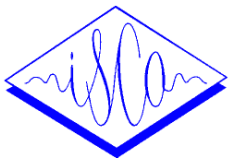

3rd International Conference on
Spoken Language Processing (ICSLP 94)
Yokohama, Japan
September 18-22, 1994

Speech Recognition Using Tree-Structured Probability Density Function

Takao Watanabe, Koichi Shinoda, Keizaburo Takagi, Eiko Yamada
Information Technology Research Laboratories
NEC Corporation
4-1-1 Miyazaki, Miyamae-ku, Kawasaki 216, JAPAN

ABSTRACT

This paper proposes a new speech recognition method using tree-structured probability density function(*pdf*) to realize high speed HMM based speech recognition. In order to reduce likelihood calculation for a *pdf* set composed of the Gaussian *pdf*s for all mixture components, all states and all recognition units, the likelihood calculation is coarsely done for the *element pdf* (element of the *pdf* set) whose likelihood $N_k[x_t]$ at time t is not likely to be large.

The *pdf* set is expressed as the tree-structured form. A leaf node of the tree corresponds to an *element pdf*. A non-leaf node corresponds to a cluster composed of *element pdf*s. To each cluster is attached a *cluster pdf* obtained by approximating the mixture of all *element pdf*s in the cluster by a single Gaussian *pdf*. In the recognition, the likelihood set is calculated by searching the tree; by calculating the likelihood from the *cluster pdf* at the node and traversing the nodes with the largest likelihood from the root node.

Recognition experiments showed that the amount of computation was drastically reduced by the proposed method with little degradation in the recognition accuracy for both speaker-independent and speaker-adaptive modes.

1. INTRODUCTION

In speech recognition based on hidden Markov models(HMM), recently, continuous output probability with mixture density[1] is commonly used. In continuous density HMM, the number of Gaussian probability density function(*pdf*)s used is determined by the multiplication of the number of the recognition units, the number of the states for a recognition unit, and the number of mixture components in a state. The number of the *pdf*s is reduced to some extent, when HMMs with tied-mixtured type *pdf*s[1] are used, compared with the conventional continuous density HMM.

In either case, the conventional or tied-mixtured case, in the recognition process, the number of likelihood calculation times is the same as the total number of the *pdf*s. To realize a realtime speech recognition system with small computational cost, it is important to reduce the amount of likelihood calculation, as well as the amount of speech analysis processing and trellis search(word matching).

This paper proposes a new high speed speech recognition method using a *pdf* set with a tree structure.

The paper is organized as follows: In Section 2, a new recognition algorithm using tree-structured *pdf* is presented. In Section 3, experimental results are described.

2. ALGORITHM

2.1. Basic principle

Let a sequence of input vectors be

$$X = \{x_1, ..., x_t, ..., x_T\},$$

and let a *pdf* set, composed of the Gaussian *pdf*s for all mixture components, all states and all recognition units, be

$$Y = \{N_1[.], ..., N_k[.], ..., N_K[.]\}.$$

Each element in the *pdf* set is called an *element pdf*. At each time t, in the recognition, it is necessary to calculate a likelihood set

$$B_t = \{N_k[x_t], k = 1, ..., K\}. \quad (1)$$

Let's consider how to efficiently obtain the likelihood set B_t. The basic idea considered here is as follows: For the *element pdf* which is likely to correspond to time t (that is, the *element pdf*, whose likelihood $N_k[x_t]$ at time t is large), the likelihood is precisely calculated. For the *element pdf* which is unlikely to correspond to time t, calculation is done coarsely.

2.2. Likelihood calculation using tree-structured pdf

To realize the principle mentioned above, the *pdf* set Y is expressed as the form of the tree-structure, and the calculation of Eq. (1) is done by searching the tree.

Figure 1 shows the tree-structure. A leaf node of the tree corresponds to an *element pdf*. A non-leaf node corresponds to a cluster composed of *element pdf*s. Each cluster consists of other clusters or *element pdf*s. To each cluster is attached a *cluster pdf*, obtained by approximating the mixture of all *element pdf*s in the cluster by a single Gaussian *pdf*.

In the recognition, the likelihood set B_t is calculated by searching the tree. First, calculate likelihoods from the *cluster pdf*s for nodes $\{k(1,1), ..., k(1,j), ..., k(1,J_1)\}$ in the first stage (that is, the nodes which are child nodes of the root node):

$$\{N_{k(1,1)}[x_t], ...N_{k(1,j)}[x_t], ...N_{k(1,J_1)}[x_t]\}.$$

Then, select the node which gives the largest likelihood. It is possible to select the multiple nodes by selecting the N-best likelihoods. For the child nodes of the selected nodes, this procedure is repeated. This tree search procedure is continued until all selected nodes reach the leaf nodes.

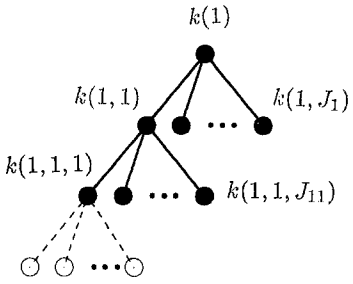

Figure 1: Tree-structure of pdf

For the selected leaf nodes, the likelihood is calculated from the *element pdfs* of the nodes. For unselected leaf nodes, the likelihoods are approximated by the likelihoods calculated from the upper node *cluster pdfs*. Using this procedure, the likelihood set B_t in Eq. (1) can be calculated very efficiently.

In the implementation, it is possible to remove leaf nodes in the tree to reduce the computation and memory requirement, instead of attaining all nodes in the tree.

When a feature vector space consists of sub-spaces, which are independent from each other, such as a cepstral parameter space and a cepstral difference parameter space, it is effective to construct the tree-structured *pdf* separately in each sub-space.

2.3. Design for tree-structured pdf

The tree-structured *pdf* is designed beforehand by a top-down clustering technique.

First, the *element pdf* set Y is set as the initial *pdf* set for clustering, and the *pdf* set is divided into clusters. The number of the cluster is given beforehand. Next, each cluster is further divided into sub-clusters. This division procedure is repeated the number of times determined beforehand.

As the clustering algorithm, the authors use the K-means algorithm[2]. The procedure is as follows:

1. Give initial *cluster pdfs*.
2. Steps 3 and 4 are repeated until the cluster division is converged.
3. Determine to which cluster each *element pdf* belongs. The authors use *divergence*[2] as the distance measurement between *cluster pdf* $N_m[.]$ and *element pdf* $N_k[.]$. The distance measurement $D[k,m]$ is given by

$$\begin{aligned} D[k,m] &= \int (N_k[x] - N_m[x]) \\ &\quad \{\log(N_k[x]) - \log(N_m[x])\} dx \\ &= \sum_i \left(\frac{\sigma_k(i)^2 + \Delta_{km}(i)^2}{\sigma_m(i)^2} + \frac{\sigma_m(i)^2 + \Delta_{km}(i)^2}{\sigma_k(i)^2}\right), \\ \Delta_{km}(i)^2 &= (\mu_k(i) - \mu_m(i))^2, \end{aligned}$$

where $\mu_k(i)$ and $\mu_m(i)$ are i-th components of the mean vectors for the *element* and *cluster pdfs*, respectively, and $\sigma_k(i)^2$ and $\sigma_m(i)^2$ are i-th diagonal components for the diagonal covariance matrices of the *element* and *cluster pdfs*, respectively.

4. For each cluster, calculate a new *cluster pdf* from the set of the *element pdfs* which belongs to the cluster. This is done by approximating the mixture of the *element pdfs* by a single Gaussian *pdf*, as follows:

$$\begin{aligned} \mu_m(i) &= \frac{1}{K}\sum_k \mu_k(i), \\ \sigma_m(i) &= \frac{1}{K}\left[\sum_k \sigma_k(i)^2 + \sum_k \mu_k(i)^2 - K\mu_m(i)^2\right], \end{aligned}$$

where Σ denotes the summation regarding element number k belonging to cluster m.

2.4. Speaker adaptation using tree-structured pdf

The proposed method is applicable to speaker-adaptive speech recognition, as well as speaker-independent speech recognition, which uses HMM mean vector mapping.

Tree-structured *pdf* must be designed after the *element pdf* set is obtained. In case of speaker-independent speech recognition, the tree design is only required once in the training phase. In case of speaker-adaptive speech recognition, however, the *element pdf* set is modified by speaker adaptation process. The tree could be reconstructed, based on the *element pdf* set after speaker adaptation. This causes an increase in the computation amount for speaker adaptation, which is undesirable for practical applications.

In speaker adaptation, instead of reconstructing the tree-structure, it is possible to use the tree-structure designed based on the *element pdf* set of a speaker-independent model, which is used as a reference model for speaker adaptation. In this study, the authors adopt the following method in speaker adaptation, using the interpolated mean vector mapping technique, which the authors developed[5].

First, the *element pdf* set is adapted. The mean vector of each *element pdf* is mapped by the speaker adaptation technique, which is based on Viterbi time-alignment using the reference model and an interpolation technique. By this, a new *element pdf* set is obtained. Then, the mean vectors of the *cluster pdfs* are calculated by averaging the the mean vectors of the *element pdfs* in the cluster. Since this process does not include the clustering procedure, efficient speaker adaptation is realized.

2.5. Discussion

The proposed method here can be interpreted as the expansion of the vector quantization(VQ) based pre-selection method[3]. In order to reduce the likelihood calculation in continuous density HMM, the method[3] has applied VQ technique to the mean vector set of the *pdfs*, and likelihoods have been calculated only for the *pdfs* in the pre-selected clusters. Pre-selection of the clusters have been done based on the distances between the input vector and the code vectors in the VQ codebook. While cluster pre-selection has been done using the different measurement

from the measurement used in HMM, coarse likelihood calculation, in the proposed method, uses the same measurement as the measurement used in HMM. In addition, the tree-structure is newly introduced into the clustering in the proposed method.

The proposed method contains the expanded concept of the tree-structured VQ, which has been used in speech coding[4]. Tree-structured VQ has been used to determine the optimal code to send in speech coding. In the proposed method, tree-structured *pdf*is used to efficiently obtain a set of likelihoods for *pdf*s. The efficient calculation of a set of values is a new application of the tree-structure processing.

3. EXPERIMENTS

3.1. Database

Evaluation experiments were conducted on speaker-independent and speaker-adaptive word recognition using demi-syllable based speech recognition which the authors have developed[5, 6]. Each demi-syllable unit was modeled by a left-to-right HMM where Gaussian mixture *pdf* was used. The total number of the *element pdf*s was 1500 for all units, all states and all mixture components. The demi-syllable HMMs were trained by 23 male speakers' phonetically balanced 250 word utterances. A 250 word set was used as the recognition vocabulary which was different from the training word set. As test samples, 250 words uttered once by five male speakers, who were different from the trained speakers, were used. As samples for speaker adaptation, the 250 phonetically balanced words uttered once by the five test speakers were used.

The utterances were sampled at an 11 kHz sampling rate, and were analyzed at a 16 msec frame period. As a feature parameter set for each time frame, 10 mel-scaled cepstral parameters, time derivatives of the mel-scaled cepstral parameters and an amplitude parameter were calculated from FFT based spectrum with 0.15-5.0 kHz band limitation.

3.2. Design for the tree-structured pdf

Tree-structured *pdf*s were designed separately for three feature spaces: mel-cepstral parameter space, time derivative of mel-cepstral parameter space and amplitude parameter space. The tree-structured *pdf* was designed for speaker-independent HMM. The tree-structure used is shown in Figure 1. At the first stage in the clustering, a whole *element pdf* set was divided into 16 sub-clusters. At the second stage, each sub-cluster was further divided into 16 sub-clusters.

For the recognition, only the nodes(*cluster pdf*s) obtained at the first and second stages were preserved. The remaining *element pdf*s were removed. They are shown by white circled nodes in Figure 1. The tree-structure designed for speaker-independent HMM was also used for the speaker adaptive HMM.

3.3. Speaker-independent recognition results

In the recognition, multiple nodes (that is, multiple clusters), were selected at the first stage in the tree search. The number of selected clusters at the first stage, M, largely affects the likelihood calculation times. The numbers of likelihood calculation times were 32, 64, and 96 for $M = 1$, 3, and 5, respectively, because the likelihood calculation was done 16 times at the first stage and $16M$) times at the second stage. These cases roughly correspond to the likelihood calculation reductions to 2.1%, 4.3%, and 6.4%, respectively.

Table 1: Recognition rate for selected cluster number at the first stage

Cluster number selected at the first stage	Recognition rate
1	93.8%
3	94.8%
5	95.6%
Strict method	97.2%

Table 1 shows the recognition result in speaker-independent word recognition for various selected cluster numbers. The recognition rate tends to degrade as the selected cluster number was reduced. However, when selecting five clusters, recognition rate degradation was fairly small, compared with the conventional strict calculation method.

The cluster number divided at the second stage, N, also affects the likelihood calculation times. Table 2 shows the recognition result when $N = 8$ and 16, where the number of divided clusters at the first stage and the number of selected clusters at the first stage were 16 and 5, respectively. For both cases, recognition rate degradation was small. These cases roughly correspond to the numbers of likelihood calculation times, 56 and 96, that is, likelihood calculation reduction to 3.7% and 6.4%, respectively.

3.4. Speaker-adaptive recognition results

For speaker adaptation, two implementations are permissible regarding the tree-structure design as described previously. One is to design a new tree-structure using speaker-adapted HMM, and the other is to use the tree-structure designed using speaker-independent HMM. From the preliminary experiment, it was shown that little difference exists in recognition accuracy between these two implementations. Accordingly, the tree-structure for speaker-independent HMM was used for speaker-adaptive recognition in the following experiments.

Table 3 shows the recognition result for speaker adaptation. Speaker adaptation worked effectively in the proposed method as well as in the strict method, and recognition accuracy degradation in the proposed method was little in the

Table 2: Recognition rate for divided cluster numbers at the second stage

Cluster number divided at the second stage	Recognition rate
8	95.3%
16	95.6%

Table 3: Recognition rate for speaker adaptation (1)

	Speaker-independent	Speaker-adaptive
Proposed method	95.6%	98.2%
Strict method	97.2%	98.5%

speaker adaptive mode where speaker adaptation was done using 250 word utterances.

Speaker adaptation was also effective in case where the number of utterances for speaker adaptation was relatively small. As shown in Table 4, recognition accuracy was satisfactory for speaker adaptation using only 50 word utterances.

3.5. Discussion

The experimental results indicate that the proposed tree-structured *pdf* method was effective for both speaker-independent speech recognition and speaker-adaptive speech recognition. High recognition rates, 98.2% for speaker-adaptive case and 95.6% for speaker-independent case, were obtained where the number of necessary likelihood calculation times for an input feature vector was only 96, which corresponds to the likelihood calculation reduction to 6.4%.

Table 4: Recognition rate for speaker adaptation (2)

Word number for speaker adaptation	Recognition rate
250 words	98.2%
50 words	97.2%

By the proposed method, likelihood calculation has been reduced to the extent comparable to speech analysis processing cost. Further, the likelihood calculation cost does not depend on the recognition vocabulary. The proposed method is undoubtedly effective to realize the low-cost speaker-independent or speaker-adaptive speech recognition with high accuracy and flexible vocabulary.

4. CONCLUSION

This paper proposed a new speech recognition method using tree-structured probability density function(*pdf*) to realize high speed HMM based speech recognition. In order to reduce the likelihood calculation for a *pdf* set composed of the Gaussian *pdf*s for all mixture components, all states and all recognition units, the *pdf* set was expressed as the tree-structured form. In the recognition, the likelihood set was calculated by searching the tree.

In 250 word recognition experiments using demi-syllable HMM, it was shown that the amount of computation was drastically reduced by the proposed method with little degradation in the recognition accuracy, for both speaker-independent and speaker-adaptive modes.

ACKNOWLEDGMENTS

The authors wish to thank Ken-ichi Iso, Hiroaki Hattori, and the members of Human Language Research Laboratory for helpful discussions.

REFERENCES

[1] X. D. Huang, Y. Ariki, M. A. Jack, Hidden Markov models for speech recognition, Edinburgh University Press (1990).

[2] J. Tou, R. Gonzalez, Pattern recognition principle, Addison-Wesley Publishing Company, U.S.A. (1974).

[3] E. Bocchieri, Vector quantization for the efficient computation of the continuous density likelihoods, *Proc. ICASSP-93*, pp.692-695 (1993).

[4] A. Buzo et.al., Speech coding based upon vector quantization, *IEEE Trans. ASSP-28*, 5, pp.562-574 (1980).

[5] K. Shinoda et.al., Speaker adaptation for demi-syllable based continuous density HMM, *Proc. ICASSP-91*, pp.857-860 (1991).

[6] T. Watanabe et.al., Speaker-independent speech recognition based on hidden Markov model using demi-syllable units, *Trans. IEICE*, J75-D-II, 8, pp.1281-1289 (1992) (in Japanese)