Feature Adaptation Using Linear Spectro-Temporal Transform for Robust Speech Recognition

Nguyen, Duc Hoang Ha; Xiao, Xiong; Chng, Eng Siong; Li, Haizhou

2016

Nguyen, D. H. H., Xiao, X., Chng, E. S., & Li, H. (2016). Feature Adaptation Using Linear Spectro-Temporal Transform for Robust Speech Recognition. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 24(6), 1006-1019.

<https://hdl.handle.net/10356/84664>

<https://doi.org/10.1109/TASLP.2016.2522646>

© 2016 IEEE. Personal use of this material is permitted. Permission from IEEE must be obtained for all other uses, in any current or future media, including reprinting/republishing this material for advertising or promotional purposes, creating new collective works, for resale or redistribution to servers or lists, or reuse of any copyrighted component of this work in other works. The published version is available at: [<http://dx.doi.org/10.1109/TASLP.2016.2522646>].

Downloaded on 26 Apr 2023 18:17:37 SGT

Feature Adaptation Using Linear Spectro-Temporal Transform for Robust Speech Recognition

Duc Hoang Ha Nguyen, Xiong Xiao, *Member, IEEE*, Eng Siong Chng, *Senior Member, IEEE*, and Haizhou Li, *Fellow, IEEE*,

Abstract—Spectral information represents short term speech information within a frame of a few tens of milliseconds, while temporal information captures the evolution of speech statistics over consecutive frames. Motivated by the findings that human speech comprehension relies on the integrity of both the spectral content and temporal envelope of speech signal, we study a spectro-temporal transform framework that adapts runtime speech features to minimize the mismatch between runtime and training data, and its implementation that includes cross transform and cascaded transform. A Kullback Leibler divergence based cost function is proposed to estimate the transform parameters. We conducted experiments on the REVERB Challenge 2014 task, where clean and multi-condition trained acoustic models are tested with real reverberant and noisy speech. We found that temporal information is important for reverberant speech recognition and the simultaneous use of spectral and temporal information for feature adaptation is effective. We also investigate the combination of the cross transform with fMLLR, the combination of batch, utterance and speaker mode adaptation, and multi-condition adaptive training using proposed transforms. All experiments consistently report significant word error rate reductions.

Index Terms—Feature adaptation, temporal filtering, linear transform, robust speech recognition.

I. INTRODUCTION

Automatic Speech Recognition (ASR) decodes speech signals into text [1]. The performance of ASR systems has improved greatly in recent years due to more training data, increased computational power, and deep learning algorithm for acoustic modelling [2]. While ASR is expected to produce accurate word recognition in clean environment, its accuracy degrades considerably in noisy and reverberant acoustic environments. Robustness of ASR systems in adverse environments for real-world applications remains a challenge.

To address the robustness issue, many techniques have been proposed in the last three decades. They can be broadly categorized into two main approaches: model-based and feature-based approaches. Model-based approach aim to update the acoustic model to better represent speech features under new test conditions. Examples include maximum *a-posteriori* (MAP) adaptation [3], maximum likelihood linear regression (MLLR) [4], [5] and their variants [6]–[9], and vector Taylor series (VTS) based adaptation [10]–[12]. Feature-based approach, on the other hand, aim to bring speech signals/features closer to the ones used during training. Examples include:

Duc Hoang Ha Nguyen, Xiong Xiao, Eng Siong Chng, and Haizhou Li are with Nanyang Technological University, Singapore.

Haizhou Li is also with the National University of Singapore and the Institute for Infocomm Research, Singapore.

Fig. 1. Illustrations of 5 types of linear transform-based feature adaptation. A) scalar form transform (e.g. MVN [25]) where the transformed feature $y_t^{(d)}$ at frame t and element d is a scaled and shifted version of the observation $x_t^{(d)}$. B) vector form transform (e.g. fMLLR [4], [29]) where $y_t^{(d)}$ is a weighted sum of all observed features in current frame t plus a bias. C) trajectory form transform (e.g. MNLLF [33]) where $y_t^{(d)}$ is a weighted sum of local feature trajectory of element d . D) the proposed cross transform which is a combination of B) and C). E) the proposed Spectro-Temporal (ST) transform which uses the joint spectro-temporal information.

speech enhancement methods such as spectral subtraction [13], Wiener filter [14], minimum mean square error (MMSE) short time spectral amplitude estimator [15], [16]; dereverberation [17]–[22]; feature compensation methods such as SPLICE [23]; feature normalization methods such as cepstral mean normalization (CMN) [24], mean and variance normalization (MVN) [25], and histogram equalization (HEQ) [26]; temporal filters such as relative spectral processing (RASTA) [27], [28]; and linear feature transform such as feature-space MLLR (fMLLR) [4], [29]. For a comprehensive review of the existing techniques, readers may refer to [30]–[32]. Here, we look into several feature-based techniques to motivate our work.

A subset of feature based methods adapt the test features towards the training features, hence, they are called feature adaptation methods in this paper. Feature adaptation methods can be categorized into three types by the form of their inputs, as illustrated in Fig. 1: A) scalar form input, B) vector form input and C) trajectory form input.

By *scalar form*, we mean that the processing of each time-frequency point in the feature representation is independent once the transform parameters are determined. For example, a linear transform, $y_t^{(d)} = b^{(d)} + a^{(d)}x_t^{(d)}$, consists of only a scale factor $a^{(d)}$ and bias $b^{(d)}$. The superscript (d) indicates the element of the feature vector, and the subscript t is the frame index. Examples are cepstral mean normalization (CMN) [24]

where $a^{(d)} = 1$ and $b^{(d)} = -\mu_x^{(d)}$, where $\mu_x^{(d)}$ is the test feature's mean. Cepstral mean and variance normalization (MVN) [25] extends CMN by also using $a^{(d)} = 1/\sigma^{(d)}$ where $\sigma^{(d)}$ is the standard deviation of the test features. While CMN and MVN are linear, histogram equalization (HEQ) [26] generalizes them to nonlinear processing. In HEQ, the processed feature is obtained by $y_t = C_y^{-1}(C_x(x_t))$ for each feature element individually, where $C_x()$ is the cumulative density function (CDF) of the test feature and $C_y^{-1}()$ is the inverse CDF of the training feature¹. More complex and higher order transforms of scalar input are studied in [34], [35]. Multi-class extensions can also be applied to these scalar forms [36]–[39]. As the transform parameters, such as scale, bias, and CDF, are estimated from the whole feature trajectory, the temporal information is therefore in a way used in the scalar form processing. However, this is different from the explicit use of temporal information that will be introduced in temporal filtering later in this paper.

In the *vector form* (Fig. 1B), we use a whole feature vector at each time t as the input. The fMLLR [4], [29], [40] and feature space stochastic matching [41] are examples, in which we have $\mathbf{y}_t = \mathbf{A}\mathbf{x}_t + \mathbf{b}$, where $\mathbf{A} \in \mathcal{R}^{D \times D}$ and $\mathbf{b} \in \mathcal{R}^D$ are the transform parameters for feature vectors of D dimensions. The parameters are usually estimated to fit the processed features to a reference acoustic model [4], [29]. If we reduce the $\mathbf{A}$ matrix into scalar form as shown in Fig. 1B, it is clear that the output feature $y_t^{(d)}$ is a weighted sum of all feature elements at current frame t , i.e. $[x_t^{(1)}, \dots, x_t^{(D)}]$. From a regression point of view, we are trying to predict $y_t^{(d)}$ based on its correlation to all the input feature elements at time t . In state-of-the-art ASR systems, a feature vector typically contains the cepstrum of the frame as well as its first and second-order time derivatives, hence carries temporal information.

In the *trajectory form* (Fig. 1C), the feature processing operates on the feature trajectories along the time axis. This form is usually referred to as temporal filtering of feature trajectories, such as RASTA [28], ARMA [42] and TSN [43]. The temporal filters can also be interpreted as linear transform of feature trajectory. Unlike the vector form transform that takes feature vectors as the inputs, the temporal filter takes feature trajectories as the inputs.

Various designs of linear temporal filtering have been studied in the past. From the filtering point of view, the temporal filters modify the power spectral density (PSD) of the feature trajectories, which can be seen as a form of modulation spectra of speech. The modulation spectra of speech signals are shown to be correlated with speech intelligibility and not all modulation frequencies are equally important. Generally speaking, speech energy in the 1-16Hz modulation frequency range is important to speech intelligibility and automatic speech recognition, while very low (<1 Hz) and high (>16 Hz) modulation frequency ranges are less important and often affected by noise [44]–[48]. Therefore, the RASTA filter [28] uses a bandpass filter to remove very low (<1 Hz) and high modulation frequencies (>16 Hz). Similarly, the autoregressive moving average (ARMA) filter uses a low pass filter and is

usually used after MVN to form the MVA processing [42]. Recently, some temporal filters are proposed purely from an engineering point of view. For example, in [43], the modulation spectrum of noisy speech is found to deviate from that of clean speech, hence the temporal structure normalization (TSN) filter is proposed to normalize the modulation spectrum of the noisy test speech to that of the clean speech. While RASTA and ARMA filters are manually designed offline and fixed during acoustic model training and testing, the TSN filter is estimated dynamically for each utterance. The TSN filter is improved in joint spectral and temporal normalization (JSTN) [49] by introducing a maximum likelihood (ML) criterion and a reference Gaussian mixture model (GMM) trained from clean features to guide the filter design. In [33], temporal filters are estimated by maximizing the likelihood of the filtered features on a clean GMM while promoting the log determinant of the covariance of the filtered features. The temporal filters usually make use of the long term temporal information of the features, e.g. a filter of length 31 taps is used in [43], allowing the filter to use much longer temporal information than the one used in the vector form linear transform.

The temporal filters are also used to reduce the effects of reverberation which can be considered as convolutional noise in the short time Fourier transform (STFT) domain. As such, a linear filter such as long-term linear prediction [19], [50] can be applied to deconvolution. In [17]–[19], such filter is applied to STFT coefficients and in [20]–[22], to power spectra. In this paper, we look into how temporal filters can be directly applied to speech features for feature adaptation task, e.g. MFCC.

Many studies show that the spectral and temporal information of speech signals are both important to human perception of speech and sound. It is shown that human speech comprehension depends on the integrity of both the spectral content and temporal envelope of the speech signal [51]. It has been reported that human auditory neurons are tuned to detect local spectro-temporal patterns of speech [52], [53], which motivates the use of Gabor filters to extract local spectro-temporal patterns from speech spectrograms for speech recognition [54]. More recently, a two dimensional modulation filtering scheme was proposed to improve the robustness of speaker and language recognition by using a temporal autoregressive (AR) model and spectral AR model [55]. Furthering these studies, we look into the integration of the spectral and temporal information of speech in feature adaptation for robust speech recognition in this paper. A sequence of feature vectors are projected into single feature vector by a linear transform as shown in Fig. 1E.

There have been several other studies in predicting features from a sequence of feature vectors. In probabilistic optimal filtering (POF) [56], a sequence of corrupted feature vectors are mapped to clean features by class-dependent linear transforms. It is assumed that simultaneous recordings of clean and noisy speech are available for training the transforms via least square regression. Recently, neural networks (NN) and deep neural networks (DNN) [57] are used for feature compensation [58]–[63]. DNN usually takes a sequence of feature vectors as input and applies several interleaving linear transforms and nonlinear activation functions to the inputs. Both DNN and

¹To avoid cluttered notation, we omit the superscript $^{(d)}$.

Fig. 2. An illustration of a feature adaptation system. In the Feature Adaptation block, we can apply fMLLR, MNLLF or the proposed cross transform.

POF use pretrained transforms that usually do not adapt at run-time, so they do not belong to the feature adaptation approach.

Several feature adaptation methods that use consecutive feature vectors have been studied in the past. In [64], a single hidden layer NN is used to map several frames of feature vectors to a bias term to the features. The network is estimated by maximizing the likelihood of the bias compensated features. However, this study is not carried out on reverberant speech recognition and only up to 11 frames of input are used. In [65], a special type of fMLLR is used to project upto 9 frames of static features to a reduced subspace for speaker adaptation. While the reduced subspace is modeled by a conventional HMM/GMM acoustic model, the rejected space corresponding to the discarded dimensions are modeled by a single diagonal Gaussian in a similar way to heteroscedastic linear discriminant analysis (HLDA) [66]. Hence, the method can be considered as the speaker dependent version of HLDA. After the square fMLLR transform matrix is estimated, the rows of the transform corresponding to the rejected space is discarded. Later, a similar variant of fMLLR called direct CMLLR (DCMLLR) is studied in [10]. DCMLLR and its variants take multiple frames of static cepstral feature vectors as input and predict both static and dynamic cepstral features to deal with reverberation. The estimation method of the DCMLLR is similar to the one proposed in [10]. It is reported that by using upto 17 frames of contextual feature vectors one achieves consistent improvement on a reverberant speech recognition task for clean-condition trained acoustic models.

A. Contributions of this paper

Although there have been some studies on feature adaptation using spectral information, temporal information, and joint spectro-temporal information (such as DCMLLR), there hasn't been a unified view of the three types of transforms in a single framework for robust speech recognition. In this paper, we aim to provide a unified view of linear feature adaptation techniques and compare their performance in a reverberant and noisy speech recognition task.

The contributions of this paper can be summarized as follows:

- We propose a KL divergence based objective function and EM-based method for the spectro-temporal (ST) transform estimation.

- We provide a unified view of spectral, temporal, and spectro-temporal transforms. We propose a sparse and efficient ST transform called cross-transform (Fig. 1D).
- We study the ways for reliably estimating ST transforms, including sparse transforms, cascaded transforms, regularization and statistics interpolation.
- We study the effectiveness of ST transform in both clean and multi-condition adaptive training scenarios for reverberant and noisy speech recognition.

The rest of the paper is organized as follows. In Section II, we introduce the ST transform. In Section III, we discuss its implementation issues. We report the experiments in Section IV. Finally, we conclude in Section V and suggest future work.

II. FEATURE ADAPTATION USING SPECTRO-TEMPORAL INFORMATION

In this section, we introduce a generalized linear feature transform, i.e. ST transform, that utilizes spectro-temporal information for feature adaptation.

A. Feature Adaptation in ASR

ASR performance degrades in face of mismatch between the training and test features. Mismatch may occur due to various reasons, such as speaker variation, background noise, reverberation, and transmission channel. One way to reduce mismatch is to adapt the statistics of test features towards those of training features. In doing so, various techniques, such as CMN, MVN, HEQ, and fMLLR have been developed.

Fig. 2 illustrates the role of a feature adaptation block in an ASR system. It adapts the run-time noisy/reverberant data towards an acoustic model trained on clean data.

B. Generalized Linear Transform for Feature Adaptation

We now study a generalized linear feature transform, namely ST transform, as follows:

$$\mathbf{y}_t = \sum_{\tau=-L}^L \mathbf{A}_\tau \mathbf{x}_{t+\tau} + \mathbf{b} = \mathbf{W} \tilde{\mathbf{x}}_t \quad (1)$$

where $\mathbf{x}_t$ and $\mathbf{y}_t$ are the input and output feature vectors at frame t , respectively. $\mathbf{A}_\tau$ (with $\tau = -L, \dots, L$) is a sequence of transform matrices over $2L+1$ frames. $\mathbf{b}$ is a bias vector. ST transform is defined as a linear transform over a supervector of $2L+1$ stacked feature vectors $\tilde{\mathbf{x}}_t = [\mathbf{x}_{t-L}^T, \dots, \mathbf{x}_{t+L}^T, 1]^T$ with a transform matrix (or the general transform matrix) $\mathbf{W} = [\mathbf{A}_{-L}, \dots, \mathbf{A}_L, \mathbf{b}]$ as shown in Fig 3. Although equation (1) uses equal context size before and after the current frame, the proposed method can be easily used in other context settings. Similar transform structure has been used in previous studies, such as DCMLLR [10].

Note that ST transform in (1) is the most general form of linear processing of features. Both fMLLR and temporal filters are special cases of (1). Specifically, when $L = 0$, ST transform is reduced to the fMLLR transform (see Fig. 4B), and when $\mathbf{A}_\tau$ is diagonal for all τ and the bias $\mathbf{b}$ is omitted, ST transform is reduced to temporal filter (see Fig. 4C). When $L = 0$ and $\mathbf{A}_0$ is diagonal, ST transform becomes the scalar form MVN (see Fig. 4A).

C. Review of Maximum Likelihood Based Feature Adaptation

Maximum likelihood (ML) criterion is widely used for estimating square feature transform parameters. In this section, we review the ML criterion and point out the difficulty of directly applying it for estimating rectangular transforms.

Without loss of generality, we take the feature transform for speech denoising as an example to illustrate the ML criterion. Let's assume that the noise corruption process can be approximated by an invertible linear transform, and the observed noisy feature vector can be represented as $\mathbf{y} = \mathcal{A}\mathbf{x}$ and $\mathbf{x}$ is the unobserved clean feature vector. If we know the probability density function (PDF) of the clean features $f(\mathbf{x})$, e.g. from an acoustic model, we can obtain the PDF of the corrupted features directly by applying the *change of variables* formula [67]: $g(\mathbf{y}) = f(\mathbf{x})|\det(d\mathbf{x}/d\mathbf{y})| = f(\mathbf{x})|\det(\mathcal{A}^{-1})|$, where $\mathbf{x} = \mathcal{A}^{-1}\mathbf{y}$. In practice, the inverse transform $\mathcal{A}^{-1}$ is estimated such that the likelihood of the observed noisy features is maximum when evaluated on $g(\mathbf{y})$. Then we can apply the estimated transform $\hat{\mathcal{A}}^{-1}$ to reverse the corruption process and adapt the noisy features towards the clean model by using $\hat{\mathbf{x}} = \hat{\mathcal{A}}^{-1}\mathbf{y}$. For square transform such as stochastic matching [41] and fMLLR [4], it is straightforward to apply the ML criterion to estimate the transform, as the determinant of a square matrix can be computed.

If the transform is not square, it won't be straightforward to directly use the ML framework as the determinant of the Jacobian matrix does not exist and the transform is not invertible. This is the case when we project a sequence of feature vectors to a single feature vector and the transform is a $D \times M$ matrix where $D < M$. In the studies of [65] and DCMLLR [10], instead of estimating a rectangular transform, a square transform of size $M \times M$ is estimated by using the ML criterion but the last $M - D$ rows of the transform are discarded. The discarded dimensions are modeled by a single Gaussian for all phone classes, hence all discriminative information are pushed into the first D dimensions of the projected space. The tying of all phone classes to a single Gaussian for discarded dimensions is conceptually the same as the tying used in HLDA [66], hence this kind of transform can be seen as the speaker dependent HLDA transform [65]. However, a potential drawback of this approach is that the computational cost of transform estimation may be high for large M , where most of the rows of the transform will be discarded.

To avoid estimating a full square transform, one can use a technique called Jacobian compensation which replaces the Jacobian term of the ML objective function with the determinant of the sample covariance matrix of the transformed features. This method has been used in transforms where no theoretical Jacobian can be computed. For example, it is used in vocal tract length normalization (VTLN) [68] to correct the systematic error in choosing the warping factor.

In the following section, we propose a new objective function for feature adaptation based on minimizing the Kullback-Leibler (KL) divergence between two distribution functions, i.e. the transformed features' sample distribution and the acoustic model. We will show that under certain assumptions,

Fig. 3. An illustration of the Spectro-Temporal feature transform, or ST transform, in Eq 1.

the KL divergence objective function will lead to the objective function of ML criterion with Jacobian compensation. We will also provide an expectation-maximization (EM) based method for iteratively estimating the ST transform parameters.

D. Minimum Kullback-Leibler (KL) Divergence Criterion

The ST transform $\mathbf{W}$ is estimated to minimize a KL divergence between the distribution of the transformed features, $p_{\mathbf{y}}$, and the reference distribution of the training features, parameterized as p_{Λ} . The reference distribution can be of any form, such as a GMM, to represent the distribution of the training data. We have the following KL divergence,

$$D_{\text{KL}}(p_{\mathbf{y}}||p_{\Lambda}) = \int_{\mathbf{y}} p_{\mathbf{y}}(\mathbf{y}) \log \frac{p_{\mathbf{y}}(\mathbf{y})}{p_{\Lambda}(\mathbf{y})} d\mathbf{y} = -H(p_{\mathbf{y}}) + H(p_{\mathbf{y}}, p_{\Lambda}) \quad (2)$$

where $H(p_{\mathbf{y}}) = -\int_{\mathbf{y}} p_{\mathbf{y}}(\mathbf{y}) \log(p_{\mathbf{y}}(\mathbf{y})) d\mathbf{y}$ is the entropy of distribution $p_{\mathbf{y}}$, and $H(p_{\mathbf{y}}, p_{\Lambda}) = -\int_{\mathbf{y}} p_{\mathbf{y}}(\mathbf{y}) \log p_{\Lambda}(\mathbf{y}) d\mathbf{y}$ is the cross entropy of $p_{\mathbf{y}}$ and p_{Λ} . As $\mathbf{y}$ is determined by $\mathbf{W}$ through (1), the transform matrix $\mathbf{W}$ can be estimated by minimizing $D_{\text{KL}}(p_{\mathbf{y}}||p_{\Lambda})$ to render the transformed features to have distribution closer to that of the reference.

For (2) to work, we need some run-time adaptation data from the test environment. If such data are limited, it is a challenge to reliably characterize the test environment. Hence, we choose to parameterize $p_{\mathbf{y}}(\mathbf{y})$ as a single Gaussian distribution with full covariance matrix. In this way, $H(p_{\mathbf{y}})$ can be shown to depend only on the determinant of the covariance matrix $\Sigma_{\mathbf{y}}$ as follows

$$H(p_{\mathbf{y}}) \approx K + \frac{1}{2} \log |\Sigma_{\mathbf{y}}| \quad (3)$$

where K is a constant. In practice, the covariance matrix is estimated from the available run-time adaptation data.

The second term $H(p_{\mathbf{y}}, p_{\Lambda})$ in (2) specifies the cross entropy between $p_{\mathbf{y}}$ and p_{Λ} . One way to evaluate this term is to use a Gaussian approximation of $p_{\mathbf{y}}$. This doesn't provide an easy solution to the optimization of KL divergence function. Instead, we use Monte Carlo approximation, i.e. we use the available adaptation feature vectors as random samples drawn from the distribution $p_{\mathbf{y}}$ and evaluate the cross-entropy by

$$H(p_{\mathbf{y}}, p_{\Lambda}) \approx -\sum_{t=1}^T \frac{1}{T} \log(p_{\Lambda}(\mathbf{y}_t)) \quad (4)$$

where the integration over $\mathbf{y}$ in (2) is replaced with summation over adaptation feature vectors $\mathbf{y}_t$ and T is the number of available adaptation feature vectors. The equation (4) is the negative average log likelihood of the transformed features evaluated in the reference feature distribution function.

With the approximation of $H(p_{\mathbf{y}})$ in (3) and $H(p_{\mathbf{y}}, p_{\Lambda})$ in (4), the KL divergence can be rewritten as

$$D_{KL}(p_{\mathbf{y}}||p_{\Lambda}) \approx -K - \frac{1}{2} \log |\Sigma_{\mathbf{y}}| - \frac{1}{T} \sum_{t=1}^T \log(p_{\Lambda}(\mathbf{y}_t)) \quad (5)$$

In this way, we are actually maximizing both the log likelihood of the transformed features on reference feature distribution as well as the log determinant of the transformed features covariance matrix.

The reference distribution p_{Λ} can be any distribution function of training speech features. Here, we use a GMM with diagonal covariance matrices for simplicity. A GMM of M Gaussians is defined by a set of parameters $\{c_m, \mu_m, \Sigma_m\}$, i.e. $p_{\Lambda}(\mathbf{y}_t) = \sum_{m=1}^M c_m \mathcal{N}(\mathbf{y}_t; \mu_m, \Sigma_m)$, where μ_m and Σ_m are the mean and diagonal covariance matrix of the m^{th} Gaussian component. The parameters of GMM can be estimated from training data. With this definition of $p_{\Lambda}(\mathbf{y}_t)$, the KL divergence in (5) can be rewritten as

$$D_{KL}(p_{\mathbf{y}}||p_{\Lambda}) \approx -K - \frac{1}{2} \log |\Sigma_{\mathbf{y}}| - \frac{1}{T} \sum_{t=1}^T \log \left(\sum_{m=1}^M c_m \mathcal{N}(\mathbf{y}_t; \mu_m, \Sigma_m) \right) \quad (6)$$

Applying (6), we propose to estimate ST transform by

$$\hat{\mathbf{W}} = \arg \min_{\mathbf{W}} f(\mathbf{W}) \quad (7)$$

where the cost function is defined as:

$$f(\mathbf{W}) = \frac{\beta}{2T} \|\mathbf{W} - \mathbf{W}_0\|_F^2 - \frac{\lambda}{2} \log |\Sigma_{\mathbf{y}}| - \frac{1}{T} \sum_{t=1}^T \log \left(\sum_{m=1}^M c_m \mathcal{N}(\mathbf{y}_t; \mu_m, \Sigma_m) \right) \quad (8)$$

The above cost function includes the KL divergence as well as a Frobenius matrix norm term (also called L_2 norm term) $\|\mathbf{W} - \mathbf{W}_0\|_F^2$ to regulate the transform. $\mathbf{W}_0$ is the initial parameters of the transform, in which $\mathbf{b}$ and $\mathbf{A}_{\tau}$ in (1) contain all zero's for $\tau \neq 0$ and $\mathbf{A}_0$ is the identity matrix. With this design, $\mathbf{W}_0 \tilde{\mathbf{x}}_t = \mathbf{x}_t$. With the L_2 term, the transformed features are ensured to be not too far from the initial features if $\mathbf{W}$ is near to $\mathbf{W}_0$ in the parameter space. The parameters β and λ are tunable and used to control the contributions of the Frobenius norm and data distribution $p_{\mathbf{y}}$ in the cost function, respectively. Note that if $\lambda = 1$, the term $\frac{1}{2} \log |\Sigma_{\mathbf{y}}|$ is the same as the Jacobian compensation used in VTLN [68].

The objective function in (8) contains two terms that pull the transform estimation in opposite directions. On one hand, the log likelihood term fits the transformed features to the means of Gaussians that are initially close to the observed features. Hence it will tend to shrink the variances of the transformed features and make $p_{\mathbf{y}}$ cover only a fraction of the acoustic space of p_{Λ} . On the other hand, the log determinant term is

TABLE I
ESTIMATION OF $\mathbf{W}$ TO MINIMIZE THE COST FUNCTION IN (8)

Step 1: Set $\bar{\mathbf{W}} = \mathbf{W}_0$.
Step 2: Compute statistics in (12) first, then (10) and (11).
Step 3: Estimate $\mathbf{W}$ to minimize $Q(\mathbf{W}, \bar{\mathbf{W}})$ in (9) using L-BFGS algorithm [69] with gradient defined in (13).
Step 4: If convergence is met or maximum number of iterations is reached, exit.
Otherwise set $\bar{\mathbf{W}} = \mathbf{W}$ and go to Step 2.

trying to spread $p_{\mathbf{y}}$ so it can cover a larger part of the acoustic space. The optimal solution of the transform is obtained when a balance is reached between these two forces.

E. EM Algorithm for Parameter Estimation

There is no closed form solution for the minimization problem in (8). We use an EM algorithm [70] to search for a locally optimal solution iteratively. The EM algorithm is an effective method for estimation problems with incomplete data such as (8). The auxiliary function of the EM algorithm for the cost function (8) can be written as follows:

$$\begin{aligned} Q(\mathbf{W}, \bar{\mathbf{W}}) &= \frac{\beta}{2T} \|\mathbf{W} - \mathbf{W}_0\|_2^2 - \frac{\lambda}{2} \log |\Sigma_{\mathbf{y}}| \\ &+ \frac{1}{T} \sum_{t=1}^T \sum_{m=1}^M \frac{\gamma_t(m)}{2} (\mathbf{y}_t - \mu_m)^T \Sigma_m^{-1} (\mathbf{y}_t - \mu_m) \\ &= \frac{\beta}{2T} \|\mathbf{W} - \mathbf{W}_0\|_2^2 - \frac{\lambda}{2} \log |\mathbf{W} \Sigma_{\tilde{\mathbf{x}}} \mathbf{W}^T| \\ &+ \frac{1}{2} \sum_{d=1}^D \mathbf{w}^{(d)} \mathbf{G}^{(d)} (\mathbf{w}^{(d)})^T - \sum_{d=1}^D \mathbf{w}^{(d)} \mathbf{p}^{(d)} \end{aligned} \quad (9)$$

where $\bar{\mathbf{W}}$ is the current estimate of transform parameters and $\mathbf{W}$ is the new transform parameters to be estimated. $\Sigma_{\tilde{\mathbf{x}}}$ is the covariance matrix of the stacked features $\tilde{\mathbf{x}}$. $\mathbf{w}^{(d)}$ is the d^{th} row of $\mathbf{W}$. Note that the derivation of (9) requires diagonal covariance matrices of $\Sigma_m = \text{diag}([\sigma_m^{(1)^2}, \dots, \sigma_m^{(D)^2}])$. Other statistics are defined as follows:

$$\mathbf{G}^{(d)} = \frac{1}{T} \sum_{t=1}^T \sum_{m=1}^M \frac{\gamma_t(m)}{\sigma_m^{(d)^2}} \tilde{\mathbf{x}}_t \tilde{\mathbf{x}}_t^T \quad (10)$$

$$\mathbf{p}^{(d)} = \frac{1}{T} \sum_{t=1}^T \sum_{m=1}^M \frac{\gamma_t(m)}{\sigma_m^{(d)^2}} \mu_m^{(d)} \tilde{\mathbf{x}}_t \quad (11)$$

$$\gamma_t(m) = \frac{c_m \mathcal{N}(\bar{\mathbf{W}} \tilde{\mathbf{x}}_t; \mu_m, \Sigma_m)}{\sum_{i=1}^M c_i \mathcal{N}(\bar{\mathbf{W}} \tilde{\mathbf{x}}_t; \mu_i, \Sigma_i)} \quad (12)$$

where $\gamma_t(m)$ is the occupation (posterior) probability of the m^{th} Gaussian at frame t and estimated in the E-step of the EM algorithm.

The gradient of the auxiliary function w.r.t. the d^{th} row of $\mathbf{W}$ is

$$\begin{aligned} \frac{\partial Q(\mathbf{W}, \bar{\mathbf{W}})}{\partial \mathbf{w}^{(d)}} &= -\lambda \mathbf{c}^{(d)} + \mathbf{w}^{(d)} \mathbf{G}^{(d)} - \mathbf{p}^{(d)T} + \frac{\beta}{T} (\mathbf{w}^{(d)} - \mathbf{w}_0^{(d)}) \\ \mathbf{C} &= (\mathbf{W} \Sigma_{\tilde{\mathbf{x}}} \mathbf{W}^T)^{-1} \mathbf{W} \Sigma_{\tilde{\mathbf{x}}}^T \end{aligned} \quad (13)$$

where $\mathbf{c}^{(d)}$ is the d^{th} row of $\mathbf{C}$. From the gradient, it is still difficult to obtain a closed form solution for the transform.

Hence, we use gradient based optimization method to obtain the solution for the M-step of the EM algorithm as summarized in Table I. In the M step of each EM iteration, we use L-BFGS [69] to minimize the auxiliary function.

F. Adaptive Training

Adaptive training is a scheme to perform feature adaptation when training the acoustic models [71], [72]. The training data are divided into homogeneous subsets, e.g. by speakers or acoustic conditions, and one or more transforms are estimated for each subset to adapt the features towards the model. The purpose of adaptive training is to reduce feature variation in training data and produce a sharper model.

It will be straightforward to apply the ST transforms to adaptive training. Similar to other adaptive training schemes, the acoustic model parameters and the transform parameters are updated alternatively until convergence.

G. A Unified View on Feature Processing

We now discuss a unified view on feature processing methods under the framework of ST transform. Table II summarizes a list of feature adaptation methods with reference to the proposed ST transform.

First, the proposed minimum KL divergence criterion for parameter estimation can be seen as a generalization of the ML criterion in fMLLR. For example, if we set the context size $L = 0$, the equation (1) becomes

$$\mathbf{y}_t = \mathbf{A}_0 \mathbf{x}_t + \mathbf{b} = \mathbf{W} \mathbf{x}_t \quad (14)$$

where $\mathbf{W} = [\mathbf{A}_0, \mathbf{b}]$ is identical to the fMLLR transform. In addition, the KL divergence approximation of equation (6) can be written as

$$D_{\text{KL}}(p_{\mathbf{y}} || p_{\Lambda}) \approx -K - \frac{1}{2} \log |\mathbf{A}_0 \Sigma_{\mathbf{x}} \mathbf{A}_0^T| - \frac{1}{T} \sum_{t=1}^T \log(p_{\Lambda}(\mathbf{y}_t)) \quad (15)$$

$$= -K' - \log |\mathbf{A}_0| - \frac{1}{T} \sum_{t=1}^T \log(p_{\Lambda}(\mathbf{y}_t)) \quad (16)$$

where we have used the property that $\Sigma_{\mathbf{y}} = \mathbf{A}_0 \Sigma_{\mathbf{x}} \mathbf{A}_0^T$ when $L = 0$. The divergence function in (16) is actually the negative of the log likelihood objective function of fMLLR. Hence, fMLLR is a special case of ST transform without using the contextual frames. With reference to the original fMLLR algorithm, the EM algorithm described in Section II-E has several different properties, i.e. tunable contribution from log determinant of linear transform matrix and the use of L_2 norm. The use of L_2 norm has similar effect as imposing a Gaussian prior distribution on the transform parameters and is expected to perform similarly with feature space maximum a posteriori linear regression (fMAPLR) [73].

Second, ST transform can also be seen as a generalization of temporal filters, such as MNLLF [33]. In MNLLF filters, the feature trajectories are filtered separately. In ST transform, all the feature trajectories are filtered simultaneously.

Fig. 4. Illustration of 5 special types of ST transform $\mathbf{W}$. Yellow cells denote free parameters to be estimated and empty cells represent parameters whose values are set to 0. A) scalar form transform, B) vector form transform, C) trajectory form transform, D) cross transform and E) full ST transform.

III. IMPLEMENTATION ISSUES

In this section, we will discuss several practical issues in implementing ST transform. In particular, the estimation of parameters given limited adaptation data. We will discuss 3 approaches to address this issue: 1) sparse ST transform; 2) cascaded transforms; 3) regularization and statistics smoothing.

A. Sparse Generalized Linear Transform

ST transform in its full capacity is characterized by a large set of parameters, that requires a large number of training data. For example, If $L = 10$, i.e. we use a context of 21 feature vectors centred at the current frame, there will be $39 \times 39 \times 21 + 39 = 31,980$ parameters. It is very difficult, if not impossible, to reliably estimate such a large amount of parameters from a few seconds of speech. Hence, it becomes necessary to limit the number of parameters in ST transform.

One way to limit the number of parameters is to force some parameters to be zero and make the transform matrices $\mathbf{A}_\tau$ sparse. From Eq. (1), each element of the adapted feature vector is a linear weighted sum of all the feature elements in the neighboring frames. It is reasonable to believe that some elements of the spectro-temporal context are more important than others for predicting a feature element. By setting the parameters of the less important context zero, we reduce the number of free parameters and estimate the effective parameters more reliably.

In this study, we consider 3 types of sparse transforms as illustrated in Fig. 4B, 4C and 4D. In the first simplification, we set the parameters of feature elements in neighboring frames to zero, i.e. $\mathbf{A}_\tau = \mathbf{0}$ except for $\tau = 0$ (see Fig. 4B) and Table II. In this way, only the spectral information of the

TABLE II

EIGHT SPECIAL CASES OF THE ST TRANSFORM IN (1) AND THE MINIMUM KL DIVERGENCE OBJECTIVE FUNCTION IN (8). TYPE "A" TO "E" IN SECOND COLUMN REFER TO THE TRANSFORM CONFIGURATIONS AS SHOWN IN FIG. 4. M IS THE NUMBER OF GAUSSIANS IN THE REFERENCE DISTRIBUTION GMM.

Methods	Type	Parameter Estimation and Constraints
CMN [24]	A	$\mathbf{A}_0 = \mathbf{I}, \lambda = 1, \beta = 0, M = 1$
MVN [25]	A	$\mathbf{A}_0$ is diag., $\lambda = 1, \beta = 0, M = 1$
Diag fMLLR	A	$\mathbf{A}_0$ is diag., $\lambda = 1, \beta = 0, M > 1$
fMLLR [4], [29]	B	$\mathbf{A}_0$ is full, $\lambda = 1, \beta = 0, M > 1$
MNLLF [33]	C	$\mathbf{A}_\tau$ is diag., Σ_y is diagonal, $M > 1$
Temporal transform	C	$\mathbf{A}_\tau$ is diag., Σ_y is full, $M > 1$
Vector transform	D	$\mathbf{A}_\tau = \mathbf{0}$ for $\tau \neq 0, \Sigma_y$ is full, $M > 1$
Cross transform	D	$\mathbf{A}_\tau$ is diag. for $\tau \neq 0, \Sigma_y$ is full, $M > 1$
Full ST transform	E	no constraint in (1) and (8)

current frame is used, and the number of free parameters is reduced to $D(D+1)$. This simplification of the ST transform turns out to be the popular fMLLR transform.

In the second simplification, we set the parameters of the feature trajectories other than the current one to zero, i.e. $\mathbf{A}_\tau$ is set to diagonal for $-L \leq \tau \leq L$ (see Fig. 4C) and Table II. In this case, only temporal information in the current feature trajectory is used for feature adaptation, and the number of free parameters is reduced to $D(2L+1)$. This simplification leads to temporal filtering of features. We have studied this type of sparse transform in [33] and found it useful for dealing with reverberation. We note that the two simplifications above do not make use of spectral and temporal information simultaneously.

From fMLLR, we know that single frame spectral information allows us to handle short term feature variations such as speaker variation and additive noise distortions. From temporal filter, we learn that the temporal trajectory of a feature element along the time axis removes long term variation such as reverberation. Hence, we propose the third sparse transform that benefits from the best of both fMLLR transform and temporal filter. In particular, we restrict $\mathbf{A}_\tau$ in (1) to be diagonal for $\tau \neq 0$ to capture the temporal information, while keeping $\mathbf{A}_0$ as a full matrix to incorporate the spectral information of current frame. With the new transform in Fig. 4D, the number of free parameters is reduced significantly, while both spectral and temporal information can be partially modeled. Specifically, the ratio of free parameters of the new design over the full ST transform is $\frac{2LD+D^2+D}{2LD^2+D^2+D} = \frac{2L+D+1}{2LD+D+1}$. For example, with $L = 10$ and $D = 39$, the ratio is $\frac{2 \times 10 + 39 + 1}{2 \times 10 \times 39 + 39 + 1} = \frac{60}{820} \approx 7\%$. Examining Fig. 1D and Fig. 4D, we find that such a transform practically applies a cross-shape mask on the features, hence, we call it the cross transform.

Compared to the full ST transform, the sparse transforms limit the number of parameters, therefore reduce the need of computation and memory during parameter estimation. As there are only a few non-zero elements in each row of the sparse transforms, the statistics in equation (10) and (11) is much smaller than those of full transforms. For example, each of $\mathbf{G}^{(d)}$ is a $M \times M$ matrix, where M is the number

of parameters used to predict feature element d . For full transform, $M = (2L+1)D$, so $\mathbf{G}^{(d)}$ can be a large matrix for larger L . For cross transform, $M = 2L+D$, hence it requires much less memory and computation for $\mathbf{G}^{(d)}$.

B. Cascaded Transform

The cross transform represents a way of spectro-temporal processing of features without a significant increase in the number of free parameters. To achieve a similar objective, one may consider carry out fMLLR and temporal filter one after another.

In the case where fMLLR is followed by a temporal filter, an element in the fMLLR output vector is the weighted sum of all elements in the input vector, an element of the temporal filter output vector is therefore the weighted sum of all elements across multiple frames within the context window of the temporal filter. This is also true if temporal filter is followed by fMLLR.

If we consider the full ST transform as a 2-dimensional filtering of the time-quefrequency² representation of the speech (e.g. cepstral features), then fMLLR and temporal filter can be seen as 1-dimensional filters, one along the quefrequency axis and the other along the time axis. Applying the two 1-dimensional filters in sequence will be effectively the same as applying a 2-dimensional filtering with its weight matrix having a rank of 1. The advantage of such cascaded transform is that it has a much fewer number of parameters and require much less memory and computation to estimate than the full ST transform. It would be interesting to see how such cascaded transforms perform against the full ST transform and cross transform.

C. Interpolation of Statistics

With a fewer number of parameters, cross transform and cascaded transform are expected to work better than full ST transform given a limited amount of data. However, in many applications, the adaptation data are the test sentence itself which is of several seconds in length. In such cases, it is even difficult to estimate the fMLLR or temporal filters. To address this issue, we apply L_2 norm regularization on the parameters and smooth the sufficient statistics in the EM algorithm. The L_2 norm regularization was already introduced in (8) as the Frobenius norm. Next we discuss how the statistics smoothing works.

The EM algorithm for transform estimation relies on several sufficient statistics such as the mean vector and covariance matrix of the input feature vectors, the $\mathbf{G}^{(d)}$ and $\mathbf{p}^{(d)}$ for all dimensions as defined in (10) and (11). Generally speaking, if the test environment is stable, more adaptation data will result in more reliable estimation of these statistics, which will in turn lead to better adapted features. The idea of statistics smoothing is to interpolate the statistics computed from the adaptation data with the statistics computed from some prior

²Assuming we are applying the transforms on cepstral features, one dimension of the feature representation is time and the other is quefrequency.

data in the following way:

$$\hat{\mathbf{G}}^{(d)} = \alpha \mathbf{G}^{(d,0)} + (1 - \alpha) \mathbf{G}^{(d)} \quad (17)$$

$$\hat{\mathbf{p}}^{(d)} = \alpha \mathbf{p}^{(d,0)} + (1 - \alpha) \mathbf{p}^{(d)} \quad (18)$$

where $\mathbf{G}^{(d,0)}$ and $\mathbf{p}^{(d,0)}$ are prior statistics computed from prior data; and $\mathbf{G}^{(d)}$ and $\mathbf{p}^{(d)}$ are statistics computed from adaptation/test data. The tunable parameter α is used to control the level of smoothing; i.e. if $\alpha = 0$, we ignore the prior statistics, while $\alpha = 1$ we ignore the contribution from the adaptation data. Similar statistics smoothing approach have been proposed for fMLLR in [74]–[76]. In addition, the mean and covariance matrix of extended features $\tilde{\mathbf{x}}$ can also be approximated as in [77]

$$\hat{\boldsymbol{\mu}}_{\tilde{\mathbf{x}}} = \alpha \boldsymbol{\mu}_{\tilde{\mathbf{x}}}^{(0)} + (1 - \alpha) \boldsymbol{\mu}_{\tilde{\mathbf{x}}} \quad (19)$$

$$\hat{\boldsymbol{\Sigma}}_{\tilde{\mathbf{x}}} = E(\tilde{\mathbf{x}}\tilde{\mathbf{x}}^T) - \hat{\boldsymbol{\mu}}_{\tilde{\mathbf{x}}} \hat{\boldsymbol{\mu}}_{\tilde{\mathbf{x}}}^T \quad (20)$$

$$E(\tilde{\mathbf{x}}\tilde{\mathbf{x}}^T) = \alpha E_{\tilde{\mathbf{x}}}^{(0)} + \frac{(1 - \alpha)}{T} \sum_{t=1}^T \tilde{\mathbf{x}}_t \tilde{\mathbf{x}}_t^T \quad (21)$$

where $\boldsymbol{\mu}_{\tilde{\mathbf{x}}}^{(0)}$ and $E_{\tilde{\mathbf{x}}}^{(0)}$ are the prior expected values of $\tilde{\mathbf{x}}$ and $\tilde{\mathbf{x}}\tilde{\mathbf{x}}^T$ and computed from the prior data. $\boldsymbol{\mu}_{\tilde{\mathbf{x}}}$ is the expected value of $\tilde{\mathbf{x}}$, computed from adaptation/test data. In practice, the prior data can be the training data, or development data that are from similar environment as the test data.

IV. EXPERIMENTS

A. Experimental Settings

1) *Task Description*: To evaluate the proposed transforms, we conduct experiments on the REVERB Challenge 2014 benchmark task for noisy and reverberant speech recognition [78].

In these experiments, we assume clean condition training, where only clean speech data are available at system training stage whereas test data are noisy and reverberant. We will only show the results on multi-condition training using cross transform. The clean training data consist of 7,861 clean utterances (about 17.5 hours from 92 speakers) from the WSJCAM0 database [79], while the multi-condition training data are the reverberation and noise corrupted version of the clean data. The clean data are recorded in a quiet room using a head-mounted close-talking microphone (Sennheiser HMD414-6). A triphone-based HMM/GMM acoustic model is used in the baseline system. The context-dependent triphone models are clustered into 3,115 tied states and 10 Gaussians are used to model the feature distribution of each tied state. Mel-frequency cepstral coefficients (MFCC) are used as acoustic features with utterancewise MVN postprocessing if not otherwise specified. Particularly, the first 13 (c0-c12) MFCCs and their first and second derivatives are extracted from each 25ms frame with 10ms hopping time. Hence, the frame rate is 100 frames per second. The features of every utterance are then normalized to zero mean and unit variances. The word error rate (WER) on clean test data is about 12.3%, which is the upper bound to achieve by adapting the features of reverberant speech.

The development (dev) and evaluation (eval) data sets are taken from actual meeting room recording of MC-WSJ-AV

[80]. The dev and eval sets are similar to each other in terms of noise and reverberation characteristics. Both dev and eval data are divided into two subsets according to the distance between the microphone and the speaker, i.e. near subset with a distance of 100cm and far subset with a distance of 250cm. The reverberation time T_{60} for the meeting room is about 0.7s. There are totally 179 utterances (about 0.3 hour and from 10 speakers) in the dev set and 372 utterances (about 0.6 hour and from 20 speakers) in the eval set. For multi-condition adaptive training, we also show results on simulated room 3 far distance test set, which contains 28 speakers and 362 utterances. For more details of the REVERB Challenge 2014 task, readers may refer to [78].

2) *Feature Adaptation Schemes*: We evaluate four types of linear transforms for feature adaptation against reverberation and noise distortions in ASR experiments, including fMLLR, temporal filter, cross transform, and full ST transform. We also test the cascading of these transforms, for example, “fMLLR $\circ$ Temporal” represents fMLLR followed by temporal filters. The $\circ$ operator is used to denote the fact that combination “fMLLR $\circ$ Temporal” will generate a ST transform whose transform matrix is of rank 1. These transforms are implemented under the same framework of the EM algorithm in Table I and estimated separately. The number of EM-iterations is set to 10 except for multi-condition training whose settings will be explained later. The Jacobian weight λ is set to 1 in all the experiments. The context size L is set to 10 for temporal, cross and full transforms unless otherwise stated. The L_2 norm weight β is tuned according to the development data.

We also carry out experiments with four different adaptation schemes, including full batch mode, speaker mode, utterance mode, and hybrid mode. In the full batch mode, one feature transform (e.g. temporal filter, fMLLR, cross or full ST transform) is estimated for each setting of the microphone distances. In the real eval set, each microphone distance contains about 180 utterances and the average utterance length is 7 seconds including silence. With that, we suppose that we have sufficient adaptation data to estimate most of the transforms under study.

In the speaker mode, one feature transform is estimated for each test speaker and distance combination. There are about 18 utterances for each speaker on average. Although the speaker mode has less adaptation data per transform than the full batch mode, it allows the adaptation of features to reduce speaker variations.

In the utterance mode, one feature transform is estimated for each test utterance. As each utterance is only about 7s long on average, it is a challenge to estimate most of the feature transforms. We don’t try the full ST transform here due to its large amount of parameters. An advantage of utterance mode processing is that the features can be adapted to address both speaker variation and small variations of reverberant distortions across utterances.

In the hybrid mode, we estimate the utterance based transform and smooth the sufficient statistics between the utterance based statistics and the full batch mode statistics. In this way, we expect more reliable estimation yet we are able to follow the reverberation change from utterance to utterance.

3) *Feature Adaptation Reference Model*: A reference model p_Λ in (5) is required to describe the distribution of the training features for the estimation of feature transform parameters. In theory, the HMM/GMM based acoustic model can be used as a reference model. However, it requires two-pass decoding: the first pass generates the hypotheses from which we obtain the Gaussian occupancy probabilities $\gamma_t(m)$ for estimating feature transforms, and the second pass generates the final recognition output using the transformed features. As decoding is time consuming, we choose to use a simple GMM as the reference model for feature adaptation. For GMM, the Gaussian occupancy probabilities can be readily computed efficiently once the feature vector is observed. This allows us to perform multiple iterations of EM algorithm, in each iteration the Gaussian occupancy probabilities are updated using the latest transformed features. A GMM reference model that contains 4,416 Gaussians is obtained by pooling the Gaussians from a monophone-based HMM/GMM model. As both the GMM and HMM/GMM are trained from the same clean training corpus, if the distribution of the transformed features matches the GMM well, it is also expected to match the HMM/GMM reasonably well. For speaker adaptive training (SAT), we choose to use the HMM/GMM acoustic model as the reference model.

B. Effect of Window Length L

One of the most important considerations in ST transform is the context window size which is equal to $2L + 1$. We report the effect of window size on the development data in Fig. 5. From the figure, we can see that all transforms that explicitly use temporal information, including temporal filter, cross transform, and the full ST transform, produces lower WER as the window size increases. At window length of 1 frame, the full transform, cross transform, and fMLLR have the same performance as they are practically the same. However, the temporal filter and MVN give different results when window size is 1 although they have the same scalar form. This is because MVN is equivalent to using a single Gaussian as a reference model, while temporal filter using a GMM of 4,416 Gaussians as reference model.

The performance saturates at around 21 frames or $L = 10$, across the board. The window length of 21 frames corresponds to about 0.3s of temporal information with a frame rate of 100Hz if we also include the temporal information in the dynamic features. For comparison, the fMLLR only uses temporal information up to 0.1s through the dynamic features. The results show that long term temporal information ($>0.1s$) is useful for feature adaptation to improve the robustness of ASR system against reverberation and noise distortions. Our results agree with that reported for DCMLLR [10] where upto 17 frames of context is found to produce consistent gain. In the following experiments, we will fix the window length to be 21 for temporal filter, cross transform, and the full ST transform.

C. Spectral, Temporal, and Spectro-Temporal Transforms

In the *full batch mode* adaptation, two instances of these transforms are estimated, one for near and one for far distance

Fig. 5. Effect of the input window length of the ST transforms on WER on dev set of REVERB Challenge 2014. All transforms are estimated in the full batch mode. Longer window size allows more temporal information to be used in feature adaptation.

microphone. Each distance has 186 utterances. The estimated transforms mainly normalize the reverberation effect. The results are reported in Table III.

We have two major observations from the results in the full batch mode. First, cross transform (63.4%) and the full ST transform (64.8%) perform significantly better than temporal filter (70.7%) and fMLLR (66.7%) because of the use of spectro-temporal information. Second, the full ST transform performs slightly worse than the cross transform (64.8% vs 63.4%). This could be due to the fact that the full ST transform is not as reliably estimated as the cross transform considering its large number of parameters.

In the *speaker mode* adaptation, one transform is estimated for each speaker. The results in the speaker mode show similar pattern as the full batch mode results, i.e. cross transform and the full ST transform produce better results than fMLLR and temporal filter. More interesting, the results of cross transform and the full ST transform in speaker mode are better than the ones in full batch mode even though the adaptation data in speaker mode are less than the ones in full batch mode. This observation suggests that estimating one transform for each speaker may help to remove speaker variation at the same time when removing reverberation distortion.

In the *utterance mode* adaptation, one transform is estimated for each utterance of average 7s in length. Utterance mode adaptation could be effective in the cases where acoustic environment changes from utterance to utterance. However, the estimation of the transforms are more challenging due to the limited adaptation data. In this experiment where the testing acoustic environment is relatively consistent across utterances, the utterance mode adaptation doesn't show an advantage over batch mode or speaker mode.

D. Experiments for Cascaded Transforms

As discussed in the previous section, combining a temporal filter and fMLLR in tandem is an alternative to exploiting

TABLE III
PERFORMANCE OF FEATURE ADAPTATION IN WER (%) FOR THE EVAL SET OF REVERB CHALLENGE 2014. THE FULL ST TRANSFORM IS NOT ESTIMATED IN UTTERANCE MODE DUE TO INSUFFICIENT ADAPTATION DATA. “NEAR” AND “FAR” DENOTE FOR THE NEAR AND FAR TEST SETS.

Methods		Full Batch Mode			Speaker Mode			Utterance Mode		
		Near	Far	Avg.	Near	Far	Avg.	Near	Far	Avg.
Utterancewise MVN		80.2	76.6	78.4	80.2	76.6	78.4	80.2	76.6	78.4
Spectral adaptation (fMLLR)		66.1	67.3	66.7	67.3	66.6	67.0	69.2	67.1	68.1
Temporal Filter		71.7	69.7	70.7	72.5	70.6	71.5	72.5	71.2	71.9
Spectro-Temporal	Cross Transform	63.6	63.2	63.4	62.7	63.2	63.0	65.9	65.6	65.8
	Full ST Transform	64.3	65.3	64.8	62.8	64.6	63.7	-	-	-

TABLE IV
PERFORMANCE OF CASCADED TRANSFORMS IN WER (%) FOR THE EVAL SET OF REVERB CHALLENGE 2014.

Methods	Full Batch Mode			Speaker Mode			Utterance Mode		
	Near	Far	Avg.	Near	Far	Avg.	Near	Far	Avg.
Temporal Filter $\circ$ fMLLR	63.9	64.3	64.1	63.0	63.0	63.0	66.2	65.1	65.7
fMLLR $\circ$ Temporal Filter	62.8	64.0	63.4	61.6	63.7	62.6	65.7	65.4	65.5
Cross Transform $\circ$ fMLLR	62.8	63.8	63.3	60.4	61.4	60.9	64.6	64.8	64.7
fMLLR $\circ$ Cross Transform	62.8	64.0	63.4	60.8	62.8	61.8	65.4	65.2	65.3

spectro-temporal information. In general, we can cascade any transform in different combinations to take advantage of the spectral or temporal properties of the transforms. However, we don’t cascade full ST transform with others because it already cover spectro-temporal information in a best effort. In this subsection, we investigate the performance of several cascaded transforms as shown in Table IV.

Overall, the cascaded transforms outperform individual transforms including fMLLR, temporal filter and cross transform. This suggests that cascading of transforms is also an effective way of using spectro-temporal information without significant increase in the number of free parameters. The different ways of combinations provide similar performance. The best results for all three modes (i.e. the full batch mode, the speaker mode and the utterance mode) are obtained by cascading cross transform and fMLLR.

Note that each adaptation mode offers a unique way of data processing, cascading two transforms in the same mode is not optimal. To leverage across different modes, we investigate a hybrid mode technique.

E. Hybrid Adaptation and Statistics Smoothing

In many practical applications, such as meeting transcription, the recordings are first diarized into speaker clusters and segmented into sentence-like units. In such case, it is useful to first apply a full batch mode feature adaptation to deal with session-wise reverberation and noise distortions, then use utterance mode adaptation to remove speaker variations and other sentence-wise variations, e.g. due to speaker movement and change of background noise. In this section, we adopt such a strategy. In addition, we use statistics smoothing to improve the robustness of feature transform estimation in the utterance mode. Specifically, the sufficient statistics computed from the current sentence is interpolated with that from the batch mode.

We summarize the results of the batch+utterance mode adaptation in Table V. The prefix “fb” and “utt” denote

the batch mode and the utterance mode, respectively. The prefix “smooth” denotes that statistics interpolation described in Section III-C is applied when estimating the corresponding transform. From Table V, we observe that the combination of batch and utterance mode transforms performs the best. For example, fb-Cross $\circ$ utt-fMLLR (60.3%) outperforms fb-Cross $\circ$ fb-fMLLR (63.3%) by 3% absolute in WER. In addition, the use of statistics smoothing provides further gain ranging from 1.4% to 3.0%. The best performance is obtained by fb-Cross $\circ$ smooth-utt-fMLLR (58.9%).

To further understand the hybrid mode, we plot the average log likelihood scores of the transformed features in Fig 6. The x-axis represents the number of EM-iterations and y-axis the log likelihood scores averaged over the eval set. The first 10 iterations are used to estimate the full batch mode transforms, while the second 10 iterations are for utterance mode fMLLR transforms. We observe that for the full batch model, cross transform achieves the highest likelihood, followed by full ST transform, with temporal filter being the last. Although the full ST transform is supposed to be more detailed than the cross transform, it achieves lower likelihood than the cross transform possibly due to the difficulty in optimizing the objective function with a large amount of parameters. In the second stage (utterance mode fMLLR), likelihood is improved significantly whenever the statistics smoothing is applied. This shows that statistics smoothing is very useful in reliable estimation of the fMLLR transform. It is also worth pointing out that the final likelihood achieved by each combination is highly correlated with the respective speech recognition performance. This shows that the proposed minimum KL divergence cost function is suitable for feature adaptation.

F. Combination of Feature Adaptation and Model Adaptation

In addition to feature adaptation, acoustic model adaptation is another important approach to reduce the mismatch between training and testing features. Model adaptation can be more

TABLE V

WER (%) OF HYBRID MODE ADAPTATION WITH STATISTICS SMOOTHING ON THE EVAL SET OF REVERB CHALLENGE 2014. PREFIX “FB” AND “UTT” DENOTE TRANSFORM ESTIMATED IN FULL BATCH MODE AND UTTERANCE MODE, RESPECTIVELY. “SMOOTH” DENOTES THE STATISTICS SMOOTHING METHOD IS APPLIED.

Methods	Near	Far	Avg.
Utterancewise MVN	80.2	76.6	78.4
smooth-utt-fMLLR	65.2	64.5	64.8
fb-Temporal $\circ$ fb-fMLLR	63.9	64.3	64.1
fb-Temporal $\circ$ utt-fMLLR	63.6	63.8	63.7
fb-Temporal $\circ$ smooth-utt-fMLLR	61.3	60.0	60.7
fb-Cross $\circ$ fb-fMLLR	62.8	63.8	63.3
fb-Cross $\circ$ utt-fMLLR	60.0	60.5	60.3
fb-Cross $\circ$ smooth-utt-fMLLR	59.7	58.2	58.9
fb-Full ST $\circ$ fb-fMLLR	62.0	64.4	63.2
fb-Full ST $\circ$ utt-fMLLR	61.1	61.6	61.4
fb-Full ST $\circ$ smooth-utt-fMLLR	59.5	60.3	59.9

Fig. 6. Log likelihood per frame averaged over the eval set. The first 10 EM-iterations are used for full batch mode transform estimation, while the last 10 iterations are for fMLLR transforms estimated in either full batch mode or utterance mode with and without statistics smoothing.

flexible than feature adaptation. For example, different phonetic classes can be adapted using different transforms as in multi-class CMLLR model adaptation. We are interested in whether the proposed feature adaptation method is complementary to the CMLLR model adaptation methods with 256 class-dependent linear transforms.

Table VI summarizes the results of combining feature adaptation and model adaptation. It can be observed that the CMLLR model adaptation alone reduces the WER to 66.1% from the MVN baseline of 78.4%. When the best feature adaptation configuration (58.9%) is also applied, a WER of 55.9% is obtained. This suggests that the ST transform based feature adaptation complements model adaptation to some extent. Note that the CMLLR with one class is equivalent to the full batch mode fMLLR. Hence, the complementary gain is from the fact that the CMLLR uses 256 linear transforms, one for each phonetic class.

TABLE VI

WER (%) ON THE EVAL SET BY COMBINING THE BEST ST TRANSFORM AND A 256-CLASS CMLLR MODEL ADAPTATION.

Methods	Near	Far	Avg.
Utterancewise MVN	80.2	76.6	78.4
CMLLR	66.3	65.9	66.1
fb-Cross $\circ$ smooth-utt-fMLLR	59.7	58.2	58.9
fb-Cross $\circ$ smooth-utt-fMLLR + CMLLR	55.5	56.3	55.9

G. Multi-condition Adaptive Training

Most practical ASR systems are trained with speech data collected from diverse acoustic conditions, and this is called the multi-condition training scheme. In this section, we will examine the effectiveness of the ST transforms in the multi-condition adaptative training scheme. Due to space limit of the paper, we only show the results of speaker-based cross transform to demonstrate the effectiveness of ST transform in adaptive training.

The adaptive training requires the use of HMM/GMM acoustic model as the reference model to estimate ST transforms. The procedures of adaptive training with ST transform is the same as that with fMLLR transforms or VTLN. We first initialize the SAT acoustic model with the multi-condition speaker independent (SI) model. Then we estimate the ST transforms for each training speaker and adapt the features towards the SAT model. After that, the SAT model is updated from the adapted features. The estimation of the ST transforms and the update of SAT model is carried out in alternate manner until convergence. During testing, the test speaker’s sentences are first decoded by the SI model to obtain the hypothesis used to estimate the first pass ST transform. A second pass decoding and ST transform estimation are carried out based on the features adapted by the first pass ST transform and the SAT model. The second pass ST transform is applied to transform the features for final decoding. We implemented the adaptive training procedures on the Kaldi [81] platform.

The WER on multi-condition training is shown in Table VIII. For comparison, we also performed SAT on clean models with the results listed in Table VII. Note that we are using speaker based CMN for feature preprocessing here. We also listed the most difficult simulated test case (far distance, room 3) of REVERB Challenge. From the two tables, using SAT models generally performs better than using SI models. For cross transform, we experimented with two SAT models, i.e. the one trained with fMLLR transform (SAT1) and the one trained with cross transform (SAT2). From the results, using cross transform during model training produces a small gain over using fMLLR for multi-condition scheme, but degrades the performance slightly for clean condition scheme. This could be due to the fact that it is not necessary to use context frames for feature adaptation in clean condition training as there is no reverberation. It is observed that the gain of cross transform over fMLLR becomes larger with SAT model than with SI model, especially for the multi-condition scheme.

TABLE VII
PERFORMANCE OF CROSS TRANSFORM IN WER (%) FOR THE EVAL SET USING CLEAN CONDITION TRAINING SCHEME. SAT1 MODEL IS TRAINED WITH FMLLR WHILE SAT2 MODEL IS TRAINED WITH CROSS TRANSFORM. "SIM" REFERS TO THE SIMULATED FAR ROOM 3 TEST CONDITION.

Model	Methods	Near	Far	Sim	Avg
SI	CMN	76.8	75.6	77.3	76.6
SI	fMLLR	67.3	66.5	65.0	66.3
SI	Cross	64.1	65.3	62.7	64.0
SAT1	fMLLR	63.1	62.9	65.1	63.7
SAT1	Cross	60.9	60.7	60.9	60.8
SAT2	Cross	60.5	60.9	62.5	61.3

TABLE VIII
PERFORMANCE OF CROSS TRANSFORM IN WER (%) FOR THE EVAL SET USING MULTI-CONDITION TRAINING SCHEME. SAT1 MODEL IS TRAINED WITH FMLLR WHILE SAT2 MODEL IS TRAINED WITH CROSS TRANSFORM.

Model	Methods	Near	Far	Sim	Avg
SI	CMN	51.4	51.2	43.7	48.8
SI	fMLLR	41.5	42.9	36.6	40.4
SI	Cross	41.0	42.5	34.7	39.4
SAT1	fMLLR	41.1	42.5	33.0	38.9
SAT1	Cross	40.1	41.6	31.7	37.8
SAT2	Cross	40.1	40.5	30.6	37.1

V. CONCLUSIONS

In this paper, we proposed a spectro-temporal transform for feature adaptation as well as a minimum KL divergence based criterion for estimating the transform parameters. The new feature adaptation technique makes use of both spectral and temporal information, hence is suitable for dealing with both spectral variability (e.g. speaker variation and noise distortions) and temporal variability (e.g. reverberation). We provide a unified perspective to a group of feature adaptation techniques that include fMLLR and temporal filters. To overcome insufficient training data, we proposed a sparse generalized transform, called cross transform, that has a reduced number of parameters than a full ST transform.

We conducted the experiments on REVERB Challenge 2014 benchmarking task where the clean and multi-condition trained acoustic models are tested on reverberant and noisy read speech recorded in meeting rooms. The experimental results confirmed that the spectro-temporal transform outperforms a spectral or temporal only adaptation, with cross transform achieving the best results. We have also explored alternative approaches to ST transform implementation, such as cascaded transform and interpolation of statistics. All experiments positively validated the idea of ST transform.

There are still many areas of feature adaptation that are worthy of further exploration. The cross transform is an empirically designed sparse transform. Another possibility is to discover a sparse transform automatically by using sparse constraints such as L_1 on the parameters used in [82]. This could reveal which time-frequency location is the most useful for predicting the feature at the current location. To fur-

ther improve the reliability of parameter estimation, subspace methods may be applied such as the one in [83] for fMLLR. Another direction is to further increase the flexibility of the transform without significantly increasing the number of free parameters. For example, we can introduce nonlinear hidden nodes into the transform, similar to a multilayer perceptron. Nonlinear processing may be necessary as the distortion in cepstral domain are generally nonlinear. Yet another direction is to model the training feature's distribution better. GMM only captures spectral and short term temporal information. Considering that the ST transform uses both spectral and temporal information, it is more reasonable to use a reference distribution that also captures these information of the training features.

REFERENCES

- [1] X. D. Huang, A. Acero, and H. W. Hon, *Spoken language processing: A guide to theory, algorithm, and system development*. Prentice-Hall, 2001.
- [2] G. E. Hinton, L. Deng, D. Yu, G. E. Dahl, A. Mohamed, N. Jaitly, A. Senior, V. Vanhoucke, P. Nguyen, T. Sainath, and B. Kingsbury, "Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups," *Signal Processing Magazine, IEEE*, vol. 29, no. 6, pp. 82–97, 2012.
- [3] J. L. Gauvain and C. H. Lee, "Maximum *a posteriori* estimation for multivariate Gaussian mixture observations of Markov chains," *IEEE Trans. Speech and Audio Processing*, vol. 2, no. 2, pp. 291–298, Apr. 1994.
- [4] M. J. F. Gales, "Maximum likelihood linear transformations for HMM-based speech recognition," *Computer Speech and Language*, vol. 12, pp. 75–98, 1998.
- [5] C. J. Leggetter and P. C. Woodland, "Maximum likelihood linear regression for speaker adaptation of continuous density hidden Markov models," *Computer Speech and Language*, vol. 9, no. 2, pp. 171–185, Apr. 1995.
- [6] K. Shinoda and C.-H. Lee, "A structural Bayes approach to speaker adaptation," *IEEE Trans. Speech Audio Process.*, vol. 9, no. 3, pp. 276–287, Mar 2001.
- [7] O. Siohan, C. Chesta, and C.-H. Lee, "Joint maximum a posteriori adaptation of transformation and HMM parameters," *IEEE Trans. Speech and Audio Processing*, vol. 9, no. 4, pp. 417–428, May 2001.
- [8] O. Siohan, T. A. Myrvoll, and C.-H. Lee, "Structural maximum a posteriori linear regression for fast HMM adaptation," *Computer Speech & Language*, vol. 16, no. 1, pp. 5 – 24, 2002.
- [9] M. Rahim and B. H. Juang, "Signal bias removal by maximum likelihood estimation for robust telephone speech recognition," *IEEE Trans. Speech and Audio Processing*, vol. 4, no. 1, pp. 19–, Jan 1996.
- [10] M. Gales and Y.-Q. Wang, "Model-based approaches to handling additive noise in reverberant environments," in *Joint Workshop on Hands-free Speech Communication and Microphone Arrays (HSCMA)*, May 2011, pp. 121–126.
- [11] 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 and Language*, vol. 23, no. 3, pp. 389–405, Jul. 2009.
- [12] A. Acero, L. Deng, T. Kristjansson, and J. Zhang, "HMM Adaptation Using Vector Taylor Series for Noisy Speech Recognition," in *Proc. ICSLP '00*, Beijing, China, Oct. 2000, pp. 869–872.
- [13] S. F. Boll, "Suppression of acoustic noise in speech using spectral subtraction," *IEEE Trans. Acoustics, Speech and Signal Processing*, vol. 27, no. 2, pp. 113–120, 1979.
- [14] J. S. Lim and A. V. Oppenheim, "Enhancement and bandwidth compression of noisy speech," *Proc. IEEE*, vol. 67, no. 12, pp. 1586–1604, Dec. 1979.
- [15] Y. Ephraim and D. Malah, "Speech enhancement using a minimum mean square error short time spectral amplitude estimator," *IEEE Trans. Acoustics, Speech and Signal Processing*, vol. 32, no. 6, pp. 1109–1121, Dec. 1984.
- [16] —, "Speech enhancement using a minimum mean square error Log-spectral amplitude estimator," *IEEE Trans. Acoustics, Speech and Signal Processing*, vol. 33, no. 2, pp. 443–445, Apr. 1985.

- [17] B. W. Gillespie and L. Atlas, "Strategies for improving audible quality and speech recognition accuracy of reverberant speech," in *Proc. ICASSP '03*, vol. 1, April 2003, pp. 1-676-1-679 vol.1.
- [18] J. Hopgood and P. Rayner, "Blind single channel deconvolution using nonstationary signal processing," *IEEE Trans. Speech and Audio Processing*, vol. 11, no. 5, pp. 476-488, Sept 2003.
- [19] T. Nakatani, T. Yoshioka, K. Kinoshita, M. Miyoshi, and B.-H. Juang, "Speech dereverberation based on variance-normalized delayed linear prediction," *IEEE Trans. Audio, Speech, and Language Processing*, vol. 18, no. 7, pp. 1717-1731, Sept 2010.
- [20] H. Kameoka, T. Nakatani, and T. Yoshioka, "Robust speech dereverberation based on non-negativity and sparse nature of speech spectrograms," in *Proc. ICASSP '09*, April 2009, pp. 45-48.
- [21] J. Erkelens and R. Heusdens, "Correlation-based and model-based blind single-channel late-reverberation suppression in noisy time-varying acoustical environments," *IEEE Trans. Audio, Speech, and Language Processing*, vol. 18, no. 7, pp. 1746-1765, Sept 2010.
- [22] K. Kumar, B. Raj, R. Singh, and R. Stern, "An iterative least-squares technique for dereverberation," in *Proc. ICASSP '11*, May 2011, pp. 5488-5491.
- [23] L. Deng, A. Acero, M. Plumpe, and X. D. Huang, "Large-vocabulary speech recognition under adverse acoustic environment," in *Proc. ICSLP '00*, Beijing, China, Oct. 2000, pp. 806-809.
- [24] B. S. Atal, "Effectiveness of linear prediction characteristics of the speech wave for automatic speaker identification and verification," *The Journal of the Acoustical Society of America*, vol. 55, no. 6, pp. 1304-1312, 1974. [Online]. Available: <http://scitation.aip.org/content/asa/journal/jasa/55/6/10.1121/1.1914702>
- [25] O. Viikki and K. Laurila, "Cepstral domain segmental feature vector normalization for noise robust speech recognition," *Speech Communication*, vol. 25, pp. 133-147, 1998.
- [26] S. Molau, F. Hilger, and H. Ney, "Feature space normalization in adverse acoustic conditions," in *Proc. ICASSP '03*, vol. 1, Hong Kong, China, Apr. 2003, pp. 656-659.
- [27] H. Hermansky and N. Morgan, "RASTA processing of speech," *IEEE Trans. Speech and Audio Processing*, vol. 2, no. 4, pp. 578-589, 1994.
- [28] H. Hermansky, N. Morgan, A. Bayya, and P. Kohn, "Compensation for the Effect of the Communication Channel in Auditory-Like Analysis of Speech (RASTA-PLP)," in *European Conference on Speech Communication and Technology*, 1991, pp. 1367-1370.
- [29] V. Digalakis, D. Rtischev, and L. G. Neumeyer, "Speaker adaptation using constrained estimation of Gaussian mixtures," *IEEE Trans. Speech and Audio Processing*, vol. 3, no. 5, pp. 357-366, 1995.
- [30] T. Yoshioka, A. Sehr, M. Delcroix, K. Kinoshita, R. Maas, T. Nakatani, and W. Kellermann, "Making Machines Understand Us in Reverberant Rooms: Robustness Against Reverberation for Automatic Speech Recognition," *Signal Processing Magazine, IEEE*, vol. 29, no. 6, pp. 114-126, Nov 2012.
- [31] J. Li, L. Deng, Y. Gong, and R. Haeb-Umbach, "An Overview of Noise-Robust Automatic Speech Recognition," *IEEE Trans. Audio, Speech, and Language Processing*, 2014.
- [32] J. Li, L. Deng, R. Haeb-Umbach, and Y. Gong, *Robust Automatic Speech Recognition - A Bridge to Practical Applications*, 1st ed. Elsevier, 2015.
- [33] X. Xiao, E. S. Chng, and H. Li, "Temporal filter design by minimum KL divergence criterion for robust speech recognition," in *Proc. ICASSP '13*, Vancouver, Canada, May 2013.
- [34] Y. H. Suk, S. H. Choi, and H. S. Lee, "Cepstrum third-order normalisation method for noisy speech recognition," *Electronics Letters*, vol. 35, no. 7, pp. 527-528, apr 1999.
- [35] C.-W. Hsu and L.-S. Lee, "Higher Order Cepstral Moment Normalization for Improved Robust Speech Recognition," *IEEE Trans. Audio, Speech, and Language Processing*, vol. 17, no. 2, pp. 205-220, Feb. 2009.
- [36] X. Xiao, J. Li, H. Li, and E. S. Chng, "Feature Normalization Using Structured Full Transforms for Robust Speech Recognition," in *Proc. INTERSPEECH '11*, Florence, Italy, Aug. 2011, pp. 693-696.
- [37] Y. Suh, M. Ji, and H. Kim, "Probabilistic class histogram equalization for robust speech recognition," *IEEE Signal Processing letters*, vol. 14, no. 4, pp. 287-290, 2007.
- [38] L. Garcia, C. Ortuzar, A. De la Torre, and J. C. Segura, "Class-Based Parametric Approximation to Histogram Equalization for ASR," *Signal Processing Letters, IEEE*, vol. 19, no. 7, pp. 415-418, July 2012.
- [39] X. Xiao, E. Chng, and H. Li, "Attribute-based histogram equalization (HEQ) and its adaptation for robust speech recognition," in *Proc. INTERSPEECH '13*. ISCA, 2013, pp. 876-880.
- [40] G. Saon, G. Zweig, and M. Padmanabhan, "Linear feature space projections for speaker adaptation," in *Proc. ICASSP '01*, vol. 1, 2001, pp. 325-328 vol.1.
- [41] A. Sankar and C.-H. Lee, "A maximum-likelihood approach to stochastic matching for robust speech recognition," *IEEE Trans. Speech and Audio Processing*, vol. 4, no. 3, pp. 190-202, May 1996.
- [42] C.-P. Chen and J. A. Billes, "MVA Processing of Speech Features," *IEEE Trans. Audio, Speech, and Language Processing*, vol. 15, no. 1, pp. 257-270, 2007.
- [43] X. Xiao, E. S. Chng, and H. Li, "Normalization of the Speech Modulation Spectra for Robust Speech Recognition," *IEEE Trans. Audio, Speech, and Language Processing*, vol. 16, no. 8, pp. 1662-1674, Nov. 2008.
- [44] T. Houtgast and H. J. M. Steeneken, "The Modulation Transfer Function in Room Acoustics as a Predictor of Speech Intelligibility," *Acustica*, vol. 28, pp. 66-73, 1973.
- [45] —, "A review of the MTF concept in room acoustics and its use for estimating speech intelligibility in auditoria," *Journal of the Acoustical Society of America*, vol. 77, no. 3, pp. 1069-1077, 1985.
- [46] R. Drullman, J. M. Festen, and R. Plomp, "Effect of temporal envelope smearing on speech reception," *Journal of the Acoustical Society of America*, vol. 95, no. 2, pp. 1053-1064, 1994.
- [47] N. Kanedera, T. Arai, H. Hermansky, and M. Pavel, "On the relative importance of various components of the modulation spectrum for automatic speech recognition," *Speech Communication*, vol. 28, no. 1, pp. 43-55, 1999.
- [48] T. Arai, M. Pavel, H. Hermansky, and C. Avendano, "Syllable intelligibility for temporally filtered LPC cepstral trajectories," *Journal of the Acoustical Society of America*, vol. 105, no. 5, pp. 2783-2791, 1999.
- [49] X. Xiao, E. S. Chng, and H. Li, "Joint Spectral and Temporal Normalization of Features for Robust Recognition of Noisy and Reverberated Speech," in *Proc. ICASSP '12*, Kyoto, Japan, Apr. 2012, pp. 4325-4328.
- [50] K. Kinoshita, M. Delcroix, T. Nakatani, and M. Miyoshi, "Suppression of late reverberation effect on speech signal using long-term multiple-step linear prediction," *IEEE Trans. Audio, Speech, and Language Processing*, vol. 17, no. 4, pp. 534-545, May 2009.
- [51] E. Ahissar, S. Nagarajan, M. Ahissar, A. Protopapas, H. Mahncke, and M. M. Merzenich, "Speech comprehension is correlated with temporal response patterns recorded from auditory cortex," *Proceedings of the National Academy of Sciences*, vol. 98, no. 23, pp. 13 367-13 372, 2001. [Online]. Available: <http://www.pnas.org/content/98/23/13367.abstract>
- [52] F. E. Theunissen, K. Sen, and A. J. Doupe, "Spectral-temporal receptive fields of nonlinear auditory neurons obtained using natural sounds," *The Journal of Neuroscience*, vol. 20, no. 6, pp. 2315-2331, 2000.
- [53] T. Chi, P. Ru, and S. A. Shamma, "Multiresolution spectrotemporal analysis of complex sounds," *The Journal of the Acoustical Society of America*, vol. 118, no. 2, pp. 887-906, 2005.
- [54] M. Kleinschmidt, D. Gelbart *et al.*, "Improving word accuracy with gabor feature extraction," in *INTERSPEECH*, 2002.
- [55] S. Ganapathy, S. H. Mallidi, and H. Hermansky, "Robust feature extraction using modulation filtering of autoregressive models," *IEEE/ACM Transactions on Audio, Speech and Language Processing (TASLP)*, vol. 22, no. 8, pp. 1285-1295, 2014.
- [56] L. Neumeyer and M. Weintraub, "Probabilistic optimum filtering for robust speech recognition," in *Proc. ICASSP '94*, Adelaide, Australia, Apr. 1994, pp. 417-420.
- [57] G. E. Hinton, S. Osindero, and Y. Teh, "A fast learning algorithm for deep belief nets," *Neural Computation*, pp. 1527-1554, 2006.
- [58] A. L. Maas, Q. V. Le, T. M. O'Neil, O. Vinyals, P. Nguyen, and A. Y. Ng, "Recurrent neural networks for noise reduction in robust asr," in *INTERSPEECH*, 2012.
- [59] F. Weninger, J. Geiger, M. Wöllmer, B. Schuller, and G. Rigoll, "Feature enhancement by deep lstm networks for asr in reverberant multisource environments," *Computer Speech & Language*, vol. 28, no. 4, pp. 888-902, 2014.
- [60] B. Li and K. C. Sim, "A spectral masking approach to noise-robust speech recognition using deep neural networks," *IEEE/ACM Transactions on Audio, Speech and Language Processing (TASLP)*, vol. 22, no. 8, pp. 1296-1305, 2014.
- [61] J. Du, Q. Wang, T. Gao, Y. Xu, L. Dai, and C.-H. Lee, "Robust speech recognition with speech enhanced deep neural networks," in *INTERSPEECH*, 2014.
- [62] Y. Xu, J. Du, L.-R. Dai, and C.-H. Lee, "A regression approach to speech enhancement based on deep neural networks," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 23, no. 1, pp. 7-19.

- [63] X. Xiao, S. Zhao, D. H. H. Nguyen, X. Zhong, D. L. Jones, E. S. Chng, and H. Li, "The NTU-ADSC systems for Reverberation Challenge 2014," in *Reverberation Challenge Workshop*, 2014.
- [64] K. Yao, D. Yu, L. Deng, and Y. Gong, "A fast maximum likelihood nonlinear feature transformation method for gmm-hmm speaker adaptation," *Neurocomputing*, vol. 128, pp. 145–152, 2014.
- [65] G. Saon, G. Zweig, and M. Padmanabhan, "Linear feature space projections for speaker adaptation," in *Acoustics, Speech, and Signal Processing, 2001. Proceedings.(ICASSP'01). 2001 IEEE International Conference on*, vol. 1. IEEE, 2001, pp. 325–328.
- [66] N. Kumar and A. G. Andreou, "Heteroscedastic discriminant analysis and reduced rank HMMs for improved speech recognition," *Speech Communication*, vol. 26, no. 4, pp. 283–297, 1998. [Online]. Available: <http://www.sciencedirect.com/science/article/pii/S0167639398000612>
- [67] A. Papoulis and S. U. Pillai, *Probability, random variables and stochastic processes*, 4th ed. McGraw Hill, 2002.
- [68] G. Saon, G. Zweig, B. Kingsbury, L. Mangu, and U. V. Chaudhari, "An architecture for rapid decoding of large vocabulary conversational speech," in *Proc. INTERSPEECH '03*, Geneva, Switzerland, Sep. 2003.
- [69] D. C. Liu and J. Nocedal, "On the limited memory BFGS method for large scale optimization," *Math. Program.*, vol. 45, no. 3, pp. 503–528, Dec. 1989.
- [70] A. P. Dempster, N. M. Laird, and D. B. Rubin, "Maximum likelihood from incomplete data via the EM algorithm," *Journal of Royal Statistical Society, Series B (Methodological)*, vol. 39, no. 1, pp. 1–38, 1977.
- [71] T. Anastasakos, J. McDonough, R. Schwartz, and J. Makhoul, "A compact model for speaker-adaptive training," in *Spoken Language, 1996. ICSLP 96. Proceedings., Fourth International Conference on*, vol. 2. IEEE, 1996, pp. 1137–1140.
- [72] M. J. Gales, "Cluster adaptive training of hidden markov models," *Speech and Audio Processing, IEEE Transactions on*, vol. 8, no. 4, pp. 417–428, 2000.
- [73] X. Lei, J. Hamaker, and X. He, "Robust Feature Space Adaptation For Telephony Speech Recognition," in *Proc. INTERSPEECH '06*, 2006.
- [74] S. Yoshizawa, A. Baba, K. Matsunami, Y. Mera, M. Yamada, and K. Shikano, "Unsupervised speaker adaptation based on sufficient hmm statistics of selected speakers," in *Proc. ICASSP '01*, vol. 1, 2001, pp. 341–344 vol.1.
- [75] Y. Li, H. Erdogan, Y. Gao, and E. Marcheret, "Incremental On-Line Feature Space MLLR Adaptation for Telephony Speech Recognition," in *Proc. ICSLP '02*, Denver, USA, Sep. 2002, pp. 1417–1420.
- [76] Z. Zajc, L. Machlica, and L. Miller, "Initialization of fmlr with sufficient statistics from similar speakers," in *Text, Speech and Dialogue*, ser. Lecture Notes in Computer Science, I. Habernal and V. Matouek, Eds. Springer Berlin Heidelberg, 2011, vol. 6836, pp. 187–194.
- [77] O. Viikki, D. Bye, and K. Laurila, "A recursive feature vector normalization approach for robust speech recognition in noise," in *Proc. ICASSP '98*, Seattle, WA, May 1998, pp. 733–736.
- [78] K. Kinoshita, M. Delcroix, T. Yoshioka, T. Nakatani, E. Habets, R. Haeb-Umbach, V. Leutnant, A. Sehr, W. Kellermann, R. Maas, S. Gannot, and B. Raj, "The REVERB Challenge: A Common Evaluation Framework for Dereverberation and Recognition of Reverberant Speech," in *Proceedings of the IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA-13)*, 2013.
- [79] T. Robinson, J. Fransen, D. Pye, J. Foote, and S. Renals, "WSJCAM0: a British English speech corpus for large vocabulary continuous speech recognition," in *Proc. ICASSP '95*, 1995, pp. 81–84.
- [80] M. Lincoln, I. McCowan, J. Vepa, and H. K. Maganti, "The multi-channel Wall Street Journal audio visual corpus (MC-WSJ-AV): specification and initial experiments," in *Proc. ASRU '05*, 2005, pp. 357–362.
- [81] D. Povey, A. Ghoshal, G. Boulianne, L. Burget, O. Glembek, N. Goel, M. Hannemann, P. Motlicek, Y. Qian, P. Schwarz, J. Silovsky, G. Stemmer, and K. Vesely, "The kaldi speech recognition toolkit," in *IEEE 2011 Workshop on Automatic Speech Recognition and Understanding*. IEEE Signal Processing Society, Dec. 2011, IEEE Catalog No.: CFP11SRW-USB.
- [82] J. Li, M. Yuan, and C.-H. Lee, "Lasso model adaptation for automatic speech recognition," in *Proc. ICML Workshop on Learning Architectures, Representations, and Optimization for Speech and Visual Information Processing*, 2011.
- [83] K.-T. Chen, W.-W. Liao, H.-M. Wang, and L.-S. Lee, "Fast Speaker Adaptation Using Eigenspace-Based Maximum Likelihood Linear Regression," in *International Conference on Spoken Language Processing*, 2000.

PLACE
PHOTO
HERE

Duc Hoang Ha Nguyen received his B.Eng and M.Sc degrees in University of Science, Ho Chi Minh city, Vietnam, in 2002 and 2006, respectively. From 2007 to 2009, he was a research assistant at Viskom, OEAW, Vienna, Austria. Since 2009, he has been pursuing his Ph.D at the School of Computer Engineering, Nanyang Technological University, Singapore. His research focuses on robust speech recognition.

PLACE
PHOTO
HERE

Xiong Xiao (S'06-M'10) received his B.Eng and Ph.D degrees in computer engineering from Nanyang Technological University (NTU) in 2004 and 2010, respectively. He joined Temasek laboratories @ NTU in 2009 and is now a senior research scientist. His research interests include robust speech processing, spoken document retrieval, and signal processing.

PLACE
PHOTO
HERE

Eng Siong Chng (SM'05) received the B.Eng. (honors) degree in electrical and electronics engineering and the Ph.D. degree from the University of Edinburgh, Edinburgh, U.K., in 1991 and 1996, respectively.

He is currently an Associate Professor in the School of Computer Engineering, Nanyang Technological University (NTU), Singapore. Prior to joining NTU in 2003, he was with the Institute of Physics and Chemical Research, Riken, as a Postdoctoral Researcher working in the area of

signal processing and classification (1996), the Institute of System Science (ISS, currently known as I²R) as a member of research staff to transfer the Apple-ISS speech and handwriting technologies to ISS (1996-1999), Lernout and Hauspie (now part of Nuance) as a Senior Researcher in speech recognition (1999-2000), and Knowles Electronics as a Manager for the Intellisonic microphone array research (2001-2002). His research interests are in pattern recognition, signal, speech, and video processing. His research interests are in pattern recognition, signal, speech and video processing. He has published over 50 papers in international journals and conferences. He is currently leading the speech and language technology program (<http://www3.ntu.edu.sg/home/aseschng/SpeechTechWeb/default.htm>) in Emerging Research Lab at the School of Computer Engineering, NTU.

PLACE
PHOTO
HERE

Haizhou Li (M'91-SM'01-F'14) received the B.Sc., M.Sc., and Ph.D degree in electrical and electronic engineering from South China University of Technology, Guangzhou, China in 1984, 1987, and 1990 respectively. Dr Li is currently the Principal Scientist, Department Head of Human Language Technology in the Institute for Infocomm Research (I²R), Singapore. He is also an adjunct Professor at the Nanyang Technological University, National University of Singapore and the University of New South Wales, Australia. His research interests in-

clude automatic speech recognition, speaker and language recognition, natural language processing, and computational intelligence.

Prior to joining I²R, he taught in the University of Hong Kong (1988-1990) and South China University of Technology (1990-1994). He was a Visiting Professor at CRIN in France (1994-1995), a Research Manager at the Apple-ISS Research Centre (1996-1998), a Research Director in Lernout & Hauspie Asia Pacific (1999-2001), and the Vice President in InfoTalk Corp. Ltd. (2001-2003).

Dr Li is currently the Editor-in-Chief of IEEE/ACM Transactions on Audio, Speech and Language Processing (2015-2017), a Member of the Editorial Board of Computer Speech and Language (2012-2016), the President of the International Speech Communication Association (2015-2017), and the President of Asia Pacific Signal and Information Processing Association (2015-2016). He was an elected Member of IEEE Speech and Language Processing Technical Committee (2013-2015), the General Chair of ACL 2012, and INTERSPEECH 2014.

Dr Li is a Fellow of the IEEE. He was a recipient of the National Infocomm Award 2002 and the President's Technology Award 2013 in Singapore. He was named one the two Nokia Visiting Professors in 2009 by the Nokia Foundation.