# Speaker Modeling Technique Based on Regression Class for Speaker Identification with Sparse Training

Zhonghua Fu and Rongchun Zhao

School of Computer Science, Northwestern,
Polytechnical University, Xi'an 710072, P.R. China
mailfzh@vip.sina.com

**Abstract.** Speaker modeling technique with sparse training data is an active branch of robust speaker recognition research. This paper presents a novel modeling approach named Multi-EigenSpace modeling technique based on Regression Class (RC-MES), which integrates the common eigenspace technique and the regression class (RC) idea of Maximum Likelihood Linear Regression (MLLR). RC-MES not only solves the problem of prior knowledge limitation of Gaussian Mixture Models (GMM) but also remedies the shortcoming of common eigenspace that confuses speaker differences and phoneme differences. The eigenvoice analysis in RC can provide better discrimination ability between different speakers. The experimental results on speaker identification of 75 males show that, when enrolment data is sparse, RC-MES provides significant improvement over GMM, and the number of eigenvoices in RC-MES is fewer than that in common eigenspace.

## 1 Introduction

Speaker recognition is one of the most flexible approaches in biometric recognition field. One key issue in speaker recognition is the speaker modeling technique. Gaussian Mixture Models (GMMs) [1] might be the most successful one, but this data-driven approach depends entirely on training data so that the recognition performance will deteriorate drastically when training data is sparse [2][3]. In practice, for some low-security tasks, clients might be impatient if the enrolment procedure extends 5 seconds. Therefore new modeling approach is needed for speaker recognition with sparse training data.

The shortcoming of GMMs consists in the limitation of prior knowledge. One possible way to solve this problem is adopting speaker adaptation techniques [4]. Thyes [2] proposed an eigenvoices approach that client and test speaker models are confined to a low-dimensional linear subspace obtained previously from a different set of training data. This approach ignores the phoneme differences that are integrated in speech data, which will possibly influence the discrimination ability of Eigenspace. In other words, the recognition performance will be further improved if every tester speaks the same utterances.

S.Z. Li et al. (Eds.): Sinobiometrics 2004, LNCS 3338, pp. 610–616, 2004.

This paper proposed a new approach named Multi-EigenSpace modeling based on Regression Class (RC-MES). We employ the concept of Regression Class (RC) from Maximum Likelihood Linear Regression (MLLR) approach used in speaker adaptation [5]. The traditional eigenspace is separated into several sub eigenspaces according to the phoneme differences. Then the Eigenvoices analysis is carried out in each subspace. In the experiments of speaker identification on 75 males, the new RC-MES technique is shown to provide significant performance improvements over GMMs when enrolment data is sparse, and the number of eigenvoices in RC-MES is fewer than that in common eigenspace.

In the next section we briefly review the eigenvoices approach and the regression class idea used in MLLR. We next describe the RC-MES approach and then deduce the parameters estimation formulas. This is followed by a description of the experiment data, design and results.

## 2 Eigenvoices and Regression Class

The eigenvoice approach [2] constrains the adapted model to be a linear combination of a small number of basis vectors obtained offline from a set of reference speakers. The modeling procedure contains two parts, offline part and online part.

In offline part, firstly a reference set of $n$ well-trained speaker-dependent (SD) models and a speaker-independent (SI) model are built on a large speech database. From each of the SD models, a "supervector" that contains the means of the Gaussian components in GMMs is extracted, noting that the number $D$ of extracted parameters and the order must be the same for all speakers. Then a dimen-sionality reduction technique (DRT) such as principle component analysis (PCA) is applied to the $n$ supervectors to get $R$ eigenvectors, namely "eigenvoices". Those Eigenvoices are orthogonal to span the eigenspace. The computationally intensive SD training and DRT steps are carried out offline before recognition begins.

In online part, each new speaker $S$ is represented by a point in eigenspace, and his supervector is assume to be a linear combination of the eigenvoices:

$$P = e(0) + w(1) \cdot e(1) + \cdots + w(R) \cdot e(R) \tag{1}$$

where $e(0), e(1), \cdots, e(R)$ are the eigenvoices, $w(1), \cdots, w(R)$ are the corresponding weights. Thus the modeling problem for the new speaker is to estimate the weight vector.

The eigenvoices can be thought as the basis vectors that correspond to the maximum-variance directions in the original speaker space. However, in speaker recognition, the differences between two utterances relate to not only the differences between speakers, but also the corresponding phoneme differences. When enrolment time is limited and the training context is unrestricted, arbitrarily building a single eigenspace will confuse these two kinds of differences. In ideal situation, to comparing utterances from different speakers with the same context will emphasize the speaker differences. Therefore, we decide to restrict the eigenspace to certain phoneme level by using RC approach referred in MLLR [5].

Leggetter has introduced RC idea in the MLLR approach, which proposed a feasible way to adapt those models that no corresponding adaptation data is available.

MLLR adapts the mean vectors of continuous density HMM's by multiplying the mean vector for the initial model with a transformation matrix:

$$\hat{\vec{\mu}} = W_s \cdot \vec{\xi}_s \tag{2}$$

where $W_s$ is the transformation matrix for mixture component $s$, $\hat{\vec{\mu}}$ is the adapted mean vector, $\xi_s$ is the extended mean vector for mixture component $s$

$$\vec{\xi}_s = \left[w, \mu_{s1}, \cdots, \mu_{sn}\right]' = [w, \vec{\mu}_s]' \tag{3}$$

where $\vec{\mu}_s$ is the original mean vector, $w$ is an offset term and n is the number of features. A RC is a set of mixture components for which it is assumed that the same transformation matrix may be used for all components in the RC.

By tying mixture components the main object is to tie components that are assumed to undertake a somewhat similar transformation. That is to say, inside the RC, the components are very similar. So to calculate the Eigen-voices in RC level might be a possible way to separate the speaker differences from phoneme differences.

The number of RCs and the clustering of Gaussian mixtures are the essential problems of MLLR. Theoretically speaking, the division of RCs can be a single globe RC that contains all components or a tiny RC that each component belongs to a different RC. In practical, the number of RCs should be defined according to the data quantity available during enrollment. The base rule is to make sure that each RC has enough adaptation data to estimate the transformation matrix. Commonly, a RC tree is built according to some distance measure between Gaussian components beforehand, where the root node corresponds to the globe RC and the leave node corresponds to the tiny RC.

## 3 RC-MES Approach

In RC-MES, the eigenvoices and the RC idea are integrated by adapting new speaker model in subeigenspace based on RC. The modeling steps are as follows:

**Offline Steps:**

(1) Building SI model (in terms of GMM) for each phoneme based on a large speech corpus such as TIMIT database [9]. Then a RC tree is built using the divergence measure [6] between all Gaussian components as the distance measure.

(2) Determining the division of the RC tree according to the adaptation material available during enrolment, while keeping sufficient adaptation data for each RC.

(3) Based on this division (assume all components are divided into $S$ RCs), rebuilding a new GMM inside each class to get $S$ SI models of RC level ($SI_{RC}$).

(4) For each of $R$ reference speakers, training $S$ SD models of RC level ($SD_{RC}$). For example, with $S$ $SI_{RC}$, for speaker $S_i$, distributing his feature vectors into $S$ classes based on the maximum likelihood rule. Then in each RC, we adapt the $SI_{RC}$ to $S_i$ dependent model via MAP using the adapting data belongs to that class. Now we have $R \times S$ $SD_{RC}$ .

(5) In each RC, with the $R$ SD models and one SI model, we can calculate $k+1$ eigenvoices $(e_i(0), e_i(1), \cdots, e_i(k)), i = 1, \cdots, S$ .

**Online Steps:**

(1) Obtaining the enrolment data of a new speaker and executing feature extraction.

(2) With $S$ $SI_{RC}$, separating the enrolment feature vectors into $S$ RCs based on the maximum likelihood (ML) rule.

(3) Inside of each RC, e.g. class $j$, according to eigenvoices $(e_j(0), e_j(1), \cdots, e_j(k))$ and adaptation data of the class, the weight vector $(w_j(1), \cdots, w_j(k))$ is estimated. Then for all RCs, we have $S$ groups of weight vectors $(w_i(1), \cdots, w_i(k)), i = 1, \cdots, S$ .

(4) In each RC, iterating the estimation process of the weight vector until likelihood score reaches its maximum. Then constructing the supervector of the RC level for new speaker to build his or her $SD_{RC}$.

(5) Integrating all $SD_{RC}$ of the new speaker to build the final SD GMM.

In the above procedure, we assume that the amount of enrolment data available is known in advance, and if it is not, all possible divisions of RC must be defined in offline steps and the final division is built in dynamic manner [5] according to the data available.

## 4 Parameters Estimation of RC-MES

In RC-MES, the model adaptation for a new speaker is actually the estimation of weights of eigenvoices $w_s(k), k = 0, \cdots, K; s = 1, \cdots, S$ ($k$ is the index of eigenvoice, $s$ is the index of RC, $w_s(0) = 1$) and the estimation of weights and covariance matrixes of GMMs. Since the eigenvoices of RC-MES are dispersed into RCs, the estimation is carried out in each regression class. The eigenvoice of RC $s$ can be written as

$$e^s(j) = \left[e_0^s(j)^T, e_1^s(j)^T, e_2^s(j)^T, \cdots, e_m^s(j)^T, \cdots\right], j = 0, 1, \cdots, K \tag{4}$$

where $e_m^s(j)$ is the means vector of $m$-th component in RC $s$.

Firstly, for each observation from the new speaker, one needs to calculate the likelihood score for each $SI_{RC}$ well trained in offline steps and to deliver the observations to each RC. Then we obtain a group of observation sets, $O^{(1)}, O^{(2)}, \cdots, O^{(S)}$, each consisting of an individual number of observation vectors. That is

$$O = \bigcup_{s=1 \cdots S} O^{(s)}, o_t \in O^{(s)} \quad when \ \ s = \arg\max_{j=1 \cdots S} \left(p\left(o_t \middle| SI_{RC}(j)\right)\right) \tag{5}$$

where $p(\bullet)$ is the observation probability of GMM, $o_t$ is the observation vector at time $t$.

The estimation of weight vectors of eigenvoices in each RC is as same as the method proposed in [7]. We also use a maximum-likelihood estimator called maximum likelihood Eigen-decomposition (MLED) to derive the estimation formula of $w(j)$ in RC-MES. In each RC, for $i = 0, \cdots, K$, the weights of eigenvoices are iterative re-estimated using

$$\sum_{t=1}^{T}\sum_{m=1}^{M} r^{(m)}(t)\cdot\left[e_m^s(i)\right]'\cdot C_m^{-1}\cdot o_t^{(s)} = \sum_{t=1}^{T}\sum_{m=1}^{M} r^{(m)}(t)\cdot\left\{\left[\sum_{k=0}^{K} w_s(k)\cdot e_m^s(k)\right]'\cdot C_m^{-1}\cdot e_m^s(i)\right\} \quad (6)$$

where $o_t^{(s)}$ is observation vector at time *t*, which belongs to RC *s*, $C_m^{-1}$ is the inverse of covariance matrix of *m*-th component in RC *s*, $r^{(m)}(t)$ is the occupation probability

$$r^{(m)}(t) = P\left(i_t = m \middle| o_t^{(s)}, \lambda_s\right) = p_m^s \cdot b_m^s\left(o_t^s\right) \Big/ \sum_{k=1}^{M} p_k^s \cdot b_k^s\left(o_t^s\right) \quad (7)$$

where $p_m^s$ and $b_m^s()$ are the weight and pdf of *m*-th component in RC *s*.

In eq. (5) there are *K*+1 equations to solve for the *K*+1 unknown weights ($w_s(i)$ values) of RC *s*. The new model thus obtained yields new values for the occupation probabilities $r^{(m)}(t)$; this estimation process can be iterated until converge. The same estimation procedures are executed in other RCs. Note that in each RC, the number of Gaussian components is rather small, therefore the computation cost is similar with the traditional eigenspace.

After the estimation of weights of eigenvoices, the supervector of the new speaker is obtained, so does the means of each Gaussian component in each RC. Then the components in each RC are bound together to build the GMM for the new speaker, the corresponding covariance matrixes come directly from $SI_{RC}$. The component weights are normalized and then re-estimated once using EM algorithm.

## 5 Speaker Identification Experiments

The database used in the identification experiments consists utterances collected from 75 males. The reading contents are selected from TIMIT database and recorded with 8kHz sampling frequency and 8-bit quantification. Each speaker has about 27s speech data after silence removing, using adaptive energy thresholds. The feature extraction includes the pre-emphasis and short time analysis using hamming window, then Mel-Frequecy Cepstral Coefficients (MFCC), ΔMFCC and ΔΔMFCC are calculated as feature vector. The corresponding parameters are list in Table1.

**Table 1.** Parameters used for feature extraction

| Para. | Form |
|---|---|
| Pre-emphasis | $1-0.97z^{-1}$ |
| Window type | Hamming |
| Frame length | 25ms |
| Frame shift | 10ms |
| Features | 12MFCC+12ΔMFCC+12$\Delta^2$MFCC ($c_0$ removed) |

The building procedures of eigenvoices and RC are as follows:

(1) Selecting 100 males from TIMIT database and extracting feature vectors;

(2) Training GMM with 6 components for each phoneme using HTK toolbox [8].

(3) Because the adaptation data is already known, we don't build the whole regression class tree but divide the GMMs into 10 RCs instead.

(4) Re-normalized the Gaussian weights inside each RC to build $SI_{RC}$.

(5) With these $SI_{RC}$, separating the data from 100 males into each RC according to maximum likelihood rule. Then building SD GMM with 3 components in each RC for each of the 100 speakers, i.e. each speaker has a GMM with 30 components.

(6) Calculating eigenvoices in each RC.

During online steps, the speech features from experiment database are separated into each RCs. Then the MELD algorithm is used to estimate the weight vectors of Eigenvoices in each RC. Finally after the binding, each new speaker obtains his GMM.

Table 2 illustrates the comparison results of GMM, common eigenspace and RC-MES. It is clear that when training data is sparse, the performance of GMM is deteriorating drastically. The recognition rate of GMM with 30 components is only 47.2%, even lower than GMM with 10 components. On the other hand, eigenspace approaches all provide better performances, which proves that using prior knowledge can overcome the shortcoming of GMM. RC-MES provides the best performance and the result of 40 eigenvoices is close to that of 65 eigenvoices in common eigenspace, which indicates that by separating the speaker differences from phoneme differences, the discriminate ability of eigenspace is enhanced.

**Table 2.** Recognition Rate (RR) using GMM vs. Common EigenSpace (CES) vs. RC-MES

| Train time | Test time | GMM(%) | | CES(%) | | RC-MES(%) | |
|---|---|---|---|---|---|---|---|
| | | RR | Gaussians | RR | Eigen-voices | RR | Eigen-voices |
| 20s | 5s | 85.2 | 30 | 93.2 | 65 | 96.1 | 65 |
| | | 80.3 | 10 | 88.5 | 40 | 92.9 | 40 |
| 10s | 5s | 47.2 | 30 | 90.8 | 65 | 95.2 | 65 |
| | | 79.0 | 10 | 86.7 | 40 | 90.5 | 40 |

## 6 Conclusions

This paper has presented a new speaker modeling technique called Multi-Eigen-Space technique based on Regression Class (RC-MES), which integrates the common eigenspace technique and the regression class idea of MLLR. This technique provides a better solution for speaker recognition application where the training and enrolment speech is limited. This technique employs the prior knowledge about speaker differences and remedies the shortcoming of common eigenspace that confuses speaker differences and

phoneme differences. The experimental results on speaker identification show that RC-MES provides significant improvement over the GMM approach, and the number of eigenvoices in RC-MES is fewer than that in common eigenspace. Future work will focus on noise corrupted speech and microphone distortion situations.

## Acknowledgements

This paper is supported by Doctoral innovation Foundation of Northwestern Polytechnical University.

## References

1. D. A. Reynolds, "Speaker identification and verification using Gaussian mixture speaker models", Speech Communication, Vol. 17, Issues 1-2, pp. 91-108, August 1995.
2. O. Thyes, R. Kuhn, P. Nguyen, J. –C. Junqua, "Speaker identification and verification using eigenvoices", ICSLP2000, Beijing-China, Vol.2, pp. 242~246, Oct. 2000.
3. N. J. –C. Wang, W. –H. Tsai, L.–S. Lee, "Eigen-MLLR coefficients as new feature parameters for speaker identification", Eurospeech, Vol. 2, pp. 1385-1388, 2001.
4. C. Tadj, M. Gabrea et al, " Towards robustness in speaker verification: enhancement and adapataion", The 2002 45th Midwest Symposium on Circuits and Systems, Vol. 3, pp. 320-323, Aug. 2002.
5. C. J. Leggetter, P. C. Woodland, "Maximum likelihood linear regression for speaker adaptation of Continuous Density Hidden Markov Models", Computer Speech and Language, Vol. 9, pp. 171-185, 1995.
6. J P Campbell, JR. "Speaker recognition: a tutorial", Proceedings of the IEEE, Vol. 85(9), Sept. 1997.
7. R. Kuhn, J-C Junqua, P. Nguyen, N. Niedzielski, "Rapid speaker adaptation in Eigenvoice space. IEEE Trans", On Speech and Audio Processing. Vol.8 (6), pp. 695-706, Nov. 2000.
8. S. J. Young, D. Kershaw, J. Odell, and P. Woodland: The HTK Book (for HTK Version 3.0), Http://htk.eng.cam.ac.uk/docs.shtml, 2000.
9. J. Garofolo, et al. "DARPA TIMIT Acoustic-Phonetic Continuous Speech Corpus CD-ROM", National Institute of Standards and Technology, 1993.