DISCRIMINANT APPROACHES FOR GMM BASED SPEAKER DETECTION SYSTEMS

Alexandre Preti, Nicolas Scheffer, Jean-François Bonastre

LIA, Université d'Avignon
Agroparc, BP 1228
84911 Avignon CEDEX 9, France
{alexandre.preti,nicolas.scheffer,jean-francois.bonastre}@univ-avignon.fr

ABSTRACT

This paper presents some experiments on discriminative training for GMM/UBM based speaker recognition systems. We propose two MMIE adaptation methods for GMM component weights suitable for speaker recognition. The impact on performance of this training methods is compared to the standard weight estimation/adaptation criterion, MLE and MAP on standard GMM based systems and on SVM based systems. The results enforce the difficulty to introduce discriminative behaviour in a GMM based system whereas it is inherent in SVM based systems.

1. INTRODUCTION

Gaussian Mixture Models (GMM) based systems for speaker recognition have shown robust results for several years and are widely used in speaker recognition applications. In fact, they give quite good performance when representing cepstral coefficients to model speaker identity and are effective with quite small amount of training data to enrol a client [1] [4] .
This technological choice is confirmed by the results of the NIST annual Speaker Recognition Evaluation. Support Vector Machine (SVM) systems are known to bring a discriminative behavior contrary to classical GMM systems. Indeed, generative methods do not introduce discriminant behavior even if the speaker models are derived from a generic model, the Universal Background Model (UBM) which represents a standard impostor. As a result, since several years, the interest in using a discriminative classifier like SVMs has grown in the community. Performance of the GLDS kernel [2] is now comparable to the generative methods.

The work presented here intends to combine GMM modeling and discriminative approaches on a speaker recognition task. One popular discriminative criterion for generative model is called Maximum Mutual Information Estimation (MMIE). To introduce discriminative training using MMIE, our approach consists in using information from Gaussian weights. Extending the work done in speech recognition [5, 6] to speaker recognition, we could expect a performance improvement by training discriminative weights using information from the impostors models or from the world model. Each Gaussian weight of the target model is compared with all components of the world model to determine the mutual information. The target weights are modified depending on the corresponding mutual information. To assess the behavior of this discriminative training, different weight adaptation methods are compared.
The main goal of this work is to understand the mechanism behind the SVM based system and see if it can be applied in a classical GMM/UBM based system.

This paper is organized as follows: the different tools, the baseline UBM/GMM system as well as the experimental protocol are presented in section 2. The two next sections 3 and 4 describe the estimation of Gaussian weights with both MLE and MMIE criteria. The different systems used for experiments, are presented in section 5. They are based both on GMM and SVM approaches. Results in section 6 aim at comparing systems performance with parameters estimated from a generative criterion MLE, and a discriminative criterion MMIE. For each of these experiments, performance using GMM system only and a SVM based system are compared. The section 7 is dedicated to fusion experiments using one of the discriminant-based system and the GMM/UBM baseline system. Section 8 finally concludes this work.

2. TOOLS AND PROTOCOL

2.1. Database

All experiments presented in section 5 are performed based upon the NIST 2005 database, common evaluation trials, male speakers only. This condition consists of 274 speakers. Train and test utterances contain 2.5 minutes of speech in average (telephone conversation). The whole speaker detection experiment consists in 13624 tests (951 target tests). Each

test is made independently and the use of information from other tests to take a decision on the current test is forbidden. The performance are evaluated throught classical DET performance curves.

2.2. Baseline speaker recognition system

The LIA_SpkDet system [7] developed at the LIA laboratory is used as baseline in this paper. Built from the ALIZE platform [8][9], it was evaluated during the NIST SRE'04 and SRE'05 campaigns, where it obtained about the best performance for a cepstral GMM-UBM system. Both the LIA_SpkDet system and the ALIZE platform are distributed under an open source licence.
The LIA_SpkDet system is based on classical UBM-GMM approach and T-Norm approach for likelihood score normalization. The background model used for the experiments is the same as the background model used by the LIA for the NIST SRE 2005 campaign (male only). The training is performed based on NIST SRE 1999 and 2002 databases, and consists in about 1 million of speech frames. For the front-end processing, the signal is characterized by 32 coefficients including 16 linear frequency cepstral coefficients (LFCC) (filter-bank analysis) and their first derivative coefficients extracted with SPRO [10]. A frame removal based on a three component GMM energy modeling is computed. A mean and variance normalization process is finally applied on coefficients. The world and target models contain 2048 components.

3. USING GAUSSIAN WEIGHTS WITH A GENERATIVE APPROACH

Different methods are used to estimate/adapt GMM Gaussian weights. The most known methods are the Maximum Likelihood Estimation (MLE) and the Maximum A Posteriori (MAP) criterion associated with the EM algorithm. In this paper, we also propose an implementation of the MMIE criterion for speaker recognition. In the following parts of this paper, adaptation of Gaussian weights only is computed.

3.1. Maximum Likelihood Estimation

In this approach, the model parameters are estimated by maximizing the likelihood of the training data for this model. The MLE objective function is given by:

$$P(X|M,\theta) = \prod_{i=1}^{n} P(x_i|M,\theta) \quad (1)$$

Since no close form solution exists to estimate the true parameter, the EM algorithm is used to estimate the weight of the model.

$$\hat{w_i} = \frac{w_i L(X|G_i)}{\sum_{j=1}^{nb_G} w_j L(X|G_j)} \quad (2)$$

Where w_i is the *a priori* weigth of the Gaussian i.

3.2. Maximum A Posteriori

Considering the Bayesian adaptation, the model parameters are obtain thanks to the EM algorithm by maximizing the likelihood *a posteriori* of the training data given the model [3]. One way to estimate these a posteriori weights for a mixture i of a client c is given by [4]:

$$\hat{c_{ic}} = \alpha_i c_{ic} + (1-\alpha_i) c_{iw} \quad (3)$$

α_i is defined as follows : $\alpha_i = \frac{n_i}{ni+r}$, where r is the *relevance* factor, $n_i = \sum_{t=1}^{T} P(i|x_t)$ and c_{iw} is the weight of the world model for the Gaussian i.

4. USING GAUSSIAN WEIGHTS WITH A DISCRIMINATIVE APPROACH

This section investigates the different systems within the use of a discriminative approach, begining with a description of the MMIE criterion, ending with the use of SVM based systems.

4.1. Maximum Mutual Information Estimation of Gaussian Weights

MMIE introduces the influence of impostor models in the estimation process of the target model parameters. It takes into account the mutual information between the model parameters (computed first by an EM iteration) and the parameters in all the impostor models. Indeed the influence of a parameter is reduced if the mutual information is maximum. Previous work on MMIE criterion for speech processing [5, 11, 12] has shown that it consists in maximizing the objective function (1) :

$$F_\lambda = \sum_{r=1}^{R} log(\frac{P_\lambda(O_r|M_{w_r}) * P(W_r)}{\sum_{\hat{W}} P_\lambda(O_r M_{\hat{w}}) * P(\hat{W})}) \quad (4)$$

where O_r represents the observation sequences, W_r the correct transcription, $\hat{W}$ the wrong transcriptions, M_{wr} and $M_{\hat{w}}$ the models associated with the correct and wrong transcriptions respectively.

Woodland and Poovey [5, 6] suggested a weight estimation rule for a particular state j based on the maximization of the following function :

$$F_W(\lambda) = \sum_{m=1}^{M} \left[\gamma_{jm}^{num} * log(\hat{c_{jm}}) - \frac{\gamma_{jm}^{den}}{c_{jm}} * \hat{c_{jm}} \right] \quad (5)$$

In [16] it has been demonstrated that maximizing this function can be performed by optimizing each term of the sum, which is a convex function, resulting in:

$$\hat{c_{jm}} = ArgMax_c \left(\gamma_{jm}^{num} * log(c) - \frac{\gamma_{jm}^{den}}{c_{jm}} * c \right) \quad (6)$$

and

$$\frac{\gamma_{jm}^{num}}{\gamma_{jm}^{den}} = \frac{\sum_{X\in\omega^k} \frac{L(X|G_{jm})}{L(X|S_j)+\sum_{i\neq k} L(X|S_i)}}{\sum_l \sum_{X\in\omega^l} \frac{L(X|G_{lm})}{L(X|S_l)+\sum_{i\neq l} L(X|S_i)}} \quad (7)$$

However, an approximation of this equation is proposed in [16]

$$\hat{c_{jm}} = c_{jm} * \frac{c_{jm}}{\sum_k c_{km}} \quad (8)$$

To adapt these results for speaker recognition we have to consider the observation sequence as the target features, the model corresponding to the correct transcription as the target model (ML trained), the wrong transcriptions as the impostor features taken from the world features and the model corresponding to the wrong transcriptions as the world model. In the approach proposed here, we choose to use the world model to represent the wrong transcriptions. It is justified by the fact that the world model represents a mean impostor model. We use it to discriminate from a single model instead of a cohort of models like it is commonly done in speech recognition (discrimination from a cohort of impostor models could be investigated in future work). A percentage of the world features is taken to represent the impostor examples used in the discriminative function. Then we obtain :

$$DL = \frac{L(X|G_{jc})}{L(X|M_c) + L(X|M_w)} \quad (9)$$

$$\frac{\gamma_{jm}^{num}}{\gamma_{jm}^{den}} = \frac{\sum_{X\in\Omega^c} DL}{\sum_{X\in\Omega^w} DL + \sum_{X\in\Omega^c} DL} \quad (10)$$

where G_{jc} refers to Gaussian j of the client model c, M_c the client model, M_w the world model, Ω^w the impostor *corpus*, Ω^c the client corpus. This adaptation function will be further named as MMIE.
The approximated function (8) can also be adapted to speaker recognition :

$$\hat{c_{jc}} = c_{jc} * \frac{c_{jc}}{c_{jw} + c_{jc}} \quad (11)$$

This approximation will be further named as *MMIE approx.* [1]

4.2. SVM classification using GMM weights

Support Vector machine classification

Support Vector Machines are described by Vapnik [13] and are usually used as a binary classifier in speaker verification (target/non-target). To answer a linearly separable problem, the SVM gives the optimal hyperplane that maximizes the margin between the two classes, among the several possible hyperplane.
Let the separating hyperplane be defined by $x\dot{w}+b = 0$ where w is the normal to the hyperplane. For linearly separable data labelled $\{x_i, y_i\}$, $x_i \in \Re_d^N, y_i \in \{-1, 1\}, i = 1 \ldots N$, the optimum boundary chosen according to the maximum margin criterion is found by minimizing the objective function:

$$E = \|w\|_2^2$$
$$\text{with} \quad (x_i w + b) y_i \geq 1 \, , \forall i$$

The solution for the optimal boundary, w_0, is a linear combination of a subset of the training data, x_s called the *support vectors*. Data may be classified depending on the sign of $xw_0 + b$.

In speaker detection, this means that a speaker is modelled by its training data and by an optimum subset of impostor data, the closest impostors. This reduces considerably the problem size and is one of the reason of the SVM ability of dealing with large size of input training feature set.
Generally, data is not linearly separable, and the introduction of *slack variables* is necessary.

Mapping GMM parameters into a vector

In order to use the ability of the GMM to model the client, the weight of each component is used in the SVM to perform a verification task. Precisely, for each utterance (test and train) a GMM is trained following the different criteria presented in 3 and 4.
The weight of each component is explicitly used as a feature of an input vector. For each test segment X, the vector V_X has the following form:

$$V_X = [c_{1c} \ldots\ldots c_{nc}] \quad (12)$$

The size of the input vector is the same as the number of components in the GMM. This data will be labeled as 1 for a target segment, and -1 for an impostor.

5. SYSTEMS PRESENTATION

In this section, the different systems using MLE and MMIE criteria are presented. For each of these criteria, a GMM based system and a SVM based system are proposed. The LIA_SpkDet system based on classical UBM/GMM described in 2.2 will be further named GMM. The SVM system will be further named SVM.

5.1. MLE weight only systems

GMM

Models estimated via MAP and MLE are derived from the same background model. Gaussian weights of the target models are estimated using MLE. Two EM iterations are computed in order to adapt the Gaussian weights restricted to the weights parameters. Weight only MAP adaptation is performed using a *relevance* factor of 14.

[1] Notice that this approximation can be applied because the speaker models and the world model share the same mean and covariance parameters.

SVM

For every speaker model, a vector of weights is produced which represents the speaker utterance mapped onto a fixed length vector, here 2048. In order to build impostor models (i.e. negative labeled data), speakers coming from the background model are used, here 161 speakers. During the training, the input of the classifier is the concatenation of all impostor vectors and the speaker vector issued from its training data. During the verification process, the test vector is given as an input to SVM models. To map test utterances into the same space, a GMM has to be trained on all tests with the MLE criterion. For each of these tests, the weight vector will be used in the SVM. The maximum margin decision is found by passing the test weight vector through a linear kernel using the SVMLight Toolkit [14]. The output of the test is used as a score for verification. On each score, TNorm score normalization technique [15] is applied.

5.2. MMIE weight only systems

Here are introduced the two systems using the MMIE criterion to adapt Gaussian weights only.

GMM

The MMIE models are obtained as follows. The background model is gathered from the baseline system 2.2. The target models are then estimated by deriving only weights of the world model (MLE criterion via EM algorithm). A process is then applied on these models to adapt weights via the MMIE criterion. We compare the two approaches, the Woodland based adaptation function, *MMIE*, and the approximation of this function described in [16], *MMIE Approx*. For the MMIE only one iteration is computed whereas for the *MMIE Approx* three iterations are computed (the approximated function effect is less accurate). For each Gaussian of the target model, the MMIE adaptation function is computed. It represents the discriminative power of the target Gaussian weight comparing to the impostor Gaussian weight. We use 5% of the world features (value determined empirically). MMIE training is performed on the target and impostor models.

SVM

The same process as in 5.1 has been applied to produce the input vectors for the classification task. Indeed, instead of taking the parameter estimated from MLE, Gaussian weights estimated thanks to the MMIE criterion are used.

6. RESULTS

Experimental results obtained with the various weight adaptation methods are given below. Tables 1 and 2 summarize the results.

Weight Adaptation type	DCF	EER
MAP	7.21	16.51
MMIE	5.96	15.45
MLE	5.91	16.51
MMIE Approx	5.73	14.83

Table 1. *Comparing MAP, MLE, MMIE, MMIE Approx Gaussian weight estimation for the GMM system (w/o Tnorm).*

Weight Adaptation type	DCF	EER
MMIE Approx	5.08	14.61
MAP	4.76	13.43
MMIE	4.63	13.77
MLE	4.39	13.67

Table 2. *Comparing MAP, MLE, MMIE, MMIE Approx Gaussian weight estimation for the GMM system (Tnormed).*

Comparing MLE and MMIE in a GMM based system

The first experiments aim at evaluating the impact of MMIE training compared to standard weight estimation methods on a GMM system : MAP and MLE. These experiments have shown (see Table 1) that the system using the approximated function of the MMIE adaptation function improves performance of the system compared to a classic MLE estimation (DET curve 1, no score normalization). In fact, MMIE brings 3% and 11% relative reduction of DCF and EER respectively. No significative gain arises when using the standard MMIE criterion. Figures 1 and 2 show that performance of such a system is very close to a standard MLE system.

The TNorm score normalization is done using 161 impostors. When using TNorm, the MLE system outperfoms the MMIE one (DET curve 2 and table 2). Indeed, this normalization gives better results for the MLE weights estimation than for the MMIE. During the experiments, we notice that the impostor score distribution is no more Gaussian when adapting weights with MMIE, which could explain the poor results using TNorm.

Comparing MLE and MMIE with SVM systems

MLE parameters seem to be more suited when using an SVM system. Results on table 3 show that the difference between MLE and MMIE parameters is 18% relative for the DCF and 13% relative for the EER without score normalization. The TNorm score normalization technique does not improve the performance of the SVM system (nearly one percent gain for DCF and EER).

Using the top component selection

In the previous section, the classical top-component computation optimization is used. For a given test, likelihoods of the

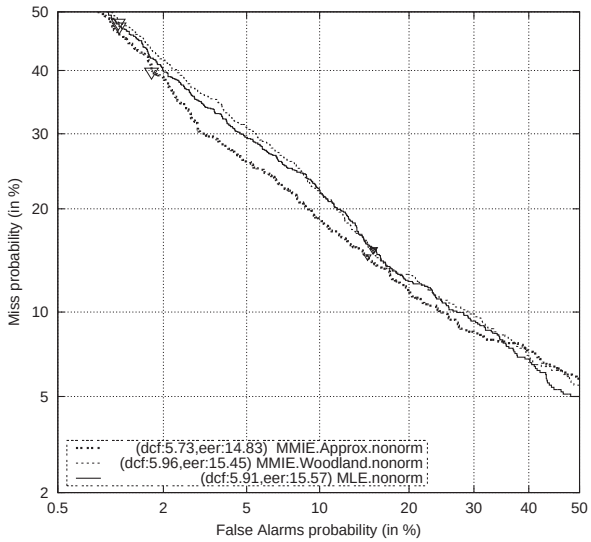

Fig. 1. DET curves for MMIE and MLE weight adaptation (w/o TNorm)

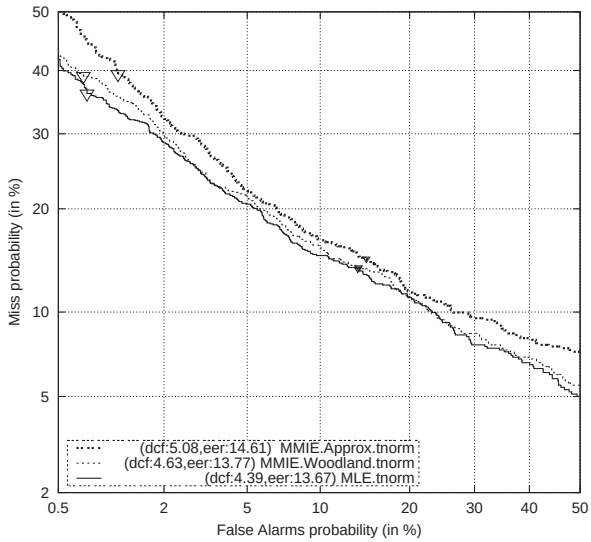

Fig. 2. DET curves for MMIE and MLE weight adaptation (TNormed)

Weight Adaptation type	DCF	EER
SVM MLE	4.30	12.00
SVM MLE Tnorm	4.25	11.9
SVM MMIE Approx	5.11	13.56
SVM MMIE Approx Tnorm	4.82	13.26

Table 3. *SVM results using MLE and MMIE weights as input vector of the SVM, with and without TNorm*

individual Gaussian components are computed (weighted by the corresponding world model component weight), a subset of the winning components is selected and client likelihoods are computed using only the selected components. A top 10 component selection (top_ten) is used during the LLR computation.
Even if this solution is commonly used in state-of-the-art speaker recognition systems, it is not straightforward to use it when only Gaussian weights are adapted for a target model and when discriminant weights are used. In order to assess this choice, we have set an experiment without using top component selection (i.e. using all the components, 2048 in our case). Results are shown in the figure 3. This figure proposes the DET curve for MMIE-based weight only system without component selection and with top_ten selection.

Surprisingly, when we expected similar results or better results without the top_ten selection, we observe a significative loss compared to the top_ten based system. One hypothesis for explaining this result is the difference in term of arithmetic domain between the weights and the likelihoods. It seems that using a component when it obtains a low likelihood for a given frame disturbs the results, even if this component is penalized by its weight. In our scheme, top_ten selection seems to work as a test-data classification of useful components which is able to increase the effect of discriminative weights.

7. FUSING SCORES OF SYSTEMS

The system combinations presented here are performed by computing an arithmetic mean of TNORM scores of two different systems. We fuse the baseline system 2.2 (MAP Mean only T-normed) with these of our different experiments. Results are listed on table 4 and 5.

Fusion Type	DCF	EER
baseline	3.49	8.73
MLE weight, baseline	3.30	9.57
MMIE weight, baseline	3.32	9.38
MMIE Approx weight, baseline	3.26	10.19

Table 4. *Fusions of a baseline GMM/UBM (MAP mean) with (1) the MLE weight adaptation (2) the MMIE weight adaptation (3) the MMIE Approx weight adaptation TNormed. Fusion Ratio 0.5 : Baseline; 0.5 : MLE, MMIE, MMIE Approx*

A win is observe for all the proposed fusions, with a relative gain in DCF of about 6% using MMIE, compared to state-of-the-art GMM/UBM baseline system. Furthermore, MMIE training does not give the expected results compared to MLE, as MLE weights achieve comparable results.

On the other hand, fusions with SVM systems give quite good results. A significative gain both on DCF and EER is reported in table 5. In fact the system improves the DCF and the EER from 9% and 7.6% relative gain respectively when fusing the SVM system with the GMM/UBM baseline. This gain

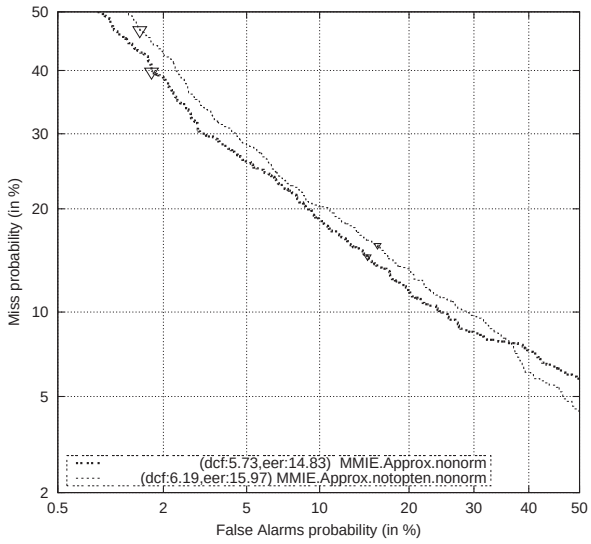

Fig. 3. DET curves for MMIE Approx with and without top ten component selection

Fusion Type	DCF	EER
baseline	3.49	8.73
MLE SVM, baseline	3.20	8.11
MMIE SVM, baseline	3.31	8.41

Table 5. *Fusions of a baseline GMM/UBM (MAP mean) with (1) the SVM system using MLE weight (2) the SVM system using the MMIE weight TNormed. Fusion Ratio 0.7 : GMM ; 0.3 : SVM*

is more than encouraging. MMIE fusions obtained a comparable gain (slightly smaller in fact). This latter result was expected, as the method leads to add discriminant information in a classifier able to extract the discriminative aspects of the input data.

8. DISCUSSION

SVM discriminative behavior gives quite good results on a Speaker Recognition task. It is confirmed by the experiments reported in this paper. Particularly, a relative improvement of about 9% was achieved, compared to our state-of-the-art GMM/UBM baseline, using NIST-SRE05 protocol.
The main objective of this work was to add an explicit discriminative behavior in a GMM/UBM system. The results showed an improvement when a MMIE-based discriminant system is fused with the baseline. Furthermore, results were comparable using a classical MLE approach. However, these results confirm the hypothesis of using the Gaussian weights as a way of discriminating speakers.
One possible reason for this mixed conclusion is the poor performance achieved by Tnorm score normalization with MMIE-based GMM systems. It seems that in this case the score distributions do not follow the underlined Tnorm hypothesis (normal distribution). Two solutions are open to solve this problem: normalizing the Tnorm score distribution or using a different score normalization.
Concerning the top component selection during likelihood computation, further investigations are needed for validating our hypothesis. It seems that a selection of a sub-set of components given a specific test frame is mandatory when the weights parameters are adapted for the client models.
We will also investigate how to use the discriminant information of Gaussian weights directly in the target speaker model. We wish to propose a speaker model training algorithm based on EM but incorporating the discriminant information.

9. ACKNOWLEDGEMENTS

This work was supported by the French *Ministère de la recherche et de l'industrie* under the CIFRE grant number 858/2005 in association with the Thales company.

10. REFERENCES

[1] NIST Speaker Recognition Evaluation campaigns web site, http://www.nist.gov/speech/tests/spk/index.htm

[2] Campbell, W. M. and Campbell, J. P. and Reynolds, D. A. and Jones, D. A. and Leek, T. R, "High-Level Speaker Verification with Support Vector Machines", ICASSP Conference, Montreal, CANADA, may 2004, pages 73-76.

[3] D. A. Reynolds, T. F. Quatieri, R. B. Dunn, "Speaker verification using adapted Gaussian mixture models", *Digital Signal Processing (DSP), a review journal Special issue on NIST 1999 speaker recognition workshop*, vol. 10(1-3), pp 19-41, 2000.

[4] F. Bimbot, J.-F. Bonastre, C. Fredouille, G. Gravier, I. Magrin-Chagnolleau, S. Meignier, T. Merlin, J. Ortega-Garcia, D. Petrovska, D. A. Reynolds, "A tutorial on text-independent speaker verification", *EURASIP Journal on Applied Signal Processing*, 2004, Vol.4, pp.430-451

[5] Woodland, P.C., Povey, D. , "Large Scale Discriminative Training of Hidden Markov Models for Speech Recognition". Computer Speech and Language, Vol. 16, pp. 25-47, 2002.

[6] Woodland, P.C., Povey, D. , "Large Scale Discriminative Training for Speech Recognition". Proceedings of International Workshop on Automatic Speech Recognition, 2000.

[7] LIA_SpkDet system web site, http://www.lia.univ-avignon.fr/heberges/ALIZE/LIA_RAL

[8] ALIZE project web site, http://www.lia.univ-avignon.fr/heberges/ALIZE/

[9] J.-F. Bonastre, F. Wils, S. Meignier, ”ALIZE, a free toolkit for speaker recognition”, *Proceedings of ICASSP05*, Philadelphia (USA), 2005

[10] ”SPRO: a free speech signal processing toolkit”, Guillaume Gravier,http://www.irisa.fr/metiss/guig/spro/

[11] Normandin, Y., Lacouture, R., Cardin, R., ”MMIE Training for Large Vocabulary Continuous Speech Recognition”. Proceedings of the IEEE, ICASP94, pp. 1367-1371, 1994.

[12] Normandin, Y., Morgera, D., ”An Improved MMIE Training Algorithm for Speaker-Independent Small Vocabulary”, Continuous Speech Recognition. Proceedings of the IEEE, ICASSP91, pp. 537-540, 1991.

[13] Vapnik, V. N., Statistical Learning Theory, Wiley, 1998.

[14] Joachims, T., ”Making large-Scale SVM Learning, Practical. Advances in Kernel Methods - Support Vector Learning”, MIT Press, 1999.

[15] Auckenthaler, R., Carey, M. and Lloyd-Thomas, H., ”Score normalization for text-independent speaker verification system”, Digital Signal Processing (DSP), a review journal - Special issue on NIST 1999 speaker recognition workshop, pages 42-54, vol 10, 2000.

[16] C. Levy , G. Linares, P. Nocera, J.F. Bonastre, ”Embedded mobile phone digit recognition”,Digital Signal Processing for In-Vehicle and Mobile Systems 2,chap 7, 2006 .