Fast Scoring for Mixture of PLDA in I-Vector/PLDA Speaker Verification

Man-Wai Mak
Center for Signal Processing, Dept. of Electronic and Information Engineering
The Hong Kong Polytechnic University, Hong Kong SAR
E-mail: enmwmak@polyu.edu.hk

***Abstract*—With the ubiquitous of mobile phones, users of speaker verification systems will perform authentication anywhere at anytime. As a result, practical speaker verification systems need to deal with utterances of different noise levels. Recently, an SNR-dependent mixture of PLDA model was proposed to deal with such practical situation. However, the scoring function of this model is significantly more complex than the conventional one. This paper proposes a method to reduce the computation burden of this mixture PLDA model. The idea is based on the observation that for most utterances, the posterior probabilities of SNR are very sparse so that it is possible to consider the top Gaussian only during scoring. The method effectively reduces the computational complexity from $\mathcal{O}(K^2D^3)$ to $\mathcal{O}(D^3)$, where K and D are the number of mixtures and i-vector dimension, respectively. Experimental results based on NIST 2012 SRE suggest that the proposed method can reduce computation time by 60% with very minor degradation in performance.**

I. Introduction

In the past decades, a number of methods have been proposed to reduce the effect of background noise in speaker verification systems. Some of these methods address the problem in the front-end or during feature extraction stage, e.g., [1]–[4]. There were also attempts to use speech enhancement techniques to reduce the effect of noise [5]. While the effectiveness of these feature-based approaches has been demonstrated, recent researches have found that techniques that operate on the backend classification stage are more promising. Among them, the joint factor analysis (JFA) [6] and i-vector/PLDA framework [7], [8] have been by far the most successful.

The i-vector/PLDA framework comprises two stages of factor analyses and dimension reduction. In the first stage, the acoustic characteristics of an utterance is represented by a low-dimensional vector called the i-vector that lives in the subspace of the GMM-supervector space [9] that is formed by stacking the mean vectors of a universal background model (UBM) [10]. In other words, the acoustic variabilities of utterances are modelled by a factor analyzer in which the latent space is of much lower dimension than the GMM-supervector space. Given an utterance, its spectral features (typically MFCC) are aligned with the UBM in a frame-by-frame basis. Then, the posterior probabilities (also known as *responsibility*) of individual mixtures in the UBM are estimated to compute the zero- and first-order sufficient statistics. Based on the sufficient statistics, the posterior density of the latent factors of the factor analysis model is computed and the posterior mean is considered as the i-vector. The space in which the i-vector lives is called the total variability space [7], [11].

Because the total variability space accounts for both speaker and other variabilities – such as channel, reverberation, and noise – a second stage of dimension reduction and normalization is required to suppress the channel effects. State-of-the-art speaker verification systems typically use a supervised factor analyzer called probabilistic linear discriminant analysis (PLDA) [12] to further suppress these variabilities. Some systems [7], [13], [14] also apply linear discriminant analysis (LDA) [15] and within-class covariance normalization (WCCN) [16] to reduce the dimension of i-vectors and to normalise their covariance before applying PLDA. It has been found that vector-length normalization is a simple but effective way to make the i-vectors more amendable to Gaussian PLDA modelling [17].

Recent methods to address noise robustness in speaker verification systems are typically built on top of the i-vector/PLDA framework. For example, in [18]–[21], multi-condition training was applied. Clean and noisy utterances are pooled together to train a PLDA model so that it becomes more robust to noisy test utterances. In [22], multiple PLDA models are trained, one for each condition. Hasan and Hansen [23] performed mixture of probabilistic PCA on feature space so that the posterior means of the mixture-dependent acoustic factors can be incorporated into an i-vector extractor. This idea has been further enhanced by replacing the UBM by a mixture of acoustic factor analyzers for i-vector extraction [24]. Recently, Lei et al. [25] proposed adapting a clean UBM to noisy utterances using vector Taylor series. I-vectors are then extracted based on the noise-adapted UBM. The idea is to clean up the i-vectors so that they become independent of additive and convolutive noise. Li and Mak [26] proposed an SNR-invariant PLDA model by introducing an SNR factor and an SNR subspace to the conventional PLDA model. The results show that the SNR factor is very effective in suppressing the variability in i-vectors caused by SNR variations in the utterances.

II. Mixture of SNR-Dependent PLDA

In most practical situations, a speaker verification system needs to deal with utterances of different noise levels because users may use the system in different acoustic environments,

e.g., offices, streets, restaurants, and subway stations, etc. As a result, the utterances received by the system may have a wide range of SNR. To tackle the varying noise levels, Mak [27] argued that the SNR of utterances should be divided into a number of regions so that the utterances in each region can be modelled more accurately by an SNR-dependent PLDA model. Based on this idea, Mak [27] proposed a mixture model called SNR-dependent mixture of PLDA or mPLDA in short.

A. Model Parameters

In mPLDA, i-vectors are modelled by a mixture of SNR-dependent factor analyzers with parameters

$$\begin{aligned}\underline{\boldsymbol{\theta}} &= \{\underline{\boldsymbol{\lambda}}, \underline{\boldsymbol{\omega}}\} = \{\boldsymbol{\lambda}_k, \boldsymbol{\omega}_k\}_{k=1}^K \\ &= \{\pi_k, \mu_k, \sigma_k, \mathbf{m}_k, \mathbf{V}_k, \boldsymbol{\Sigma}_k\}_{k=1}^K,\end{aligned} \tag{1}$$

where $\boldsymbol{\lambda}_k = \{\pi_k,\ \mu_k,\ \sigma_k\}$ contains the prior probability, mean and standard deviation of the SNR in the k-th group, and $\boldsymbol{\omega}_k = \{\mathbf{m}_k, \mathbf{V}_k, \boldsymbol{\Sigma}_k\}$ comprises the mean i-vector, factor loading matrix, and residual covariance of the k-th factor analyzer corresponding to the k-th SNR group. The EM formulations for estimating $\underline{\boldsymbol{\theta}}$ in Eq. 1 can be found in [27].

B. Mixture Alignments

Denote y_k's as the indicator variables specifying which of the factor analyzers is responsible for generating the i-vector $\mathbf{x}$, and denote ℓ as the SNR of the corresponding utterance. Then, the posterior probability of y_k is

$$\gamma_\ell(y_k) \equiv P(y_k = 1|\ell, \underline{\boldsymbol{\lambda}}) = \frac{\pi_k \mathcal{N}(\ell|\mu_k, \sigma_k^2)}{\sum_{k'=1}^K \pi_{k'} \mathcal{N}(\ell|\mu_{k'}, \sigma_{k'}^2))}. \tag{2}$$

Eq. 2 implies that the alignments of i-vectors in the mixture model are purely based on the posterior probabilities of SNR. This property gives the SNR-dependent mixture of PLDA an advantage over the conventional mixture of factor analysis [28] in that the i-vector clusters obtained by the EM algorithm is more prominent.

C. Likelihood Ratio Scores

Given target-speaker's i-vector $\mathbf{x}_s$ and test i-vector $\mathbf{x}_t$ and the SNR ℓ_s and ℓ_t (in dB) of the corresponding utterances, the same-speaker marginal likelihood is

$$\begin{aligned}&p(\mathbf{x}_s, \mathbf{x}_t, \ell_s, \ell_t|\text{same-speaker}) \\ &= p(\ell_s)p(\ell_t)p(\mathbf{x}_s, \mathbf{x}_t|\ell_s, \ell_t, \text{same-speaker}) \\ &= p_{st} \sum_{k_s=1}^K \sum_{k_t=1}^K \int p(\mathbf{x}_s, \mathbf{x}_t, y_{k_s}=1, y_{k_t}=1, \mathbf{z}|\underline{\boldsymbol{\theta}}, \ell_s, \ell_t)\mathrm{d}\mathbf{z} \\ &= p_{st} \sum_{k_s=1}^K \sum_{k_t=1}^K \gamma_{\ell_s,\ell_t}(y_{k_s}, y_{k_t}) \\ &\quad \times \int p(\mathbf{x}_s, \mathbf{x}_t|y_{k_s}=1, y_{k_t}=1, \mathbf{z}, \underline{\boldsymbol{\omega}})p(\mathbf{z})\mathrm{d}\mathbf{z} \\ &= p_{st} \sum_{k_s=1}^K \sum_{k_t=1}^K \gamma_{\ell_s,\ell_t}(y_{k_s}, y_{k_t}) \\ &\quad \times \mathcal{N}\left(\begin{bmatrix}\mathbf{x}_s^\mathsf{T} & \mathbf{x}_t^\mathsf{T}\end{bmatrix}^\mathsf{T} | \begin{bmatrix}\mathbf{m}_{k_s}^\mathsf{T} & \mathbf{m}_{k_t}^\mathsf{T}\end{bmatrix}^\mathsf{T}, \hat{\mathbf{V}}_{k_sk_t}\hat{\mathbf{V}}_{k_sk_t}^\mathsf{T} + \hat{\boldsymbol{\Sigma}}_k\right)\end{aligned}$$

where $p_{st} = p(\ell_s)p(\ell_t)$, $\hat{\mathbf{V}}_{k_sk_t} = [\mathbf{V}_{k_s}^\mathsf{T}\ \ \mathbf{V}_{k_t}^\mathsf{T}]^\mathsf{T}$, $\hat{\boldsymbol{\Sigma}}_k = \mathrm{diag}\{\boldsymbol{\Sigma}_{k_s}, \boldsymbol{\Sigma}_{k_t}\}$ and

$$\begin{aligned}&\gamma_{\ell_s,\ell_t}(y_{k_s}, y_{k_t}) \equiv P(y_{k_s}=1, y_{k_t}=1|\ell_s, \ell_t, \underline{\boldsymbol{\lambda}}) \\ &= \frac{\pi_{k_s}\pi_{k_t}\mathcal{N}([\ell_s\ \ell_t]^\mathsf{T}|[\mu_{k_s}\ \mu_{k_t}]^\mathsf{T}, \mathrm{diag}\{\sigma_{k_s}^2, \sigma_{k_t}^2\})}{\sum_{k'_s=1}^K \sum_{k'_t=1}^K \pi_{k'_s}\pi_{k'_t}\mathcal{N}([\ell_s\ \ell_t]^\mathsf{T}|[\mu_{k'_s}\ \mu_{k'_t}]^\mathsf{T}, \mathrm{diag}\{\sigma_{k'_s}^2, \sigma_{k'_t}^2\})}.\end{aligned} \tag{3}$$

Similarly, the different-speaker marginal likelihood is

$$p(\mathbf{x}_s, \mathbf{x}_t, \ell_s, \ell_t|\text{diff-speaker}) = p(\mathbf{x}_s, \ell_s|\text{Spk } s)p(\mathbf{x}_t, \ell_t|\text{Spk } t),$$

where

$$\begin{aligned}p(\mathbf{x}_s, \ell_s|\text{Spk } s) &= p(\ell_s)\sum_{k_s=1}^K \int p(\mathbf{x}_s, y_{k_s}=1, \mathbf{z}|\underline{\boldsymbol{\theta}}, \ell_s)\mathrm{d}\mathbf{z} \\ &= p(\ell_s)\sum_{k_s=1}^K \gamma_{\ell_s}(y_{k_s})\mathcal{N}\left(\mathbf{x}_s|\mathbf{m}_{k_s}, \mathbf{V}_{k_s}\mathbf{V}_{k_s}^\mathsf{T} + \boldsymbol{\Sigma}_{k_s}\right),\end{aligned}$$

and similarly for $p(\mathbf{x}_t, \ell_t|\text{Spk } t)$. Therefore, the likelihood ratio S_{mPLDA} is given by Eq. 4 at the bottom of next page.

Note that Eq. 4 is likely to cause numerical problems if they are evaluated directly because the determinant of $\hat{\mathbf{V}}_{k_s}\hat{\mathbf{V}}_{k_s}^\mathsf{T} + \hat{\boldsymbol{\Sigma}}_{k_s}$ could exceed the double-precision representation. This problem, however, can be avoided by computing the logarithm of determinant and noting the identity: $|\alpha\mathbf{A}| = \alpha^D|\mathbf{A}|$, where α is a scalar and $\mathbf{A}$ is a $D \times D$ matrix.. Thus, we can rewrite Eq. 4 as Eq. 5 shown at the bottom of next page, where $\hat{\boldsymbol{\Lambda}}_{k_sk_t} = \hat{\mathbf{V}}_{k_s}\hat{\mathbf{V}}_{k_t}^\mathsf{T} + \hat{\boldsymbol{\Sigma}}_{k_sk_t}$, $\boldsymbol{\Lambda}_{k_s} = \mathbf{V}_{k_s}\mathbf{V}_{k_s}^\mathsf{T} + \boldsymbol{\Sigma}_{k_s}$, $\hat{\boldsymbol{\Sigma}}_{k_sk_t} = \mathrm{diag}\{\boldsymbol{\Sigma}_{k_s}, \boldsymbol{\Sigma}_{k_t}\}$, and

$$\mathcal{D}(\mathbf{a}\|\mathbf{b}) = (\mathbf{a}-\mathbf{b})^\mathsf{T}\mathbf{S}_a^{-1}(\mathbf{a}-\mathbf{b}), \tag{7}$$

where $\mathbf{S} = \mathrm{cov}(\mathbf{a}, \mathbf{a})$. In this work, $\alpha = 5$. Note that because Eq. 5 is derived from Bayes' rule, mPLDA does not require score normalization.

D. Complexity Analysis

Note that the determinants in Eq. 5 can be pre-computed, so as the covariance matrices $\hat{\boldsymbol{\Lambda}}_{k_sk_t}$ and $\boldsymbol{\Lambda}_{k_s}$. As a result, the major computation burden in the scoring function lies in the computation of Mahalanobis distance $\mathcal{D}(\cdot\|\cdot)$. More precisely, the computational complexity of the numerator and denominator of Eq. 5 are $\mathcal{O}(K^2(2D)^3)$ and $\mathcal{O}(KD^3)$, respectively, where D is the dimensionality of i-vectors (after LDA+WCCN in our case) and K is the number of mixtures. Therefore, the overall complexity is $\mathcal{O}(K^2D^3)$. Table I summarises the computational complexity of the three scoring methods.

TABLE I
COMPUTATIONAL COMPLEXITY OF THE LIKELIHOOD-RATIO SCORING FUNCTION IN PLDA, MPLDA, AND FAST MPLDA.

Method	Computational Complexity
PLDA	$\mathcal{O}(D^3)$
mPLDA	$\mathcal{O}(K^2D^3)$
Fast mPLDA	$\mathcal{O}(D^3)$

III. Fast Scoring for mPLDA

When comparing with the computational complexity of the original PLDA ($\mathcal{O}(D^3)$), the computational complexity of mPLDA is K^2 times as much. Our results suggest that this will be a burden of mPLDA when K is larger than 2. It is therefore, imperative to find a method to reduce the scoring complexity of mPLDA.

A. Sparseness Analysis of SNR Posteriors

In Eq. 5, it is necessary to evaluate the likelihood for each combination of k_s and k_t. This is the major computation burden in mPLDA, especially when the number of mixture is large. If the posterior probabilities of SNR $[\gamma_{\ell_s,\ell_t}(y_{k_s}, y_{k_t})]$ are sparse, we may drop the combinations of (k_s, k_t) that lead to small posterior $\gamma_{\ell_s,\ell_t}(y_{k_s}, y_{k_t})$ when computing the likelihood.

Fig. 1(a) shows the average SNR posteriors for $K = 3$, sorted in descending order of $\gamma_{\ell_s,\ell_t}(y_{k_s}, y_{k_t})$. There are totally 9 combinations of (k_s, k_t) when $K = 3$. Evidently, the maximum average posterior dominates among the 9 combinations and is significantly larger than the first runner-up. Fig. 1(b) shows the individual posterior probabilities of SNR for 150 combinations of target-speaker utterances' SNR (ℓ_s) and test-utterances' SNR (ℓ_t). The figure further confirms the dominance of the winner among the 9 combinations.

B. Scoring Function

In the extreme case, we may only keep the combination that leads to maximum posterior. Based on this idea, Eq. 4 reduces to Eq. 6 at the bottom of this page, where

$$(k_s, k_t) = \arg \max_{(k_s,k_t)} \gamma_{\ell_s,\ell_t}(y_{k_s}, y_{k_t}).$$

A comparison between Eq. 6 and Eq. 5 reveals that the complexity has been reduced by K^2 times.

Eq. 6 can be written as[1]

$$\log S_{\text{mPLDA}}(\mathbf{x}_s, \mathbf{x}_t) = \log \left\{ \frac{\gamma_{k_s k_t} \mathcal{N}\left(\begin{bmatrix}\mathbf{x}_s \\ \mathbf{x}_t\end{bmatrix} \middle| \begin{bmatrix}\mathbf{m}_{k_s} \\ \mathbf{m}_{k_t}\end{bmatrix}, \begin{bmatrix}\mathbf{\Phi}_{k_s} & \mathbf{\Psi}_{k_s k_t} \\ \mathbf{\Psi}_{k_t k_s} & \mathbf{\Phi}_{k_t}\end{bmatrix} \right)}{\gamma_{k_s}\gamma_{k_t} \mathcal{N}\left(\begin{bmatrix}\mathbf{x}_s \\ \mathbf{x}_t\end{bmatrix} \middle| \begin{bmatrix}\mathbf{m}_{k_s} \\ \mathbf{m}_{k_t}\end{bmatrix}, \begin{bmatrix}\mathbf{\Phi}_{k_s} & \mathbf{0} \\ \mathbf{0} & \mathbf{\Phi}_{k_t}\end{bmatrix} \right)} \right\} \quad (8)$$

[1]To simplify notation, hereafter, we define $\gamma_{k_s k_t} \equiv \gamma_{\ell_s,\ell_t}(y_{k_s}, y_{k_t})$, $\gamma_{k_s} \equiv \gamma_{\ell_s}(y_{k_s})$, and $\gamma_{k_t} \equiv \gamma_{\ell_t}(y_{k_t})$.

where

$$\begin{aligned} \mathbf{\Phi}_{k_s} &= \mathbf{V}_{k_s}\mathbf{V}_{k_s}^\mathsf{T} + \mathbf{\Sigma}_{k_s} \\ \mathbf{\Psi}_{k_s k_t} &= \mathbf{V}_{k_s}\mathbf{V}_{k_t}^\mathsf{T} \\ \mathbf{\Psi}_{k_t k_s} &= \mathbf{V}_{k_t}\mathbf{V}_{k_s}^\mathsf{T} \\ \mathbf{\Phi}_{k_t} &= \mathbf{V}_{k_t}\mathbf{V}_{k_t}^\mathsf{T} + \mathbf{\Sigma}_{k_t}. \end{aligned} \quad (9)$$

Using block matrix inversion [29], the log-likelihood ratio can be written as

$$\begin{aligned} &\log S_{\text{mPLDA}}(\mathbf{x}_s, \mathbf{x}_t) \\ &= \log \gamma_{k_s k_t} - \log \gamma_{k_s} - \log \gamma_{k_t} \\ &+ \frac{1}{2}\begin{bmatrix}\mathbf{x}_s \\ \mathbf{x}_t\end{bmatrix}^\mathsf{T} \begin{bmatrix}\mathbf{Q}_{k_s k_t} & \mathbf{P}_{k_s k_t} \\ \mathbf{P}_{k_s k_t}^\mathsf{T} & \mathbf{Q}_{k_t k_s}\end{bmatrix} \begin{bmatrix}\mathbf{x}_s \\ \mathbf{x}_t\end{bmatrix} \\ &+ \begin{bmatrix}\mathbf{x}_s \\ \mathbf{x}_t\end{bmatrix}^\mathsf{T} \begin{bmatrix}\mathbf{Q}_{k_s k_t} & \mathbf{P}_{k_s k_t} \\ \mathbf{P}_{k_s k_t}^\mathsf{T} & \mathbf{Q}_{k_t k_s}\end{bmatrix} \begin{bmatrix}\mathbf{m}_{k_s} \\ \mathbf{m}_{k_t}\end{bmatrix} \\ &- \begin{bmatrix}\mathbf{m}_{k_s} \\ \mathbf{m}_{k_t}\end{bmatrix}^\mathsf{T} \begin{bmatrix}\mathbf{Q}_{k_s k_t} & \mathbf{P}_{k_s k_t} \\ \mathbf{P}_{k_s k_t}^\mathsf{T} & \mathbf{Q}_{k_t k_s}\end{bmatrix} \begin{bmatrix}\mathbf{m}_{k_s} \\ \mathbf{m}_{k_t}\end{bmatrix} \\ &- \frac{1}{2}\log|\mathbf{D}_1| + \frac{1}{2}\log|\mathbf{D}_2| \end{aligned} \quad (10)$$

where

$$\begin{aligned} \mathbf{Q}_{k_s k_t} &= \mathbf{\Phi}_{k_s}^{-1} - \left(\mathbf{\Phi}_{k_s} - \mathbf{\Psi}_{k_s k_t}\mathbf{\Phi}_{k_t}^{-1}\mathbf{\Psi}_{k_t k_s}\right)^{-1} \\ \mathbf{Q}_{k_t k_s} &= \mathbf{\Phi}_{k_t}^{-1} - \left(\mathbf{\Phi}_{k_t} - \mathbf{\Psi}_{k_t k_s}\mathbf{\Phi}_{k_s}^{-1}\mathbf{\Psi}_{k_s k_t}\right)^{-1} \\ \mathbf{P}_{k_s k_t} &= \mathbf{\Phi}_{k_s}^{-1}\mathbf{\Psi}_{k_s k_t}\left(\mathbf{\Phi}_{k_t} - \mathbf{\Psi}_{k_t k_s}\mathbf{\Phi}_{k_s}^{-1}\mathbf{\Psi}_{k_s k_t}\right)^{-1} \\ \mathbf{P}_{k_t k_s} &= \left(\mathbf{\Phi}_{k_t} - \mathbf{\Psi}_{k_t k_s}\mathbf{\Phi}_{k_s}^{-1}\mathbf{\Psi}_{k_s k_t}\right)^{-1}\mathbf{\Psi}_{k_t k_s}\mathbf{\Phi}_{k_s}^{-1} \\ &= \mathbf{P}_{k_s k_t}^\mathsf{T} \end{aligned} \quad (11)$$

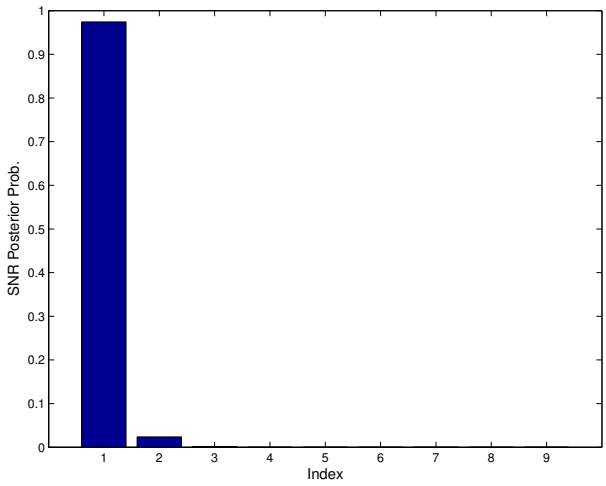
and

$$\mathbf{D}_1 = \begin{bmatrix}\mathbf{\Phi}_{k_s} & \mathbf{\Psi}_{k_s k_t} \\ \mathbf{\Psi}_{k_t k_s} & \mathbf{\Phi}_{k_t}\end{bmatrix};\ \mathbf{D}_2 = \begin{bmatrix}\mathbf{\Phi}_{k_s} & \mathbf{0} \\ \mathbf{0} & \mathbf{\Phi}_{k_t}\end{bmatrix} \quad (12)$$

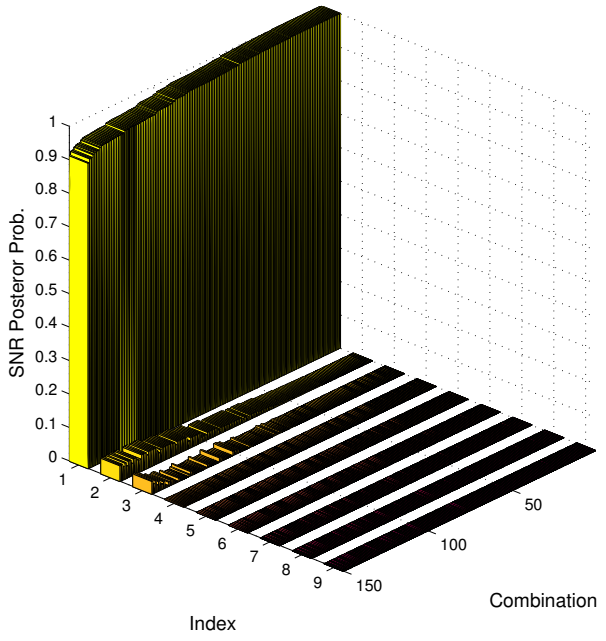
$$S_{\text{mPLDA}}(\mathbf{x}_s, \mathbf{x}_t) = \frac{\sum_{k_s=1}^K \sum_{k_t=1}^K \gamma_{\ell_s,\ell_t}(y_{k_s}, y_{k_t}) \mathcal{N}\left(\left[\mathbf{x}_s^\mathsf{T}\ \mathbf{x}_t^\mathsf{T}\right]^\mathsf{T} \middle| \left[\mathbf{m}_{k_s}^\mathsf{T}\ \mathbf{m}_{k_t}^\mathsf{T}\right]^\mathsf{T}, \hat{\mathbf{V}}_{k_s k_t}\hat{\mathbf{V}}_{k_s k_t}^\mathsf{T} + \hat{\mathbf{\Sigma}}_{k_s k_t} \right)}{\left[\sum_{k_s=1}^K \gamma_{\ell_s}(y_{k_s}) \mathcal{N}\left(\mathbf{x}_s | \mathbf{m}_{k_s}, \mathbf{V}_{k_s}\mathbf{V}_{k_s}^\mathsf{T} + \mathbf{\Sigma}_{k_s}\right)\right]\left[\sum_{k_t=1}^K \gamma_{\ell_t}(y_{k_t}) \mathcal{N}\left(\mathbf{x}_t | \mathbf{m}_{k_t}, \mathbf{V}_{k_t}\mathbf{V}_{k_t}^\mathsf{T} + \mathbf{\Sigma}_{k_t}\right)\right]} \quad (4)$$

$$= \frac{\sum_{k_s=1}^K \sum_{k_t=1}^K \gamma_{\ell_s,\ell_t}(y_{k_s}, y_{k_t}) \exp\left\{-\frac{1}{2}\log|\alpha\hat{\mathbf{\Lambda}}_{k_s k_t}| - \frac{1}{2}\mathcal{D}\left(\left[\mathbf{x}_s^\mathsf{T}\ \mathbf{x}_t^\mathsf{T}\right]^\mathsf{T} \middle\| \left[\mathbf{m}_{k_s}^\mathsf{T}\ \mathbf{m}_{k_t}^\mathsf{T}\right]^\mathsf{T}\right)\right\}}{\left[\sum_{k_s=1}^K \gamma_{\ell_s}(y_{k_s}) \exp\left\{-\frac{1}{2}\log|\alpha\mathbf{\Lambda}_{k_s}| - \frac{1}{2}\mathcal{D}\left(\mathbf{x}_s \| \mathbf{m}_{k_s}\right)\right\}\right]\left[\sum_{k_t=1}^K \gamma_{\ell_t}(y_{k_t}) \exp\left\{-\frac{1}{2}\log|\alpha\mathbf{\Lambda}_{k_t}| - \frac{1}{2}\mathcal{D}\left(\mathbf{x}_t \| \mathbf{m}_{k_t}\right)\right\}\right]} \quad (5)$$

$$S_{\text{mPLDA}}(\mathbf{x}_s, \mathbf{x}_t) \approx \frac{\gamma_{\ell_s,\ell_t}(y_{k_s}, y_{k_t}) \mathcal{N}\left(\left[\mathbf{x}_s^\mathsf{T}\ \mathbf{x}_t^\mathsf{T}\right]^\mathsf{T} \middle| \left[\mathbf{m}_{k_s}^\mathsf{T}\ \mathbf{m}_{k_t}^\mathsf{T}\right]^\mathsf{T}, \hat{\mathbf{V}}_{k_s k_t}\hat{\mathbf{V}}_{k_s k_t}^\mathsf{T} + \hat{\mathbf{\Sigma}}_{k_s k_t} \right)}{\gamma_{\ell_s}(y_{k_s}) \mathcal{N}\left(\mathbf{x}_s | \mathbf{m}_{k_s}, \mathbf{V}_{k_s}\mathbf{V}_{k_s}^\mathsf{T} + \mathbf{\Sigma}_{k_s}\right) \gamma_{\ell_t}(y_{k_t}) \mathcal{N}\left(\mathbf{x}_t | \mathbf{m}_{k_t}, \mathbf{V}_{k_t}\mathbf{V}_{k_t}^\mathsf{T} + \mathbf{\Sigma}_{k_t}\right)} \quad (6)$$



(a)



(b)

Fig. 1. (a) Posterior probabilities of SNR for $K = 3$. (a) Average posterior probabilities sorted in descending order. The horizontal axis represents the index to the 9 combinations of k_s and k_t in Eq. 3. (b) Individual SNR posterior probabilities of 150 combinations of target-speaker utterances' SNR (ℓ_s) and test-utterances' SNR (ℓ_t).

Expanding Eq. 10, we have

$$\begin{aligned}
&\log S_{\text{fast-mPLDA}}(\mathbf{x}_s, \mathbf{x}_t) \\
&= \log \gamma_{k_s k_t} - \log \gamma_{k_s} - \log \gamma_{k_t} \\
&\quad + \frac{1}{2}\mathbf{x}_s^{\mathsf{T}} \mathbf{Q}_{k_s k_t} (\mathbf{x}_s + 2\mathbf{m}_{k_s}) + \frac{1}{2}\mathbf{x}_t^{\mathsf{T}} \mathbf{Q}_{k_t k_s} (\mathbf{x}_t + 2\mathbf{m}_{k_t}) \\
&\quad + \mathbf{x}_s^{\mathsf{T}} \mathbf{P}_{k_s k_t} (\mathbf{x}_t + \mathbf{m}_{k_t}) + \mathbf{x}_t^{\mathsf{T}} \mathbf{P}_{k_s k_t}^{\mathsf{T}} \mathbf{m}_{k_s} \\
&\quad - \frac{1}{2}\log|\mathbf{D}_1| + \frac{1}{2}\log|\mathbf{D}_2| \\
&\quad - \mathbf{m}_{k_s}^{\mathsf{T}} \mathbf{Q}_{k_s k_t} \mathbf{m}_{k_s} - \mathbf{m}_{k_t}^{\mathsf{T}} \mathbf{Q}_{k_t k_s} \mathbf{m}_{k_t} \\
&\quad - \mathbf{m}_{k_s}^{\mathsf{T}} \mathbf{P}_{k_s k_t} \mathbf{m}_{k_t} - \mathbf{m}_{k_t}^{\mathsf{T}} \mathbf{P}_{k_s k_t}^{\mathsf{T}} \mathbf{m}_{k_s}.
\end{aligned} \tag{13}$$

Note that the last six terms in Eq. 13 are independent of $\mathbf{x}_s$ and $\mathbf{x}_t$. Therefore, they can be pre-computed before scoring. In the sequel, we refer to the scoring function in Eq. 13 as fast mPLDA scoring.

Note also that the conventional PLDA scoring is a special case of Eq. 13 where $k_s = k_t = K = 1$ and $\mathbf{m}_{k_s} = \mathbf{m}_{k_t} = \mathbf{0}$, i.e., there is only one mixture and therefore the SNR posterior γ always equal to 1.0. Specifically, the scoring function for PLDA is

$$\log S_{\text{PLDA}}(\mathbf{x}_s, \mathbf{x}_t) = \frac{1}{2}\mathbf{x}_s^{\mathsf{T}}\mathbf{Q}\mathbf{x}_s + \frac{1}{2}\mathbf{x}_t^{\mathsf{T}}\mathbf{Q}\mathbf{x}_t + \mathbf{x}_s^{\mathsf{T}}\mathbf{P}\mathbf{x}_t \tag{14}$$

where we have dropped the subscripts for $\mathbf{P}$ and $\mathbf{Q}$ for clarity. Similar to fast mPLDA, $\mathbf{P}$ and $\mathbf{Q}$ can be pre-computed using Eq. 11.

IV. Experiments

A. Speech Data and Acoustic Features

The male speech files in the core set of NIST 2012 Speaker Recognition Evaluation (SRE) [30] were used for performance evaluation. In 2012 SRE, noise was added to the test segments of common conditions 3 and 4, resulting in the SNR distribution shown in Fig. 2. Because mPLDA is designed to improve the robustness of PLDA systems under noisy environments, this paper focuses on these two common conditions. Table II shows the acoustic conditions of the test segments in these common conditions. Enrollment utterances with length less than 10 seconds and the summed-channel utterances were removed. However, we ensured that all target speakers have at least one utterance for enrollment. The speech files in NIST 2005–2010 SREs were used as development data for training gender-dependent UBMs, total variability matrices, LDA-WCCN [7], PLDA and mPLDA models.

Speech regions in the speech files were extracted by using a two-channel VAD [31]. 19 MFCCs together with energy plus their 1st- and 2nd- derivatives were extracted from the speech regions, followed by cepstral mean normalization and feature warping [4] with a window size of 3 seconds. A 60-dim acoustic vector was extracted every 10ms, using a Hamming window of 25ms. For each clean training file, we randomly select one out of the 30 noise files from the PRISM dataset [32] and added the noise waveform to the file at an SNR of 6dB and 15dB using the FaNT tool [33].

TABLE II
Conditions of Test segments in CC3 and CC4 of NIST 2012 SRE.

Common Condition	Test-segment Conditions
CC3	Interview speech with added noise
CC4	Phone call speech with added noise

B. SNR Measurements

To measure the "actual" SNR of speech files (including the original and noise contaminated ones), we used the voltmeter function of FaNT and the speech/non-speech decisions of our VAD [31], [34] as follows. Given a speech file, we passed the waveform to the G.712 frequency weighting filter in FaNT and

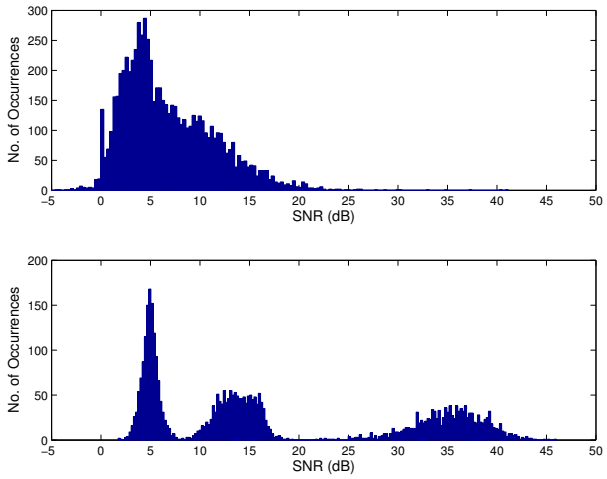


Fig. 2. Distributions of SNR in Common Condition 3 (top panel) and Common Condition 4 (bottom panel) in NIST 2012 SRE. See Section IV-B for the procedure of SNR measurement.

then estimated the speech energy using the voltmeter function (`sv-p56.c` from the ITU-T Software Tool Library [35]). Then, we extracted the non-speech segments based on the VAD's decisions and passed the non-speech segments to the voltmeter function to estimate the noise energy. The difference between the signal and noise energies in the log domain gives the measured SNR of the file.

C. PLDA and Mixture of PLDA

The i-vector system is based on gender-dependent UBMs with 1024 mixtures and total variability matrices with 500 total factors. Microphone and telephone utterances from NIST 2005–2008 SREs were used for training the UBMs and total variability matrices. Following [14], within-class covariance normalization (WCCN) [16] and i-vector length normalization [17] were applied to the 500-dimensional i-vectors. Then, linear discriminant analysis (LDA) [15] and WCCN were applied to reduce the dimension to 200 before training the PLDA and mixture of PLDA models with 150 latent variables.

To train the mPLDA models for CC4, we pooled the 6dB (tel), 15dB (tel), and original (tel+mic) speech files in 2006–2010 SRE—excluding speakers with less than two utterances—into a single training set. The EM algorithm specified in [27] were used to train a mixture PLDA models with $K = 3$ and $K = 4$. The number of speaker factors (M) was set to 150 in all cases. As shown in Table II, the test segments in CC3 comprises interview speech with added noise. As a result, we only used microphone enrollment utterances to train the PLDA and mPLDA models for CC3.

As for PLDA scoring, we followed the conventional PLDA scoring function [17] as specified in Eq. 14. For mPLDA scoring and fast mPLDA scoring, we applied Eq. 5 and Eq. 13, respectively.

V. Results and Discussions

Tables III(a) and III(b) show the breakdown of computation time within each invocation of the Matlab functions that implement mPLDA scoring (Eq. 5) and fast mPLDA scoring (Eq. 13). The computation time was measured by Matlab's Profiler when K was set to 3, average over all the trials in CC4. When estimating the computation time for Eq. 5 and Eq. 13, it is assumed that the posterior probabilities γ's have already been computed. I-vector preprocessing includes the time to perform i-vector whitening, length-normalization, LDA, and WCCN projection. All measurements were done on an Intel Quad CPU Q9550 running at 2.83GHz.

For mPLDA, the computation of likelihoods in Eq. 5 takes over 60% of the overall time. This is because when $K = 3$, there are 9 Mahalanobis distances in the numerator and 6 Mahalanobis distances in the denominator. For fast mPLDA, on the other hand, the computation of SNR posteriors consumes 43% of the overall time. This suggests that the scoring function in Eq. 13 is very efficient. The computation saving comes from omitting the likelihoods with small SNR posterior γ. Comparing the overall time in Tables III(a) and III(b) reveals that fast mPLDA reduces the scoring time by more than 60%.

Table IV shows the EER and minimum DCF (minC_{Primary}) achieved by PLDA (baseline), mPLDA, and fast mPLDA in CC3 and CC4 of NIST 2012 SRE. Also shown are the scoring time (in seconds) to perform all trials using different methods. Results show that the EER of mPLDA is significantly lower than that of PLDA, although the former is slightly inferior in terms of minDCF. While the computational complexity of PLDA and fast mPLDA is the same (see Table I), the actual scoring time of fast mPLDA is still significantly longer than that of PLDA. The reason is that computing the SNR posterior probabilities takes time, as shown in Table III. Table IV also demonstrates that the fast scoring approach proposed in this paper reduces the overall scoring time by more than 60%.

VI. Conclusions

This paper proposes to speed up the scoring process of SNR-dependent mixture of PLDA. This is achieved by omitting the computation of the likelihood terms when their corresponding SNR posterior is small. In the extreme case, only the likelihood whose SNR posterior is the largest is considered. It was found that the SNR posteriors are sparse so that even for this extreme case, the loss in performance is minor but the scoring time can be cut by half. In future work, it is interesting to consider not only the top SNR posterior but also a few runner-ups when evaluating the scoring function to see if it is possible to reduce scoring time without sacrificing verification performance.

Acknowledgment

This work was in part supported by The RGC of Hong Kong SAR, Grant No. PolyU 152117/14E.

TABLE IV
PERFORMANCE AND SCORING TIME OF CONVENTIONAL PLDA, MPLDA (EQ. 5) AND MPLDA WITH FAST SCORING (EQ. 13) UNDER COMMON CONDITIONS 3 AND 4 IN NIST 2012 SRE (MALE, CORE SET).

Method	K	CC3			CC4		
		EER(%)	minDCF	Time(sec.)	EER(%)	minDCF	Time(sec.)
PLDA	–	5.77	0.227	67	3.49	0.308	703
mPLDA	2	4.79	0.255	333	3.31	0.305	6098
	3	4.79	0.257	1046	2.99	0.316	6502
	4	4.75	0.255	2558	3.24	0.317	24575
Fast mPLDA	2	4.89	0.238	249	3.23	0.295	3529
	3	4.89	0.248	254	3.10	0.314	2286
	4	4.68	0.239	331	3.17	0.318	2585

TABLE III
I-VECTOR PREPROCESSING TIME AND COMPUTATION TIME IN DIFFERENT PARTS OF (A) MPLDA IN EQ. 5 AND (B) FAST MPLDA IN EQ. 13. THE I-VECTOR PREPROCESSING TIME INCLUDES WHITENING, LENGTH-NORMALIZATION, LDA, AND WCCN. THE COMPUTATION TIME WAS OBTAINED BY USING MATLAB PROFILER. IN BOTH CASES, THE TIME REQUIRED FOR COMPUTING $S_{\text{MPLDA}}(\mathbf{x}_s, \mathbf{x}_t)$ AND $S_{\text{FAST-MPLDA}}(\mathbf{x}_s, \mathbf{x}_t)$ ASSUMES THAT $\gamma_{\ell_s,\ell_t}(y_{k_s}, y_{k_t})$, $\gamma_{\ell_s}(y_{k_s})$ AND $\gamma_{\ell_t}(y_{k_t})$ HAVE BEEN COMPUTED.

Function	Time (ms)	% of Scoring Time
$S_{\text{mPLDA}}(\mathbf{x}_s, \mathbf{x}_t)$ in Eq. 5	2.295	61.3%
I-vector Preprocessing	0.678	18.1%
$\gamma_{\ell_s,\ell_t}(y_{k_s}, y_{k_t})$	0.343	9.2%
$\gamma_{\ell_s}(y_{k_s})$ & $\gamma_{\ell_t}(y_{k_t})$	0.216	5.8%
Other operations and overhead	0.209	5.6%
Overall	3.741	100%

(a) mPLDA

Function	Time (ms)	% of Scoring Time
I-vector Preprocessing	0.520	37.0%
$\gamma_{\ell_s,\ell_t}(y_{k_s}, y_{k_t})$	0.372	26.5%
$S_{\text{fast-mPLDA}}(\mathbf{x}_s, \mathbf{x}_t)$ in Eq. 13	0.206	20.6%
$\gamma_{\ell_s}(y_{k_s})$ & $\gamma_{\ell_t}(y_{k_t})$	0.203	14.4%
Other operations and overhead	0.106	7.5%
Overall	1.407	100%

(b) Fast mPLDA

REFERENCES

[1] S. O. Sadjadi, T. Hasan, and J.H.L Hansen, "Mean Hilbert envelope coefficients (MHEC) for robust speaker recognition.," in *Proc. of Interspeech*, 2012, pp. 1696–1699.

[2] Y. Shao and D.L. Wang, "Robust speaker identification using auditory features and computational auditory scene analysis," in *Proc. of IEEE International Conference on Acoustics, Speech and Signal Processing(ICASSP)*, March 2008, pp. 1589–1592.

[3] Q. Li and Y. Huang, "Robust speaker identification using an auditory-based feature," in *Proc. of IEEE International Conference on Acoustics, Speech and Signal Processing(ICASSP)*, March 2010, pp. 4514–4517.

[4] J. Pelecanos and S. Sridharan, "Feature warping for robust speaker verification," in *Proc. of Odyssey: The Speaker and Language Recognition Workshop*, Crete, Greece, Jun. 2001, pp. 213–218.

[5] R. Saeidi and D. A van Leeuwen, "The Radboud University Nijmegen submission to NIST SRE-2012," in *Proc. of the NIST Speaker Recognition Evaluation Workshop*, 2012.

[6] P. Kenny, P. Ouellet, N. Dehak, V. Gupta, and P. Dumouchel, "A study of inter-speaker variability in speaker verification," *IEEE Transactions on Audio, Speech and Language Processing*, vol. 16, no. 5, pp. 980–988, 2008.

[7] N. Dehak, P. Kenny, R. Dehak, P. Dumouchel, and P. Ouellet, "Front-end factor analysis for speaker verification," *IEEE Transactions on Audio, Speech and Language Processing*, vol. 19, no. 4, pp. 788–798, May 2011.

[8] P. Kenny, "Bayesian speaker verification with heavy-tailed priors," in *Proc. of Odyssey: Speaker and Language Recognition Workshop*, Brno, Czech Republic, June 2010.

[9] W. M. Campbell, D. E. Sturim, and D. A. Reynolds, "Support vector machines using GMM supervectors for speaker verification," *IEEE Signal Processing Letters*, vol. 13, no. 5, pp. 308–311, May 2006.

[10] D. A. Reynolds, T. F. Quatieri, and R. B. Dunn, "Speaker verification using adapted Gaussian mixture models," *Digital Signal Processing*, vol. 10, no. 1–3, pp. 19–41, Jan. 2000.

[11] W. Rao, M.W. Mak, and K.A. Lee, "Normalization of total variability matrix for i-vector/plda speaker verification," in *ICASSP'2015*, 2015, pp. 4180–4184.

[12] S.J.D. Prince and J.H. Elder, "Probabilistic linear discriminant analysis for inferences about identity," in *Proc. of IEEE 11th International Conference on Computer Vision (ICCV)*, 2007, pp. 1–8.

[13] W. Rao and M. W. Mak, "Boosting the performance of i-vector based speaker verification via utterance partitioning," *IEEE Trans. on Audio, Speech and Language Processing*, vol. 21, no. 5, pp. 1012–1022, 2013.

[14] M. McLaren, M.I. Mandasari, and D.A. Leeuwen, "Source normalization for language-independent speaker recognition using i-vectors," in *Proc. of Odyssey 2012: The Speaker and Language Recognition Workshop*, 2012, pp. 55–61.

[15] C.M. Bishop, *Pattern Recognition and Machine Learning*, Springer: New York, 2006.

[16] A. Hatch, S. Kajarekar, and A. Stolcke, "Within-class covariance normalization for SVM-based speaker recognition," in *Proc. of the 9th International Conference on Spoken Language Processing*, Pittsburgh, PA, USA, Sep. 2006, pp. 1471–1474.

[17] D. Garcia-Romero and C.Y. Espy-Wilson, "Analysis of i-vector length normalization in speaker recognition systems," in *Proc. of Interspeech*, 2011, pp. 249–252.

[18] D. A.van Leeuwen and R. Saeidi, "Knowing the non-target speakers: The effect of the i-vector population for PLDA training in speaker recognition," in *Proc. of IEEE International Conference on Acoustics, Speech and Signal Processing(ICASSP), 2013*, Vancouver, BC, Canada, May 2013, pp. 6778 – 6782.

[19] Y. Lei, L. Burget, L. Ferrer, M. Graciarena, and N. Scheffer, "Towards noise-robust speaker recognition using probabilistic linear discriminant analysis," in *Proc. of IEEE International Conference on Acoustics, Speech and Signal Processing(ICASSP)*, Kyoto, Japan, March 2012, pp. 4253 – 4256.

[20] T. Hasan, S. O. Sadjadi, G. Liu, N. Shokouhi, H. Boril, and J. H. L. Hansen, "CRSS system for 2012 NIST speaker recogntion evaluation,"

in *Proc. of IEEE International Conference on Acoustics, Speech and Signal Processing(ICASSP)*, 2013, pp. 6783–6787.

[21] P. Rajan, T. Kinnunen, and V. Hautamäki, "Effect of multicondition training on i-vector PLDA configurations for speaker recognition," in *Proc. of Interspeech*, 2013, pp. 3694–3697.

[22] D. Garcia-Romero, X. Zhou, and C.Y. Espy-Wilson, "Multicondition training of Gaussian PLDA models in i-vector space for noise and reverberation robust speaker recognition," in *Proc. of IEEE International Conference on Acoustics, Speech and Signal Processing(ICASSP)*, 2012, pp. 4257–4260.

[23] T. Hasan and J.H.L. Hansen, "Acoustic factor analysis for robust speaker verification," *IEEE Transactions on Audio, Speech and Language Processing,*, vol. 21, no. 4, pp. 842–853, 2013.

[24] T. Hasan and J.H.L. Hansen, "Maximum likelihood acoustic factor analysis models for robust speaker verification in noise," *IEEE Transactions on Audio, Speech And Language Processing*, vol. 22, no. 2, pp. 381–391, 2014.

[25] Y. Lei, L. Burget, and N. Scheffer, "A noise robust i-vector extractor using vector Taylor series for speaker recognition.," in *Proc. of IEEE International Conference on Acoustics, Speech and Signal Processing(ICASSP)*, 2013, pp. 6788–6791.

[26] N. Li and M. W. Mak, "SNR-invariant PLDA modeling in nonparametric subspace for robust speaker verification," *Audio, Speech, and Language Processing, IEEE/ACM Transactions on*, vol. 23, no. 10, pp. 1648–1659, Oct 2015.

[27] M. W. Mak, "SNR-dependent mixture of PLDA for noise robust speaker verification," in *Interspeech'2014*, 2014, pp. 1855–1859.

[28] Z. Ghahramani and G. E. Hinton, "The EM algorithm for mixtures of factor analyzers," Technical report CRG-TR-96-1, Department of Computer Science, University of Toronto, 1996.

[29] K. B. Petersen and M. S. Pedersen, "The matrix cookbook," Oct 2008.

[30] NIST, "The NIST year 2012 speaker recognition evaluation plan," *http://www.nist.gov/itl/iad/mig/sre12.cfm*, 2012.

[31] M. W. Mak and H. B. Yu, "A study of voice activity detection techniques for NIST speaker recognition evaluations," *Computer, Speech and Language*, vol. 28, no. 1, pp. 295–313, Jan 2013.

[32] L. Ferrer, H. Bratt, L. Burget, H. Cernocky, O. Glembek, M. Graciarena, A. Lawson, Y. Lei, P. Matejka, O. Plchot, et al., "Promoting robustness for speaker modeling in the community: The PRISM evaluation set," .

[33] "http://dnt.kr.hsnr.de/download.html," .

[34] H.B. Yu and M.W. Mak, "Comparison of voice activity detectors for interview speech in nist speaker recognition evaluation," in *Interspeech*, 2011, pp. 2353–2356.

[35] Simao Ferraz De Campos Neto, "The ITU-T software tool library," *International journal of speech technology*, vol. 2, no. 4, pp. 259–272, 1999.