A Discriminative Training Algorithm for VQ-Based Speaker Identification

Jialong He, Li Liu, and Günther Palm

***Abstract*— A novel method, referred to as *group vector quantization* (GVQ), is proposed to train VQ codebooks for closed-set speaker identification. In GVQ training, speaker codebooks are optimized for vector groups rather than for individual vectors. An evaluation experiment has been conducted to compare the codebooks trained by the Linde–Buzo–Grey (LBG), the learning vector quantization (LVQ), and the GVQ algorithms. It is shown that the frame scores from the GVQ trained codebooks are less correlated, therefore, the sentence level speaker identification rate increases more quickly with the length of test sentences.**

***Index Terms*— Neural networks, speaker identification, training algorithms, vector quantization.**

I. Introduction

In this correspondence,[1] we address the problem of closed-set speaker identification based on the vector quantization (VQ) speaker model. VQ-based speaker recognition is a conventional and successful method [1]. Due to its simplicity, the VQ speaker model is often used as a reference when studying other methods [2]. In VQ-based speaker recognition, each speaker is characterized with several prototypes known as *code vectors*, and the set of code vectors for each speaker is referred to as that speaker's *codebook*. Normally, a speaker's codebook is trained to minimize the quantization error for the training data from that speaker. The most commonly used training algorithm is the Linde–Buzo–Grey (LBG) algorithm [3]. The codebook trained based on the criterion of minimizing the quantization error tends to approximate the density function of the training data. A straightforward extension to the VQ speaker model is the matrix quantization (MQ) in which a block of frames is considered each time during the quantization procedure [4]. The same algorithm as that used for training a VQ model can also be used for generating MQ codebooks, the only difference is that, instead of a single vector, a block of vectors is used as a single training sample.

Both VQ and MQ codebooks are trained nondiscriminatively, that is, the parameters of a speaker model are estimated solely from the training data of that speaker. In closed-set speaker identification, however, a classification decision is made based on all speaker models in the system, therefore, the criterion of minimizing the quantization error does not necessarily lead to an optimal decision. A more suitable optimization criterion for speaker identification is to minimize the classification error rate.

Kohonen proposed a discriminative training procedure called *learning vector quantization* (LVQ) [5]. The goal of LVQ training is to reduce the number of misclassified vectors. The LVQ trained codebook is used to define directly the classification borders between classes. It was shown that after applying LVQ training, the correct classification rate for feature vectors increases significantly [6]. However, we found that even though the LVQ codebook can give a better frame level performance than the corresponding LBG codebook, the sentence level performance may not be improved. Sometimes it may even be degraded. This is because the speech feature vectors are highly correlated, and this correlation has not been taken into account if a model is optimized for individual vectors.

To overcome this weakness, we propose a modified version of the LVQ, *group vector quantization* (GVQ). The GVQ training procedure is similar to that of LVQ, but in each iteration a number of vectors rather than a single vector are considered. The average quantization error of this vector group is used to determine whether the speaker models should be modified or not. The goal of GVQ training is to reduce the number of misclassified vector groups. The GVQ training procedure can be viewed as a combination of the LVQ training method and the idea of matrix quantization.

Manuscript received April 11, 1997; revised May 11, 1998. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Richard C. Rose.

J. He was with the Abteilung Neuroinformatik, University of Ulm, 89069 Ulm, Germany. He is currently with the Department of Speech and Hearing Science, Arizona State University, Tempe, AZ 85287-1908 USA (e-mail: jhe@asu.edu).

L. Liu and G. Palm are with the Abteilung Neuroinformatik, University of Ulm, 89069 Ulm, Germany.

Publisher Item Identifier S 1063-6676(99)02743-1.

[1] C source code of the GVQ, LBG, and LVQ algorithms are available from ftp://ftp.informatik.uni-ulm.de/pub/NI/jialong/GVQPROG.ZIP.

II. VQ-Based Speaker Identification

A. LBG and LVQ Algorithms

In principle, the training vectors contain all available information and can be used directly to represent a speaker. However, such a direct representation is not practical when there is a large amount of training vectors. VQ provides an effective means for compressing the short-term spectral representation of speech signals. In vector quantization, mapping vector $\boldsymbol{x}$ into its nearest code vector $\boldsymbol{y}_{NN}$ leads to a quantization error $e(\boldsymbol{x}) = |\boldsymbol{x} - \boldsymbol{y}_{NN}|$. Suppose there are N vectors, $\{\boldsymbol{x}_t\}_1^N$, to be quantized, the average quantization error is given by

$$E = \frac{1}{N} \sum_{t=1}^{N} e(\boldsymbol{x}_t). \tag{1}$$

The task of designing a codebook is to find a set of code vectors so that E is minimized. However, since a direct solution is not available, one has to rely on an iterative procedure to minimize E. The commonly used method is the LBG algorithm [3].

In the identification phase, a vector sequence can be obtained from a given test sentence by using short-time analysis techniques. The classification for the vector sequence is determined using a decision rule. Usually, the test sentence is classified as from the speaker whose model gives the smallest average quantization error. This sentence level decision rule is known as the average distance decision rule. Alternatively, the identity of the unknown speaker can be determined by taking a majority voting from all test vectors. With this decision rule, each test vector is assigned to a speaker whose model contains the global nearest code vector to this vector. Since the majority voting decision rule can be regarded as a special case of the average distance decision rule, we will rely on the latter in the following discussions.

The LBG trained codebook is optimal in the sense that the quantization error is minimized. If the codebook is used for applications such as speech coding, this criterion is a proper one. However, in speaker identification, the codebook is used for classification. Minimizing the quantization error does not necessarily lead to the best classification performance. A more suitable criterion for training a VQ-based classifier is to minimize the classification error rate. The LVQ algorithm is a well-known discriminative training procedure in which the codebooks of all classes are trained together and the code

vectors are modified depending on the local difference of density functions [5]. There are three variants of the basic algorithm; we will use the LVQ3 because it usually gives a better performance.

B. Group Vector Quantization

As mentioned before, the identity of a speaker is determined from a vector sequence rather than from a single vector. In our previous studies, we encountered the problem that LVQ trained codebooks perform better at the frame level, but at the sentence level, they often give a lower speaker identification rate than the corresponding LBG codebooks. It is known from the sequential hypothesis testing theory that the classification performance improves with the length of sequences; however, the rate of this improvement depends on the degree of correlation between individual scores [7]. It might be that the frame scores from an LVQ trained codebook are highly correlated. To reduce this correlation, we develop the following training procedure, the GVQ, to optimize the codebooks for speaker identification.

Like LVQ, the GVQ algorithm also needs a good initialization to the codebooks. This can be done with the LBG method. Suppose all codebooks are initialized and the code vectors are labeled with their corresponding class membership, the following procedure is used to fine-tune the positions of code vectors to create GVQ codebooks.

1) Randomly choose a speaker, designate its ID as p.
2) From the training data belonging to p, take N vectors $X = \{\boldsymbol{X}_t\}_1^N$ as a group.
3) Calculate average quantization errors E_i using (1) for all speaker models (i indicates the model's ID). If speaker q's codebook ($q \neq p$) gives the smallest quantization error, go to step 4, otherwise go to step 5.
4) If $(E_p - E_q)/Ep, <w$, where w is a preselected threshold, for each vector $\boldsymbol{x}_t$ in the group, find its nearest code vector from speaker p's codebook (denoted as y^p_{NN}), and the nearest code vector from speaker q's codebook (denoted as y^q_{NN}), adjust the two code vectors simultaneously by
$$\begin{aligned} \boldsymbol{y}^p_{NN} &\Leftarrow \boldsymbol{y}^p_{NN} + \alpha(\boldsymbol{x}_t - \boldsymbol{y}^p_{NN}) \\ \boldsymbol{y}^q_{NN} &\Leftarrow \boldsymbol{y}^q_{NN} - \alpha(\boldsymbol{x}_t - \boldsymbol{y}^q_{NN}) \end{aligned} \tag{2}$$
where α is a small constant known as the learning rate. If the ratio $(E_p - E_q)/E_p$ is larger than the threshold w, this vector group is simply ignored. After finishing this step, go to step 6.
5) In this case, the current vector group is correctly classified. To keep the codebook still approximating the density function, the code vectors are moved more closer to their training data. For each vector in the group, adjust its nearest code vector in speaker p's codebook by
$$\boldsymbol{y}^p_{NN} \Leftarrow \boldsymbol{y}^p_{NN} + 0.1\alpha(\boldsymbol{x}_t - \boldsymbol{y}^p_{NN}). \tag{3}$$
After processing all vectors, go to step 6.
6) If the number of iterations is less than the desired number, go to step 1, otherwise the training procedure stops.

During iteration, the threshold value w shrinks linearly with the iteration loop. In step 2, we must decide the way of choosing vectors to compose a vector group and the number of vectors in a group. Two different ways of choosing vectors have been tried. The first one is to take vectors sequentially from the training sentences, so that the vectors in each group come from adjacent segments of speech. Fig. 1(a) illustrates this approach. Alternatively, we can choose N vectors randomly from the training data belonging to the current speaker. This approach is illustrated in Fig. 1(b). In this case, the vectors in a group may come from different sentences. As for the size of vector groups, if the goal is to reduce the error rate for individual vectors (like that in LVQ training), N should be set to one. However, if we want to promote the sentence level performance, N should be set to a value larger than one. The optimal value should be determined through experiments.

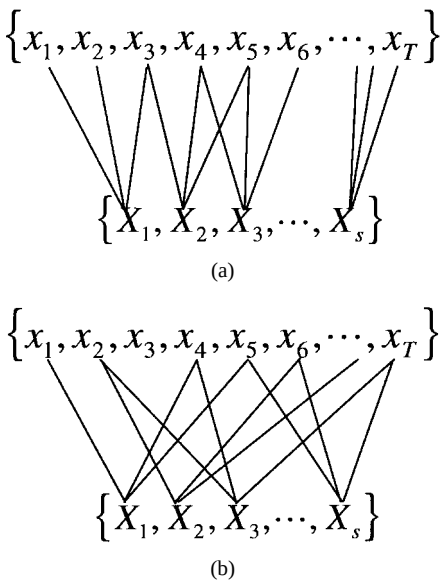

Fig. 1. Illustration of two selection methods for composing a vector group. (a) Sequential selection method. (b) Random selection method.

III. Evaluation Experiment

We did an evaluation experiment to demonstrate the effectiveness of GVQ training. Evaluation speech was taken from the YOHO database [8]. Forty speakers (20 males, 20 females) were used in our experiment. A standard preprocessing technique was used to derive spectral feature vectors [9]. A feature vector was composed of 16 mel frequency cepstral coefficients (MFCC's). The analysis window size was 32 ms (256 samples) with 16 ms overlapping. Data from all enrollment sessions were used for training. There were about 4500 training vectors from each speaker. Since the GVQ is a discriminative training procedure, to reduce memory demand and save experimental time, silence and unvoiced segments were discarded based on an energy threshold. Discarding unvoiced segments will inevitably degrade the overall performance, but this is not a serious problem because our objective is to compare the relative performance of codebooks trained with different methods. In the test sessions, each speaker provided 40 test sentences, the total number of test sentences was 1600. The average length of the test sentences was about 48 frames (after removing silence and unvoiced frames).

A. Learning Rate and Group Size

There are two parameters that should be specified in the GVQ training procedure. The first one is the learning rate α and the other is the group size N. Fig. 2 shows the evaluation results with different values of these two parameters. Because the codebook was initialized with the LBG algorithm, the initial point indicates the performance obtained with the LBG trained codebook. It is seen that in all situations the speaker identification performance improves after applying GVQ training. A good choice for the learning rate is $\alpha = 0.2$. An α that is too large may cause the performance unstable, while anα that is too small may need more training iterations. In this experiment, the number of iterations equals to the number of total training vectors.

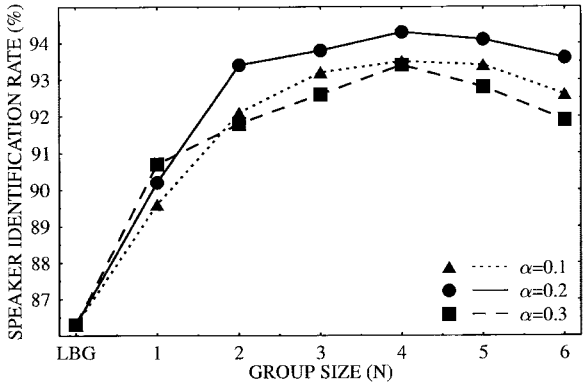

Fig. 2. Identification performance of GVQ codebooks trained with different learning rates and group sizes.

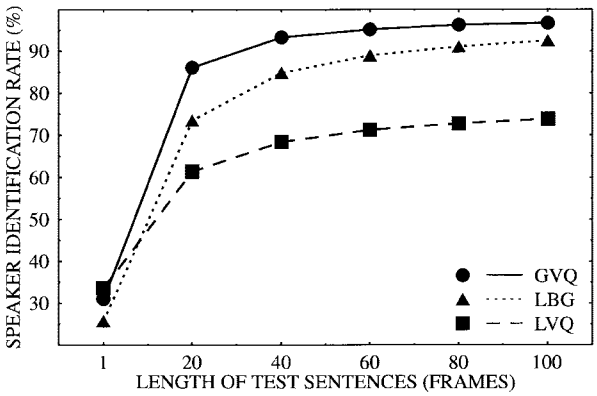

Fig. 3. Identification performance versus the length of test sentences from the codebooks trained with the three methods.

If the codebook is optimized for individual vectors $(N = 1)$, even though the sentence level performance is also improved, it is lower than that obtained with the codebooks trained with more vectors in a group. It is seen that $N = 4$ gives a good result. In the following experiments, we will fix the learning rate $\alpha = 0.2$ and the group size $N = 4$.

B. Length of Test Utterances

It is known that the identification performance improves with the length of test sentences. However, the rate of this improvement depends on the degree of correlation between the frame scores. To show how the identification rate increases with the length of test utterances, we first concatenated all 40 test sentences from the same speaker into a long one and then cut it into several pieces of specified lengths. The classification rate as a function of the length is plotted in Fig. 3. It is interesting to see that at the frame level $(L = 1)$, the LVQ trained codebook gives the highest classification rate, but it is soon outperformed by both the LBG and the GVQ codebooks. Since the rate of improvement of the LVQ curve is slower than that of the other two curves, this indirectly indicates that the frame scores from the LVQ codebook are more correlated. To gain more insight into this phenomenon, we define a measurement H as

$$H(X) = E_c(X) - \min_{w \neq c} \{E_w(X)\} \tag{4}$$

where X is a test vector sequence. $E_c(X)$ is the average quantization error from the correct codebook, that is, the codebook with the same ID number as the X's ID. Accordingly, $\{E_w(X)\}$ are the quantization errors from all other codebooks. It is easy to see that X is correctly classified if $H(X) < 0$, otherwise, X is misclassified.

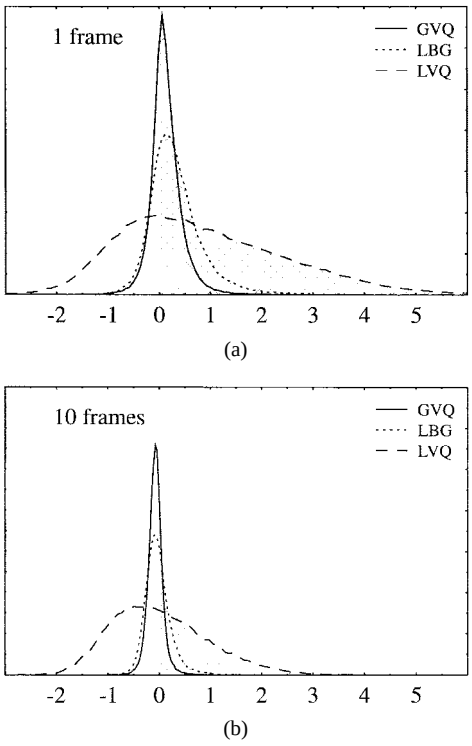

Fig. 4. (a) Histogram of $H(X)$ with single test frame. (b) Same codebooks, but with ten test frames.

The histograms of H obtained with the data used in the last experiment are shown in Fig. 4. The grid area in each figure represents the misclassification rate $(H(X) > 0)$. From Fig. 4 we see that both mean and variance of H in the ten-frame case are smaller than that in the one-frame case. In other words, the histograms tend to shift leftward and to becomes more narrow with the increase of the length of sequences. It is interesting to note that the scores from the LVQ codebook are so diverse that the variance of H is very large. On the other hand, due to the different learning rule used in GVQ training, the variance of H becomes even smaller than that with the LBG codebook.

By assuming that H has a normal distribution, the error rate can be calculated from the mean (μ) and the standard deviation (σ) of H. In fact, under the normal assumption, the error rate is a function of the ratio μ/σ. A smaller μ/σ (large magnitude if $\mu/\sigma < 0$) yields a lower error rate. The plot in Fig. 5 shows how the ratio μ/σ changes with the length of sequences for the three codebooks. We see that even though at the very beginning the LVQ curve is lower than the LBG curve, but the order becomes inverse for longer sequences. On the other hand, the falling speed of the GVQ curve is the fastest among the three, implying that the frame scores from the GVQ codebook are less correlated.

C. Codebook Size

In the above experiments, the codebook size is fixed at 16, which is very small comparing to that used by other researchers. In general, the performance improves with the codebook size. Table I gives the speaker identification rates for different codebook sizes. Since the Gaussian mixture model (GMM) becomes very popular for speaker recognition [2], we also tested the GMM with the same speech data and included the result here as a reference. The "model order"

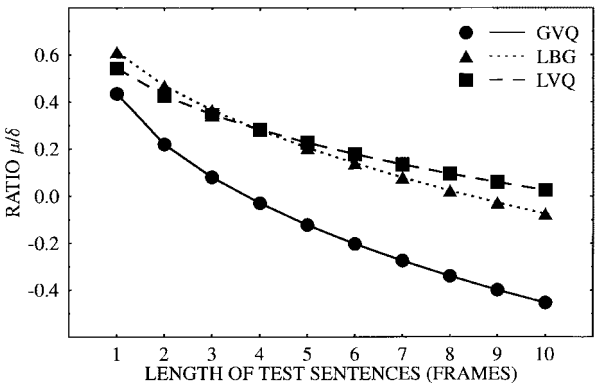

Fig. 5. Ratio of mean value and standard deviation of $H(X)$ as a function of the sequence length.

TABLE I
SPEAKER IDENTIFICATION RATE (PERCENT) AS A FUNCTION OF CODEBOOK SIZES. THE PERFORMANCE OF GMM IS INCLUDED AS A REFERENCE

Model Order	4	8	16	32	64
LBG	73.4	81.6	86.3	90.7	92.9
GVQ (Random)	83.4	88.6	91.9	93.6	95.1
GVQ (Sequential)	84.6	88.7	94.1	95.0	95.7
GMM	74.6	83.8	89.8	93.4	94.8

in Table I indicates the number of code vectors or the number of mixtures per speaker.

As expected, the performance improves with the model order. Besides, the GMM outperforms the conventional VQ speaker model trained with the LBG algorithm. However, if the codebook is further trained with the GVQ algorithm, it can perform even better than the GMM. Because the standard GMM is trained nondiscriminatively using the EM algorithm, we also tried to train the GMM discriminatively and achieved a very good result [10]. For the YOHO database, we found that the sequential selection method is somewhat better than the random selection method. The explanation is that the YOHO database contains very limited vocabulary. In another experiment with the TIMIT database, the random selection method is superior.

IV. SUMMARY AND CONCLUSIONS

In this correspondence, we have proposed a method to train VQ codebooks for closed-set speaker identification. The optimization criterion of the GVQ training procedure is to reduce the number of misclassified vector groups. Since a vector group is equivalent to a short sentence, the sentence level performance (i.e., speaker identification rate) is optimized more directly in the GVQ training procedure.

The effectiveness of the GVQ training procedure is demonstrated experimentally. It has been shown that the codebook trained with the GVQ algorithm can give a much higher speaker identification rate than both the LBG and the LVQ trained codebooks; it even outperforms the same order GMM. A proper choice for the learning rate α is 0.2 and for the group size is 4. The sequential selection method is slightly better than the random one for the YOHO database. The major force that leads to a significant improvement in performance is that the frame scores from GVQ trained codebook are less correlated.

REFERENCES

[1] A. E. Rosenberg and F. K. Soong, "Evaluation of a vector quantization talker recognition system in text independent and text dependent modes," *Comput. Speech Lang.*, vol. 22, pp. 143–157, 1987.
[2] D. A. Reynolds and R. C. Rose, "Robust text-independent speaker identification using Gaussian mixture speaker models," *IEEE Trans. Speech Audio Processing*, vol. 3, pp. 72–83, 1995.
[3] Y. Linde, A. Buzo, and R. M. Gray, "An algorithm for vector quantizer design," *IEEE Trans. Commun.*, vol. 20, pp. 84–95, 1980.
[4] M. S. Chen, P. H. Lin, and H. C. Wang, "Speaker identification based on matrix quantization method," *IEEE Trans. Signal Processing*, vol. 41, pp. 398–403, 1993.
[5] T. Kohonen, "The self-organizing map," *Proc. IEEE*, vol. 78, pp. 1464–1480, 1990.
[6] J. He, L. Liu, and G. Palm, "A text-independent speaker identification system based on neural networks," *Proc. Int. Conf. Spoken Language Processing (ICSLP)*, Yokohama, Japan, Sept. 1994, pp. 1851–1854.
[7] K. Fukunaga, *Introduction to Statistical Pattern Recognition*. New York: Academic, 1990, pp. 51–119.
[8] J. Godfrey, D. Graff, and A. Martin, "Public databases for speaker recognition and verification," in *Proc. ESCA Workshop on Automatic Speaker Recognition, Identification and Verification*, Martigny, Switzerland, 1994, pp. 39–42.
[9] J. W. Picone, "Signal modeling techniques in speech recognition," *Proc. IEEE*, vol. 79, pp. 1215–1247, 1993.
[10] J. He, L. Liu, and G. Palm, "A discriminative training algorithm for Gaussian mixture speaker models," *Proc. EUROSPEECH'97*, Rhodes, Greece, vol. 2, pp. 959–962.

Equalization of Speech and Audio Signals Using a Nonlinear Dynamical Approach

Albert M. Chan and Henry Leung

***Abstract*— We present the minimum phase space volume (MPSV) technique, a nonlinear dynamical technique for enhancing speech and audio signals corrupted by convolutional noise. The MPSV technique requires no assumptions or *a priori* information about the original signal, remains effective when the inverse filter order is overestimated, and significantly outperforms the LS method.**

I. INTRODUCTION

Existing speech enhancement techniques, such as interference subtraction, comb filtering, and speech resynthesis, are designed to remove additive noise [1]. However, another equally important category of degradation sources has received less attention: *convolutional* noise [2]. These distortions may be caused by the acoustical prop-

Manuscript received October 8, 1996; revised October 8, 1998. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Dennis R. Morgan.

A. M. Chan is with the Department of Electrical Engineering and Computer Science, Massachusetts Institute of Technology, Cambridge, MA 02139 USA.
H. Leung is with the Department of Electrical and Computer Engineering, University of Calgary, Calgary, Alta., Canada T2N 1N4 (e-mail: leungh@enel.ucalgary.ca).

Publisher Item Identifier S 1063-6676(99)02730-3.