SNR-Invariant PLDA Modeling in Nonparametric Subspace for Robust Speaker Verification

Na Li, and Man-Wai Mak, *Senior Member, IEEE*

***Abstract*—While i-vector/PLDA framework has achieved great success, its performance still degrades dramatically under noisy conditions. To compensate for the variability of i-vectors caused by different levels of background noise, this paper proposes an SNR-invariant PLDA framework for robust speaker verification. First, nonparametric feature analysis (NFA) is employed to suppress intra-speaker variation and emphasize the discriminative information inherited in the boundaries between speakers in the i-vector space. Then, in the NFA-projected subspace, SNR-invariant PLDA is applied to separate the SNR-specific information from speaker-specific information using an identity factor and an SNR factor. Accordingly, a projected i-vector in the NFA subspace can be represented as a linear combination of three components: speaker, SNR, and channel. During verification, the variability due to SNR and channels are integrated out when computing the marginal likelihood ratio. Experiments based on NIST 2012 SRE show that the proposed framework achieves superior performance when compared with the conventional PLDA and SNR-dependent mixture of PLDA.**

***Index Terms*—i-vector, PLDA, SNR-invariant, nonparametric feature analysis, speaker verification.**

I. Introduction

In text-independent speaker verification [1], [2], most state-of-the-art systems use an i-vector [3] to represent the acoustic characteristics of an utterance. Unlike joint factor analysis (JFA) [4], [5], [6] in which the channel and speaker variabilities are compressed into two distinct subspaces respectively, the i-vector framework learns a single low-dimensional subspace called the total variability subspace, through which utterances of variable-length can be represented as fixed-length i-vectors whose elements are the latent variables of a factor analyzer [7]. Such a representation greatly simplifies the modeling process as the dimension of i-vectors is much lower than that of GMM-supervectors [8], [9]. Statistical techniques, such as linear discriminant analysis (LDA) [10], within-class covariance normalization (WCCN) [11], and probabilistic LDA (PLDA) [12], can be applied to suppress the channel- and session-variability in i-vectors. Typically, LDA is applied to the i-vectors followed by the WCCN. The former aims to find a low-dimensional subspace of the total variability space in which intersession variability is minimal, and the latter further compensates for intersession variability by normalizing the within-speaker covariance while maintaining the direction of the subspace found by LDA. Cosine distance between the target-speaker's i-vector and test i-vector is then used as a similarity measure between the target speaker and the test speaker. Alternatively, by assuming that the priors on the latent variables of the PLDA model follow a Gaussian distribution or Student's t distribution, the Gaussian PLDA and heavy-tailed PLDA [13] were proposed respectively. To deal with the non-Gaussian behavior of i-vectors, length normalization [14], [15], [16] is applied to the i-vectors so that the resulting vectors are more amendable to Gaussian PLDA modeling.

In the i-vector/PLDA framework, the purpose of LDA is to find a low-dimensional subspace for PLDA modeling. This is achieved by finding a subspace that maximizes the between-class separation and minimizes the within-class variation. LDA assumes that the density of each class is a Gaussian and that the classes share the same covariance structure. This assumption results in the computation of two matrices: within-class scatter matrix and between-class scatter matrix. The accuracy of these matrices depends on a number of factors, of which the accuracy of class-dependent mean vectors is of fundamental importance. In the context of i-vector speaker verification, these class-dependent means are the means of speaker-dependent i-vectors, each of which is estimated from the utterances (sessions) of a training speaker. In most practical situations, the number of training speakers could be as large as several hundred, but many of these speakers may only have a few sessions. The limited number of sessions per speaker could lead to inaccurate class-dependent mean vectors. As there is no mechanism to suppress the effect of these inaccurate mean vectors on the scatter matrices, the performance of LDA will suffer.

To address the limitation of LDA, a nonparametric feature analysis (NFA) [17] approach was proposed and successfully applied to face recognition. Instead of using the class-dependent means to compute the within-class scatter matrix, NFA replaces the class-dependent means by the nearest neighbors to each training vector. Similarly, to compute the between-class scatter matrix, NFA uses all of the training vectors and their nearest neighbors from other classes rather than utilizing the class-dependent means and the global mean. This strategy is very effective in capturing the discriminative class boundary information inherited in the training set.

Recently, noise robust speaker verification has received increasing attention due to its great practical value [18], [19], [20], [21]. Studies have shown that background noise has severe effects on the performance of speaker recognition systems [22], [23], [24]. A recent study [25] also found that mismatches in radio channels could have detrimental effect on performance.

Although traditional PLDA model addresses session mismatch well, it does not consider the effect of noise at varying

The authors are with The Department of Electronic and Information Engineering, The Hong Kong Polytechnic University, Hong Kong, SAR (Email: lina011779@126.com; enmwmak@polyu.edu.hk). This work was in part supported by The RGC of Hong Kong SAR (Grant No. PolyU 152117/14E) and Hong Kong Polytechnic University (Grant No. G-YN18).

signal-to-noise ratio (SNR). To improve the robustness of i-vector/PLDA systems, several methods have been proposed. For example, Hasan and Hansen [26], [27] proposed a two stage factor analysis scheme in which the posterior means and covariances of acoustic factors in the first stage are used for i-vector extraction in the second stage. Kheder *et al.* [28] clean up noisy i-vectors by using an additive noise model in the i-vector space.

Another direction is to pooled clean and noisy utterances together to train a robust PLDA model [29], [30], [31], [32], [33]. In particular, Garcia-Romero *et al.* [34] applied multi-condition training to train multiple PLDA models, one for each condition. A robust system was then constructed by combining all of the PLDA models according to the posterior probability of each condition. Mak [35] proposed an adaptive multi-condition training algorithm called mixture of SNR-dependent PLDA to handle test utterances with a wide range of SNR. The main limitations of these multi-condition training methods are that (1) the system performance degrades when the distributions of SNR in the training set and the test set are not consistent, (2) it is necessary to train multiple PLDA models, which increases computation complexity during the training and recognition stages, and (3) the noise level of each test utterance should be estimated when computing the verification score.

Motivated by the limitations of LDA and multi-condition training in i-vector/PLDA framework, we propose a new framework for SNR-invariant speaker verification by incorporating the SNR variability into PLDA modeling in the reduced nonparametric subspace. As a preprocessor for PLDA modeling, NFA aims to maximize inter-speaker separation and emphasize the boundaries between speakers in the i-vector space by constructing two nonparametric scatter matrices. In the back-end modeling stage, we propose a noise robust speaker verification method that can deal with the mismatch caused by noise with a wide range of SNR. Inspired by the hidden factor analysis [36] approach to age invariant face recognition, we assume that the noise-related variability and the speaker-related variability embedded in the NFA- or LDA-projected i-vectors can be modeled by an SNR factor and a speaker (identity) factor in a linear generative model. We refer to it as SNR-invariant PLDA. In this model, the identity component and the SNR component live in two different subspaces which can be obtained by an expectation-maximization (EM) algorithm. In the verification stage, both target and test i-vectors are projected onto the nonparametric subspace. Then, SNR variability and channel variability are integrated out when the likelihood ratio is computed.

II. I-VECTOR/PLDA FOR SPEAKER VERIFICATION

Conventional i-vector/PLDA framework consists of three parts: i-vector extraction, inter-session variability compensation, and PLDA modeling.

A. I-vector Extraction

Inspired by the observation that the channel subspace still contains speaker-related information in joint factor analysis (JFA), Dehak *et al.* [3] proposed to jointly model speaker and channel variabilities in a combined subspace called the total variability space. Using this subspace, a speaker- and channel-dependent GMM-supervector $\mathbf{M}_i$ can be expressed as:

$$\mathbf{M}_i = \mathbf{M} + \boldsymbol{T}\mathbf{x}_i \tag{1}$$

where $\mathbf{M}$ is the speaker- and channel-independent GMM-supervector obtained by stacking the mean vectors of a universal background model (UBM) [37], $\boldsymbol{T}$ is a low-rank matrix whose columns define the bases of the total variability space, and the loading factor $\mathbf{x}_i$ is a low-dimensional identity vector referred to as i-vector. The i-vector $\mathbf{x}_i$ is assumed to follow a standard normal distribution $\mathcal{N}(\mathbf{0}, \mathbf{I})$. Given the centralized Baum-Welch statistics from a set of training utterances, the matrix $\boldsymbol{T}$ is estimated via an expectation maximization (EM) algorithm identical to that of the joint factor analysis [4] but with the speaker labels ignored. Given $\boldsymbol{T}$ and an utterance, the posterior mean of the latent factor is estimated and considered as the i-vector $\mathbf{x}_i$ of the utterance.

B. Inter-Session Variability Compensation

As i-vectors comprise both speaker- and channel-characteristics, it is important to compensate for inter-session variability. Typically, this can be done by LDA projection followed by WCCN.

1) Linear Discriminant Analysis: The main purpose of LDA [10] is to reduce the dimensionality of i-vectors before PLDA modeling [13]. LDA is a parametric discriminant analysis technique as it uses the parametric form of Gaussian distributions to represent the scatter matrices. LDA aims to determine an optimal projection $\mathbf{W}$ that maximizes the between-class scatter and minimizes the within-class scatter of the projected data. Specifically, the optimal projection matrix $\mathbf{W}_{lda}$ is calculated as [38]:

$$\mathbf{W}_{lda} = \underset{\mathbf{W}}{\operatorname{argmax}} \frac{|\mathbf{W}^\top \mathbf{S}_b \mathbf{W}|}{|\mathbf{W}^\top \mathbf{S}_w \mathbf{W}|} \tag{2}$$

where $\mathbf{S}_w$ and $\mathbf{S}_b$ are the within-class scatter matrix and between-class scatter matrix, respectively. The within-class scatter matrix is defined as:

$$\mathbf{S}_w = \sum_{i=1}^{S} \sum_{j=1}^{N_i} (\mathbf{x}_{ij} - \boldsymbol{\mu}_i)(\mathbf{x}_{ij} - \boldsymbol{\mu}_i)^\top \tag{3}$$

where $\mathbf{x}_{ij}$ is the *j*-th i-vector from speaker *i*, S is the number of speakers, N_i denotes the number of i-vectors belonging to speaker i, and $\boldsymbol{\mu}_i$ is the mean of the i-vectors from speaker *i*. The between-class scatter matrix is given by

$$\mathbf{S}_b = \sum_{i=1}^{S} N_i (\boldsymbol{\mu}_i - \boldsymbol{\mu})(\boldsymbol{\mu}_i - \boldsymbol{\mu})^\top \tag{4}$$

where $\boldsymbol{\mu}$ is the global mean of all i-vectors. The solution of Eq. 2 comprises the eigenvectors of $\mathbf{S}_w^{-1}\mathbf{S}_b$.

2) Within Class Covariance Normalization: Given the LDA-projected i-vectors, WCCN [11] aims to further suppress the intra-speaker variability in the reduced subspace. The WCCN projection matrix can be calculated as the square-root

of the inverse of the standard within-class covariance matrix:

$$\mathbf{W}_{wccn} = \mathbf{S}_{wccn}^{-1/2} \tag{5}$$

where

$$\mathbf{S}_{wccn} = \sum_{i=1}^{S} \frac{1}{N_i} \sum_{j=1}^{N_i} (\mathbf{W}_{lda}^{\top}\mathbf{x}_{ij} - \hat{\boldsymbol{\mu}}_i)(\mathbf{W}_{lda}^{\top}\mathbf{x}_{ij} - \hat{\boldsymbol{\mu}}_i)^{\top} \tag{6}$$

where $\hat{\boldsymbol{\mu}}_i$ denotes the mean of the LDA-projected i-vectors from speaker *i*.

C. Gaussian PLDA Modeling

Prince and Elder [39], [40] proposed a probabilistic LDA (PLDA) approach to increasing the separability between the facial images of different persons. Kenny [13] brought this idea to the speaker recognition community by assuming that the i-vectors follow a Student's t distribution. The resulting model is commonly referred to as heavy-tailed PLDA models. Shortly after Kenny's work, it was discovered that PLDA models with Gaussian priors on the latent factors can achieve almost the same performance as heavy-tailed PLDA if the i-vectors have been length-normalized [14], [15], [16]. The resulting model is known as Gaussian PLDA models in the literature.

In Gaussian PLDA, a length-normalised i-vector $\hat{\mathbf{x}}_{ij}$ from the j-th session of speaker i is regarded as an observation generated from a probabilistic model of the form:

$$\hat{\mathbf{x}}_{ij} = \mathbf{m} + \mathbf{V}\mathbf{h}_i + \mathbf{G}\mathbf{r}_{ij} + \boldsymbol{\epsilon}_{ij}. \tag{7}$$

In the model, $\mathbf{m}$ is the global mean of i-vectors, $\mathbf{V}$ defines the speaker subspace with the speaker factor $\mathbf{h}_i$ following a standard normal distribution, $\mathbf{G}$ defines the channel subspace with the channel factor $\mathbf{r}_{ij}$ that follows a standard normal distribution, and $\boldsymbol{\epsilon}_{ij}$ is a residual term following a Gaussian distribution with zero mean and diagonal covariance matrix $\boldsymbol{\Sigma}$.

According to [13], [14], the PLDA model in Eq. 7 can be divided into two parts: (1) the speaker part ($\mathbf{m} + \mathbf{V}\mathbf{h}_i$) that depends on the i-th speaker only and (2) the channel part ($\mathbf{G}\mathbf{r}_{ij} + \boldsymbol{\epsilon}_{ij}$) that depends not only on the speaker but also on his/her sessions. As i-vectors are of sufficiently low dimension, the term $\mathbf{G}\mathbf{r}_{ij}$ can be absorbed into $\boldsymbol{\Sigma}$ if the latter is a full covariance matrix. Accordingly, the Gaussian PLDA model can be simplified as follow:

$$\hat{\mathbf{x}}_{ij} = \mathbf{m} + \mathbf{V}\mathbf{h}_i + \boldsymbol{\epsilon}_{ij}, \tag{8}$$

where $\boldsymbol{\epsilon}_{ij} \sim \mathcal{N}(\mathbf{0}, \boldsymbol{\Sigma})$ with $\boldsymbol{\Sigma}$ being a full covariance matrix. This paper adopts this simplified model.

III. SNR-Invariant PLDA Modeling in Nonparametric Subspace

In order to enhance the noise robustness of i-vector/PLDA systems, several multi-condition training methods have been proposed to generate multiple SNR-dependent PLDA models [34], [35], [41]. In spite of the promising results obtained by these methods, the systems work well only under some restricted conditions. For example, the SNRs of training and test utterances must be within the same range and it is necessary to measure the SNRs of the test utterances during verification. Moreover, computational complexity increases significantly because multiple PLDA models must be trained. To address these limitations, this paper proposes a framework for SNR-invariant PLDA modeling in nonparametric subspace. Before SNR-invariant PLDA modeling, nonparametric feature analysis (NFA) is applied to the i-vectors to maximize the discriminative information embedded in the i-vectors. Then, an SNR-invariant PLDA model is constructed using the NFA-projected i-vectors as input.

A. Nonparametric Feature Analysis

To address the limitations of LDA, a nonparametric feature analysis (NFA) technique was proposed for face recognition [17]. Inspired by this study, we applied NFA to i-vector based speaker verification as follows. Denote $\mathbf{x}_{ij}$ as the *j*-th WCCN-whitened and length-normalized i-vector from speaker *i*. The nonparametric within-class scatter matrix and between-class scatter matrix in NFA are calculated as follows:

$$\mathbf{S}_w^{\text{NFA}} = \sum_{i=1}^{S} \sum_{l=1}^{k_1-1} \sum_{j=1}^{N_i} (\mathbf{x}_{ij} - \psi_l(\mathbf{x}_{ij}, i))(\mathbf{x}_{ij} - \psi_l(\mathbf{x}_{ij}, i))^{\top} \tag{9}$$

$$\begin{aligned} \mathbf{S}_b^{\text{NFA}} = \sum_{i=1}^{S} \sum_{\substack{r=1 \\ r\neq i}}^{S} \sum_{l=1}^{k_2} \sum_{j=1}^{N_i} \omega(i,r,l,j)(\mathbf{x}_{ij} - \psi_l(\mathbf{x}_{ij}, r)) \\ (\mathbf{x}_{ij} - \psi_l(\mathbf{x}_{ij}, r))^{\top} \end{aligned} \tag{10}$$

where S is the total number of speakers in the training set, N_i is the number of i-vectors from speaker *i*, $\psi_l(\mathbf{x}_{ij}, r)$ is the *l*-th nearest neighbor from speaker *r* to $\mathbf{x}_{ij}$, and $k_1 - 1$ and k_2 respectively denote the number of the nearest neighbors selected during the computation of $\mathbf{S}_w^{\text{NFA}}$ and $\mathbf{S}_b^{\text{NFA}}$. The weighting term $\omega(i,r,l,j)$ in Eq. 10 is defined as:

$$\begin{aligned} \omega(i,r,l,j) &= \frac{\min\{d^{\alpha}(\mathbf{x}_{ij}, \psi_l(\mathbf{x}_{ij}, i)), d^{\alpha}(\mathbf{x}_{ij}, \psi_l(\mathbf{x}_{ij}, r))\}}{d^{\alpha}(\mathbf{x}_{ij}, \psi_l(\mathbf{x}_{ij}, i)) + d^{\alpha}(\mathbf{x}_{ij}, \psi_l(\mathbf{x}_{ij}, r))} \\ &= \frac{1}{1 + g(\mathbf{x}_{ij})^{\alpha}}, \qquad i \neq r \end{aligned} \tag{11}$$

where $d(\mathbf{x}_1, \mathbf{x}_2)$ is the Euclidean distance between vector $\mathbf{x}_1$ and vector $\mathbf{x}_2$, and α controls the rate of change of the weight with respect to the distance ratio $g(\mathbf{x}_{ij})$. In this paper, α is set to 2.

Note that LDA is parametric in that the global mean $\boldsymbol{\mu}$ and class-dependent means $\boldsymbol{\mu}_i$ in Eqs. 3 and 4 are parameters of Gaussian distributions. On the other hand, NFA is non-parametric in that the terms in Eqs. 9 and 10 are vectors near the decision boundaries. They are not the parameters of distributions.

In Eq. 11, if $d^{\alpha}(\mathbf{x}_{ij}, \psi_l(\mathbf{x}_{ij}, i)) < d^{\alpha}(\mathbf{x}_{ij}, \psi_l(\mathbf{x}_{ij}, r))$, then

$$g(\mathbf{x}_{ij}) = \frac{d(\mathbf{x}_{ij}, \psi_l(\mathbf{x}_{ij}, r))}{d(\mathbf{x}_{ij}, \psi_l(\mathbf{x}_{ij}, i))},$$

otherwise

$$g(\mathbf{x}_{ij}) = \frac{d(\mathbf{x}_{ij}, \psi_l(\mathbf{x}_{ij}, i))}{d(\mathbf{x}_{ij}, \psi_l(\mathbf{x}_{ij}, r))}.$$

For a selected i-vector $\mathbf{x}_{ij}$ near a speaker boundary (e.g., $\mathbf{x}_{i1}$ and $\mathbf{x}_{i2}$ shown in Fig. 1), the between-class distance

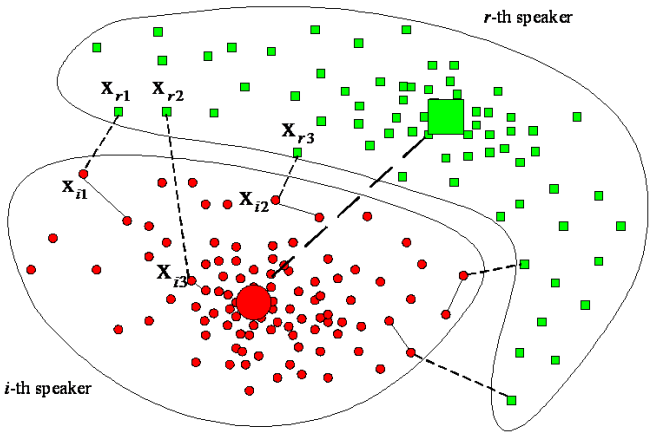


Fig. 1. Hypothetic example illustrating the between-class distance $d(\mathbf{x}_{ij}, \psi_l(\mathbf{x}_{ij}, r))$ [dashed lines] and the within-class distance $d(\mathbf{x}_{ij}, \psi_l(\mathbf{x}_{ij}, i))$ [solid lines] for the training i-vectors $\mathbf{x}_{ij}$ of speaker i. For $\mathbf{x}_{i1}$ and $\mathbf{x}_{i2}$, the weighting function $\omega(\cdot)$ in Eq. 11 approaches 0.5. For $\mathbf{x}_{i3}$, $\omega(\cdot)$ approaches 0.0.

$d(\mathbf{x}_{ij}, \psi_l(\mathbf{x}_{ij}, r))$ [dashed line] is comparable to the within-class distance $d(\mathbf{x}_{ij}, \psi_l(\mathbf{x}_{ij}, i))$ [solid line], causing $g(\mathbf{x}_{ij})$ to approach 1.0. Therefore, the weighting function $\omega(\cdot)$ in Eq. 11 approaches 0.5. On the other hand, if the selected i-vector $\mathbf{x}_{ij}$ is far away from a speaker boundary, e.g, $\mathbf{x}_{i3}$, the between-class distance $d(\mathbf{x}_{ij}, \psi_l(\mathbf{x}_{ij}, r))$ will be much larger than the between-class distance $d(\mathbf{x}_{ij}, \psi_l(\mathbf{x}_{ij}, i))$, causing the weighting function $\omega(\cdot)$ to approach 0.0. As a result, the discriminative speaker boundary information in the training set can be emphasized by the weighting function.

Similar to the standard LDA, the NFA projection matrix comprises the eigenvectors of $(\mathbf{S}_w^{\text{NFA}})^{-1}\mathbf{S}_b^{\text{NFA}}$.

Compared with the traditional LDA, both the within-class and between-class scatter matrices in NFA have a nonparametric form. Instead of using the speaker-dependent i-vector means, NFA selects some of the nearest samples from the same speaker or from other speakers to calculate the scatter matrices, making the NFA more capable of capturing the structural information of speaker boundaries than LDA. Moreover, the contribution of each neighbor sample to the scatter matrix is controlled by the ratio between the inter-speaker distance to intra-speaker distance, which effectively increases the influence of the i-vectors that are close to the inter-speaker boundaries.

B. SNR-Invariant PLDA Modeling

In this section, we propose a new modeling approach, namely SNR-invariant PLDA, for robust speaker verification. Unlike Gaussian PLDA, SNR-invariant PLDA has two labeled latent factors representing SNR-specific and identity-specific information, respectively.

1) Generative Model: The SNR-invariant PLDA model is inspired by the notion of Gaussian PLDA in which i-vectors from the same speaker should share an identical latent identity factor. Similarly, we assume that i-vectors derived from utterances that fall within a narrow SNR range should share similar SNR-specific information. From a modeling standpoint, both SNR-specific and identity-specific information can be captured using latent factors. We refer to these latent factors as SNR factor and identity factor in the sequel.

Under the above assumptions, an NFA-projected[1] i-vector can be regarded as an observation generated from a linear generative model that comprises three components: (1) SNR component, (2) identity component, and (3) channel variability and the remaining variability that cannot be captured by the first two components. Assume that we have a set of D-dimension reduced i-vectors $\mathcal{X} = \{\hat{\mathbf{x}}_{ij}^k | i = 1, \ldots, S; j = 1, \ldots, H_i(k); k = 1, \ldots, K\}$ obtained from S speakers in NFA subspace, where $\hat{\mathbf{x}}_{ij}^k$ is the *j*-th sample from speaker *i* at the *k*-th SNR sub-group. In the SNR-invariant PLDA model, $\hat{\mathbf{x}}_{ij}^k$ can be expressed as:

$$\hat{\mathbf{x}}_{ij}^k = \mathbf{m} + \mathbf{V}\mathbf{h}_i + \mathbf{U}\mathbf{w}_k + \boldsymbol{\epsilon}_{ij}^k \tag{12}$$

where $\mathbf{m}$ is a $D \times 1$ vector representing the global offset, $\mathbf{h}_i$ is a $P \times 1$ vector denoting the latent identity factor with prior distribution $\mathcal{N}(\mathbf{0}, \mathbf{I})$, $\mathbf{w}_k$ is a $Q \times 1$ vector denoting the latent SNR factor with a prior distribution of $\mathcal{N}(\mathbf{0}, \mathbf{I})$, $\boldsymbol{\epsilon}_{ij}^k$ is a $D \times 1$ vector denoting the residual which follows a Gaussian distribution $\mathcal{N}(\mathbf{0}, \boldsymbol{\Sigma})$, $\mathbf{V}$ is a $D \times P$ matrix whose columns span the speaker subspace, and $\mathbf{U}$ is a $D \times Q$ matrix whose columns span the SNR subspace. $\mathbf{h}_i$ and $\mathbf{w}_k$ are assumed to be statistically independent.

The proposed SNR-invariant PLDA is different from the conventional PLDA in that the former makes use of multiple labels (speaker IDs and SNR levels) for training the loading matrices, whereas the latter only uses the speaker IDs. To exploit the SNR information in the training utterances, the SNR-invariant PLDA has an additional subspace called SNR subspace, which results in an extra latent factor called SNR factor. Unlike the term $\mathbf{G}\mathbf{r}_{ij}$ in Eq. 7, which is speaker- and session-dependent, the SNR component $\mathbf{U}\mathbf{w}_k$ in Eq. 12 depends on the SNR sub-groups. For instance, i-vectors within the k-th SNR sub-group will share the same SNR factor $\mathbf{w}_k$.

2) EM Algorithm: Denote $\boldsymbol{\theta} = \{\mathbf{m}, \mathbf{V}, \mathbf{U}, \boldsymbol{\Sigma}\}$ as the parameters of the SNR-invariant PLDA model. These parameters can be learned from a training set using maximum likelihood estimation. Given an initial value $\boldsymbol{\theta}$, we aim to find a new estimate $\boldsymbol{\theta}'$ that maximizes the auxiliary function:

$$\begin{aligned}\mathbf{Q}(\boldsymbol{\theta}'|\boldsymbol{\theta}) &= \mathbb{E}_{\mathbf{h},\mathbf{w}}\left\{\ln p(\mathcal{X}, \mathbf{h}, \mathbf{w}|\boldsymbol{\theta}') \middle| \mathcal{X}, \boldsymbol{\theta}\right\} \\ &= \mathbb{E}_{\mathbf{h},\mathbf{w}}\left\{\sum\nolimits_{ijk} \ln\left[p(\hat{\mathbf{x}}_{ij}^k|\mathbf{h}_i, \mathbf{w}_k, \boldsymbol{\theta}')p(\mathbf{h}_i, \mathbf{w}_k)\right] \middle| \mathcal{X}, \boldsymbol{\theta}\right\}\end{aligned} \tag{13}$$

To maximize Eq.13, we need to estimate the posterior distributions of the latent variables given the model parameters $\boldsymbol{\theta}$. Denote $N_i = \sum_{k=1}^{K} H_i(k)$ as the number of training samples from the *i*-th speaker and $M_k = \sum_{i=1}^{S} H_i(k)$ as the number of training samples falling in the *k*-th SNR group. Then the E-step is as follows:

$$\mathbf{L}_i^1 = \mathbf{I} + N_i \mathbf{V}^\top \boldsymbol{\Phi}_1^{-1} \mathbf{V} \qquad i = 1, \ldots, S \tag{14}$$

[1]The same formulations also applied to LDA-projected i-vectors.

$$\mathbf{L}_k^2 = \mathbf{I} + M_k \mathbf{U}^\top \mathbf{\Phi}_2^{-1} \mathbf{U} \qquad k = 1, \ldots, K \tag{15}$$

$$\langle \mathbf{h}_i | \mathcal{X} \rangle = (\mathbf{L}_i^1)^{-1} \mathbf{V}^\top \mathbf{\Phi}_1^{-1} \sum_{k=1}^{K} \sum_{j=1}^{H_i(k)} (\hat{\mathbf{x}}_{ij}^k - \mathbf{m}) \tag{16}$$

$$\langle \mathbf{w}_k | \mathcal{X} \rangle = (\mathbf{L}_k^2)^{-1} \mathbf{U}^\top \mathbf{\Phi}_2^{-1} \sum_{i=1}^{S} \sum_{j=1}^{H_i(k)} (\hat{\mathbf{x}}_{ij}^k - \mathbf{m}) \tag{17}$$

$$\langle \mathbf{h}_i \mathbf{h}_i^\mathsf{T} | \mathcal{X} \rangle = (\mathbf{L}_i^1)^{-1} + \langle \mathbf{h}_i | \mathcal{X} \rangle \langle \mathbf{h}_i | \mathcal{X} \rangle^\mathsf{T} \tag{18}$$

$$\langle \mathbf{w}_k \mathbf{w}_k^\mathsf{T} | \mathcal{X} \rangle = (\mathbf{L}_k^2)^{-1} + \langle \mathbf{w}_k | \mathcal{X} \rangle \langle \mathbf{w}_k | \mathcal{X} \rangle^\mathsf{T} \tag{19}$$

$$\langle \mathbf{w}_k \mathbf{h}_i^\mathsf{T} | \mathcal{X} \rangle = \langle \mathbf{w}_k | \mathcal{X} \rangle \langle \mathbf{h}_i | \mathcal{X} \rangle^\mathsf{T} \tag{20}$$

$$\langle \mathbf{h}_i \mathbf{w}_k^\mathsf{T} | \mathcal{X} \rangle = \langle \mathbf{h}_i | \mathcal{X} \rangle \langle \mathbf{w}_k | \mathcal{X} \rangle^\mathsf{T} \tag{21}$$

where

$$\mathbf{\Phi}_1 = \mathbf{U}\mathbf{U}^\top + \mathbf{\Sigma}, \quad \mathbf{\Phi}_2 = \mathbf{V}\mathbf{V}^\top + \mathbf{\Sigma}$$

and $\langle . \rangle$ denotes expectation.

Given Eq. 14–Eq. 21, the model parameters $\boldsymbol{\theta}'$ can be estimated via the M-step as follows:

$$\mathbf{m} = \frac{1}{N} \sum_{i=1}^{S} \sum_{k=1}^{K} \sum_{j=1}^{H_i(k)} \hat{\mathbf{x}}_{ij}^k \tag{22}$$

$$\mathbf{V}' = \left\{ \sum_{i=1}^{S} \sum_{k=1}^{K} \sum_{j=1}^{H_i(k)} \left[(\hat{\mathbf{x}}_{ij}^k - \mathbf{m}) \langle \mathbf{h}_i | \mathcal{X} \rangle - \mathbf{U} \langle \mathbf{w}_k \mathbf{h}_i^\mathsf{T} | \mathcal{X} \rangle \right] \right\} \left[\sum_{i=1}^{S} \sum_{k=1}^{K} \sum_{j=1}^{H_i(k)} \langle \mathbf{h}_i \mathbf{h}_i^\mathsf{T} | \mathcal{X} \rangle \right]^{-1} \tag{23}$$

$$\mathbf{U}' = \left\{ \sum_{i=1}^{S} \sum_{k=1}^{K} \sum_{j=1}^{H_i(k)} \left[(\hat{\mathbf{x}}_{ij}^k - \mathbf{m}) \langle \mathbf{w}_k | \mathcal{X} \rangle - \mathbf{V} \langle \mathbf{h}_i \mathbf{w}_k^\mathsf{T} | \mathcal{X} \rangle \right] \right\} \left[\sum_{i=1}^{S} \sum_{k=1}^{K} \sum_{j=1}^{H_i(k)} \langle \mathbf{w}_k \mathbf{w}_k^\mathsf{T} | \mathcal{X} \rangle \right]^{-1} \tag{24}$$

$$\mathbf{\Sigma}' = \frac{1}{N} \sum_{i=1}^{S} \sum_{k=1}^{K} \sum_{j=1}^{H_i(k)} \Big[(\hat{\mathbf{x}}_{ij}^k - \mathbf{m})(\hat{\mathbf{x}}_{ij}^k - \mathbf{m})^\top - \mathbf{V} \langle \mathbf{h}_i | \mathcal{X} \rangle (\hat{\mathbf{x}}_{ij}^k - \mathbf{m})^\top - \mathbf{U} \langle \mathbf{w}_k | \mathcal{X} \rangle (\hat{\mathbf{x}}_{ij}^k - \mathbf{m})^\top \Big] \tag{25}$$

where $N = \sum_{i=1}^{S} N_i = \sum_{k=1}^{K} M_k$. Algorithm 1 shows the procedures of applying the EM algorithm and APPENDIX A shows the derivations of Eq. 14–Eq. 25.

C. Likelihood Ratio Scores

Given a test sample $\hat{\mathbf{x}}_t$ and a target sample $\hat{\mathbf{x}}_s$ in the NFA subspace, the likelihood ratio score can be computed as follows:

$$\begin{aligned} L(\hat{\mathbf{x}}_s, \hat{\mathbf{x}}_t) &= \ln \frac{P(\hat{\mathbf{x}}_s, \hat{\mathbf{x}}_t | \text{same-speaker})}{P(\hat{\mathbf{x}}_s, \hat{\mathbf{x}}_t | \text{different-speakers})} \\ &= \text{const} + \frac{1}{2} \hat{\mathbf{x}}_s^\top \mathbf{Q} \hat{\mathbf{x}}_s + \frac{1}{2} \hat{\mathbf{x}}_t^\top \mathbf{Q} \hat{\mathbf{x}}_t + \hat{\mathbf{x}}_s^\top \mathbf{P} \hat{\mathbf{x}}_t \end{aligned} \tag{26}$$

Algorithm 1 EM Algorithm for SNR-Invariant PLDA

Input:

Development data set consists of NFA-projected i-vectors $\mathcal{X} = \{\hat{\mathbf{x}}_{ij}^k | i = 1, \ldots, S; j = 1, \ldots, H_i(k); k = 1, \ldots, K\}$, with identity labels and SNR group labels.

Initialization:

$\mathbf{\Sigma} \leftarrow 0.01 \times \mathbf{I}$;

$\mathbf{V}, \mathbf{U} \leftarrow$ eigenvectors of PCA projection matrix learned using data set $\mathcal{X}$;

Parameter Estimation:

1) Compute $\mathbf{m}$ via Eq. 22;
2) Compute $\mathbf{L}_i^1$ and $\mathbf{L}_k^2$ according to Eq. 14 and Eq. 15;
3) Compute the sufficient statistics using Eq. 16 to Eq. 21;
4) Update the model parameters using Eq. 23 to Eq. 25;
5) Go to step 2 until convergence;

Return: the parameters of the SNR-invariant PLDA model $\boldsymbol{\theta} = \{\mathbf{m}, \mathbf{V}, \mathbf{U}, \mathbf{\Sigma}\}$.

where

$$\begin{aligned} \mathbf{P} &= \mathbf{\Sigma}_{tot}^{-1} \mathbf{\Sigma}_{ac} (\mathbf{\Sigma}_{tot} - \mathbf{\Sigma}_{ac} \mathbf{\Sigma}_{tot}^{-1} \mathbf{\Sigma}_{ac})^{-1}, \\ \mathbf{Q} &= \mathbf{\Sigma}_{tot}^{-1} - (\mathbf{\Sigma}_{tot} - \mathbf{\Sigma}_{ac} \mathbf{\Sigma}_{tot}^{-1} \mathbf{\Sigma}_{ac})^{-1}, \\ \mathbf{\Sigma}_{ac} &= \mathbf{V}\mathbf{V}^\top, \quad \text{and} \quad \mathbf{\Sigma}_{tot} = \mathbf{V}\mathbf{V}^\top + \mathbf{U}\mathbf{U}^\top + \mathbf{\Sigma}. \end{aligned}$$

See APPENDIX B for the derivations of Eq. 26.

IV. Experimental Setup

A. Speech Data and Front-End Processing

All experiments were performed on the core set of NIST 2012 Speaker Recognition Evaluation (SRE)[42]. We divided the speech data into three categories: (1) development data, (2) enrollment data, and (3) test data.

- *Development Data*: The microphone and telephone speech files from NIST 2005–2008 SREs were used as development data to train the gender-dependent UBMs and total variability matrices. The telephone and microphone speech files in 2006–2010 SREs, excluding speakers with less than 2 utterances, were used to train the PLDA and SNR-invariant PLDA models. Subsets of these speakers were used to train the LDA, NFA, and WCCN projection matrices. The composition of these subsets will be elaborated in Section V. The speaker labels in the development data were obtained from the target-speaker table files in NIST 2012 SRE.[2]
- *Enrollment Data*: Enrollment data comprise the conversations of target speakers, as defined by the speaker-table files in NIST 2012 SRE. Each target speaker has one or more conversations recorded over different channels and with different durations. All of the 10-second utterances and summed-channel utterances were removed from the target segments. But we ensure that each target speaker has at least one utterance for enrollment.

[2]Starting from 2012 SRE, it is legitimate to use target speakers as development data. In fact, the speakers in the target-speaker table are speakers from 2006–2010 SREs.

- *Test Data*: All test data were extracted from NIST 2012 SRE, as defined by the `core.ndx` file in the evaluation plan. This paper focuses on common conditions (CC) 2, 4, and 5 of the evaluation plan. Table I shows the conditions of the test segments under these common conditions.

A two-channel VAD [43], [44] was applied to detect the speech regions of each utterance. 19 Mel frequency cepstral coefficients together with log energy plus their 1st- and 2nd-derivatives were extracted from the speech regions as detected by the VAD, followed by cepstral mean normalization [45] and feature warping [46] with a window size of 3 seconds. A 60-dim acoustic vector was extracted every 10ms, using a Hamming window of 25ms.

B. Creating Noisy Speech for Multi-Condition Training

The SNR distributions of test utterances in CC2, CC4, and CC5 are shown in Fig. 2. Because the SNR range of the test utterances in CC4 is much wider than that of CC2 and CC5, the SNR mismatch between the training and the test utterances has significant effect on the test trials in CC4. To address this issue, we added noise to the telephone data of the training data. Specifically, for each telephone speech file, a noise waveform file was randomly selected from the 30 noise waveform files in the PRISM data set [47] and added to the file at a target SNR using the FaNT tool [48]. The target SNR was chosen from a target SNR set $\{6\text{dB}, 7\text{dB}, \ldots, 15\text{dB}\}$ in turn so that 10 noise-corrupted files were produced for each clean file.

For experiments on CC4, 14,226 noise corrupted files from male speakers and 22,356 noise corrupted files from female speakers were selected randomly and combined with the original (tel+mic) training data to train the gender-dependent subspace projection matrices, SNR-invariant PLDA models and Gaussian PLDA models. The ratio between the numbers of selected noise corrupted files and original files is 1:1. As shown in Fig.3, the selected noise corrupted files have a fairly flat distribution of SNR. A flat SNR distribution is desirable because it ensures that the resulting PLDA model will not bias towards a specific SNR.

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 [43], [44] as follows. Given a speech file, we passed the waveform to the G.712 frequency weighting filter in FaNT and then estimated the speech energy using the voltmeter function (`sv-p56.c` from the ITU-T Software Tool Library [49]). 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. While the measured SNR is close to the target SNR, they will not be exactly the same. This explains why we have a continuous SNR distribution in Fig. 3. Unless stated otherwise, all SNR in the sequel means measured SNR.

TABLE I
TEST SEGMENT CONDITIONS FOR CC2, CC4, AND CC5 OF NIST 2012 SRE.

Common Condition	Test Segment Conditions
CC2	Phone call speech.
CC4	Phone call speech with added noise.
CC5	Phone call speech intentionally collected in a noisy environment.

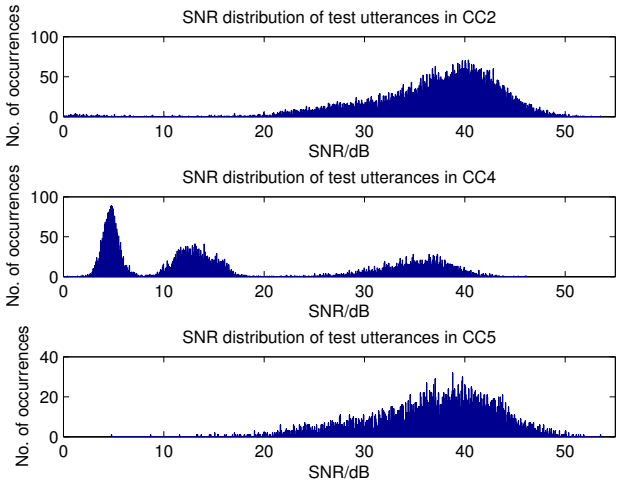


Fig. 2. SNR distributions of test utterances in CC2, CC4 and CC5 of NIST 2012 SRE.

C. I-vector Preprocessing

I-vectors were extracted based on gender-dependent UBMs with 1024 mixtures and total variability matrices with 500 total factors. Similar to [50], we applied within-class covariance normalization (WCCN) [11] to whiten the i-vectors, followed by length normalization (LN) to reduce the non-Gaussian behavior of the 500-dimensional i-vectors. Then, nonparametric feature analysis (NFA) or LDA was applied to reduce intra-speaker variability and emphasize discriminative information. This procedure projects the i-vectors onto a 200-dimensional subspace so that the amount of training data should be sufficient to estimate a reliable NFA-projected matrix or LDA-projected matrix. Then SNR-invariant PLDA and Gaussian PLDA models with 150 latent identity factors were trained.

In the following, we refer to the i-vector/PLDA framework in which the i-vectors have gone through WCCN+LN+LDA+WCCN as conventional PLDA system. Similarly, we refer to the i-vector/PLDA framework in which the i-vectors have gone through WCCN+LN+NFA as N-PLDA. We used WCCN+LN+NFA rather than WCCN+LN+NFA+WCCN because we found that the transformation matrix corresponding to the last WCCN in the processing pipeline is close to an identity matrix. As a result, it does not have effect on the NFA-projected i-vectors. The proposed framework of SNR-invariant PLDA in nonparametric subspace is referred to as NS-PLDA. As a comparison, SNR-invariant PLDA modeling in the subspace

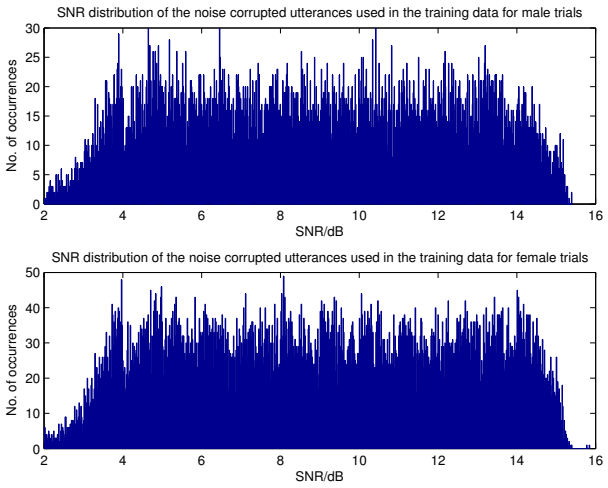


Fig. 3. SNR distribution of the noise corrupted utterances used in the training data for the experiments on CC4.

reduced by WCCN+LN+LDA+WCCN is named as S-PLDA. Also, the SNR-dependent mixture of PLDA in [35] was used as a comparison, which is named as mPLDA in the sequel. Table II summarizes the nomenclatures of various methods.

TABLE II
NOMENCLATURES OF VARIOUS PLDA MODELS. *WCCN*: WITHIN-CLASS COVARIANCE ANALYSIS; *LN*: LENGTH NORMALIZATION; *LDA*: LINEAR DISCRIMINANT ANALYSIS; *NFA*: NONPARAMETRIC FEATURE ANALYSIS; *mPLDA*: SNR-DEPENDENT MIXTURE OF PLDA; *SI-PLDA*: SNR-INVARIANT PLDA.

Method Name	I-vector Preprocessing	Generative Model
PLDA	WCCN+LN+LDA+WCCN	PLDA (Eq. 7)
mPLDA	WCCN+LN+LDA+WCCN	mPLDA in [35]
N-PLDA	WCCN+LN+NFA	PLDA (Eq. 7)
S-PLDA	WCCN+LN+LDA+WCCN	SI-PLDA (Eq. 12)
NS-PLDA	WCCN+LN+NFA	SI-PLDA (Eq. 12)

V. RESULTS AND ANALYSIS

In this section, we evaluate the performance of different systems using equal error rate (EER), minimum DCF (minDCF) [42] and DET curves [51].

A. Effects of Nonparametric Feature Analysis

This experiment aims to investigate the contribution of nonparametric feature analysis on the discriminative power of preprocessed i-vectors. To this end, the performance of PLDA and N-PLDA is compared using the experimental results on CC2 and CC5 of NIST 2012 SRE. Because test utterances in CC2 and CC5 have high SNRs (see Fig. 2), we only used the original (tel+mic) utterances in NIST 2006–2010 SREs to train the models, excluding speakers with less than two utterances. For the NFA, we used a subset of the development data set in which each speaker has at least 16 utterances, which amounts to 431 male and 469 female speakers.

Results in Table III show that N-PLDA outperforms PLDA on both common conditions, suggesting that NFA is a better i-vector preprocessing method than LDA. In addition, N-PLDA can achieve good performance with different numbers of nearest neighbors. However, when k_1 and k_2 are very small or very large (smaller than 3 or larger than 12), the performance of N-PLDA degrades. The reason is that if k_1 and k_2 are too small, there will not be enough within-class and between-class neighboring i-vectors to estimate the scatter matrices in Eq. 9 and Eq. 10, leading to inaccurate NFA-projection matrices. On the other hand, when the number of nearest neighbors is too large (say $k_1 = k_2 = 16$), the merit of nearest neighbours in NFA is lost. This is because a large value of k_1 means that the within-speaker scatter matrix in Eq. 9 is based on intra-speaker distances rather than the distances between the closest i-vectors of the same speaker. Similarly, a large value of k_2 means that the between-speaker scatter matrix in Eq. 10 will be based on some non-boundary i-vectors, which defeats the purpose of NFA.

Note that k_1 and k_2 are speaker-independent. Therefore, by selecting a value for these parameters within a legitimate range, the value can be applied to all training speakers. Possible range for k_1 and k_2 is $[1, \min(N_i)]$, where N_i is the number of sessions from speaker i. Because we used speakers with at least 16 sessions for training the NFA-projected matrices, $\min(N_i) = 16$. We found that the middle of this range (i.e., $k_1 = k_2 = 8$) is appropriate for Eq. 9 and Eq. 10 to obtain good estimates of the scatter matrices.

Table III shows that the gain of NFA is higher for male than for female. This is primarily caused by the abundant of female data available for estimating the LDA and NFA projection matrices. According to Eqs. 3 and 4, LDA requires estimating the speaker-dependent means $\boldsymbol{\mu}_i$'s for all speakers. Any speakers with a limited number of sessions will lead to inaccurate $\boldsymbol{\mu}_i$ and thus inaccurate covariance. On the other hand, because NFA does not compute speaker-dependent means, it is still effective even if the number of sessions of some speakers is small. For female, the effect of data scarcity on LDA is less prominent because the number of female speakers is 469, as opposed to 431 male speakers. Therefore, benefit of NFA is less prominent for female.

B. Performance of SNR-Invariant PLDA

In this subsection, we report results on CC2, CC4 and CC5 to compare the proposed framework with the state-of-the-art framework.

For experiments on CC4, the original (tel+mic) in NIST 2006–2010 SREs together with the selected noise corrupted telephone utterances described in Section IV-A were used to train the gender-dependent SNR-invariant PLDA models and conventional PLDA models. Speakers with less than two utterances were excluded in the training data. For the training of gender-dependent NFA projection matrices, we used a subset of the corresponding training data set in which each speaker has at least 24 utterances (including both original and noise contaminated ones). The numbers of nearest neighbors (k_1 and k_2 in Eq. 9 and Eq. 10) used for computing the

TABLE III
PERFORMANCE OF PLDA AND N-PLDA ON CC2 AND CC5 OF NIST 2012 SRE CORE SET. k_1 AND k_2 ARE THE PARAMETERS IN EQ. 9 AND EQ. 10, WHICH CONTROL THE NUMBERS OF THE NEAREST NEIGHBORS IN NFA.

Method	k_1&k_2	Male				Female			
		CC2		CC5		CC2		CC5	
		EER(%)	minDCF	EER(%)	minDCF	EER(%)	minDCF	EER(%)	minDCF
PLDA	–	2.40	0.332	2.80	0.303	2.19	0.335	2.34	0.331
N-PLDA	4	2.12	0.301	2.54	0.303	1.90	0.341	2.30	0.325
	8	**2.04**	0.304	**2.48**	**0.302**	**1.88**	0.334	**2.21**	**0.321**
	12	2.17	**0.296**	2.54	0.325	1.91	**0.333**	2.26	0.321
	16	2.25	0.332	2.57	0.322	2.10	0.353	2.28	0.323

TABLE V
PERFORMANCE OF PLDA, MPLDA, N-PLDA, S-PLDA, AND NS-PLDA ON CC4 OF NIST 2012 SRE (CORE SET). K IS THE NUMBER OF SNR SUB-GROUPS AND Q IS THE DIMENSION OF SNR FACTORS IN SNR-INVARIANT PLDA. SEE TABLE II FOR THE I-VECTOR PREPROCESSING FOR DIFFERENT METHODS.

Training Data	Method	K	Q	Male		Female	
				EER(%)	minDCF	EER(%)	minDCF
Original (tel+mic)	PLDA	–	–	3.93	0.375	3.83	0.423
Original (tel+mic) and noise corrupted tel	PLDA	–	–	3.39	0.325	3.10	0.354
	mPLDA in [35]	–	–	2.88	0.329	2.71	0.332
	N-PLDA	–	–	3.13	0.312	2.82	0.341
	S-PLDA	3	40	3.20	0.300	2.95	0.327
		5	40	3.10	0.296	2.97	0.326
		6	40	3.02	0.296	2.93	0.320
		7	30	3.13	0.302	2.93	0.326
		8	30	3.24	0.306	2.89	0.329
	NS-PLDA	3	40	2.72	0.289	2.36	0.314
		5	40	2.67	0.291	2.38	0.322
		6	10	**2.63**	0.288	2.44	0.320
		6	40	**2.63**	**0.287**	2.43	0.319
		6	50	2.67	0.289	2.45	0.318
		7	30	**2.63**	0.294	2.32	0.316
		8	10	2.72	0.291	**2.24**	**0.313**
		8	30	2.70	0.292	2.29	**0.313**
		8	50	2.76	0.290	2.33	0.318

NFA projection matrix were set to 12 and 10 for male and female, respectively. Based on the argument in Section V-A, these numbers are half of the minimum number of sessions per training speakers.

In order to verify the effectiveness of SNR invariant PLDA, the same preprocessing and NFA projection were applied in both N-PLDA systems and NS-PLDA systems. To train the SNR-invariant PLDA models in S-PLDA systems and NS-PLDA systems, the training data set (including the original and the noise corrupted utterances in Fig. 3) was divided into K groups according to the measured SNRs of the utterances. Specifically, the measured SNRs of the whole training data set were divided into K intervals such that each interval corresponds to one SNR sub-group. The k-th SNR sub-group comprises the i-vectors whose corresponding utterances have SNR falling in the k-th SNR interval. For example, when $K = 8$, the SNR divisions for the training data and the number of training utterances falling in each of the divisions are shown in Table IV. The intervals were set such that they progressively increase when SNR increases.

In our experiments, we make sure that the number of speaker factors plus SNR factors is no more than 200. Because the speaker factor was set to 150, we set the SNR factor (Q) to 40 or 30 according to the number of SNR sub-groups (K). To investigate the effect of varying the number of SNR factors, we set $Q = 10$ and $Q = 50$ for fixed K. The results suggest that the performance does not vary significantly when we vary Q from 10 to 50.

Results in Table V show that incorporating noise corrupted telephone utterances into the training data can improve the system performance on CC4. Comparing different methods, we observed that N-PLDA, mPLDA, S-PLDA, and NS-PLDA outperform the PLDA, and the best result was achieved by NS-PLDA. Moreover, the performance of S-PLDA and NS-PLDA stays stable when the number of SNR groups increases from 5 to 8. Comparing mPLDA and S-PLDA, we can see that S-PLDA achieves a lower minDCF, while, mPLDA achieves a lower EER. The results also suggest that NFA can improve the performance of PLDA and SNR-invariant PLDA when it is used as a preprocessor and that S-PLDA and NS-PLDA can address SNR mismatch under noisy conditions.

For SNR-invariant PLDA, it is important to determine an

TABLE VI
PERFORMANCE OF PLDA, MPLDA, N-PLDA, S-PLDA, AND NS-PLDA ON CC2 AND CC5 OF NIST 2012 SRE (CORE SET). K IS THE NUMBER OF SNR SUB-GROUPS IN SNR-INVARIANT PLDA. SEE TABLE II FOR THE I-VECTOR PREPROCESSING FOR DIFFERENT METHODS.

Method	K	Male				Female			
		CC2		CC5		CC2		CC5	
		EER(%)	minDCF	EER(%)	minDCF	EER(%)	minDCF	EER(%)	minDCF
PLDA	–	2.40	0.332	2.80	0.303	2.19	0.335	2.34	0.331
mPLDA	–	2.47	0.283	2.80	0.287	2.07	0.328	2.46	0.342
N-PLDA	–	2.04	0.304	2.48	0.302	1.88	0.334	2.21	0.321
S-PLDA	3	2.38	0.316	2.80	0.302	1.90	0.303	2.37	0.319
	6	2.47	0.314	2.93	0.304	1.91	0.311	2.42	0.318
NS-PLDA	3	**1.96**	**0.277**	**2.47**	**0.273**	1.74	**0.290**	2.07	**0.294**
	6	1.99	0.278	2.48	0.275	**1.72**	**0.290**	**2.04**	**0.294**

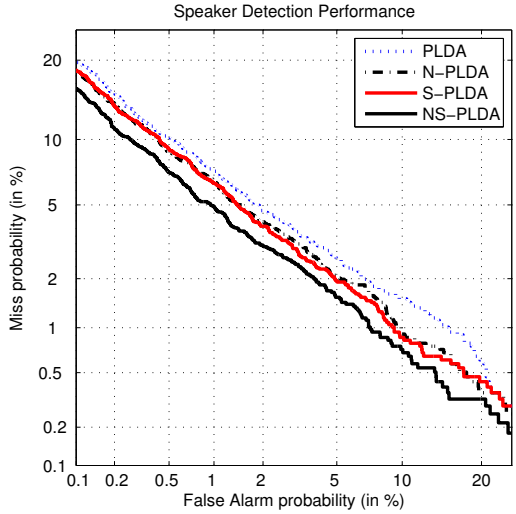


Fig. 4. The DET Performance of PLDA, N-PLDA, S-PLDA ($K = 6$), and NS-PLDA ($K = 6$) for male speakers on CC4 of NIST 2012 SRE (core set). See Table II for the nomenclatures in the legend.

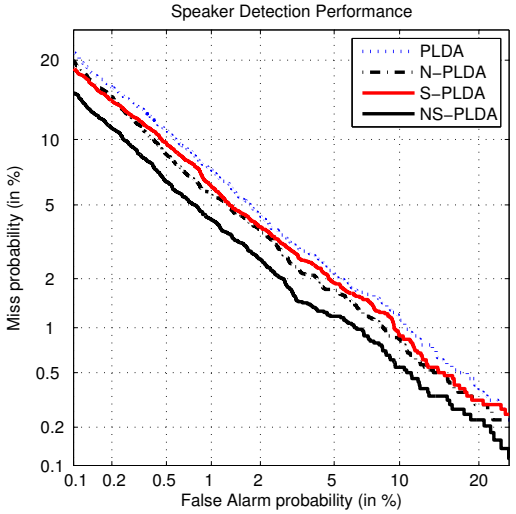


Fig. 5. The DET Performance of PLDA, N-PLDA, S-PLDA ($K = 6$), and NS-PLDA ($K = 8$) for female speakers on CC4 of NIST 2012 SRE (core set). See Table II for the nomenclatures in the legend.

appropriate value of K. In particular, in the two extreme cases where K is either very small (≤ 3) or very large (same as the number of training i-vectors), the SNR-invariant PLDA will not be effective. This is because for the former, each of the SNR factors ($\mathbf{w}_k$ in Eq. 12) will need to represent the i-vectors with a wide range of SNR. On the other hand, for the latter case, there will be so many SNR factors in Eq. 12 that each i-vectors are considered to be obtained from a unique SNR. This means that in such situation the SNR-invariant PLDA model reduces to the traditional Gaussian PLDA, which only considers the session variability instead of the variability caused by different levels of SNR.

TABLE IV
SNR SUB-GROUP DIVISIONS FOR THE TRAINING DATA SET WHEN $K = 8$. THE NUMBER OF TRAINING UTTERANCES IN THE SNR SUB-GROUPS ($K = 8$ IN EQ. 12), WHERE AN SNR SUB-GROUP IS DEFINED BY THE SNR RANGE IN DB. THE TRAINING UTTERANCES WERE PRODUCED BY ADDING BABBLE NOISE TO THE ORIGINAL (CLEAN) TRAINING UTTERANCES AT SNR BETWEEN 6 TO 15 DB.

Sub-Group	SNR Range (dB)	No. of Utterances	
		Male	Female
1	2–4	3556	3866
2	4–6	3748	4346
3	6–8	4246	4734
4	8–11	5064	5224
5	11–15	5457	5889
6	15–20	3047	4684
7	20–35	3918	6020
8	> 35	4194	8534

Fig. 4 and Fig. 5 show the DET curves of different methods under CC4. For the curves of S-PLDA and NS-PLDA, we reported the results of male speakers for $K = 6$ in Table V. For female speakers, the results for $K = 6$ and $K = 8$ in Table V were reported. Again, NS-PLDA performs the best at all of the operating points in Fig. 4 and Fig. 5.

Because the test utterances in CC2 and CC5 have high SNRs and the corresponding SNR ranges are narrower than that in CC4, only the original (tel+mic) utterances in NIST 2006–2010 SREs were used to train the gender-dependent models for the experiments on CC2 and CC5. Results of different methods are listed in Table VI. For N-PLDA, the results were obtained when the number of nearest neighbors was set to 8, i.e., $k_1 =$

$k_2 = 8$ in Eqs. 9 and 10. The dimension of the SNR factor in S-PLDA and NS-PLDA was set to 40. Evidently, NS-PLDA achieves the best performance among all methods. S-PLDA outperforms PLDA when the number of SNR subgroups was set to 3, but its performance becomes poor when $K = 6$. The reason is that the SNR range of CC2 and CC5 is narrower than that of CC4, which does not require as many SNR sub-groups as that in CC4.

VI. Conclusion

A new framework, namely SNR-invariant PLDA, in which the pre-processed i-vectors are assumed to live in a non-parametric subspace has been proposed. The framework is designed to address the SNR mismatch in practical speaker verification. This work has two main contributions.

1) Nonparametric feature analysis was introduced to extract effective feature subset and discriminative boundary information. This is achieved by constructing two nonparametric scatter matrices.
2) An SNR-invariant PLDA modeling was proposed to deal with the mismatch caused by different levels of background noise. By assuming that the i-vectors share the same SNR-specific information when the corresponding utterances' SNRs fall within a narrow range, we incorporate an SNR factor to the conventional Gaussian PLDA model. In this model, both SNR-specific and identity-specific factors are learned by supervised learning and are compressed into two different subspaces.

The proposed framework was compared against state-of-the-art i-vector/PLDA systems on the NIST SRE 2012 data set. NFA outperformed LDA on three common conditions and was shown to be a suitable preprocessor for PLDA algorithm. The proposed NS-PLDA framework achieves much better performance than PLDA, N-PLDA, mPLDA, and S-PLDA systems in noisy situation. It is worth noting that the SNR-invariant PLDA model can be used to deal with cross channel problems as well.

Appendix A

To simplify notations, we use $\mathbf{x}_{ij}^k$ instead of $\hat{\mathbf{x}}_{ij}^k$ in Eq. 12 to represent LDA- or NFA-projected i-vectors.

The posterior density of $\mathbf{h}_i$ can be obtained according to the Bayesian rule:

$$\begin{aligned} p(\mathbf{h}_i|\mathcal{X},\boldsymbol{\theta}) &= \frac{p(\mathcal{X}|\mathbf{h}_i,\boldsymbol{\theta})p(\mathbf{h}_i)}{p(\mathcal{X})} \propto p(\mathcal{X}|\mathbf{h}_i,\boldsymbol{\theta})p(\mathbf{h}_i) \\ &= \prod_{k=1}^{K}\prod_{j=1}^{H_i(k)}\Big[p(\mathbf{x}_{ij}^k|\mathbf{h}_i,\boldsymbol{\theta})p(\mathbf{h}_i)\Big] \\ &= \prod_{k=1}^{K}\prod_{j=1}^{H_i(k)}\Big[\mathcal{N}(\mathbf{x}_{ij}^k|\mathbf{m}+\mathbf{V}\mathbf{h}_i,\boldsymbol{\Phi}_1)\mathcal{N}(\mathbf{h}_i|0,\mathbf{I})\Big] \\ &\propto \exp\Bigg\{\mathbf{h}_i^\top\mathbf{V}^\top\boldsymbol{\Phi}_1^{-1}\sum_{k=1}^{K}\sum_{j=1}^{H_i(k)}(\mathbf{x}_{ij}^k-\mathbf{m}) \\ &\quad -\frac{1}{2}\mathbf{h}_i^\top(\mathbf{I}+N_i\mathbf{V}^\top\boldsymbol{\Phi}_1^{-1}\mathbf{V})\mathbf{h}_i\Bigg\} \end{aligned} \tag{27}$$

where $\boldsymbol{\Phi}_1 = \mathbf{U}\mathbf{U}^\top + \boldsymbol{\Sigma}$ and $N_i = \sum_{k=1}^{K} H_i(k)$ is the number of i-vectors from the *i*-th speaker. Suppose $\mathbf{h}$ follows a Gaussian distribution $\mathcal{N}(\mathbf{h}|\boldsymbol{\mu},\mathbf{C})$, we can obtain the following property:

$$\begin{aligned} \mathcal{N}(\mathbf{h}|\boldsymbol{\mu},\mathbf{C}) &\propto \exp\Big\{-\frac{1}{2}(\mathbf{h}-\boldsymbol{\mu})^\top\mathbf{C}^{-1}(\mathbf{h}-\boldsymbol{\mu})\Big\} \\ &\propto \exp\Big\{\mathbf{h}^\top\mathbf{C}^{-1}\boldsymbol{\mu}-\frac{1}{2}\mathbf{h}^\top\mathbf{C}^{-1}\mathbf{h}\Big\}. \end{aligned} \tag{28}$$

Let us define

$$\mathbf{L}_i^1 \equiv \mathbf{I} + N_i\mathbf{V}^\top\boldsymbol{\Phi}_1^{-1}\mathbf{V}. \tag{29}$$

Then, comparing Eq. 27 and Eq. 28, the posterior mean and 2nd-order posterior moment of $\mathbf{h}_i$ can be estimated as:

$$\langle\mathbf{h}_i|\mathcal{X}\rangle = (\mathbf{L}_i^1)^{-1}\mathbf{V}^\top\boldsymbol{\Phi}_1^{-1}\sum_{k=1}^{K}\sum_{j=1}^{H_i(k)}(\mathbf{x}_{ij}^k-\mathbf{m}) \tag{30}$$

$$\langle\mathbf{h}_i\mathbf{h}_i^\mathsf{T}|\mathcal{X}\rangle = (\mathbf{L}_i^1)^{-1} + \langle\mathbf{h}_i|\mathcal{X}\rangle\langle\mathbf{h}_i|\mathcal{X}\rangle^\mathsf{T}. \tag{31}$$

Similarly, to compute the posterior mean and posterior moment of $\mathbf{w}_k$, we define

$$\mathbf{L}_k^2 \equiv \mathbf{I} + M_k\mathbf{U}^\top\boldsymbol{\Phi}_2^{-1}\mathbf{U} \tag{32}$$

where $\boldsymbol{\Phi}_2 = \mathbf{V}\mathbf{V}^\top + \boldsymbol{\Sigma}$ and $M_k = \sum_{i=1}^{S} H_i(k)$ is the number of i-vectors falling in the *k*-th SNR group. The posterior mean and 2nd-order posterior moment of $\mathbf{w}_k$ can be obtained as follows:

$$\langle\mathbf{w}_k|\mathcal{X}\rangle = (\mathbf{L}_k^2)^{-1}\mathbf{U}^\top\boldsymbol{\Phi}_2^{-1}\sum_{i=1}^{S}\sum_{j=1}^{H_i(k)}(\mathbf{x}_{ij}^k-\mathbf{m}) \tag{33}$$

$$\langle\mathbf{w}_k\mathbf{w}_k^\mathsf{T}|\mathcal{X}\rangle = (\mathbf{L}_k^2)^{-1} + \langle\mathbf{w}_k|\mathcal{X}\rangle\langle\mathbf{w}_k|\mathcal{X}\rangle^\mathsf{T} \tag{34}$$

To maximize $Q(\boldsymbol{\theta}'|\boldsymbol{\theta})$ in Eq. 13, we first maximize the log-likelihood function $\mathcal{L}(\mathcal{X};\boldsymbol{\theta}')$ with respect to $\boldsymbol{\theta}'$:

$$\begin{aligned} &\mathcal{L}(\mathcal{X};\boldsymbol{\theta}') \equiv \ln p(\mathcal{X},\mathbf{h},\mathbf{w}|\boldsymbol{\theta}') = \ln\Big[p(\mathcal{X}|\mathbf{h},\mathbf{w},\boldsymbol{\theta}')p(\mathbf{h},\mathbf{w})\Big] \\ &= \sum_{i,k,j}\ln\Big[p(\mathbf{x}_{ij}^k|\mathbf{h}_i,\mathbf{w}_k,\boldsymbol{\theta}')p(\mathbf{h}_i)p(\mathbf{w}_k)\Big] \\ &= \sum_{i,k,j}\Big[\ln\mathcal{N}(\mathbf{x}_{ij}^k|\mathbf{m}+\mathbf{V}'\mathbf{h}_i+\mathbf{U}'\mathbf{w}_k,\boldsymbol{\Sigma}') \\ &\qquad +\ln\mathcal{N}(\mathbf{h}_i|0,\mathbf{I})+\ln\mathcal{N}(\mathbf{w}_k|0,\mathbf{I})\Big] \\ &\propto -\frac{1}{2}\sum_{i,k,j}\Big[\ln|\boldsymbol{\Sigma}'| + (\mathbf{x}_{ij}^k-\mathbf{m}-\mathbf{V}'\mathbf{h}_i-\mathbf{U}'\mathbf{w}_k)^\top \\ &\qquad \boldsymbol{\Sigma}'^{-1}(\mathbf{x}_{ij}^k-\mathbf{m}-\mathbf{V}'\mathbf{h}_i-\mathbf{U}'\mathbf{w}_k)+\mathbf{h}_i^\top\mathbf{h}_i+\mathbf{w}_k^\top\mathbf{w}_k\Big] \end{aligned} \tag{35}$$

Differentiating Eq. 35 with respect to $\mathbf{V}'$, $\mathbf{U}'$, and $\boldsymbol{\Sigma}'$, followed by setting $\langle\frac{\partial\mathcal{L}}{\partial\mathbf{V}'}|\mathcal{X},\boldsymbol{\theta}\rangle = 0$, $\langle\frac{\partial\mathcal{L}}{\partial\mathbf{U}'}|\mathcal{X},\boldsymbol{\theta}\rangle = 0$, and $\langle\frac{\partial\mathcal{L}}{\partial\boldsymbol{\Sigma}'}|\mathcal{X},\boldsymbol{\theta}\rangle = 0$, we obtain Eq. 23–Eq. 25.

Appendix B

To simplify notations, we use $\mathbf{x}_s$ and $\mathbf{x}_t$ instead of $\hat{\mathbf{x}}_s$ and $\hat{\mathbf{x}}_t$ in Eq. 26 to represent the NFA-projected i-vectors. If $\mathbf{x}_s$

and $\mathbf{x}_t$ are from the same speaker, then we have

$$\begin{bmatrix} \mathbf{x}_s \\ \mathbf{x}_t \end{bmatrix} = \begin{bmatrix} \mathbf{m} \\ \mathbf{m} \end{bmatrix} + \begin{bmatrix} \mathbf{V} & \mathbf{U} & \mathbf{0} \\ \mathbf{V} & \mathbf{0} & \mathbf{U} \end{bmatrix} \begin{bmatrix} \mathbf{h} \\ \mathbf{w}_s \\ \mathbf{w}_t \end{bmatrix} + \begin{bmatrix} \boldsymbol{\epsilon}_s \\ \boldsymbol{\epsilon}_t \end{bmatrix} \quad (36)$$

where $\mathbf{h}$ represents the speaker factor shared by both i-vectors and $\mathbf{w}_s$ and $\mathbf{w}_t$ represent the SNR factors of the two utterances, respectively. Eq. 36 can be written in a compact form:

$$\tilde{\mathbf{x}}_{st} = \tilde{\mathbf{m}} + \tilde{\mathbf{A}}\tilde{\mathbf{z}}_{st} + \tilde{\boldsymbol{\epsilon}}_{st}$$

where the tilde denotes the stacking of vectors and

$$\tilde{\mathbf{A}} = \begin{bmatrix} \mathbf{V} & \mathbf{U} & \mathbf{0} \\ \mathbf{V} & \mathbf{0} & \mathbf{U} \end{bmatrix}.$$

Assuming that the NFA-projected i-vectors follow a Gaussian distribution, the distribution of $\tilde{\mathbf{x}}_{st}$ can be obtained by marginalizing over all possible latent factors as follows:

$$\begin{aligned} p(\tilde{\mathbf{x}}_{st}|\text{same-speaker}) &= \int p(\tilde{\mathbf{x}}_{st}|\tilde{\mathbf{z}}_{st})p(\tilde{\mathbf{z}}_{st})\mathrm{d}\tilde{\mathbf{z}}_{st} \\ &= \int \mathcal{N}(\tilde{\mathbf{x}}_{st}|\tilde{\mathbf{m}} + \tilde{\mathbf{A}}\tilde{\mathbf{z}}_{st}, \tilde{\boldsymbol{\Sigma}})\mathcal{N}(\tilde{\mathbf{z}}_{st}|\mathbf{0}, \mathbf{I})\mathrm{d}\tilde{\mathbf{z}}_{st} \\ &= \mathcal{N}(\tilde{\mathbf{x}}_{st}|\tilde{\mathbf{m}}, \tilde{\mathbf{A}}\tilde{\mathbf{A}}^{\mathsf{T}} + \tilde{\boldsymbol{\Sigma}}) \\ &= \mathcal{N}\left(\begin{bmatrix}\mathbf{x}_s \\ \mathbf{x}_t\end{bmatrix} \middle| \begin{bmatrix}\mathbf{m} \\ \mathbf{m}\end{bmatrix}, \begin{bmatrix}\boldsymbol{\Sigma}_{tot} & \boldsymbol{\Sigma}_{ac} \\ \boldsymbol{\Sigma}_{ac} & \boldsymbol{\Sigma}_{tot}\end{bmatrix}\right) \end{aligned} \quad (37)$$

where $\tilde{\boldsymbol{\Sigma}} = \mathrm{diag}\{\boldsymbol{\Sigma}, \boldsymbol{\Sigma}\}$, $\boldsymbol{\Sigma}_{tot} = \mathbf{V}\mathbf{V}^{\mathsf{T}} + \mathbf{U}\mathbf{U}^{\mathsf{T}} + \boldsymbol{\Sigma}$ and $\boldsymbol{\Sigma}_{ac} = \mathbf{V}\mathbf{V}^{\mathsf{T}}$. If $\mathbf{x}_s$ and $\mathbf{x}_t$ are from the utterances of two different speakers, we have

$$\begin{bmatrix} \mathbf{x}_s \\ \mathbf{x}_t \end{bmatrix} = \begin{bmatrix} \mathbf{m} \\ \mathbf{m} \end{bmatrix} + \begin{bmatrix} \mathbf{V} & \mathbf{0} & \mathbf{U} & \mathbf{0} \\ \mathbf{0} & \mathbf{V} & \mathbf{0} & \mathbf{U} \end{bmatrix} \begin{bmatrix} \mathbf{h}_s \\ \mathbf{h}_t \\ \mathbf{w}_s \\ \mathbf{w}_t \end{bmatrix} + \begin{bmatrix} \boldsymbol{\epsilon}_s \\ \boldsymbol{\epsilon}_t \end{bmatrix} \quad (38)$$

which can be compactly written as

$$\tilde{\mathbf{x}}_{st} = \tilde{\mathbf{m}} + \bar{\mathbf{A}}\bar{\mathbf{z}}_{st} + \tilde{\boldsymbol{\epsilon}}_{st}.$$

The distribution of $\tilde{\mathbf{x}}_{st}$ is obtained by marginalizing over $\bar{\mathbf{z}}_{st}$:

$$\begin{aligned} p(\tilde{\mathbf{x}}_{st}|\text{diff-speaker}) &= \int p(\tilde{\mathbf{x}}_{st}|\bar{\mathbf{z}}_{st})p(\bar{\mathbf{z}}_{st})\mathrm{d}\bar{\mathbf{z}}_{st} \\ &= \int \mathcal{N}(\tilde{\mathbf{x}}_{st}|\tilde{\mathbf{m}} + \bar{\mathbf{A}}\bar{\mathbf{z}}_{st}, \tilde{\boldsymbol{\Sigma}})\mathcal{N}(\bar{\mathbf{z}}_{st}|\mathbf{0}, \mathbf{I})\mathrm{d}\bar{\mathbf{z}}_{st} \\ &= \mathcal{N}(\tilde{\mathbf{x}}_{st}|\tilde{\mathbf{m}}, \bar{\mathbf{A}}\bar{\mathbf{A}}^{\mathsf{T}} + \tilde{\boldsymbol{\Sigma}}) \\ &= \mathcal{N}\left(\begin{bmatrix}\mathbf{x}_s \\ \mathbf{x}_t\end{bmatrix} \middle| \begin{bmatrix}\mathbf{m} \\ \mathbf{m}\end{bmatrix}, \begin{bmatrix}\boldsymbol{\Sigma}_{tot} & \mathbf{0} \\ \mathbf{0} & \boldsymbol{\Sigma}_{tot}\end{bmatrix}\right) \end{aligned} \quad (39)$$

Combining Eq. 37 and Eq. 39, we have the log-likelihood ratio score:

$$\begin{aligned} S_{\text{LR}}(\mathbf{x}_s, \mathbf{x}_t) &= \ln \frac{\mathcal{N}\left(\begin{bmatrix}\mathbf{x}_s \\ \mathbf{x}_t\end{bmatrix} \middle| \begin{bmatrix}\mathbf{m} \\ \mathbf{m}\end{bmatrix}, \begin{bmatrix}\boldsymbol{\Sigma}_{tot} & \boldsymbol{\Sigma}_{ac} \\ \boldsymbol{\Sigma}_{ac} & \boldsymbol{\Sigma}_{tot}\end{bmatrix}\right)}{\mathcal{N}\left(\begin{bmatrix}\mathbf{x}_s \\ \mathbf{x}_t\end{bmatrix} \middle| \begin{bmatrix}\mathbf{m} \\ \mathbf{m}\end{bmatrix}, \begin{bmatrix}\boldsymbol{\Sigma}_{tot} & \mathbf{0} \\ \mathbf{0} & \boldsymbol{\Sigma}_{tot}\end{bmatrix}\right)} \\ &= \frac{1}{2}\begin{bmatrix}\mathbf{x}_s^{\mathrm{T}} & \mathbf{x}_t^{\mathrm{T}}\end{bmatrix}\begin{bmatrix}\mathbf{Q} & \mathbf{P} \\ \mathbf{P} & \mathbf{Q}\end{bmatrix}\begin{bmatrix}\mathbf{x}_s \\ \mathbf{x}_t\end{bmatrix} + \text{const} \\ &= \frac{1}{2}[\mathbf{x}_s^{\mathrm{T}}\mathbf{Q}\mathbf{x}_s + 2\mathbf{x}_s^{\mathrm{T}}\mathbf{P}\mathbf{x}_t + \mathbf{x}_t^{\mathrm{T}}\mathbf{Q}\mathbf{x}_t] + \text{const} \end{aligned} \quad (40)$$

where

$$\begin{aligned} \mathbf{P} &= \boldsymbol{\Sigma}_{tot}^{-1}\boldsymbol{\Sigma}_{ac}(\boldsymbol{\Sigma}_{tot} - \boldsymbol{\Sigma}_{ac}\boldsymbol{\Sigma}_{tot}^{-1}\boldsymbol{\Sigma}_{ac})^{-1}, \\ \mathbf{Q} &= \boldsymbol{\Sigma}_{tot}^{-1} - (\boldsymbol{\Sigma}_{tot} - \boldsymbol{\Sigma}_{ac}\boldsymbol{\Sigma}_{tot}^{-1}\boldsymbol{\Sigma}_{ac})^{-1}. \end{aligned}$$

REFERENCES

[1] F. Bimbot, J.-F. Bonastre, C. Fredouille, G. Gravier, I. Magrin-Chagnolleau, S. Meignier, T. Merlin, J. Ortega-García, D. Petrovska-Delacrétaz, and D. A. Reynolds, "A tutorial on text-independent speaker verification," *EURASIP journal on applied signal processing*, vol. 2004, pp. 430–451, 2004.

[2] T. Kinnunen and H. Li, "An overview of text-independent speaker recognition: from features to supervectors," *Speech communication*, vol. 52, no. 1, pp. 12–40, 2010.

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

[4] P. Kenny, "Joint factor analysis of speaker and session variability: Theory and algorithms," *CRIM, Montreal,(Report) CRIM-06/08-13*, 2005.

[5] P. Kenny, G. Boulianne, P. Ouellet, and P. Dumouchel, "Improvements in factor analysis based speaker verification," in *Acoustics, Speech and Signal Processing, 2006. ICASSP 2006 Proceedings. 2006 IEEE International Conference on*, vol. 1. IEEE, 2006, pp. I–I.

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

[7] N. Dehak, R. Dehak, P. Kenny, N. Brümmer, P. Ouellet, and P. Dumouchel, "Support vector machines versus fast scoring in the low-dimensional total variability space for speaker verification." in *Interspeech*, vol. 9, 2009, pp. 1559–1562.

[8] W. M. Campbell, D. E. Sturim, D. A. Reynolds, and A. Solomonoff, "SVM based speaker verification using a GMM supervector kernel and NAP variability compensation," in *Proc. ICASSP*, vol. 1, Toulouse, France, May 2006, pp. 97–100.

[9] N. Dehak, R. Dehak, P. Kenny, and P. Dumouchel, "Comparison between factor analysis and GMM support vector machines for speaker verification." in *Odyssey*, 2008.

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

[11] 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.

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

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

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

[15] P.-M. Bousquet, D. Matrouf, and J.-F. Bonastre, "Intersession compensation and scoring methods in the i-vectors space for speaker recognition." in *INTERSPEECH*, 2011, pp. 485–488.

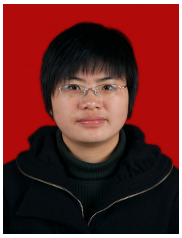
[16] P.-M. Bousquet, A. Larcher, D. Matrouf, J.-F. Bonastre, and O. Plchot, "Variance-spectra based normalization for i-vector standard and probabilistic linear discriminant analysis," in *Odyssey: The Speaker and Language Recognition Workshop*, 2012, pp. 157–164.

[17] Z. Li, D. Lin, and X. Tang, "Nonparametric discriminant analysis for face recognition," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 31, no. 4, pp. 755–761, 2009.

[18] D. A. Reynolds, "Channel robust speaker verification via feature mapping," in *Acoustics, Speech, and Signal Processing, 2003. Proceedings.(ICASSP'03). 2003 IEEE International Conference on*, vol. 2. IEEE, 2003, pp. II–53.

[19] R. Saeidi, K.-A. Lee, T. Kinnunen, T. Hasan, B. Fauve, P.-M. Bousquet, E. Khoury, P. Sordo Martinez, J. M. K. Kua, C. You *et al.*, "I4U submission to NIST SRE 2012: A large-scale collaborative effort for noise-robust speaker verification," in *Proc. Interspeech*, 2013.

[20] X. Zhao, Y. Wang, and D. Wang, "Robust speaker identification in noisy and reverberant conditions," *IEEE/ACM Transactions on Audio, Speech and Language Processing*, vol. 22, no. 4, pp. 836–845, 2014.
[21] Y. Lei, L. Burget, and N. Scheffer, "A noise robust i-vector extractor using vector taylor series for speaker recognition," in *Acoustics, Speech and Signal Processing (ICASSP), 2013 IEEE International Conference on*. IEEE, 2013, pp. 6788–6791.
[22] M. I. Mandasari, M. McLaren, and D. A. van Leeuwen, "The effect of noise on modern automatic speaker recognition systems," in *Acoustics, Speech and Signal Processing (ICASSP), 2012 IEEE International Conference on*. IEEE, 2012, pp. 4249–4252.
[23] R. Togneri and D. Pullella, "An overview of speaker identification: Accuracy and robustness issues," *Circuits and systems Magazine, IEEE*, vol. 11, no. 2, pp. 23–61, 2011.
[24] J. Ming, T. J. Hazen, J. R. Glass, and D. A. Reynolds, "Robust speaker recognition in noisy conditions," *Audio, Speech, and Language Processing, IEEE Transactions on*, vol. 15, no. 5, pp. 1711–1723, 2007.
[25] W. Zhu, S. O. Sadjadi, and J. W. Pelecanos, "Nearest neighbor based i-vector normalization for robust speaker recognition under unseen channel conditions," in *Proc. Int. Conf. on Acoustics, Speech and Signal Processing (ICASSP)*, 2015, pp. 4684–4688.
[26] T. Hasan and J. Hansen, "Acoustic factor analysis for robust speaker verification," *Audio, Speech, and Language Processing, IEEE Transactions on*, vol. 21, no. 4, pp. 842–853, 2013.
[27] ——, "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.
[28] W. B. Kheder, D. Matrouf, J.-F. Bonastre, M. Ajili, and P.-M. Bousquet, "Additive noise compensation in the I-vector space for speaker recognition," in *Proc. Int. Conf. on Acoustics, Speech and Signal Processing (ICASSP)*, 2015, pp. 4190–4194.
[29] 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 *Acoustics, Speech and Signal Processing (ICASSP), 2013 IEEE International Conference on*. IEEE, 2013, pp. 6778–6782.
[30] Y. Lei, L. Burget, L. Ferrer, M. Graciarena, and N. Scheffer, "Towards noise-robust speaker recognition using probabilistic linear discriminant analysis," in *Proc. ICASSP 2012*, Kyoto, Japan, March 2012, pp. 4253 – 4256.
[31] T. Hasan, S. O. Sadjadi, G. Liu, N. Shokouhi, H. Boril, and J. H. Hansen, "CRSS systems for 2012 NIST speaker recognition evaluation," in *Acoustics, Speech and Signal Processing (ICASSP), 2013 IEEE International Conference on*. IEEE, 2013, pp. 6783–6787.
[32] P. Rajan, T. Kinnunen, and V. Hautamäki, "Effect of multicondition training on i-vector PLDA configurations for speaker recognition," in *Proc. Interspeech*, 2013, pp. 3694–3697.
[33] G. Liu and J. Hansen, "An investigation into back-end advancements for speaker recognition in multi-session and noisy enrollment scenarios," *IEEE/ACM Trans. on Audio, Speech, and Language Processing*, vol. 22, no. 12, pp. 1978–1992, 2014.
[34] D. Garcia-Romero, X. Zhou, and C. Espy-Wilson, "Multicondition training of gaussian PLDA models in i-vector space for noise and reverberation robust speaker recognition," in *Acoustics, Speech and Signal Processing (ICASSP), 2012 IEEE International Conference on*, 2012, pp. 4257–4260.
[35] M. W. Mak, "SNR-dependent mixture of PLDA for noise robust speaker verification," in *Interspeech'2014*, 2014, pp. 1855–1859.
[36] D. Gong, Z. Li, D. Lin, J. Liu, and X. Tang, "Hidden factor analysis for age invariant face recognition," in *Computer Vision (ICCV), 2013 IEEE International Conference on*. IEEE, 2013, pp. 2872–2879.
[37] 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.
[38] K. Fukunaga, *Introduction to Statistical Pattern Recognition*. Boston, MA: Academic, 1990.
[39] S. Prince and J. Elder, "Probabilistic linear discriminant analysis for inferences about identity," in *Computer Vision, 2007. ICCV 2007. IEEE 11th International Conference on*, 2007, pp. 1–8.
[40] S. J. Prince, *Computer vision: models, learning, and inference*. Cambridge University Press, 2012.
[41] X. M. Pang and M. W. Mak, "Fusion of SNR-dependent PLDA models for noise robust speaker verification," in *ISCSLP'2014*, 2014, pp. 619–623.
[42] NIST, "The NIST year 2012 speaker recognition evaluation plan," *http://www.nist.gov/itl/iad/mig/sre12.cfm*, 2012.
[43] 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 2014.
[44] H. Yu and M. Mak, "Comparison of voice activity detectors for interview speech in nist speaker recognition evaluation," in *Interspeech*, 2011, pp. 2353–2356.
[45] B. S. Atal, "Effectiveness of linear prediction characteristics of the speech wave for automatic speaker identification and verification," *J. Acoust. Soc. Am.*, vol. 55, no. 6, pp. 1304–1312, Jun. 1974.
[46] J. Pelecanos and S. Sridharan, "Feature warping for robust speaker verification," in *Proc. Odyssey: The Speaker and Language Recognition Workshop*, Crete, Greece, Jun. 2001, pp. 213–218.
[47] 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."
[48] "http://dnt.kr.hsnr.de/download.html."
[49] S. F. D. C. Neto, "The itu-t software tool library," *International journal of speech technology*, vol. 2, no. 4, pp. 259–272, 1999.
[50] M. McLaren, M. Mandasari, and D. Leeuwen, "Source normalization for language-independent speaker recognition using i-vectors," in *Odyssey 2012: The Speaker and Language Recognition Workshop*, 2012, pp. 55–61.
[51] A. Martin, G. Doddington, T. Kamm, M. Ordowski, and M. Przybocki, "The DET curve in assessment of detection task performance," in *Proc. Eurospeech'97*, 1997, pp. 1895–1898.



Na Li received the B.S. degree in Environmental Engineering, M.S. and Ph.D. degrees in Acoustics from Northwestern Polytechnical University (NPU), Xi'an, China, in 2007, 2010, and 2015, respectively. From 2011 to 2013, she served as a Research Assistant in Shenzhen Institutes of Advanced Technology, Chinese Academy of Sciences, China. She is currently a Research Associate in the Department of Electronic and Information Engineering at The Hong Kong Polytechnic University. Her current research interests include speaker recognition, voice conversion, and machine learning.

Man-Wai Mak (M'93–SM'15) received a PhD in Electronic Engineering from the University of Northumbria in 1993. He joined the Department of Electronic and Information Engineering at The Hong Kong Polytechnic University in 1993 and is currently an Associate Professor in the same department. He has authored more than 150 technical articles in speaker recognition, machine learning, and bioinformatics. Dr. Mak also coauthored a postgraduate textbook *Biometric Authentication: A Machine Learning Approach, Prentice Hall, 2005* and a research monograph *Machine Learning for Protein Subcellular Localization Prediction, De Gruyter, 2015*. He served as a member of the IEEE Machine Learning for Signal Processing Technical Committee in 2005-2007. He has served as an associate editor of IEEE Trans. on Audio, Speech and Language Processing. He is currently an editorial board member of Journal of Signal Processing Systems and Advances in Artificial Neural Systems. He also served as Technical Committee Members of a number of international conferences, including ICASSP and Interspeech. Dr. Mak's research interests include speaker recognition, machine learning, and bioinformatics.