A New Fast and Memory Effective I-Vector Extraction Based On Factor Analysis of KLD Derived GMM Supervector

Zhi-Yi Li, Wei-Qiang Zhang, Yao Tian, Jia Liu

Tsinghua National Laboratory for Information Science and Technology
Department of Electronic Engineering, Tsinghua University, Beijing 100084, China
lizhiyi06@mails.tsinghua.edu.cn

Abstract

At present, i-vector model has become the state-of-the-art technology for speaker recognition. It represents speech utterance to a low-dimensional fix-length compact i-vector. For some real application, i-vector extraction procedure is relatively slow and requires too much memories. Some numerical approximation based fast extraction methods have been proposed to speed up the computation and to save memory meanwhile. However they are all at the expense of more or less performance degradation. From a novel model approximation viewpoint, we first propose a novel fast i-vector extraction method based on subspace factor analysis from Kullback-Leibler divergence derived Gaussian Mixture Models supervector. Experimental results on NIST SRE datasets demonstrate that the proposed method is more faster and performs more better than all the existing methods at the similar run time ratio. Besides, due to the different modeling viewpoint, we proposed a combination method with factorized subspace based extraction. This method can avoid the accuracy degradation and even can perform better than the standard one, while its extraction speed can be 10 times faster than the standard method.

Index Terms: i-vector, speaker recognition, fast extraction, Kullback-Leibler divergence, factor analysis

1. Introduction

Speaker recognition refers to recognizing speakers’ identities from their voices. It can be widely used in many areas [1]. Over the past several years, speaker recognition technology has made great progress and has been promoted from laboratory research to practical applications. For mass data processing applications, fast computing of speaker recognition system is very necessary and for some portable platforms, both the computing and memory resources of devices are relatively limited [2].

At present, inspired by Joint Factor Analysis (JFA) [3] and some following work [4][5], Dehak et al has recently proposed the state-of-the-art i-vector based speaker recognition technology in [6]. Similar with the previous JFA modeling method, this technology is also based on the widely known Gaussian Mixture Model-Universal Background Model (GMM-UBM) framework and subspace factor analysis. Unlike JFA, which models two separated speaker subspace and channel subspace, i-vector modeling method is based on only one single space, named as total variability space, which simultaneously contains the speaker and channel variabilities. Hence, during i-vector extraction, there is no distinction between the speaker effects and the channel effects in GMM supervector space. The reason of this simplified modeling is that related experiments show the channel factors in JFA also contain information to do speaker recognition [5].

In i-vector modeling method, variable-length speech utterance can represent to be a low-dimensional fix-length compact vector referred to as i-vector. As the matter of fact, i-vector is the posterior mean vector of hidden variable of factor analysis model. This compact representation makes it very convenient to try various fast and effective channel compensation approaches and classifiers.

However, standard i-vector extraction procedure is relatively slow and requires too much memories for real application. Some i-vector approximate extraction methods have been been proposed to speed up the procedure and saving memory at the expense of the performance degradation. In [7], one fast i-vector extraction method based on subspace eigen-decomposition (SED) is proposed by diagonalization of i-vector posterior covariance matrix. This method is fast and memory effective, but have large performance degradation. In [2], an improved fast i-vector extraction method is proposed based on factorized sub-space (FSS). This method approximates the total variability matrix by a linear combination and rotation of the atoms in a shared dictionary. This method can obtain more accurate results with less memory requirement, but it is not as fast as the former eigen-decomposition technique. The above existing methods are both based on the numeric approximation. In this paper, from a different model approximation viewpoint, we propose a novel fast extraction method based on subspace factor analysis of Kullback-Leibler divergence derived GMM supervector from.

The content of this paper is organized as follows: the standard i-vector extraction procedure is introduced in section 2. Section 3 recalls the fast i-vector approximate extraction method based on subspace eigen-decomposition (SED). Section 4 recalls the other fast i-vector approximate extraction method based on factorized sub-space (FSS). Section 5 illustrates our proposed novel fast extraction method based on subspace factor analysis of Kullback-Leibler divergence derived GMM supervector. In Section 6, the experimental results are demonstrated and Section 7 gives the conclusions.

2. Standard i-vector extraction

In i-vector model [6], speaker and channel information in GMM mean supervector \( \mathbf{M} \) is supposed to be in a single common low-dimensional subspace.

\[
\mathbf{M} = \mathbf{n} + \mathbf{T} \mathbf{w}
\]  

where \( \mathbf{n} \) is a both speaker and channel independent mean component (usually using UBM supervector instead). Matrix \( \mathbf{T} \) is a low-rank rectangular subspace named as total variability sub-
space. It has \( C \times F \) rows and \( M \) columns, where \( C \) is the GMM mixture number, \( F \) is feature dimension and \( M \) is the dimension of subspace. \( w \) is a random \( M \) dimensional vector with a standard normal prior distribution. In order to simplify the expression, matrix \( T \) and first-order statistics can be normalized by \( \Sigma^{-1/2} \), where \( \Sigma \) stands for the block-diagonal matrix whose cth diagonal element \( \Sigma_c \) is the covariance matrix of UBM’s cth mixture Gaussian component as in [2].

Following the notations in [2], from the feature vector set \( \mathcal{X}_s = \{ x_1, x_2, \ldots, x_f \} \) of a speech utterance, s, the i-vector \( w_s \) of this utterance can be obtained by computing the mean point estimation of the posterior distribution \( p(w_s | \mathcal{X}_s) \). As shown in [2], i-vector \( w_s \) can be equivalently computed using normalized subspace matrix \( T \leftarrow \Sigma_c^{-1/2}T \) and normalized first-order statistics \( f_s \leftarrow \Sigma_c^{-1/2}f_s \) as follows:

\[
 w_s = L_c^{-1}T^T f_s
\]

where \( L_c \) is the precision matrix of the \( w_s \)’s posterior distribution, which can be computed as follows:

\[
 L_c = I + T^T N_c T = I + \sum_c n_{s,c} T^T_c T_c
\]

where \( n_{s,c} = \sum_t \gamma_{s,c,d} \) is the zero-order statistic of features corresponding to the cth UBM component and \( \gamma_{s,c,d} \) is the posterior occupation probability. \( N_c \) is the \( C \times F \) super-vector stacking by \( n_{s,c} \). \( T_c \) is the sub-matrix of \( T \) corresponding to the cth mixture component and normalized \( f_s \) is a \( C \times F \) super-vector, which is stacking by the normalized first-order statistics \( f_{s,c} \), as follows:

\[
 f_{s,c} = \Sigma_c^{-1/2} \left( \sum_t (\gamma_{s,c} x_t) - n_{s,c} m_c \right)
\]

In (2) and (3), it is easily to be seen that the extraction time of i-vector is mainly spent on the computation of \( L \) matrix and its inversion. As the analysis in [8], the computation complexity of standard i-vector extraction can be expressed as \( O(CM^2 + M^3 + CFM) \). Since memory cost of this method is mainly on storing subspace matrix \( T \) and all pre-computed matrices \( T^T_c T_c \), the storage complexity can be expressed as \( O(CM^2 + CFM) \).

### 3. Subspace eigen-decomposition based fast i-vector extraction

In [2] and [7], a subspace eigen-decomposition (SED) based fast i-vector extraction method is presented. As described in [2], Since \( T^T_c T_c \) is symmetric and semi-definite positive, it can be eigen-decomposed to be the matrix production as follows:

\[
 T^T_c T_c = G_c D_c G_c^T
\]

where \( G_c \) is an orthogonal matrix, and matrix \( D_c \) is diagonal. Then, \( D_c \) can be equivalently expressed as:

\[
 D_c = (T_c G_c)^T (T_c G_c) = G_c^T T^T_c T_c G_c
\]

For speeding up the computation of matrix \( L_c \) and its inverse, in this method, a single common orthogonal approximated matrix \( Q \) is used to replace the each matrix \( G_c \). After replacement, due to the production of \( D_c \) no more diagonal, this method forces it to be diagonal one:

\[
 \tilde{D}_c = \text{Diag} \{ Q^T T^T_c T_c Q \}
\]

where function \( \text{Diag} \{ \cdot \} \) is to setting off-diagonal elements of matrix zeros. Common matrix \( Q \) can be obtained by eigen-decomposition of weighted matrix like in [7].

\[
 W = \sum_c \omega_c T^T_c T_c = Q \Lambda Q^{-1}
\]

where \( \omega_c \) is the weight of the UBM’s cth mixture component. In this method, approximated i-vector can be fast extraction as follows:

\[
 w_s = \tilde{L}_c^{-1} T^T f_s
\]

\[
 \tilde{L}_c = Q(I + \sum_c n_{s,c} \tilde{D}_c)Q^T
\]

Computation complexity of this fast i-vector extraction can be shown as \( O(CM + M + CFM) \) and the storage complexity is \( O(CM + M^2 + CFM) \). Compared to the standard extraction, this approximation method is very fast and memory effective but its performance is a little far from the standard one due to its off-diagonal information discarding [8].

### 4. Factorized subspace based fast i-vector extraction

In order to overcome the performance degradation of last method, one improved fast i-vector extraction method is proposed in [2]. The target of this method is to get accurate estimation of \( T \) matrix by factorized subspace.

In equation (6), since \( D_c \) is diagonal, the production component \( T_c G_c \) can be furtherly decomposed to the production of a orthonormal \( F \times F \) matrix \( O \) and a \( F \times M \) sparse matrix \( \Pi_c^{(n)} \), which has at most one non-null element per row. Then \( T_c \) can be factorized as follows:

\[
 T_c = O_c \Pi_c^{(n)} G_c^T
\]

Unlike in SED, one key point of this method is using a common single but larger \( K \times M \) dictionary matrix \( Q \) to replace the every \( G_c \). Corresponding to the changed size of \( Q \), \( \Pi_c^{(n)} \) is resized to new \( F \times K \) matrix \( \Pi_c \). The new \( \Pi_c \) also satisfies the same sparsity condition. Such each matrix \( T_c \) can now be approximated as follows:

\[
 \tilde{T}_c \approx O_c \Pi_c Q
\]

As described in [2], the approximation \( \tilde{T}_c \) can be seen as a linear combination and rotation of \( F \) vectors, which are selected from a set of \( K \) atoms collected in a single common \( K \times M \) dictionary matrix \( Q \). Compared to the former SED method, the advantage of this one can make the estimation of \( T_c \) more accurate by using larger dictionary \( Q \).

In this method, all factorized matrices \( O_c \) and \( \Pi_c \) can be iteratively estimated by minimizing a weighted average square norm of the differences between raw standard \( T_c \) and their approximations \( \tilde{T}_c \) as:

\[
 \min_{O_c, \Pi_c, Q_c} \sum_c \omega_c \| T_c - O_c \Pi_c Q_c \|^2
\]

Based on the all approximation \( \tilde{T}_c \), another key point of this method is using the Conjugate Gradient (CG) to solve the linear equation instead of computation of the inversion matrix of \( L_c \). Here, CG solver can take advantages of sparsity of matrix \( \Pi_c \) to save the extraction time and storage requirement.

\[
 \tilde{L}_c w_s = \sum_c \tilde{T}_c f_{s,c} = Q^T \sum_c O_c \Pi_c f_{s,c}
\]
where the matrix \( \tilde{L}_s \) can be computed as:
\[
\tilde{L}_s = I + \sum_c n_{s,c} (O \cdot \Pi Q)^T (O \cdot \Pi Q)
\]
\[
= I + Q^T \sum_c n_{s,c} \Pi s \cdot \Pi q
\]
(15)

Similar as former analysis, we give the computation complexity of this extraction method as \( O(MK) + O(CF) + O(CK) + O(CF^2) \), where \( R \) is denoted as the iteration number of CG. We can see that the computation complexity is affected by both the dictionary size and the iteration number of CG. The storage complexity can also be given as \( O(MK) + O(CF) + O(CK) + O(CF^2) \).

5. Factor analysis of Kullback-Leibler derived GMM supervector

It can be seen that both the above fast extraction methods are based on the numerical approximation of standard T or \( T^T T \) matrix either by eigen-decomposition or by factorized subspace. By reviewing the standard i-vector extraction in (2) and (3), it can be obviously observed that if we can share the \( L \) and \( L^{-1} \) matrix for all the utterances, the bottleneck problem of the extraction will be solved completely. Such how to share the \( L \) matrix reasonably is a new following problem to be solved.

Based on the above motivation, we propose a novel i-vector fast extraction method based on the Kullback-Leibler subspace factor analysis from a model approximation viewpoint. In this method, we derive a new Hilbert space based Kullback-Leibler divergence between two adapted GMM models \( \lambda_{UBM} \) and \( \lambda_{UBM} \) from UBM with maximum a posterior (MAP).

\[
D(\lambda_s||\lambda_{UBM}) = \int \lambda_s(x) \log \left( \frac{\lambda_s(x)}{\lambda_{UBM}(x)} \right) dx
\]
(16)

In fact, since standard the Kullback-Leibler divergence of two GMMs does not satisfy the Mercer condition, it can not give the closed form expression directly. Under the log-sum inequality and only mean adaptation, we can use the upper-bounder approximation of KL divergence to give the closed form solution just like in [9].

\[
D(\lambda_s||\lambda_{UBM}) \leq \frac{1}{2} \sum_c \omega_c \left( \mathcal{N}(m_c^s; \Sigma_c) || \mathcal{N}(m_c^s; \Sigma_c) \right)
\]
\[
= \frac{1}{2} \sum_c \omega_c (m_c^s - m_c_b^s)^T \Sigma_c^{-1} (m_c^s - m_c_b^s)
\]
(17)

From the solution in (17), we can derived a new composed-parameter built Hilbert linear space different from the GMM mean supervector space in standard i-vector model. In this subspace, a mean adapted GMM is represented as a single weighted GMM mean supervector mean, which is composed of the weights and covariances of UBM and the adapted GMM mean vectors.

\[
\mathcal{M} = \sqrt{\Sigma} \Sigma^{-\frac{1}{2}} m
\]
(18)

The inner product of two points in this space has been successfully used as the linear kernel function in GSV-SVM speaker recognition system as in [9]. Given a sequence set of feature vectors \( \chi_s = \{x_1, x_2, ..., x_t \} \) extracted from a speech utterance \( s \), the \( c \)th mixture component \( m_c \) of adapted GMM can be obtained by the zero-order statistic \( n_{c} \) and no-normalized first-order origin statistic \( f_c^{(0)} \):

\[
\bar{m}_c = \alpha (f_c^{(0)}) n_c + (1 - \alpha) m_c^{(UBM)}
\]
(19)

where the adaptation coefficient \( \alpha \) controls the balance between old ubm and new estimation as in [10]. In the above KL divergence derived GMM supervector, the zero-order statistic and the first-order statistic are well integrated into a new weighted GMM supervector.

In this new weighted GMM supervector Hilbert space, we model the total variability subspace as:

\[
\mathcal{M} = \mu + \mathcal{T} \tilde{w}
\]
(20)

where \( \mu \) is also the both speaker and channel independent mean component, but here can not use the UBM mean supervector instead. \( \tilde{w} \) is the also a \( M \) dimensional random i-vector with a standard normal prior distribution. \( \mathcal{T} \) is the approximation of standard total variability subspace \( T \), which can be estimation by Expectation Maximization (EM) based factor analysis method [11] as follows.

E-step: In the E-step, we aim to estimate the posteriori precision matrix \( \mathcal{L} \), posteriori first-order moment \( E(\tilde{w}_s) \) and posteriori second-order moment \( E(\tilde{w}_s \tilde{w}_s^T) \).

\[
\mathcal{L} = I + \mathcal{T} \Sigma^{-1} \mathcal{T}^T
\]
(21)

\[
E(\tilde{w}_s) = L^{-1} \mathcal{T} \Sigma^{-1} (M_s - \mu)
\]
(22)

\[
E(\tilde{w}_s \tilde{w}_s^T) = L^{-1} + E(\tilde{w}_s) E(\tilde{w}_s)^T
\]
(23)

M-step: In the M-Step, we aim to update the parameter set \( \{ \mu, \mathcal{T}, \Sigma \} \) as follows:

\[
\mu = \frac{1}{S} \sum_s \tilde{M}_s
\]
(24)

\[
\mathcal{T} = \left( \sum_s (\tilde{M}_s - \mu) E(\tilde{w}_s)^T \right) \left( \sum_s E(\tilde{w}_s \tilde{w}_s^T) \right)^{-1}
\]
(25)

\[
\Sigma = \sum_s \text{Diag} \left( (\tilde{M}_s - \mu) (\tilde{M}_s - \mu)^T - \mathcal{T} E(\tilde{w}_s) (\tilde{M}_s - \mu) \right)
\]
(26)

where the \( S \) is the utterance number in training set and the \( \Sigma \) in this section means the residual noise term of this modeling method. After about 20 iterations, approximation subspace \( \mathcal{T} \) can be converged.

Using the approximation \( \mathcal{T} \), we can extract the i-vector for utterance \( s \) as follows:

\[
\tilde{w}_s = L^{-1} \mathcal{T} \Sigma^{-1} (\tilde{M}_s - \mu)
\]
(27)

where the precision matrix \( L \) of the \( \tilde{w}_s \)’s posterior distribution can be computed as follows:

\[
L = I + \mathcal{T} \Sigma^{-1} \mathcal{T}^T
\]
(28)

In the above extraction procedure, posterior precision matrix \( L \) is unrelated to the special utterance \( s \), such we can share a single common matrix \( L \) and its inversion \( L^{-1} \) for all the utterance, so it can be the fastest i-vector extraction method.

Similar to the former analysis of complexity, the computation complexity of this method can be shown as \( O(CF^2) \). We can see that the computation complexity is minimum among the all four extraction methods. The storage complexity can be expressed as \( O(M^2) + O(CF^2) \), so it is also more memory efficient than the standard i-vector extraction.
6. Experimental Results

In our experiments, we focus on testing the both accuracy and the speed of all above four i-vector extraction methods, which are standard baseline, SED, FSS and our proposed KLF methods, respectively.

To be fair, we compare these four methods based on the same frontend. In the frontend, we extracted 39-dimensional (13 basic+Δ+ΔΔ) Mel frequency cepstral coefficients (MFCC) with 25 ms frame length and 10 ms frame shift [12]. The gender-dependent 2048 diagonal covariance UBM are modeled. The dimension of i-vector is fixed to 400. And we used two popular i-vector classifiers here. One is the cosine distance scoring (CDS) [6] after doing the linear discriminant analysis (LDA) [13] and within class covariance normalization (WCCN) [14] and the dimension of LDA is set to 200. The other is the Gaussian Probability linear discriminant analysis (GPLDA) [15][16] after doing length normalization [17]. The speaker factor in GPLDA was fixed to 250 dimensions and the channel factor was using the full-rank dimension. The data for training are all from the NIST SRE 04/05/06 and Switchboard I and II.

The recognition performance is shown in terms of both Equal Error Rate (EER) and Minimum Detection Cost Functions (MinDCF08 and MinDCF10) defined in the SRE 2008 and 2010 plans respectively [18] [19]. Table 1 shows the performance of all four methods on the NIST SRE 2008 short2-short3 tel-tel female condition using the CDS classifier. In FSS method, we test two dictionary sizes (2K and 5K) and two terminal thresholds (10 and 100) of CG solver. The results in tables demonstrate that the proposed KLF method is the about 30 times faster than the standard one and has the best performance and the smallest accuracy degradation at the similar time ratio compared with the SED and FSS(2k-100) method. Since FSS(5k-10) method can be closer to the standard one, we will use this parameter setting for the following experiments.

Table 1: The performance of four methods on the NIST SRE 2008 short2-short3 tel-tel female condition using CDS.

<table>
<thead>
<tr>
<th>sre 2008 female short2-short3</th>
<th>Memory (MB)</th>
<th>Time Ratio</th>
<th>EER (%)</th>
<th>Min DCF08</th>
</tr>
</thead>
<tbody>
<tr>
<td>baseline</td>
<td>755</td>
<td>15.7</td>
<td>6.43</td>
<td>0.319</td>
</tr>
<tr>
<td>SED</td>
<td>128</td>
<td>1.0</td>
<td>6.63</td>
<td>0.352</td>
</tr>
<tr>
<td>FSS (2K-10)</td>
<td>36</td>
<td>0.97</td>
<td>6.73</td>
<td>0.313</td>
</tr>
<tr>
<td>FSS (2K-100)</td>
<td>36</td>
<td>0.52</td>
<td>7.11</td>
<td>0.356</td>
</tr>
<tr>
<td>FSS (5K-10)*</td>
<td>62</td>
<td>1.14</td>
<td>6.56</td>
<td>0.324</td>
</tr>
<tr>
<td>FSS (5K-100)</td>
<td>62</td>
<td>0.85</td>
<td>6.86</td>
<td>0.341</td>
</tr>
<tr>
<td>KLF</td>
<td>129</td>
<td>0.42</td>
<td>6.40</td>
<td>0.328</td>
</tr>
<tr>
<td>KLF + FSS*</td>
<td>190</td>
<td>1.50</td>
<td>6.28</td>
<td>0.309</td>
</tr>
</tbody>
</table>

Where the run time of SED method as the reference value 1. Besides, since the different modeling viewpoint of KLF, we experiment and give the the performance of complementary combination of KLF and FSS method. The results demonstrates that the combination of two fast methods can not only avoid the performance degradation but also can obtain a little improvement compared with the standard method and the speed is 10 times faster than the standard one. Table 2 gives the performance of four methods on both male and female condition using GPLDA.

Table 2: The performance of four methods on the NIST SRE 2008 short2-short3 tel-tel female and male condition using GPLDA.

<table>
<thead>
<tr>
<th>sre 2008 short2-short3</th>
<th>(f)(%) EER</th>
<th>Min DCF08</th>
<th>(m)(%) EER</th>
<th>Min DCF08</th>
</tr>
</thead>
<tbody>
<tr>
<td>baseline</td>
<td>5.73</td>
<td>0.301</td>
<td>3.51</td>
<td>0.197</td>
</tr>
<tr>
<td>SED</td>
<td>6.63</td>
<td>0.324</td>
<td>4.29</td>
<td>0.228</td>
</tr>
<tr>
<td>FSS*</td>
<td>5.96</td>
<td>0.301</td>
<td>3.90</td>
<td>0.204</td>
</tr>
<tr>
<td>KLF</td>
<td>5.49</td>
<td>0.306</td>
<td>4.27</td>
<td>0.238</td>
</tr>
<tr>
<td>KLF + FSS*</td>
<td>5.43</td>
<td>0.288</td>
<td>3.50</td>
<td>0.196</td>
</tr>
</tbody>
</table>

7. Conclusions

In this work, we proposed a new fast memory effective i-vector extraction method based on factor analysis of Kullback-Leibler divergence derived GMM Supervector from a novel model approximation viewpoint. This method is more faster than existing numerical approximation based methods and can obtained the best performance at the similar extraction time ratio. Besides, experimental results demonstrate that it is complementary with existing factorized subspace based fast extraction method. The combination of two method can avoid the accuracy degradation and even perform better than the standard one while its speed can be 10 times faster than standard one.

8. Acknowledgements

This work was supported by the National Natural Science Foundation of China under Grant No.61370034 and No.61273268.
9. References


