# Noise adaptive training for subspace Gaussian mixture models

# Noise adaptive training for subspace Gaussian mixture models

*Liang Lu, Arnab Ghoshal, and Steve Renals*

Centre for Speech Technology Research, University of Edinburgh, Edinburgh, UK
{liang.lu, a.ghoshal, s.renals}@ed.ac.uk

## Abstract

Noise adaptive training (NAT) is an effective approach to normalise the environmental distortions in the training data. This paper investigates the model-based NAT scheme using joint uncertainty decoding (JUD) for subspace Gaussian mixture models (SGMMs). A typical SGMM acoustic model has much larger number of surface Gaussian components, which makes it computationally infeasible to compensate each Gaussian explicitly. JUD tackles the problem by sharing the compensation parameters among the Gaussians and hence reduces the computational and memory demands. For noise adaptive training, JUD is reformulated into a generative model, which leads to an efficient expectation-maximisation (EM) based algorithm to update the SGMM acoustic model parameters. We evaluated the SGMMs with NAT on the Aurora 4 database, and obtained higher recognition accuracy compared to systems without adaptive training.

**Index Terms**: adaptive training, noise robustness, joint uncertainty decoding, subspace Gaussian mixture models

## 1. Introduction

Modern state-of-the-art automatic speech recognition (ASR) systems are normally trained on large amount of, but heterogeneous acoustic data which are recorded from different speakers and in various environmental conditions. This induces nuisance variability in the acoustic data which is irrelevant to the task of speech recognition, and hence reduces the recognition accuracy of an ASR system. Adaptive training is an effective technique to normalise such kind of variability in the canonical acoustic model. A typical example is speaker adaptive training (SAT) [1], in which speaker-dependent transformations are introduced during the model training process to account for the speaker related variability. Similar adaptive training scheme has also been proposed to normalise the variability induced by the environmental noise, which is referred as noise adaptive training (NAT) [2, 3], including some variants such as irrelevant variability normalisation (IVN) [4] and joint adaptive training (JAT) [5].

The application of NAT depends on a particular choice of the noise compensation algorithms that may be used in either feature- or model-domain. Recent work has proposed numerous such kind of algorithms for noise robust ASR with both strengths and weaknesses. For instance, vector Taylor series (VTS) [6] and model-based joint uncertainty decoding (JUD) [7] rely on the mismatch function that models the relationship between clean and noise corrupted speech. Using the mismatch function has the advantage that the amount of adaptation data can be small, which is suitable for rapid adaptation. But it also has the strict requirement of acoustic features to be used, which limits its application domains. SPLICE [2, 8] and front-end JUD [9] get rid of the constraint of the mismatch function by learning the mapping between clean and noisy speech from the stereo training data. However, such kind of data is normally hard to be obtained, and it may not generalise well to unseen noise conditions. Noisy constrained maximum likelihood linear regression (NCMLLR) [10], which is a purely data-driven method, is more flexible from this perspective. It relies on neither the mismatch function as VTS or JUD, nor the stereo training data as SPLICE, but estimates the noise compensation transformations by maximum likelihood criterion for each homogeneous block of acoustic data. However, it requires larger amount of training data to achieve good performance, and hence not suitable for rapid adaptation.

While most of the research on noise robustness are based on the GMM-based acoustic models, we have previously shown in [12, 13] that state-of-the-art performance can be achieved using the recently proposed subspace Gaussian mixture models (SGMMs) [11]. Using a compact model representation, an SGMM acoustic model usually has much larger number of surface Gaussians. For noise compensation, JUD was used to compromise between the accuracy and computational cost [13]. In this paper, we study the application of NAT to the SGMMs based on JUD transformations. The adaptive training algorithm is derived from the generative nature of the JUD transformation as in [10], which leads to an efficient EM-based algorithm to update the acoustic model parameters. We experimented the NAT algorithm on the Aurora 4 dataset and demonstrated the effectiveness of the propose approach.

## 2. Joint uncertainty decoding

In joint uncertainty decoding (JUD) [9], the likelihood of a noisy speech observation $\mathbf{y}_t$ at time frame $t$ given the model component $m$ is obtained by marginalising out the latent clean speech variable $\mathbf{x}_t$ as

$$p(\mathbf{y}_t \mid m) = \int p(\mathbf{x}_t, \mathbf{y}_t \mid m) d\mathbf{x}_t \tag{1}$$

$$\approx \int p(\mathbf{y}_t \mid \mathbf{x}_t, r) p(\mathbf{x}_t \mid m) d\mathbf{x}_t \tag{2}$$

where equation (2) is obtained by using the approximation of $p(\mathbf{y}_t|\mathbf{x}_t, m) \approx p(\mathbf{y}_t|\mathbf{x}_t, r)$, and $r$ denotes the regression class that component $m$ belongs to. By using smaller number of regression classes, this approximation can significantly reduce the computational cost at the expense of slightly worse recognition accuracy [7].

By assuming the joint distribution of $\mathbf{x}_t$ and $\mathbf{y}_t$ being Gaussian, the analytical form of the marginal likelihood is given as

$$p(\mathbf{y}_t \mid m) \approx |\mathbf{A}^{(r)}| \mathcal{N}\left(\mathbf{A}^{(r)}\mathbf{y}_t + \mathbf{b}^{(r)}; \boldsymbol{\mu}_m, \boldsymbol{\Sigma}_m + \boldsymbol{\Sigma}_b^{(r)}\right). \tag{3}$$

Here, $\mathcal{T} = \left[\left(\mathbf{A}^{(r)}, \mathbf{b}^{(r)}, \boldsymbol{\Sigma}_b^{(r)}\right), r = 1, \ldots, R\right]$ are referred as the JUD transformation parameters, which is computed for each regression class, and $R$ is the total number of regression classes. These parameters were originally estimated using the stereo training data as SPLICE [9], but it was later replaced by using the VTS style scheme [14, 7], which uses the similar mismatch function as that is used in the standard VTS based noise compensation [6, 15]. Following [16, 17], we used the extended mismatch function which introduces the phase factor to capture the correlations between the noise and clean speech when applying JUD to an SGMM acoustic model [12, 13], which can be expressed as

$$\mathbf{y}_t^{(s)} = \mathbf{x}_t^{(s)} + \mathbf{h}_t + \mathbf{C}\log\Big[\mathbf{1} + \exp\Big(\mathbf{C}^{-1}(\mathbf{n}_t - \mathbf{x}_t^{(s)} - \mathbf{h}_t)\Big) + 2\boldsymbol{\alpha} \bullet \exp\Big(\mathbf{C}^{-1}(\mathbf{n}_t - \mathbf{x}_t^{(s)} - \mathbf{h}_t)/2\Big)\Big], \tag{4}$$

where the subscript $^{(s)}$ corresponds to the static coefficients, $\mathbf{1}$ is the unit vector, $\log(\cdot)$, $\exp(\cdot)$ and $\bullet$ denote the element-wise logarithm, exponentiation and multiplication. $\mathbf{n}_t$ and $\mathbf{h}_t$ are static additive and convolutional noise, respectively. $\mathbf{C}$ is the truncated discrete cosine transform (DCT) matrix, and $\mathbf{C}^{-1}$ indicates its pseudoinverse. $\boldsymbol{\alpha}$ denotes the phase factor [16, 17].

### 2.1. Reformulation as a generative model

In nature, JUD can be represented by a generative model for each regression class $r$ as [10]

$$\mathbf{y}_t = \mathbf{H}^{(r)}\mathbf{x}_t + \mathbf{g}^{(r)} + \mathbf{n}_t^{(r)}, \quad \mathbf{n}_t^{(r)} \sim \mathcal{N}\Big(\mathbf{0}, \mathbf{\Phi}^{(r)}\Big) \tag{5}$$

where $\mathbf{H}^{(r)}$ is a linear transform, $\mathbf{g}^{(r)}$ denote the bias term and $\mathbf{n}_t^{(r)}$ is a Gaussian additive noise. From equation (5), the conditional distribution of $\mathbf{y}_t$ given $\mathbf{x}_t$ for each regression class can be obtained as

$$p(\mathbf{y}_t|\mathbf{x}_t, r) = \mathcal{N}\Big(\mathbf{y}_t; \mathbf{H}^{(r)}\mathbf{x}_t + \mathbf{g}^{(r)}, \mathbf{\Phi}^{(r)}\Big). \tag{6}$$

Given this distribution, the original JUD likelihood function (3) can be obtained by substituting equation (6) into (2) by setting the JUD transformation parameters to be $\mathbf{A}^{(r)} = \mathbf{H}^{(r)-1}, \mathbf{b}^{(r)} = -\mathbf{H}^{(r)-1}\mathbf{g}^{(r)}$ and $\mathbf{\Sigma}_b^{(r)} = \mathbf{A}^{(r)}\mathbf{\Phi}^{(r)}\mathbf{A}^{(r)T}$.

The generative view of JUD is particularly useful, since it makes it possible to estimate the JUD transforms in a data-driven fashion. It is more flexible as it gets rid of the mismatch function (4). For instance, a successful example can be found in [10] which is also known as Noisy-CMLLR. Meanwhile, an EM algorithm can also be derived to update the acoustic model parameter for adaptive training as in [10, 18]. This algorithm will be used in this paper for noise adaptive training of SGMMs which will be further discussed in section 3.

### 2.2. Compensation of SGMMs

The SGMM acoustic model is proposed by Povey et. al [11], in which the GMM parameters of each HMM state are derived from low-dimensional model subspace. It has obtained success in standard telephone speech transcription [11] as well as in the cross-lingual and multilingual settings [19, 20]. For noise compensation with JUD [12, 13], one of the key configurations is using the universal background model (UBM) in the SGMM as the regression model for JUD which enjoys particular advantages in implementation simplicity and computational efficiency. This will result in the likelihood function for the HMM state $j$ as

$$p(\mathbf{y}_t \mid j, \mathcal{T}) = \sum_{k=1}^{K_j} c_{jk} \sum_{i=1}^{I} w_{jki} |\mathbf{A}^{(r)}| \times \mathcal{N}\Big(\mathbf{A}^{(r)}\mathbf{y}_t + \mathbf{b}^{(r)}; \boldsymbol{\mu}_{jki}, \mathbf{\Sigma}_i + \mathbf{\Sigma}_b^{(r)}\Big) \tag{7}$$

where $c_{jk}$ and $w_{jki}$ are the sub-state and Gaussian component weights, $I$ denotes the number of Gaussians in the UBM and $K_j$ is the number of sub-states for state $j$. $\mathbf{\Sigma}_i$ is the covariance matrix that is state-independent. The Gaussian mean and weight are derived from

$$\boldsymbol{\mu}_{jki} = \mathbf{M}_i\mathbf{v}_{jk}, \quad w_{jki} = \frac{\exp \mathbf{w}_i^T\mathbf{v}_{jk}}{\sum_{i'=1}^{I} \exp \mathbf{w}_{i'}^T\mathbf{v}_{jk}}. \tag{8}$$

Here, $\mathbf{M}_i$ and $\mathbf{w}_i$ are the mean and weight projections, and $\mathbf{v}_{jk}$ is the state vector which is normally low dimensional. The regression class index $r$ will be replaced by the UBM component index $i$ if using the UBM as the regression model for JUD [13]. As the usual practice [3, 17], noise compensation was employed on the per-utterance basis in which the noise condition is assumed to be the same. This means that the JUD transformation $\mathcal{T}$ will depend on the utterance index. For clarity, we do not introduce an additional notation for the utterance index without causing confusions. In the following, we will further refer $\mathcal{M}$ the SGMM acoustic model parameters.

## 3. Noise adaptive training

Noise adaptive training (NAT) of the acoustic model involves joint optimisation of the acoustic model parameters $\mathcal{M}$ and the transformation parameters $\mathcal{T}$. For an SGMM acoustic model, the objective function for NAT can be expressed as

$$\mathcal{Q}\Big(\tilde{\mathcal{M}}, \tilde{\mathcal{T}}; \mathcal{M}, \mathcal{T}\Big) = \sum_{jkit} \gamma_{jki}(t) \log |\mathbf{A}^{(r)}| \times \mathcal{N}\Big(\mathbf{A}^{(r)}\mathbf{y}_t + \mathbf{b}^{(r)}; \boldsymbol{\mu}_{jki}, \mathbf{\Sigma}_i + \mathbf{\Sigma}_b^{(r)}\Big) \tag{9}$$

where $\gamma_{jki}(t)$ is the posterior probability for frame $t$, $\tilde{\mathcal{M}}$ and $\tilde{\mathcal{T}}$ denote the new estimate of the model and transformation parameters. Note that this objective function is for a particular training utterance that the transformation parameters $\mathcal{T}$ depends on. The overall objective function for the total training utterances can be obtained by summing equation (9) with the corresponding $\mathcal{T}$.

Directly optimising the parameters in both of the $\mathcal{M}$ and $\mathcal{T}$ is normally infeasible, especially for an SGMM acoustic model since the objective function is more complex. Analogous to SAT [1], a common practice is to interleave the update of $\mathcal{M}$ and $\mathcal{T}$ one after another [3, 5]. In this paper, we adopt the same principle for adaptive training of SGMMs. The estimation of $\mathcal{T}$ given $\mathcal{M}$ has been detailed in [13], whereas in this paper, we focus on the estimation of the acoustic model parameter $\mathcal{M}$ given the estimate of $\mathcal{T}$.

### 3.1. Optimisation

In literature, there are two optimisation approaches to update the acoustic model parameters $\mathcal{M}$ for NAT. The first one is the second-order gradient-based approach, in which, a particular set of parameters $\theta$ in $\mathcal{M}$ is updated by

$$\tilde{\theta} = \theta - \zeta\left(\frac{\partial^2\mathcal{Q}(\cdot)}{\partial^2\theta}\right)^{-1}\left(\frac{\partial\mathcal{Q}(\cdot)}{\partial\theta}\right) \tag{10}$$

where $\tilde{\theta}$ is the new value of $\theta$, $\zeta$ is the learning rate and $\mathcal{Q}(\cdot)$ denotes the objective function (9). Typical examples of its applications are [5] for JUD-GMM system and [3] for VTS-GMM system. Depending on the form of Hessian it used, it may yield faster convergence. However, the drawbacks of this approach are that the computation of the gradient and Hessian terms in (10) can be complex, especially for the SGMM-based acoustic models due to the compact model representation. Furthermore, the discriminative criteria may not be simply applied with this type of optimisation as discussed in [18].

The second type of optimisation is based on the EM algorithm, which is derived from the generative perspective of JUD transformation as equation (5). The essence of this method is to estimate the sufficient statistics of the expected "pseudo-clean" speech feature $\mathbf{x}_t$. It is obtained by computing its conditional distribution which depends on the component $m$ as

$$p(\mathbf{x}_t|\mathbf{y}_t, r, m) = \frac{p(\mathbf{y}_t|\mathbf{x}_t, r)p(\mathbf{x}_t|m)}{\int p(\mathbf{y}_t|\mathbf{x}_t, r)p(\mathbf{x}_t|m)d\mathbf{x}_t}. \tag{11}$$

As shown in [10], an analytical solution can be obtained from equation (6), which gives the conditional expectations as

$$\mathbb{E}[\mathbf{x}_t|\mathbf{y}_t, r, m] = \tilde{\mathbf{x}}_t^{(rm)} \tag{12}$$

$$\mathbb{E}[\mathbf{x}_t\mathbf{x}_t^T|\mathbf{y}, r, m] = \tilde{\mathbf{\Sigma}}_x^{(rm)} + \tilde{\mathbf{x}}_t^{(rm)}\tilde{\mathbf{x}}_t^{(rm)T} \tag{13}$$

where

$$\begin{aligned} \tilde{\mathbf{x}}_t^{(rm)} &= \tilde{\mathbf{A}}^{(rm)}\mathbf{y}_t + \tilde{\mathbf{b}}^{(rm)} \\ \tilde{\mathbf{\Sigma}}_x^{(rm)} &= \left(\mathbf{\Sigma}_x^{(m)-1} + \mathbf{\Sigma}_b^{(r)-1}\right)^{-1} \\ \tilde{\mathbf{A}}^{(rm)} &= \tilde{\mathbf{\Sigma}}_x^{(rm)}\mathbf{\Sigma}_b^{(r)-1}\mathbf{A}^{(r)} \\ \tilde{\mathbf{b}}^{(rm)} &= \tilde{\mathbf{\Sigma}}_x^{(rm)}\left(\mathbf{\Sigma}_x^{(m)-1}\boldsymbol{\mu}_x^{(m)} + \mathbf{\Sigma}_b^{(r)-1}\mathbf{b}^{(r)}\right) \end{aligned}$$

where $\boldsymbol{\mu}_x^{(m)}$ and $\mathbf{\Sigma}_x^{(m)}$ are the mean and covariance of Gaussian component $m$. Given the expectations, the statistics can be accumulated in the standard fashion to re-estimate the acoustic model parameters. This method makes the implementation much simpler and hence has been used in this work.

### 3.2. Model update

Using the EM-based algorithm as aforementioned, it only involves minor changes in the original model estimation formula of the SGMMs presented in [11]. Taking the estimation of the Gaussian mean projection $\mathbf{M}_i$ for instance, the auxiliary function is

$$\mathcal{Q}(\mathbf{M}_i) = tr\left(\mathbf{M}_i^T\mathbf{\Sigma}_i^{-1}\mathbf{Y}_i\right) - \frac{1}{2}tr\left(\mathbf{M}_i^T\mathbf{\Sigma}_i^{-1}\mathbf{M}_i\mathbf{Q}_i\right) \tag{14}$$

where the sufficient statistics $\mathbf{Y}_i$ and $\mathbf{Q}_i$ will be obtained as

$$\mathbf{Y}_i = \sum_{jkt}\gamma_{jki}(t)\mathbb{E}[\mathbf{x}_t|\mathbf{y}_t, r, m]\mathbf{v}_{jk}^T \tag{15}$$

$$\mathbf{Q}_i = \sum_{jkt}\gamma_{jki}(t)\mathbf{v}_{jk}\mathbf{v}_{jk}^T \tag{16}$$

Note that in an SGMM, the Gaussian component index $m$ will be replaced by $jki$ as in equation (7), and the regression class index $r$ is replaced by $i$. It also worths emphasising that the posteriori probability $\gamma_{jki}(t)$ should be computed using the noisy feature vector $\mathbf{y}_t$ by the likelihood function (7) in the adaptive training scheme.

Likewise, other types of SGMM acoustic model parameters such as $\mathbf{v}_{jk}$ and $\mathbf{\Sigma}_i$ can be estimated in the same fashion using the expectations of the "pseudo-clean" feature vectors. The EM-based algorithm for NAT is similar to some feature enhancement methods which also estimate $\mathbf{x}_t$ given $\mathbf{y}_t$, e.g. [6]. However, a fundamental difference is that the conditional expectations directly relate to the acoustic model structure as in (12) and (13), while for feature enhancement they are normally derived using a frond-end GMM. Due to the more close match to the acoustic model, NAT was found to outperform its feature enhancement counterpart in [21].

Finally, it is worthwhile to point out that the UBM that associated with the SGMM acoustic model needs also be updated during the adaptive training. This is because that the UBM is used as a clustering of the Gaussian components in the SGMM when applying JUD [13]. After NAT, the SGMM is based on "pseudo-clean" feature space which is different from that the UBM is originally trained on. In this work, the UBM is updated using the weighted average of the corresponding Gaussian component in the SGMM as

$$\mathbf{\Sigma}_i^{ubm} = \mathbf{\Sigma}_i \tag{17}$$

$$w_i^{ubm} = \frac{\sum_{jkt}\gamma_{jki}(t)}{\sum_{jkit}\gamma_{jki}(t)} \tag{18}$$

$$\boldsymbol{\mu}_i^{ubm} = \sum_{jkt}\gamma_{jki}(t)\mathbf{M}_i\mathbf{v}_{jk} \tag{19}$$

where $w_i^{ubm}$, $\boldsymbol{\mu}_i^{ubm}$ and $\mathbf{\Sigma}_i^{ubm}$ are the weight, mean and covariance matrix for component $i$ in the UBM respectively. Updating the UBM was found to improve the recognition accuracy of the NAT system.

### 3.3. Training recipe

To sum up, the NAT recipe for an SGMM acoustic model used in this paper is as follows.

1. Initialise the acoustic model $\mathcal{M}$ by the standard maximum likelihood training.
2. For each training utterance, initialise the noise model parameters for $\mathbf{n}_t$ and $\mathbf{h}_t$ in (4).
3. Re-estimate the noise model parameters given $\mathcal{M}$.
4. Obtain the JUD transformation parameters $\mathcal{T}$.
5. Given $\mathcal{M}$ and $\mathcal{T}$, compute the posterior probability $\gamma_{jki}(t)$ using equation (7).
6. Accumulate the statistics using the conditional expectations (12) (13) and update $\mathcal{M}$.
7. Go to step 4 until convergence.
8. Update the UBM using equations (17) - (19).
9. Go to step 1 until the number of iterations is reached.

While this paper focuses on the NAT algorithm for the SGMMs, more details about noise model and JUD transform estimation used in step 2 to step 4 can be found in [13].

## 4. Experiments

The experiments were performed using the Aurora 4 corpus, which is derived from the Wall Street Journal (WSJ0) 5,000-word (5k) closed vocabulary transcription task. The clean training set contains about 15 hours of audio, and Aurora 4 provides a noisy version of the training set which is contaminated by

Table 1: Word error rates (WERs) of SGMM systems with and without noise adaptive training.

| Methods | A | B | C | D | Avg |
|---|---|---|---|---|---|
| Clean model | 5.2 | 58.2 | 50.7 | 72.1 | 59.9 |
| +JUD | 5.1 | 13.1 | 12.0 | 23.2 | 16.8 |
| MST model | 6.8 | 15.2 | 18.6 | 32.3 | 22.2 |
| +JUD | 7.4 | 13.3 | 14.7 | 24.1 | 17.6 |
| NAT model | 6.5 | 20.3 | 19.8 | 39.7 | 27.6 |
| +JUD | 6.1 | 11.3 | 11.9 | 22.4 | 15.7 |

the artificial noise in different conditions. This training set enables the multi-style training (MST) as well as noise adaptive training (NAT) of the acoustic models. The test set has 300 utterances from 8 speakers. The first test set, set A (`test01`), was recorded using a close talking microphone, similar to the clean training data. The data comprising set B (`test02` to `test07`) was obtained by adding six different types of noise, with randomly selected signal-to-noise ratios ranging from 5dB to 15dB, to set A. Set C (`test08`) was recording using a desk-mounted secondary microphone and the same type of noise used for set B was added to this test set to form set D (`test09` to `test14`).

In the following experiments, we used 39 dimensional feature vectors which is derived from 12th order mel frequency cepstral coefficients, plus the zeroth order coefficient (C0), with delta and acceleration features. We used the standard WSJ0 5k bigram language model [22]. The SGMM systems have about 3900 tied triphone states, 16,000 sub-states, and we used $I = 400$ in the UBM, which results in 6.4 million surface Gaussians. As mentioned before, the phase-sensitive mismatch function (4) was used for estimating the JUD transforms. Based on the previous findings in [13], all the entries in $\boldsymbol{\alpha}$ were empirically set to be 2.5 in both training and decoding stages unless otherwise specified.

### 4.1. Results

The experimental results are given in Table 1 using the clean, MST and NAT acoustic models. The NAT system was trained following the recipe in section 3.3, where we performed 4 iterations in step 7 which yielded convergence, and only 1 iterations in step 9. As expected, the MST system significantly outperforms the clean trained system without JUD compensation since the mismatch between the training and testing data is reduced. However, with JUD compensation we observe the opposite results with WER as 17.6% (MST) vs. 16.8% (Clean). This may due to that the MST model captures much noise related variability from the training data which makes it not suitable for rapid adaptation towards to a particular noise condition using the limited adaptation data. The NAT system, on the other hand, normalises the irrelevant variability in the training data using noise dependent JUD transforms. Without JUD in the decoding stage, this model results in higher WER since it does not match the testing data well. With JUD adaptation, it significantly outperforms the MST and clean system with WER at 15.7%, which is slightly better that 16.0% by the adaptive trained GMM system using VTS on the same dataset [23].

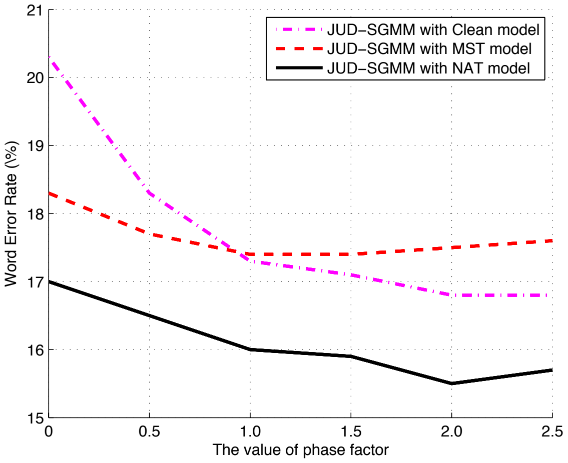

Figure 1: Results of tuning the value of phase factor $\boldsymbol{\alpha}$ in the decoding stage.

Our results are based on the phase-sensitive mismatch function. Previous work on the phase factor $\boldsymbol{\alpha}$ in equation (4) has shown that it is able to bring significant gains in both VTS and JUD based noise robust speech recognition systems [16, 17, 13]. Though the theoretical values of elements in $\boldsymbol{\alpha}$ should be in the range of $[-1, 1]$ [16], empirical studies demonstrated that better results can be obtained by setting the value of $\boldsymbol{\alpha}$ out of this range [17, 13]. A good explanation for this is that $\boldsymbol{\alpha}$ can be viewed as additional model parameters whose value can be tuned to mitigate the mismatch between the training and testing data [24]. While previous studies regarding to this issue are mainly based on systems training on clean data [17, 13], we obtain further insight based on our MST and NAT systems. Figure 1 shows the WER of the systems using the three models by empirically tuning the values of $\boldsymbol{\alpha}$ in the decoding stage as in [17, 13]. It shows that tuning the value of $\boldsymbol{\alpha}$ results in gains for all the three systems, e.g. 15.5% ($\boldsymbol{\alpha} = \mathbf{2.0}$) vs. 17.0% ($\boldsymbol{\alpha} = \mathbf{0}$) for NAT system. However, the improvement is much less for MST and NAT systems that trained on multi-condition data compared to that trained on the clean data. These results support the previous argument stating that $\boldsymbol{\alpha}$ can be tuned to account for the mismatch between the training and testing conditions. Note that, the results were obtained by tuning $\boldsymbol{\alpha}$ in the decoding phase only, future work will be on the investigation of its effect on the training stage for NAT system.

## 5. Conclusions

This paper studies the noise adaptive training (NAT) algorithm for an SGMM acoustic model using multi-condition training data. Our method is based the joint uncertainty decoding (JUD) noise compensation technique. For adaptive training, the EM-based optimisation algorithm is employed which is derived from reformulating JUD adaptation into a generative model. This algorithm is proven to be simple for implementation, and effective in terms of recognition accuracy. Evaluation was carried out on the Aurora 4 dataset, and with NAT, the SGMM system achieved the lowest WER at 15.5% which is state-of-the-art on this task. The experiments are also helpful to understand the effect of phase factor in the mismatch function. Future work will be on applying the discriminative criterion to the adaptive trained system that has been found effective with GMM based systems [18, 24].

**Acknowledgement** This work is funded by the EPSRC Programme Grant EP/I031022/1 (Natural Speech Technology).

# 6. References

[1] Y. Anastasakos, J. McDonough, R. Schwartz, and J. Makhoul, "A compact model for speaker-adaptive training," in *Proc. ICSLP*, 1996.

[2] L. Deng, A. Acero, M. Plumpe, and X. Huang, "Large-vocabulary speech recognition under adverse acoustic environments," in *Proc. ICSLP*, 2000.

[3] O. Kalinli, M. Seltzer, J. Droppo, and A. Acero, "Noise adaptive training for robust automatic speech recognition," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 18, no. 8, pp. 1889–1091, 2010.

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

[5] H. Liao and M. Gales, "Adaptive training with joint uncertainty decoding for robust recognition of noisy data," in *Proc. ICASSP*. IEEE, 2007.

[6] P. Moreno, B. Raj, and R. Stern, "A vector Taylor series approach for environment-independent speech recognition," in *Proc. ICASSP*, vol. 2. IEEE, 1996, pp. 733–736.

[7] H. Liao, "Uncertainty decoding for noise robust speech recognition," Ph.D. dissertation, University of Cambridge, 2007.

[8] J. Droppo, A. Acero, and L. Deng, "Uncertainty decoding with SPLICE for noise robust speech recognition," in *Proc. ICASSP*. IEEE, 2002.

[9] H. Liao and M. Gales, "Joint uncertainty decoding for noise robust speech recognition," in *Proc. INTERSPEECH*. Citeseer, 2005.

[10] D. Kim and M. Gales, "Noisy constrained maximum-likelihood linear regression for noise-robust speech recognition," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 19, no. 2, pp. 315–325, 2011.

[11] D. Povey, L. Burget, M. Agarwal, P. Akyazi, F. Kai, A. Ghoshal, O. Glembek, N. Goel, M. Karafiát, A. Rastrow, R. Rose, P. Schwarz, and S. Thomas, "The subspace Gaussian mixture model—A structured model for speech recognition," *Computer Speech & Language*, vol. 25, no. 2, pp. 404–439, 2011.

[12] L. Lu, K. Chin, A. Ghoshal, and S. Renals, "Noise compensation for subspace Gaussian mixture models," in *Proc. INTERSPEECH*, 2012.

[13] ——, "Joint uncertainty decoding for noise robust subspace Gaussian mixture models," *IEEE Transactions on Audio, Speech, and Language Processing*, 2013.

[14] H. Xu, L. Rigazio, and D. Kryze, "Vector Taylor series based joint uncertainty decoding," in *Proc. Interspeech*, 2006.

[15] A. Acero, L. Deng, T. Kristjansson, and J. Zhang, "HMM adaptation using vector Taylor series for noisy speech recognition," in *Proc. ICSLP*, 2000.

[16] 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 Transactions on Speech and Audio Processing*, vol. 12, no. 2, pp. 133–143, 2004.

[17] J. Li, L. Deng, D. Yu, Y. Gong, and A. Acero, "A unified framework of HMM adaptation with joint compensation of additive and convolutive distortions," *Computer Speech & Language*, vol. 23, no. 3, pp. 389–405, 2009.

[18] F. Flego and M. Gales, "Discriminative adaptive training with VTS and JUD," in *Proc. ASRU*. IEEE, 2009, pp. 170–175.

[19] L. Lu, A. Ghoshal, and S. Renals, "Regularized subspace Gaussian mixture models for cross-lingual speech recognition," in *Proc. IEEE ASRU*, 2011.

[20] L. Burget, P. Schwarz, M. Agarwal, P. Akyazi, K. Feng, A. Ghoshal, O. Glembek, N. Goel, M. Karafiát, D. Povey, A. Rastrow, R. Rose, and S. Thomas, "Multilingual acoustic modeling for speech recognition based on subspace Gaussian mixture models," in *Proc. IEEE ICASSP*, 2010, pp. 4334–4337.

[21] J. Li, M. Seltzer, and Y. Gong, "Improvements to VTS feature enhanecement," in *Proc. ICASSP*. IEEE, 2012, pp. 4677–4680.

[22] D. Paul and J. Baker, "The design for the Wall Street Journal-based CSR corpus," in *Second International Conference on Spoken Language Processing*, 1992.

[23] F. Flego and M. Gales, "Factor analysis based VTS discriminative adaptive training," in *Proc. ICASSP*. IEEE, 2012, pp. 4669–4672.

[24] M. Gales and F. Flego, "Discriminative classifiers with adaptive kernels for noise robust speech recognition," *Computer Speech & Language*, vol. 24, no. 4, pp. 648–662, 2010.