TOWARDS AN INTELLIGENT ACOUSTIC FRONT-END FOR AUTOMATIC SPEECH RECOGNITION:BUILT-IN SPEAKER NORMALIZATION (BISN)[§]

Umit H. Yapanel and John H. L. Hansen

Robust Speech Processing Group, Center for Spoken Language Research
Univ. of Colorado at Boulder, CO, 80309, USA
{yapanel, jhlh}@cslr.colorado.edu, WEB : http://cslr.colorado.edu

ABSTRACT

Much effort has transpired over the past three decades in the formulation of "ideal" acoustic features which represent the speech signal in a *discriminative* and *compact* manner while being robust to adverse conditions and invariant to speaker differences. A good way of making ASR systems invariant to speaker differences is to perform *speaker normalization* on the input features. The most popular speaker normalization technique is the vocal tract length normalization (VTLN). However, its implementation requires immense computational resources and not practically applicable in real-time/embedded ASR systems. In this paper, we propose a new speaker normalization algorithm entitled *Built-in Speaker Normalization* (BISN) which is performed *on-the-fly* within the newly proposed PMVDR acoustic front-end and reduces computational resources significantly enabling its use within contemporary ASR systems. Evaluations using an in-car extended digit recognition task showed that *on-the-fly* implementation of the BISN algorithm produced a relative word error rate (WER) reduction of 24% compared to a no speaker normalization baseline.

1. INTRODUCTION

Although current *speaker independent* automatic speech recognition (ASR) systems perform well in most of the real world applications, the performance gap between speaker *dependent* and *independent* settings is significant. Many researchers would agree that there is still a substantial potential in finding an "ideal" acoustic front-end for the speech signal that successfully maintains the information needed for efficient speech recognition, *especially in noise*, while eliminating irrelevant *speaker-dependent* traits [1].

A significant step towards this "ideal" acoustic front-end was taken by the formulation of Perceptual MVDR Coefficients (PMVDRs) which are more effective than MFCCs for a number of tasks, especially in noise [2, 3]. However, they still lack the speaker-invariance property. This paper introduces a new and computationally efficient speaker normalization algorithm within the PMVDR [2, 3] framework which we call *Built-in Speaker Normalization* (BISN). BISN is computationally very efficient and can be completely integrated into the front-end taking an important step towards the "ideal" acoustic front-end described above, which we desperately need in order to make the ASR technology *pervasive*.

Andreou *et al.* proposed maximum likelihood-based speaker normalization procedures to extract and use acoustic features which are robust to variations in vocal tract length [4]. The algorithm reduced speaker-dependent variations between formant frequencies through a simple linear warping of the frequency axis, which was implemented by re-sampling the speech waveform in the time domain. Although the transformation was simple, the estimation of the warping factor required over 5 minutes of speech for each speaker and the estimation process was computationally intensive. Lee and Rose [5, 6] proposed a more efficient set of speaker normalization procedures similar to those of Andreou. Other nonlinear transformations have also been considered recently by researchers. For example, the feasibility of Bi-linear and All-pass Transforms (BLT, APT) for the application to the speaker normalization problem has been extensively studied by McDonough [7, 8]. He has shown that BLT can also be implemented in the cepstral domain. The optimal BLT parameters were estimated by a Gaussian Mixture Model (GMM) as the one maximizing the likelihood of the incoming data.

[§]This work was supported by U.S. Air Force Research Laboratory, Rome NY under Contract No.FA8750-04-1-0058.

2. THE PMVDR ACOUSTIC FRONT-END

For the details of the PMVDR computation, we refer readers to [2, 3]. PMVDR framework implements a new acoustic feature-extraction algorithm which *eliminates* the use of a filterbank to incorporate perceptual considerations. Instead, the FFT spectrum is directly warped using *interpolation* before envelope extraction. The envelope is extracted via a low-order all-pole MVDR spectrum which is shown to be superior to the Linear Prediction (LP)-based envelopes[9]. Utilizing direct warping on the FFT power spectrum by removing the filterbank processing step leads to the preservation of almost *all* information that exists in the short-term spectrum and accurate positioning of the perceptually very important formant peaks. Also, using the MVDR method to extract the *upper* envelope contributes greatly to the superior performance in noisy conditions [9, 2, 3]. The flow diagram of the PMVDR algorithm is given in Fig. 1.

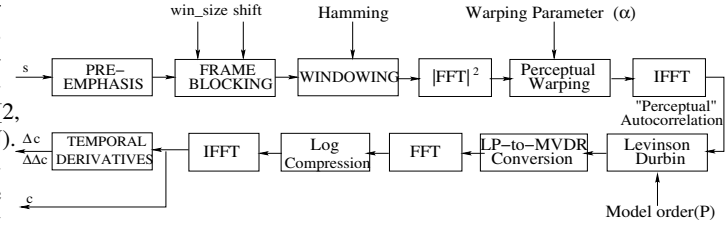

Fig. 1. *Flow diagram of PMVDR front-end*

3. THE "MEANING" OF PERCEPTUAL WARPING

Almost all acoustic front-ends proposed for ASR use some form of nonlinear warping on the FFT spectrum at some level. The argument for applying a non-linear warping, often referred to as *perceptual warping*, to the speech spectrum in the feature extraction process is strongly tied to the fact that the human auditory system performs similar type of processing to place more emphasis

on lower frequencies. In all of our experiments, when a perceptual warp is introduced, it always yields better recognition accuracy (on the order of 20%, relative). We claim that the *perceptual warp was actually meant to remove some of the existing inter-speaker variability in the feature set*. To justify this claim, we conducted an analysis within the framework in [10, 11, 2]. We extracted the PMVDR features for the CU-Move [12] training set (see Sec. 6) first with *no perceptual warp*, with a *bark scale* ($\alpha = 0.57$ at $16kHz$), and then with the *BISN warp factors* (see Sec. 5). Afterwards, we computed the variation of the trace measure (TM). The larger the TM, the more effectively the speaker variability is removed [10, 11, 2]. Figure 2 shows the variation of the trace measure (with respect to the *minimum* of number of speech classes and feature dimension) for the three cases. It verifies that using the perceptual warp *indeed* leads to the removal of significant inter-speaker variability. However, using the BISN warps specifically estimated for *each speaker* further reduces the inter-speaker variability.

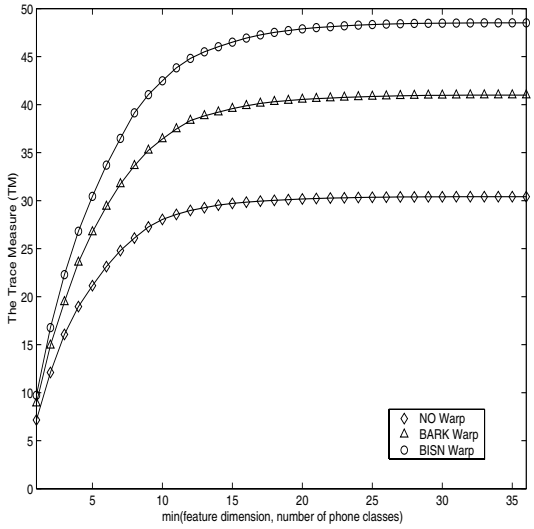

Fig. 2. *Variation of TM for NO warp (diamonds), BARK warp (triangles), and BISN warp (circles) for the CU-Move data*

4. CLASSICAL VTLN

In VTLN, the speech spectrum is linearly warped with an optimal warp factor (β) [5, 6, 13]. The speaker-dependent parameter, β, is determined by conducting likelihood computations. Generally a single Gaussian ($1G$) HMM set (λ) which is trained on a large population of speakers is used to estimate the warp factor. Assume that we have N_i utterances with $\mathbf{X}_i^\beta$ denoting the set of feature vectors and $\mathbf{W}_i$ denoting the corresponding transcriptions and the goal is to estimate the optimal warp factor ($\hat{\beta}_i$) for speaker i. Here, $\hat{\beta}_i$ is estimated by maximizing the likelihood of the warped features given the HMM model, λ and the transcriptions, $\mathbf{W}_i$,

$$\hat{\beta}_i = \arg\max_{\beta} Pr(\mathbf{X}_i^\beta | \lambda, \mathbf{W}_i). \quad (1)$$

Optimum warp factors are estimated by searching over a one dimensional grid of 33 points (a step size of γ=0.01 in this case) in the range of $[0.84, 1.16]$[1]. After estimating the warp factors, all utterances are parameterized and then a "canonical" HMM set (λ_N) is re-estimated from this warped feature set. During recognition, optimal warp factors are estimated by a two-pass strategy. For classical VTLN experiments, we use *all the available data from each test speaker* to estimate the optimal warps. On the average, the estimation of the optimal VTLN warp for a speaker requires 18 *times* the computational resources needed for one feature extraction and one likelihood computation.

[1]Our search was over this range, but one may reduce the dimension of the search space at the expense of performance

5. BUILT-IN SPEAKER NORMALIZATION (BISN)

The inter-speaker variability analysis showed that *perceptual warping* is in fact a *speaker normalization warping*, too. Originating from this fact, we propose to adjust the perceptual warp parameter within the PMVDR front-end for each speaker specifically and call this new warp the *self normalization warp* (SNW). This should, in turn, normalize for the speaker differences. Since this procedure does not require 2 *applications of warping to the spectrum* (for perceptual warp and for VTLN warp), as in classical VTLN, it is more efficient. Also, the normalization is achieved by only adjusting an *internal parameter* of the PMVDR front-end (i.e. α), making it a *built-in* procedure. The estimation of the self normalization warp (SNW), α_i, for speaker i, is done the same way as the VTLN. In a typical setting with a $\alpha = 0.57$ (Bark scale at $16kHz$), the search space for the SNWs can be chosen as $[0.49, 0.65]$ with a step size of $\gamma = 0.01$. In this case, the search requires 10 *times* the computational resources needed for one feature extraction and one likelihood computation which is still computationally expensive.

5.1. Binary Tree Searh (BTS) Approach

The likelihood of the data from a specific speaker is typically monotonically increasing (with the changing warp factor) up to a maximum, i.e. until reaching the *optimal warping factor*, and then becomes monotonically decreasing. Using this monotonicity property, we can devise a much more efficient search algorithm than the linear search approach. Let the $1G$ HMM set be trained with α_{mw} (e.g. 0.57) and the search space be chosen as $[\alpha_l, \alpha_u]$ (e.g. [0.49, 0.65]) with a step size γ (e.g. 0.01) resulting in an N_l-point ($N_l = (\alpha_u - \alpha_l)/\gamma + 1$, e.g. $N_l = 17$) one dimensional search space. We can summarize the proposed binary tree search (BTS) algorithm is as follows;

1. Compute the likelihood, P_{mw} for α_{mw}, referred to as the *middle warp* since it is the center of our search space.
2. Compute the *lower warp* as $\alpha_{lw} = (\alpha_l + \alpha_{mw})/2$ and similarly *upper warp* as $\alpha_{uw} = (\alpha_u + \alpha_{mw})/2$. This divides the warp space into *lower and upper regions*, whose middle warps are α_{lw} and α_{uw}, respectively.
3. Compute the likelihood, P_{lw}, for α_{lw}, if $P_{lw} > P_{mw}$, then disregard the upper region, consider the lower region as the new search space whose middle warp is α_{lw} and go to Step 2. If $P_{lw} < P_{mw}$ then compute P_{uw}, for α_{uw}. If $P_{uw} > P_{mw}$ then disregard the lower region, consider the upper region as the new search space whose middle warp is α_{uw} and go to Step 2. For the last case, where $P_{uw} < P_{mw}$, take the new search space to be $[\alpha_{lw}, \alpha_{uw}]$, whose middle warp is α_{mw} and go to Step 2. In all cases, the search space is reduced by half.

Recursively repeating steps 2 and 3, we estimate the self normalization warp(SNW) by an average of 6 *times* the computational resources needed for one feature extraction and one likelihood computation (with the example settings above). This means a reduction

System/WER	Female	Male	Overall
MFCC (Baseline)	9.16	13.22	**11.12**
PMVDR (w/o SN)	5.57	8.76	**7.11**
PMVDR w/ SN			
VTLN	4.30	7.12	**5.66**
BISN	4.16	7.17	**5.61**
BISN/BTS	4.16	7.17	**5.61**
BISN/MS-BTS(off-line)	4.13	7.16	**5.59**
BISN/MS-BTS(on-the-fly)	3.90	7.04	**5.42**

Table 1. *WERs[%] of CU-Move for different speaker normalization (SN) algorithms.*

of 40% in the computational load in the search stage by moving from *linear search* to *binary tree search* (BTS).

5.2. Model versus Feature Space Search

The search for the SNWs is conducted in the *feature space*, so the contribution of the Jacobian is not taken into account which may cause some systematic errors in SNW estimation. When the search is conducted in the *model space*, the need to compensate for the Jacobian is eliminated [14]. In the model-based search, we train a $1G$ HMM set for each warp in the search space *off-line*. We then extract the features for the no warp case *only once* and then compute the likelihood against different warped models. Integrated with the BTS approach, the model-based search requires only 1 *feature extraction and* 6 *likelihood computations*. We call this approach model space-binary tree search (MS-BTS). First, we train $1G$ HMM models for each warping factor in the search space. An example search space would be in in $[0.49,\ 0.65]$ with a step size of $\gamma = 0.01$ and with the center warp of $\alpha_C = 0.57$. Assume that the input features are extracted with the warp α_N. We pick the model (trained with the warp α_M) yielding the maximum likelihood given the features. The search is performed via the binary tree search (BTS) approach. The optimal SNW α_o is given in Eq.2. The rest of the normalization is the same as classical VTLN.

$$\alpha_o = \alpha_C + \alpha_N - \alpha_M \tag{2}$$

6. CU-MOVE EXTENDED DIGITS TASK

For all experiments, we use SONIC [15], the Univ. of Colorado's LVCSR System. The acoustic models are decision-tree state clustered HMMs with associated Gamma probability density functions to model state-durations. We used a window length of $25ms$ and a skip rate of $10ms$ by Hamming windowing the frame data before further processing. The 39 dimensional feature set contains 12 statics, deltas and delta-deltas along with normalized-log energy, delta and delta-delta energy. Cepstral Mean Normalization (CMN) was utilized on the final feature vectors. The speech data used in the experimentation was obtained from the CU-Move Extended Digits Corpus. The database and noise conditions are analyzed in [12] in detail. A total of 60 speakers balanced across gender and age (18-70 yrs. old) were used in the training set. The test set contained another 50 speakers, again gender- and age-balanced. The HMMs were trained using a decision-tree HMM trainer and contained a total of $10K$ Gaussians. The vocabulary size was 40. We used the optimized settings ($\alpha = 0.57$ and $P = 24$) for PMVDR on the CU-Move task [3]. The recognition performance is summarized in Table 6 for different speaker normalization(SN) approaches.

We now turn to an approach for the *on-the-fly* application of BISN w/MS-BTS in a real world scenario for which, we have all training data in advance and can estimate the SNWs off-line. However, for the test we will not have access to all data from a speaker to determine the SNWs. Moreover, we do not have the information as to when the speaker changes occur. So the algorithm should in fact be able to *adapt* the SNWs to the changing speaker and also be flexible (i.e. slowly changing) even for the same speaker to account for the slight variations in the vocal tract characteristics. By making clever use of all the algorithms developed so far, it is possible to establish a co-operation between the front-end and the recognizer which enables the front-end to normalize itself automatically without the need to perform recognition twice, or re-train the models. We give the block-diagram of the self-normalization front end (BISN w/MS-BTS) in Figure 3.

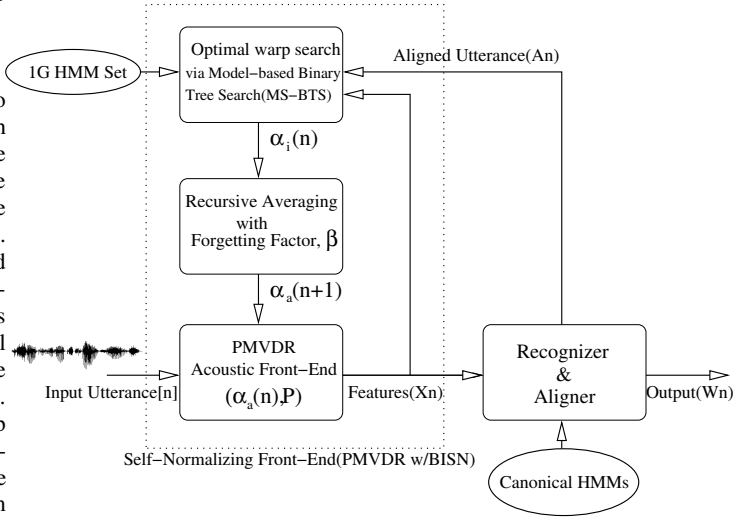

Fig. 3. *The block diagram of the self normalizing front-end (PMVDR w/BISN) in a real-word application scenario.*

Assume that we have the canonical models, λ_N and that recognition is performed for small sections of speech (i.e. utterances). Then the self-normalizing front-end operates as follows; (1) Parameterize the input utterance n with the warp $\alpha_a(n)$, (2) Pass the output features to both recognizer and optimal SNW search algorithm (MS-BTS), (3) Recognize the utterance and pass the transcription (with alignment) information A_n to MS-BTS block, (4) Determine the optimal SNW, i.e. the instantaneous warp for the current utterance n, $\alpha_i(n)$. (5) Pass $\alpha_i(n)$ through a *recursive averaging block* with a forgetting factor (β) to obtain an averaged version, i.e. $\alpha_a(n+1)$. (6) Supply $\alpha_a(n+1)$ to the PMVDR front-end, since this is an *estimate* for the *incoming utterance* n+1. Note that, we never perform recognition twice and sequentially we refine the SNW estimate to accommodate for variations in the vocal tract characteristics even for the same speaker. Recursive averaging also ensures quick adaptation of SNW to changing speakers.

$$\alpha_a(n+1) = \alpha_i(n)(1-\beta) + \alpha_a(n)\beta, \quad n = 0, 1, \ldots, N \tag{3}$$

where $\alpha_a(n)$ is the averaged warp used in the parameterization of utterance n, $\alpha_i(n)$ is the instantaneous warp estimated for utterance n given the features from the front-end X_n and alignment from the recognizer A_n, $\alpha_a(n+1)$ is the estimated warp factor to be used in the parameterization of utterance n+1. As an initial condition for the first utterance, we can choose to use the

center warp of our search space, i.e. $\alpha_a(0)$=α_C=0.57. N is the total number of utterances in the test set. β provides a means for smoothing the SNW estimate and for accounting for changes in vocal tract characteristics. Since the instantaneous SNW $\alpha_i(n)$ is estimated from a short segment of data (as short as one spoken digit), it fluctuates considerably. We give the variation of instantaneous SNW ($\alpha_i(n)$) and recursively averaged SNW ($\alpha_a(n)$) for a comparison in Fig. 4. The fixed self normalization warps obtained from the off-line BISN w/MS-BTS algorithm are also superimposed on the averaged SNW graph. The averaged SNW tracks the fixed SNW permitting slow variations within the same speaker. Allowing some flexibility for the warp factor even within the same speaker compensates for variations which may stem from *Lombard effect*, *stress* or a number of other *physiological* factors. It is also shown that the averaged SNW successfully and quickly adapts to new speakers with *no need to detect speaker turns*. We observed that the particular value of β is not that crucial as long as it is within the range of [0.4, 0.8].

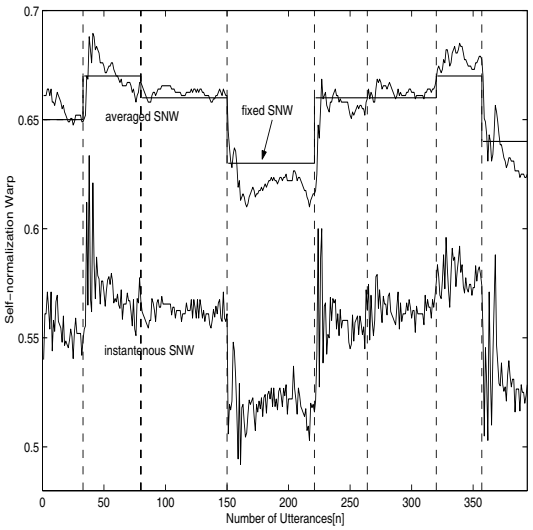

Fig. 4. *The variation of the ins. SNW, avg. SNW and fixed SNW, speaker turns are also marked (the avg. SNW and fixed SNW are shifted upwards by* 0.1 *for proper illustration).*

Lastly, we consider computational efficiency of all algorithms. We use the *number of feature extractions (NFE)* required for the whole system (both for the search(S) and recognition(R)), *the number of likelihood computations (NLC)*, and the *number of recognition passes (NRP)* to evaluate computational efficiency. Table 2 clearly illustrates the computational gain obtained by moving from *classical VTLN* to the *on-the-fly version of BISN w/MS-BTS*. The BISN algorithm eliminates the need to perform 2 times warping on the FFT spectrum. Integration of MS-BTS algorithm within the BISN framework for an on-the-fly application eliminates the need for extracting the features twice. The features extracted for recognition are also passed to the MS-BTS block for SNW estimation for the incoming utterance. Since the estimation is performed sequentially, the need to perform recognition twice is also eliminated. The only drawback of MS-BTS is that we need to store all $1G$ models trained for each point in the search space (17 $1G$ models for BISN) in memory all the time. However, since these are only $1G$ models, they do not require large amount of memory.

Algorithm	**NFE(S+R)**	**NLC**	**NRP**
VTLN (Baseline)	18+1	18	2
BISN	10+1	10	2
BISN/BTS	6+1	6	2
BISN/MS-BTS-off-line	1+1	6	2
BISN/MS-BTS-on-the-fly	0+1	6	1
TOTAL GAIN[%]	**94.7**	**66.7**	**50.0**

Table 2. *Computational complexity (**NFE:** Number of Feature Extractions, **S:** Search, **R:** Recognition, **NLC:** Number of Likelihood Computations, **NRP:** Number of Recognition Passes)*

7. CONCLUSIONS

We proposed a new and efficient algorithm to perform *on-the-fly* speaker normalization which can easily be implemented within the PMVDR front-end. In the classical VTLN, we need to perform 2 times warping of the spectrum, first to account for perceptual considerations and second to normalize for speaker differences. The proposed BISN algorithm, on the other hand, estimates a self-normalization warp (SNW) for each speaker which is shown to perform both perceptual warp and speaker normalization on a single warp. The Model Space-Binary Tree Search (MS-BTS) algorithm was developed to reduce the computational load in the search stage. Moving the search base from the feature to model space reduced the need to extract the features for each point in the search space, which in turn reduced the computational load significantly. A sequential real-time implementation of the BISN w/MS-BTS algorithm also eliminated the need to perform multi-pass recognition. BISN w/MS-BTS algorithm is also more accurate than the classical VTLN with very light computational requirements.

8. REFERENCES

[1] M. J. Hunt, "Spectral signal processing for ASR," in *ASRU*, Keystone, Colorado, 1999, vol. 1, pp. 17–26.

[2] U. H. Yapanel and J H. L. Hansen, "A new perspective on feature extraction for robust in-vehicle speech recognition," in *EUROSPEECH*, Switzerland, 2003.

[3] U. H. Yapanel, J. H. L. Hansen, and S. Dharanipragada, "A new perceptually motivated MVDR-based acoustic front-end(PMVDR) for robust automatic speech recognition," *Submitted to Speech Communications*, August 2004.

[4] A. Andreou, T Kamm, and J. Cohen, "Experiments in vocal tract normalization," in *CAIP workshop:Frontiers in Speech Recognition II*, 1994.

[5] L. Lee and R. C. Rose, "Speaker normalization using effi cient frequency warping procedures," in *ICASSP*, Atlanta, Georgia, 1996, pp. 353–66.

[6] L. Lee and R. C. Rose, "A frequency warping approach to speaker normalization," *IEEE Transactions on Speech & Audio Processing*, vol. vol. 6(1), pp. 49–60, January, 1998.

[7] J. McDonough, W. Byrne, and X. Luo, "Speaker normalization with all-pass transforms," in *ICSLP*, Sydney, Australia, 1998.

[8] J. McDonough, *Speaker Compensation with All-pass Transforms*, Ph.D. thesis, The John Hopkins University, Baltimore, MD, 2000.

[9] M. N. Murthi and B. D. Rao, "All-pole modeling of speech based on the minimum variance distortionless response spectrum," *IEEE Trans. Acoustic Speech and Signal Processing*, vol. 8(3), pp. 221–39, May 2000.

[10] R. Haeb-Umbach, "Investigations on inter-speaker variability in the feature space," in *ICASSP*, Phoenix, Arizona, 1999, pp. 397–400.

[11] U. H. Yapanel, S. Dharanipragada, and J H. L. Hansen, "Perceptual MVDR-based cepstral coeffi cients (PMCCs) for high accuracy speech recognition," in *Proceedings of EUROSPEECH*, Geneva, Switzerland, 2003.

[12] J. H. L. Hansen, X. X. Zhang, M. Akbacak, U. H. Yapanel, B. Pellom, W. Ward, and P. Angkititrakul, *DSP for In-Vehicle and Mobile Systems*, chapter 2. CU-MOVE:Advanced In-Vehicle Speech Systems for Route Navigation, Kluwer Publishers, 2004.

[13] P. Zhan and M. Westphal, "Speaker normalization based on frequency warping," in *ICASSP*, Atlanta, Georgia, 1997.

[14] R. Sinha and S. Umesh, "A method for compensation of jacobian in speaker normalization," in *ICASSP*, Hong Kong, 2003.

[15] B. Pellom, "SONIC: The university of colorado continuous speech recognizer," Tech. Rep. TR-CSLR-2001-01, CSLR, University of Colorado at Boulder, Boulder, Colorado, March 2001.