COMPARISON OF SCORING METHODS USED IN SPEAKER RECOGNITION WITH JOINT FACTOR ANALYSIS

Ondřej Glembek[1], *Lukáš Burget*[1], *Najim Dehak*[2,3], *Niko Brümmer*[4], *Patrick Kenny*[2]

[1]Speech@FIT group, Faculty of Information Technology, Brno University of Technology, Czech Republic
[2]Centre de Recherche Informatique de Montréal (CRIM), Montréal, Canada
[3]École de Technologie Supérieure (ETS), Montréal, Canada
[4]Agnitio, Stellenbosch, South Africa
{glembek,burget}@fit.vutbr.cz, {najim.dehak,patrick.kenny}@crim.ca,
nbrummer@agnitio.es

ABSTRACT

The aim of this paper is to compare different log-likelihood scoring methods, that different sites used in the latest state-of-the-art Joint Factor Analysis (JFA) Speaker Recognition systems. The algorithms use various assumptions and have been derived from various approximations of the objective functions of JFA. We compare the techniques in terms of speed and performance. We show, that approximations of the true log-likelihood ratio (LLR) may lead to significant speedup without any loss in performance.

***Index Terms*—** GMM, fast scoring, speaker recognition, joint factor analysis

1. INTRODUCTION

Joint Factor Analysis (JFA) has become the state-of-the-art technique in the problem of speaker recognition[1]. It has been proposed to model the speaker and session variabilities in the parameter space of the Gaussian Mixture Model (GMM) [1]. The variabilities are determined by subspaces in the parameter space, commonly called the *hyper-parameters*.

Many sites used JFA in the latest NIST evaluations, however they report their results using different scoring methods [2, 3, 4]. The aim of this paper is to compare these techniques in terms of speed and performance. Note that we expect the reader to be familiar with JFA. For an introduction to this technique, we refer the reader to [5, 2, 6]

The theory about JFA and each technique is given in Sec. 2. Starting with the conventional frame-by-frame GMM evaluation in Sec. 2.1, where the whole feature file of each utterance is processed, the sections 2.2 to 2.5 describe methods which work with the collected statistics only and which differ mostly in the way they treat channel compensation. In Sec. 2.2, integration over the whole distribution of channel factors for the given test utterance is performed. In Sec. 2.3, the likelihood of each utterance given testing model is computed using a channel point estimate. In Sec. 2.4, the channel factor point estimate is estimated using UBM only. In Sec 2.5, the formula is further simplified by using the first order Taylor series approximation.

2. THEORETICAL BACKGROUND

Joint factor analysis is a model used to treat the problem of speaker and session variability in GMMs. In this model, each speaker is represented by the means, covariance, and weights of a mixture of C multivariate Gaussian densities defined in some continuous feature space of dimension F. The GMM for a target speaker is obtained by adapting the Universal Background Model (UBM) mean parameters. In Joint Factor Analysis [2], the basic assumption is that a speaker- and channel- dependent supervector of means $\mathbf{M}$ can be decomposed into a sum of two supervectors: a speaker supervector $\mathbf{s}$ and a channel supervector $\mathbf{c}$

$$\mathbf{M} = \mathbf{s} + \mathbf{c}, \tag{1}$$

where $\mathbf{s}$ and $\mathbf{c}$ are normally distributed. In [5], Kenny et al. described how the speaker dependent supervector and channel dependent supervector can be represented in low dimensional spaces. The first term in the right hand side of (1) is modeled by assuming that if $\mathbf{s}$ is the speaker supervector for a randomly chosen speaker then

$$\mathbf{s} = \mathbf{m} + \mathbf{V}\mathbf{y} + \mathbf{D}\mathbf{z}, \tag{2}$$

where $\mathbf{m}$ is the speaker and channel independent supervector (UBM), $\mathbf{D}$ is a diagonal matrix, $\mathbf{V}$ is a rectangular matrix of low rank and $\mathbf{y}$ and $\mathbf{z}$ are independent random vectors having standard normal distributions. In other words, $\mathbf{s}$ is assumed to be normally distributed with mean $\mathbf{m}$ and covariance matrix $\mathbf{V}\mathbf{V}^* + \mathbf{D}\mathbf{D}^*$. The components of $\mathbf{y}$ and $\mathbf{z}$ are respectively the speaker and common *factors*.

The channel-dependent supervector $\mathbf{c}$, which represents the channel effect in an utterance, is assumed to be distributed according to

$$\mathbf{c} = \mathbf{U}\mathbf{x}, \tag{3}$$

where $\mathbf{U}$ is a rectangular matrix of low rank (known as eigenchannel matrix), $\mathbf{x}$ is a vector distributed with standard normal distribution. This is equivalent to saying that $\mathbf{c}$ is normally distributed with zero mean and covariance $\mathbf{U}\mathbf{U}^*$. The components of $\mathbf{x}$ are the channel factors in factor analysis modeling.

The underlying task in JFA is to train the hyperparameters $\mathbf{U}$, $\mathbf{V}$, and $\mathbf{D}$ on a large training set. In the Bayesian framework, posterior distribution of the factors (knowing their priors) can be computed using the enrollment data. The likelihood of test utterance $\mathcal{X}$ is then computed by integrating over the posterior distribution of $\mathbf{y}$ and $\mathbf{z}$,

[1] In the meaning of speaker verification

and the prior distribution of $\mathbf{x}$ [7]. In [8], it was later shown, that using mere MAP point estimates of $\mathbf{y}$ and $\mathbf{z}$ is sufficient. Still, integration over the prior distribution of $\mathbf{x}$ was performed. We will further show, that using the MAP point estimate of $\mathbf{x}$ gives comparable results. Scoring is understood as computing the log-likelihood ratio (LLR) between the target speaker model $\mathbf{s}$ and the UBM, for the test utterance $\mathcal{X}$.

There are many ways in which JFA can be trained and which different sites have experimented with. Not only the training algorithms differ, but also the results were reported using different scoring strategies.

2.1. Frame by Frame

Frame-by-Frame is based on a full GMM log-likelihood evaluation. The log-likelihood of utterance $\mathcal{X}$ and model $\mathbf{s}$ is computed as an average frame log-likelihood [2]. It is practically infeasible to integrate out the channel, therefore MAP point estimates are used for channel factors $\mathbf{x}$ (as well as for speaker and common factors $\mathbf{y}$ and $\mathbf{z}$). The formula is as follows

$$\log P(\mathcal{X}|\mathbf{s}) = \sum_{t=1}^{T} \log \sum_{c=1}^{C} w_c \mathcal{N}\left(\mathbf{o}_t; \boldsymbol{\mu}_c, \boldsymbol{\Sigma}_c\right), \tag{4}$$

where $\mathbf{o}_t$ is the feature vector at frame t, T is the length (in frames) for utterance $\mathcal{X}$, C is number of Gaussians in the GMM, and w_c, $\boldsymbol{\Sigma}_c$, and $\boldsymbol{\mu}_c$ the cth Gaussian weight, mean, and covariance matrix, respectively.

2.2. Integrating over Channel Distribution

This approach is based on evaluating an objective function as given by Equation (13) in [2]:

$$P(\mathcal{X}|\mathbf{s}) \quad = \quad \int P(\mathcal{X}|\mathbf{s}, \mathbf{x}) \mathcal{N}(\mathbf{x}; \mathbf{0}, \mathbf{I}) \mathrm{d}\mathbf{x} \tag{5}$$

As was said in the previous paragraph, it would be difficult to evaluate this formula in the frame-by-frame strategy. However, (4) can be approximated by using fixed alignment of frames to Gaussians, i.e., assume that each frame is generated by a single (best scoring) Gaussian. In this case, the likelihood can be evaluated in terms of the sufficient statistics. If the statistics are collected in the Baum-Welch way, the approximation is equal to the GMM EM auxiliary function, which is a lower bound to (5). The closed form (logarithmic) solution is then given as[3]:

$$\begin{aligned} \log \tilde{P}(\mathcal{X}|\mathbf{s}) \quad = \quad & \sum_{c=1}^{C} N_c \log \frac{1}{(2\pi)^{F/2} |\boldsymbol{\Sigma}_c|^{1/2}} \\ & - \frac{1}{2} \mathrm{tr}(\boldsymbol{\Sigma}^{-1} \mathbf{S}_{\mathbf{s}}) - \frac{1}{2} \log|\mathbf{L}| \\ & + \frac{1}{2} \|\mathbf{L}^{-1/2} \mathbf{U}^* \boldsymbol{\Sigma}^{-1} \mathbf{F}_{\mathbf{s}}\|^2. \end{aligned} \tag{6}$$

where for the first term, C is the number of Gaussians, N_c is the data count for Gaussian c, F is the feature vector size, $\boldsymbol{\Sigma}_c$ is covariance matrix for Gaussian c. These numbers will be equal both for UBM and the target model, thus the whole term will cancel out in the computation of the log-likelihood ratio (LLR).

For the second term of (6), $\boldsymbol{\Sigma}$ is the block-diagonal matrix of separate covariance matrices for each Gaussian, $\mathbf{S}_{\mathbf{s}}$ is the second order moment of $\mathcal{X}$ around speaker $\mathbf{s}$ given as

$$\mathbf{S}_{\mathbf{s}} = \mathbf{S} - 2\mathrm{diag}(\mathbf{F}\mathbf{s}^*) + \mathrm{diag}(\mathbf{N}\mathbf{s}\mathbf{s}^*), \tag{7}$$

where $\mathbf{S}$ is the $CF \times CF$ block-diagonal matrix whose diagonal blocks are uncentered second order cumulants $\mathbf{S}_c$. This term is independent of speaker, thus will cancel out in the LLR computation (note that this was the only place where second order cumulants appeared, therefore are not needed for scoring). $\mathbf{F}$ is a $CF \times 1$ vector, obtained by concatenating the first order cumulants. $\mathbf{N}$ is a $CF \times CF$ diagonal matrix, whose diagonal blocks are $N_c \mathbf{I}_F$, i.e., the occupation counts for each Gaussian ($\mathbf{I}_F$ is $F \times F$ identity matrix).

The $\mathbf{L}$ in the third term of (6) is given as

$$\mathbf{L} = \mathbf{I} + \mathbf{U}^* \boldsymbol{\Sigma}^{-1} \mathbf{N} \mathbf{U}, \tag{8}$$

where $\mathbf{I}$ is a $CF \times CF$ identity matrix, $\mathbf{U}$ is the eigenchannel matrix, and the rest is as in the second term. The whole term, however, does not depend on speaker and will cancel out in the LLR computation.

In the fourth term of (6), let $\mathbf{L}^{1/2}$ be a lower triangular matrix, such that

$$\mathbf{L} = \mathbf{L}^{1/2} \mathbf{L}^{1/2*} \tag{9}$$

i.e., $\mathbf{L}^{-1/2}$ is the inverse of the Cholesky decomposition of $\mathbf{L}$.

As was said, terms one and three in (6), and second order cumulants $\mathbf{S}$ in (7) will cancel out. Then the formula for the score is given as

$$\begin{aligned} Q_{\mathrm{int}}(\mathcal{X}|\mathbf{s}) \quad = \quad & \mathrm{tr}(\boldsymbol{\Sigma}^{-1} \mathrm{diag}(\mathbf{F}\mathbf{s}^*)) \\ & + \frac{1}{2} \mathrm{tr}(\boldsymbol{\Sigma}^{-1} \mathrm{diag}(\mathbf{N}\mathbf{s}\mathbf{s}^*)) \\ & + \frac{1}{2} \|\mathbf{L}^{-1/2} \mathbf{U}^* \boldsymbol{\Sigma}^{-1} \mathbf{F}_{\mathbf{s}}\|^2. \end{aligned} \tag{10}$$

2.3. Channel Point Estimate

This function is similar to the previous case, except for the fact, that the channel factor $\mathbf{x}$ is known. This way, there is no need for integrating over the whole distribution of $\mathbf{x}$, and only its point estimate is taken for LLR computation. The formula is directly adopted from [6] (Theorem 1),

$$\begin{aligned} \log \tilde{P}(\mathcal{X}|\mathbf{s}, \mathbf{x}) \quad = \quad & \sum_{c=1}^{C} N_c \log \frac{1}{(2\pi)^{F/2} |\boldsymbol{\Sigma}_c|^{1/2}} \\ & - \frac{1}{2} \mathrm{tr}(\boldsymbol{\Sigma}^{-1} \mathbf{S}) \\ & + \mathbf{M}^* \boldsymbol{\Sigma}^{-1} \mathbf{F} + \frac{1}{2} \mathbf{M}^* \mathbf{N} \boldsymbol{\Sigma}^{-1} \mathbf{M}, \end{aligned} \tag{11}$$

where $\mathbf{M}$ is given by (1). In this formula, the first and second terms cancel out in LLR computation, leading to scoring function

$$\begin{aligned} Q_{\mathrm{x}}(\mathcal{X}|\mathbf{s}, \mathbf{x}) \quad = \quad & \mathbf{M}^* \boldsymbol{\Sigma}^{-1} \mathbf{F} \\ & + \frac{1}{2} \mathbf{M}^* \mathbf{N} \boldsymbol{\Sigma}^{-1} \mathbf{M}, \end{aligned} \tag{12}$$

hence

$$\mathrm{LLR}_{\mathrm{x}}(\mathcal{X}|\mathbf{s}) = Q_{\mathrm{x}}(\mathcal{X}|\mathbf{s}, \mathbf{x}_{\mathbf{s}}) - Q_{\mathrm{x}}(\mathcal{X}|\mathrm{UBM}, \mathbf{x}_{\mathrm{UBM}}), \tag{13}$$

where $\mathbf{x}_{\mathrm{UBM}}$ are channel factors estimated using UBM, and $\mathbf{x}_{\mathbf{s}}$ are channel factors estimated using speaker $\mathbf{s}$.

[2] All scores are normalized by frame length of the tested utterance, therefore the log-likelihood is average.

[3] For derivation of (6) and for exact definition of statistics $\mathbf{N}$, $\mathbf{F}$, $\mathbf{F}_{\mathbf{s}}$, $\mathbf{S}_c$, see equations (14) to (19) in [2].

2.4. UBM Channel Point Estimate

In [3], the authors assumed, that the shift of the model caused by the channel is identical both to the target model and the UBM[4]. Therefore, the $\mathbf{x}$ factors for utterance $\mathcal{X}$ are estimated using the UBM and then used for scoring. Formally written:

$$\begin{aligned} \mathrm{LLR}_{\mathrm{LPT}}(\mathcal{X}|\mathbf{s}) &= Q_{\mathrm{x}}(\mathcal{X}|\mathbf{s}, \mathbf{x}_{\mathrm{UBM}}) \\ &\quad - Q_{\mathrm{x}}(\mathcal{X}|\mathrm{UBM}, \mathbf{x}_{\mathrm{UBM}}) \qquad (14) \end{aligned}$$

Note, that when computing the LLR, the $\mathbf{Ux}$ in the linear term of (11) will cancel out, leaving the compensation to the quadratic term of (11).

2.5. Linear Scoring

Let us keep the LPT assumption and let $\mathbf{m_c}$ be the channel compensated UBM:

$$\mathbf{m_c} = \mathbf{m} + \mathbf{c} \qquad (15)$$

Furthermore, let us assume, that we move the origin of supervector space to $\mathbf{m_c}$.

$$\bar{\mathbf{M}} = \mathbf{M} - \mathbf{m_c} \qquad (16)$$

$$\bar{\mathbf{F}} = \mathbf{F} - \mathbf{N}\mathbf{m_c} \qquad (17)$$

Eq. (12) can now be rewritten to

$$\begin{aligned} Q_{\mathrm{xmod}}(\mathcal{X}|\bar{\mathbf{M}}, \mathbf{x}) &= \bar{\mathbf{M}}^{*}\mathbf{\Sigma}^{-1}\bar{\mathbf{F}} \\ &\quad + \frac{1}{2}\bar{\mathbf{M}}^{*}\mathbf{N}\mathbf{\Sigma}^{-1}\bar{\mathbf{M}}. \qquad (18) \end{aligned}$$

When approximating (18) by the first order Taylor series (as a function of $\bar{\mathbf{M}}$), only the linear term is kept, leading to

$$Q_{\mathrm{lin}}(\mathcal{X}|\bar{\mathbf{M}}, \mathbf{x}) = \bar{\mathbf{M}}^{*}\mathbf{\Sigma}^{-1}\bar{\mathbf{F}}. \qquad (19)$$

Realizing, that the channel compensated UBM is now a vector of zeros, and substituting (19) to (14), the formula for computing the LLR simplifies to

$$\mathrm{LLR}_{\mathrm{lin}}(\mathcal{X}|\mathbf{s}, \mathbf{x}) = (\mathbf{Vy} + \mathbf{Dz})^{*}\mathbf{\Sigma}^{-1}(\mathbf{F} - \mathbf{Nm} - \mathbf{Nc}). \qquad (20)$$

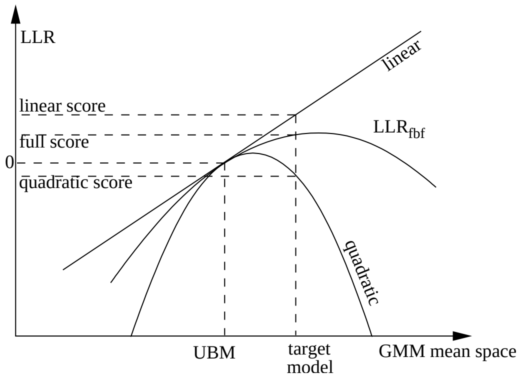

Fig. 1. An illustration of the scoring behavior for frame-by-frame, LPT, and linear scoring.

Given the fact, that the $\tilde{P}$-function is a lower bound approximation of the real frame-by-frame likelihood function, there are cases, when the LPT original function fails. Fig. 1 shows that the linear function can sometimes be a better approximation of the full LLR.

3. EXPERIMENTAL SETUP

3.1. Test Set

The results of our experiments are reported on the Det1 and Det3 conditions of the NIST 2006 speaker recognition evaluation (SRE) dataset [9].

The real-time factor was measured on a special test set, where 49 speakers were tested against 50 utterances. The speaker models were taken from the t-norm cohort, while the test utterances were chosen from the original z-norm cohort, each having approximately 4 minutes, totally giving 105 minutes.

3.2. Feature Extraction

In our experiments, we used cepstral features, extracted using a 25 ms Hamming window. 19 mel frequency cepstral coefficients together with log energy are calculated every 10 ms. This 20-dimensional feature vector was subjected to feature warping [10] using a 3 s sliding window. Delta and double delta coefficients were then calculated using a 5 frames window giving a 60-dimensional feature vectors. These feature vectors were modeled using GMM and factor analysis was used to treat the problem of speaker and session variability.

Segmentation was based on the BUT Hungarian phoneme recognizer [11] and relative average energy thresholding. Also short segments were pruned out, after which the speech segments were merged together.

3.3. JFA Training

We used gender independent Universal Background Models, which contain 2048 Gaussians. This UBM was trained using LDC releases of Switchboard II, Phases 2 and 3; switchboard Cellular, Parts 1 and 2 and NIST 2004-2005 SRE. The (gender independent) factor analysis models were trained on the same quantities of data as the UBM.

Our JFA is composed by 300 speaker factors, 100 channel factors, and diagonal matrix $\mathbf{D}$. While $\mathbf{U}$ was trained on the NIST data olny, $\mathbf{D}$ and $\mathbf{V}$ were trained on two disjoint sets comprising NIST and Switchboard data.

3.4. Normalization

All scores, as presented in the previous sections, were normalized by the number of frames in the test utterance. In case of normalizing the scores (zt-norm), we worked in the gender dependent fashion. We used 220 female, and 148 male speakers for t-norm, and 200 female, 159 male speakers for z-norm. These segments were a subset of the JFA training data set.

3.5. Hardware and Software

The frame-by-frame scoring was implemented in C++ code, which calls ATLAS functions for math operations. Matlab was used for the rest of the computations. Even though C++ produces more optimized code, the most CPU demanding computations are performed via the tuned math libraries that both Matlab and C++ use. This fact is important for measuring the real-time factor. The machine on which the real-time factor (RTF) was measured was a Dual-Core AMD Opteron 2220 with cache size 1024 KB. For the rest of the experiments, computing cluster was used.

4. RESULTS

Table 1 shows the results without any score normalization. The reason for the loss of performance in the case of LPT scoring could possibly be due to bad approximation of the likelihood function around

[4] The authors identified themselves under abbreviation LPT, therefore we will refer to this approach as to LPT assumption

UBM, ,i.e., the inability to adapt the model to the test utterance (in the $\mathbf{U}$ space only). Fig. 1 shows this case.

Table 1. *Comparison of different scoring techniques in terms of EER and DCF. No score normalization was performed here.*

	Det1		Det3	
	EER	DCF	EER	DCF
Frame-by-Frame	**4.70**	**2.24**	**3.62**	**1.76**
Integration	5.36	2.46	4.17	1.95
Point estimate	5.25	2.46	4.17	1.96
Point estimate LPT	16.70	6.84	15.05	6.52
Linear	5.53	2.97	3.94	2.35

Table 2 shows the results after application of zt-norming. While the frame-by-frame scoring outperformed all the fast scorings in the un-normalized case, normalization is essential for the other methods.

Table 2. *Comparison of different scoring techniques in terms of EER and DCF. zt-norm was used as score normalization.*

	Det1		Det3	
	EER	DCF	EER	DCF
Frame-by-Frame	2.96	1.50	1.80	0.91
Integration	2.90	1.48	1.78	0.91
Point estimate	**2.90**	**1.47**	1.83	**0.89**
Point estimate LPT	3.98	2.01	2.70	1.36
Linear	2.99	1.48	**1.73**	0.95

4.1. Speed

The aim of this experiment was to show the approximate real time factor of each of the systems. The time measured included reading necessary data connected with the test utterance (features, statistics), estimating the channel shifts, and computing the likelihood ratio. Any other time, such as reading of hyper-parameters, models, etc. was not comprised in the result. Each measuring was repeated 5 times and averaged. Table 3 shows the real time of each algorithm. Surprisingly, the integration LLR is faster then the point estimate.

Table 3. *Real time factor for different systems*

	Time [s]	RTF
Frame-by-Frame	1010	$1.60e^{-1}$
Integration	50	$7.93e^{-3}$
Point estimate	160	$2.54e^{-2}$
Point estimate LPT	36	$5.71e^{-3}$
Linear	**13**	$2.07e^{-3}$

This is due to implementation, where the channel compensation term in the integration formula is computed once per an utterance, while in the point estimate case, each model needs to be compensated for each trial utterance.

5. CONCLUSIONS

We have showed a comparison of different scoring techniques that different sites have recently used in their evaluations. While, in most cases, the performance does not change dramatically, the speed of evaluation is the major difference. The fastest scoring method is the Linear scoring. It can be implemented by a simple dot product, allowing for fast scoring of huge problems (e.g., z-, t- norming).

6. ACKNOWLEDGMENTS

This research was conducted under the auspices of the 2008 Johns Hopkins University Summer Workshop, and partially supported by NSF Grant No IIS-0705708 and by a gift from Google Inc. It was also partly supported by European projects AMIDA (FP6-033812) and MOBIO (FP7-214324), by Grant Agency of Czech Republic under project No. 102/08/0707, by Czech Ministry of Education under project No. MSM0021630528 and by Czech Ministry of Defense. The hardware used in this work was partially provided by CESNET under project No. 201/2006. Lukáš Burget was partly supported by Grant Agency of Czech Republic under project No. GP102/06/383.

7. REFERENCES

[1] Robert B. Dunn Douglas A. Reynolds, Thomas F. Quatieri, "Speaker verification using adapted gaussian mixture models," *Digital Signal Processing*, pp. 19–41, January 2000.

[2] P. Kenny, G. Boulianne, P. Oullet, and P. Dumouchel, "Joint factor analysis versus eigenchannes in speaker recognition," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 15, no. 7, pp. 2072–2084, 2007.

[3] C. Vair, D. Colibro, F. Castaldo, E. Dalmasso, and P. Laface, "Loquendo - politecnico di torino's 2006 nist speaker recognition evaluation system," in *Proceedings of Interspeech 2007*, 2007, pp. 1238–1241.

[4] Niko Brümmer, Lukáš Burget, Jan Černocký, Ondřej Glembek, František Grézl, Martin Karafiát, David Leeuwen van, Pavel Matějka, Petr Schwarz, and Albert Strasheim, "Fusion of heterogeneous speaker recognition systems in the stbu submission for the NIST speaker recognition evaluation 2006," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 15, no. 7, pp. 2072–2084, 2007.

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

[6] P. Kenny, "Joint factor analysis of speaker and session variability : Theory and algorithms - technical report CRIM-06/08-13. Montreal, CRIM, 2005," 2005.

[7] P. Kenny and P. Dumouchel, "Experiments in speaker verification using factor analysis likelihood ratios," in *Proceedings of Odyssey 2004*, 2004.

[8] P. Kenny, G. Boulianne, P. Ouellet, and P. Dumouchel, "Factor analysis simplified," in *Proc. of the International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, Toulouse, France, March 2005, pp. 637– 640.

[9] "National institute of standard and technology," http://www.nist.gov/speech/tests/spk/index.htm.

[10] S. Sridharan J. Pelecanos, "Feature warping for robust speaker verification," in *Proceedings of Odyssey 2006: The Speaker and Language Recognition Workshop*, 2006, pp. 213–218.

[11] P. Schwarz, P. Matějka, and J. Černocký, "Hierarchical structures of neural networks for phoneme recognition," in *Proc. of the International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, Toulouse, France, May 2006, pp. 325–328.