HIGH-PERFORMANCE HMM ADAPTATION WITH JOINT COMPENSATION OF ADDITIVE AND CONVOLUTIVE DISTORTIONS VIA VECTOR TAYLOR SERIES

Jinyu Li[1], Li Deng, Dong Yu, Yifan Gong, and Alex Acero

Microsoft Corporation, One Microsoft Way, Redmond, WA 98052
jinyuli@ece.gatech.edu, {deng;dongyu;ygong;alexac}@microsoft.com

ABSTRACT

In this paper, we present our recent development of a model-domain environment-robust adaptation algorithm, which demonstrates high performance in the standard Aurora 2 speech recognition task. The algorithm consists of two main steps. First, the noise and channel parameters are estimated using a nonlinear environment distortion model in the cepstral domain, the speech recognizer's "feedback" information, and the Vector-Taylor-Series (VTS) linearization technique collectively. Second, the estimated noise and channel parameters are used to adapt the static and dynamic portions of the HMM means and variances. This two-step algorithm enables Joint compensation of both Additive and Convolutive distortions (JAC).

In the experimental evaluation using the standard Aurora 2 task, the proposed JAC/VTS algorithm achieves 91.11% accuracy using the clean-trained simple HMM backend as the baseline system for the model adaptation. This represents high recognition performance on this task without discriminative training of the HMM system. Detailed analysis on the experimental results shows that adaptation of the dynamic portion of the HMM mean and variance parameters is critical to the success of our algorithm.

Index Terms— vector Taylor series, joint compensation, additive and convolutive distortions, robust ASR

1. INTRODUCTION

Environment robustness in speech recognition remains an outstanding and difficult problem despite many years of research and investment [1]. The difficulty arises due to many possible types of distortions, including additive and convolutive distortions and their mixes, which are not easy to predict accurately during recognizers' development. As a result, the speech recognizer trained using clean speech often degrades its performance significantly when used under noisy environments if no compensation is applied. Different methodologies have been proposed in the past for environment robustness in speech recognition over the past two decades. There are two main classes of approaches. In the first class, the distorted speech features are enhanced with advanced signal processing methods; Examples include the ETSI advanced front end (AFE) [2] and stereo-based piecewise linear compensation for environments (SPLICE) [3]. The other class of techniques operates on the model domain to adapt or adjust the model parameters so that the system becomes better matched to the distorted environment; Examples include parallel model combination (PMC) [4] and joint compensation of additive and convolutive distortions (JAC) [5]. The model-based techniques have shown better performance than the feature-based approaches [5][6].

With the expectation-maximization (EM) method [7], JAC [5] directly estimates the noise and channel distortion parameters in the log-spectral domain, adjusts the acoustic HMM parameters in the same log-spectral domain, and then converts the parameters to the cepstral domain. Note, however, that no strategy for HMM variance adaptation was given in [5] and the techniques for estimating the distortion parameters involve a number of unnecessary approximations.

A similar JAC model-adaptation method was proposed in [8] where both the static mean and variance parameters in the cepstral domain are adjusted using the vector Taylor series (VTS) expansion techniques. In that work, however, noise was estimated on the frame-by-frame basis. This process is complex and computationally costly and the resulting estimate may not be reliable. (The work in [9] indicates roughly N times of computation using frame-by-frame estimation compared with the batch noise estimation which is reported in this paper, where N is the number of frames in the utterance.) Furthermore, no adaptation was made for the delta or dynamic portions of HMM parameters, which is known to be important for high performance robust speech recognition [6].

The JAC algorithm proposed in [10] directly used VTS to estimate the noise and channel mean but adapted the feature instead of the model. In that work, no delta or dynamic portions of the features were compensated either. The work in [6], on the other hand, proposed a framework to adjust both the static and delta/dynamic portions of the HMM parameters given the known noise and channel parameters. However, while it was mentioned that the iterative EM algorithm can be used for the estimation of the noise and channel parameters, no actual algorithm was developed and reported in [6]. Further, the recent study on uncertainty decoding [11] also intended to jointly compensate for the additive and convolutive distortions. However, the proposed technique does not take advantage of the power of the well established parameter-free, nonlinear distortion model for the effects of noise and channel. Instead, it introduced a large number of trainable parameters and turned the nonlinear estimation problem into a linear one. Finally, the well-known adaptation method of maximum likelihood linear regression (MLLR) [12], as well as its counterpart in the feature space (fMLLR) [13] (also known as constrained MLLR [14]), was used to adapt the clean-trained model to the distorted acoustic environments. In order to

[1] This work was carried out at Microsoft Research, Redmond while the first author worked as a student intern.

achieve sufficient performance, MLLR often requires significantly more transformation matrices than one, and this results in the special requirement for the amount of adaptation data [15][16]. Hence, the methods based on MLLR may not be suitable for online adaptation with only one utterance available. From the results reported in [15][16], even with a large number of adaptation utterances, the performance is still significantly lower than what we will report in this paper using the JAC/VTS approach.

The study presented in this paper can be viewed as an extension to the work of [5], [6], [8], and [10] by carrying out JAC on both static and dynamic MFCCs with noise and channel parameters being rigorously and systematically estimated on an utterance-by-utterance basis using VTS. In particular, moving away from noise estimation on a frame-by-frame basis significantly reduced the cost of computation.

The rest of the paper is organized as follows. In Section 2, we present our new JAC/VTS algorithm and its implementation steps. We also compare the algorithm with a number of related previous algorithms. Experimental evaluation of the algorithm is provided in Section 3, where the effectiveness of adapting the HMM variances (both the static and dynamic portions) is also demonstrated. We show that our new algorithm can achieve higher than 91% word recognition accuracy averaged over all distortion conditions on the Aurora2 task with the standard simple back-end clean-trained model and standard MFCCs. We summarize our study and draw conclusions in Section 4.

2. JAC/VTS ADAPTATION ALGORITHM

In this section, we first derive the adaptation formulas for the HMM means and variances in the MFCC (both static and dynamic) domain using VTS approximation assuming that the estimates of the additive and convolutive parameters are known. We then derive the algorithm which jointly estimates the additive and convolutive distortion parameters based on VTS approximation. A summary description follows on the implementation steps of the entire algorithm which were used in our experiments. Finally, the proposed method and other JAC-family methods are compared and discussed.

2.1 Algorithm for HMM Adaptation Using Joint Noise and Channel Estimates

Figure 1 shows a model for degraded speech with both noise (additive) and channel (convolutive) distortions. The observed distorted speech signal *y[m]* is generated from clean speech signal *x[m]* with noise *n[m]* and the channel *h[m]* according to

$$y[m] = x[m]*h[m] + n[m]. \quad (1)$$

With discrete Fourier transformation, the following equivalent relations can be established in the spectral domain and the log-spectral domain by ignoring the phase, respectively:

$$|Y[k]| = |X[k]|\,|H[k]| + |N[k]| \quad (2)$$

$$\log|Y[k]| = \log\left[|X[k]|\,|H[k]| + |N[k]|\right] \quad (3)$$

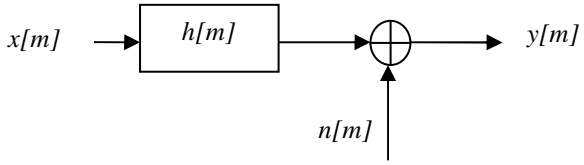

Figure 1: A model for environment distortion

Rearranging and mapping Eq. (3) to log-filter-bank domain, and then multiplying the non-square discrete cosine transform (DCT) matrix to both sides yield the following well-established nonlinear distortion model [10]:

$$y = x + h + C\log(1+\exp(C^{-1}(n-x-h))), \quad (4)$$

where C^{-1} is the (pseudo) inverse DCT matrix. *y*, *x*, *n* and *h* are the vector-valued distorted speech, clean speech, noise, and channel, respectively, all in the MFCC domain.

Using the VTS approximation (as was used in [6]), we have

$$\begin{aligned}\mu_y &\approx \mu_x + \mu_h + C\log(1+\exp(C^{-1}(\mu_n-\mu_x-\mu_h))) \\ &= \mu_x + \mu_h + g(\mu_x,\mu_h,\mu_n)\end{aligned} \quad (5)$$

where

$$g(\mu_x,\mu_h,\mu_n) = C\log(1+\exp(C^{-1}(\mu_n-\mu_x-\mu_h))) \quad (6)$$

and μ_y, μ_x, μ_h, and μ_n are the mean vectors of the cepstral signal *y*, *x*, *h*, and *n*, respectively.

Using (5), we compute the following derivatives,

$$\frac{\partial\mu_y}{\partial\mu_h} = C \bullet diag\left(\frac{1}{1+\exp(C^{-1}(\mu_n-\mu_x-\mu_h))}\right)\bullet C^{-1} = G, \quad (7)$$

$$\frac{\partial\mu_y}{\partial\mu_n} = I - G, \quad (8)$$

where *diag(.)* stands for the diagonal matrix with its diagonal component value equal to the value of the vector in the argument. For the given noise mean vector μ_n and channel mean vector μ_h, the value of *G(.)* depends on the mean vector μ_x. Specifically, for the *k*-th Gaussian in the *j*-th state, the element of *G(.)* matrix is:

$$G(j,k) = C \bullet diag\left(\frac{1}{1+\exp(C^{-1}(\mu_n-\mu_{x,jk}-\mu_h))}\right)\bullet C^{-1}. \quad (9)$$

Then, the first-order VTS is applied to obtain the relationship between the Gaussian mean vectors (the *k*-th Gaussian in the *j*-th state) in the adapted HMM for the degraded speech and in the original clean-speech HMM:

$$\begin{aligned}\mu_{y,jk} &\approx \mu_{x,jk} + \mu_h + g(\mu_{x,jk},\mu_h,\mu_n) \\ &= \mu_{x,jk} + \mu_{h,0} + g(\mu_{x,jk},\mu_{h,0},\mu_{n,0}) \\ &\quad + G(j,k)(\mu_h-\mu_{h,0}) + (I-G(j,k))(\mu_n-\mu_{n,0})\end{aligned}, \quad (10)$$

where $\mu_{n,0}$ and $\mu_{h,0}$ are the VTS expansion points for μ_n and μ_h, respectively, and (10) is applied only to the static portion of the MFCC vector.

The covariance matrix $\Sigma_{y,jk}$ in the adapted HMM can be estimated as a weighted summation of $\Sigma_{x,jk}$, the covariance matrix of the clean HMM, and Σ_n, the covariance matrix of noise, i.e.,

$$\Sigma_{y,jk} \approx G(j,k)\Sigma_{x,jk}G(j,k)^T + (I-G(j,k))\Sigma_n(I-G(j,k))^T. \quad (11)$$

Here, no channel variance is taken into account because we treat the channel as a fixed, deterministic quantity in a given utterance.

For the delta and delta/delta portions of MFCC vectors, the adaptation formulas for the mean vector and covariance matrix are:

$$\mu_{\Delta y,jk} \approx G(j,k)\mu_{\Delta x,jk}, \quad (12)$$

$$\mu_{\Delta\Delta y,jk} \approx G(j,k)\mu_{\Delta\Delta x,jk}, \quad (13)$$

$$\Sigma_{\Delta y,jk} \approx G(j,k)\Sigma_{\Delta x,jk}G(j,k)^T + (I-G(j,k))\Sigma_{\Delta n}(I-G(j,k))^T, \quad (14)$$

$$\Sigma_{\Delta\Delta y,jk} \approx G(j,k)\Sigma_{\Delta\Delta x,jk}G(j,k)^T + (I-G(j,k))\Sigma_{\Delta\Delta n}(I-G(j,k))^T \quad (15)$$

2.2 Algorithm for Re-estimation of Noise and Channel

EM algorithm is developed as part of the overall JAC/VTS algorithm to estimate the noise and channel mean vectors using the VTS approximation. Let Ω_s denote the set of states, Ω_m denote the set of Gaussians in a state, θ_t denote the state index, and ε_t denote the Gaussian index at time frame *t*. λ and $\overline{\lambda}$ are the new and old parameter sets for the mean of noise and channel. The auxiliary Q function for an utterance is

$$Q(\lambda|\overline{\lambda}) = \sum_t \sum_{j\in\Omega_s} \sum_{k\in\Omega_m} p(\theta_t = j, \varepsilon_t = k | Y, \overline{\lambda}) \cdot \log p(y_t | \theta_t = j, \varepsilon_t = k, \lambda) \quad (16)$$

where $p(y_t|\theta_t = j, \varepsilon_t = k, \lambda) \sim N(y_t; \mu_{y,jk}, \Sigma_{y,jk})$, is Gaussian with mean vector $\mu_{y,jk}$ and covariance matrix $\Sigma_{y,jk}$.

To simplify the formula, in the remainder of this section we use $\gamma_t(j,k)$ to denote the posterior probability for the *k*-th Gaussian in the *j*-th state of the HMM, i.e.,

$$\gamma_t(j,k) = p(\theta_t = j, \varepsilon_t = k | Y, \overline{\lambda}). \quad (17)$$

To maximize the auxiliary function in the M-step of the EM algorithm, we take derivative of Q with respect to μ_n and μ_h, and set the derivatives to zero to obtain

$$\sum_t \sum_{j\in\Omega_s} \sum_{k\in\Omega_m} \gamma_t(j,k)(I-G(j,k))^T \Sigma_{y,jk}^{-1}[y_t - \mu_{y,jk}] = 0, \quad (18)$$

$$\sum_t \sum_{j\in\Omega_s} \sum_{k\in\Omega_m} \gamma_t(j,k) G(j,k)^T \Sigma_{y,jk}^{-1}[y_t - \mu_{y,jk}] = 0. \quad (19)$$

After substituting the VTS approximation (10) into (18) with $\mu_h = \mu_{h,0}$, the noise mean vector μ_n can be solved, given its old estimate, as

$$\mu_n = \mu_{n,0} + \left\{\sum_t \sum_{j\in\Omega_s} \sum_{k\in\Omega_m} \gamma_t(j,k)(I-G(j,k))^T \Sigma_{y,jk}^{-1}(I-G(j,k))\right\}^{-1} \bullet \left\{\sum_t \sum_{j\in\Omega_s} \sum_{k\in\Omega_m} \gamma_t(j,k)(I-G(j,k))^T \Sigma_{y,jk}^{-1}[y_t - \mu_{x,jk} - \mu_{h,0} - g(\mu_{x,jk}, \mu_{h,0}, \mu_{n,0})]\right\} \quad (20)$$

Similarly, by substituting (10) into Eq. (19) with $\mu_n = \mu_{n,0}$, the channel mean vector is estimated as

$$\mu_h = \mu_{h,0} + \left\{\sum_t \sum_{j\in\Omega_s} \sum_{k\in\Omega_m} \gamma_t(j,k) G(j,k)^T \Sigma_{y,jk}^{-1} G(j,k)\right\}^{-1} \bullet \left\{\sum_t \sum_{j\in\Omega_s} \sum_{k\in\Omega_m} \gamma_t(j,k) G(j,k)^T \Sigma_{y,jk}^{-1}[y_t - \mu_{x,jk} - \mu_{h,0} - g(\mu_{x,jk}, \mu_{h,0}, \mu_{n,0})]\right\} \quad (21)$$

Eqs. (20) and (21) constitute each iteration of the EM algorithm.

In this study, the variance of noise has not been re-estimated as part of the EM algorithm and will be considered in our future work.

2.3 Algorithm implementation

The implementation steps for the JAC/VTS HMM adaptation algorithm described so far in this section and used in our experiments are summarized and described in the following:

1. Read in a distorted speech utterance;
2. Set the channel mean vector to all zeros;
3. Initialize the noise mean vector and diagonal covariance matrix using the first and last *N* frames (speech-free) from the utterance using sample estimates;
4. Compute the Gaussian-dependent *G(.)* with (9), and update/adapt the HMM parameters with (10)–(15);
5. Decode the utterance with the adapted HMM parameters;
6. Compute posterior probabilities of (17) and then re-estimate the noise and channel mean using (20) and (21);
7. Compute the Gaussian-dependent *G(.)* with (9), and update/adapt the HMM parameters with (10)–(15);
8. Use the final adapted model to obtain the utterance output transcription;
9. Goto step 1.

These steps are for one pass decoding and one-iteration EM re-estimation of noise and channel mean, as we have carried out in our experiments to be presented in the next section. If multiple-pass decoding is desired, there would be a loop between steps 5 and 7 and multiple-iteration EM for noise and channel estimation would be implemented by looping between steps 6 and 7.

2.4 Comparisons with other JAC-family methods

The JAC/VTS algorithm for HMM adaptation presented in this section is the most comprehensive one among a number of other algorithms within the same JAC algorithm family published in the literature. We summarize the differences between our new algorithm, which is a direct extension of the work in [6], and other related algorithms below.

Our algorithm differentiates itself from the JAC method of [5] in the following aspects:

1. Our algorithm directly works on the cepstral or MFCC domain, without the need (as required in [5]) to first adapt the model in the spectral domain and then convert back to the cepstral domain;
2. VTS is used as the basis to derive close-form update/adaptation formulas, instead of using gradient ascent;
3. In noise and channel estimation (see (18) and (19)), the HMM-state-dependent variances and the derivatives of the distorted cepstral mean with respect to noise and channel [i.e., *G*(.) and *I*-*G*(.)] are used. All these have been discarded in the JAC method of [5];
4. Dynamic or delta portions of the HMM mean vector are adapted in a different way;
5. Our algorithm adapts HMM variances within the JAC framework. No variance adaptation is presented in [5];
6. In our algorithm, the channel is initialized as zero for each new utterance, while [5] uses the estimated channel in the previous utterance as the initial value.

While sharing the basic distortion modeling method in the cepstral domain, the algorithm is also different from the method in [8] in two ways:

1. The re-estimation of noise and channel is different. With VTS, our algorithm directly solves noise and channel in a close form for each utterance. In contrast, the technique of [8] estimates the noise parameters for each frame (similar to our earlier work of [9]), which is more complex, more computational intensive, and less reliable than our current batch estimation method;
2. Our algorithm provides the strategy to adapt the dynamic or delta portions of the HMM mean and variance parameters, which are not adapted in [8].

Finally, the work in [10] also used VTS for noise and mean estimation, and it differs from our algorithm in two ways also:

1. The final adjustment in [10] is in the feature space, while our algorithm carries out adaptation in the model space;
2. The VTS algorithm in [10] adjusts only the static portion, not the dynamic portion, of cepstra. In contrast, our algorithm adjusts/adapts both portions. (As will be shown in the experiments, the adaptation of the dynamic portion of the HMM parameters is critical for the success of our algorithm.)

3. EXPERIMENTS AND DISCISSIONS

The effectiveness of the JAC/VTS algorithm presented in Section 2 has been evaluated on the standard Aurora 2 task of recognizing digit strings in noise and channel distorted environments. The clean training set, which consists of 8440 clean utterances, is used to train the baseline maximum likelihood estimation (MLE) HMMs. The test material consists of three sets of distorted utterances. The data in set-A and set-B contain eight different types of additive noise, while set-C contain two different types of noise plus additional channel distortion. Each type of noise is added into a subset of clean speech utterances, with seven different levels of signal to noise ratios (SNRs). This generates seven subgroups of test sets for a specified noise type, with clean, 20db, 15db, 10db, 5db, 0db, and -5db SNRs. The baseline experiment setup follows the standard script provided by ETSI [17], including the simple "backend" of HMMs trained using the HTK toolkit.

In the simple backend provided by [17], there are 11 whole-digit HMMs, one for each of the 11 English digits, including the word "oh". Each HMM has 16 states, with simple left-to-right structure and no skips over states. Each state is modeled by a Gaussian mixture model (GMM) with 3 Gaussians. All HMM's covariance matrices are diagonal. In addition, there are one "sil" and one "sp" model. The "sil" model consists of 3 states, and each state is modeled by a GMM with 6 Gaussians. The "sp" model has only one state and is tied to the middle state of the "sil" model.

The features are 13-dimension MFCCs, appended by their first- and second-order time derivatives. The cepstral coefficient of order 0 is used instead of the log energy in the original script. (This gives a slightly worse baseline of 58.70% Acc than the standard ETSI baseline for clean-trained simple backend model (60.06% Acc)).

The new JAC/VTS algorithm presented in this paper is then used to adapt the above MLE HMMs utterance by utterance for the entire test set (Sets-A, B, and C). The detailed implementation steps described in Section 2.3 are used in the experiments. We use the first and last N=20 frames from each utterance for initializing the noise means and variances. Only one-pass processing is used in the reported experiments.

To examine the effects of individual contributions of HMM adaptation in the overall JAC/VTS algorithm, we conducted experiments by adapting an increasingly large parameter sets in the HMMs. As shown in Table 1, when only the HMMs' static mean vectors are adapted (using Eq. (10)), the average accuracy is improved from the baseline (no adaptation) of 58.70% to 73.34%. When the delta portion of the mean vectors is also adapted (using Eq.(12)), the accuracy further improves to 79.60%. Adding adaptation of the acceleration (delta-delta) portion of the mean vectors (using Eq. (13)) gives even higher accuracy of 84.81%. This shows that the adjustment of dynamic portions of HMM mean parameters is highly effective in improving the recognizer's performance.

The effects of adapting various portions of the HMM variances are shown from Row-6 to Row-8 in Table 1. Adapting the static portion of the HMM variances (using Eq. (11)) improves the recognition accuracy to as high as 89.55%, which is further increased to 91.11% after adapting the delta portion of the HMM variances (using Eq. (14)). However, with Eq.(15), the acceleration portion of variance adaptation drops the recognition accuracy slightly to 89.84%. This may be attributed to the empirical nature and poor approximation underlying the adaptation formula of Eq.(15). At present, we have not developed a rigorous distortion modeling framework for the dynamic portion of the HMM parameters, unlike the static portion for which the modeling framework of Eq. (4) has been firmly established. This is one of our future research items.

Table 1: Recognition accuracy of the baseline (clean-trained simple backend HMM system with no adaptation) and the several adaptive HMM systems. Different rows show the accuracy obtained using the JAC/VTS algorithm to adapt different subsets of the HMM parameters. New adapted HMM parameters are gradually added to examine the detailed effects of the algorithm. Recognition results from the standard Aurora-2 test sets (A, B, C) are used in computing the accuracy.

Baseline & Adapted HMM Systems	Recognition Accuracy
Baseline (MLE)	58.70%
JAC adapting static mean	73.34%
+ JAC adapting delta mean	79.60%
+ JAC adapting acceleration mean	84.81%
+ JAC adapting static variance	89.55%
+ JAC adapting delta variance	91.11%
+ JAC adapting acceleration variance	89.84%

Table 2 lists detailed test results for clean-trained simple backend HMM system after the JAC/VTS adaptation on all static, delta, and acceleration portions of the HMM mean vectors, and on the static and delta portions of the HMM variances. Because the standard evaluation is on the SNRs from 0db to 20db, we do not list the performance for clean and -5db conditions.

Examining the results of Table 2 in detail, we see that the individual recognition accuracy for 20db, 15db, 10db, 5db, and 0db SNRs are 98.36%, 97.52%, 95.49%, 89.94%, and 74.26%,

respectively. It is clear that the performance degrades quickly for low SNRs despite the application of HMM adaptation. This is likely due to the unsupervised nature of our current JAC/VTS algorithm. This makes the effectiveness of the algorithm heavily dependent on the model posterior probabilities of Eq. (17). Under low-SNR conditions, the situation is much worse since the relatively low recognition accuracy forbids utterance decoding from providing correct transcription. Consequently, the estimates of noise and channel under low-SNR conditions tend to be less reliable, resulting lower adaptation effectiveness. Hence, how to obtain and exploit more reliable information for adaptation in low-SNR is a challenge for the future enhancement of our current JAC/VTS algorithm.

It is interesting to compare the proposed JAC/VTS with other adaptation methods on the Aurora2 task. In [18], the JAC update formulas for static mean and variance parameters proposed in [8] and the update formulas for dynamic mean parameters in [6] are used to adapt clean-trained complex backend model (with much higher number of mixture components than the simple-backend model), the accuracy measure reaches only 87.74%. This again demonstrates the advantage of our newly developed JAC/VTS method.

In [15], two schemes of MLLR are used to adapt models with the adaptation utterances selected from test sets A and B. The adapted model is tested on test sets A and B; no result is reported for test set C. Even with as many as 300 adaptation utterances, the average Acc of set A is only 80.95% for MLLR scheme 1, and 78.72% for MLLR scheme 2. And the average Acc of set B is 81.40% for MLLR scheme 1, and 82.12% for MLLR scheme 2. All of these accuracy measures are far below those (around 90%) obtained by our method.

In [16], fMLLR and its projection variant (fMLLR-P) [19] are used to adapt the acoustic features. The adaptation policy is to accumulate sufficient statistics for the test data of each speaker, which requires more adaptation utterances. However, the adaptation result is far from satisfactory. For fMLLR, the accuracy measures of sets A, B, and C are 71.8%, 75.0%, and 71.4%, respectively. For fMLLR-P, the corresponding measures are 71.5%, 74.7%, and 71.1%, respectively.

By comparing the results obtained from MLLR [15] and fMLLR [16], the advantage of JAC/VTS becomes clear. JAC/VTS only takes the current utterance for unsupervised adaptation and achieves excellent adaptation results. The success of JAC/VTS is attributed to its powerful physical environment distortion modeling. As a result, JAC/VTS only needs to estimate the noise and channel parameters for each utterance. This parsimony is important since the statistics from that utterance alone is already sufficient for the estimation (this is not the same for other methods such as MLLR). The estimated noise and channel parameters then allow for "nonlinear" adaptation for all parameters in all HMMs. Such nonlinear adaption is apparently more powerful than "linear" adaptation as in the common methods of MLLR and fMLLR.

There may be a concern that based on the results for the low-SNR cases in Table 2, the proposed method may not be effective due to the unsupervised nature of the JAC/VTS adaptation. In fact, poor results for low-SNR cases are common for any type of unsupervised adaptation, including MLLR or fMLLR, because under these difficult acoustic conditions the adaptation cannot obtain reliable posterior probabilities and transcriptions. However, as reported in [15], even with supervised adaptation and with as many as 300 adaptation utterances, MLLR still achieves worse results at low SNR than those shown in Table 2 with JAC/VTS. (For example, for MLLR scheme 1, the average Accs for set A under the 20, 15, 10, 5, and 0 db SNR conditions are 96.2%, 93.6%, 87.0%, 72.8% and 39.0%, respectively, all lower than the corresponding column in Table 2.)

To explore possible upper bounds of our adaptation method, we designed and conducted the following "diagnostic" experiment (limited to only the 5db car noise condition for test set-A due to the high computational cost). For each test utterance, we extracted its noise (available from the Aurora2 database), then we added the same noise, scaled in its magnitude so that the SNR for each utterance stays at 5 db, into the entire clean training set in Aurora2. Using the noise-added training set, we retrain the acoustic models. The retrained models would be considered to have perfectly matched the current test utterance under this specific noise condition, and are used to decode the current test utterance. (Due to the high computational cost, we only conducted this experiment for 60 utterances (i.e., retrained 60 sets of acoustic models, one for each test-set utterance). The average Acc of these 60 utterances is 91%, which is surprisingly low, and even with a number of approximations made in JAC/VTS, its performance is so high that it matches the "ideal" case. We conjecture two possible reasons for this. First, while the re-training process matches the noise condition of the test utterance, the acoustic model is still speaker independent. JAC/VTS adapts the model with both noise and channel estimation. From the algorithmic view, the channel and speaker characteristic are not distinguished. Therefore, JAC/VTS adapts the current model to both the speaker and acoustic environment, giving its high performance. Second, the scaling of the added noise is carried out in a crude, utterance-by-utterance manner to match the 5 dB target SNR. If the scaling could be done in a more precise, segment-by-segment fashion, the accuracy in the "diagnostic" experiment would be higher.

Table 2: Detailed recognition accuracy results with clean-trained simple backend HMMs using VTS-based JAC on the standard Aurora2 database. MFCCs are used as the acoustic features.

Clean Training - Results														
	A					B					C			
	Subway	Babble	Car	Exhibition	Average	Restaurant	Street	Airport	Station	Average	Subway M	Street M	Average	**Average**
20 dB	98.37	98.1	98.87	98.15	98.37	97.94	98.07	98.6	98.8	98.35	98.53	98.13	98.33	**98.36**
15 dB	97.42	97.13	98.12	97.41	97.52	96.87	97.34	97.64	97.93	97.45	97.61	97.7	97.66	**97.52**
10 dB	95.43	95.34	96.24	94.45	95.37	94.9	94.83	96.21	96.85	95.70	95.46	95.22	95.34	**95.49**
5 dB	90.7	89.12	91.11	88.18	89.78	89.59	89.18	91.74	90.9	90.35	89.65	89.27	89.46	**89.94**
0 dB	78.05	69.04	74.5	73.96	73.89	71.38	73.76	79.18	74.64	74.74	76.94	71.1	74.02	**74.26**
Average	91.99	89.75	91.77	90.43	**90.98**	90.14	90.64	92.67	91.82	**91.32**	91.64	90.28	**90.96**	**91.11**

4. CONCLUSION

In this paper, we have presented our recent development of the JAC/VTS algorithm for HMM adaptation and demonstrated its effectiveness in the standard Aurora 2 environment-robust speech recognition task. The algorithm consists of two main steps. First, the noise and channel parameters are estimated using a nonlinear environment distortion model in the cepstral domain, the speech recognizer's "feedback" information, and the vector-Taylor-series (VTS) linearization technique collectively. Second, the estimated noise and channel parameters are used to adapt the static and dynamic portions of the HMM means and variances. This two-step algorithm enables joint compensation of both additive and convolutive distortions (JAC).

In the experimental evaluation using the standard Aurora 2 task, the proposed JAC/VTS algorithm has achieved 91.11% accuracy using the clean-trained simple HMM backend as the baseline system for model adaptation. This represents high performance in this task without discriminative training of the HMM system. Detailed analysis on the experimental results has shown that the adaptation of the dynamic portion of the HMM mean and variance parameters is critical to the success of our algorithm.

Several research issues will be addressed in the future to further increase the effectiveness of the algorithm presented in this paper. First, only the mean vectors of noise and channel are re-estimated in a principled way using all available information including the linearized environment distortion model and speech recognizer's "feedback" in the current algorithm implementation. The variance of noise is only estimated empirically from the start and end frames in an utterance. Further improvement is expected with more principled estimation of noise variance. Second, the current treatment of the dynamic portion of the adapted HMM parameters is highly heuristic. More principled adaptation strategies on these parameters are needed. Third, the success of our JAC/VTS algorithm relies on accurate and reliable recognizer's "feedback" information represented by the posterior probabilities. Under the condition of low-SNR, such "feedback" information tends to be unreliable, resulting in poor estimates of noise and channel parameters. Overcoming this difficulty will be a significant boost to the current JAC/VTS algorithm under low-SNR conditions. Fourth, we seek to further improve the quality of the speech distortion model as represented in Eq. (4) by incorporating the missing term for the phase asynchrony between the clean speech and the mixing noise, and to developed the enhanced HMM adaptation algorithm.

5. ACKNOWLEDGEMENT

We would like to thank Dr. Jasha Droppo at Microsoft research for the help in setting up the experimental platform.

6. REFERENCES

[1] A. Peinado and J. Segura, *Speech Recognition over Digital Channels --- Robustness and Standards*. John Wiley and Sons Ltd (West Sussex, England), 2006.

[2] D. Macho, L. Mauuary, B. Noe, Y. M. Cheng, D. Ealey, D. Jouvet, H. Kelleher, D. Pearce, and F. Saadoun, "Evaluation of a noise-robust DSR front-end on Aurora databases," in *Proc. ICSLP*, pp. 17–20, 2002.

[3] L. Deng, A. Acero, M. Plumpe, and X. Huang, "Large vocabulary speech recognition under adverse acoustic environments," *Proc. ICSLP*, Vol.3, pp.806-809, 2000.

[4] M. J. F. Gales and S. Young, "An improved approach to the hidden Markov model decomposition of speech and noise," *Proc. ICASSP*, Vol. I, pp. 233–236, 1992.

[5] Y. Gong, "A method of joint compensation of additive and convolutive distortions for speaker-independent speech recognition," *IEEE Trans. Speech and Audio Proc.*, Vol. 13, No. 5, pp. 975-983, 2005.

[6] A. Acero, L. Deng, T. Kristjansson, and J. Zhang, "HMM adaptation using vector Taylor series for noisy speech recognition," *Proc. ICSLP*, Vol.3, pp. 869-872, 2000.

[7] A. Dempster, N. Laird, and D. Rubin, "Maximum likelihood from incomplete data via the EM algorithm," *Journal of the Royal Statistical Society*, Series B, 39(1):1–38, 1977.

[8] D. Y. Kim, C. K. Un, and N. S. Kim, "Speech recognition in noisy environments using first order vector Taylor series," *Speech Communication*, Vol. 24, pp. 39-49, 1998.

[9] L. Deng, J. Droppo, and A. Acero, "Recursive estimation of nonstationary noise using iterative stochastic approximation for robust speech recognition," *IEEE Trans. Speech and Audio Proc.*, Vol.11, No.6, pp. 568-580, 2003.

[10] P. Moreno, *Speech Recognition in Noisy Environments*. PhD. Thesis, Carnegie Mellon University, 1996.

[11] H. Liao and M. J. F. Gales, "Adaptive training with joint uncertainty decoding for robust recognition of noisy data," *Proc. ICASSP*, Vol. IV, pp. 389-392, 2007.

[12] C. J. Leggetter and P. C. Woodland, "Maximum likelihood linear regression for speaker adaptation of continuous density HMMs," *Comput., Speech, Lang,*, Vol. 9, No. 2, pp. 171–185, 1995.

[13] Y. Li, H. Erdogan, Y. Gao, and E. Marcheret, "Incremental online feature space MLLR adaptation for telephony speech recognition," *Proc Interspeech*, pp. 1417-1420, 2002.

[14] M.J.F. Gales, "Maximum likelihood linear transformations for HMM-based speech recognition," *Computer Speech and Language*, Vol. 12, pp. 75-98, 1998.

[15] X. Cui and A. Alwan, "Noise robust speech recognition using feature compensation based on polynomial regression of utterance SNR," *IEEE Trans. Speech and Audio Proc.*, Vol. 13, No. 6, pp. 1161-1172, 2005.

[16] G. Saon, H. Huerta, and E. E. Jan, "Robust digit recognition in noisy environments: the IBM Aurora 2 system," *Proc. Interspeech*, pp. 629-632, 2001.

[17] H. G. Hirsch and D. Pearce, "The Aurora experimental framework for the performance evaluation of speech recognition systems under noisy conditions," *Proc. ISCA ITRW ASR*, 2000.

[18] Y. Hu and Q. Huo, "Irrelevant variability normalization based HMM training using VTS approximation of an explicit model of environmental distortions," *Proc. Interspeech*, pp. 1042-1045, 2007.

[19] G. Saon, G. Zweig, and M. Padmanabhan, "Linear feature space projections for speaker adaptation," *Proc. ICASSP*, vol., pp.325– 328, 2001.