HMM Adaptation Using Linear Spline Interpolation with Integrated Spline Parameter Training for Robust Speech Recognition

Michael L. Seltzer and Alex Acero

Speech Technology Group, Microsoft Research, Redmond, WA USA

{mseltzer,alexac}@microsoft.com

Abstract

We recently proposed a method for HMM adaptation to noisy environments called Linear Spline Interpolation (LSI). LSI uses linear spline regression to model the relationship between clean and noisy speech features. In the original algorithm, stereo training data was used to learn the spline parameters that minimize the error between the predicted and actual noisy speech features. The estimated splines are then used at runtime to adapt the clean HMMs to the current environment. While good results can be obtained with this approach, the performance is limited by the fact that the splines are trained independently from the speech recognizer and as such, they may actually be suboptimal for adaptation. In this work, we introduce a new Generalized EM algorithm for estimating the spline parameters using the speech recognizer itself. Experiments on the Aurora 2 task show that using LSI adaptation with splines trained in this manner results in a 20% improvement over the original LSI algorithm that used splines estimated from stereo data and a 28% improvement over VTS adaptation.

Index Terms: HMM adaptation, noise robustness, vector Taylor series

1. Introduction

It is well known that the performance of speech recognition systems degrades in noise. One of the most effective ways to improve the robustness of such systems is to adapt the acoustic models to the current environmental conditions. While such adaptation can be performed in a data driven way, better performance is typically obtained when the relationship between clean speech, noise and noisy speech is exploited. Unfortunately, the best way to do so is not obvious, as this relationship is nonlinear in the model domain.

Several different methods for handling this nonlinearity have been proposed. For example, in data-driven Parallel Model Combination, Monte Carlo sampling is used to generate samples of the noisy speech distribution from the distributions of clean speech and noise [1]. In Vector Taylor Series (VTS) adaptation, e.g. [2], the nonlinear function that describes noisy speech features as a function of the clean speech and noise features is linearized around expansion points defined by the speech and noise models. In [3], an Unscented Transform is used to estimate the noisy speech distribution using a small set of speech and noise sample points.

We recently introduced a novel HMM adaptation scheme called Linear Spline Interpolation (LSI) [4] in which the nonlinearity is modeled using linear spline regression. The transformations used to adapt the acoustic models are determined by interpolating the parameters of the linear spline. The spline parameters are learned offline by minimizing the error between the predicted noisy speech and actual noisy speech using a corpus of stereo clean/noisy training data.

While LSI was shown to outperform VTS adaptation, its performance may be limited by the fact that the spline parameters are trained independently from the speech recognizer, and as such may actually be sub-optimal for acoustic model adaptation and recognition. Specifically, the spline is trained by minimizing the mean-squared error of a collection of stereo frames, while the speech recognizer is trained using a HMM-based maximum likelihood criterion.

In this work, we significantly improve the LSI adaptation algorithm by designing an integrated spline parameter training algorithm that learns the spline parameters using the same maximum likelihood criterion as the speech recognizer. We develop a Generalized EM algorithm that uses multi-condition training data and the recognizer itself to optimize the spline parameters. We demonstrate through a series of experiments that training the spline parameters in this way results in a substantial improvement in accuracy.

The remainder of the paper is as follows. In Section 2 we review the LSI adaptation algorithm. We then present the new algorithm for training of the spline parameters in Section 3. In Section 4, we evaluate the performance of the proposed approach. Finally, we present some conclusions in Section 5.

2. Linear Spline Interpolation

If x, n, and y are the log mel spectral representations of the clean speech, noise, channel and noisy speech, respectively, then y can be expressed as

$$y = n + \log(1 + e^{(x-n)} + 2\alpha e^{(x-n)/2}) \tag{1}$$

where α is a random variable that represents the relative phase between the clean speech and the noise [5]. Because $E[\alpha] = 0$, most algorithms in the literature ignore its effect and operate on the simplified expression $y = n + \log(1 + e^{(x-n)})$. This expression also shows the relationship between the *a priori* SNR $u = x - n$ and the *a posteriori* SNR $v = y - n$ in the log mel spectral domain:

$$v = \log(1 + e^{u}) \tag{2}$$

In this work, we use a linear spline regression to model the relationship in (2). In linear spline regression, pairs of data (u, v) are modeled using K segments, each defined by a linear regression. The spline parameters are estimated under the constraint that neighboring regression lines must intersect at the segment boundaries, called *knots*. Thus, the kth segment is defined as

$$v = a_k u + b_k + \epsilon_k, \quad \forall\, U_{k-1} < u \leq U_k \tag{3}$$

where a_k, b_k, and ϵ_k are the slope, y-intercept, and error of the kth line segment, respectively, and U_{k-1} and U_k are the knots that define the segment boundaries. In each segment, the error ϵ_k is modeled as a zero-mean Gaussian with variance ψ_k^2. This variance is a key component of the LSI algorithm as it implicitly captures the uncertainty caused by the phase asynchrony between the clean speech and noise.

2.1. MMSE Estimate of Noisy Speech

We can use the linear spline to construct an MMSE estimate of y given x and n. To do so, we first construct an MMSE estimate of the *a posteriori* SNR v given the *a priori* SNR u. This estimate is a weighted sum of the means of the segment-conditional posterior distribution $p(v|u,k) = \mathcal{N}(v; a_k u + b_k, \sigma^2_{\epsilon_k})$. It can be written

$$\hat{v} = \sum_k w_k \int v p(v|u,k) dv = \sum_k w_k (a_k u + b_k). \quad (4)$$

where w_k represents the probability that u lies in the kth spline segment. By substituting the definitions of u and v into (4) and rearranging terms, the MMSE estimate of y can be computed as

$$\hat{y} = (1 - \textstyle\sum_k w_k a_k) n + (\sum_k w_k a_k) x + \sum_k w_k b_k \quad (5)$$

The corresponding mean and variance of $p(y)$ can be by computing the first and second moments of y using (5). This provides the following adaptation formulae for log mel spectral components:

$$\mu_y = (1 - \sum_k w_k a_k)\mu_n + (\sum_k w_k a_k)\mu_x + \sum_k w_k b_k \quad (6)$$

$$\sigma_y^2 = (1 - \sum_k w_k a_k)^2 \sigma_n^2 + (\sum_k w_k a_k)^2 \sigma_x^2 + \sum_k w_k^2 \sigma^2_{\epsilon_k} \quad (7)$$

Recall that the interpolation weights w_k in (6) and (7) represent the probability that u lies in the kth spline segment. To compute these weights, we first compute the $p(u)$ from $p(x)$ and $p(n)$. Because x and n are independent Gaussian random variables, u is also Gaussian with mean $\mu_u = \mu_x - \mu_n$ and variance $\sigma_u^2 = \sigma_x^2 + \sigma_n^2$. We then compute w_k as

$$w_k = \int_{U_k - 1}^{U_k} p(u) du = \Phi(U_k; \mu_u, \sigma_u^2) - \Phi(U_{k-1}; \mu_u, \sigma_u^2)$$

where Φ is the continuous density function (CDF) of a Gaussian distribution, and $\{U_{k-1}, U_k\}$ are the knots of the kth segment.

2.2. HMM Adaptation using Linear Spline Interpolation

To transform the log mel spectral adaptation formulae to the cepstral domain, we define the following terms

$$\boldsymbol{A} = diag\left(\textstyle\sum_k w_{1k} a_{1k},\ \cdots, \sum_k w_{Lk} a_{Lk}\right) \quad (8)$$

$$\boldsymbol{b} = \left[\textstyle\sum_k w_{1k} b_{1k},\ \cdots, \sum_k w_{Lk} b_{Lk}\right]^T \quad (9)$$

$$\boldsymbol{\Psi}_\epsilon = diag\left(\textstyle\sum_k w_{1k}^2 \psi_{1k}^2,\ \cdots, \sum_k w_{Lk}^2 \psi_{Lk}^2\right) \quad (10)$$

where $\left\{a_{lk}, b_{lk}, \psi_{lk}^2\right\}$ are the spline parameters for the kth spline segment of the lth log mel coefficient. We additionally define $\boldsymbol{e} = \boldsymbol{Cb}$, $\boldsymbol{F} = \boldsymbol{CAD}$ and $\boldsymbol{G} = \boldsymbol{I} - \boldsymbol{F}$, where $\boldsymbol{C}$ is the truncated DCT and $\boldsymbol{D}$ is the pseudo-inverse of $\boldsymbol{C}$. The cepstral model parameters can now be transformed as

$$\boldsymbol{\mu}_y = \boldsymbol{F}\boldsymbol{\mu}_x + \boldsymbol{G}\boldsymbol{\mu}_n + \boldsymbol{e} \quad (11)$$

$$\boldsymbol{\Sigma}_y = \boldsymbol{F}\boldsymbol{\Sigma}_x\boldsymbol{F}^T + \boldsymbol{G}\boldsymbol{\Sigma}_n\boldsymbol{G}^T + \boldsymbol{C}\boldsymbol{\Psi}_\epsilon\boldsymbol{C}^T \quad (12)$$

Note that even though $\boldsymbol{\Sigma}_y$ is a full matrix, we assume it is diagonal for decoding purposes.

The adaptation equations for the delta model parameters are similar to those of the static parameters. If we assume the spline weights w_{lk} are constant over the span of the delta features, the adaptation formulae for the delta components are

$$\boldsymbol{\mu}_{\Delta y} = \boldsymbol{F}\boldsymbol{\mu}_{\Delta x} + \boldsymbol{G}\boldsymbol{\mu}_{\Delta n} \quad (13)$$

$$\boldsymbol{\Sigma}_{\Delta y} = \boldsymbol{F}\boldsymbol{\Sigma}_{\Delta x}\boldsymbol{F}^T + \boldsymbol{G}\boldsymbol{\Sigma}_{\Delta n}\boldsymbol{G}^T + \boldsymbol{C}\boldsymbol{\Psi}_{\Delta\epsilon}\boldsymbol{C}^T \quad (14)$$

Note that we typically assume the noise is stationary so that $\boldsymbol{\mu}_{\Delta n} = 0$. The delta-delta parameters are adapted in the same way, simply substituting the delta-delta parameters for the delta parameters in (13) and (14).

2.3. Spline parameter estimation using stereo data

In [4, 6], the spline parameters were learned directly from features extracted from stereo data by minimizing the error between the actual and predicted noisy speech values, as

$$\epsilon_k^2 = \sum_{n=1}^{N_k} \left\{a_k u_n + b_k - \ln(1 + e^{u_n})\right\}^2,\ U_{k-1} < u_n \leq U_k \quad (15)$$

where N_k is the number of samples that lie in the kth segment. Note that we are minimizing the distance to the mode of the data given by (2). The parameters that minimize (15) for all K segments and satisfy the spline adjacency constraints are found by solving a system of linear equations, as described in [7].Once the $\{a_k, b_k\}$ parameters have been estimated, the spline variance parameter is simply $\psi_k^2 = \epsilon_k^2 / N_k$.

3. Maximum Likelihood Spline Estimation

In this section, we propose a new method to estimate the spline parameters using an HMM-based maximum likelihood criterion. Let us assume that we have a speech recognizer trained from clean speech with parameters defined as Λ_X. We also assume that we have a multi-condition training set $\mathcal{Y} = \{Y^{(i)}\}_{i=1}^I$. Each utterance in the training set has an associated distortion model $\Phi_N^{(i)} = \{\boldsymbol{\mu}_n^{(i)}, \boldsymbol{\Sigma}_n^{(i)}\}$. Finally, let us define the complete set of spline parameters as $\Phi_S = \{a_{lk}, b_{lk}, \psi_{lk}^2\ \forall\ (l,k)\}$. We are seeking the set of spline parameters Φ_S that maximizes the likelihood of the training data when LSI adaptation is performed on the clean acoustic model. We can define this mathematically as

$$\Phi_S^{(ML)} = \underset{\Phi_S}{\text{argmax}} \prod_{i=1}^{I} \mathcal{L}(Y^{(i)}, \Lambda_Y^{(i)}) \quad (16)$$

where $\Lambda_Y^{(i)} = LSI(\Lambda_X, \Phi_N^{(i)}, \Phi_S)$ is the acoustic model adapted for the ith utterance using LSI as described in Section 2. Because the expression in (16) cannot be maximized directly, we start with the following auxiliary function

$$Q(\Phi_S, \hat{\Phi}_S) = \sum_i \sum_t \sum_s \gamma_t^{(is)} \log(p(\boldsymbol{y}_t^{(i)}|s)) \quad (17)$$

where $\gamma_t^{(is)}$ is the posterior probability of the adapted Gaussian s at frame t of utterance i and $p(\boldsymbol{y}_t^{(i)}|s) = \mathcal{N}(\boldsymbol{y}_t^{(i)}; \boldsymbol{\mu}_y^{(is)}, \boldsymbol{\Sigma}_y^{(is)})$ is the likelihood of the observation under the adapted model with mean and variance defined according to (11)–(14). Our goal is find the

spline parameters Φ_S that maximize the likelihood defined by (17). To do so, we employ a Generalized EM approach in which the M-step is computed using gradient descent as

$$\hat{\Phi}_S = \Phi_S + \eta \Delta \Phi_S \tag{18}$$

where $\Delta\Phi_S$ is the gradient of auxiliary function $Q(\Phi_S, \hat{\Phi}_S)$ with respect to the parameter vector Φ_S and η defines the learning rate. To compute $\Delta\Phi_S$, we need the partial derivatives of $Q(\Phi_S, \hat{\Phi}_S)$ with respect to each of the spline parameters.

3.1. Gradient update for the spline slope parameters

If we assume diagonal covariance matrices, then we can define the gradient of Q with respect to a spline slope parameter a_{lk} as

$$\frac{\partial Q}{\partial a_{lk}} = -\frac{1}{2} \sum_{i,t,s} \gamma_t^{(is)} \sum_p \frac{\partial}{\partial a_{lk}} \left(\log(\sigma_{y,p}^{2(is)}) + \frac{(y_{t,p}^{(i)} - \mu_{y,p}^{(is)})^2}{\sigma_{y,p}^{2(is)}} \right)$$

where p indexes the components of the observation vector, mean, and variance. Taking the partial derivative and then rearranging terms gives

$$\frac{\partial Q}{\partial a_{lk}} = -\frac{1}{2} \sum_{i,t,s} \gamma_t^{(is)} \sum_p \left[\frac{1}{\sigma_{y,p}^{2(is)}} \frac{\partial \sigma_{y,p}^{2(is)}}{\partial a_{lk}} \left(1 + \frac{(y_{t,p}^{(i)} - \mu_{y,p}^{(is)})^2}{\sigma_{y,p}^{2(is)}} \right) - 2 \frac{(y_{t,p}^{(i)} - \mu_{y,p}^{(is)})}{\sigma_{y,p}^{2(is)}} \frac{\partial \mu_{y,p}^{(is)}}{\partial a_{lk}} \right] \tag{19}$$

The partial derivatives of the mean and variance in (19) can be derived from (11) and (12) as

$$\frac{\partial \mu_{y,p}^{(is)}}{\partial a_{lk}} = \sum_q \frac{\partial f_{p,q}^{(is)}}{\partial a_{lk}} \left(\mu_{x,q}^{(s)} - \mu_{n,q}^{(i)} \right) \tag{20}$$

$$\frac{\partial \sigma_{y,p}^{2(is)}}{\partial a_{lk}} = 2 \sum_q \frac{\partial f_{p,q}^{(is)}}{\partial a_{lk}} \left(f_{p,q}^{(is)} \sigma_{x,q}^{2(s)} - g_{p,q}^{(is)} \sigma_{n,q}^{2(i)} \right) \tag{21}$$

where $f_{p,q}^{(is)}$ and $g_{p,q}^{(is)}$ are the (p,q)th components of $\boldsymbol{F}^{(is)}$ and $\boldsymbol{G}^{(is)}$ respectively and

$$\frac{\partial f_{p,q}^{(is)}}{\partial a_{lk}} = w_{lk}^{(is)} (c_{p,l} d_{l,q}) \tag{22}$$

where $c_{p,l}$ and $d_{l,q}$ are the corresponding components of $\boldsymbol{C}$ and $\boldsymbol{D}$, respectively.

As shown in (11)–(14), the slope parameters a_{lk} are a function of both the static *and* dynamic Gaussian parameters. As such the summation over d in (19) is computed over all feature vector components. For the dynamic means and variances, the partial derivatives are identical to (20) and (21) except with the dynamic components substituted for the static components.

Finally, the complete expression for the gradient can be found by substituting (20), (21), and (22) into (19). We omit it here for space considerations.

3.2. Gradient update for the spline y-intercept parameters

The y-intercept parameters are found simply as

$$\frac{\partial Q}{\partial b_{lk}} = \sum_{i,t,s} \gamma_t^{(is)} \sum_d \frac{(y_{t,d}^{(i)} - \mu_{y,p}^{(is)})}{\sigma_{y,p}^{2(is)}} \frac{\partial \mu_{y,p}^{(is)}}{\partial b_{lk}} \tag{23}$$

where the partial derivative term in (23) can be written as

$$\frac{\partial \mu_{y,p}^{(is)}}{\partial b_{lk}} = w_{lk}^{(is)} c_{p,l} \tag{24}$$

Because we assume the spline weights are constant over the span of the dynamic features, the adapted delta mean $\boldsymbol{\mu}_{\Delta y}^{(is)}$ is independent of the y-intercept parameters in $\boldsymbol{b}^{(is)}$. As a result, the summation over d in (23) only includes the static coefficients.

3.3. Gradient update for the spline variances

The gradient of Q with respect to the spline variance ψ_{lk}^2 can be written as

$$\begin{aligned} \frac{\partial Q}{\partial \psi_{lk}^2} &= -\frac{1}{2} \sum_{i,t,s} \gamma_t^{(is)} \sum_p \frac{\partial}{\partial \psi_{lk}^2} \left(\log(\sigma_{y,p}^{2(is)}) + \frac{(y_{t,p}^{(i)} - \mu_{y,p}^{(is)})^2}{\sigma_{y,p}^{2(is)}} \right) \\ &= -\frac{1}{2} \sum_{i,t,s} \gamma_t^{(is)} \sum_p \frac{\partial \sigma_{y,p}^{2(is)}}{\partial \psi_{lk}^2} \left(\frac{1}{\sigma_{y,p}^{2(is)}} - \frac{(y_{t,p}^{(i)} - \mu_{y,p}^{(is)})^2}{(\sigma_{y,p}^{2(is)})^2} \right) \end{aligned} \tag{25}$$

where the partial derivative term in (25) can be written as

$$\frac{\partial \sigma_{y,p}^{2(is)}}{\partial \psi_{lk}^2} = w_{lk}^{(is)2} c_{p,l}^2 \tag{26}$$

To ensure that the variances remain positive, we will actually optimize $\tilde{\psi}_{lk}^2 = \log(\psi_{lk}^2)$. This simply requires changing the gradient to

$$\frac{\partial Q}{\partial \tilde{\psi}_{lk}^2} = \psi_{lk}^2 \frac{\partial Q}{\partial \psi_{lk}^2} \tag{27}$$

The gradient expressions for the spline variances of the delta and delta-delta coefficients $\{\psi_{\Delta,lk}^2, \psi_{\Delta\Delta,lk}^2\}$ can be similarly derived and have been omitted for space considerations.

4. Experiments

In order to evaluate the proposed method for spline parameter training, a series of experiments were performed using the Aurora 2 corpus [8]. Aurora 2 consists of data degraded with eight types of noise at SNRs between 0 dB and 20 dB. Evaluation is performed using three test sets that contain noise types seen in the training data (Set A), unseen in the training data (Set B), and additive noise plus channel distortion (Set C).

The acoustic models were trained from the clean training set using HTK with the standard "complex back end" Aurora 2 recipe. An HMM with 16 states per digit and 20 Gaussians per state is created for each digit as a whole word. There is a three-state silence model with 36 Gaussians per state and a one state short pause model tied to the middle state of silence. Standard 39-dimensional MFCC features consisting of 13 static, delta, and delta-delta features were computed from power spectral observations and C0 was used instead of log energy. Noise is assumed to be stationary and Gaussian with a diagonal covariance. The baseline word accuracy with no compensation is 62.6%. In all experiments, the noise mean and variance were estimated from the first and last 20 frames and the channel distortion was assumed to be zero ($\boldsymbol{\mu}_h^{(i)} = 0$).

We evaluated VTS adaptation with LSI adaptation using two different methods of training the spline parameters. In both LSI cases, the number of spline segments and the knot locations

Test Set	VTS	LSI - stereo	LSI - ML
Set A	88.14	89.15	91.87
Set B	88.35	89.17	91.01
Set C	88.41	90.19	92.08
Avg	88.28	89.37	91.57

Table 1: Accuracy obtained with model adaptation using VTS, LSI with spline parameters learned from stereo data and LSI with maximum likelihood spline parameter training.

were identical. The splines had 36 segments, which was shown to have good performance in [4]. The knot locations were chosen empirically, with knots more densely placed at SNRs near 0 dB based on the observation that the spline variance changes more quickly in this region.

In the first case, stereo data consisting of the clean and multi-condition training sets of Aurora 2 was used to train a linear spline for each log mel filterbank coefficient using the approach described in Section 2.3. In the second case, the splines were trained from the multi-condition training data and the speech recognizer described above using the Generalized EM framework described in Section 3. For the Generalized EM, gradient-descent was performed using RProp [9]. The spline parameters learned from the stereo data in the first case were used as the initial values for the optimization and 50 iterations of learning were performed.

The results of these experiments are shown in Table 1. As the table shows, LSI adaptation outperforms VTS adaptation using either method of spline parameter training. However, there is a significant reduction in word error rate using splines trained using the integrated EM approach with a relative reduction of word error rate of 28.1% over VTS and a 20.7% over the original LSI algorithm.

Finally, we compared the performance of LSI to a version of VTS that uses the phase-sensitive environmental distortion model that includes the α term as shown in (1). Theoretically, α has a mean of 0 and $-1 \leq \alpha \leq 1$. However, in [10] the value of α was handtuned for optimal performance on the Aurora 2 task and the best performance was obtained with $\alpha = 2.5$. Although this value lacks a well-understood physical interpretation, it represents the best performance on this task using a clean acoustic model trained with maximum likelihood. A comparison of VTS, phase-sensitive VTS with $\alpha = 2.5$ and LSI with integrated spline training is shown in Table 2. The best performance at each SNR is shown in bold. As the table indicates, the LSI slightly outperforms the phase-sensitive VTS on average. However, it is more interesting to note that LSI has the best performance at SNRs of 10 dB or greater, while VTS has the best performance at the lower SNRs. The reasons for these differences in performance are currently under investigation.

5. Conclusions

In this paper, we significantly improved our recently proposed HMM adaptation algorithm called Linear Spline Interpolation. In the original LSI algorithm, the spline parameters were trained using a corpus of stereo training data by minimizing the error between the predicted and actual noisy speech. In this work, we proposed a new method of training the spline parameters using a Generalized EM algorithm that is tightly integrated with the speech recognizer itself. This approach was shown experimentally to be superior to both the original method of spline parameter training and the well-known VTS adaptation algorithm. In the future, we plan to improve the spline training further by incorporating the online noise and channel re-estimation that was performed in [6] into the spline training. We also plan to develop a noise adaptive training for LSI to jointly estimate both the spline parameters and the HMM parameters without clean speech data.

SNR (dB)	VTS ($\alpha = 0$)	VTS ($\alpha = 2.5$)	LSI (ML)
∞	**99.62**	99.61	99.60
20	98.71	98.73	**99.05**
15	97.34	97.87	**98.31**
10	93.66	95.64	**96.19**
5	84.91	**90.21**	90.13
0	66.88	**74.66**	74.58
-5	37.21	**43.32**	40.20
Avg	88.28	91.37	**91.57**

Table 2: Accuracy obtained using standard VTS adaptation, VTS with a handtuned α value, and the proposed LSI approach with maximum likelihood parameter training.

6. References

[1] M. J. F. Gales and S. J. Young, "Robust continuous speech recognition using parallel model combination," *IEEE Trans. on Sp. and Audio Proc.*, vol. 4, pp. 352–359, 1996.

[2] A. Acero, L. Deng, T. Kristjansson, and J. Zhang, "HMM Adaptation Using Vector Taylor Series for Noisy Speech Recognition," in *Proc. of ICSLP*, 2000.

[3] Y. Hu and Q. Huo, "An HMM compensation approach using unscented transformation for noisy speech recognition," in *Proc. ISCSLP*, 2006, pp. 346–357.

[4] K. Kalgaonkar, M. L. Seltzer, and A. Acero, "Noise robust model adaptation using linear spline interpolation," in *Proc. of ASRU*, Trento, Italy, 2009.

[5] L. Deng, J. Droppo, and A. Acero, "Enhancement of log mel power spectra of speech using a phase-sensitive model of the acoustic environment and sequential estimation of the corrupting noise," *IEEE Trans. on Sp. and Audio Proc.*, vol. 12, no. 2, pp. 133–143, March 2004.

[6] M. L. Seltzer, K. Kalgaonkar, and A. Acero, "Acoustic model adaptation via linear spline interpolation for robust speech recognition," in *Proc. of ICASSP*, Dallas, TX, 2010.

[7] J. E. Ertel and E. B. Fowlkes, "Some algorithms for linear spline and piecewise multiple linear regression," *Journal of the Amer. Stat. Assc.*, vol. 71, no. 355, pp. 640–648, Sept. 1976.

[8] H.G. Hirsch and D. Pearce, "The AURORA experimental framework for the performance evaluation of speech recognition systems under noisy conditions," in *Proc. of ISCA ITRW ASR*, Paris, France, September 2000.

[9] M. Riedmiller and H. Braun, "A direct adaptive method for faster backpropagation learning: The RPROP algorithm," in *IEEE Int. Conf. on Neural Networks*, 1993.

[10] J. Li, L. Deng, D. Yu, and A. Acero, "HMM adaptation using a phase-sensitive acoustic distortion model for environment robust speech recognition," in *Proc. of ICASSP*, Las Vegas, NV, 2008.