UTD-CRSS Systems for 2012 NIST Speaker Recognition Evaluation

The CRSS SRE Team

Taufiq Hasan John H.L. Hansen
Gang Liu Keith W. Godin
Seyed Omid Sadjadi Abhinav Misra
Navid Shokouhi Ali Ziaei
Hynek Bořil
Center for Robust Speech Systems (CRSS)
Erik Jonsson School of Engineering & Computer Science
The University of Texas at Dallas, USA

***Abstract*— This document briefly describes the systems submitted by the Center for Robust Speech Systems (CRSS) from The University of Texas at Dallas (UTD) for the 2012 NIST Speaker Recognition Evaluation. We developed a state-of-the-art i-vector based speaker recognition system [1]. Probabilistic linear discriminant analysis (PLDA) [2] along with several other back-ends are used for channel/noise compensation. Given that the emphasis of the NIST SRE-2012 is on noisy and short duration test conditions, in our system development we focused on: (1) novel robust acoustic features, (2) new feature normalization schemes, (3) various back-end strategies for multiple session enrollment.**

I. Introduction

Consistent with previous year's evaluation, this year the core task has been speaker detection. However, there are some differences: (1) real and artificially added noise in test data, (2) short and long duration utterances in test, (3) multiple segments for training, and (4) allowing the system to train models using all the target speakers data. These important differences in the current SRE lead us to take some special care in designing the development system.

II. Preparation of the Development System

In the process of preparing the development system, we had close collaboration with the I4U consortium. Two set of speaker ID tasks were prepared, namely *Dev* and *Eval* [1]. The motivation to do this was to train and test the system on *Dev* and run it on *Eval* to verify if the methods used in *Dev* also provide benefit in *Eval*. The *Eval* task was designed to be closer to the actual SRE'12 evaluation, so that the fusion and calibration parameters trained on *Dev* can also be tested on *Eval*. In preparing these tasks, all the SRE'12 target speakers were first separated into three disjoint sets for enrollment and test. The utterances from set-1 was used for *Dev-Train*, set-2 for *Dev-Test* and *Eval-Train*, and set-3 for *Eval-Test*. Care was taken so that the train/test pairs do not have the same session (identical LDC ID), ensuring a channel mismatch. The speakers that have a single utterance were not used in *Dev-Train*, but used in *Eval-Train*. The *Eval-Train* list also contained the 100 new evaluation time released speakers. This way, this list could be directly used for the actual SRE'12 evaluation. For the i-vector extractor and discriminative model training, we used the *Dev-Train* utterances with other data.

A. Noisy file Generation

We collected 10 HVAC noise files and generated 10 crowd noise files by summing up 500-800 NIST SRE utterances from both male and female speakers. The noise files were again separated into three disjoint sets with both noise types having equal number of files in each set. We used our in-house tools to generate the noisy files with the psophometric weighting (ITU-T Recommendation O.41) method as mentioned by NIST. The active speech level was measured according to the ITU-T Recommendation P.56. For degrading the test files, we used FaNT toolkit with G-712 weighting option. For each training and test file in *Dev* and *Eval*, one 6 dB and one 15 dB noisy version was created. The noise file was selected randomly from the corresponding set where the utterance belongs to.

B. Short Duration Segments

To handle the short duration files in SRE-2012 test, we cropped the test files to have active speech durations of 20 to 160 second with a 20 second interval. We used the VAD labels from VAD-2 (see Section III) to cut the test files. The duration values were assigned to the test files randomly to have a uniform distribution of active speech durations in the specified range. These short and long files were used together to prepare a second Dev/Eval task. We denote the latter task as the "mixed duration test" condition.

III. System components

A. Voice Activity Detection (VAD)

1) VAD Algorithm-1 (VAD-1): In this algorithm, to remove silence and low energy speech segments, a two stage voice activity detection (VAD) is performed. In the first stage, which is used before feature extraction, a soft VAD based

[1] These tasks were developed by Rahim Saeidi of RUN, following the extensive discussions and feedback from the I4U members.

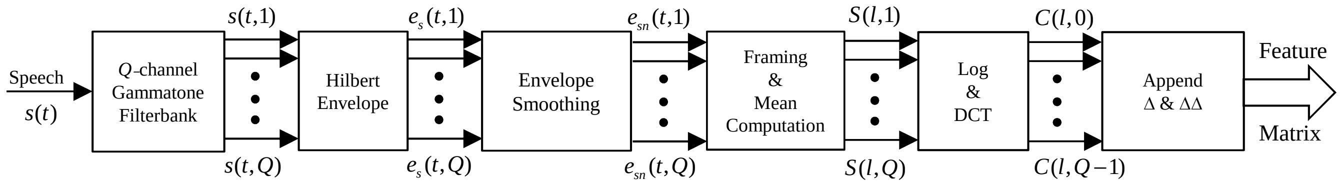

Fig. 1. Block diagram of the MHEC feature extraction framework. The symbols represent the output signals at each stage.

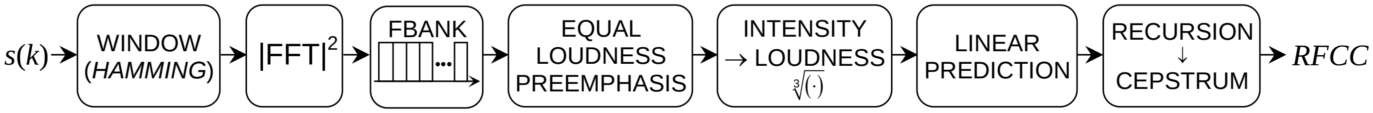

Fig. 2. Block diagram of RFCC front-end.

on perceptual spectral flux and several voicing measures is utilized to remove the non-speech segments [3]. This strategy saves large amount of computations, since in this manner features are only extracted from speech segments. In the second stage, which is applied after the feature extraction, an energy based method is employed to drop the low-energy speech frames as well as the residual non-speech frames from the soft VAD in the first stage. These low energy frames are easily affected by noise and channel variabilities, and do not carry much speaker-dependent information.

2) VAD Algorithm-2 (VAD-2): The main algorithm used in this VAD very closely follows [4]. The VAD is performed on both channel A and B, and segments where speech is detected in channel B is removed from channel A. Since the interviewer channel is usually corrupted by a noise floor to mask the interviewee speech, a spectral subtraction based speech enhancement is always performed before VAD on channel B. For channel A, first a simple SNR estimation algorithm based on 2 GMM's is used. If the SNR is less than 18 dB, channel A is enhanced using spectral subtraction before processed by the VAD. Feature extraction is performed first on the whole utterance. The non-speech feature vectors detected using the VAD algorithm is removed in the second stage. This is done to have a more accurate representation of the delta coefficients.

B. Acoustic Features

Before extracting features, all waveforms are first downsampled to 8 kHz. All of our feature extraction blocks use 25ms frames with 10ms skip-rate. All of these features use 12 cepstral coefficients and log-energy/C_0 and delta and double delta coefficients are appended, thus providing a 39 dimensional feature vector. The description of the individual features are provided below:

1) Mean Hilbert Envelope Coefficients (MHEC): MHEC features have been shown to be an effective alternative to the conventional MFCCs for robust SID under reverberant and noisy mismatched conditions [5], [6]. A block diagram illustrating the procedure for extracting the MHECs is depicted in Fig. 1.

First, the pre-emphasized speech signal $s(t)$ is decomposed into 24 bands through a 24-channel Gammatone filter-bank covering the frequency range of 300–3400 Hz. Next, the Hilbert envelope $e_s(t,j)$ is calculated and smoothed using a low-pass filter with a cut-off frequency of 20 Hz. In the next stage, the low-pass filtered $e_{sn}(t,j)$ is blocked into frames of 25 ms duration with a skip rate of 10 ms. To estimate the temporal envelope amplitude in frame l, the sample mean $S(l,j)$ is computed. Note that $S(l,j)$ is a measure of the spectral energy at the center frequency of the j^{th} channel, and therefore provides a short-term spectral representation of the speech signal $s(t)$. The next two stage (i.e., log compression, DCT, delta calculation) is commonly used in the extraction of conventional cepstral features such as the MFCCs. Here, only the first 12 coefficients (excluding C_0) are retained after DCT and appended with the log-energy for each frame. The final output is a matrix of 39-dimensional cepstral features, entitled the mean Hilbert envelope coefficients (MHEC). The MHEC features are further processed through cepstral mean and variance normalization (CMVN). It is worth noting here that MHECs are extracted from the audio signals pre-processed with VAD-1.

2) Rectangular Filter-bank Cepstral Coefficients (RFCC): The RFCC front-end is inspired by perceptual linear prediction (PLP) cepstral features [7]. The original Bark frequency trapezoid filters are replaced by a bank of 24 uniform non-overlapping rectangular filters distributed over a linear frequency scale. The block scheme of the RFCC front-end is shown in Fig. 2. RFCC was initially proposed for robust ASR in noisy/Lombard speech codnitions (*20Bands-LPC*) [8]. RFCCs are extracted using an open source feature extraction and enhancement tool CTUCopy [9] and normalized using conventional feature Gaussianization [10]. The tools and a recipe for RFCC extraction are available at [11].

3) MFCC-QCN-RASTA$_{LP}$: This front-end uses the conventional MFCC features extracted using HTK tools. Number of filter-banks used is 24, 12 cepstral coefficients and energy is used. This feature stream is processed by Quantile Cepstral Normalization (QCN) [8] and RASTA$_{LP}$ [12].

C. Feature Normalizations

1) Quantile-Based Cepstral Normalization (QCN): Similar to cepstral mean-variance normalization (CMVN), QCN [8] aims at minimizing the mismatch between distributions of training and test samples. Unlike CMVN, QCN does not make any assumptions about the distribution properties and instead performs an alignment of the sample dynamic ranges estimated from distribution quantiles. In our previous studies, QCN provided superior performance gains in ASR under noise and Lombard effect [8] and reverberation [13] compared to other popular normalizations.

2) RASTA$_{LP}$: Temporal filtering is known to reduce the effects of noise and reverberation on speech systems. Recently proposed RASTA$_{LP}$ [12] is a low-pass filter that approximes the low-pass component of the popular RASTA filter [14]. Due to the low order of the RASTA$_{LP}$ filter, the adverse transient effects seen in original RASTA as significantly reduced. In addition, RASTA$_{LP}$ bypasses the mean subtraction functionality of RASTA and can be conveniently combined with distribution normalizations of choice. In our previous ASR studies, RASTA$_{LP}$ considerably outperformed RASTA in noisy, Lombard effect, and reverberated conditions [13], [15].

D. UBM Training

Gender dependent UBMs having diagonal-covariance matrices with 1024 mixtures are trained on telephone utterances selected from the Switchboard II Phase 2 and 3, Switchboard Cellular Part 1 and 2, and the NIST 2004, 2005, 2006 SRE enrollment data. Iterations per mixture split begins with 4 while gradually increases to 15 for higher order mixtures. For front-end and VAD-2 development, we used data sub-sampling for fast UBM training [16], [17] to perform a large number of experiments. After the front-ends have been finalized, we always used all the data for training the UBM.

E. I-vector Extractor Training

For the training the i-vector extractor, the UBM training dataset and additional SRE-12 target speaker's data is both clean and noisy versions are used. Five iterations are used for the EM training. Our i-vector size was 600. All i-vectors are mean normalized and then length normalized using radial Gaussianization [18].

F. Back-end Classifiers

1) I-vector averaged PLDA (PLDA-1): This is the standard PLDA back-end. We reduce the i-vector dimension to 400 using LDA first, then perform mean normalization and Radial Gaussianization on the i-vectors before the PLDA modeling. This diagonal covariance noise based PLDA model utilizes 400 eigenvoice dimensions. For noisy and mixed duration test conditions, we added some i-vectors extracted from noisy and short duration utterances in our PLDA training. These development i-vectors were also used in other back-ends.

2) Cosine-Distance Scoring (NAP-CDS): The iVectors of Multiple sessions of the same enroll speakers are averaged first. LDA is performed for dimensionality reduction, then a modified Nuisance Attribute Projection (NAP) [19] is performed for channel compensation, and finally cosine distance metric is employed for scoring.

3) Regularized Logistic Regression (RLG): In this backend, an L2-regularized logistic regression is applied using the LIBLINEAR package [20].

4) SVM anti-modeling (SVM-Anti): The framework is based on SVM anti-modeling. A cosine kernel is used in UBS-SVM backend as described in [21].

5) Scores-averaged PLDA (PLDA-2): Different from PLDA-1, the i-vectors of the same speaker is not grouped and averaged. Each test file will be tested against each sessions i-vector of the involved enroll speaker and the log-likelihood of are averaged. And then the averaged score is taken as the one for the involved enroll-test trial.

G. Score Fusion and Calibration

The CRSS fusion and calibration system is mainly based on the bosaris toolkit [22]. One major benefit obtained from this toolkit was obtained by incorporating side-information/quality measures. Various features and implementations were used and eventually the feature (quality measure) resulting in the best overall system performance using the active speech duration measured using VAD-1 [3]. For the model quality measure we used the mean of the effective speech duration of all the train-files used for the model speaker and for the test-file the total duration of speech in that test file was used. An estimate of the Signal-to-Noise Ratio (SNR) (computed using the WADA algorithm [23]) was also used in the second and third alternate submissions. The system descriptions in each of the submissions is as below:

IV. The sub-systems

In this section, we describe the subsystems that were used in our submission. In total, we have developed five subsystems, four of which are SVM based and one of them is GMM based. All of the SVM systems use the factor analysis front-end. A brief description of the subsystems are given below.

V. The CRSS submissions

This section describes the system results that were actually submitted. Below is the list of all available combination of acoustic front-ends and back-ends: (1) MHEC-PLDA-1, (2) MHEC-NAP-CDS, (3) MHEC-RLG, (4) MHEC-SVM-Anti, (5) MHEC-PLDA-2, (6) RFCC-PLDA-1, (7) RFCC-NAP-CDS, (8) RFCC-RLG, (9) RCC-SVM-Anti, (10) RFCC-PLDA-2, (11) MFCC-NAP-CDS and (12) MFCC-PLDA-2. The CRSS submissions are summarized in Table IV.

VI. Other Developments

1) PMVDR front-end: The power spectrum estimation method used in the extraction of MFCC features is not robust to noise and channel degradations, resulting in large variations in estimated parameters. To alleviate this, a noise robust perceptual spectrum estimation technique with minimum variance was proposed in [24]. The acoustic features extracted using the perceptual MVDR spectrum have been shown to outperform the conventional MFCCs under noisy conditions for ASR [24] as well as speaker recognition applications [25]. In our

TABLE I

CRSS-UTD SUB-SYSTEMS FOR PRIMARY SUBMISSION USING LONG DURATION TRAIN/TEST

#	Feature/VAD/Norm	Back-end	Male						Female					
			Dev			Eval			Dev			Eval		
			%EER	minDCF*	actDCF*	%EER	minDCF*	actDCF*	%EER	minDCF*	actDCF*	%EER	minDCF*	actDCF*
1	MHEC-VAD1-CMVN	PLDA-1	0.72	2.62	9.64	1.15	3.86	13.74	1.60	5.41	10.83	1.43	5.47	14.30
2		NAP-CDS	0.91	3.21	15.98	0.92	3.16	16.58	1.06	4.61	17.26	1.04	4.19	17.83
3		RLG	1.21	4.56	8.57	1.29	4.55	6.31	1.72	6.26	9.09	1.48	5.17	6.33
6	RFCC-VAD2-Warp	PLDA-1	0.82	3.07	10.72	1.16	4.48	13.69	1.57	5.64	11.59	1.37	5.23	13.80
7		NAP-CDS	1.01	3.90	16.72	0.81	3.35	17.14	1.17	4.85	18.33	0.78	3.45	18.48
8		RLG	1.25	5.09	11.65	1.20	4.50	8.42	1.65	6.40	11.75	1.04	4.29	7.27

TABLE II

CRSS-UTD SUB-SYSTEMS FOR ALTERNATE SUBMISSIONS USING MIXED DURATION TRAIN/TEST

#	Feature/VAD/Norm	Back-end	Male						Female					
			Dev			Eval			Dev			Eval		
			%EER	minDCF*	actDCF*	%EER	minDCF*	actDCF*	%EER	minDCF*	actDCF*	%EER	minDCF*	actDCF*
1	MHEC-VAD1-CMVN**	PLDA-1	1.358	5.31	16.698	1.934	7.13	21.59	2.496	8.89	17.48	2.448	9.85	22.71
2		NAP-CDS	1.845	6.28	16.73	1.460	5.14	16.70	2.156	8.46	18.23	1.663	6.69	17.94
3		RLG	2.761	9.29	11.97	2.206	7.50	8.52	3.884	13.23	14.76	2.590	9.18	9.58
4		SVM-Anti	1.905	6.65	15.33	1.460	5.20	15.04	2.293	8.85	17.07	1.719	6.94	16.43
5		PLDA-2	1.139	4.60	32.55	1.382	5.69	42.69	2.163	8.15	31.42	1.709	7.78	38.40
6	RFCC-VAD2-Warp	PLDA-1	1.325	5.68	15.03	1.883	7.42	18.15	2.353	8.92	16.37	2.283	9.33	19.50
7		NAP-CDS	1.690	6.55	17.12	1.157	4.45	13.69	1.574	5.64	11.59	1.369	5.23	13.80
8		RLG	2.365	9.03	14.47	1.810	7.08	10.10	3.286	11.90	16.10	1.955	7.62	9.81
9		SVM-Anti	1.753	6.72	15.60	1.423	5.48	15.74	2.200	8.44	17.40	1.442	6.21	16.73
10		PLDA-2	0.990	4.79	27.72	1.271	5.64	35.27	1.815	7.52	27.22	1.383	6.73	31.72
11	MFCC-VAD2-QCN-RASTA$_{LP}^{***}$	NAP-CDS	1.684	6.43	16.64	1.422	5.32	16.77	2.225	9.06	18.42	1.869	7.65	18.21
12		PLDA-2	1.048	4.76	29.26	1.221	5.57	39.22	2.132	8.38	29.52	1.777	8.37	36.83

TABLE III

FUSION AND CALIBRATION PERFORMANCE ON EVAL SET USING SIDE INFORMATION

#	Systems Fused	Fusion Method	Side Information	Compound LLR	Male			Female		
					%EER	minDCF*	actDCF*	%EER	minDCF*	actDCF*
1	2,3,5,7,8,10	Linear	None	No	0.82	3.32	3.79	0.85	4.13	4.21
2	2,3,5,7,8,10	Linear	None	Yes	0.67	2.48	2.88	0.66	2.81	2.97
3	2,3,5,7,8,10	Linear+quality	SNR,Duration	No	0.75	2.62	2.86	0.69	2.86	2.93
4	2,3,5,7,8,10	Linear+quality	SNR,Duration	Yes	**0.64**	**2.17**	**2.45**	**0.59**	**2.26**	**2.46**

* All the DCF values are multiplied by 100.

** Results from this front-end using VAD-1, are sub-optimal compared the front-ends using VAD-2, since the test files were cropped using VAD-2 for the mixed duration tests.

*** Results from this front-end may be sub-optimal since the feature-warping operation was unintentionally always kept "ON" before QCN and RASTA$_{LP}$.

TABLE IV

LIST OF CRSS SUBMISSIONS

Submission Name	Task	Systems Fused	Fusion method	Side Information	Fusion Training Set
CRSS_01_core_core_primary	core-core	1,2,3,6,7,8	Linear+quality	Duration	*Dev*, full duration test
CRSS_02_core_core_alternate	core-core	{2,3,5,7,8,10},{11,12}*	Linear+quality	SNR,Duration	*Dev*, mixed duration test
CRSS_03_core_core_alternate	core-core	2,3,5,7,8,10,11,12	Linear+quality	SNR,Duration	*Dev*, mixed duration test
CRSS_04_core_core_alternate	core-core	2,3,5,7,8,10,11,12	Linear+quality	Duration	*Dev*, mixed duration test
CRSS_05_core_core_alternate	core-core	2,3,5,7,8,10,11,12	Linear	None	*Dev*, mixed duration test
CRSS_01_core_extended_primary	core-extended	{2,5,7,10},{11,12}*	Linear+quality	Duration	*Dev*, mixed duration test
CRSS_02_core_extended_primary	core-extended	2,5,7,10,11,12	Linear+quality	Duration	*Dev*, mixed duration test
CRSS_03_core_extended_primary	core-extended	{2,5,7,10},{11,12}*	Linear+quality	Duration	*Eval*, mixed duration test
CRSS_04_core_extended_primary	core-extended	2,5,7,10,11,12	Linear+quality	Duration	*Eval*, mixed duration test
CRSS_05_core_extended_primary	core-extended	2,4,5,7,9,10,11,12	Linear	None	*Dev*, mixed duration test
CRSS_06_core_extended_primary	core-extended	2,4,5,7,9,10,11,12	Linear	None	*Eval*, mixed duration test

* The systems in braces were first linearly fused using equal weights before the second stage of fusion using quality measures.

system, the PMVDR features are extracted from audio files pre-processed with VAD-1. Similar to MHECs, the PMVDR features are post-processed with CMVN.

2) Speaker Diarization: We initially attempted to performs speaker diarization for the SRE'08 interview segments, where both speakers are prominent in the target speaker's channel. However, due to time constraints, we were not able to incorporate this in our final submission. In our speaker diarization system, GMM mean-super-vectors are extracted for each speech segment and an unsupervised clustering is performed for diarization. There are three issues we addressed, estimating the number of speakers, initialization of the k-means algorithm for clustering and the distance measure. We used SVD to estimate number of speakers and initialization in utilized to cosine distance metric for distance measure.

VII. COMPUTATIONAL RESOURCES

The speaker recognition system was implemented on our in-house high-performance Dell computing cluster, running

Rocks 6.0 (Mamba) Linux distribution. The cluster comprises of eight 6C Intel Xeon 2.67 GHz CPU's, four 10C Intel Xeon 2.40 GHz CPU's, and 18 quad-core Intel Xeon 2.33 GHz CPU's, yielding a total of 408 processors. The total amount of internal RAM on the cluster exceeds 1 TB. All our data including audio files, features, statistics, etc. are stored on a 30 TB Dell PowerVault MD1000 direct attached storage.

VIII. CPU EXECUTION TIME

We tested the system's scoring process using one CPU of 2.67 GHz clock speed and 24 GB RAM. We selected a 5 minute utterance (exact duration of 301.59 seconds) and calculated the time required to perform feature extraction (MFCC with QCN RASTA$_{LP}$), voice activity detection (using VAD-2), extraction of zero and first order statistics and the 600 dimensional i-vector. The time required for this chain of processes is for the selected utterance is 45.17s. This is computed by averaging the elapsed time obtained from three independent runs. Scoring an utterance using our PLDA model takes 0.1 seconds on average. This provides us with the real-time factor (RTF) of 0.15 for test. For training the models, it depends on how many enrollment utterances are provided. Since the UBM and TV matrices are trained off-line, speaker enrollment requires only to extract the corresponding i-vectors, thus the time required will be a multiple of the number of enrollment utterances provided for a speaker.

IX. ACKNOWLEDGEMENTS

We would like to thank all the members of the I4U consortium for their valuable discussion in the email group, especially Rahim Saeidi and Ville Hautamäki. Also, we would like to give our special thanks to Niko Brummer for his detailed suggestions on fusion and calibration using the Bosaris toolkit.

REFERENCES

[1] N. Dehak, P. Kenny, R. Dehak, P. Dumouchel, and P. Ouellet, "Front-end factor analysis for speaker verification," *IEEE Trans. Audio, Speech, and Lang. Process.*, vol. 19, no. 99, pp. 788 – 798, May 2010.

[2] P. Matejka, O. Glembek, F. Castaldo, M. Alam, O. Plchot, P. Kenny, L. Burget, and J. Cernocky, "Full-covariance UBM and heavy-tailed PLDA in i-vector speaker verification," in *Proc. ICASSP*, Florence, Italy, Oct. 2011, pp. 4828 – 4831.

[3] S. O. Sadjadi and J. H. L. Hansen, "Unsupervised Speech Activity Detection using Voicing Measures and Perceptual Spectral Flux," *Signal Processing Letters, IEEE (submitted)*, Dev. 2012.

[4] J. Sohn and W. Sung, "A voice activity detector employing soft decision based noise spectrum adaptation," in *Proc. ICASSP*, vol. 1. IEEE, 1998, pp. 365–368.

[5] S. O. Sadjadi and J. H. L. Hansen, "Assessment of single-channel speech enhancement techniques for speaker identification under mismatched conditions," in *Proc. INTERSPEECH*, Makuhari, Japan, Sept. 2010, pp. 2138–2141.

[6] ——, "Hilbert envelope based features for robust speaker identification under reverberant mismatched conditions," in *Proc. IEEE ICASSP*, Prague, Czech Republic, May 2011, pp. 5448–5451.

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

[8] H. Bořil and J. H. L. Hansen, "Unsupervised equalization of lombard effect for speech recognition in noisy adverse environments," *Audio Speech and Language Processing, IEEE Transactions on*, pp. 1379–1393, Sep. 2010.

[9] P. Fousek, "CTUCopy – universal speech enhancer and feature extractor," 2007. [Online]. Available: http://noel.feld.cvut.cz/speechlab/

[10] J. Pelecanos and S. Sridharan, "Feature warping for robust speaker verification," in *Proc. Odyssey*, 2001, pp. 213–218.

[11] [Online]. Available: http://www.utdallas.edu/~hynek/tools.html

[12] H. Bořil and J. H. L. Hansen, "UT-scope: Towards LVCSR under lombard effect induced by varying types and levels of noisy background," in *Proc. ICASSP*, May. 2011, pp. 4472 – 4475.

[13] H. Bořil, F. Grézl, and J. H. L. Hansen, "Front-end compensation methods for LVCSR under Lombard effect," in *Proc. Interspeech*, Florence, Italy, 2011, pp. 1257–1260.

[14] H. Hermansky and N. Morgan, "RASTA processing of speech," *IEEE Transactions on SAP*, vol. 2, no. 4, pp. 578 –589, Oct. 1994.

[15] O. S. Sadjadi, H. Bořil, and J. H. L. Hansen, "A comparison of front-end compensation strategies for robust LVCSR under room reverberation and increased vocal effort," in *Proc. ICASSP*, Kyoto, Japan, 2012, pp. 4701–4704.

[16] T. Hasan, Y. Lei, A. Chandrasekaran, and J. H. L. Hansen, "A novel feature sub-sampling method for efficient universal background model training in speaker verification," in *Proc. ICASSP*, March 2010, pp. 4494 – 4497.

[17] T. Hasan and J. H. L. Hansen, "A study on universal background model training in speaker verification," *Audio Speech and Language Processing, IEEE Transactions on*, pp. 1890–1899, Sep. 2011.

[18] D. Garcia-Romero and C. Y. Espy-Wilson, "Analysis of i-Vector length normalization in speaker recognition systems," in *Proc. Interspeech*, Florence, Italy, Oct. 2011, pp. 249 – 252.

[19] A. Solomonoff, W. Campbell, and I. Boardman, "Advances in channel compensation for SVM speaker recognition," in *Proc. ICASSP*, vol. 1, 2005, pp. 629–632.

[20] R. Fan, K. Chang, C. Hsieh, X. Wang, and C. Lin, "Liblinear: A library for large linear classification," *The Journal of Machine Learning Research*, vol. 9, pp. 1871–1874, 2008.

[21] J. H. H. Gang Liu, Jun-Won Suh, "A fast speaker verification with universal background support data selection," in *Proc. ICASSP*, Kyoto, Japan, 2012, pp. 4793–4796.

[22] N. Brummer and E. de Villiers, "The bosaris toolkit: Theory, algorithms and code for surviving the new dcf," in *NIST SRE Analysis Workshop*, Atlanta, USA, Dec. 2011.

[23] C. Kim and R. Stern, "Robust signal-to-noise ratio estimation based on waveform amplitude distribution analysis," *INTERSPEECH-2008*, pp. 2598–2601, 2008.

[24] U. H. Yapanel and J. H. L. Hansen, "A new perceptually motivated mvdr-based acoustic front-end (pmvdr) for robust automatic speech recognition," *Speech Commun.*, vol. 50, pp. 142–152, February 2008.

[25] A. D. Lawson, P. Vabishchevich, M. C. Huggins, P. A. Ardis, B. Battles, and A. R. Stauffer, "Survey and evaluation of acoustic features for speaker recognition," in *Proc. IEEE ICASSP*, Prague, Czech Republic, May 2011, pp. 5444–5447.