# COMPACT ACOUSTIC MODELING BASED ON ACOUSTIC MANIFOLD USING A MIXTURE OF FACTOR ANALYZERS

*A. Wen-Lin Zhang, C. Bi-Cheng Li**

Zhengzhou Information Science and Technology Institute
Zhengzhou 450002, China
(e-mail: zwlin_2004@163.com, lbclm@163.com)

*B. Wei-Qiang Zhang*

Tsinghua University
Department of Electronic Engineering
Beijing 100084, China
(e-mail: wqzhang@tsinghua.edu.cn)

## ABSTRACT

A compact acoustic model for speech recognition is proposed based on nonlinear manifold modeling of the acoustic feature space. Acoustic features of the speech signal is assumed to form a low-dimensional manifold, which is modeled by a mixture of factor analyzers. Each factor analyzer describes a local area of the manifold using a low-dimensional linear model. For an HMM-based speech recognition system, observations of a particular state are constrained to be located on part of the manifold, which may cover several factor analyzers. For each tied-state, a sparse weight vector is obtained through an iteration shrinkage algorithm, in which the sparseness is determined automatically by the training data. For each nonzero component of the weight vector, a low-dimensional factor is estimated for the corresponding factor model according to the maximum a posteriori (MAP) criterion, resulting in a compact state model. Experimental results show that compared with the conventional HMM-GMM system and the SGMM system, the new method not only contains fewer parameters, but also yields better recognition results.

***Index Terms***— Acoustic model, nonlinear manifold, mixture of factor analyzers, subspace Gaussian mixture model.

## 1. INTRODUCTION

Acoustic modeling is of great importance for speech recognition. Conventional continuous speech recognition systems are based on hidden Markov models (HMM) that represent monophone or triphone units. The state level probabilities are estimated using the Gaussian mixture models (GMMs). To deal with data sparsity, state tying method [1] is usually adopted to reduce the model size and enhance recognition speed. Recently, various basis approaches, where the model parameters are derived from sets of basis vectors or functions, are emerging to further reduce the model parameters for robust estimation. For example, in semi-continuous hidden Markov models (SC-HMMs) [2], the basis is constructed by a set of continuous Gaussian distributions. The Gaussian means and variances are shared among all the tied states and only the weights differ. In subspace Gaussian mixture model (SGMM) [3], all the states share a common structure but the means and mixture weights are allowed to vary in a subspace of the full parameter space, controlled by a global mapping from a vector space to the space of GMM parameters. Both SC-HMMs and SGMM can be derived from the canonical state model (CSM) [4] framework, where every context-dependent state is obtained by transformations of a finite set of canonical states.

In this paper, a new compact acoustic modeling method is proposed based on an manifold-based compressive sensing method. All acoustic features are assumed to belong to a nonlinear manifold, which is modeled by a mixture of factor analyzers (MFA) [5, 6]. The MFA can approximate a nonlinear manifold by a set of low-dimensional factor models. For each low-dimensional factor model, the mean vector corresponds to a point sampled from the manifold and the columns of the factor loading matrix roughly span the local tangent space at that sample point. For each observation, the unobserved local factor is its coordinate in the local tangent space corresponding to that factor model. In this paper, the acoustic features belonging to each tied state of the HMM-based speech recognition system is assumed to locate in a local part of the manifold, which may be across several factor models. So each tied state can be modeled by selecting a few factor models from the mixture components of the MFA and estimating the weight and local factor for each factor model.

This method can also be derived from the CSM framework, where a mixture of factor analyzers is used as the canonical state and each tied state model is compressive sensed on that nonlinear manifold. It is different from the SGMM in that for each state the local factors corresponding to different mixtures are estimated independently and the weight vector is subjected to a sparse constraint instead of a subspace constraint.

All the model parameters of the new method can be

*This work was supported in part by the National Natural Science Foundation of China (No. 60872142, No. 61175017 and No. 61005019).

trained in a iterative way according to the maximum likelihood criterion. For weight vector, an iterative shrinkage method is proposed to obtain a sparse solution, where the degree of sparsity is determined automatically by the training data. Because the MFA can be viewed as a degraded Gaussian mixture model, the new method falls into the standard HMM-GMM framework and all the standard techniques can also be applied.

In the next section, modeling of the acoustic manifold using a mixture of factor analyzers is presented. In Section 3, Bayesian estimation of the state model is described, and comparisons with previous basis methods are given. The training method for various model parameters are summarized in Section 4. In Section 5, we present experiments on the acoustic modeling of a continuous speech recognition system using the DARPA Resource Management Continuous Speech Corpus (RM). Finally, conclusions are given in Section 6

## 2. ACOUSTIC MANIFOLD MODELING USING A MIXTURE OF FACTOR ANALYZERS

Many researchers have shown that speech sounds may exist on a low-dimensional manifold nonlinearly embedded in high dimensional space [7, 8]. The mixture of factor analyzers (MFA) can approximate such a nonlinear manifold using many low-dimensional linear factor models [6]. Each low-dimensional linear factor model describes the distribution of the data in a local area of the manifold.

Let $\boldsymbol{o}_t$ denotes an acoustic feature vector at time $t$, the MFA model is a probabilistic generative model which obeys the following mixture distribution:

$$\begin{cases} p\left(\boldsymbol{o}_t \mid \{\boldsymbol{y}_i\}_{i=1}^{I}\right) = \sum_{i=1}^{I} w_i \mathcal{N}\left(\boldsymbol{o}_t \mid \boldsymbol{M}_i \boldsymbol{y}_i + \overline{\boldsymbol{\mu}}_i, \boldsymbol{\Sigma}_i\right) & \text{(1a)} \\ p\left(\boldsymbol{y}_i\right) = \mathcal{N}\left(\boldsymbol{y}_i \mid \boldsymbol{0}, \boldsymbol{I}\right), \quad i = 1, 2, \cdots, I & \text{(1b)} \end{cases}$$

where $\mathcal{N}\left(\cdot \mid \boldsymbol{\mu}, \boldsymbol{\Sigma}\right)$ denotes a Gaussian distribution with mean $\boldsymbol{\mu}$ and covariance matrix $\boldsymbol{\Sigma}$. $I$ is the number of mixtures. Each mixture is a factor analyzer with a weighting factor of $w_i$ ($w_i > 0$ and $\sum_i w_i = 1$). For the $i$th factor analyzer, $\overline{\boldsymbol{\mu}}_i$ is the mean vector, $\boldsymbol{M}_i$ denotes the factor loading matrix, and $\boldsymbol{y}_i$ is the latent factor which is Gaussian distributed with zero mean and unit diagonal covariance matrix (Equation (1b)), $\boldsymbol{\Sigma}_i$ is the conditional covariance matrix given $\boldsymbol{y}_i$, which is a diagonal matrix in the standard MFA model [5].

Let $D$ denotes the dimension of the observation data, $D_i$ denotes the dimension of $i$th factor $\boldsymbol{y}_i$, $D_i < D$. Each factor analyzer is a low-dimensional signal model with the factor $\boldsymbol{y}_i$ as the latent variable. If we marginalize over the latent variable, each factor analyzer obeys a Gaussian distribution with mean $\overline{\boldsymbol{\mu}}_i$ and covariance matrix $\boldsymbol{M}_i \boldsymbol{M}_i^T + \boldsymbol{\Sigma}_i$. So an MFA model is in its intrinsic a degraded Gaussian mixture model, which can concurrently performs clustering and, within each cluster, local dimensionality reduction.

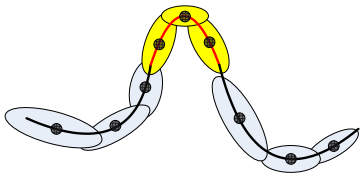

**Fig. 1**. Approximation of a nonlinear manifold using many local low-dimensional linear models

It is well-known that locally, a nonlinear manifold can be well approximated by its tangent plane, with the quality of this approximation depending on the local curvature of the manifold. Therefore, from a geometrical point of view, an MFA model as in (1) may be considered a candidate for manifold-modeled data, where the mean vectors $\overline{\boldsymbol{\mu}}_i$ correspond to points sampled from the manifold, the columns of $\boldsymbol{M}_i$ roughly span the $D_i$-dimensional local tangent spaces, the covariance matrix $\boldsymbol{\Sigma}_i$ describes the manifold curvature, and the weight $w_i$ reflect the probability of observation data fall into this local area. The above geometrical interpretation can be illustrated by Figure1. In Figure1, the black bold curve denotes a nonlinear manifold, which cannot be modeled directly by any linear models. But when we look at a tiny area on the manifold, it can be approximated by a tangent plane of any sample point on it. The distribution of the data on that tangent plane can be modeled by a low-dimensional Gaussian model, which is denoted by the tiny ellipse in Figure1. The center of each ellipse (denoted by the solid black point) is the mean of the Gaussian distribution, which corresponds to the sample point of each local area. The direction of the major axes forms a set of basis vectors for the tangent space and the size of the ellipse is proportional to the manifold curvature at that sample point.

In this paper, we use MFA to model the nonlinear manifold of the acoustic feature space. This can be trained using the Expectation Maximization (EM) algorithm [5] in an unsupervised manner. At the beginning of the EM algorithm, we must choose the underlining local dimensions $D_i$ for each factor analyzer and obtain an initial model. In our implementation, we start with a traditional HMM-GMM system and obtain a big GMM by clustering the Gaussians in it to a predefined number of clusters $I$. The clustering procedure is similar to that of the training of the universal background model (UBM) in [3]. Then we do principal component analysis (PCA) on the covariance matrix of each Gaussian component $i$ and sort the eigenvalues in a descent order as $\lambda_{i1}, \lambda_{i_2}, \cdots, \lambda_{iD}$. We select $D_i$ as the number of top eigenvalues which contribute to 90% cumulative contribution rate of all eigenvalues. We initialize $\Sigma_i$ and $\boldsymbol{M}_i$ with that of a probabilistic principal component analyzer (PPCA) using a closed solution proposed by [9]. $w_i$ and $\overline{\boldsymbol{\mu}}_i$ are set to the cor-

responding mixture weight and mean vector of the Gaussian component $i$ in the UBM respectively. Then we run a few iterations of the EM algorithm on the whole training data set in an unsupervised manner to obtain the MFA for the acoustic feature space.

## 3. STATE MODELING BASED ON THE ACOUSTIC MANIFOLD

### 3.1. Bayesian State Modeling based on MFA

Once we get the nonlinear manifold of the acoustic space, the state-dependent model can be constrained on that manifold. In the HMM-based system, each state model describes the feature distribution of part of particular phone, so it should cover a local part of the manifold, which could be in itself across several factor models. This can be illustrated intuitively by Figure 1, where the red arc corresponds to a state, which covers three local factor models.

For state $j$, the mathematical formulation of the state model is given as following:

$$\begin{cases} p\left(\boldsymbol{o}(t) \mid j\right) = \sum_{i=1}^{I} w_{ji} \mathcal{N}\left(\boldsymbol{o}(t) \mid \boldsymbol{\mu}_{ji}, \boldsymbol{\Sigma}_i\right) & \text{(2a)} \\ \boldsymbol{\mu}_{ji} = \boldsymbol{M}_i \boldsymbol{y}_{ji} + \overline{\boldsymbol{\mu}}_i & \text{(2b)} \\ p\left(\boldsymbol{y}_{ji}\right) = \mathcal{N}\left(\boldsymbol{y}_{ji} \mid \boldsymbol{0}, \boldsymbol{I}\right), \quad i = 1, 2, \cdots, I & \text{(2c)} \end{cases}$$

where $\boldsymbol{y}_{ji}$ is the state-dependent unobserved factor for factor model $i$ specific to state $j$, which follows a standard Gaussian distribution (Equation (2c)). Geometrically, $\boldsymbol{y}_{ji}$ can be viewed as the local coordinate of the state-dependent mean vector $\boldsymbol{\mu}_{ji}$ in the subspace spanned by columns of $\boldsymbol{M}_i$ and centered at $\overline{\boldsymbol{\mu}}_i$.

$w_{ji}$ is the state-specific weight which gives the probability of the observations of state $j$ falling into the local area corresponding to factor model $i$. Defining a weight vector $\boldsymbol{w}_j = [w_{j1}, w_{j2}, \cdots, w_{jI}]^T$, from the above description, $\boldsymbol{w}_j$ should be a sparse vector with most elements being zero. Each nonzero element of $\boldsymbol{w}_j$ corresponds to a local area which state $j$ covers.

In Equation (2), the covariance matrix $\boldsymbol{\Sigma}_i$ is shared among all states and can be re-estimated as a full covariance matrix after $\boldsymbol{y}_{ji}$ and $w_{ji}$ are obtained. The state-dependent factor $\boldsymbol{y}_{ji}$ can be estimated using the maximum a posteriori criterion. The weight vector $\boldsymbol{w}_j$ could be estimated in a maximum likelihood manner subject to the sparse constraint. Training procedure of various model parameters will be presented in Section 4.

### 3.2. Comparison with previous methods

The state model of Equation (2) is simple and have close relationships with previous methods, such as SC-HMM and SGMM.

First of all, it falls into the general framework of CSM ( [4]). Here the canonical state plays the role of prior distribution of the acoustic features in the acoustic space, which is modeled by a nonlinear manifold. For each context dependent state, the mean and weight are re-estimated subject to the local subspace and sparse constraints respectively.

As in SC-HMM, a Gaussian pool is obtained using all the training data in MFA-based method. However, in the MFA-based method, each Gaussian is degenerated to a low-dimensional factor-analyzed model, which coincides with the low-dimensional manifold assumption. This underlining low-dimension assumption not only simplifies the model, but also brings some robustness of the estimated model. Another difference lies in that besides the weight vectors, the component means are also adapted for each context-dependent state.

The formulation of Equation (2) looks very similar to that of SGMM, where for state $j$ the probabilistic distribution can be written as

$$\begin{cases} p\left(\boldsymbol{o}_t \mid j\right) = \sum_{i=1}^{I} w_{ji} \mathcal{N}\left(\boldsymbol{o}_t \mid \boldsymbol{\mu}_{ji}, \boldsymbol{\Sigma}_i\right) & \text{(3a)} \\ \boldsymbol{\mu}_{ji} = \boldsymbol{M}_i \boldsymbol{v}_j & \text{(3b)} \\ w_{ji} = \dfrac{\exp(\boldsymbol{w}_i^T \boldsymbol{v}_j)}{\sum_{i'=1}^{I} \exp(\boldsymbol{w}_{i'}^T \boldsymbol{v}_j)} & \text{(3c)} \end{cases}$$

Comparing Equation (2) and (3), we can conclude that the differences of MFA-based state model and the SGMM lie in the following three aspects:

- SGMM assumes a global linear subspace of the model parameters, while the MFA-based method assumes a nonlinear manifold of the acoustic space which is modeled by multiple locally linear models. In MFA-based state model, for state $j$ there are $N_j$ ($N_j = ||\boldsymbol{w}_j||_0 << I$) mixtures, each contains a latent vector $\boldsymbol{y}_{ji}$. However, in SGMM there are $I$ mixture components for each state and all components share the same phone vector $\boldsymbol{v}_j$. To increase the model capacity, sub-states are usually used in SGMM. For an SGMM which contains $M_j$ sub-states in state $j$, there are $M_j$ phone vectors and $M_j \times I$ mixture components, which makes it computationally very expensive in recognition time. To accelerate the recognition speed, a per-frame Gaussian selection process is need for each observation $\boldsymbol{o}_t$ before the likelihood is computed [3]. The MFA-based state model is much more simpler, no sub-state splitting and Gaussian pre-selection process is needed.

- In standard SGMM, the coordinate vector $\boldsymbol{v}_j$ is estimated freely, that is, no prior information is applied. However, in our new method, each local factor model is centered at different sample location, and the coordinate vector $\boldsymbol{y}_{ji}$ has a natural normal prior distribution, from which we can derive a MAP-based estima-

tion, which is usually more robust than an ML-based estimation method.

- In SGMM, the subspace of the logarithmic weight vector and that of the Gaussian means are sharing the same coordinate vector $\boldsymbol{v}_j$, which is a mathematical trick, making the estimation of the $\boldsymbol{v}_j$ very complicated. Some mathematic approximation must be applied to obtain a closed form updating formula [3]. However, in the MFA-based state model, no subspace is assumed for the weight vector $\boldsymbol{w}_j$, only a sparse constraint is applied, which makes it is much easier for both the estimation of the local coordinate vector $\boldsymbol{y}_{ji}$ and that of the weight vector $\boldsymbol{w}_j$.

## 4. ESTIMATION OF THE MODEL PARAMETERS

The parameters of the proposed MFA-based acoustic model can be categorized into two sets: one set $\boldsymbol{\Lambda}_1$ contains parameters that are shared among all context-dependent tied state, that is $\{\overline{\boldsymbol{\mu}}_i, \boldsymbol{M}_i, \boldsymbol{\Sigma}_i\}_{i=1}^{I}$, which is the mean vector, factor loading matrix and conditional covariance matrix for each factor model; the other set $\boldsymbol{\Lambda}_2$ contains parameters that are state-specific, that is $\left\{\boldsymbol{w}_j, \{\boldsymbol{y}_{ji}\}_{i\in\boldsymbol{I}_j}\right\}_{j=1}^{J}$, where $J$ is the number of different tied states in the system and $\boldsymbol{I}_j$ is the set of indices of nonzero components in $\boldsymbol{w}_j$. The whole training procedure can be summarized as following:

**Algorithm 1** Training procedure of the MFA-based acoustic model

1: Train the background MFA. Perform force alignment of the training data using a baseline HMM-GMM system.
2: Set $\boldsymbol{I}_j = \{1, 2, \cdots, I\}$, $w_{ji} = \frac{1}{I}$, $\boldsymbol{y}_{ji} = \mathbf{0}$ for $i \in \boldsymbol{I}_j$ and $j = 1, 2, \cdots, J$.
3: **for** $k = 0$ to $K$ **do**
4: Update the state-dependent factors $\{\boldsymbol{y}_{ji}\}_{i\in I_j}$ for each state $j$.
5: Update the state-independent factor loading matrices $\boldsymbol{M}_i$ for each factor $i$.
6: Update the state-independent means and covariance matrices $\{\overline{\boldsymbol{\mu}}_i, \boldsymbol{\Sigma}_i\}_{i=1}^{I}$.
7: Update the state-dependent weight vector $\boldsymbol{w}_j$ for each state $j$.
8: If $|\boldsymbol{I}_j| > minC$, shrink $\boldsymbol{w}_j$ automatically and update $\boldsymbol{I}_j$ for each state $j$.
9: **end for**

In Algorithm 1, $K$ is a predefined number of iterations (i.e. $K = 20$). In Step 1, the MFA describing the acoustic manifold is trained using the method presented in Section 2. In Step 2, we initialize each state-dependent model using a flat start method. From Step 3 to Step 9, we update each set of parameters in sequence for a predefined number ($K$) of EM iterations.

For each iteration, in Step 4 and 5, the state-dependent factors $\boldsymbol{y}_{ji}$ and state-independent factor loading matrices $\boldsymbol{M}_i$ are updated in sequence using the MAP and ML criteria respectively. The updating formulas are similar to that of the state vectors and model projection matrices in SGMM [3]. A more simpler formula can be derived for $\boldsymbol{y}_{ji}$ as a consequence of decoupling of the weight vector from sharing the same coordinate vector under a weight subspace constraint. In Step 6, we update the state-independent means and covariance matrices using almost the same formula as in SGMM [3].

In Step 7, we update $w_{ji}$ for each nonzero weight index $i \in \boldsymbol{I}_j$ of state $j$ using a simple EM algorithm. Then, in Step 8, for each state $j$, if the number of nonzero components $|\boldsymbol{I}_j|$ is above a predefined threshold $minC$, we shrink the weight vector $\boldsymbol{w}_j$ automatically according to the sparse constraint. We use a heuristic weight shrinkage strategy as Algorithm 2.

**Algorithm 2** A heuristic weight shrinkage strategy for $\boldsymbol{w}_j$

1: Sort $w_{j1}, w_{j2}, \cdots, w_{jI}$ in a descent order as $w'_{j1}$, $w'_{j2}$, $\cdots$, $w'_{jI}$.
2: Calculate the cumulative contribution rate of each component $i$ as $s_i = \sum_{k=1}^{i} w'_{jk}$.
3: Find $n = \arg\min_k\{k : s_k \geq 0.9\}$, and set $\tau_j = w'_{jn}$.
4: Shrink the weight vector $\boldsymbol{w}_j$ according to $w_{ji} \leftarrow [w_{ji} - \tau_j]_+$.

In Algorithm 2, $[w_{ji} - \tau_j]_+ = \max\{w_{ji} - \tau_j, 0\}$. Here we prune the weight $w_{ji}$ according to a threshold $\tau_j$ determined by the 90% cumulative contribution rate. When we plug Algorithm 2 to Step 8 of Algorithm 1, we obtain a iteration shrinkage process for updating of the weight vector according to the sparse constraint. Note that the degree of sparseness is determined automatically by the training data, only a lower bound (determined by $minC$) is applied to prevent over-fitting.

Using the Maximum Mutual Information (MMI) criterion [10, 11], discriminative training of all model parameters can also be derived. In our experiments, after the maximum likelihood model is obtained, we ran three iterations of Step 4 to Step 7 of Algorithm 1 using the discriminative training method. Weight shrinkage was not performed during the discriminative training procedure. The standard MMI-based discriminative training method was applied. The derivation of the updating formulas are not presented here for brevity.

## 5. EXPERIMENTS

### 5.1. System Description

All experiments were based on the DARPA Resource Management Continuous Speech Corpus (RM) [12] and the Kaldi speech recognition toolkit [13]. The experimental settings were following Kaldi's RM "s5" recipe, where we train on the speaker independent training and development set (about

4000 utterances) and test on the six DARPA development set runs Mar and Oct'87, Feb and Oct'89, Feb'91 and Sep'92 (about 1500 utterances total).

For the acoustic frontend, we extract 13 Mel-frequency cepstral coefficients (MFCCs), apply cepstral mean and variance normalization, splice 7 frames (3 on each side of the current frame) and uses LDA to project down to 40 dimensions, together with MLLT. First of all, a conventional HMM-GMM system was trained with speaker adaptive training (with fMLLR). Then starting from the HMM-GMM baseline system, an SGMM system was trained using 400 Gaussian components in the universal background model (UBM), 41-D phonetic subspace and 40-D speaker subspace. State splitting was applied to increase the number of sub-states for large model capacity. The final number of sub-states is 7495. MMI-based discriminatively training methods were applied on the both the HMM-GMM and the SGMM systems to obtain better recognition results.

For the training of our MFA-based system, we start from the UBM used to initialize the SGMM system, and train the background MFA using the method described in Section 2. Then we run Algorithm 1 with $K = 50$, together with the weight shrinkage Algorithm 2. Finally, three iterations of the MMI-based discriminatively training method were performed to obtain a discriminatively trained system. The number of different tied states was 2036, which is the same as the SGMM system. Different from SGMM, there is no "speaker subspace" in the MFA-based system.

Note that as in Kaldi's "s5" recipe, the speaker-dependent transformation matrices from the SAT-based HMM-GMM system were applied to transform the acoustic features for both the training and testing of the SGMM system and the MFA-based system.

### 5.2. Distribution of local dimensions

One of the major differences of our MFA-based system compared with the SGMM system lies in that the parameter subspaces are distinct for different Gaussian components in the MFA-based system. Using the training procedure described in Section 2, the intrinsic dimensions ($D_i$) of the local factor analyzer (each corresponds to a Gaussian component) are different according to the 90% cumulative contribution rate criterion. Figure2 shows the histogram of the local dimensions of all factor analyzers.

In our SGMM system, the dimension of the phonetic subspace is fixed to 40. From Figure2, it can be observed that most of the local dimensions are around 26 in the MFA-based system, showing a more compact acoustic subspace.

### 5.3. Effect of weight shrinkage

Another difference of our MFA-based system compared with the SGMM system is that the weight vectors are subject to a sparse constraint other than a subspace constraint. Using the iterative shrinkage algorithm of Algorithm 2 (plugged into Step 8 of Algorithm 1), the weight vectors were getting sparser and sparser during the iterations. In our experiments, for each state, we set the minimal number of nonzero weight components ($minC$) to 10 and run 50 iterations of the weight shrinkage algorithm. After each iteration, we calculate the average number ($\overline{I}$) of nonzero weight components among all states. Figure3 shows $\overline{I}$ changing with the iteration number.

From Figure3, it can be observed that our weight shrinkage algorithm is effective, $\overline{I}$ seems to converge around 20 after 30 iterations. After 50 iterations, $\overline{I} = 18$. Figure4 gives the histogram of the numbers of nonzero components for all states. The average number of nonzero components is 18.

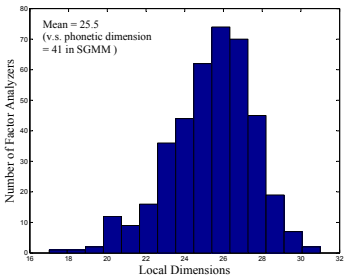


**Fig. 2**. Histogram of the local dimensions of all factor analyzers

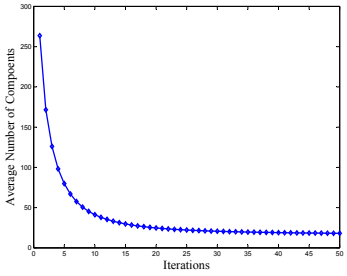


**Fig. 3**. Average number of nonzero weight components ($\overline{I}$) changing with the iteration number

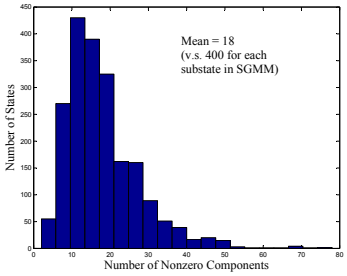


**Fig. 4**. Histogram of the numbers of nonzero components for all states

**Table 1**. Results of all testing systems (% WER)

| | WER |
|---|---|
| HMM-GMM + SAT | 1.88 |
| HMM-GMM + SAT + MMI | 1.70 |
| SGMM | 1.64 |
| SGMM + MMI | 1.54 |
| MFA-based | 1.51 |
| MFA-based + MMI | 1.36 |

In the baseline SGMM system, there are 7495 substates, each contains 400 nonzero components. Gaussian pre-selection is required to prune the Gaussians for each frame prior to the likelihood calculation. For the MFA-based system, the Gaussian components are pruned at the training time, so a more compact model is obtained and the decoder can be simplified.

### 5.4. Comparison of WERs

The recognition results were measured in average Word Error Rate (WER). Table 1 summarizes the recognition results of all testing systems. From Table 1, it can be observed that the MFA-based system outperforms both the HMM-GMM and the SGMM system in WERs, even when the two baseline systems were discriminatively trained. With MMI-based discriminative training, the MFA-based system obtained an average WER of 1.36, which is one of the best results on the RM corpus as far as we know. Statistical significance tests using the standard NIST SCTK toolkit show that compared with the SGMM+MMI system, the improvement of the MFA-based system is not significant, but that of the MFA+MMI system is significant according to the standard MP, SI and WI tests at a 5% level of significance.

## 6. CONCLUSION

In this paper, a new compact acoustic modeling method based on compressive sensing on the nonlinear acoustic manifold is proposed. The acoustic space of human speech is assumed to be a nonlinear manifold, which can be modeled by an mixture of factor analyzers (MFA). The states of an HMM-based speech recognition system are constrained to be located on the manifold, each covers several local factor models. For each state model, a sparse weight vector is estimated using an iterative shrinkage algorithm, and for each nonzero component, a local factor is estimated for the corresponding local factor model in a maximum a posteriori manner. Experimental results on the RM corpus show that the new method obtains better performance in WER than the conventional HMM-GMM system and the SGMM system with a much more compact acoustic model. Because the free parameters in the new method is very limited and the MFA can be shared among different languages, the new method is very suitable for low-resource and multilingual speech recognition. We will look at these for our future directions.

## 7. REFERENCES


[1] S. J. Young and P. C. Woodland, "The use of state tying in continuous speech recognition," in *Proc. Eurospeech*, 1993, vol. 3, pp. 2203–2206.

[2] K. Riedhammer, T. Bocklet, A. Ghoshal, and D. Povey, "Revisiting semi-continuous hidden Markov models," in *Proc. of ICASSP*, 2012, pp. 4721–4724.

[3] D. Povey, L. Burget, M. Agarwal, et al., "The subspace Gaussian mixture model – a structured model for speech recognition," *Computer Speech and Language*, vol. 25, 2011.

[4] M. J. F. Gales and K. Yu, "Canonical state models for automatic speech recognition," in *Proc. of Interspeech*, 2010, pp. 58–61.

[5] Z. Ghahramani and G. Hinton, "The EM algorithm for mixtures of factor analyzers," Tech. Rep. CRG-TR-96-1, Univ. of Toronto, Toronto, ON, Canada, 1997.

[6] L. Carin, R. G. Baraniuk, V. Cevher, D. Dunson, M. I. Jordan, G. Sapiro, and M. B. Wakin, "Learning low-dimensional signal models," *IEEE Signal Process. Mag.*, vol. 28, no. 2, pp. 39–51, Mar. 2011.

[7] R. Togneri, M. D. Alder, and Y. Attikiouzel, "Dimension and structure of the speech space," *IEE Proceedings I*, vol. 139, no. 2, pp. 123–127, 1992.

[8] A. Jansen and P. Niyogi, "Intrinsic Fourier analysis on the manifold of speech sounds," in *Proc. of ICASSP*, 2006, vol. 1.

[9] M. Tipping and C. M. Bishop, "Probabilistic principal component analysis," *Journal of the Royal Statistical Society: Series B(Statistical Methodology)*, vol. 3, no. 61, pp. 611–622, 1999.

[10] D. Povey, *Discriminative Training for Large Voculabulary Speech Recognition*, Ph.d. thesis, Cambridge University, 2004.

[11] D. Povey, D. Kanevsky, B. Kingsbury, B. Ramabhadran, G. Saon, and K. Visweswariah, "Boosted MMI for model and feature-space discriminative training," in *Proc. of ICASSP*, 2008, pp. 4057–4060.

[12] P. Price, W. M. Fisher, J. Bernstein, and D. S. Pallett, *Resource Management RM1 2.0*, Linguistic Data Consortium, Philadelphia, USA, 1993.

[13] D. Povey, A. Ghoshal, Gilles Boulianne, et al., "The Kaldi speech recognition toolkit," in *Proc. ASRU*, 2011.