A Speaker Adaptation Algorithm Using Principal Curves in Noisy Environments

Wang Jingying. Wang Zuoying

Speech Recognition Lab. Department of Electronics Engineering.
Tsinghua University, China. 100084
wangjingying02@tsinghua.org.cn

10.21437/Interspeech.2006-341

Abstract

A new speaker adaptation method of speech recognition is proposed in this paper utilizing principal curves algorithm. The key feature of this method is the construction of a transformation function based on the correlation information between observations of different acoustic states. This is an important a priori information crucial to improving system's recognition performance. Herein the relationships between the statistics information required choosing the best reconstruction of an audio speech pattern and the codebook state parameters of the new algorithm are described, and then the method is applied to a large database of continuous speech. Experiment results on large vocabulary continuous speech recognition database showed that this new method is superior to MLLR adaptation approach in noisy cases, demonstrating that the principal curves speaker adaptation algorithm successfully exploits the correlation information and improve robustness.

Index Terms: principal curves, correlation information, speaker adaptation, maximum likelihood linear regression.

1. Introduction

Robustness, roughly defined as the scope of different inputs for which more accurate results are obtainable, is an important factor to consider when characterizing a speech recognition system. Speaker adaptation techniques aim to improve a speech recognition system's robustness. In recent years, a lot of speaker adaptation methods have been proposed, including the Maximum Likelihood Linear Regression (MLLR) method[1], the Maximum A Posterior (MAP) method[2], EigenVoice(EV)[3] method, etc. The MLLR method is a transformation-based approach, which aims at using mutual correlation information of observations by transformation matrices for the model parameters that maximize the likelihood of the adaptation data. The MAP is a Bayesian adaptation approach, which adjusts codebook parameters using maximum a posteriori estimates. And the EV approach constrains the adapted model to be a linear combination of a small number of basis vectors obtained from a set of reference speakers.

In MLLR, only state cluster takes distance between states into consideration. Eigenvoice describes the variation between reference speakers. By contrast, the principal curves speaker adaptation (PCSA) method proposed here utilizes a nonlinear curve to describe relations between different acoustics states. Since observations made about a specific feature of a speech segment may be faulty, particularly if the environment is noisy, use of a curve joining all states can make the statistic information more accurate and thus improve system performance. Our goal is to improve robustness resorting to speaker adaptation method in noisy cases.

The remainder of this work is organized as follows: The basic ideas of the principal curves algorithm and a description of the Principal curves speaker adaptation algorithm are presented in Section II. Experiment results found when our method was applied to a large database of speech segments (including comparison to other speaker adaptation methods) are presented in section III. Section IV summarizes our work.

2. Principal curves speaker adaptation algorithm

2.1. Principal curves algorithm

Principal curves have been widely used in many fields, such as in feature extraction and ice flow identification in satellite imagery [4-8]. In order to describe the distribution of observations of pairs of variables, one variable is usually treated as a response variable, while the other is the so-called explanatory variable. In contrast to this, by adding a latent variable, the iterative principal curves algorithm proposed by Hastie and Werner [4] allows the two variables to be treated symmetrically.

In the following description of the principal curves iteration algorithm, vectors are denoted by bold lower case characters, e.g. $\mathbf{v}$, while bold upper case characters denote matrices, e.g. $\mathbf{A}$.

The curve $\mathbf{f}$ is called a principal curve if

$$E(\mathbf{x} \mid \lambda_{\mathbf{f}}(\mathbf{x}) = \lambda) = \mathbf{f}(\lambda) \quad (1)$$

for a.e λ.

Here λ denotes latent variable, $\mathbf{x}$ is the real sample, $\lambda_{\mathbf{f}}(\mathbf{x})$ denotes the value of λ for which $\mathbf{f}(\lambda)$ is closest to $\mathbf{x}$.

Principal curve is such a curve that every point on it is the average of all samples that are projected on λ.

2.2. Principal curves speaker adaptation

The following algorithm applies the principal curves framework to speaker adaptation (note that formula below are given in either component or matrix form). We break the algorithm into five segments (note also that only correlations of the same feature dimension for all states are considered).

For $d^{th}(d = 1, 2, \ldots, L)$ dimension component, perform step (1) through step (4):

Let $\mathbf{x}_{i,d} = (s_{i,d}, c_{i,d})$ such that

$$\begin{pmatrix} s_{i,d} \\ c_{i,d} \end{pmatrix} = \begin{pmatrix} f_1(\lambda_{i,d}) \\ f_2(\lambda_{i,d}) \end{pmatrix} + \begin{pmatrix} \varepsilon_1 \\ \varepsilon_2 \end{pmatrix} \quad (2)$$

where $s_{i,d}$ is the feature statistics, $c_{i,d}$ is the codebook mean, ε_1 and ε_2 denotes estimation error, $\mathbf{f} = (f_1, f_2)$ denotes principal curves.

(1) The iterative principal curves algorithm is initialized by the covariance matrix:

(1.a) First set the $M \times M$ covariance matrix $\mathbf{\Sigma}_d$ between different states, as

$$\mathbf{\Sigma}_d = (\Sigma_{ij,d})_{i=1,\ldots,M;\, j=1,\ldots,M} \quad (3)$$

where

$$\Sigma_{ij,d} = \frac{1}{N}\sum_{k=1}^{N}[(p_{k,i,d} - \mu_{i,d})(p_{k,j,d} - \mu_{j,d})] \quad (4)$$

$$\mu_{i,d} = \frac{1}{N}\sum_{k=1}^{N} p_{k,i,d} \quad (5)$$

Here d is the index for components (feature dimensions) $(d = 1,2,\ldots,L)$, i is the index for states (codebook and observations) $(i = 1,2,\ldots,M)$, k is the index of speakers $(k = 1,2,\ldots,N)$, $p_{k,i,d}$ denotes components of the speaker super vector, speaker super vector is composed by concatenating the mean vectors of all his\her HMM Gaussian distributions, and $\mu_{i,d}$ are the means of all the speaker super vectors.

(1.b)Carry out eigen-decomposition on $\mathbf{\Sigma}_d$:

$$\mathbf{\Sigma}_d = \mathbf{A}_d \mathbf{\Lambda} \mathbf{A}_d^T \quad (6)$$

(1.c) Calculate $\boldsymbol{\lambda}_d^{(0)}$ utilizing

$$\mathbf{s}_d = \overline{\mathbf{c}} + \mathbf{A}_d \boldsymbol{\lambda}_d^{(0)} \quad (7)$$

where

$\mathbf{s}_d = (s_{1,d}, s_{2,d}, \ldots, s_{M,d})^T$, $\overline{\mathbf{c}} = (\overline{c}_1, \overline{c}_2, \ldots, \overline{c}_M)^T$,

$\overline{c}_i = \frac{1}{L}\sum_{d=1}^{L} c_{i,d}$ $(i = 1,2,\ldots,M)$,

$\boldsymbol{\lambda}_d^{(0)} = (\lambda_{1,d}^{(0)}, \lambda_{2,d}^{(0)}, \ldots, \lambda_{M,d}^{(0)})^T$

(2) Repetition over iteration counter j (the superscript denotes iteration counter):

(2.a) Conditional expectation step: for each $\lambda_{i,d}^{(j-1)}$, obtain

$$\mathbf{f}^{(j)}(\lambda_{i,d}^{(j-1)}) = E(\mathbf{x} \mid \lambda_{i,d}^{(j-1)}) \quad (8)$$

here we estimate $\mathbf{f}^{(j)}(\lambda_{i,d}^{(j-1)})$ by average all $\mathbf{x}_{m,d}$ for which $\lambda_{m,d}^{(j-1)}$ is closest to $\lambda_{i,d}^{(j-1)}$,let Q denotes the set of these $\mathbf{x}_{m,d}$. namely:

$$\mathbf{f}^{(j)}(\lambda_{i,d}^{(j-1)}) = \sum_{m \in Q} w_{im} \mathbf{x}_{m,d} \quad (9)$$

where $w_{i,m}$ denotes weight.

Then we get a piecewise line called as principal curve by joining up all $\mathbf{f}^{(j)}(\lambda_{i,d}^{(j-1)})$.

(2.b) Projection Step: find every observation's projection $\mathbf{f}(\lambda_{i,d}^{(j)})$ on the principal curve constructed at step (2.a) through

$$\lambda_{i,d}^{(j)} = \arg\min_{\lambda} \| \mathbf{x}_{i,d} - \mathbf{f}^{(j)}(\lambda) \| \quad (10)$$

Sort $(\lambda_{i,d}^{(j)}, \mathbf{f}^{(j)}(\lambda_{i,d}^{(j)}))$ in increasing order of $\lambda_{i,d}^{(j)}$ $(i = 1,2,\ldots,M)$. Let:

$$\lambda_{i,d}^{(j)} = \sum_{a=1}^{i} \left| \mathbf{f}^{(j)}(\lambda_{a,d}^{(j)}) - \mathbf{f}^{(j)}(\lambda_{a-1,d}^{(j)}) \right| \quad (11)$$

here $i = 2,\ldots,M$ and $\lambda_{1,d}^{(j)} = 0$;

(3) If the change in $\sum_{i=1}^{M} \left\| \mathbf{x}_{i,d} - \mathbf{f}^{(j)}(\lambda_{i,d}^{(j)}) \right\|$ is below some previous given threshold, go to step (4); else go to step (2)

(4) Replace codebook state mean with iterated value $f_1(\lambda_{i,d}^{(j)})$, $(i = 1,2,\ldots,M)$.

(5) If $d = L$, stop; else $d = d + 1$, go to step (1).

It can be demonstrated that the principal curves is a nearest curve describing joint behavior of feature statistics and codebook mean of all states [4].

Recall that the goal is to exploit correlation between observations of different acoustics states to improve a system's recognition performance. A typical variable reflecting such state correlation information is covariance matrix between states. Therefore the iterative principal curves algorithm is initialized by covariance matrix. Furthermore, the principal curves algorithm uses a curve joining all the states' feature statistic information and codebook means (as alluded to in the introduction). The projections of the feature statistics onto this curve are taken as the final updated codebook parameters. Since the state label may be error in noisy environment, the weight average computation in equation (9) may introduce the feature statistics of the right state into the new coordinate, so the codebook mean can be effectively modified.

Comparing to MLLR, PCSA is a nonlinear transformation method. PCSA utilized latent variable λ joining up all state, so the correlation information between states mainly are showed by the sort of latent variable. While in MLLR the correlation information only are showed in the clustering step.

3. Experiment results and discussions

3.1. Acoustic model

The acoustic part of our model is based on a modified Hidden Markov Model (HMM) called the Duration Distribution Based Hidden Markov Model (DDBHMM). The DDBHMM [9] is an inhomogeneous HMM which relies on the fact that the state duration distribution is relatively stationary. Whereas the standard HMM uses the state transition probability, the DDBHMM utilizes the duration distribution probability. Given T frames of state observations $o_t(t = 1,2,\ldots,T)$ which combined make up a feature vector of speech $O = (o_1, o_2, \ldots, o_T)$ and vector word strings

$W = (w_1, w_2, \ldots, w_K)$, the optimum word string W^* is defined as :

$$W^* = \arg\max_W P(O|W) = \arg\max_W \{\max_{S_2,\cdots,S_M} \prod_{i=1}^{M} P_i(\tau_i) \prod_{t=S_i+1}^{S_{i+1}} b_i(o_t)\} \quad (12)$$

where $P(O|W)$ is the probability of the observation sequence O given the word string W, τ_i is the number of frames of the observation vectors belonging to i^{th} state (or the duration of state i : $\tau_i = S_{i+1} - S_i, (i = 1,2,\ldots,M)$, S_i is the state segment point and M is the number of states. $P_i(\tau)$ denotes the duration distribution function of the i^{th} state, while $b_i(o_t)$ is the probability density of observation vector o_t in state i.

3.2. Experiment conditions

All experiments were run using 863 "large vocabulary continuous speech recognition" databases, which are sponsored by the National 863 High-Tech Project of China for the assessment of large vocabulary continuous speech recognition systems. White noises from the Noise92 database were selected as speech contaminants. The noises were artificially added at a variety of signal-to-noise ratios ranging from 15dB to 25dB. Here SNR=10*log (speech energy/noise energy). Data was divided into two parts. The first part, consisting of the first 76 files, served to train the system and the second part, consisting of 7 files, was used to evaluate/test the recognition rate. The 83 files came from 83 different male speakers, and every file was comprised of more than 600 sentences. Each file was about 0.66 of an hour long containing about 0.457 of an hour of speech, so every sentence was approximately 2.74 seconds long.

Chinese speech is modeled as diphoneme. That is, each word is composed of an initial and a final. An initial includes 2 states, whereas a final includes 4 states. Thus, 100 initials and 164 finals combine to produce $100 \times 2 + 164 \times 4 + 1 = 857$ states (including 1 state representing silence). Each state observation is modeled as a single Gaussian distribution. The system employs feature vectors consisting of 14 MFCCs with normalized energies and their first- and second-order derivatives (a total of 45 parameters). The DDBHMM is built and the codebooks are trained using these feature vectors.

3.3. Experiment results and discussions

Table 1 gives the acoustics recognition error rate of 100-best candidates PCSA compared with MLLR under clean and white noise environment. In our experiments, 10 to 120 sentences are used for each speaker for adaptation and the remaining sentences for recognition. The same computations are conducted to the other 7 files and the average error rate of the recognized 7 speakers is given in table 1. The file is first recognized as rough state segmentation, and then the recognized results are used as state label to instruct the following recognition.

MLLR approach here is grouped into 14 clusters. Since MAP needs exact state labels, which can't be obtained when noise exists, MAP is inferior to baseline, so we didn't mention its result. The experiment results are shown in table 1.

From the table it can be seen that against white noise background PCSA is superior to MLLR. From clean, SNR 25dB to 15dB of adapting 120 sentences, relative error rate decreases respectively by 15.45%, 45.34%, 42.23%, and 33.82% of PCSA compared to baseline, while 9.76%, 32.60%, 33.50% and 31.18% of MLLR. Furthermore, the more adaptation data there are, the lower recognition error rate there is. Adapting 60 sentences compared to adapting 120 sentences under 15dB white noise, the relative error rate is cut down from 19.20% (15.65%, 19.37%) to 33.82% (12.82%, 19.37%). Here SNR=10*log (speech energy/noise energy) and recognition error rate= 100% *(recognized error syllables)/ (recognized total syllables).

Figure 1 analyze the effect of $w_{i,m}$ in step (2.a), here we consider two kind of values of $w_{i,m}$: the first is average weight of all points in neighborhood, the second is LWRLS (locally weighted running-lines smoother) [4]. We can know from figure 1 that the LWRLS is superior to average weight.

In PCSA, no assumption of any probability distribution of λ is made. Since in fact noisy speech may not take Gaussian distribution, the proposed method exhibits some satisfactory results. Furthermore, principal curves construct a nonlinear curve describing the relation of observations of different states and codebook parameters. Since the inherent correlation information of acoustic state is exploited, PCSA not only adapt speaker information but also reduce noise's effect. The system's performance is finally improved.

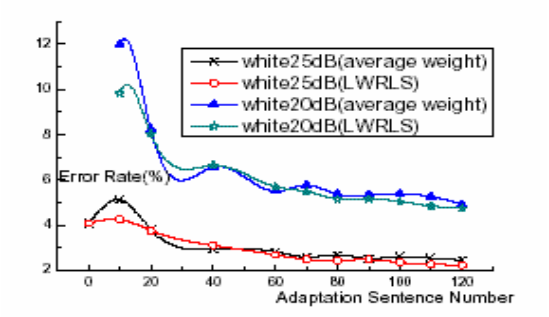

Figure 1. Weight effect on error rate

4. Conclusion

Based on principal curves, the paper has presented a new speaker adaptation method that takes state correlation into account. This approach gets an initial estimate of latent variable utilizing covariance matrix, and it does not assume any probability of latent variable. Furthermore, correlation between states is modeled through a principal curve. Experiments on 863 large vocabulary continuous speech recognition databases confirm that this algorithm is beneficial, compared to MLLR especially in noisy cases.

References

[1] 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, no. 2, pp. 171-185,1999

[2] Jean-Luc Gauvain, Chin-Hui Lee. “Maximum a posterior estimation for multivariate Gaussian mixture observations of markov chains”, IEEE Trans. On Speech and Audio Processing, Vol. 2, No.2, pp. 291-298, April 1994
[3] Roland Kuhn, Jean-Claude Junqua, Patrick Nguyen, Nancy Niedzielski. “Rapid speaker adaptation in eigenvoice space”, IEEE Transactions on speech and audio processing. Vol.8, No.6 2000
[4] Trevor Hastie, Werner Stuetzle,” Principal curves”, Journal of the American Statistical Association, Vol. 84, No. 406, pp. 502-516, Jun., 1989.
[5] K. Reinhard and M. Niranjan, ”Parametric subspace modeling of speech transitions”, Speech Communication, Vol. 27, no. 1, pp. 19-42, 1997.
[6] Reinhard K. , and Niranjan M,” Subspace models for speech transitions using principal curves,” Proceedings of Institute of Acoustics, Vol. 20, no. 6, pp. 53-60, 1998
[7] Yong Guan, Hongwei Qi, Wenju Liu, and Jue Wang,” Improving performance of text-incorrelation speaker identification by utilizing contextual principal curves filtering,” in Proc. Interspeech 2004: pp. 1781-1784.
[8] Jeffrey D. Banfield, Adrian E. Raftery. ”Ice floe identification in satellite images using mathematical morphology and clustering about principal curves”, Journal of the American Statistical Association, Vol. 87, no. 417, pp. 7-16, Mar., 1992
[9] X. Xiao, Wang Zuoying. ”Duration Distribution based HMM for speech recognition”, Acta of Electronica Sinica, Vol. 32, no.1, pp.46-49, 2004(in Chinese)

Table 1. Recognition Error rate of PCSA comparable with MLLR

Number of sentences for adaptation	Clean		White 25dB		White20dB		White15dB	
	PCSA	MLLR	PCSA	MLLR	PCSA	MLLR	PCSA	MLLR
0(Baseline)	1.23%	1.23%	4.08%	4.08%	8.24%	8.24%	19.37%	19.37%
10	1.66%	1.30%	4.25%	4.46%	9.83%	10.85%	23.56%	24.84%
20	1.48%	1.15%	3.76%	3.73%	8.05%	8.38%	20.30%	21.12%
40	1.35%	1.17%	3.11%	3.17%	6.64%	6.94%	17.09%	17.76%
60	1.23%	1.13%	2.69%	2.92%	5.70%	6.20%	15.65%	15.47%
80	1.23%	1.12%	2.44%	2.79%	5.16%	5.74%	14.30%	14.89%
100	1.13%	1.11%	2.35%	2.80%	5.05%	5.61%	13.48%	14.13%
120	1.04%	1.11%	2.23%	2.75%	4.76%	5.48%	12.82%	13.33%