] which was originally applied on the task of speaker verification. JFA attempts to capture both the variability of the desired factor of interest (e.g. speaker, language, etc.) and the undesired channel or session variability in two distinct eigenspaces of low dimensionality. For real life speech signals, the assumption of zero mutual information between the two subspaces does not hold and hence important information of the desired factor could be lost in the session eigenspace. This type of information loss is prevented in the total variability framework by estimating a single low dimensional subspace, i.e. the identity or i-vector space, that models all variability together. To compensate for the undesired variability due to the presence of the undesired factors, variability compensation methods such as Within-Class Covariance Normalization (WCCN) [23], Linear Discriminative analysis (LDA) and Nuisance Attribute Projection (NAP) [7], are typically applied within the i-vector space.

2.1. Total Variability Modeling

The first step in training a class-specific i-vector system is to train a prior model that represents general and relevant acoustic characteristics of speech. To this end, a Gaussian Mixture Model (GMM) is trained independent of the classes using all available training data and is commonly referred to as the Universal Background Model (UBM).

Let us now define a UBM with C Gaussian mixture components as $\boldsymbol{\Lambda} = \{\boldsymbol{\lambda}_1, \boldsymbol{\lambda}_2, \ldots, \boldsymbol{\lambda}_C\}$ where each mixture component is characterized by $\boldsymbol{\lambda}_c = \{\rho_c, \boldsymbol{\mu}_c, \boldsymbol{\Sigma}_c\}$ with mixture weight ρ_c, Gaussian mean $\boldsymbol{\mu}_c$ and (diagonal) covariance matrix $\boldsymbol{\Sigma}_c$. Applying the total variability approach to the problem of language identification implies to write each utterance j as a language- and session-dependent supervector $\mathbf{M}_j$:

$$\mathbf{M}_j = \mathbf{m} + \mathbf{T}\mathbf{w}_j \tag{1}$$

where $\mathbf{m}$ is a language- and session-independent supervector constructed from stacking the Gaussian mean vectors of all C UBM components. The total variability matrix $\mathbf{T}$ spans a low-dimensional total variability subspace of rank K. The i-vector of the utterance is then given by a normally distributed vector $\mathbf{w}_j$ containing the corresponding K total factors. Note that the probability function of supervectors (1) given $\mathbf{w}_j$ is Gaussian distributed with mean $\mathbf{m} + \mathbf{T}\mathbf{w}_j$ and covariance matrix denoted by $\boldsymbol{\Sigma}$ which explains the residual variability not captured in the eigenspace defined by the column vectors of $\mathbf{T}$.

Let $\mathbf{y}_{j,t}$ denote the D-dimensional feature vector at a time frame t of utterance j. The zeroth order Baum-Welch statistics of the utterance for a UBM mixture component $\boldsymbol{\lambda}_c$ are given by

$$N_j^c = \sum_{t=1}^{T} P(c|\mathbf{y}_{j,t}, \boldsymbol{\lambda}_c) \tag{2}$$

where the sum of the occupancy probabilities is taken over all T frames that are present in the utterance. Similarly, the centralized first order Baum-Welch statistics are computed as

$$\mathbf{F}_j^c = \frac{1}{N_j^c} \sum_{t=1}^{K} P(c|\mathbf{y}_{j,t}, \boldsymbol{\lambda}_c)(\mathbf{y}_{j,t} - \boldsymbol{\mu}_c). \tag{3}$$

Rearranging the statistics (2)-(3) over all C mixture components, we stack the vectors $\mathbf{F}_j^c$ into the supervector $\mathbf{F}_j$, and we define the $CD \times CD$ diagonal matrices $\mathbf{N}_j$ which are composed of C diagonal blocks of respectively $N_j^c\mathbf{I}$. The total variability framework (1) can now be restated as

$$\mathbf{F}_j = \mathbf{T}\mathbf{w}_j \tag{4}$$

where

$$\begin{aligned} P(\mathbf{w}_j|\boldsymbol{\Lambda}) &= N(\mathbf{0}, \mathbf{I}) \\ P(\mathbf{F}_j|\mathbf{w}_j, \boldsymbol{\Lambda}) &= N(\mathbf{T}\mathbf{w}_j, \mathbf{N}_j^{-1}\boldsymbol{\Sigma}) \end{aligned} \tag{5}$$

Note that the distribution of $\mathbf{F}_j$ is conditioned on $\mathbf{w}_j$ and the UBM $\boldsymbol{\Lambda}$.

The total variability matrix $\mathbf{T}$ is then iteratively trained by the EM-algorithm described in [24] for only one factor in the JFA and by considering each training utterance as being produced by a new speaker. The Expectation-step involves the computation of the posterior probability $P(\mathbf{w}_j|\mathbf{F}_j, \boldsymbol{\Lambda})$ using Bayes' rule and (5). The estimated i-vectors are then explained as the expected values of the probability function and given as

$$\mathbf{w}_j = \boldsymbol{\beta}_j^{-1}\mathbf{T}'\boldsymbol{\Sigma}^{-1}\mathbf{N}_j\mathbf{F}_j \tag{6}$$

with

$$\boldsymbol{\beta}_j = \mathbf{I} + \mathbf{T}'\boldsymbol{\Sigma}^{-1}\mathbf{N}_j\mathbf{T}. \tag{7}$$

The Maximization-step of the EM algorithm updates matrices $\mathbf{T}$ and $\boldsymbol{\Sigma}$ such that the global likelihood defined over all training utterances is maximized. The updated matrices are found by linear regression using the estimated i-vectors (6) as explanatory variables [8]. The total variability matrix $\mathbf{T}$ is randomly initialized, while $\boldsymbol{\Sigma}$ is initialized by the covariance matrices of the UBM.

Experimental evidence of the benefit of total variability modeling as compared to JFA was given in [11] for the task of speaker verification, while [14] presents its benefits when applied in language identification. The iterative EM procedure of the training stage and the i-vector extraction are dominated by the computationally expensive matrix products of (6) and (7). This yields a complexity of $O(K^3 + CK^2 + CDK)$ for each utterance that is evaluated.

2.2. Simplified i-Vector training

The framework of total variability modeling has been extended in [15, 16] to a computationally more attractive version, named simplified i-vector modeling, by exploiting a pre-normalization step during the factor analysis. This pre-normalization was done by re-weighting the first order Baum-Welch statistics (3) as follows:

$$\mathbf{F}_j^c \leftarrow \sqrt{\frac{N_j^c}{n_j}}\mathbf{F}_j^c \tag{8}$$

with $n_j = \sum_{c=1}^{C} N_j^c$. This way, the occupancy probabilities N_j^c are factored out from the covariance matrix of $\mathbf{F}_j$, now computed as $n_j\boldsymbol{\Sigma}$, which results to all supervector dimensions being equally treated in the i-vector modeling. Thanks to this approximation, the matrix multiplications that involve $\mathbf{N}_j$ in (6) and (7) are replaced by the scaling factor n_j. This reduces the total complexity to $O(K^3 + CDK)$.

To avoid the costly matrix inversion of $\boldsymbol{\beta}_j$ during training, the use of a precomputed cache table was also proposed in [16]. The table is updated at each iteration after the M-step and decodes the matrix product $\boldsymbol{\beta}_j^{-1}\mathbf{T}'\boldsymbol{\Sigma}^{-1}$ into a set of table entries that are selected based on their value for n_j. It was shown that a limited quantization error can be assured for a table size of the order of a few hundred entries, which is typically much smaller than the number of utterances in the training set. The table lookup strategy further reduces the complexity in training mode to $O(CDK)$.

As shown in [15], the simplification of the i-vector systems slightly reduces the performance of the conventional i-vector baseline. However, the sacrifice in performance is often neglible and tolerated given the measured speed increase of more than 100 times compared to the baseline.

2.3. UBM-fused i-Vector training

The prenormalization strategy of Section 2.2 allows a straightforward implementation of the UBM fused Total Variability modeling. Let L denote the total number of UBMs to be jointly exploited in the i-vector training, then the supervector $\mathbf{F}_j$ of (4) is redefined as:

$$\mathbf{F}_j \leftarrow \left[\mathbf{F}_1^\top, \mathbf{F}_2^\top, \ldots, \mathbf{F}_L^\top\right]^\top \tag{9}$$

by combined stacking of the UBMs first order Baum-Welch statistics. Secondly, all UBM-specific components of supervector (9) are re-weighted by applying formula (8), with n_j summed over all C_l UBM components, hence

$$n_j = \sum_{l=1}^{L} \sum_{c_l=1}^{C_l} N_j^{c_l}. \tag{10}$$

Finally, the system is trained by the EM-procedure of 2.1, but with $\mathbf{\Sigma}$ initialized from the covariance matrices of all Gaussian components in (9). Note that when all UBMs have a number of $C_l = C/L$ components, the computational complexity during training remains $O(CDK)$, while the total complexity is reduced by a factor L when compared to conventional system level fusion of the same number of systems.

UBM fusion significantly improves the LID accuracy, as will be experimentally shown in Section 4. The explanation for this effect lies in the estimation of a better, more discriminant i-vector space, which is now derived from the BW-stats of a fused UBM. The fusion allows to exploit the diversity of multiple features, while jointly increasing the number of dominant (and non-redundant) UBM components in the utterance.

3. Language Identification Experiments

3.1. Data Corpus

The performance of the proposed LID system was evaluated on the DARPA Robust Automatic Transcription of Speech (RATS) data corpus [17]. The Linguistic Data Consortium (LDC) collected audio recordings of five target languages (Arabic, Dari, Farsi, Pashto, and Urdu) and 10 non-target languages. The recordings were about 2 minutes long and were retransmitted through 8 different radio communication channels, introducing various aspects of speech degradation. A training and development set of these retransmitted data was distributed by the LDC to all participants of the DARPA RATS program. The official development set, denoted by DEV-2, was split into four test sets where each utterance corresponds to a speech segment of duration 120, 30, 10 or 3 seconds. In this work, we will only focus on the short durations of DEV-2, i.e. 10 and 3 seconds, while parameter tuning is applied on a distinct validation set, i.e. the TEST set of [13, 25].

3.2. Front-end processing and Background Modeling

Prior to the front-end feature extraction, all audio files are denoised by standard Wiener Filtering [26] and trimmed to speech-only audio segments by applying Voice Activity Detection (VAD) [27].

We will use three standard features, i.e. Mel-Frequency Cepstral Coefficients (MFCC) [28], Perceptual Linear Prediction (PLP) coefficients [29], Gammatone Frequency Cepstral Coefficients (GFCC) [30]. Each representation yields a 44-dimensional feature vector by adding first order derivatives to 22 static components. A fourth feature representation is constructed by combining the speech streams of [27] into a single feature vector, that combines a Gammatone filtered power spectrum with acoustic modulations extracted by Gabor filters [31], and measure streams of voicing and long-term spectral variability. For better UBM modeling, these features are subsequently decorrelated and dimensionality reduced to 88 feature dimensions by the application of Principal Component Analysis (PCA). We will refer to this representation as the Fused Speech Stream (FuSS) feature. All features are mean variance normalized. For each feature representation, UBMs of 2048 and 512 components are trained which are respectively used for the single UBM and the UBM-fused systems.

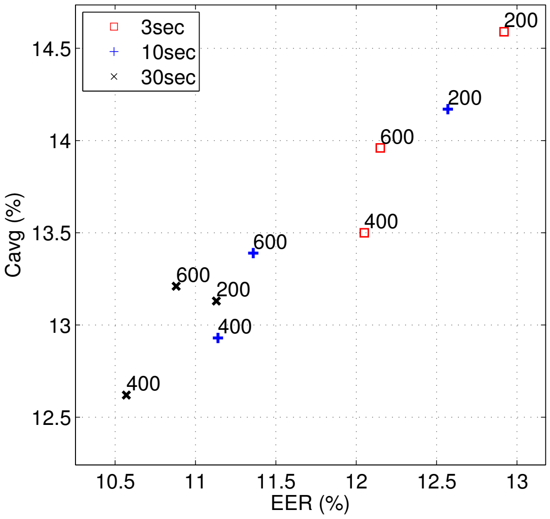

Figure 1: EER versus Cavg for the UBM-fused i-vector system evaluated on the 3 seconds TEST set. The system was trained on training utterances of 3, 10 and 30 seconds duration. Different i-vector dimensions of 200, 400 and 600 were used to tune this value for reporting on the DEV-2 set.

3.3. LID System Details

The UBM-fused i-vector system of Section 2 is trained using the strategy of label-regularized supervision [16]. We apply WCCN on the i-vector space to suppress undesired variability. A duration-specific Support Vector Machine (SVM) is finally trained on the normalized i-vectors to obtain the language output probabilities for each utterance of TEST and DEV-2. The SVM uses a 5th order polynomial kernel.

4. Evaluation

The performance of the systems is evaluated in terms of equal error rate (EER), minimum average cost (Cavg) and minimum detection cost function (DCF) as proposed by NIST [32].

The presented UBM-fused system was trained on respectively 3, 10 and 30 sec duration utterances. The EER against Cavg was first evaluated on the 3 sec TEST utterances and is given by Figure 1. The figure clearly shows that the best performance is obtained in the case of longer duration training. The dimension of the extracted i-vectors, i.e. the rank K of the total variability matrix, was tuned on the TEST set and its constellation was also shown in Figure 1. The plot suggests that a near-optimal system performance is obtained for $K = 400$.

Table 1 shows the evaluation metrics on DEV-2 for both the 3 sec and 10 sec duration task. Rows 1-4 corresponds to systems using a single UBM model trained on either MFCCs, GFCCs, PLPs of FuSS features as mentioned in Section 3.2. All systems

System	Feature(s)	UBM	*DEV-2 - 3 sec*									*DEV-2 - 10 sec*					
			3 sec training			*10 sec training*			*30 sec training*			*10 sec training*			*30 sec training*		
			EER	Cavg	DCF	EER	Cavg	DCF	EER	Cavg	DCF	EER	Cavg	DCF	EER	Cavg	DCF
1	PLP	1×2048	19.58	20.29	19.41	17.36	17.65	16.76	16.58	17.74	16.30	13.62	15.43	13.23	12.80	14.98	12.53
2	MFCC	1×2048	17.89	18.90	17.81	17.10	17.94	16.64	16.45	17.51	16.15	12.43	14.26	11.94	11.97	13.93	11.63
3	GFCC	1×2048	17.75	18.52	17.31	16.58	17.39	16.06	16.19	17.27	15.84	11.97	14.45	11.75	11.52	13.39	11.17
4	FuSS	1×2048	16.84	17.38	16.55	15.40	15.73	14.72	15.01	15.76	14.41	10.97	13.29	10.81	10.60	12.90	10.33
5	*system fusion 1-4*		14.36	14.90	13.71	13.45	13.74	12.43	13.05	14.27	12.58	10.15	12.25	9.80	9.23	13.39	11.17
6	*UBM-fusion*	4×512	14.49	15.08	13.72	13.19	14.21	12.75	**12.01**	**13.80**	**11.89**	9.69	11.96	9.58	**9.14**	**11.61**	**8.99**

Table 1: EER, Cavg and DCF metrics (all in %) for different LID systems evaluated on the DEV-2 test. The performance of system level fusion of the individual systems is compared to the proposed UBM-fused system. Numbers are shown for different durations of training utterances and tested on the DEV-2 short-duration utterances of 3 and 10seconds.

that use standard features have comparable results, while the FuSS features clearly stands out and indicate the potential of this representation for robust speech processing purposes.

Linear combination of the SVM output probabilities of these four systems results in the system fusion scores, that are given by row 5. Row 6 presents the performance of the proposed UBM-fused system which jointly exploits 4 UBMs of 512 components each. Hence, the derived UBM statistics for all systems are computed on the same number of Gaussian components. For the case of long-duration (30 sec) training, UBM-fusion shows a consistent improvement over system level fusion, showing the potential of the method. The results are promising given the fact that only one i-vector space, hence also one SVM training, is required as opposed to system level fusion. However not experimentally derived, additional improvements could be expected when the systems are trained on even longer duration such as 120 seconds of the RATS training corpus.

The reported EER of 12.01% on the 3 sec DEV-2 set relatively improves our previous RATS Phase-II system [15] with 22.3%, and shows to be very competitive to the state-of-the-art LID performance obtained by score fusion of multiple systems [13, 21].

5. Conclusions

We introduced UBM-fused total variability modeling and applied it to the task of language identification. The method extends the simplified i-vector approach and achieves a high performance with low complexity. We showed that fusing UBMs, each trained on a different feature representation, results in the estimation of an i-vector space that is more discriminant between the classes of interest. The approach is most successful when the feature representations are more diverse and appears be very effective when the duration of the test utterances is short. Results were given as part of the Phase-III RATS LID Evaluation of DARPA and illustrate the potential of the proposed method on the short-duration test sets as compared to system level fusion.

6. Acknowledgments

The authors wish to thank Ming Li of SYSU-CMU Joint Institute of Engineering for software distribution and useful discussions and Mary Francis for her devotion and help in all SAIL research efforts. This research was supported by the Defense Advanced Research Projects Agency (DARPA) and the National Science Foundation (NSF).

7. References

[1] M. A. Zissman, “Language identification using phoneme recognition and phonotactic language modeling,” in *Proc. ICASSP*, vol. 5. IEEE, 1995, pp. 3503–3506.

[2] Y. Yan and E. Barnard, “An approach to automatic language identification based on language-dependent phone recognition,” in *Proc. ICASSP*, vol. 5. IEEE, 1995, pp. 3511–3514.

[3] E. Singer, P. A. Torres-Carrasquillo, T. P. Gleason, W. M. Campbell, and D. A. Reynolds, “Acoustic, phonetic, and discriminative approaches to automatic language identification,” in *Proc. Interspeech*, 2003.

[4] P. Matejka, P. Schwarz, J. Cernockỳ, and P. Chytil, “Phonotactic language identification using high quality phoneme recognition,” in *Proc. Interspeech*, 2005, pp. 2237–2240.

[5] P. A. Torres-Carrasquillo, D. A. Reynolds, and J. Deller Jr, “Language identification using gaussian mixture model tokenization,” in *Proc. ICASSP*, vol. 1. IEEE, 2002, pp. I–757.

[6] E. Wong and S. Sridharan, “Methods to improve gaussian mixture model based language identification system,” in *Proc. Interspeech*, 2002.

[7] W. M. Campbell, D. E. Sturim, D. A. Reynolds, and A. Solomonoff, “SVM based speaker verification using a GMM supervector kernel and NAP variability compensation,” in *Proc. ICASSP*, vol. 1, 2006.

[8] P. Kenny, G. Boulianne, and P. Dumouchel, “Eigenvoice modeling with sparse training data,” *Speech and Audio Processing, IEEE Transactions on*, vol. 13, no. 3, pp. 345–354, 2005.

[9] P. Kenny, G. Boulianne, P. Ouellet, and P. Dumouchel, “Joint factor analysis versus eigenchannels in speaker recognition,” *Audio, Speech, and Language Processing, IEEE Transactions on*, vol. 15, no. 4, pp. 1435–1447, 2007.

[10] F. Verdet, D. Matrouf, J.-F. Bonastre, and J. Hennebert, “Factor analysis and svm for language recognition.” in *Proc. Interspeech*, 2009, pp. 164–167.

[11] N. Dehak, P. J. Kenny, R. Dehak, P. Dumouchel, and P. Ouellet, “Front-end factor analysis for speaker verification,” *Audio, Speech, and Language Processing, IEEE Transactions on*, vol. 19, no. 4, pp. 788–798, 2011.

[12] D. Martınez, O. Plchot, L. Burget, O. Glembek, and P. Matejka, “Language recognition in ivectors space,” *Proceedings of Interspeech, Firenze, Italy*, pp. 861–864, 2011.

[13] K. J. Han, S. Ganapathy, M. Li, M. K. Omar, and S. Narayanan, “TRAP language identification system for RATS phase II evaluation,” in *Proc. Interspeech*, 2013.

[14] N. Dehak, P. A. Torres-Carrasquillo, D. A. Reynolds, and R. Dehak, “Language recognition via i-vectors and dimensionality reduction,” in *Proc. Interspeech*, 2011, pp. 857–860.

[15] M. Li and S. Narayanan, “Simplified supervised i-vector modeling with application to robust and efficient language identification and speaker verification,” *Computer Speech & Language*, 2014.

[16] M. Li, A. Tsiartas, M. Segbroeck, and S. Narayanan, “Speaker verification using simplified and supervised i-vector modeling,” in *Proc. ICASSP*, 2013.

[17] K. Walker and S. Strassel, “The RATS Radio Traffic Collection System,” in *Odyssey 2012-The Speaker and Language Recognition Workshop*, 2012.

[18] A. Kanagasundaram, R. Vogt, D. B. Dean, S. Sridharan, and M. W. Mason, “I-vector based speaker recognition on short utterances,” in *Proceedings of the 12th Annual Conference of the International Speech Communication Association*. International Speech Communication Association (ISCA), 2011, pp. 2341–2344.

[19] A. Larcher, P. Bousquet, K. A. Lee, D. Matrouf, H. Li, and J.-F. Bonastre, “I-vectors in the context of phonetically-constrained short utterances for speaker verification,” in *Proc. ICASSP*. IEEE, 2012, pp. 4773–4776.

[20] A. Lawson, M. McLaren, Y. Lei, V. Mitra, N. Scheffer, L. Ferrer, and M. Graciarena, “Improving language identification robustness to highly channel-degraded speech through multiple system fusion,” in *Proc. Interspeech*, 2013.

[21] J. Ma, B. Zhang, S. Matsoukas, S. H. Mallidi, F. Li, and H. Hermansky, “Improvements in language identification on the RATS noisy speech corpus,” in *Proc. Interspeech*, 2013.

[22] R. Travadi, M. Van Segbroeck, and S. S. Narayanan, “Modified-prior i-vector estimation for language identification of short duration utterances,” in *Proc. Interspeech*, 2014, submitted.

[23] A. O. Hatch, S. S. Kajarekar, and A. Stolcke, “Within-class covariance normalization for SVM-based speaker recognition,” in *Proc. Interspeech*, 2006.

[24] P. Kenny, “Joint factor analysis of speaker and session variability: Theory and algorithms,” *CRIM, Montreal,(Report) CRIM-06/08-13*, 2005.

[25] K. J. Han and J. Pelecanos, “Frame-based phonotactic language identification,” in *Spoken Language Technology Workshop (SLT)*. IEEE, 2012, pp. 303–306.

[26] A. G. Adami, L. Burget, S. Dupont, H. Garudadri, F. Grezl, H. Hermansky, P. Jain, S. S. Kajarekar, N. Morgan, and S. Sivadas, “Qualcomm-ICSI-OGI features for ASR.” in *Proc. Interspeech*, 2002.

[27] M. Van Segbroeck, A. Tsiartas, and S. Narayanan, “A robust frontend for VAD: Exploiting contextual, discriminative and spectral cues of human voice,” in *Proc. Interspeech*, 2013.

[28] S. Davis and P. Mermelstein, “Comparison of parametric representations for monosyllabic word recognitions in continuously spoken sentences,” *IEEE Transactions on Acoustics, Speech and Signal Processing*, vol. 28, no. 4, pp. 357–366, Aug. 1980.

[29] H. Hermansky, “Perceptual linear predictive (PLP) analysis of speech,” *Journal of the Acoustical Society of America*, vol. 87, no. 4, pp. 1738–1752, Apr. 1990.

[30] Y. Shao, S. Srinivasan, and D. Wang, “Incorporating auditory feature uncertainties in robust speaker identification,” in *Proc. ICASSP*, 2002, pp. 277–280.

[31] M. Kleinschmidt, “Spectro-temporal gabor features as a front end for ASR,” in *Proc. Forum Acusticum Sevilla*, 2002.

[32] A. F. Martin and C. S. Greenberg, “The NIST 2010 speaker recognition evaluation,” in *Proc. Interspeech*, 2010, pp. 2726–2729.