MULTI-PARAMETER FREQUENCY WARPING FOR VTLN BY GRADIENT SEARCH

Sankaran Panchapagesan and Abeer Alwan

Department of Electrical Engineering
University of California, Los Angeles, U.S.A.
panchap, abeera @ icsl.ucla.edu

ABSTRACT

The current method for estimating frequency warping (FW) functions for vocal tract length normalization (VTLN) is by maximizing the ASR likelihood score by an exhaustive search over a grid of FW parameters. Exhaustive search is inefficient when estimating multi-parameter FWs, which have been shown to give improvements in recognition accuracy over single parameter FWs [8]. Here we develop a gradient search algorithm to obtain the optimal FW parameters for MFCC features, since previous work focussed on PLP cepstral features [8]. The novel calculation involved was that of the gradient of the Mel filterbank with respect to the FW parameters. Even for a single parameter, the gradient search method was more efficient than grid search by a factor of around 1.6 on the average for male children speakers tested on models trained from adult males. When used to estimate multi-parameter sine-log allpass transform (SLAPT, [8]) FWs for VTLN, more than 50% reduction in word error rate was obtained with five parameter SLAPT compared to single-parameter piecewise linear FW.

1. INTRODUCTION

Vocal tract length normalization (VTLN) is one of the most widely used techniques for speaker adaptation, where the spectral variation caused by inter-speaker differences in vocal tract length is reduced by spectral frequency warping (FW) or its equivalent during feature extraction. The estimation and implementation of FW functions for VTLN have been studied extensively ([1] - [9]).

One method of estimating the FW is by comparing formant frequencies of the test speaker with those from a reference speaker in the training set [1, 4, 16]. A second method is to train Gaussian mixture models (GMMs) to estimate a FW parameter [2, 3]. The currently preferred method in speaker adaptation is to perform a search over a grid of warping factors to determine the one that maximizes the likelihood score over adaptation data [5, 6, 13]. This avoids the ambiguities of a formant based approach and has also been shown to give better results [7].

For standard filterbank based Mel frequency cepstral coefficient (MFCC) features commonly used in speech recognition, a simple and efficient implementation of FW is by modifying the center frequencies of the filterbank via the inverse FW function [3]. This is equivalent to a speaker dependent normalization of the Mel curve by FW, and has been shown to give better results than direct warping of the spectrum [6].

Examples of FW functions include linear, piecewise linear (PL) and the Bilinear Transform (BLT), which are controlled by a single parameter (SPFWs), and multiple-parameter (MP) FWs like the allpass transforms (APT) [2, 3, 8]. In [8], it was demonstrated that MPFWs are more effective than SPFWs for front end speaker normalization. The results of studies on estimating optimal Mel curves

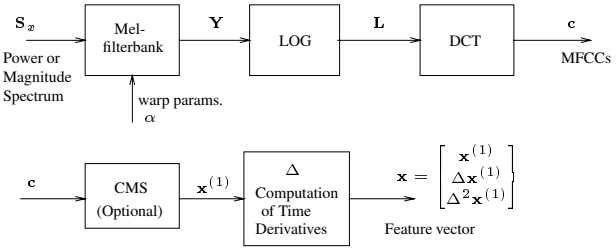

Fig. 1. MFCC feature computation with CMS.

both speaker dependent and independent, showed similar indications [10, 11]. However, the latter studies used computationally intensive exhaustive recognition experiments to estimate the FW.

In this context, when there are multiple FW parameters to be estimated, the current grid search VTLN estimation scheme is also inefficient. This is also especially true when it is desired to perform combined optimization of VTLN with other front end parameters for speaker and channel normalization [13]. Though the advantage of using gradient based techniques to estimate FW functions for speaker adaptation was demonstrated in [8], the features were PLP cepstra and not MFCCs.

In this paper, we develop a method for estimating MPFWs for MFCC features by gradient based optimization of the ASR likelihood score. The estimated MPFWs are used to obtain speaker specific Mel curves for improved recognition performance. The novel part of the calculation is that of the gradient of the Mel filterbank with respect to the FW parameters. The efficiency of gradient search over grid search is quantified for single-parameter VTLN.

In Sec. 2, we discuss the objective function used for VTLN estimation. The steps involved in the calculation of the gradient of the objective function are given in Sec. 3. In Sec. 4, the computation of the gradient of the filterbank with respect to the FW parameter is described, and the computational requirements for the gradient of the objective function are discussed in Sec. 5. Experimental results are presented and discussed in Sec. 6.

2. THE VTLN OBJECTIVE FUNCTION

Fig. 1 shows the extraction scheme for standard MFCC based features, using a filterbank (Fig. 2 (a)). Note that the order of the (optional) cepstral mean subtraction (CMS) and time derivative computation blocks can be interchanged

In [3, 7, 15], the optimal warp factor(s) is(are) chosen by maximizing the likelihood score of the recognizer over the adaptation data:

$$\hat{\alpha} = \arg\max_{\alpha} [\log P(\mathbf{X}^{\alpha}, \Theta^{\alpha} | W, \Lambda)] \quad (1)$$

where α is(are) the FW parameter(s), $\mathbf{X}^{\alpha}$ is the normalized adaptation data, W is the word (or other unit) transcription, Λ are the

corresponding HMMs, and Θ^α are the HMM states with which $\mathbf{X}^\alpha$ are aligned by the Viterbi algorithm during ASR decoding.

As noted in [8, 9], this is not strictly a ML scheme since the Jacobian determinant factor for the normalization function is not included in the likelihood. In fact, since MFCC features are obtained from the spectrum after non-invertible operations such as filterbank integration and dimensionality reduction after DCT, the final normalized features are not invertible functions of the unnormalized features and the Jacobian determinant cannot be computed.

However, we will continue to use it as the objective function because of its simplicity and effectiveness in improving recognition accuracy, as evidenced by the experimental results and also its popularity in practice. The extension of the calculations to more rigorous but also more computationally intensive objective functions like MMI is straightforward.

For simplicity of the equations, let there be only one adaptation utterance, $\mathbf{X}^\alpha = \{\mathbf{x}_1^\alpha, \mathbf{x}_2^\alpha, \ldots, \mathbf{x}_T^\alpha\}$ where T is the number of feature vectors, and let $\Theta = \{s_1, s_2, \ldots, s_T\}$ be the Viterbi state sequence. An approximation made for fast VTLN estimation is to obtain the best frame-state alignment of the adaptation data with the HMMs just once with the un-normalized features, and keep the alignment fixed in the optimization [6]. Then, since the state sequence probability does not depend on α, the objective function of Eq. 1 may be simplified to:

$$\mathcal{F}(\alpha) = \sum_{t=1}^{T} \log \sum_{m=1}^{M} c_{tm} \mathcal{N}(\mathbf{x}_t^\alpha; \mu_{tm}, \Sigma_{tm}) \tag{2}$$

where $\sum_{m=1}^{M} c_{tm} = 1$ for the Gaussian mixture state output distribution.

3. GRADIENT OF THE OBJECTIVE FUNCTION

The gradient of the objective function in Eq. 2 is

$$\nabla_\alpha \mathcal{F}(\alpha) = \sum_{t=1}^{T} J_{\mathbf{x}_t^\alpha}(\alpha)^T \left(\sum_{m=1}^{M} \gamma_{tm} \Sigma_{tm}^{-1} (\mu_{tm} - \mathbf{x}_t^\alpha) \right) \tag{3}$$

where for two vector variables $\mathbf{y}$ and $\mathbf{z}$, $J_{\mathbf{z}}(\mathbf{y})$ denotes the Jacobian matrix of partial derivatives $\left[\frac{\partial \mathbf{z}_i}{\partial \mathbf{y}_j}\right]$, and γ_{tm} is the posterior probability of mixture m of state s_t given that the feature vector $\mathbf{x}_t^\alpha$ was generated by state s_t. Here we have used the fact that $\nabla_{\mathbf{x}} \mathcal{N}(\mathbf{x}; \mu, \Sigma) = \mathcal{N}(\mathbf{x}; \mu, \Sigma) \cdot \left(\Sigma^{-1}(\mu - \mathbf{x})\right)$.

Recall the extraction scheme in Fig. 1, and let the intermediate variables during feature extraction be denoted as in the figure. To compute $J_{\mathbf{x}_t^\alpha}(\alpha)$, we work backwards computing the Jacobian of each block. The calculations for the time derivative, CMS, DCT and Log blocks are quite simple and are similar to those in [12].

- **Time derivative computation and concatenation**:
  Since differentiation with respect to time (Δ) is a linear operation the order of differentiations with respect to time and α are interchangeable.

$$\Rightarrow J_{\mathbf{x}_t^\alpha}(\alpha) = \begin{bmatrix} J_{\mathbf{x}_t^{(1)}}(\alpha) \\ J_{\Delta \mathbf{x}_t^{(1)}}(\alpha) \\ J_{\Delta^2 \mathbf{x}_t^{(1)}}(\alpha) \end{bmatrix} = \begin{bmatrix} J_{\mathbf{x}_t^{(1)}}(\alpha) \\ \Delta J_{\mathbf{x}_t^{(1)}}(\alpha) \\ \Delta^2 J_{\mathbf{x}_t^{(1)}}(\alpha) \end{bmatrix}$$

- **CMS**:

$$\mathbf{x}_t^{(1)} = \mathbf{c}_t - \frac{1}{T} \sum_{\tau=1}^{T} \mathbf{c}_\tau$$

$$\Rightarrow J_{\mathbf{x}_t^{(1)}}(\alpha) = J_{\mathbf{c}_t}(\alpha) - \frac{1}{T} \sum_{\tau=1}^{T} J_{\mathbf{c}_\tau}(\alpha)$$

- **DCT**:

$$\begin{aligned} \mathbf{c}_t &= C \cdot \mathbf{L}_t \\ \Rightarrow J_{\mathbf{c}_t}(\alpha) &= C \cdot J_{\mathbf{L}_t}(\alpha) \end{aligned}$$

- **LOG**:

$$\begin{aligned} \mathbf{L}_t &= \log \mathbf{Y}_t \\ \Rightarrow J_{\mathbf{L}_t}(\alpha) &= \text{diag}\left(\frac{1}{\mathbf{Y}_t(1)}, \ldots, \frac{1}{\mathbf{Y}_t(N_f)} \right) \cdot J_{\mathbf{Y}_t}(\alpha) \end{aligned}$$

  where N_f is the number of filters in the Mel filterbank.

- **Mel Filterbank**:
  The computation of $J_{\mathbf{Y}_t}(\alpha) = \left[\frac{\partial \mathbf{Y}_t}{\partial \alpha}\right]$, is novel and more involved, and is outlined next.

4. COMPUTATION OF $J_{\mathbf{Y}}(\alpha)$

We consider the Mel filterbank output from a single speech frame and ignore the time-dependence since it is not essential to the calculations. The output of the ith Mel bin is given by

$$\mathbf{Y}(i) = \int_0^{f_s/2} H_i(f; \alpha) S_x(f) df \tag{4}$$

where $S_x(f) = |X(f)|$ or $|X(f)|^2$ is the magnitude or power spectrum of a speech frame, f_s is the sampling frequency, and $H_i(f; \alpha)$ is the ith filter in the Mel filterbank which depends on the FW parameter through the center frequencies. Algorithmically, the integral is computed using a summation approximation and the FFT. In the analysis, we use the integral form for $\mathbf{Y}(i)$.

To obtain the Jacobian $J_{\mathbf{Y}}(\alpha)$, notice that in the expression in Eq. 4, only the filter $H_i(f; \alpha)$ depends on α. So:

$$\frac{\partial \mathbf{Y}(i)}{\partial \alpha} = \int_0^{f_s/2} \left(\frac{\partial}{\partial \alpha} H_i(f; \alpha) \right) S_x(f) df \tag{5}$$

The weights of the ith filter, $H_i(f; \alpha)$, are assumed to be triangular and half-overlapping with the $i-1$st and $i+1$st filters. Let f_i^0, $1 \le i \le N_f$ be the original center frequencies of the filter bank, and let them be modified to f_i^α by the FW function ($f_1 = 0$ and $f_{N_f} = f_s/2$ are not changed). Using the unit step function $U(\cdot)$, $H_i(f; \alpha)$ may be expressed as:

$$\begin{aligned} H_i(f; \alpha) = &\left(\frac{f - f_{i-1}^\alpha}{f_i^\alpha - f_{i-1}^\alpha} \right) [U(f - f_{i-1}^\alpha) - U(f - f_i^\alpha)] \\ &+ \left(\frac{f_{i+1}^\alpha - f}{f_{i+1}^\alpha - f_i^\alpha} \right) [U(f - f_i^\alpha) - U(f - f_{i+1}^\alpha)] \end{aligned} \tag{6}$$

We can write

$$\frac{\partial}{\partial \alpha} H_i(f; \alpha) = \sum_j \frac{\partial H_i(f; \alpha)}{\partial f_j^\alpha} \cdot \frac{\partial f_j^\alpha}{\partial \alpha} \tag{7}$$

For brevity, denote $\frac{\partial H_i(f;\alpha)}{\partial f_j^\alpha}$ by $H'_{i,j}$. It is clear that $H'_{i,j}(f)$ will be non-zero only for $j = i-1, i, i+1$ (except of course, for $i = 1$ and $i = N_f$, where $H'_{1,0}$ and H'_{N_f, N_f+1} are not defined). To compute these, we differentiate Eq. 6 using the product rule, and use the facts $\frac{dU(t)}{dt} = \delta(t)$, and $t \cdot \delta(t) \equiv 0$, where $\delta(t)$ is the Dirac delta function, whose use as the *generalized* derivative of the unit step function is common in system theory [14]. Though the calculations could be performed without using it, the Dirac delta function is a very useful mathematical device which simplifies calculations. It can then be shown that

$$H'_{i,i-1} = -\frac{f_i^\alpha - f}{(f_i^\alpha - f_{i-1}^\alpha)^2} \cdot [U(f - f_{i-1}^\alpha) - U(f - f_i^\alpha)] \tag{8}$$

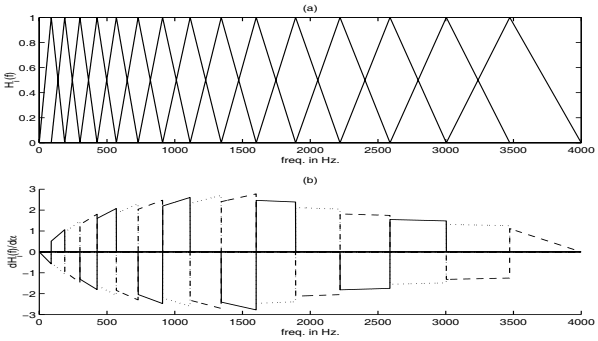

Fig. 2. The shapes of (a) the Mel filter bank, $H_i(f;\alpha)$, and (b) the derivative filterbank, $\frac{\partial}{\partial\alpha}H_i(f;\alpha)$, for the PL FW at $\alpha = 1$. In (b), the filters are alternately plotted with solid, dotted and dash-dotted lines. f_s is 8kHz and the number of filters is 15.

$$H'_{i,i} = -\frac{f - f^{\alpha}_{i-1}}{(f^{\alpha}_{i} - f^{\alpha}_{i-1})^2} \cdot [U(f - f^{\alpha}_{i-1}) - U(f - f^{\alpha}_{i})] + \frac{f^{\alpha}_{i+1} - f}{(f^{\alpha}_{i+1} - f^{\alpha}_{i})^2} \cdot [U(f - f^{\alpha}_{i}) - U(f - f^{\alpha}_{i+1})] \quad (9)$$

$$H'_{i,i+1} = \frac{f - f^{\alpha}_{i}}{(f^{\alpha}_{i+1} - f^{\alpha}_{i})^2} \cdot [U(f - f^{\alpha}_{i}) - U(f - f^{\alpha}_{i+1})] \quad (10)$$

Here we use two kinds of FW functions, first the piecewise linear (PL) FW function, which is described by a single parameter α that controls the initial slope of the function:

$$f^{\alpha}_{i} = \begin{cases} \alpha f & 0 \le f \le f_r \\ \alpha f_r + \left(\frac{f_s/2 - \alpha f_r}{f_s/2 - f_r}\right)(f - f_r), & f_r < f \le f_s/2 \end{cases} \quad (11)$$

$$\frac{\partial}{\partial\alpha} f^{\alpha}_{i} = \begin{cases} f & 0 \le f \le f_r \\ f_r \cdot \left(\frac{f_s/2 - f}{f_s/2 - f_r}\right) & f_r < f \le f_s/2 \end{cases} \quad (12)$$

where f_r is a fixed reference frequency, around $0.7 f_s/2$ [1].
We also consider the sine-log all-pass transform (SLAPT) a MPFW [8]:

$$f^{\alpha}_{i} = f + \frac{f_s}{2} \cdot \sum_{k=1}^{K} \alpha_k \sin\left(\frac{2k\pi f}{f_s}\right),\ 0 \le f \le \frac{f_s}{2} \quad (13)$$

$$\frac{\partial}{\partial\alpha_k} f^{\alpha}_{i} = \left(\frac{f_s}{2}\right) \cdot \sin\left(\frac{2k\pi f}{f_s}\right),\ 1 \le k \le K \quad (14)$$

where is f_s is the sampling frequency, and $\alpha = [\alpha_1 \ldots \alpha_K]$. SLAPT FWs have the advantage of mathematical simplicity as well as the ability to approximate arbitrary FW functions by choosing K sufficiently large.

Using Eqs. 7-12, the expression for $\frac{\partial}{\partial\alpha}H_i(f;\alpha)$ can be computed, and used to form a filterbank which has the same number of channels as the original filterbank, for each FW parameter. This is shown in Fig 2 (b) for the PL FW at $\alpha = 1$. $J_{\mathbf{Y}}(\alpha)$ may then be computed from Eq. 5 using a summation approximation.

5. COMPUTATIONAL COST OF THE GRADIENT

From the equations in Sec. 3, we note that in the feature extraction, for the Δ, CMS, and DCT blocks, the computation of the derivatives are of exactly the same complexity as the blocks themselves. In the case of the LOG block, the computation of the derivative is less intensive than the block itself. From the conclusions of Sec. 4, it follows that the derivative of the filterbank block with respect to each parameter requires approximately the same number of computations as the original filterbank. Since this is typically the most expensive block in the feature extraction, comparing Eqs. 2 and 3 we can say that if the computational cost of $\mathcal{F}(\alpha)$ is C, then that of $\nabla_\alpha\mathcal{F}(\alpha)$ is approximately equal to nC where n is the number of FW parameters.

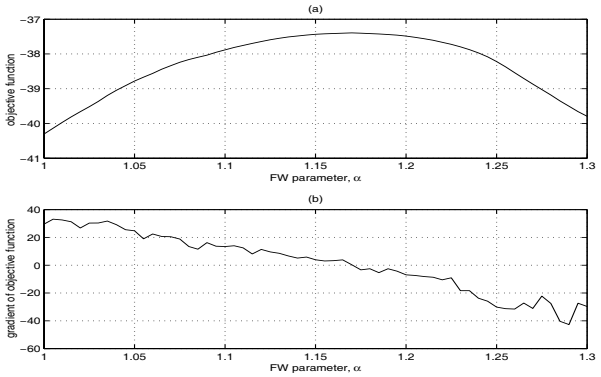

Fig. 3. (a) $\mathcal{F}(\alpha)$ and (b) $\nabla\mathcal{F}(\alpha)$ with respect to α for PL VTLN

6. EXPERIMENTAL RESULTS

The focus of our research is on improving children's speech recognition, where VTLN has been shown to give significant reductions in recognition word error rate (WER). Since there doesn't yet exist a database for large vocabulary recognition (LVR) of children's speech, we tested VTLN on connected digit recognition of children's speech using the TIDIGITS database. But many of the results of this paper should also be applicable also to LVR of children's speech since the objective function is likely to have a similar shape.

Monophone models for connected digit recognition were trained from the adult male speakers and tested on the male children from TIDIGITS. There were 21 HMMs, including 19 monophones, silence and short pause models and the monophone HMMs had 4 emitting states each and 3 Gaussian mixtures per state. The first 13 MFCCs and their first and second time derivatives, were used as features, with CMS performed on each utterrance. The ten children with the worst WER were chosen for further experiments. The baseline WER was 62.33 %, and is comparable to that in [16].

Eleven adaptation utterrances, one of each of the 11 digits (zero to nine and 'oh') were used to estimate the FW for each boy test speaker.

As a verification of our calculations, $\mathcal{F}(\alpha)$ and $\nabla\mathcal{F}(\alpha)$ were plotted for the PL FW for a typical speaker, and are shown in Fig. 3. One modification was to normalize $\mathcal{F}(\alpha)$ by the total number of speech frames. Though $\mathcal{F}(\alpha)$ appears to have a smooth concave shape, $\nabla\mathcal{F}(\alpha)$ appears to be non-monotone and has local variations. These may be caused by the summation approximation used to compute $\nabla\mathcal{F}(\alpha)$ with the FFT, since the the derivative filterbank has discontinuous steps. Otherwise, it is seen that $\nabla\mathcal{F}(\alpha) = 0$ roughly at the point of maximum of $\nabla\mathcal{F}(\alpha)$.

In the optimization for PL FW, because of the non-smooth behavior of $\nabla\mathcal{F}(\alpha)$, a quasi-Newton method would be inappropriate and a simple gradient search with backtracking was used and the initial step size was coarsely tuned for fast convergence. The stopping condition was a tolerance on the magnitude of the gradient, which is related to the accuracy of the estimated parameter. For SLAPT with multiple parameters, a quasi-Newton method with BFGS update was found to converge faster than the plain gradient descent method.

Table 1. Results of MPFW VTLN Experiments: models trained on adult males and tested on 10 'difficult' male children from TIDIGITS

FW Func.	No Warp	PL FW	SLAPT-1	SLAPT-2
Avg. WER	62.23 %	22.64 %	20.25 %	18.55 %
	SLAPT-3	SLAPT-4	SLAPT-5	SLAPT-6
	17.07 %	11.94 %	10.91 %	10.58 %

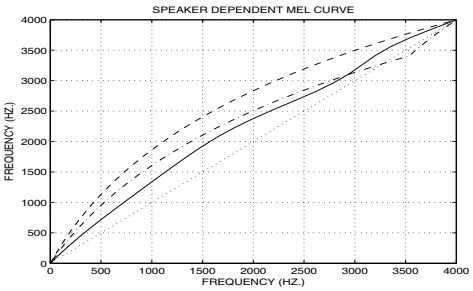

Fig. 4. Speaker dependent Mel curve. The dashed line shows the regular Mel curve, the dash-dotted line shows the Mel curve estimated with PL FW, the solid line shows the estimated Mel curve with SLAPT-4 FW, and the dotted line is the identity curve.

The results of VTLN experiments are shown in Table 1. Adaptation using SLAPT FW with n parameters is denoted SLAPT-n. It is seen that as the number of parameters in the SLAPT FW is increased, the WER decreases, and SLAPT-5 shows an improvement of more than 50% compared to the single-parameter PL FW.

As mentioned in Sec. 1, VTLN by modifying the center frequencies of the filterbank may also be viewed as a speaker dependent normalization of the Mel curve. Fig. 4 shows the Mel curves estimated for a test speaker with PL FW and SLAPT-4. For this speaker, the WER decreased from 37.6% with PL FW to 8.3% with SLAPT-4.

Efficiency: We wish to quantify the computational efficiency of the gradient search over the grid search for the case of single parameter PL FW (n=1). When comparing the two methods, the accuracy of $\hat{\alpha}$ in the gradient search which is controlled by the stopping condition, was chosen so as to be approximately the same as in the grid search, where the α step is typically 0.02 for PL FW [3, 5, 15]. Therefore, the WERs are usually the same for both methods.

Since the costs of evaluating $\mathcal{F}(\alpha)$ and $\nabla\mathcal{F}(\alpha)$ are both approximately C (Sec. 5), the total cost of an algorithm may be measured as an integral multiple of C. In grid search, the warp factor is initially 1 (no warping) and then increased (for children speakers tested on adult-trained models) gradually in steps till the likelihood score starts decreasing. If the optimal warp factor is $\hat{\alpha}$ and the step size is $\Delta\alpha$, the number of function evaluations is $(\hat{\alpha}/\Delta\alpha + 2)$ and the cost is $(\hat{\alpha}/\Delta\alpha+2)C$. In gradient search, if the number of iterations is m, there are $(m+1)$ evaluations each of $\mathcal{F}(\alpha)$ and $\nabla\mathcal{F}(\alpha)$, and so the cost is $2(m+1)C$. Backtracking during any iteration in the gradient search involves an additional cost of C. On the average over all the test speakers, the gradient search was found to be more efficient than the grid search by a factor of around 1.6. For some children, $\hat{\alpha}$ was as high as 1.25 and the efficiency factor was around 3.

For $n > 1$, the grid search becomes more tedious and the computational savings of a tuned gradient search or a quasi-Newton method using the BFGS update are expected to be greater.

7. CONCLUSIONS

A gradient search algorithm was developed for VTLN estimation with MFCC features. The novel calculation was that of the gradient of the filterbank with respect to the FW parameters. The cost of computing the gradient of the VTLN objective function is approximately the number of FW parameters times the cost of evaluating the function. For male children speakers tested on models trained from adult males, the algorithm was used to estimate multiple-parameter SLAPT FW for VTLN, and more than 50% relative reduction in word error rate was obtained compared to single-parameter PL VTLN. For single parameter PL VTLN, the algorithm was more efficient than the widely used grid search by a factor of around 1.6. For multiple parameters, grid search would be inefficient and the computational savings of gradient search would be greater. The method also has potential for combined optimization of other front-end parameters for fast speaker and channel normalization and different objective functions like MMI need to be explored. [1]

8. REFERENCES

[1] E. Eide and H. Gish, "A parametric approach to vocal tract length normalization," *Proc. ICASSP*, pp.346-349, 1996.

[2] S. Wegmann, D. McAllaster, J. Orloff and B. Peskin, "Speaker normalization on conversational telephone speech," *Proc ICASSP*, pp.339-341, 1996.

[3] L. Lee and R. C. Rose, "Speaker normalization using efficient frequency warping procedures," *ICASSP*, pp.353-356, 1996.

[4] E. B. Gouvea and R. M. Stern, "Speaker normalization through formant-based warping of the frequency scale," *Eurospeech* 1997, vol. 3, pp.1139-1142.

[5] A. Potamianos and R. C. Rose, "On combining frequency warping and spectral shaping in HMM-based speech recognition," *Proc. ICASSP*, Vol. , pp.1275-1278, April 1997.

[6] P. Zhan and A. Waibel, "Vocal tract length normalization for large vocabulary continuous speech recognition," Technical Report, CMU-CS-97-148, May 1997.

[7] P. Zhan and M. Westphal, "Speaker normalization based on frequency warping," *Proc. ICASSP* vol. 2, pp.1039-1042, 1997.

[8] J. W. McDonough, "Speaker Compensation with all-pass transforms," Ph.D. dissertation, JHU, Baltimore, Maryland, 2000.

[9] M. Pitz, S. Molau, R. Schlueter, H. Ney, "Vocal Tract normalization equals linear transformation in cepstral space", *Eurospeech 2001*, pp.721-724.

[10] T. Kamm, H. Hermansky and A. G. Andreou, "Learning the Mel-scale and optimal VTN mapping," Technical Report, CSLP, Johns Hopkins University, 1997.

[11] G. Stemmer et al, "Acoustic Normalization of Children's Speech," *Eurospeech* 2003, pp.1313-1316.

[12] K. Visweswariah and R. Gopinath, "Adaptation of front end parameters in a speech recognizer," *Interspeech*-04, 21-24.

[13] R. Zhao and Z. Wang, "Robust speech recognition based on spectral adjusting and warping," ICASSP 2005.

[14] N. Levan, "Systems and Signals," 3rd ed., Optimization Software Publications division, Springer Verlag, 1983.

[15] A. Potamianos and S. Narayanan, "Robust recognition of children's speech," *IEEE Trans. SAP*, vol.11, No. 6, Nov. 2003.

[16] X. Cui and A. Alwan, "Adaptation of Children's Speech with Limited Data Based on Formant-like Peak Alignment," *Computer Speech and Language*, to appear.

[1] This work was supported in part by the NSF