https://doi.org/10.1016/j.csl.2017.02.009 **This is the Preprint Version.**

Fast Scoring for PLDA with Uncertainty Propagation via I-vector Grouping

Wei-wei Lin[a], Man-Wai Mak[a], Jen-Tzung Chien[b]

[a] *Dept. of Electronic and Information Engineering, The Hong Kong Polytechnic University*
[b] *Dept. of Electrical and Computer Engineering, The National Chiao Tung University*

Abstract

The i-vector/PLDA framework has gained huge popularity in text-independent speaker verification. This approach, however, lacks the ability to represent the reliability of i-vectors. As a result, the framework performs poorly when presented with utterances of arbitrary duration. To address this problem, a method called uncertainty propagation (UP) was proposed to explicitly model the reliability of an i-vector by an utterance-dependent loading matrix. However, the utterance-dependent matrix greatly complicates the evaluation of likelihood scores. As a result, PLDA with UP, or PLDA-UP in short, is far more computational intensive than the conventional PLDA. In this paper, we propose to group i-vectors with similar reliability, and for each group the utterance-dependent loading matrices are replaced by a representative one. This arrangement allows us to pre-compute a set of representative matrices that cover all possible i-vectors, thereby greatly reducing the computational cost of PLDA-UP while preserving its ability in discriminating the reliability of i-vectors. Experiments on NIST 2012 SRE show that the proposed method can perform as good as the PLDA with UP while the scoring time is only 3.18% of it.

Keywords: Speaker verification, i-vector/PLDA, Uncertainty Propagation, duration mismatch.

1. Introduction

Recent years have witnessed the significant advances in text-independent speaker recognition. With the state-of-the-art techniques like i-vector, PLDA and DNN acoustic models, an EER of 0.59% on NIST speaker recognition evaluation has been reported [1]. Despite of these great advances, short-utterance speaker recognition remains a great challenge, as evident by a number of studies showing that system performance degrades rapidly when only short utterances are available [2, 3, 4]. However, in real applications users may not be willing to provide long utterances, especially during verification.

It has now become clear that naive applications of advanced text-independent methods, such as i-vector/PLDA, to short-utterance speaker verification could result in performance even poorer than that of the GMM and HMM modeling [5, 6]. One of the problems associated with short-utterance speaker verification is duration mismatch, where the length of enrolment utterances and test utterances are very different. Hasan *et al.* [7] assumed that duration mismatches can cause a shift in PLDA scores and proposed a duration-dependent quality measure function to compensate for the shift. Kanagasundaram *et al.* [4] compared joint factor analysis (JFA), i-vector PLDA, and i-vectors equipped with various subspace projections and variance normalization techniques under short-utterance scenarios. They found that no significant performance difference between JFA and i-vector PLDA when the enrollment and test utterances are very short and that JFA and PLDA offer marginally better performance than i-vectors with LDA followed by WCCN. Li *et al.* [8] noticed that for GMM-UBM systems, when both enrollment and test utterances are very short, the Gaussian components covered by the test utterances will not be properly trained during enrollment. To address this problem, they proposed to distribute speech signals into a number of phonetic sub-regions and model speakers within the sub-regions by region-specific GMMs.

A special concern for i-vector/PLDA is that it has no ability to represent the reliability of i-vectors. This problem is especially severe in short-utterance speaker verification. Recall that an i-vector is a maximum-a-posteriori (MAP) estimate of the latent variable in a factor analysis model. For short utterances, the number of acoustic frames is not enough to estimate i-vectors reliablely. By ignoring the time dimension, i-vectors estimated from long and short utterances are essentially treated as equally reliable. Kenny *et al.* [9] proposed to tightly couple i-vector extraction with PLDA modelling instead of treating them as two separated procedures. Specifically, the posterior covariance matrix of the latent factor is propagated into the PLDA model by introducing an extra loading matrix to represent the reliability of the i-vector. The method is called uncertainty propagation (UP) and the modified PLDA model is called PLDA-UP in this paper.

The extra loading matrix in PLDA-UP is utterance-dependent. As a result, the scoring of PLDA-UP is much more computationally intensive than conventional PLDA. Besides, PLDA-UP also requires to store the posterior covariance matrices of target-speakers' i-vectors, which is much more memory consuming than storing the i-vectors alone. Thus, both computational cost and memory consumption restrict the applications of PLDA-UP. To reduce the computational cost of PLDA-UP, Cumani *et al.* [10] proposed using MAP-estimated i-vectors to represent target speakers and propagating the posterior covariance matrix of test utterances into the PLDA model. This method relies on the assumption that enrolment utterances tend to be long. In [11], the author proposed to diagonalise the matrices involved in scoring to reduce the computational cost of full matrix operations. Although this approach significantly reduces the computational cost and does not require long enrolment utterances, it still degrades the performance of PLDA-UP when test utterances are very short.

The utterance-dependent matrix in PLDA-UP has no speaker specific information. The only role it plays is to convey the reliability of i-vector. Intuitively, if two utterances are close in duration, the corresponding i-vectors should have similar reliability. Based on this assumption, we have proposed in [12] to group i-vectors according to their utterance durations and model the reliability of i-vectors in each group by a single representative loading matrix.

Because these representative loading matrices can be pre-computed based on development data, we can pre-compute all of the relevant terms during scoring, thus saving lots of computation. In this paper, we extend our previous work in the following aspects:

- We introduce a metric for measuring the distance between two covariance matrices. Through this metric, we define a within-group distance to measure the quality of the grouping schemes.
- More extensive experiments are carried out to compare the performance of different grouping schemes. Also, the effectiveness of PLDA-UP and the proposed fast scoring schemes on utterances with different length-ranges was investigated.

Experimental results on the NIST 2012 SRE show that the proposed method can perform as good as the PLDA-UP in all four different length-ranges investigated, and the scoring time can be as low as 3.18% of the PLDA-UP.

The organiaztion of this paper is as follows. In Section 2 and Section 3, we give a brief review of i-vector/PLDA framework and PLDA-UP. We show why PLDA-UP can deal with length variability and the source of computational burden is also identified. We then present the proposed fast scoring schemes in Section 4. Experimental setup and results are presented in Section 5 and Section 6, respectively. Finally, we conclude our findings in Section 7.

2. Review of I-vector/PLDA

2.1. I-vector Extraction

The i-vector approach is an extension of joint factor analysis [13, 14]. It aims to extract from the acoustic vectors of an utterance a low-dimensional vector that incorporates most of the speaker information. It assumes that the speaker- and channel-dependent GMM-supervectors live in a low dimensional space:

$$\boldsymbol{\beta} = \mathbf{m} + \mathbf{T}\boldsymbol{\eta}, \tag{1}$$

where $\mathbf{m}$ is the speaker- and channel-independent GMM-supervector constructed by stacking up the means of a universal background model (UBM); $\mathbf{T}$ is a low-rank total variability matrix whose columns span the subspace where speaker- and channel-specific information varies; $\boldsymbol{\eta}$ is a latent variable which is assumed to follow a standard normal distribution. Given an utterance, its i-vector is a maximum-a-posteriori (MAP) estimate of the latent variable $\boldsymbol{\eta}$, which we denote as $\boldsymbol{\omega}$. To estimate an i-vector of an utterance with T acoustic frames, $\mathcal{O} = \{\mathbf{o}_1, \ldots, \mathbf{o}_T\}$, the Baum-Welch statistics are used:

$$N_c = \sum_{t=1}^{T} \gamma_c(\mathbf{o}_t) \tag{2}$$

$$\tilde{\mathbf{f}}_c = \sum_{t=1}^{T} \gamma_c(\mathbf{o}_t)(\mathbf{o}_t - \mathbf{m}_c), \quad c = 1, \ldots, C \tag{3}$$

where

$$\gamma_c(\mathbf{o}_t) = \frac{\lambda_c \mathcal{N}\left(\mathbf{o}_t | \mathbf{m}_c, \boldsymbol{\Sigma}_c\right)}{\sum_{c=1}^{C} \lambda_c \mathcal{N}\left(\mathbf{o}_t | \mathbf{m}_c, \boldsymbol{\Sigma}_c\right)}, \tag{4}$$

where $\mathbf{m}_c$ and $\boldsymbol{\Sigma}_c$ are the mean vector and covariance matrix of the c-th mixture in the UBM. The i-vector $\boldsymbol{\omega}$ and its posterior covariance matrix $\text{cov}(\boldsymbol{\eta}, \boldsymbol{\eta})$ can be obtained by [13, 15]:

$$\boldsymbol{\omega} = \text{cov}(\boldsymbol{\eta}, \boldsymbol{\eta}) \sum_{c=1}^{C} \mathbf{T}_c^{\mathsf{T}} \boldsymbol{\Sigma}_c^{-1} \tilde{\mathbf{f}}_c \tag{5}$$

$$\text{cov}(\boldsymbol{\eta}, \boldsymbol{\eta}) = \mathbf{L}^{-1} = \left(\mathbf{I} + \sum_{c=1}^{C} N_c \mathbf{T}_c^{\mathsf{T}} \boldsymbol{\Sigma}_c^{-1} \mathbf{T}_c \right)^{-1}, \tag{6}$$

where $\mathbf{L}$ is a precision matrix and $\mathbf{T}_c$ is the c-th partition of $\mathbf{T}$, i.e. $\mathbf{T} = [\mathbf{T}_1^{\mathsf{T}}, \ldots, \mathbf{T}_C^{\mathsf{T}}]^{\mathsf{T}}$.

2.2. Probabilistic Linear Discriminant Analysis

To suppress undesired intra-speaker variability in i-vectors, channel compensation is applied. Probabilistic linear discriminant analysis (PLDA) is found to be the most effective. Because of the heavy-tailed behaviour of i-vector distributions, early PLDA is based on Students's t distribution [16]. Garcia-Romero and

Espy-Wilson [17] found later that by simply length-normalizing the i-vectors, Gaussian PLDA can perform equally well. Because of the nice analytical solution that Gaussian PLDA can offer, it is more preferable in practice.

2.2.1. Pre-processing for Gaussian PLDA

To use Gaussian PLDA, two pre-processing steps are necessary to Gaussianalize i-vectors. First, a whitening transform is applied to i-vectors:

$$\boldsymbol{\omega}^{\text{wht}} = \mathbf{W}^{\mathsf{T}}(\boldsymbol{\omega} - \bar{\boldsymbol{\omega}}), \tag{7}$$

where $\bar{\boldsymbol{\omega}}$ is the global mean of i-vectors, $\mathbf{W}$ is a transformation matrix obtained from the Cholesky decomposition of the within-class covariance matrix of i-vectors [18] and $\boldsymbol{\omega}^{\text{wht}}$ is the whitened i-vector. The second step is to apply a simple length-normalization to the whitened i-vectors:

$$\boldsymbol{\omega}^{\text{l-norm}} = \frac{\boldsymbol{\omega}^{\text{wht}}}{\|\boldsymbol{\omega}^{\text{wht}}\|}. \tag{8}$$

It is customary to include linear discriminant analysis (LDA) and within-class covariance normalization (WCCN) [18] in the pre-processing steps. The whole pre-processing can be written in a more succinct fashion:

$$\mathbf{w} = \frac{\mathbf{P}(\boldsymbol{\omega} - \bar{\boldsymbol{\omega}})}{\|\boldsymbol{\omega}^{\text{wht}}\|}, \tag{9}$$

where $\mathbf{P}$ denotes the transformation matrix that combines whitening, LDA and WCCN and $\mathbf{w}$ is the pre-processed i-vector that is ready for PLDA modelling.

2.2.2. Gaussian PLDA as a Generative Model

Given R i-vectors $\{\mathbf{w}_r; r = 1, \ldots, R\}$ from a speaker, PLDA assumes that they can be decomposed in the following manner:

$$\mathbf{w}_r = \boldsymbol{\mu} + \mathbf{V}\mathbf{h} + \mathbf{G}\mathbf{z}_r + \boldsymbol{\epsilon}_r. \tag{10}$$

This decomposition has two distinct parts: (1) the speaker-dependent part, $\boldsymbol{\mu} + \mathbf{V}\mathbf{h}$, which is the same for all i-vectors from the same speaker; (2) the utterance-dependent part, $\mathbf{G}\mathbf{z}_r + \boldsymbol{\epsilon}_r$, which varies even for the utterances from the same speaker. In Eq. 10, $\boldsymbol{\mu}$ is the global mean of i-vectors and the matrix $\mathbf{V}$ represents the speaker subspace on which the speaker factor $\mathbf{h}$ can vary. The columns of matrix $\mathbf{U}$ span the subspace where the channel factor $\mathbf{z}_r$ varies. $\boldsymbol{\epsilon}_r$ models the residue that is not captured by both speaker and channel subspaces and is assumed to follow a Gaussian distribution with zero mean and a diagonal covariance matrix.

The low dimensionality of i-vector makes it possible to conflate the channel variability and residue by using a full covariance matrix $\boldsymbol{\Sigma}$ such that:

$$\mathbf{w}_r = \boldsymbol{\mu} + \mathbf{V}\mathbf{h} + \boldsymbol{\epsilon}_r, \qquad \boldsymbol{\epsilon}_r \sim \mathcal{N}\left(\mathbf{0}, \boldsymbol{\Sigma}\right). \tag{11}$$

2.2.3. Scoring in Gaussian PLDA

Given a target speaker's i-vector $\mathbf{w}_s$ and a test i-vector $\mathbf{w}_t$, the log-likelihood ratio of the same-speaker hypothesis to different-speaker hypothesis can be computed by [17]:

$$\begin{aligned} S_{LR}(\mathbf{w}_s, \mathbf{w}_t) &= \log \frac{p(\mathbf{w}_s, \mathbf{w}_t|\text{same-speaker})}{p(\mathbf{w}_s, \mathbf{w}_t|\text{different-speaker})} \\ &= \frac{1}{2}\mathbf{w}_s^{\mathsf{T}}\boldsymbol{\Phi}\mathbf{w}_s + \mathbf{w}_s^{\mathsf{T}}\boldsymbol{\Psi}\mathbf{w}_t + \frac{1}{2}\mathbf{w}_t^{\mathsf{T}}\boldsymbol{\Phi}\mathbf{w}_t + \text{const} \end{aligned} \tag{12}$$

where

$$\boldsymbol{\Phi} = \boldsymbol{\Sigma}_{tot}^{-1} - (\boldsymbol{\Sigma}_{tot} - \boldsymbol{\Sigma}_{ac}\boldsymbol{\Sigma}_{tot}^{-1}\boldsymbol{\Sigma}_{ac})^{-1} \tag{13}$$

$$\boldsymbol{\Psi} = \boldsymbol{\Sigma}_{tot}^{-1}\boldsymbol{\Sigma}_{ac}(\boldsymbol{\Sigma}_{tot} - \boldsymbol{\Sigma}_{ac}\boldsymbol{\Sigma}_{tot}^{-1}\boldsymbol{\Sigma}_{ac})^{-1} \tag{14}$$

$$\boldsymbol{\Sigma}_{ac} = \mathbf{V}\mathbf{V}^{\mathsf{T}} \qquad \boldsymbol{\Sigma}_{tot} = \mathbf{V}\mathbf{V}^{\mathsf{T}} + \boldsymbol{\Sigma}. \tag{15}$$

Note that Eqs. 13–14 can be computed beforehand. Only Eq. 12 needs to be evaluated during verification. As a result, PLDA scoring is very efficient.

3. Gaussian PLDA with Uncertainty Propagation

Despite the great success of the i-vector/PLDA framework, its performance becomes very poor if both the enrolment and test utterances have a wide range of durations. There are several reasons for this. First, in i-vector extraction, the duration of utterances is totally ignored, i.e., utterances are represented by vectors of fixed dimension regardless of their duration. Recall that an i-vector is the MAP estimate of latent variable $\boldsymbol{\eta}$; the accuracy of such estimate depends on the number of acoustic vectors. By ignoring durations, all i-vectors are treated as equally reliable. Second, in PLDA modelling, it is assumed that all of the intra-speaker variabilities are represented by the covariance matrix $\boldsymbol{\Sigma}$, which is the same across all i-vectors. This is apparently not a satisfactory assumption because short utterances have more severe intra-speaker variabilities than long utterances.

To better accommodate utterance-length variability, a modified PLDA is proposed in [9]. The basic idea is to tightly couple i-vector extraction and PLDA modelling by propagating the uncertainty during i-vector extraction into the PLDA model. Recall that the posterior covariance matrix in Eq. 6 represents the uncertainty of the MAP point-estimate in i-vector extraction. The shorter the utterance, the larger the posterior covariances. By propagating this information into PLDA and using a loading matrix to model the variability due to duration variation, this PLDA model can better handle the length-variability than the conventional PLDA model.

3.1. Preprocessing for Gaussian PLDA with UP

The pre-processing steps in Section. 2.2.1 also need to be applied to the posterior covariance matrices. If only linear transform $\mathbf{P}$ is applied to an i-vector, the corresponding pre-processed covariance matrix can be obtained by:

$$\mathrm{cov}(\mathbf{P}\boldsymbol{\eta}, \mathbf{P}\boldsymbol{\eta}) = \mathbf{P}\mathbf{L}^{-1}\mathbf{P}^{\mathsf{T}}, \tag{16}$$

which we denote as $\boldsymbol{\Lambda}$. When length-normalization is applied to an i-vector, the pre-processed covariance matrix can be approximated by [9]:

$$\boldsymbol{\Lambda} \leftarrow \frac{\mathbf{P}\mathbf{L}^{-1}\mathbf{P}^{\mathsf{T}}}{\|\boldsymbol{\omega}^{\mathrm{wht}}\|}. \tag{17}$$

Other methods to deal with this non-linear transform on posterior matrix can be found in [9, 19].

3.2. Generative Model for Gaussian PLDA with UP

To propagate the uncertainty of an i-vector into the PLDA model, an utterance-dependent loading matrix is added to the factor analysis model:

$$\mathbf{w}_r = \boldsymbol{\mu} + \mathbf{V}\mathbf{h} + \mathbf{U}_r\mathbf{z}_r + \boldsymbol{\epsilon}_r, \tag{18}$$

where $\mathbf{U}_r$ is the Cholesky decomposition of the posterior covariance matrix $\boldsymbol{\Lambda}_r$, and $\mathbf{z}_r$ is a latent variable assumed to follow a standard normal distribution. The intra-speaker variability of $\mathbf{w}_r$ in Eq. 18 is:

$$\mathrm{cov}(\mathbf{w}_r, \mathbf{w}_r|\mathbf{h}) = \boldsymbol{\Lambda}_r + \boldsymbol{\Sigma}, \tag{19}$$

where $\boldsymbol{\Lambda}_r$ varies from utterances to utterances, thus reflecting the reliability of i-vector $\mathbf{w}_r$.

3.3. Scoring in Gaussian PLDA with UP

Given a target speaker's i-vector $\mathbf{w}_s$ together with its posterior covariance matrix $\boldsymbol{\Lambda}_s$ and a test i-vector $\mathbf{w}_t$ together with its posterior covariance matrix $\boldsymbol{\Lambda}_t$, the log-likelihood ratio can be written as:

$$\begin{aligned}
S_{LR}(\mathbf{w}_s, \mathbf{w}_t; \boldsymbol{\Lambda}_s, \boldsymbol{\Lambda}_t) &= \log \frac{p(\mathbf{w}_s, \mathbf{w}_t; \boldsymbol{\Lambda}_s, \boldsymbol{\Lambda}_t|\text{same-speaker})}{p(\mathbf{w}_s, \mathbf{w}_t; \boldsymbol{\Lambda}_s, \boldsymbol{\Lambda}_t|\text{different-speaker})} \\
&= \log p\left(\begin{bmatrix}\mathbf{w}_s \\ \mathbf{w}_t\end{bmatrix} \middle| \begin{bmatrix}\mathbf{0} \\ \mathbf{0}\end{bmatrix}, \begin{bmatrix}\boldsymbol{\Sigma}_s & \boldsymbol{\Sigma}_{ac} \\ \boldsymbol{\Sigma}_{ac} & \boldsymbol{\Sigma}_t\end{bmatrix}\right) \\
&\quad - \log p\left(\begin{bmatrix}\mathbf{w}_s \\ \mathbf{w}_t\end{bmatrix} \middle| \begin{bmatrix}\mathbf{0} \\ \mathbf{0}\end{bmatrix}, \begin{bmatrix}\boldsymbol{\Sigma}_s & \mathbf{0} \\ \mathbf{0} & \boldsymbol{\Sigma}_t\end{bmatrix}\right) \\
&= \frac{1}{2}\mathbf{w}_s^{\mathsf{T}}\mathbf{A}_{s,t}\mathbf{w}_s + \mathbf{w}_s^{\mathsf{T}}\mathbf{B}_{s,t}\mathbf{w}_t + \frac{1}{2}\mathbf{w}_t^{\mathsf{T}}\mathbf{C}_{s,t}\mathbf{w}_t + D_{s,t}
\end{aligned} \tag{20}$$

where

$$\mathbf{A}_{s,t} = \mathbf{\Sigma}_s^{-1} - (\mathbf{\Sigma}_s - \mathbf{\Sigma}_t^{-1}\mathbf{\Sigma}_{ac})^{-1} \tag{21}$$

$$\mathbf{B}_{s,t} = \mathbf{\Sigma}_s^{-1}\mathbf{\Sigma}_{ac}(\mathbf{\Sigma}_t - \mathbf{\Sigma}_{ac}\mathbf{\Sigma}_s^{-1}\mathbf{\Sigma}_{ac})^{-1} \tag{22}$$

$$\mathbf{C}_{s,t} = \mathbf{\Sigma}_t^{-1} - (\mathbf{\Sigma}_t - \mathbf{\Sigma}_s^{-1}\mathbf{\Sigma}_{ac})^{-1} \tag{23}$$

$$D_{s,t} = -\frac{1}{2}\log\begin{vmatrix}\mathbf{\Sigma}_s & \mathbf{\Sigma}_{ac}\\ \mathbf{\Sigma}_{ac} & \mathbf{\Sigma}_t\end{vmatrix} + \frac{1}{2}\log\begin{vmatrix}\mathbf{\Sigma}_s & \mathbf{0}\\ \mathbf{0} & \mathbf{\Sigma}_t\end{vmatrix} \tag{24}$$

$$\mathbf{\Sigma}_t = \mathbf{V}\mathbf{V}^{\mathsf{T}} + \mathbf{\Lambda}_t + \mathbf{\Sigma} \tag{25}$$

$$\mathbf{\Sigma}_s = \mathbf{V}\mathbf{V}^{\mathsf{T}} + \mathbf{\Lambda}_s + \mathbf{\Sigma} \tag{26}$$

$$\mathbf{\Sigma}_{ac} = \mathbf{V}\mathbf{V}^{\mathsf{T}}. \tag{27}$$

It is worth to notice that Eqs. 21–24 involve terms dependent on both the target speaker's utterance and the test utterance, which means that these terms need to be evaluated during scoring.

4. Fast Scoring via I-vector Grouping

The computational burden of PLDA-UP comes from the utterance-dependent loading matrix $\mathbf{U}_r$ in Eq. 18, where the uncertainty is represented by $\mathbf{U}_r\mathbf{U}_r^{\mathsf{T}}$. If we have a group of i-vectors with similar reliability, one prescribed loading matrix should be sufficient to model the reliability of all of the i-vectors in the group. Furthermore, if the prescribed loading matrix can be estimated from development data, the utterance-dependent terms in Eqs. 21–24 can be precomputed, which would greatly speed up the scoring process.

Suppose we have a collection of i-vectors from a speaker and they are distributed into K groups indexed by k, with the members within the k-th group indexed by (k, i). Then, the factor analysis model can be written as:

$$\mathbf{w}_{k,i} = \boldsymbol{\mu} + \mathbf{V}\mathbf{h} + \mathbf{U}_k\mathbf{z}_{k,i} + \boldsymbol{\epsilon}_{k,i}, \tag{28}$$

where the loading matrices $\{\mathbf{U}_k\}_{k=1}^K$ are obtained from development data. Different grouping schemes [12] will be explored in this paper:

- Grouping i-vectors by utterance durations.
- Grouping i-vectors by the characteristics of the posterior covariance matrices.

4.1. Three Approaches to Grouping I-vectors

In this section, we describe and assess the quality of proposed grouping schemes. The first scheme is based on utterance durations and the last two are based on the characteristics of posterior covariance matrices.

One intuitive way to group i-vectors with similar reliability is to group them according to the durations of their utterances. This can be easily done by dividing the time axis (starting from the shortest duration) into a number of equal-length intervals. Then, for each interval, the uncertainties of i-vectors are represented by the posterior covariance matrix of the i-vector whose utterance duration falls on or nearest to the middle of that interval. For example, if the interval is between 10 to 20 seconds, we select the covariance matrix whose corresponding utterance duration is closest to 15 seconds.

Suppose the time axis is divided into K equal-length time intervals indexed by k. Then, the i-th i-vector in the k-th interval is denoted as $\mathbf{w}_{k,i}$ and its pre-processed posterior covariance matrix is denoted as $\mathbf{\Lambda}_{k,i}$,[1] where $i = 1, 2, \ldots, I_k$. Among the I_k posterior covariance matrices in the k-th interval, the one with utterance-length closest to the middle of the k-th interval is selected to represent the uncertainty of all the i-vectors insider the interval. We denote the selected matrix as $\mathbf{\Lambda}_{k,r}$. As $\mathbf{\Lambda}_{k,r}$ represents the uncertainty of all of the i-vectors insider the k-th interval, we need to assume:

$$\mathbf{\Lambda}_{k,i} \approx \mathbf{\Lambda}_{k,r} \quad \forall i \neq r. \tag{29}$$

To see if the above assumption holds, we introduce a within-group distance $d(\mathbf{\Lambda}_{k,i}, \mathbf{\Lambda}_{k,r})$ to measure the distances between the selected matrix and other

[1] For simplicity, in the sequel we will refer the pre-processed posterior covariance matrix in Eq. 17 as the posterior covariance matrix $\mathbf{\Lambda}$ when the context is clear.

matrices in the k-th group [20]:

$$d(\mathbf{\Lambda}_{k,i}, \mathbf{\Lambda}_{k,r}) = \sqrt{\frac{\text{trace}\{(\mathbf{\Lambda}_{k,i} - \mathbf{\Lambda}_{k,r})^{\mathsf{T}}(\mathbf{\Lambda}_{k,i} - \mathbf{\Lambda}_{k,r})\}}{\text{trace}\{(\mathbf{\Lambda}_{k,i}^{\mathsf{T}}\mathbf{\Lambda}_{k,i}) + (\mathbf{\Lambda}_{k,r}^{\mathsf{T}}\mathbf{\Lambda}_{k,r})\}}} \quad i \neq r. \tag{30}$$

Note that the distance has a range between 0.0 and 1.0 such that the smaller the distance the more similar are the two matrices. We truncated 7,156 telephone conversations from NIST 2008–2010 SRE (see Section 6) into short segments so that their durations are uniformly distributed between 3 and 60 seconds. After i-vector extraction and pre-processing, we applied the above mentioned procedure to group i-vectors, i.e., the time axis was divided into five 11.4-second intervals starting from 3 seconds and ending at 60 seconds. $\mathbf{\Lambda}_{k,i}, i = 1, 2, \ldots, I_k$, represent the posterior covariance matrices inside the k-th interval, among which $\mathbf{\Lambda}_{k,r}$ was selected as the representative of the interval. The within-group distances are computed for $I_k - 1$ pairs of $\mathbf{\Lambda}_{k,r}$ and $\mathbf{\Lambda}_{k,i}$, where $i \neq r$, for a total of 5 groups. The results are presented in Fig. 1(a). Each box together with its whiskers represent the variability of the within-group distances of that group. The central mark inside each box indicates the median within-group distance, and the bottom and top edges of each box indicate the 25th and 75th percentiles, respectively. The whiskers extend to the most extreme non-outliers, and the outliers are represented by the '+' symbol [21].

We can see from Fig. 1(a) that the majority of the distances are quite small (75% of the distances are smaller than the value indicated by the upper edge of each box). As small distance means high similarity between representative matrix and the other matrices in the group, we conclude that selecting representative matrices based on utterance durations is a reasonable approach. Nevertheless, there are still some outliers in the five groups. The reason for the outliers is that utterance duration does not totally capture the information in the posterior covariance matrix. Even for utterances of exactly the same duration, their zero-th oder statistics (N_c in Eq. 2) can be quite different, which could result in different posterior covariance matrices. Even if the posterior covariance matrices of two i-vectors are exactly the same, i.e., $\mathbf{L}_1^{-1} = \mathbf{L}_2^{-1}$ in Eq. 6, their post-processed covariance matrices ($\mathbf{\Lambda}_1$ and $\mathbf{\Lambda}_2$ in Eq. 17) could

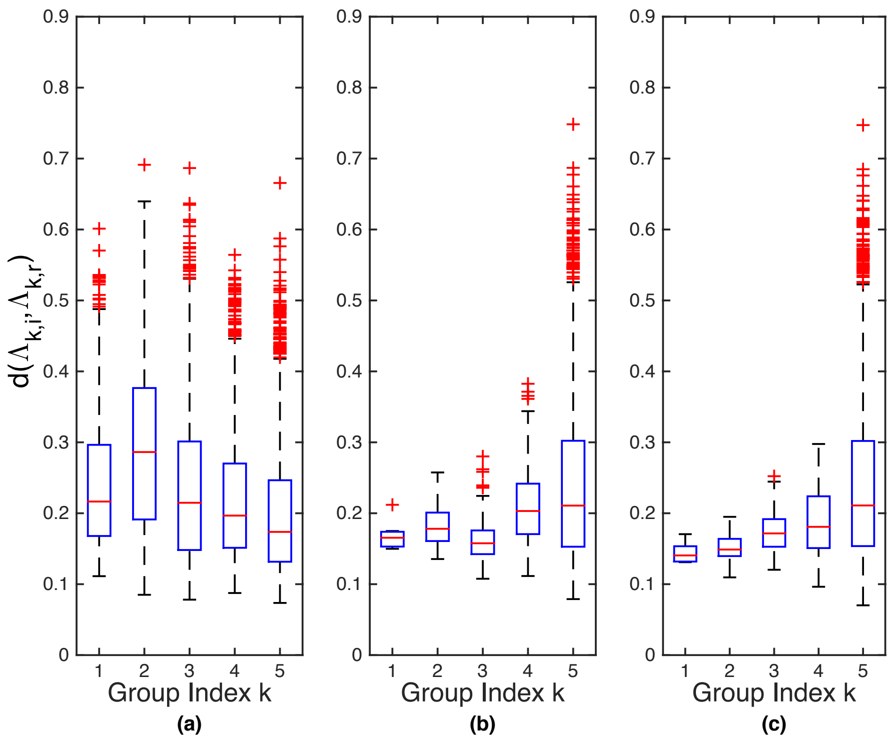

Figure 1: Distances between the representative matrix $\mathbf{\Lambda}_{k,r}$ of the k-th group and all of the other matrices in the group. I-vector grouping schemes based on (a) utterance duration, (b) the largest eigenvalue of $\mathbf{U}\mathbf{U}^{\mathsf{T}}$ and (c) the trace of $\mathbf{U}\mathbf{U}^{\mathsf{T}}$.

be different. This is because the whitened i-vectors ($\boldsymbol{\omega}_1^{\text{wht}}$ and $\boldsymbol{\omega}_2^{\text{wht}}$) are not identical in general.

To solve these problems, we propose two alternative approaches to grouping i-vectors using the characteristics of the posterior covariance matrices [12]. To this end, we define a scalar α, which is a function of the posterior covariance matrix:

$$\alpha = f(\mathbf{\Lambda}). \tag{31}$$

In Eq. 31, α could be:

1. the largest eigenvalue of $\mathbf{\Lambda}$, because the largest eigenvalue could dominate the variances of all components; and

2. the trace of $\mathbf{\Lambda}$, because the trace of a covariance matrix is the sum of its eigenvalues, which summarizs the variability of all components in the corresponding i-vector.

Specifically, we computed α for every posterior covariance matrix after preprocessing. Then we divided the α–axis into K equal-spaced intervals indexed by k. The i-vectors associated with the k-th interval are denoted as $\mathbf{w}_{k,i}$ and their posterior covariance matrices are denoted as $\mathbf{\Lambda}_{k,i}$, where $i = 1, 2, \ldots, I_k$. The posterior covariance matrix whose value of α is closest to the middle of the k-th interval is selected to represent the uncertainty of i-vectors in this interval and denoted as $\mathbf{\Lambda}_{k,r}$. Following this procedure, we divided the i-vectors extracted from the above mentioned 3–60 seconds utterances into 5 groups using the largest eigenvalues and matrix traces, respectively. To evaluate the quality of these two grouping schemes, we compute the within-group distances $d(\mathbf{\Lambda}_{k,i}, \mathbf{\Lambda}_{k,r})$ for $I_k - 1$ pairs of $\mathbf{\Lambda}_{k,i}$ and $\mathbf{\Lambda}_{k,r}$, where $i \neq r$, for a total of 5 groups. The results are shown in Fig. 1(b) and Fig. 1(c) for using the largest eigenvalues and matrix traces, respectively. When compared with Fig. 1(a), there are considerably less outliers in Groups 1–4 in both Fig. 1(b) and Fig. 1(c), although Group 5 still has a large number of outliers.

4.2. Fast Scoring Procedure

Given a target speaker's i-vector $\mathbf{w}_s$ and a test i-vector $\mathbf{w}_t$, we need to determine their group index first, which we denoted as m and n, respectively. For the grouping scheme based on utterance duration, this can be achieved by comparing their utterance duration, denoted as $l^{(s)}$ and $l^{(t)}$, with the durations of the representative matrices, $\{l_k; k = 1, \ldots, K\}$:

$$m = \underset{k \in \{1,\ldots,K\}}{\arg\min} \; |l_k - l^{(s)}| \tag{32}$$

$$n = \underset{k \in \{1,\ldots,K\}}{\arg\min} \; |l_k - l^{(t)}|. \tag{33}$$

For the grouping schemes based on the characteristics of the posterior covariance matrices, we need to evaluate the α-value of target speaker's posterior covariance matrix $\mathbf{\Lambda}_s$, which we denoted as $\alpha^{(s)}$, and the α-value of test utterance's posterior covariance matrix $\mathbf{\Lambda}_t$, which we denoted as $\alpha^{(t)}$. Then we compared $\alpha^{(s)}$ and $\alpha^{(t)}$ with the α-value of the representative matrices, $\{\alpha_k; k = 1, \ldots, K\}$, to determine the group identities of target speaker and test utterances:

$$m = \underset{k \in \{1,\ldots,K\}}{\arg\min} \; |\alpha_k - \alpha^{(s)}| \tag{34}$$

$$n = \underset{k \in \{1,\ldots,K\}}{\arg\min} \; |\alpha_k - \alpha^{(s)}|. \tag{35}$$

Then the log-likelihood ratio can be written as:

$$\mathrm{S}_{LR}(\mathbf{w}_s, \mathbf{w}_t; m, n) = \frac{1}{2}\mathbf{w}_s \mathbf{A}_{m,n} \mathbf{w}_s + \mathbf{w}_s^{\mathsf{T}} \mathbf{B}_{m,n} \mathbf{w}_t + \frac{1}{2}\mathbf{w}_t^{\mathsf{T}} \mathbf{C}_{m,n} \mathbf{w}_t + D_{m,n}, \tag{36}$$

where

$$\mathbf{A}_{m,n} = \mathbf{\Sigma}_m^{-1} - (\mathbf{\Sigma}_m - \mathbf{\Sigma}_n^{-1}\mathbf{\Sigma}_{ac})^{-1} \tag{37}$$

$$\mathbf{B}_{m,n} = \mathbf{\Sigma}_m^{-1}\mathbf{\Sigma}_{ac}(\mathbf{\Sigma}_n - \mathbf{\Sigma}_{ac}\mathbf{\Sigma}_m^{-1}\mathbf{\Sigma}_{ac})^{-1} \tag{38}$$

$$\mathbf{C}_{m,n} = \mathbf{\Sigma}_n^{-1} - (\mathbf{\Sigma}_n - \mathbf{\Sigma}_m^{-1}\mathbf{\Sigma}_{ac})^{-1} \tag{39}$$

$$D_{m,n} = -\frac{1}{2}\log \begin{vmatrix} \mathbf{\Sigma}_m & \mathbf{\Sigma}_{ac} \\ \mathbf{\Sigma}_{ac} & \mathbf{\Sigma}_n \end{vmatrix} + \frac{1}{2}\log \begin{vmatrix} \mathbf{\Sigma}_m & \mathbf{0} \\ \mathbf{0} & \mathbf{\Sigma}_n \end{vmatrix} \tag{40}$$

$$\mathbf{\Sigma}_n = \mathbf{V}\mathbf{V}^{\mathsf{T}} + \mathbf{\Lambda}_n + \mathbf{\Sigma} \tag{41}$$

$$\mathbf{\Sigma}_m = \mathbf{V}\mathbf{V}^{\mathsf{T}} + \mathbf{\Lambda}_m + \mathbf{\Sigma} \tag{42}$$

$$\mathbf{\Sigma}_{ac} = \mathbf{V}\mathbf{V}^{\mathsf{T}}. \tag{43}$$

Because Eqs. 37–40 do not depend on the test utterance, they can be pre-computed. For the grouping scheme based on utterance duration, the only extra computation is Eq. 33 during verification. For the grouping schemes based on covariance matrix's characteristics, we need to evaluate Eq. 31 and Eq. 35.

5. Experimental Setup

5.1. Acoustic Front-End Processing

Speech data from NIST 2005–2010 Speaker Recognition Evaluation (SRE) were used for system development. For performance evaluation, NIST 2012 SRE

[22] were used. For each utterance, a two-channel voice activity detector (VAD) [23] was applied to remove silent regions. Then a 25-ms Hamming window was used to extract 19 mel frequency cepstral coefficients (MFCC) and log-energy plus their first and second derivatives. Cepstral mean normalization and feature warping [24] were applied to compensate for channel variability in the MFCC vectors. In order to simulate utterances with arbitrary duration, four set of utterances with duration ranging from 3–20 seconds, 3–30 seconds, 3–40 seconds and 3–60 seconds, respectively, were created by truncating speech files from NIST 2012 SRE (core set, male speaker).

5.2. Speaker Model Training

Full-length microphone and telephone utterances from NIST 2005–2008 SREs were used to train a gender-dependent UBM with 1024 Gaussian components and an i-vector extractor with 500 total factors. Then, i-vectors were extracted from the above mentioned truncated speech files. WCCN together with length-normalization were applied to reduce the heavy-tailed behavior of i-vectors. LDA was applied to project the i-vectors to a 200 dimensional subspace with better speaker discrimination. Another WCCN was then applied to reduce the undesired high within-class variability in the LDA-projected space. Then a PLDA models were trained using the pre-processed i-vectors (Eq. 9). PLDA-UP model was trained using the pre-processed i-vectors together with their posterior covariance matrices. For fast scoring systems, we obtained the representative matrices from the truncated telephone utterances in NIST 2006–2010 SRE, following the procedures described in Section 4. According to different schemes specified in Table. 5.2, we have three fast scoring systems.

6. Results and Analysis

System performance was based on the truncated speech segments of Common Conditions 2 and 4 of NIST 2012 SRE (core set, male speakers). Equal error rate (EER), minimum detection cost function (minDCF) in NIST 2012 SRE were used as performance metrics.

System	Criteria for Grouping i-vectors
Sys. 1	Utterance length (after VAD)
Sys. 2	The largest eigenvalue of posterior covariance matrix
Sys. 3	The trace of posterior covariance matrix

Table 1: The criteria for grouping i-vectors used by the 3 systems.

Fig 2 shows a bar chart of the EERs and total scoring time of PLDA, PLDA-UP and the three fast scoring systems with different numbers of i-vector groups. Obviously, the bar chart suggests that our fast scoring systems significantly reduce the scoring time while maintaing the good performance of PLDA-UP. The following sub-sections gives a detailed analysis of the results.

6.1. Performance of Fast Scoring Systems

Table 2 shows the EER and minDCF obtained by PLDA, PLDA-UP and the three fast scoring systems in common conditions 2 and 4, respectively. The results have two implications:

- PLDA-UP outperforms the conventional PLDA in all the four duration ranges. The extent of improvement depends on the range of utterance length. We can see that the performance margin is the greatest when utterance-length ranges from 3–20 seconds.

- Dividing i-vectors into five groups ($K = 5$) seems to be sufficient for all of the four duration ranges. Only System 1 in CC2 and System 2 in CC4 show noticeable improvement in both EER and minDCF when the number of groups increases from 5 to 10.

- There is no clear winner among the three fast scoring systems. All three perform equally well as compared to PLDA-UP. In some settings, the fast scoring systems even perform better than PLDA-UP, although by a very small margin only.

			Duration Range (seconds)							
			3–20		3–30		3–40		3–60	
Method		K	EER(%)	minDCF	EER(%)	minDCF	EER(%)	minDCF	EER(%)	minDCF
CC2	PLDA	-	7.41	0.802	6.42	0.665	5.35	0.576	4.20	0.520
	PLDA-UP	-	6.25	0.714	5.43	0.637	4.73	0.563	3.81	0.493
	Sys. 1	5	6.35	0.711	5.54	0.625	4.92	0.554	3.94	0.478
		10	6.17	0.703	5.33	0.625	4.57	0.553	3.80	0.479
		15	6.11	0.710	5.33	0.628	4.69	0.562	3.81	0.479
	Sys. 2	5	6.10	0.723	5.50	0.633	4.66	0.580	3.91	0.485
		10	6.28	0.712	5.49	0.630	4.73	0.566	3.76	0.49
		15	6.30	0.715	5.42	0.620	4.62	0.572	3.77	0.495
	Sys. 3	5	6.14	0.716	5.33	0.621	4.62	0.569	3.87	0.486
		10	6.27	0.713	5.39	0.630	4.73	0.565	3.81	0.485
		15	6.25	0.715	5.36	0.628	4.75	0.567	3.84	0.487
CC4	PLDA	-	14.66	0.899	12.06	0.792	10.88	0.710	9.22	0.656
	PLDA-UP	-	13.28	0.878	11.34	0.809	10.23	0.731	8.71	0.665
	Sys. 1	5	13.24	0.869	11.16	0.791	9.98	0.720	8.68	0.641
		10	13.25	0.871	11.06	0.795	9.69	0.712	8.86	0.649
		15	13.33	0.869	11.06	0.794	9.93	0.718	8.56	0.646
	Sys. 2	5	13.14	0.878	11.63	0.813	10.44	0.734	8.82	0.662
		10	13.23	0.877	11.52	0.809	10.11	0.731	8.77	0.652
		15	13.22	0.876	11.31	0.809	10.12	0.727	8.68	0.655
	Sys. 3	5	13.34	0.875	11.47	0.807	10.55	0.739	8.97	0.659
		10	13.53	0.878	11.26	0.805	10.37	0.736	9.10	0.670
		15	13.39	0.877	11.33	0.807	10.41	0.734	9.02	0.673

Table 2: The performance of PLDA, PLDA-UP and the three fast scoring systems on the truncated speech data from NIST 2012 SRE.

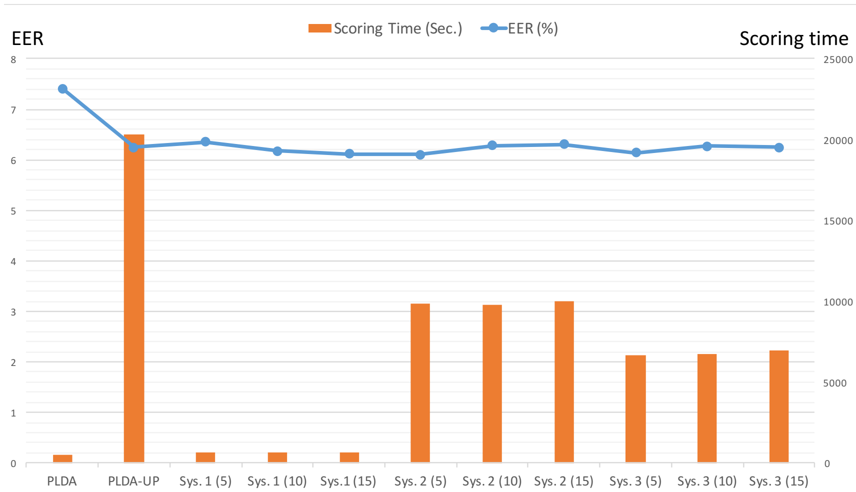

Figure 2: A bar chart showing the EERs and total scoring time of PLDA, PLDA-UP and the three fast scoring systems with different numbers of i-vector groups. For each system, the number inside the parenthesis indicates the number of i-vector groups for that fast scoring system.

6.2. Running time

The total scoring time and its breakdown for different scoring methods in CC2 of NIST 2012 SRE are shown in Table 3. Apparently, the conventional PLDA is the most economical in term of computational cost, as it only involves vector-matrix multiplications during scoring. By contrast, the PLDA-UP is the most computational expensive method, with scoring time 44 times that of the conventional PLDA. The most computational expensive part of PLDA-UP is the evaluation of Eqs. 21–24, which takes up over 60% of the scoring time. Besides Eqs. 21–24, the preprocessing of covariance matrices is also computationally expensive, taking up about 30% of the scoring time. Because our fast scoring systems do not involve utterance-dependent loading matrices, computations in Eqs. 21–24 can be done before verification, thus the scoring time is greatly reduced. However, for System 2 and System 3, we still need to preprocess the covariance matrices of test utterances, which occupies most of the scoring time of these two systems. Besides, System 2 also requires to perform eigendecomposition, which makes it the slowest one among the three systems. For System 1, because the only extra computation besides the scoring function is the simple scalar comparison in Eq. 33, its scoring time is very close to that of the conventional PLDA.

7. Conclusion

In this paper, we proposed a fast scoring method for PLDA with uncertainty propagation (UP). The utterance-dependent loading matrices in UP is replaced by similar ones obtained from development data. The experiments in NIST 2012 have shown that the proposed methods have the same ability to deal with short utterances as UP while the computational cost can be reduced to the one very close to that of the conventional PLDA. The proposed method has important implication in the real-life speaker verification, since in most applications the utterance lengths are difficult to control and computation cost is one of the main concerns beside performance.

8. Acknowledgment

This work was supported in part by The RGC of Hong Kong SAR (Grant Nos. PolyU 152117/14E and PolyU 152068/15E) and in part by the Taiwan MOST with Grant 105-2221-E-009-137-MY2.

References

[1] S. O. Sadjadi, J. Pelecanos, S. Ganapathy, The IBM speaker recognition system: Recent advances and error analysis, arXiv preprint arXiv:1605.01635.

[2] M. W. Mak, R. C. Hsiao, B. Mak, A comparison of various adaptation methods for speaker verification with limited enrollment data, in: Proc. ICASSP, Vol. 1, IEEE, 2006, pp. 929–932.

Method	Task	Time (Sec.)	% of Total Time
PLDA	Preprocess i-vectors in Eq. 9	11	2.37%
	Scoring in Eq.12	179	38.66%
	Other operations	273	58.96%
	Overall	463	100.00%
PLDA-UP	Preprocess i-vectors in Eq. 9	11	0.05%
	Preprocess $\mathbf{L}_t^{-1}$ in Eq. 17	5966	29.32%
	Evaluate $\mathbf{A}_{s,t}, \mathbf{B}_{s,t}, \mathbf{C}_{s,t}, D_{s,t}$ in Eq. 21-24	12485	61.37%
	Scoring in Eq. 20	294	1.44%
	Other operations	1585	7.79%
	Overall	20341	100.00%
Sys. 1	Preprocess i-vectors in Eq. 9	11	1.7%
	Scalar comparison in Eq. 33	11	1.7%
	Scoring in Eq.36	306	47.29%
	Other operations	319	49.3%
	Overall	647	100.00%
Sys. 2	Preprocess i-vectors in Eq. 9	11	0.11%
	Preprocess $\mathbf{L}_t^{-1}$ in Eq. 17	5966	60.33%
	Compute eigenvalues of $\mathbf{\Lambda}_t$	3275	33.12%
	Scalar comparison in Eq. 35	11	0.11%
	Scoring in Eq.36	306	3.09%
	Other operations	319	3.22%
	Overall	9888	100.00%
Sys. 3	Preprocess in Eq. 9	11	0.16%
	Preprocess $\mathbf{L}_t^{-1}$ in Eq. 17	5966	89.93%
	Compute the traces of $\mathbf{\Lambda}_t$	21	0.31%
	Scalar comparison in Eq. 35	11	0.16%
	Scoring in Eq.36	306	4.61%
	Other operations	319	4.8%
	Overall	6634	100.00%

Table 3: Detailed timing reports obtained by Matlab Profiler for experiments in CC2. We used five loading matrices (K=5) for each fast scoring system in the experiments. See Table 5.2 for the configurations of Sys. 1–3.

[3] R. Vogt, S. Sridharan, M. Mason, Making confident speaker verification decisions with minimal speech, IEEE Transactions on Audio, Speech, and Language Processing 18 (6) (2010) 1182–1192.

[4] A. Kanagasundaram, R. Vogt, D. B. Dean, S. Sridharan, M. W. Mason, I-vector based speaker recognition on short utterances, in: Proc. ISCA, IEEE, 2011, pp. 2341–2344.

[5] T. Stafylakis, P. Kenny, P. Ouellet, J. Perez, M. Kockmann, P. Dumouchel, I-vector/PLDA variants for text-dependent speaker recognition, Tech. rep., Centre de Recherche Informatique de Montreal (CRIM) (2013).

[6] A. Larcher, K. A. Lee, B. Ma, H. Li, Text-dependent speaker verification: Classifiers, databases and RSR2015, Speech Communication 60 (2014) 56–77.

[7] T. Hasan, R. Saeidi, J. H. Hansen, D. A. van Leeuwen, Duration mismatch compensation for i-vector based speaker recognition systems, in: Proc. ICASSP, IEEE, 2013, pp. 7663–7667.

[8] L. Li, D. Wang, C. Zhang, T. F. Zheng, Improving short utterance speaker recognition by modeling speech unit classes, IEEE/ACM Transactions on Audio, Speech, and Language Processing 24 (6) (2016) 1129–1139.

[9] P. Kenny, T. Stafylakis, P. Ouellet, M. J. Alam, P. Dumouchel, PLDA for speaker verification with utterances of arbitrary duration, in: Proc. ICASSP, IEEE, 2013, pp. 7649–7653.

[10] S. Cumani, O. Plchot, P. Laface, On the use of i–vector posterior distributions in probabilistic linear discriminant analysis, IEEE/ACM Transactions on Audio, Speech, and Language Processing 22 (4) (2014) 846–857.

[11] S. Cumani, Fast scoring of full posterior PLDA models, IEEE/ACM Transactions on Audio, Speech, and Language Processing 23 (11) (2015) 2036–2045.

[12] W. W. Lin, M. W. Mak, Fast scoring for PLDA with uncertainty propagation, in: Odyssey The Speaker and Language Recognition Workshop, 2016, pp. 31–38.

[13] P. Kenny, Joint factor analysis of speaker and session variability: Theory and algorithms, Tech. rep., Centre de Recherche Informatique de Montreal (CRIM) (2005).

[14] P. Kenny, G. Boulianne, P. Ouellet, P. Dumouchel, Joint factor analysis versus eigenchannels in speaker recognition, IEEE Transactions on Audio, Speech, and Language Processing 15 (4) (2007) 1435–1447.

[15] M. W. Mak, Lecture notes on uncertainty propagation for i-vector/PLDA speaker verification, Tech. rep., The Hong Kong Polytechnic University (2015).

[16] P. Kenny, Bayesian speaker verification with heavy-tailed priors, in: Odyssey, 2010, pp. 1–14.

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

[18] A. O. Hatch, S. S. Kajarekar, A. Stolcke, Within-class covariance normalization for SVM-based speaker recognition, in: Proc. Interspeech, 2006.

[19] W. Rao, M. W. Mak, K. A. Lee, Normalization of total variability matrix for i-vector/PLDA speaker verification, in: Proc. ICASSP, IEEE, 2015, pp. 4180– 4184.

[20] I. L. Dryden, A. Koloydenko, D. Zhou, Non-euclidean statistics for covariance matrices, with applications to diffusion tensor imaging, The Annals of Applied Statistics (2009) 1102–1123.

[21] MathWorks, Box plot documentation, `http://www.mathworks.com/help/stats/boxplot.html`.

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

[23] M. W. Mak, H. B. Yu, A study of voice activity detection techniques for nist speaker recognition evaluations, Computer Speech & Language 28 (1) (2014) 295–313.

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