ELSEVIER

Speech Communication 24 (1998) 193–209

SPEECH COMMUNICATION

Text-independent speaker recognition using non-linear frame likelihood transformation

Konstantin P. Markov *, Seiichi Nakagawa

Department of Information and Computer Sciences, Toyohashi University of Technology, Toyohashi, 441-8122, Japan

Received 27 January 1997; received in revised form 3 November 1997; accepted 27 February 1998

Abstract

When the reference speakers are represented by Gaussian mixture model (GMM), the conventional approach is to accumulate the frame likelihoods over the whole test utterance and compare the results as in speaker identification or apply a threshold as in speaker verification. In this paper we describe a method, where frame likelihoods are transformed into new scores according to some non-linear function prior to their accumulation. We have studied two families of such functions. First one, actually, performs likelihood normalization – a technique widely used in speaker verification, but applied here at frame level. The second kind of functions transforms the likelihoods into weights according to some criterion. We call this transformation weighting models rank (WMR). Both kinds of transformations require frame likelihoods from all (or subset of all) reference models to be available. For this, every frame of the test utterance is input to the required reference models in parallel and then the likelihood transformation is applied. The new scores are further accumulated over the whole test utterance in order to obtain an utterance level score for a given speaker model. We have found out that the normalization of these utterance scores also has the effect for speaker verification. The experiments using two databases – TIMIT corpus and NTT database for speaker recognition – showed better speaker identification rates and significant reduction of speaker verification equal error rates (EER) when the frame likelihood transformation was used. © 1998 Elsevier Science B.V. All rights reserved.

Résumé

Quand les locuteurs de référence sont représentés par un modèle de mélange de gaussiennes, l'approche conventionnelle est d'accumuler les probabilités de trame sur l'énoncé de test entier et de comparer les résultats pour l'identification du locuteur ou d'appliquer un seuil pour la vérification du locuteur. Dans cet article, nous décrivons une méthode dans laquelle les probabilités de trame sont transformées, avant d'être sommées, en de nouveaux scores, suivant une certaine fonction non-linéaire. Nous avons étudié deux familles de fonctions. La première effectue de fait une normalisation des probabilités – une technique largement utilisée en vérification du locuteur –, mais qui est appliquée ici au niveau des états. Le deuxième type de fonctions transforme les probabilités en poids, suivant un certain critère. Nous appelons cette transformation "Weighting Models Rank" (WMR). Les deux types de transformations requièrent de pouvoir disposer de tous (ou d'un sous-ensemble de tous) les modèles de référence. Pour obtenir ceci, chaque trame de l'énoncé d'entrée est incorporée en parallèle dans les modèles de référence requis, puis la transformation des probabilités est appliquée. Les nouveaux scores sont ensuite accumulés sur l'ensemble de l'énoncé pour obtenir un score de l'énoncé pour un modèle de locuteur donné. Nous avons trouvé que la normalisation de ces scores d'énoncés est également

[*] Corresponding author. Tel.: +81 532 44 6777; fax: +81 532 44 6777.

PII: S0167-6393(98)00010-7

efficace pour la vérification du locuteur. Des expériences ont été menées sur deux bases de données – TIMIT et la base de données de NTT pour la reconnaissance du locuteur. Les résultats montrent des taux d'identification du locuteur plus élevés et une réduction notable du taux d'égale erreur (EER) en vérification du locuteur quand les transformations des probabilités de trames sont utilisées.

Keywords: Speaker identification; Speaker verification; Likelihood normalization; Frame level normalization

1. Introduction

Speaker recognition has been a research topic for many years and various types of speaker models have been studied. Hidden Markov models (HMM) have become the most popular statistical tool for this task. The best results have been obtained using continuous HMM (CHMM) for modeling the speaker characteristics (Savic and Gupta, 1990; Furui, 1991; Rosenberg et al., 1991, 1994; Matsui and Furui, 1992). For the text-independent task, where the temporal sequence modeling capability of the HMM is not required, one state CHMM, also called a Gaussian mixture model (GMM), has been widely used as a speaker model (Tseng et al., 1992; Reynolds and Rose, 1995; Gish and Schmidt, 1994; Bimbot et al., 1995; Matsui and Furui, 1995). In accordance with (Matsui and Furui, 1992) our previous study (Markov and Nakagawa, 1995) showed that GMM can perform even better than CHMM with multi-states.

The objective of the speaker identification is to find a speaker model λ_i given the set of reference models $\Lambda = \{\lambda_1, \ldots, \lambda_N\}$ and sequence of test vectors (or frames) $X = \{x_1, \ldots, x_T\}$ which gives the maximum a posteriori probability $P(\lambda|X)$. This requires the calculation of all $P(\lambda_j|X)$, $j = 1, \ldots, N$, and finding the maximum among them. In speaker verification, only the claimant speaker's model λ_c is used and $P(\lambda_c|X)$ is compared with a predetermined threshold in order to accept or reject X as being uttered from the claimant speaker.

In most of the tasks, it is possible to use the likelihood $p(X|\lambda)$ instead of $P(\lambda|X)$ which does not require prior probabilities $P(\lambda)$ to be known. Another simplifying assumption is that the sequence of vectors, X, are independent and identically distributed random variables. This allows to express $p(X|\lambda)$ as (Duda and Hart, 1973)

$$p(X|\lambda) = \prod_{t=1}^{T} p(x_t|\lambda), \tag{1}$$

where $p(x_t|\lambda)$ is the likelihood of single frame x_t given model λ. This is a fundamental equation of statistical theory and is widely used in speech recognition. Generally speaking, $p(X|\lambda)$ is an *utterance level score* of X given model λ obtained from *frame level scores* $p(x_t|\lambda)$ using Eq. (1). Obviously, another ways of defining such scores can exist.

Our approach is based on the following definition of the utterance level score:

$$\mathrm{Sc}(X|\lambda) = \prod_{t=1}^{T} \mathrm{Sc}(x_t|\lambda) = \prod_{t=1}^{T} f(p(x_t|\lambda)), \tag{2}$$

where $f(\ )$ is some function of frame likelihoods $p(x_t|\lambda)$ that transforms them into new scores $\mathrm{Sc}(x_t|\lambda)$. Actually, when this function is of the type $f(x) = x$, Eq. (2) becomes equivalent to Eq. (1). As it will be discussed in Section 6.1 any linear type of $f(\ )$ does not lead to reduction of the recognition errors. That is why we have considered non-linear likelihood transformations.

The first family of such functions we have experimented with essentially performs likelihood normalization, but now applied at the frame level. The likelihood normalization approach has been successfully used at the utterance level for speaker verification (Reynolds, 1995a; Rosenberg et al., 1992; Matsui and Furui, 1995; Higgins et al., 1991) but is usually not used for speaker identification purposes. This is simply because, as shown in Section 6.2, when applied only once at the utterance level likelihoods, it is a meaningless operation. Gish and Schmidt (1994) have shown that when the speaker scores are computed over relatively short time intervals (segments of the utterances) likelihood normalization may be successful. In their system each speaker is represented by multiple uni-modal Gaussian models (a special case of a GMM) trained on data from different sessions, and only the best model's score for each speaker over a given segment is taken into account. The segment scores are further normalized in order to obtain meaningful comparison between segments. Our method, however, differs from this study in two main points. First, in our system each speaker is represented by only one GMM and, second, likelihood normalization is done on each frame instead of short time intervals.

The second family of likelihood transformations converts the frame likelihood $p(x_t|\lambda)$ into one of a set of predetermined weights $w_j,\ j = 1,\ldots,N$. This type of transformation requires likelihoods from all reference models $p(x_t|\lambda_j)$ given the current frame to be calculated and sorted. Here we introduce the variable r_λ called *rank* of the model, which corresponds to the position of its likelihood in the sorted list and is an integer number ranging from 1 to N. Weights are function of the ranks r_λ,

$$w(r_\lambda) = g(r_\lambda), \tag{3}$$

where $g(\ )$ is some function of integer argument. Obviously, we can calculate all possible weights w in advance knowing the form of $g(\ )$ and the number of reference speakers N. Since weights and models ranks are involved in this type of likelihood transformation, we call it the *weighting models rank* (WMR) technique.

The rest of the paper is organized as follows. Section 2 gives brief description of the GMM we used. Section 3 provides details of speaker identification and verification tasks. Section 4 explains in detail our likelihood transformation approach. Section 5 describes our speech databases and summarizes our experimental results. In Section 6 we present some discussions and analysis of our method. Finally, we draw some conclusions in Section 7.

2. Gaussian mixture model

A GMM is a weighted sum of M component densities and is given by the form (Reynolds and Rose, 1995)

$$p(x|\lambda) = \sum_{i=1}^{M} c_i b_i(x), \tag{4}$$

where x is a d-dimensional random vector, $b_i(x), i = 1,\ldots,M$, is the component density and $c_i, i = 1,\ldots,M$, is the mixture weight. Each component density is a d-variate Gaussian function of the form

$$b_i(x) = \frac{1}{(2\pi)^{d/2}|\Sigma_i|^{1/2}} \exp\left\{ -\frac{1}{2}(x-\mu_i)^{\mathrm{T}}\Sigma_i^{-1}(x-\mu_i)\right\}, \tag{5}$$

with mean vector μ_i and covariance matrix Σ_i. The mixture weights satisfy the constraint that

$$\sum_{i=1}^{M} c_i = 1. \tag{6}$$

The complete Gaussian mixture model is parameterized by the mean vectors, covariance matrices and mixture weights from all component densities. These parameters are collectively represented by the notation

$$\lambda = \{c_i, \mu_i, \Sigma_i\}, \quad i = 1,\ldots,M. \tag{7}$$

In our speaker recognition system, each speaker is represented by such a GMM and is referred to by his/her model λ. GMM parameters are estimated using the standard maximum likelihood estimation (MLE) method via the expectation maximization (EM) algorithm (Dempster et al., 1979).

For a sequence of T test vectors $X = x_1, x_2, \ldots, x_T$, the standard approach is to calculate the GMM likelihood as in Eq. (1) which can be written in the log domain as

$$L(X|\lambda) = \log p(X|\lambda) = \sum_{t=1}^{T} \log p(x_t|\lambda). \tag{8}$$

3. Speaker recognition tasks

3.1. Speaker identification

Given a sample of a speech utterance, speaker identification is to decide to whom of a group of N known speakers this utterance belongs. In the closed set problem, it is assured that it belongs to one of the registered speakers.

As mentioned in Section 1, in the identification task, the aim is to find the speaker i^* whose model λ_{i^*} maximizes a posteriori probability $P(\lambda_i|X)$, $1 \leqslant i \leqslant N$, which according to the Bayes' rule is

$$P(\lambda_i|X) = \frac{p(X|\lambda_i)P(\lambda_i)}{p(X)}. \tag{9}$$

Furthermore, due to lack of prior knowledge, we assume equal-likely speaker models. That is, the prior probabilities $P(\lambda_i)$ are set equal:

$$P(\lambda_i) = \frac{1}{N}, \quad 1 \leqslant i \leqslant N. \tag{10}$$

The term $p(X)$ is actually the unconditional likelihood of the occurrence of the utterance X and is the same for all speakers. Therefore, $\max_i p(X|\lambda_i)$ will maximize a posteriori probability and the identification decision can be simplified to

$$i^* = \arg \max_i p(X|\lambda_i), \tag{11}$$

where i^* is the identified speaker.

Usually, a speaker identification system consists of collection of reference speaker models λ_i, front-end analysis and decision modules. The digitized speech utterance $S(n)$ is transformed into a sequence of feature vectors X and after that the likelihoods $p(X|\lambda_i)$, corresponding to each of the speaker models, are calculated. The best one is determined in the decision module and identifies the unknown speaker.

Since our likelihood transformation method requires frame likelihoods from all reference to be available for each frame, the structure of the speaker identification system has to be modified. Fig. 1 shows the structure of our speaker identification system (Markov and Nakagawa, 1996b). In this system, input speech is analyzed and transformed into a feature vector sequence by Front-end Analysis block and then each test vector x_t is fed to all reference speaker models in parallel. The ith speaker dependent GMM produces likelihood $p(x_t|\lambda_i), i = 1, 2, \ldots, N$, and all these likelihoods are passed in the so called *Likelihood Transformation and Accumulation* block, where they are transformed (according to the chosen transformation function) and accumulated for $t = 1, 2, \ldots, T$ to form the utterance level scores $\mathrm{Sc}(X|\lambda_i)$. The speaker identification is accomplished by comparing these scores in the Decision Logic block and determining the best one. The unknown speaker is classified as the speaker, whose model has the best score.

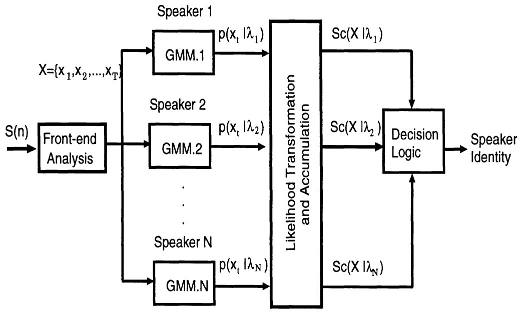


Fig. 1. Block diagram of our speaker identification system.

3.2. Speaker verification

Speaker verification is a binary decision problem where it has to be decided whether the speech utterance belongs to the claimant speaker or not. In the classical approaches, this decision is done by comparing the utterance score of the claimant speaker's model with some threshold determined at the training phase. The problem with this method is that the absolute value of the utterance score does not depend only on the speaker model used, but also on the lexical content of the speech and, therefore, a stable threshold cannot be set. One solution to this problem is to apply a likelihood normalization technique which has proven to significantly improve verification performance (Higgins et al., 1991; Rosenberg et al., 1992; Reynolds, 1995a; Matsui and Furui, 1995).

The general approach is to apply a likelihood ratio test (Fukunaga, 1990) to the input utterance $X = x_1, x_2, \ldots, x_T$ using the claimant speaker model λ_c,

$$l(X) = \frac{p(X|\lambda_c)}{p(X|\lambda_{\bar{c}})}, \tag{12}$$

where $\lambda_{\bar{c}}$ is a model representing all other possible speakers (impostors) and the prior probabilities $P(\lambda_c)$ and $P(\lambda_{\bar{c}})$ are assumed equal. The likelihood

$p(X|\lambda_c)$ is directly computed from Eq. (1) assuming that the speaker model is a GMM type,

$$p(X|\lambda_c) = \prod_{t=1}^{T} p(x_t|\lambda_c). \tag{13}$$

The likelihood $P(X|\lambda_{\bar{c}})$ is usually approximated using a collection of *background* speaker models. With the set of B background speaker models, $\{\lambda_1, \ldots, \lambda_B\}$, the background speaker's likelihood is computed as

$$p(X|\lambda_{\bar{c}}) = \frac{1}{B}\sum_{b=1}^{B} p(X|\lambda_b). \tag{14}$$

In the special case when $B = N$, i.e. all reference speakers including the claimed speaker act as background speakers and assuming that $P(\lambda_c) = P(\lambda_b)$, a posteriori probability $P(\lambda_c|X)$ scaled by the factor N approximates Eq. (12):

$$\begin{aligned} NP(\lambda_c|X) &= \frac{P(\lambda_c)p(X|\lambda_c)}{\frac{1}{N}p(X)} \approx \frac{P(\lambda_c)p(X|\lambda_c)}{\frac{1}{N}\sum_{b=1}^{N} p(X|\lambda_b)P(\lambda_b)} \\ &= \frac{p(X|\lambda_c)}{\frac{1}{N}\sum_{b=1}^{N} p(X|\lambda_b)} = l(X). \end{aligned} \tag{15}$$

In this case, using likelihood ratio test is equivalent to the speaker verification method based on a posteriori probability as reported in (Matsui and Furui, 1993).

Fig. 2 shows the structure of our speaker verification system. After the input speech signal is transformed into a sequence of feature vectors, frame likelihoods from the claimant speaker model and the background speaker models are calculated, further transformed and accumulated in the Likelihood Transformation and Accumulation block as in the speaker identification system. Then, using the claimant speaker score $\mathrm{Sc}(X|\lambda_c)$ and the background speaker scores, the likelihood ratio $l(X)$ is calculated. This is an utterance level normalization and is the same as the likelihood normalization used in the conventional speaker verification systems. It is needed since the score $\mathrm{Sc}(X|\lambda_c)$ as well as $p(X|\lambda_c)$ depends on the lexical content of the test utterance. $l(X)$ is compared with the threshold Θ and the decision is made according to the comparison result. Note that the background speaker sets for the frame normalization and for the utterance normalization need not be the same. We can choose different sets and use the combined background speaker set picking up only those scores which are necessary for the current type of normalization.

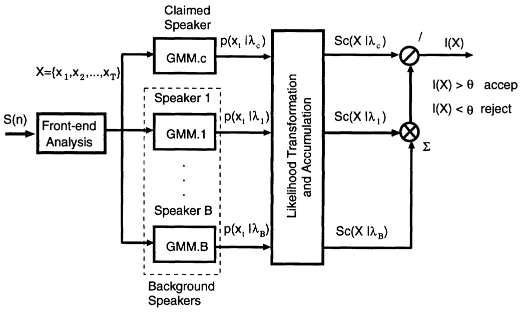


Fig. 2. Block diagram of our speaker verification system.

Setting of the threshold very much affects the performance of the verification system. For example, if the threshold is set high, a true speaker can be rejected. If it is too low, an impostor speaker could be accepted. These kinds of verification errors are measured in terms of false rejection (FR) and false acceptance (FA) rates. These error rates give us the estimate of the two kinds of errors given the threshold. Usually, verification performance is measured in terms of equal error rate (EER) (Matsui and Furui, 1995; Reynolds, 1995a). In this approach, the threshold is set such that FA and FR are equal. This is found by sorting true test scores and impostor test scores together and locating that point (or threshold) in the sorted list where the percent of impostor tests above this point is equal to the percent of the true tests below this point. Often the available test data per speaker are very few (especially the true tests) and setting a speaker dependent threshold would give results with low statistical significance. That is why, for small databases, a global threshold (same for all speakers) is used. In this case, true tests and impostor tests from all speakers are sorted together and then the threshold is located. It has to be noted that FA and FR are discrete functions of the threshold and the step from one point to the next one depends on the number of true tests for FR and impostor tests for FA. Obviously, FA and FR will intersect in one point if the number of true tests is equal to the number of impostor tests. However, in the leaving-one-out test scheme, there are always much more impostor than true tests. In this case, using the above algorithm for locating the threshold we will find the point where FR and FA are most close, but not equal. Some researches accept the EER as the FA at this point (Reynolds, 1995a). We have done detailed analysis of this situation which shows that more precise estimation of the EER can be obtained by linear approximation of FR and FA functions.

4. Frame likelihood transformations

4.1. Likelihood normalization

As we stated in Section 1, the first family of frame likelihood transformation functions performs the essentially likelihood normalization.

Given a single frame likelihood $p(x_t|\lambda_i)$ from the ith speaker model, the likelihood transformation is done using the following general function form:

$$f(p(x_t|\lambda_i)) = \frac{p(x_t|\lambda_i)}{\frac{1}{B}\sum_{b=1}^{B} p(x_t|\lambda_b)}, \qquad (16)$$

where $p(x_t|\lambda_b)$ are the frame likelihoods from the background speaker models given the same frame x_t. Different choices of the background speaker set give different transformation functions. Note that the above likelihood transformation approximates the likelihood ratio, as described in the previous section, but for a single frame. Thus, we transform the frame likelihood into a kind of confidence measure. Similar approach has been used for utterance verification purposes in speech recognition (Lleida and Rose, 1996).

Utterance level score, in this case, is obtained by inserting Eq. (16) into Eq. (2). For speaker i in the log domain we have

$$\log \mathrm{Sc}(X|\lambda_i) = \sum_{t=1}^{T}\left(\log p(x_t|\lambda_i) - \log\left(\frac{1}{B}\sum_{b=1}^{B} p(x_t|\lambda_b)\right)\right), \qquad (17)$$

$$= \sum_{t=1}^{T}\log p(x_t|\lambda_i) - \sum_{t=1}^{T}\log\left(\frac{1}{B}\sum_{b=1}^{B} p(x_t|\lambda_b)\right). \qquad (18)$$

It is easy to recognize that the first term of Eq. (18) is the standard $L(X|\lambda)$ from Eq. (8). The second term represents a correction consisting of likelihoods from the background speakers.

As in the utterance level likelihood normalization, here also arises the problem of choosing the proper background speaker set. In the closed set speaker identification task, however, we are restricted to choose from available set of N speakers. Given the speaker model i, we have experimented with the following background speaker sets:

- *All others*: the background speaker set consists of all registered speakers, except the speaker i.
- *Top M speakers*: since the likelihoods from all speaker models for the current vector x_t are available, it is possible to determine the speaker models, which have the M maximum likelihoods and the background speaker set in this case consists of these M speakers (excluding speaker i). Obviously, the top M speakers will change from frame to frame.
- *Cohort speakers*: the background speaker set consists of K acoustically closest speakers to the speaker i. The cohort speakers are determined on the training data in advance and this procedure is described in (Rosenberg et al., 1992).

4.2. Weighting models rank

This type of frame likelihood transformation in contrast to the normalization approach described in the previous section is new and is not based on any known techniques. The main idea is to transform the frame likelihood $p(x|\lambda)$ into a weight w which does not depend on the absolute value of this likelihood, but depends on its relative position with respect to the likelihoods from all other speaker models.

The WMR transformation is accomplished using the following two steps.

Step 1. For each test vector x_t, $t = 1, 2, \ldots, T$, calculate all likelihoods $p(x_t|\lambda_i)$, $i = 1, \ldots, N$, and sort them in a decreasing order. This is the same as making an N-best list of models. The model with the highest likelihood is at the top of this list and the model with the lowest likelihood – at the bottom. We can also say that each model has a *rank*, r_λ, which corresponds to the position of the model in this list and is an integer ranging from 1 to N. The weight w is defined as a function of r_λ,

$$w(r_\lambda) = g(r_\lambda). \tag{19}$$

The relations between ranks, weights and models are shown in Table 1. Actually, it is not necessary to calculate $g(r_\lambda)$ each time. Since it depends on r_λ which values are $1, 2, \ldots, N$, we can have all possible weights calculated prior to any experiments.

Step 2. For each model λ_i, find its rank r_{λ_i}, i.e. its place in the N-best list, and instead of the likelihood $p(x_t|\lambda_i)$ use the corresponding weight $w_t(r_{\lambda_i})$ as a model's frame score.

Utterance level score $\mathrm{Sc}(X|\lambda_i)$ is calculated by summing up (in the log domain) all weights for $t = 1, \ldots, T$:

Table 1
N-best list of speaker models

Rank r	Weight $w(r)$	Model
1	$w(1)$	Model λ_l (max. likelihood)
2	$w(2)$	Model λ_j
...	...	...
m	$w(m)$	Model λ_k
...	...	...
N	$w(N)$	Model λ_p (min. likelihood)

$$\log \mathrm{Sc}(X|\lambda_i) = \sum_{t=1}^{T} w_t(r_{\lambda_i}), \tag{20}$$

where $w_t(r_{\lambda_i})$ is the weight of the model i with rank r_{λ_i} at time t.

In accordance with Eq. (2), now we can define the WMR type likelihood transformation function as

$$f(p(x_t|\lambda)) = \exp(w_t(r_\lambda)). \tag{21}$$

Obviously, in this technique, the most important issue is what types of function $g(\;)$ to use. Previously we have experimented with following three typical functions (Markov and Nakagawa, 1996a):

$$g_{\exp}(r_\lambda) = \exp(A - Br_\lambda), \quad r_\lambda = 1, \ldots, N, \tag{22}$$

$$g_{\mathrm{lin}}(r_\lambda) = A - Br_\lambda, \quad r_\lambda = 1, \ldots, N, \tag{23}$$

$$g_{\mathrm{sig}}(r_\lambda) = \frac{A}{\exp Br_\lambda + 1}, \quad r_\lambda = 1, \ldots, N, \tag{24}$$

where A and B are parameters which we choose to be such that $g(1) \approx N$. Graphically these three functions are shown in Fig. 3. The reasons of choosing these functions and more detailed analysis of the WMR transformation performance are presented in Section 6.3.

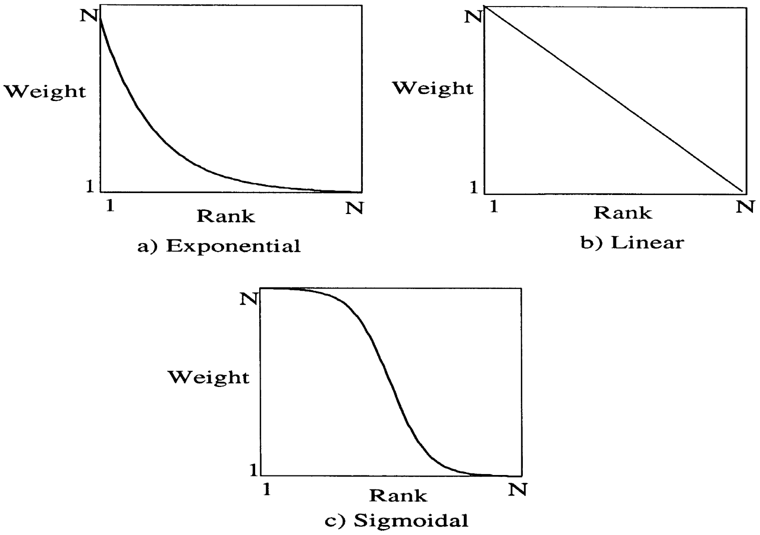


Fig. 3. Weights as functions of the model rank.

5. Experiments

We evaluated our speaker recognition system using several types of GMMs with both full and diagonal covariance matrices. As a baseline system, we used the conventional maximum likelihood testing approach based on Eq. (1) or Eq. (8).

5.1. Databases and speech analysis

We used two databases – NTT database and TIMIT corpus for the experiments.

The NTT database consists of recordings of 35 speakers (22 males and 13 females) collected in five sessions over 10 months (August 1990, September 1990, December 1990, March 1991 and June 1991) in a sound proof room (Matsui and Furui, 1992). For training the models, five same sentences for all speakers and five different sentences for each speaker, from one session (August 1990) were used. Five other sentences from the other four sessions uttered at normal, fast and slow speeds and same for each of the speakers and for each of the sessions were used as test data. Average duration of the sentences is about 4 s. The input speech was sampled at 12 kHz. 14 cepstrum coefficients were calculated by the 14th order LPC analysis at every 8 ms with a window of 21.33 ms. Then these coefficients were further transformed into 10 mel-cepstrum (cep) and 10 regressive (Δcep) coefficients. Each session's mel-cepstrum vectors were mean normalized by mean subtraction and silence parts were removed.

The well known TIMIT database, consisting of 6300 sentences (630 speakers × 10 sentences), was also used in evaluation experiments. Eight sentences – one SA (equal for all speakers), five SX (equal for groups of speakers) and two SI (different for each speaker) from each speaker were used for training and the remaining two (one SA and one SI) sentences for testing. The speech was first down-sampled to 12 kHz and then the same analysis was performed as for the NTT database, except that cepstrum vectors were not mean normalized and silence was not removed. For the TIMIT database, the cepstral mean normalization is not necessary because the training and testing conditions are the same. Removing the silence events, however, has been reported to slightly degrade the identification performance (Reynolds, 1995b).

5.2. Speaker identification experiments

Since we had a limited amount of training data – about 20–30 s of speech, for both databases, we were restricted in the number of model's parameters we could reliably estimate. That is why the models with full covariance matrix have 4 and 8 mixtures (and 16 mixtures for TIMIT database) and models with diagonal covariance matrix – 32 and 64 mixtures.

Cepstral and delta cepstral vectors are treated as separate feature streams. Therefore, frame likelihood transformations are performed independently on each feature frame. The combination is done at utterance level by summing the utterance scores from the two types of features:

$$\mathrm{Sc}^{\mathrm{comb}}(X|\lambda) = \mathrm{Sc}^{\mathrm{cep}}(X|\lambda) + \mathrm{Sc}^{\Delta\mathrm{cep}}(X|\lambda). \qquad (25)$$

5.2.1. NTT database results

Table 2 shows the identification rates, averaged over all test sessions, using frame likelihood normalization with the three types of background speaker sets – All others, Top M with $M = 10$ and Cohort and WMR transformation results. Cohort size is set to $B = 5$. Three separate experiments were done for each type of the test utterance speeds (speaking rate) – normal, slow and fast. In the table, the parts marked with "Normal speed", "Slow speed" and "Fast speed" show the identification rates in these three cases. Note that the speaker models were trained only with normal speed utterances. The column "Model type" shows the GMM structure. "4 mix. full" means a GMM with 4 mixture densities with full covariance matrices and "32 mix. diag." – GMM with 32 mixture densities with diagonal covariance matrices. The results in the "cep" rows present the identification rates when only 10-dimensional mel-cepstral feature vectors are used. Adding the cepstral derivative (Δcep) as a separate feature stream resulted in higher identification rates shown in the "cep + Δcep" rows. Table 2 shows that our frame likelihood transformation techniques give

Table 2
Identification rate (%) using GMM models and frame likelihood transformation techniques (NTT database)

		Likelihood normalization				
Model type	Feature	All others	Top 10	Cohort	WMR	Baseline
Normal speed						
4 mix. full	cep	92.8	92.7	92.4	92.4	92.3
	cep + Δcep	94.6	94.8	94.8	95.2	94.1
8 mix. full	cep	96.5	96.5	96.2	96.6	96.1
	cep + Δcep	97.0	97.0	97.0	97.3	97.0
32 mix. diag.	cep	95.5	95.5	95.2	95.0	95.0
	cep + Δcep	95.8	95.8	96.3	95.3	96.0
64 mix. diag.	cep	95.2	95.2	94.9	96.2	94.5
	cep + Δcep	95.7	95.7	95.9	95.8	95.4
Slow speed						
4 mix. full	cep	89.0	89.2	89.4	90.3	88.6
	cep + Δcep	91.6	91.6	92.4	91.0	90.8
8 mix. full	cep	92.0	92.0	92.7	93.9	91.3
	cep + Δcep	93.4	93.6	93.8	94.3	93.0
32 mix. diag.	cep	92.7	92.7	92.6	92.5	92.4
	cep + Δcep	92.6	92.6	93.0	92.6	92.3
64 mix. diag.	cep	90.9	90.9	92.0	91.4	90.0
	cep + Δcep	91.6	91.5	91.7	91.9	91.0
Fast speed						
4 mix. full	cep	90.9	90.7	91.2	89.9	90.4
	cep + Δcep	91.7	91.8	92.3	91.9	91.0
8 mix. full	cep	94.3	94.3	93.6	94.1	93.4
	cep + Δcep	94.6	94.5	94.3	94.8	94.0
32 mix. diag.	cep	92.6	92.6	93.2	91.4	91.7
	cep + Δcep	92.0	92.0	92.0	90.5	91.7
64 mix. diag.	cep	92.0	92.0	92.6	92.0	91.4
	cep + Δcep	92.3	92.3	91.9	92.4	91.4

better results than the baseline system. All types of the background speaker set give comparable identification rates. However, the more important result is that our methods are much better than the baseline at the "Slow" and "Fast" utterance speeds compared to the "Normal" speed. This fact indicates that the proposed types of frame likelihood transformation are more robust against variations of the speaking rate.

In the column "WMR", we present the exact results only for the case of exponential relationship between weights and ranks, since the other two – linear and sigmoidal were performing worse. It is noted that an identification rate of 97.3% is the best on this database (for comparison see (Matsui and Furui, 1992) with 95.6%) and is achieved using WMR technique and GMM with eight full covariance matrix mixtures.

In order to assess the significance of the improvements achieved by our methods, we have performed a statistical significance test on our best results (WMR technique with eight mixture, full covariance matrix GMM) using sign test methodology described in (Nakagawa and Takagi, 1994; Siegel, 1956). Using combined results of the all three test speeds, our proposed method showed the superiority over the baseline with a significance risk of 0.1% for the case when only cepstral features were used and 4.4% for the cepstral + Δcepstral features case.

5.2.2. *TIMIT database results*

Table 3 summarizes the results on TIMIT database. Identification rates for both the SA and SI test utterances are presented separately, because these are quite different types of sentences. SA is

Table 3
Identification rates (%) using GMMs (TIMIT database)

		Likelihood normalization				
Mod. type	Feature	All others	Top 20	Cohort	WMR	Baseline
SA test						
4 mix. full	cep	94.0	94.1	93.5	89.2	93.8
	cep + Δcep	94.8	94.9	94.9	89.8	94.6
8 mix. full	cep	97.0	97.0	97.1	97.1	96.8
	cep + Δcep	97.3	97.3	97.6	95.7	97.3
16 mix. full	cep	97.6	97.6	97.8	97.6	97.6
	cep + Δcep	96.8	96.8	97.2	98.1	96.8
32 mix. diag.	cep	95.2	95.2	95.1	94.4	95.4
	cep + Δcep	94.9	94.9	95.2	94.1	94.9
64 mix. diag.	cep	94.3	94.3	94.3	95.6	94.3
	cep + Δcep	94.0	93.8	94.3	94.6	93.8
SI test						
4 mix. full	cep	90.0	90.0	90.6	87.3	90.2
	cep + Δcep	91.1	91.4	91.1	87.0	91.1
8 mix. full	cep	93.7	93.7	93.5	94.4	93.7
	cep + Δcep	94.1	94.1	94.8	93.0	94.0
16 mix. full	cep	95.4	95.4	95.8	96.7	95.2
	cep + Δcep	93.8	93.8	94.0	95.1	93.3
32 mix. diag.	cep	92.2	92.1	93.0	94.6	92.4
	cep + Δcep	92.1	92.1	92.9	91.4	91.7
64 mix. diag.	cep	91.4	91.4	92.5	94.4	91.0
	cep + Δcep	89.8	89.8	91.4	91.6	89.8

the same for all speakers while SI is different for each speaker and, therefore, identification rates for SA are significantly higher.

TIMIT database is a popular database and often is used for speaker recognition experiments. An identification rate of 99.5% was reported in (Reynolds, 1995b) when GMM with 32 diagonal covariance matrices were used. We should notice, however, that our front-end speech analysis is quite different from (Reynolds, 1995b) where 16 kHz sampling rate and 30 cepstral coefficients were used.

In average, the performance of our likelihood transformation approach is comparable to the baseline and in some cases is slightly better. Note that the best result of 98.1% for TIMIT database is achieved using WMR approach as for the NTT database, but this time using GMM with 16 full covariance matrix mixtures. The reason is that from the TIMIT data silence was not removed and, thus, several of the GMM mixtures are necessary for modeling the silence. Since silences were removed from the NTT data, less mixtures were needed for the best performance.

5.3. Speaker verification experiments

5.3.1. NTT database results

In these experiments, each of the 35 speakers was acting as "customer" and all others as "impostors". Thus, rotating through all speakers we had 35 evaluation sets. Table 4 presents the number of customer and impostor tests used for each recording session as well as the total number of tests.

For the speaker verification, as stated above, it is proven that the utterance level likelihood normalization significantly improves the systems performance. In our experiments, we have combined our frame likelihood transformation techniques with the sentence level normalization. This is achieved by using the sentence level scores (Eq. (2)) obtained from the transformed frame likelihoods and normalizing them in the same way as in the conventional verification systems. Baseline results were also obtained using sentence level normalization of the accumulated frame likelihoods. For this normalization we used the same

Table 4
Number of customer and impostor tests for NTT database

	# Speakers	# True tests per speaker	# Impostor tests per speaker	Total # true tests	Total # impostor tests
One session	35	5	170	175	5950
Total (five sessions)	35	25	850	875	29750

types of background speaker sets as in the case of frame likelihood normalization. In order to save some space, in Table 5 we present only the results when Top M ($M = 10$) background speaker set is used for the sentence level likelihood normalization, which achieved in average the lowest EER. Results are averaged over all test sessions.

In this table, the column "Frame transformation" shows the type of the frame likelihood transformation used. "None" stands for our baseline system, that is, using only sentence level normalization with top 10 speakers as background speaker set. "All others", "Top 10" and "Cohort" define the type of the background speaker sets used for frame likelihood normalization and "WMR" stands for WMR transformation technique. Results clearly show that our approach outperforms the baseline. Among the different background speaker sets, the "Cohort" type is the best, and compared to the baseline, the WMR transformation gives up to more than two fold reduction of the EER. Results similar to our baseline system performance were reported in (Matsui and Furui, 1995).

5.3.2. TIMIT database results

It was difficult to compute the EER for all the 630 TIMIT speakers mainly because of memory limitations. That is why, for the speaker verification experiments on TIMIT database, we chose 168 speakers recommended as test speakers for the database. The same was done in (Reynolds, 1995a). There are 112 male and 56 female speakers with 2 test sentences per speaker. In the experiments, one speaker was acting as true speaker and all other as impostors. Rotating over all speakers, this gives $168 \times 2 = 336$ true tests and $167 \times 2 \times 168 = 56112$ impostor tests. Table 6 shows the EER with Top 20 background speaker set for the sentence level likelihood normalization.

Table 5
Verification EER (%) for NTT database using different frame transformations and sentence level normalization

		GMM type							
		4 mix. full		8 mix. full		32 mix. diag.		64 mix. diag.	
Test utterance speed	Frame transformation	cep	cep + Δcep	cep	cep + Δcep	cep	cep + Δcep	cep	cep + Δcep
Normal	None	2.50	1.64	1.66	1.18	1.65	1.29	1.60	1.07
	All others	2.31	1.51	1.43	1.09	1.48	1.14	1.24	0.87
	Top 10	2.30	1.48	1.44	1.09	1.48	1.13	1.24	0.88
	Cohort	2.14	1.33	1.38	0.96	1.29	1.00	1.20	0.86
	WMR	1.31	0.84	0.66	0.52	0.91	0.95	0.72	0.60
Slow	None	3.79	2.96	2.46	2.06	2.95	2.36	3.15	2.57
	All others	3.36	2.79	2.18	1.95	2.60	2.25	2.76	2.38
	Top 10	3.33	2.77	2.16	1.95	2.62	2.26	2.76	2.39
	Cohort	3.00	2.27	2.06	1.77	2.16	1.92	2.23	1.94
	WMR	1.94	2.06	1.45	1.36	1.50	1.76	1.57	1.43
Fast	None	3.07	2.26	2.01	1.43	3.06	2.88	2.65	2.66
	All others	2.78	2.15	1.89	1.27	2.91	2.71	2.42	2.44
	Top 10	2.75	2.15	1.88	1.26	2.91	2.71	2.44	2.43
	Cohort	2.65	1.93	1.66	1.09	2.51	2.58	2.06	2.44
	WMR	1.92	1.29	1.11	0.80	1.90	1.79	1.48	1.28

Table 6
Verification EER (%) for TIMIT database

	GMM type									
	4 mix. full		8 mix. full		16 mix. full		32 mix. diag.		64 mix. diag.	
Frame transformation	cep	cep + Δcep	cep	cep + Δcep	cep	cep + Δcep	cep	cep + Δcep	cep	cep + Δcep
None	0.72	0.61	0.43	0.46	0.40	0.46	0.59	0.66	0.57	0.76
All others	0.71	0.60	0.42	0.45	0.39	0.45	0.58	0.65	0.57	0.76
Top 20	0.71	0.60	0.42	0.45	0.39	0.45	0.58	0.65	0.57	0.76
Cohort	0.67	0.56	0.38	0.41	0.35	0.40	0.51	0.58	0.53	0.70
WMR	0.48	0.39	0.16	0.15	0.16	0.19	0.09	0.13	0.24	0.19

Sentence level normalization – Top 20.

In TIMIT database, the test and train conditions are the same which is big simplification for the task and, consequently, it is more difficult to outperform the baseline performance. This is evident from the results of "All others" and "Top 20" background speaker sets which in contrast to the multi-session NTT database are the same as the baseline. The "Cohort" is slightly better, and only WMR significantly reduces the EER and we obtained 0.09% ERR. For comparison, the best EER reported in (Reynolds, 1995a) is 0.24% obtained using cohort type utterance level likelihood normalization.

Using a common set of true speakers and impostors, as in these experiments, i.e. a closed set problem, does not allow us to assess the performance of the speaker verification system in a real verification scenario. This is because in the real world application the system will have knowledge only about the true speakers and neither the number of impostors nor their features can be known ahead of time. In contrary, in the above experiments, impostors are implicitly assumed to be known to the system.

In order to simulate a real verification task (an open set problem), we performed a second set of experiments where the same 168 TIMIT test speakers were acting as true speakers and all other 462 train speakers served as impostors. This makes 336 true tests and 924 impostor tests. In the open set task, we also have to address the problem of choosing the verification threshold. Calculating the EER in this case is not the best choice since it uses a posteriori threshold. Setting the threshold is a challenging problem which still has not been solved (Matsui et al., 1996).

Table 7 summarizes the results of the open set experiments in terms of both EER and a fixed threshold set to the value of the EER threshold calculated in the first set of experiments (those from Table 6). As it is apparent from the table, in terms of EER both our methods – frame level likelihood normalization with cohort background speakers and WMR technique, are better than the baseline system. Results using a fixed threshold show that in this case only WMR gives significantly lower false acceptance rate with almost the same number of false rejection errors (0.3% FR corresponds to one rejection of a true test).

6. Discussion

6.1. Linear versus non-linear frame likelihood transformation

When considering the type of the likelihood transformation function $f(\ )$ of Eq. (2), it is very important to choose the right one. Since it is not quite obvious why the linear type of $f(\ )$ is not appropriate, below we prove that the linear transformation of the frame likelihoods does not change the recognition rate.

Consider the linear transformation function $f(x) = ax + b$ and the frame likelihood $p(x_t|\lambda_i)$ of ith speaker model at time t. Then, the transformed likelihood is

$$f(p(x_t|\lambda_i)) = ap(x_t|\lambda_i) + b. \tag{26}$$

In the speaker identification task, we are interested in the model $i^\star$ which gives the best score. For the

Table 7
Verification error rates (%) for TIMIT database – open set problem

	GMM type									
	4 mix. full		8 mix. full		16 mix. full		32 mix. diag.		64 mix. diag.	
Frame transformation	cep	cep + Δcep	cep	cep + Δcep	cep	cep + Δcep	cep	cep + Δcep	cep	cep + Δcep
Equal error rate										
None	1.15	1.25	0.61	0.70	1.01	1.12	0.98	1.17	1.13	1.12
Cohort	1.10	1.16	0.58	0.63	0.85	1.02	1.00	1.03	1.10	0.88
WMR	1.18	1.41	0.50	0.62	0.74	0.70	0.69	0.76	0.73	0.75
False rejection rate										
None	0.90	0.60	0.60	0.60	0.60	0.60	0.60	0.60	0.90	0.90
Cohort	0.60	0.60	0.60	0.60	0.60	0.30	0.60	0.30	0.60	0.60
WMR	0.90	0.60	0.30	0.30	0.60	0.30	0.60	0.60	0.60	0.30
False acceptance rate										
None	1.66	1.84	1.46	1.84	1.67	1.87	1.67	1.61	2.32	2.39
Cohort	1.68	1.88	1.52	2.03	1.71	1.99	1.61	1.52	2.16	2.49
WMR	1.52	1.63	0.83	1.08	1.17	1.05	1.08	1.07	1.31	1.05

Sentence level norm. – Top 20.

standard maximum likelihood approach based on Eq. (1), it is

$$i^{\star} = \arg\max_{i} \prod_{t=1}^{T} p(x_t|\lambda_i). \tag{27}$$

When the frame likelihoods are linearly transformed, the above equation becomes

$$\begin{aligned} i^{\star} &= \arg\max_{i} \prod_{t=1}^{T} (a p(x_t|\lambda_i) + b) \\ &= \arg\max_{i}\, a^{T} \prod_{t=1}^{T} \left(p(x_t|\lambda_i) + \frac{b}{a} \right) \\ &= \arg\max_{i} \prod_{t=1}^{T} \left(p(x_t|\lambda_i) + \frac{b}{a} \right) \\ &= \arg\max_{i} \prod_{t=1}^{T} p(x_t|\lambda_i), \end{aligned} \tag{28}$$

which shows that whether the frame likelihoods are linearly transformed or not, the identified speaker is the same and, therefore, that the linear type of $f(\ )$ does not change the identification rate.

In the speaker verification task, the EER, as usually computed from the sorted list of all the true and impostor utterance scores, may change only if after the frame likelihood transformation at least two scores (one true and one impostor score) change their positions in this sorted list. In other words, if for any two models λ_{true} and λ_{imp}, their utterance likelihoods are related as

$$p(X|\lambda_{\text{true}}) < p(X|\lambda_{\text{imp}}) \quad \text{or} \quad p(X|\lambda_{\text{true}}) > p(X|\lambda_{\text{imp}}) \tag{29}$$

and the corresponding utterance scores obtained from the transformed frame likelihoods change the above inequalities, i.e.,

$$\begin{aligned} &\mathrm{Sc}(X|\lambda_{\text{true}}) > \mathrm{Sc}(X|\lambda_{\text{imp}}) \\ &\text{or} \quad \mathrm{Sc}(X|\lambda_{\text{true}}) < \mathrm{Sc}(X|\lambda_{\text{imp}}) \end{aligned} \tag{30}$$

only then the EER may change.

Let us now consider the linear transformation. Obviously, if for any i and j

$$p(X|\lambda_i) = \prod_{t=1}^{T} p(x_t|\lambda_i) > p(X|\lambda_j) = \prod_{t=1}^{T} p(x_t|\lambda_j), \tag{31}$$

then the corresponding scores will preserve the inequality

$$\begin{aligned} \mathrm{Sc}(X|\lambda_i) = \prod_{t=1}^{T} (a p(x_t|\lambda_i) + b) &> \mathrm{Sc}(X|\lambda_j) \\ &= \prod_{t=1}^{T} (a p(x_t|\lambda_j) + b), \end{aligned} \tag{32}$$

which means the linear transformation of the frame likelihoods would not change the verification error rate.

From the above considerations follows that only non-linear type of transformation is capable of changing the speaker recognition rates.

6.2. Frame versus utterance level likelihood normalization

It is well known that the likelihood normalization applied to utterance likelihoods does not change the speaker identification rate. Below we show this fact and why applied at the frame level it can be useful.

Consider, for example, the likelihood normalization using "All others" background speaker set (as the simplest case). Let us also denote, for simplicity, the likelihoods from any two speaker models i and j as p_i and p_j (it does not matter whether these are frame or utterance likelihoods) and the corresponding normalized likelihoods as Sc_i and Sc_j. Then we have

$$\mathrm{Sc}_i = \frac{p_i}{\frac{1}{N-1}\sum_{b\neq i} p_b} = \frac{p_i}{\frac{1}{N-1}\left(\sum_{b=1}^{N} p_b - p_i\right)} = \frac{p_i}{C - \frac{p_i}{N-1}}. \tag{33}$$

The same holds for Sc_j:

$$\mathrm{Sc}_j = \frac{p_j}{C - \frac{p_j}{N-1}}. \tag{34}$$

Since in the speaker recognition task we are interested in their relation, then, if the likelihoods ratio is

$$\frac{p_i}{p_j} = k, \tag{35}$$

the normalized likelihoods ratio becomes

$$\frac{\mathrm{Sc}_i}{\mathrm{Sc}_j} = \frac{p_i\left(C - \frac{p_j}{N-1}\right)}{p_j\left(C - \frac{p_i}{N-1}\right)} = k\frac{(N-1)C - p_j}{(N-1)C - kp_j} = k\frac{A - p_j}{A - kp_j}, \tag{36}$$

where $A = (N-1)C$.

Now, if $k > 1$, then

$$k\frac{A - p_j}{A - kp_j} > k \tag{37}$$

and if $k < 1$ then

$$k\frac{A - p_j}{A - kp_j} < k. \tag{38}$$

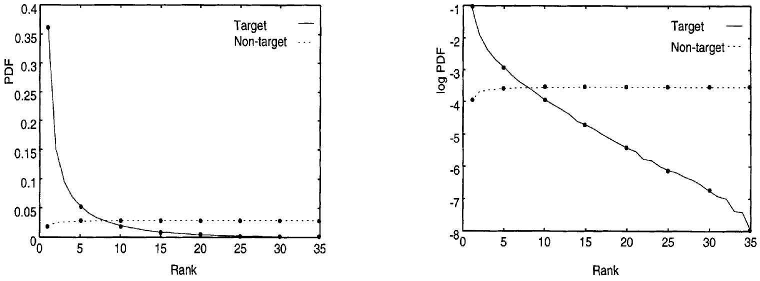
This means that this type of likelihood normalization gains the ratio between likelihoods, but does not invert the inequality. That is why, when it is applied only once at utterance level, for speaker identification task, it cannot change the identification rate (the same holds for the other types of background speaker selection), because the speaker with maximum utterance likelihood $p(X|\lambda)$ after such normalization will still have the maximum normalized likelihood $\mathrm{Sc}(X|\lambda)$.

Let us now consider the likelihood normalization when applied at frame level. For each frame we will have a gain (or loss) in the likelihood ratio which is accumulated over the whole utterance. Then, in the case when the target speaker i is misrecognized with a similarly performing (having similar utterance likelihood) speaker j because of a small number of outlier frames, the loss acquired from these outliers can become less than the gain from the majority of the frames. Thus, the target speaker utterance level score can become bigger and it will be correctly recognized.

6.3. Choosing the WMR weight function

The weight, as defined in Section 4.2, is a function of the model's rank. In order to choose an appropriate function, it is necessary to acquire some additional knowledge about the rank. Obviously, the rank of a given model is a random variable since it depends on a random variable – the model's likelihood $p(x_t|\lambda)$ (assuming x_t is itself random). Then, we can gather some statistics of the rank, which would give us insight of how to better set the weights.

Our task is to improve the true speaker models performance, because usually when a speaker is misclassified it is not due to a non-target speaker doing well, but rather to true speaker's model doing poorly (Gish and Schmidt, 1994). That is why we would be interested in probability of target speaker model having rank r as well as in probability of non-target speaker model having the same rank. In other words, we need to know the probability density functions (pdf) of target and non-target model ranks – $f_{\mathrm{target}}(r)$ and $f_{\text{non-target}}(r)$. They can be estimated easily on the training data. Fig. 4 shows these functions estimated using the NTT database and eight mixture GMM (cepstral feature vectors only) in the linear and log domains.



a) linear domain.

b) log domain.

Fig. 4. Rank's pdf for target and non-target speaker models (NTT database, eight mix. GMM).

We can see that $f_{\text{target}}(r)$ is almost exponential while $f_{\text{non-target}}(r)$ is close to uniform. The same functions obtained using all other types of GMM had a similar shape.

Generally, rank's pdf $f(r)$ as well as the weight $w(r_\lambda)$ is a function of the rank. And here naturally comes the following question: What if we use the estimated probabilities $P(r) = \Pr(r_{\lambda_{\text{target}}} = r)$ as weight-type scores (instead of conventional likelihoods) in our test? It is possible and we have done such an experiment. The obtained results were better than the baseline but worse that the WMR test. This is because the probability estimates are based on the training data and we cannot expect good generalization on unseen data (test data) of this approach. That is why, by correcting the shape of $f_{\text{target}}(r)$, i.e. by using a proper weight function, we could deal with this problem.

Since the weight w corresponds to log $P(r)$, we should focus our attention on Fig. 4(b) in order to choose a weight function. For the top 5–10 ranks, log $f_{\text{target}}(r)$ is close to exponential and after that it becomes almost linear. Setting the weight to be an exponential function of the rank will approximate the shape of log $f_{\text{target}}(r)$ for the top ranks. A linear function will be an approximation for the other values of the rank. Our preliminary experiments (Markov and Nakagawa, 1996a) showed that linear weight performs similarly or even worse than the baseline. It appears, that the most critical, in recognition point of view, is the shape of the weight function for the top ranks. To verify this, we experimented with sigmoidal weight function which is quite different, especially, for the top ranks. It performed the worst.

When we have decided what weight function to use, the next step is how to choose its parameters (recall Eqs. (22)–(24)). Let us first find the mean value of the weight for the target and non-target speakers. Weight mean can be expressed by the weight function and the rank pdf. The expected value of a function of a random variable is (Papoulis, 1991)

$$E\{g(x)\} = \int_{-\infty}^{\infty} g(x)f(x)\,\mathrm{d}x, \tag{39}$$

where $f(x)$ is the pdf of the random variable x and $g(x)$ is a function of x. If x is of discrete type, the above equation becomes

$$E\{g(x)\} = \sum_i g(x_i)\Pr(x = x_i). \tag{40}$$

Therefore, for the two means we have

$$m_{\text{target}} = \sum_{r=1}^{N} g(r, \Theta)P_{\text{target}}(r), \tag{41}$$

$$m_{\text{non-target}} = \sum_{r=1}^{N} g(r, \Theta)P_{\text{non-target}}(r), \tag{42}$$

where $g(r, \Theta)$ is the weight function with parameters $\Theta = \{\theta_i\}$. Before going to determine the optimal value of each parameter θ_i, we have to know whether it is able to change the recognition rate or not. Obviously, if a change of some parameter has the same effect on both target and non-target speaker scores (weights) the recognition rate will be the same. However, we cannot examine the speaker scores because they are not available prior to the recognition. But since the effect of the parameter change on the means m_{target} and $m_{\mathrm{non\text{-}target}}$ is the same as on the scores, we can focus our analysis on them. Now we can formulate our decision on θ_i as: *if any change of θ_i results in the same degree of change of m_{target} with respect to the change of $m_{\mathrm{non\text{-}target}}$, then θ_i will have no effect on the recognition rate.* [1] Thus, we have to analyze the following function:

$$J(\theta_i) = \frac{\frac{\partial}{\partial\theta_i}\sum_{r=1}^{N} g(r,\Theta)P_{\mathrm{target}}(r)}{\frac{\partial}{\partial\theta_i}\sum_{r=1}^{N} g(r,\Theta)P_{\mathrm{non\text{-}target}}(r)}, \tag{43}$$

and if $J(\theta_i) = \mathrm{const} \geqslant 0$ then θ_i will be of no interest. In the other case, when $J(\theta_i) \neq \mathrm{const}$, we can find the optimal θ_i as one which maximizes the difference between m_{target} and $m_{\mathrm{non\text{-}target}}$:

$$\theta_i^{\mathrm{opt}} = \arg\max_{\theta_i}\left(\sum_{r=1}^{N} g(r,\Theta)P_{\mathrm{target}}(r) - \sum_{r=1}^{N} g(r,\Theta)P_{\mathrm{non\text{-}target}}(r)\right), \tag{44}$$

which is equivalent to

$$\theta_i^{\mathrm{opt}} = \arg(J(\theta_i) = 1). \tag{45}$$

It can be easily verified that when the weight is a linear function of the rank (see Eq. (23)), then $J(A)$ as well as $J(B)$ is constant. Our experimental results confirmed that the speaker recognition rate in this case does not depend on either A or B. For the exponential weight (see Eq. (22)), the analysis shows that only the parameter B can effect the recognition rate, which is also experimentally verified.

The expression under the big parentheses in Eq. (44) can also be used as a measure of the weight function effectiveness with respect to the recognition rate. The bigger this difference is, the better recognition rate can be expected. For the example of Fig. 4, we calculated this difference for the linear and exponential weight functions. We obtained 0.39 and 0.45, respectively. It is also possible to assess in the same way the effectiveness of the conventional log-likelihood score by first averaging (normalized) log-likelihoods for each rank and then using these averages as values of discretely defined weight function. For this case we obtained 0.38 difference between target and non-target means, which is very close to that for the linear weight function and far below from that for the exponential weight function. This explains why the WMR test with exponential weight is superior to the conventional maximum likelihood test.

7. Conclusion

We have developed and experimented a non-linear frame likelihood transformation method, which allowed as to apply successfully likelihood normalization technique for the speaker identification task. For the speaker verification, the combination of frame and utterance level likelihood normalization was also successful. Another new technique, WMR transformation, was experimented with as well. Both approaches showed better results in the speaker identification and speaker verification compared to the standard accumulated likelihood methods on both the TIMIT and NTT databases. The NTT database results indicate that our transformation techniques are robust against variations in the speaker voices as well as utterance speeds. The best speaker identification rate of 97.3% and ERR of 0.52% are both achieved using WMR technique. For the TIMIT database, the identification rate of 98.1% is not the best ever reported, but we attribute this to the fact that our front-end analysis was not optimized for this database. However, even in this case, we achieved extremely low verification EER of 0.09%.

We also have shown that any linear transformation of the likelihoods at the frame level does not affect the performance of the speaker recognition system.

The number of possible frame likelihood transformation techniques, for sure, is not limited to those presented in this paper. There is a room for further research in this direction, for looking for new types of non-linear transformation functions as well as incorporating these techniques in the process of GMM training.

[1] It is not difficult to prove this statement. We skip the proof in order to save some space.

References

Bimbot, F., Magrin-Chagnolleau, I., Mathan, L., 1995. Second-order statistical measures for text-independent speaker identification. Speech Communication 17 (1–2), 177–192.

Dempster, A., Larid, N., Rubin, D., 1979. Maximum likelihood estimation from incomplete data. Journal of the Royal Statistical Society B 39 (1), 1–38.

Doddington, G., 1985. Speaker recognition – Identifying people by their voices. Proceedings of the IEEE 73 (11), 1651–1664.

Duda, R., Hart, P., 1973. Pattern Classification and Scene Analysis, Wiley, New York, p. 46.

Fukunaga, K., 1990. Introduction to Statistical Pattern Recognition, Academic Press, New York.

Furui, S., 1978. A study on personal characteristics in speech sound, Ph.D. Thesis, University of Tokyo (in Japanese).

Furui, S., 1991. Speaker-dependent-feature extraction, recognition and processing techniques. Speech Communication 10 (5–6), 505–520.

Gish, H., Schmidt, M., 1994. Text-independent speaker identification. IEEE Signal Processing Magazine, October, pp. 18–32.

Higgins, A., Bahler, L., Porter, J., 1991. Speaker verification using randomized phrase prompting. Digital Signal Processing 1, 89–106.

Linde, Y., Buzo, A., Gray, R.M., 1980. An algorithm for vector quantizer design. IEEE Transactions on Communications COM–28, 84–95.

Lleida, E., Rose, R., 1996. Efficient decoding and training procedures for utterance verification in continuous speech recognition. In: Proceedings of ICASSP'96, pp. 507–510.

Markov, K., Nakagawa, S., 1995. Text-independent speaker identification on TIMIT database. In: Proceedings of the Acoustical Society of Japan, March 1995, pp. 83–84.

Markov, K., Nakagawa, S., 1996a. Text-independent speaker recognition system using frame level likelihood processing. Technical Report of IEICE, SP96-17, June 1996, pp. 37–44.

Markov, K., Nakagawa, S., 1996b. Frame level likelihood normalization for text-independent speaker identification using Gaussian mixture models. In: Proceedings of ICSLP'96, pp. 1764–1767.

Matsui, T., Furui, S., 1992. Comparison of text-independent speaker recognition methods using VQ-distortion and discrete/continuous HMMs. In: Proceedings of ICASSP'92, Vol. II, pp. 157–160.

Matsui, T., Furui, S., 1993. Concatenated phoneme models for text-variable speaker recognition. In: Proceedings of ICASSP'93, Vol. II, pp. 391–394.

Matsui, T., Furui, S., 1995. Likelihood normalization for speaker verification using a phoneme- and speaker-independent model. Speech Communication 17 (1–2), 109–116.

Matsui, T., Nishitani, T., Furui, S., 1996. Robust methods of updating model and a priori threshold in speaker verification. In: Proceedings of ICASSP'96, Vol. I, pp. 97–100.

Nakagawa, S., Takagi, H. 1994. Statistical methods for comparing pattern recognition performance algorithms and comments on evaluating speech recognition. Journal of the Acoustical Society of Japan 50 (10), 849–854 (in Japanese).

Papoulis, A., 1991. Probability, Random Variables, and Stochastic Processes, 3rd ed. McGraw-Hill, New York, p. 105.

Reynolds, D.A., 1995a. Speaker identification and verification using Gaussian mixture speaker models. Speech Communication 17 (1–2), 91–108.

Reynolds, D.A., 1995b. Large population speaker identification using clean and telephone speech. IEEE Signal Processing Letters 2 (3), 46–48.

Reynolds, D.A., Rose, R.C., 1995. Robust text-independent speaker identification using Gaussian mixture speaker models. IEEE Transactions on Speech and Audio Processing 3 (1), 72–83.

Rosenberg, A., Lee, C., Gokcen, S., 1991. Connected word talker verification using whole word Hidden Markov Models. In: Proceedings of ICASSP'91, pp. 381–384.

Rosenberg, A., DeLong, J., Lee, C., Juang, B., Soong, F., 1992. The use of cohort normalized scores for speaker verification. In: Proceedings of ICSLP'92, pp. 599-602.

Rosenberg, A., Lee, C., Soong, F., 1994. Cepstral channel normalization techniques for HMM-based speaker verification. In: Proceedings of ICSLP'94, pp. 1835–1838.

Savic, M., Gupta, S., 1990. Variable parameter speaker verification system based on Hidden Markov Modeling. In: Proceedings of ICASSP'90, pp. 281–284.

Siegel, S., 1956. Nonparametric Statistics for the Behavioral Sciences. McGraw-Hill, New York, pp. 68–75.

Soong, F.K., Rosenberg, A.E., Rabiner, L.R., Juang, B.H., 1987. A vector quantization approach to speaker recognition. AT&T Technical Journal 66, 14–26.

Tishby, N.Z., 1991. On the application of mixture AR hidden Markov models to text independent speaker recognition. IEEE Transactions on Signal Processing 39, 563–570.

Tseng, B., Soong, F., Rosenberg, A., 1992. Continuous probabilistic acoustic map for speaker recognition. In: Proceedings of ICASSP'92, Vol. II, pp. 161–164.