WEIGHT BASED SUPER-GMM FOR SPEAKER IDENTIFICATION SYSTEMS

Guillermo Garcia, Thomas Eriksson

Communication Systems Group, Department of Signals and Systems,
Chalmers University of Technology, 412 96 Göteborg Sweden.
phone: + (46) 7721821, fax: + (46) 7721748
email: garciap@s2.chalmers.se, thomase@chalmers.se

ABSTRACT

Gaussian Mixture Models (GMMs) are widely employed as statistical models in biometric systems. In speaker identification (SID) systems, GMMs have shown their effectiveness for modeling speaker identities. However, an increase on the number of enrolled speakers reduces the interspeaker variability causing the degradation on the performance of the recognizer. In this work, we propose a speaker super-GMM which deals with the interspeaker distance by access to a larger number of GMMs but maintaining the same complexity as the baseline system. The super-GMM is constructed by the concatenation of all the speaker GMMs enrolled in the database and weighting each GMM component. These weights contain discriminative information used to determine each speaker model. To train the super-GMM, we train the weights of each mixture component using a variation of the expectation maximization (EM) algorithm which only updates the weights of the super-GMMs. Then, we apply a minimum classification error (MCE) approach to enhance the discriminative properties of the weights. Our approach has shown approximately 20% improvement on the performance (probability of error) compared to the baseline system.

Index Terms: Discriminative methods, speaker recognition, Gaussian distributions, modeling, estimation.

1. INTRODUCTION

The development of technology has opened new perspectives for the use of biometrics systems. Applications are countless in daily life, from a personal *ID-number* to forensic analyses [1]. Among the biometric systems, we will concentrate on speaker recognition systems since they are easy to deploy and are less invasive compared to other biometric authentication techniques. Speaker recognition can be defined as the process of automatically recognizing who is speaking based on the statistical information provided by speech signals. The main technique is to find a set of features that best represents a specific speaker voice. Speaker recognition systems can be divided depending on their tasks in speaker identification (SID) and speaker verification (SV). In this work, we will focus on SID systems used to determine from a set of predefined models to which of them belongs an input test utterance from an unknown speaker.

The speaker recognition process is divided in two phases independently of the task: enrollment and classification. In the enrollment phase, the expectation maximization (EM) algorithm [2] is used to estimate Gaussian Mixture Models (GMMs) for each speaker. The EM algorithm provides maximum-likelihood (ML) estimates for the unknown model parameter from a training database. In the classification phase, we compute the likelihood of test speech samples from an unknown voice belonging to a certain speaker GMM. One of the main research areas in speaker recognition is the speaker modeling wherein vector quantization (VQ) models [3], GMMs [2] and support vector machines (SVM) [4] are usually used. This research aims at finding more discriminative training methods and speakers models for closed-set speaker recognition systems with limited training data. Closed-set testing refers to that the unknown utterance/voice comes from a fixed set of known speakers.

Traditionally in speaker recognition systems, the speakers GMMs are trained using features extracted from the speech trying to represent the whole space of features belonging to that specific speaker such that even unseen features can be correctly classified. Moreover, these speaker models must be able to handle intraspeaker and interspeaker variations of the speech features [5]. Examples of GMM modeling techniques for speaker recognition systems that tackle these variations are the “flat” universal models [6, 7]. However, SID systems will always face a tradeoff between performance and the number of speakers in the system. In general as the number of speakers enrolled in the system increases, the separability (interspeaker distance) between speaker models decreases, thereby causing a degradation on the identification/ recognition performance.

In this paper, we design a super-GMM which accesses a large number of GMMs but maintains the same complexity as the baseline system. This approach consists of a super-GMM with common mixture components for all the speakers enrolled in the database, but different discriminative weights for each mixture component depending on the speaker. The speaker super-GMM uses the information available from the corresponding speaker and all the speakers enrolled in the database to improve the performance of the recognizer regarding that the complexity of the classification phase remains the similar. Moreover, we develop a method to train the discriminative weights of this super-GMM to increase the interspeaker variability in SID systems.

The rest of the paper is organized as follows: section 2 presents the baseline of the SID systems, section 3 defines the super-GMM and describes the way to construct the super-GMM, section 4 describes the training of the super-GMM, section 5 presents the experimental evaluation and the results. Section 6 shows the conclusions of this work.

2. BASELINE SYSTEM

2.1 Design Phase

In many text-independent speaker identification systems, GMMs are used as a statistical model of each speaker. The

GMM for the s-th speaker is defined as

$$p\left(x_t|\lambda^{(s)}\right) \quad = \quad \sum_{k=1}^{K} w_k^{(s)} \mathcal{N}\left(x_t, \mu_k^{(s)}, \mathbf{C}_k^{(s)}\right), \qquad (1)$$

i.e., $\mathcal{N}(x_t, \mu_k^{(s)}, \mathbf{C}_k^{(s)})$ is a weighted sum of Gaussian distributions, where $\mu_k^{(s)}$ is the mean and $\mathbf{C}_k^{(s)}$ is the covariance matrix of the k-th Gaussian distribution for speaker s. The GMM can also be defined by a set of parameters $\lambda^{(s)} = w_k^{(s)}, \mu_k^{(s)}, \mathbf{C}_k^{(s)}$. To determine these parameters, the EM algorithm can be used. The EM algorithm is an iterative algorithm that uses a training database to find ML estimates of the weights, means and covariances in the GMM (or at least approximations to the ML estimates). Each speaker GMM is a unique model and describes the particular features of his/her voice.

2.2 Classification Phase

The extracted features from an unknown speaker utterance $\{x_t\}_{t=1}^{T}$ are compared against the speaker GMM stored in the database. The speaker recognizer calculates a score for each speaker model in the database giving an estimate of the likelihood of the utterance belonging to a given speaker model

$$\mathcal{L}\left(x|\lambda^{(s)}\right) = \sum_{t=1}^{T} \log p\left(x_t|\lambda^{(s)}\right). \qquad (2)$$

Then, based on the score the speaker recognizer will decide to which model the utterance belong. The speaker with the highest log-likelihood is declared as the winner,

$$\hat{s} = \underset{1\leq s\leq S}{\arg\max}\, \mathcal{L}\left(x|\lambda^{(s)}\right). \qquad (3)$$

The goal of a speaker recognizer is to minimize the probability of error given by $\mathrm{P_e} = \Pr[s \neq \hat{s}]$ [8]. We will use it as a baseline system, against which the proposed schemes are compared.

The performance evaluation of speaker recognition systems is usually measured by the probability of error ($\mathrm{P_e}$) attained. To calculate this probability of error, we run the baseline system (i.e., enrollment and classification phases) N number of times using independent sets of speech files and then computing the average number of errors occurred.

3. SUPER-GMM

A speaker super-GMM can be defined as a high order GMM which have access to greater knowledge of the speakers enrolled in the database,

$$F(x_t|\lambda^{(s)}) = \sum_{k=1}^{M} \tilde{w}_k^{(s)} \mathcal{N}\left(x_t, \tilde{\mu}_k, \tilde{\mathbf{C}}_k\right), \qquad (4)$$

i.e., $\mathcal{N}\left(x_t, \tilde{\mu}_k^{(s)}, \tilde{\mathbf{C}}_k^{(s)}\right)$ is a weighted sum of Gaussian distributions where $\tilde{w}_k^{(s)}$ is the weight, $\tilde{\mu}_k$ is the mean and $\tilde{\mathbf{C}}_k$ is the covariance matrix for the k-th mixture components. We must denote that a super-GMM sets $\mu_k^{(s)} = \tilde{\mu}_k$ and $\mathbf{C}_k^{(s)} = \tilde{\mathbf{C}}_k$ for all the speakers ($s = 1, 2, ..., S$), yielding the mixture component weight $\left(\tilde{w}_k^{(s)}\right)$ as the only way to discriminate between the speaker models.

Moreover, the weights of the new super-GMM must fulfill

$$\sum_{k=1}^{M} \tilde{w}_k^{(s)} = 1, \qquad (5)$$

The purpose of creating a speaker super-GMM is to handle the interspeaker variabilities in SID systems without increasing the complexity of the classification phase usually performed in real time.

3.1 Creation of the Super-GMM

To create a super-GMM based on the closed-set speakers GMMs, we assume that the speakers GMMs are known and we can concatenate the GMMs of all the speakers in the database such that each speaker model contains the same parameters for each mixture component (i.e., means and variances).

Concatenating the means of all the speakers models enrolled in the system, we attain that the means of the super-GMM are defined as

$$\tilde{\mu} = \left\{\mu_k^{(s)}\right\}_{\forall k, \forall s}. \qquad (6)$$

Following the same concatenation procedure, we attain that the covariance matrix for the super-GMM is defined as

$$\tilde{\mathbf{C}} = \left\{\mathbf{C}_k^{(s)}\right\}_{\forall k, \forall s}. \qquad (7)$$

In this work, we will use speakers GMMs with diagonal covariance matrices, i.e.,

$$\mathbf{C}_k^{(s)} = \mathrm{diag}\left(\sigma_{k,1}^{2(s)}, \ldots, \sigma_{k,D}^{2(s)}\right), \qquad (8)$$

where $\sigma_{k,d}^{2(s)}$ ($d = 1, \ldots, D$) is the variance for speaker s, k-th Gaussian distribution, and d-th dimension [2].

The weights of the new super-GMM should be rearranged to fulfill (5), and will contain M values which is the total number of GMM components after the concatenation of the speakers GMMs. In the next sections, we will present the way to attain the speaker models weights.

3.2 Complexity Comparison

The complexity involved in the classification phase (i.e., the computation of log-likelihoods) is mainly due to the calculation of exponentials functions. Moreover, the number of operations is proportional to the number of speakers enrolled in the database (S), the size of the test utterance (T), and the total number of mixture components for the baseline (K) or the super-GMM (M). The main characteristic of the super-GMM is that it contains the same number of exponential operations as the baseline system.

Defining the complexity ratio as the ratio between the number of operations of the super-GMM and the baseline, we attain

$$\phi = \frac{\phi_{com} + \Delta\phi_{super}}{\phi_{com} + \Delta\phi_{baseline}}, \qquad (9)$$

where ϕ_{com} is a common number of operations for both systems including the computation of the exponential functions, $\Delta\phi_{super} = 2\mathrm{MST} = 2\mathrm{KS}^2\mathrm{T}$ is the remained number of operations proper for the super-GMM and $\Delta\phi_{baseline} = 2\mathrm{KST}$ is the remained number of operations proper for the baseline system. The average complexity ratio for our test utterance set is approximately 1.4, this increase on the complexity ratio is due to the operations (multiplications and sums) with the discriminative weights for the super-GMM case.

4. WEIGHTS TRAINING

A main requirement for a sum of Gaussian distributions to be considered a GMM is to fulfill (5). On the next sections, we will present a systematic approach to obtain the mixture component weight for each speaker model.

4.1 EM Training of Weights

After the creation of the super-GMM, we could retrain each speaker model using the EM algorithm. The reason for this retraining is to exploit the information acquired from knowing all the mixture components of all the speakers GMMs in the set. We used a variant of the EM algorithm for retraining the mixture component weights and the training data of the correspondent s-th speaker .

The algorithm is shown in Table 1, holding as inputs the super-GMM and the training database for each speaker. Our initialization procedure consists on assigning 50% of the probability to the mixture components corresponding to the s-th real speaker and 50% to the mixtures components of all the other speakers. The EM algorithm will run until convergence on the log-likelihood is achieved. For simplicity, we drop the s corresponding to the speaker since the algorithm is applied similarly to all the speakers.

```
1. Using the training database $\{x_t\}_{t=1}^{T}$.
      Compute the total likelihood $LL_t$
      $LL_t = \sum_{k=1}^{M} \tilde{w}_k \mathcal{N}\left(x_t, \tilde{\mu}_k, \tilde{\mathbf{C}}_k\right), \ \{t = 1...T\}$.
2. Normalize the likelihood.
      Compute $\eta_{k,t} = \frac{\tilde{w}_k \mathcal{N}\left(x_t, \tilde{\mu}_k, \tilde{\mathbf{C}}_k\right)}{LL_t}$.
      $\{k = 1...M\}; \ \{t = 1...T\}$.
3.   Compute the sum of weights, $\hat{w}_k$
      $\hat{w}_k = \sum_{t=1}^{T} \eta_{k,t}, \ \{k = 1, 2, ...M\}$.
4. Compute the new parameter values
    for each GMM component.
    $\tilde{w}_k = \frac{\hat{w}_k}{T}$.
```

Table 1: Variation of EM Algorithm.

This training approach still fulfills the requirements of GMMs presented in (5).

4.2 Discriminative Training

In the previous section the ML estimate was used to optimize the weights of the speaker GMMs. This approach only considers the data from the s-th speaker to model the speaker.

In a discriminative training approach MCE [9–11], we create the speakers models considering the competing speakers and a training criterion used to directly minimize the errors of the training data.

The simplest form of misclassification measure is the Bayes classifier defined for a two class problem as

$$d(x) = \Pr\left[\Lambda_1|\mathbf{x}\right] - \Pr\left[\Lambda_2|\mathbf{x}\right], \tag{10}$$

where $\Pr\left[\Lambda_1|\mathbf{x}\right]$ and $\Pr\left[\Lambda_2|\mathbf{x}\right]$ are the posterior probabilities of belonging to class Λ_1 or Λ_2 and assumed known. This classifier will rely on the difference between the classes to emit a decision having a boundary when $d(x) = 0$.

For SID systems, a misclassification measure using the log-likelihoods of a group of tests can be defined. Assuming that a stochastic model (GMM) and a group of tests[1] are available for each speaker, we can compute the log-likelihood $\mathcal{L}\left(x|\lambda^{(s)}\right)$ of each test with respect to the actual GMMs or the super-GMM, such that the number of log-likelihood values obtained for each test is equal to the total number of speakers.

For analysis purposes, we define a real speaker as the speaker to which a predetermined test belongs i.e., the correct identified speaker.

Letting $\{v^{(i)}(n)\}_{n=1}^{N}$ be a set of N log-likelihood values for the i-th real speaker, and $\{z^{(i)}(n)\}_{n=1}^{N}$ be a set of the difference between the log-likelihood of the i-th real speaker and the log-likelihood of the maximum of other speakers,

$$\begin{aligned} z^{(i)}(n) = v^{(i)}(n) - \max(&v^{(1)}(n), v^{(2)}(n) \\ &.., v^{(i-1)}(n), v^{(i+1)}(n), ..v^{(S)}(n)). \end{aligned} \tag{11}$$

Defining $\{y^{(i)}(n)\}_{n=1}^{N}$ as the set of the N log-likelihood values of the maximum of other speakers i.e.,

$$y^{(i)}(n) = \max(v^{(1)}(n), v^{(2)}(n).., v^{(i-1)}(n), v^{(i+1)}(n), ..v^{(S)}(n)). \tag{12}$$

Substituting $y^{(i)}(n)$ in (11), we attain

$$z^{(i)}(n) = v^{(i)}(n) - y^{(i)}(n), \tag{13}$$

which is similar to the simple Bayes classifier mentioned above.

After defining a misclassification measure, we require to define a cost function associated to an error occurrence. In general, a cost function is defined as

$$\Gamma(x, \lambda) = \Gamma(d(x)), \tag{14}$$

which is a function of the misclassification measure.

For SID systems, the common cost function is the probability of error for the i-th speaker, defined as

[1] one test is a set of samples from an unknown speaker $\{x_t\}_{t=1}^{T}$. Each test corresponds to a short spoken sentence (1-2 seconds) by the speaker.

$$\begin{aligned} \mathrm{P_e}^{(i)} &= \Phi\left(v^{(i)}(n) - y^{(i)}(n)\right), & (15) \\ &= \Phi\left(z^{(i)}(n)\right), & (16) \end{aligned}$$

where Φ is the unit step function which detects when $(v^{(i)}(n) - y^{(i)}(n)) \leq 0$. The cost function will assign a value (cost) when $v^{(i)}(n)$ is smaller than $y^{(i)}(n)$ (i.e., an error occurred). However, the derivative of this function is not defined in all points making difficult any optimization procedure. For this reason, we use a sigmoid translated function as a cost function defined as

$$\Upsilon^{(i)} = \frac{1}{1 + \exp\left(-z^{(i)}(n)\right)}, \tag{17}$$

which is derivable in all points.

4.3 Parameter Adjustment using Descendent Methods

After obtaining the cost function and the measure of misclassification, we would perform an optimization with respect to the parameters of the GMM specifically the weights in order to find the most discriminative weights for each mixture component.

First, we define a vector containing the weights of the i-th speaker super-GMM component $\tilde{\mathbf{w}}^{(i)} = [\tilde{w}_1^{(i)} \tilde{w}_2^{(i)} \ldots \tilde{w}_M^{(i)}]$.

Using a GPDM method (e.g., the Gauss Newton method [12]), we can find the optimal weights.

$$\tilde{\mathbf{w}}^{(i)}(r+1) = \tilde{\mathbf{w}}^{(i)}(r) + \varepsilon \mathbf{H}^{\dagger(i)}(\tilde{\mathbf{w}})\Upsilon^{(i)}, \tag{18}$$

where ε is the step size, r is the iteration time and $\mathbf{H}^{\dagger(i)}(\tilde{\mathbf{w}})$ is defined as

$$\mathbf{H}^{\dagger(i)}(\tilde{\mathbf{w}}^{(i)}) = \left(\mathbf{H}^{T(i)}(\tilde{\mathbf{w}}^{(i)})\mathbf{H}^{(i)}(\tilde{\mathbf{w}}^{(i)})\right)^{-1} \mathbf{H}^{T(i)}(\tilde{\mathbf{w}}^{(i)}), \tag{19}$$

i.e., $\mathbf{H}^{(i)}(\tilde{\mathbf{w}}^{(i)})$ is the gradient of the probability of error with respect to the weighting vector of each speaker model.

$$\left[\mathbf{H}^{(i)}(\tilde{\mathbf{w}}^{(i)})\right]_k = \frac{\partial \Upsilon^{(i)}}{\partial z^{(i)}} \frac{\partial z^{(i)}}{\partial v^{(i)}} \frac{\partial v^{(i)}}{\partial \tilde{w}_k^{(i)}}. \tag{20}$$

In order to be a GMM, the weighting vector $\mathbf{w}^{(i)}$ must fulfill (5).

5. EXPERIMENTAL EVALUATION

5.1 Database Description

The experiments were conducted using the 137 speakers of the YOHO database [13], 2 independent databases were created: training and evaluation. The training database consists of the first two folders in the enrollment session (48 files for each speaker). The evaluation database consists of the verify session obtaining 40 tests for each speaker. Each speech file, after removing silence at the beginning and end, was segmented into frames of 25 ms length with an overlap of 10 ms. Each frame was pre-emphasized and Hamming windowed. Then 12-th (truncated from 23-th) dimensional MFCCs were created. The MFCCs obtained from the training database are used to train 16 and 32-mixture GMMs. For the discriminative training, we use the same training database described above (i.e., 48 files for each speaker).

5.2 Experiments Implementation

To create the speaker super GMM, we concatenate the 16 and 32-mixture GMMs of the 137 speakers, respectively. Then, we retrain the weights of the super-GMM using the training database and the weighting EM algorithm.

Finally, to train the discriminative speakers super-GMMs, we apply the Gauss Newton algorithm defined in the previous section to the weights obtained from the EM estimation. The algorithm searches sequentially for the weights with better discriminative properties for each speaker, beginning with the first speaker until the 137-th speaker. The algorithm is repeated until convergence is achieved.

5.3 Experimental Results

From the results obtained, we observe that we have an improvement in the performance compared to the baseline. Table 2 shows the probability of error for the different approaches. The second column shows the probability of error for the baseline system. On the third column, we show the probability of error using the weighting EM algorithm. Finally, the fourth column shows the probability of error using jointly the discriminative approach and the weighted EM approach. We observe that the best performance is achieved with the joint use of the discriminative approach and the weighted EM since this method considers the information from the i-th real speaker and other speaker models to create the speaker models, attaining a maximum of 20% compared to the baseline system.

Figure 1 shows an example of the super-GMM component weights after the retraining with the EM algorithm. We can observe large component weights defining the speaker model. As mentioned before, this type of retraining can only provide maximum likelihood estimates and non discriminative training.

Figure 2 shows an example of the weighting after the discriminative training . We can observe that the larger weights ($\tilde{w}_k$) correspond to the mixtures defining the speaker and the smaller component contains the discriminative properties. Figure 3 shows the weights for the super-GMM mixture components of all the 137 speakers enrolled in the database. We can observe the difference between the discriminative components and the components defining the speaker.

Table 2: *Performance results of the super-GMM for a SID system.*

No Mixtures	$\mathrm{P_e}$ Baseline	$\mathrm{P_e}$ weighting EM	$\mathrm{P_e}$ Discriminative & EM
16	.0566	0.0504	0.0465
32	.0347	0.0325	0.0308

6. CONCLUSIONS

Being able to train a speaker GMM which tackles the inter and intraspeaker variabilities is of great importance in SID

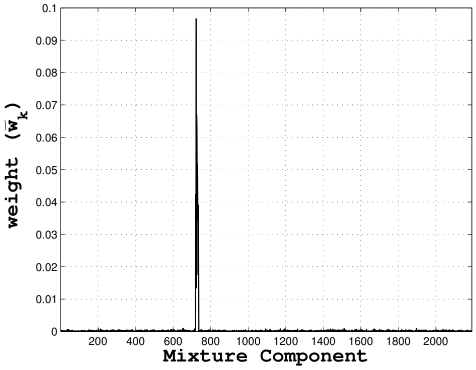

Figure 1: Example of the weighting for the GMM mixture component after the EM algorithm.

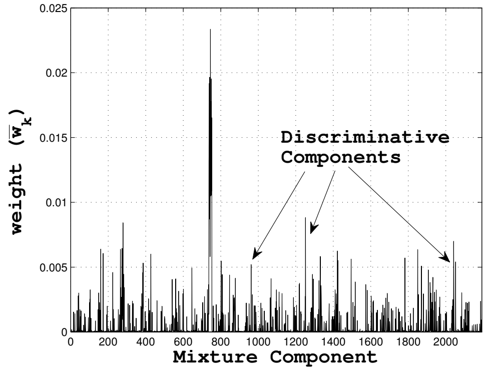

Figure 2: Example of the weighting for the GMM mixture component after the discriminative approach.

systems. In this paper, we show a new speaker super-GMM which achieves better performance compared to the baseline system, while the complexity in the classification phase is similar to the baseline. Moreover, this is an ongoing research since the discriminative training is done only on the weights of the mixture components, the means and the variances of the super-GMM could be retrained to enhance the discriminative properties of the super-GMM.

REFERENCES

[1] P. Rose, *Forensic Speaker Identification*. Taylor & Francis, 2002.

[2] D. Reynolds and R. Rose, “Robust text-independent speaker identification using Gaussian mixture speaker models,” *IEEE Transactions Speech and Audio Processing*, vol. 3, pp. 72–83, 1995.

[3] F. K. Soong, A. E. Rosenberg, L. R. Rabiner, and B. H. Juang, “A vector quantizer approach to speaker recognition,” in *Proceedings ICASSP*, vol. 1, 1985, pp. 387–390.

[4] V. Wan and W. Campbell, “Support vector machines for speaker verification and identification,” in *Proceedings NNSP*, 2000, pp. 775–784.

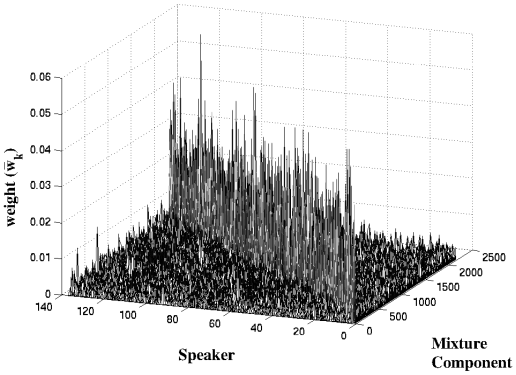

Figure 3: Weights of the super-GMM mixture components for a set of speakers.

[5] J. Campbell, “Speaker recognition: a tutorial,” in *IEEE Proceedings*, vol. 85, 1997, pp. 1437–1462.

[6] D. Reynolds, T. F. Quatieri, and R. B. Dunn, “Speaker verification using adapted Gaussian mixture models,” *Digital Signal Processing*, vol. 10, 2000.

[7] R. Zheng, S. Zhang, and B. Xu, “Text-independent speaker identification using gmm-ubm and frame level likelihood normalization,” in *Proceedings ISCSLP*, vol. 4, 2004, pp. 289–292.

[8] T. Eriksson, S. Kim, H.-G. Kang, and C. Lee, “An information-theoretic perspective on feature selection in speaker recognition,” *IEEE Signal Processing Letters*, vol. 12, no. 7, pp. 500 – 503, 2005.

[9] C. M. d. Alamo, F. C. Gil, C. d. l. T. Munilla, and L. H. Gomez, “Discriminative training of GMM for speaker recognition,” in *Proceedings ICASSP*, vol. 1, 1996, pp. 89–92.

[10] B.-H. Juang and S. Katagiri, “Discriminative learning for minimum error classification,” *IEEE Transactions Signal Processing*, vol. 40, pp. 3043–3054, 1992.

[11] O. Siohan, A. E. Rosenberg, and S. Parthasarathy, “Speaker Identification using Minimum Classification Error Training,” in *Proceedings ICASSP*, vol. 1, 1998, pp. 109–112.

[12] S. M. Kay, *Fundamentals of Statistical Signal Processing, Estimation Theory*, 2nd ed., ser. Signal Processing. Prentice Hall, 1993.

[13] J. Campbell, “Testing with YOHO CD-ROM voice verification corpus,” in *Proceedings ICASSP*, 1995, pp. 341–344.