Performance Comparison of Speaker Recognition using Vector Quantization by LBG and KFCG

DR. H. B. Kekre
Senior Professor
MPSTME, SVKM's NMIMS University
Mumbai- 400056, India

Vaishali Kulkarni
Ph.D Research Scholar
Assistant Professor, MPSTME, NMIMS University
Mumbai- 400056, India

ABSTRACT
In this paper, two approaches for speaker Recognition based on Vector quantization are proposed and their performances are compared. Vector Quantization (VQ) is used for feature extraction in both the training and testing phases. Two methods for codebook generation have been used. In the 1st method, codebooks are generated from the speech samples by using the Linde-Buzo-Gray (LBG) algorithm. In the 2nd method, the codebooks are generated using the Kekre's Fast Codebook Generation (KFCG) algorithm. For speaker identification, the codebook of the test sample is similarly generated and compared with the codebooks of the reference samples stored in the database. The results obtained for both the methods have been compared. The results show that KFCG gives better results than LBG.

General Terms
Speaker Recognition, Phone banking, Database services.

Keywords
Vector Quantization (VQ), Code Vectors, Code Book, Euclidean distance

1. INTRODUCTION
Speaker Recognition technology [1] – [3] makes it possible to extract the identity of the person speaking. This technology has made it possible to use the speaker's voice to control access to restricted services, for example, for giving commands to computer, phone access to banking, database services, shopping or voice mail, and access to secure equipment. It can be divided into Speaker Identification and Speaker Verification [3] – [5]. Speaker identification determines which registered speaker provides a given utterance from amongst a set of known speakers (also known as closed set identification). Speaker verification accepts or rejects the identity claim of a speaker (also known as open set identification).

Speaker identification task can be further classified into text-dependent or text-independent task [4, 5]. In the former case, the utterance presented to the system is known beforehand. In the latter case, no assumption about the text being spoken is made, but the system must model the general underlying properties of the speaker's vocal spectrum. In general, text-dependent systems are more reliable and accurate, since both the content and voice can be compared [3, 4].

The recognition Process

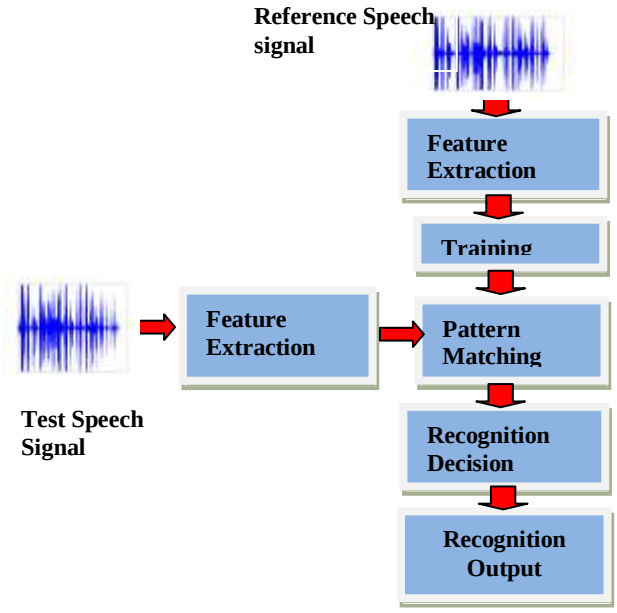

Fig. 1 Speaker Recognition system

Fig.1 shows the general block diagram of Speaker Recognition process. In the training stage, reference models are generated (or trained) from the reference speech signals by various methods. A reference model (or template) is formed by obtaining the statistical parameters from the reference speech signal. A test signal is compared with the reference templates at the pattern matching stage. The comparison may be conducted by probability density estimation or by distance (dissimilarity) measure. After comparison, the test pattern is labeled to a speaker model at the decision stage. The labeling decision is generally based on the minimum risk criterion.

Speaker Recognition systems have been developed for a wide range of applications [6] – [9]. Still, there are a number of practical limitations because of which widespread deployment of applications and services is not possible.

Vector Quantization (VQ) maps a 'k' dimensional vector space to a finite set $\mathbf{C = \{C_1, C_2, C_3 \ldots C_N\}}$.The set C is called

codebook consisting of 'N' number of codevectors and each codevector $C_i = \{c_{i1}, c_{i2}, c_{i3} \ldots c_{ik}\}$ is of dimension k. The key to VQ is the good codebook. The method most commonly used to generate codebook is the Linde-Buzo-Gray (LBG) algorithm [10], [11] which is also called as Generalized Lloyd Algorithm (GLA).VQ [10] – [12], [20] is an efficient data compression technique and has been used in various applications involving VQ-based encoding and VQ based recognition. VQ has been very popular in the field of speech recognition. [13] – [19]. We have proposed speaker identification using VQ by LBG algorithm [24].In this paper we propose speaker identification using VQ by KFCG algorithm. Also comparison of the results obtained by LBG and KFCG is shown.

In the next section we present the two codebook generation algorithms (LBG and KFCG). Section 3 consists of two approaches which are used for code book generation. Section 4 consists of results and conclusions in section 5.

2. CODEBOOK GENERATION ALGORITHMS

A. LBG Algorithm

For generating the codebooks, the LBG algorithm [11, 12] is used. The LBG algorithm steps are as follows [1, 11]:

1. Design a 1-vector codebook; this is the centroid of the entire set of training vectors.
2. Double the size of the codebook by splitting each current codebook y_n according to the rule
   $y_n^+ = y_n(1+\varepsilon)$
   $y_n^+ = y_n(1-\varepsilon)$

where n varies from 1 to the current size of the codebook, and ε is a splitting parameter.

3. Find the centroids for the split codebook. (i.e., the codebook of twice the size)
4. Iterate steps 2 and 3 until a codebook of size M is designed.

Fig. 2 shows the generation of two codevectors v_1 and v_2 using the LBG algorithm [20].

B. Kekre's Fast Codebook Generation Algorithm (KFCG)

In this algorithm for generating the codebook the following procedure is used [20] – [23]:

1. Initially we have only one cluster which is the entire training vectors. Design a 1-vector codebook; which is the centroid the cluster.
2. Split the cluster into two by comparing the first element of all the training vectors in the cluster with the first element of the centroid as follows:
   If $v_{i,1} > c_{1,1}$ then $v_{i,1}$ is grouped into C_1 (cluster 1).
   Else $v_{i,1}$ is grouped into C_2 (cluster 2).
   Where v is the training vector and c is the centroid.
3. Find the centroids of C_1 and C_2 (this is 2-vector codebook). Now split C_1 into two clusters by comparing the second element of all the training vectors in C_1 with the second element of its centroids explained in step 2 above. Similarly split C_2 into two clusters by comparing the second element of all the training vectors in C_2 with the second element of its centroid.
4. Now four clusters are formed. Centroids of these four clusters are computed (this is 4-vector codebook). These four clusters are split further by comparing the third element of the training vectors in that cluster with the third element of its centroid as explained in step 2 above.
5. The process is repeated until a codebook of size M is designed.

Fig. 3 shows the generation of codevectors using the KFCG algorithm.

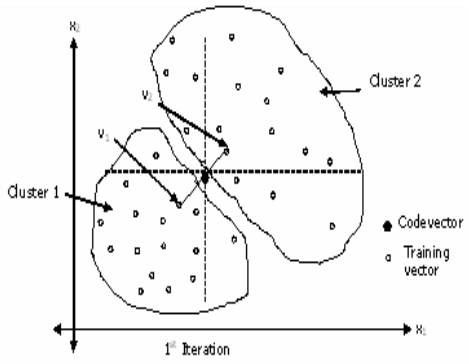

Fig. 2 LBG for 2 Dimensional case

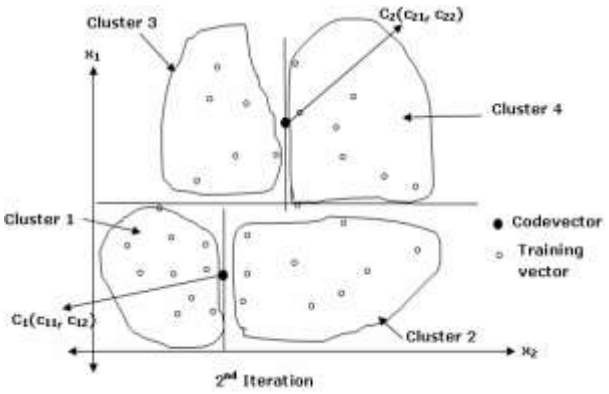

Fig. 3 KFCG for 2 Dimensional case

3. CODE BOOK GENERATION APPROACH

A. Without Overlap

The speech signal has amplitude range from -1 to +1. It is first converted into positive values by adding +1 to all the sample values. Then the sample values are converted into a 16 dimensional vector space. The code books for different size of code vectors are found using the LBG and KFCG algorithm discussed in the previous section.

B. With Overlap

The speech signal is converted into positive range in the same manner as in approach A. The samples are converted into 16 dimensional vector space by considering a overlap of 4 between the samples of consecutive blocks. E.g. the first vector was from sample 1 to 16, whereas second vector was from 13 to 28 and the third from 25 to 40 and so on. The code books were then generated similarly as in approach A.

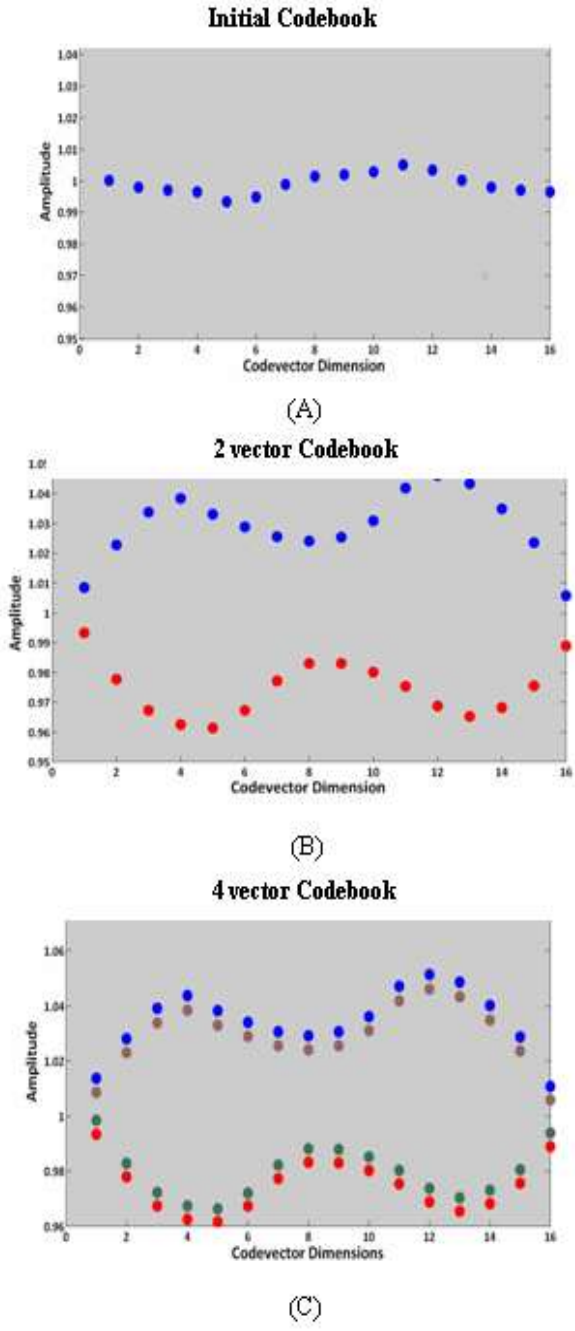

Fig. 4 Generation of 4 vector codebook using LBG

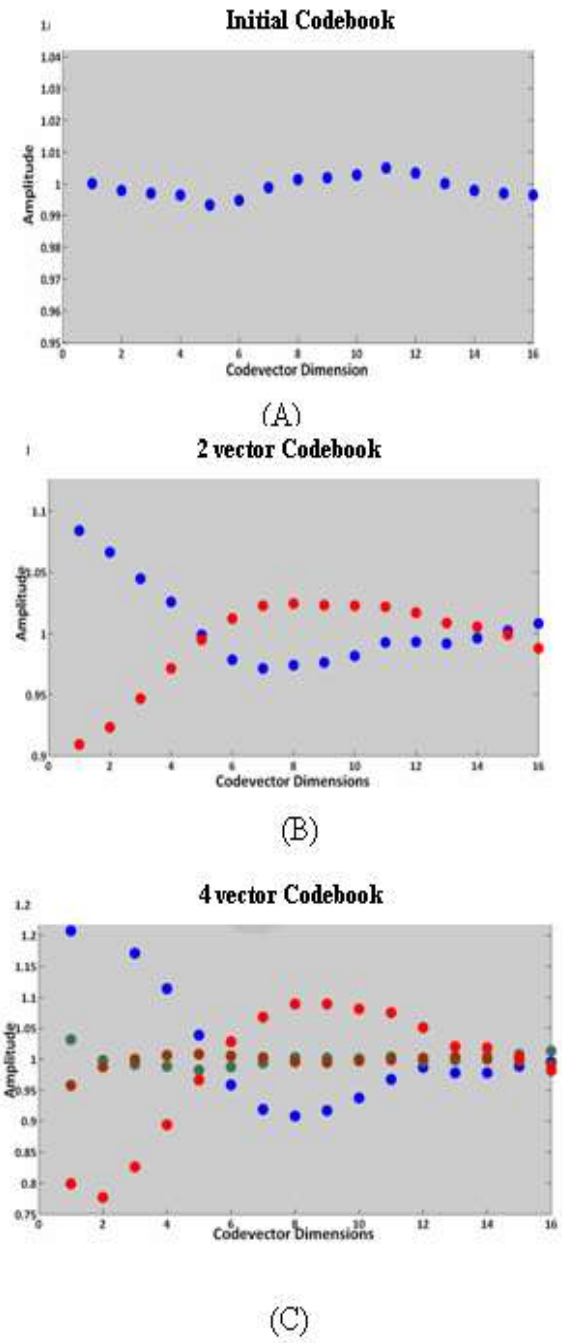

Fig. 5 Generation of 4 vector codebook using KFCG

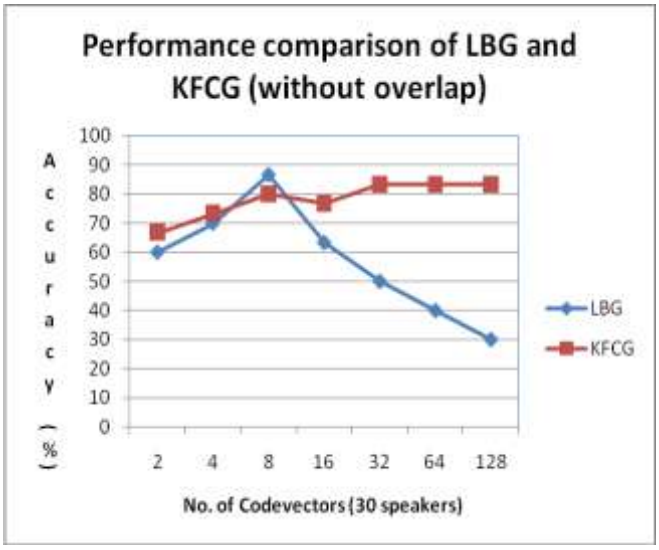

Fig 6 Performance comparison of LBG (distortion 0.005) and KFCG without overlap (30 speakers)

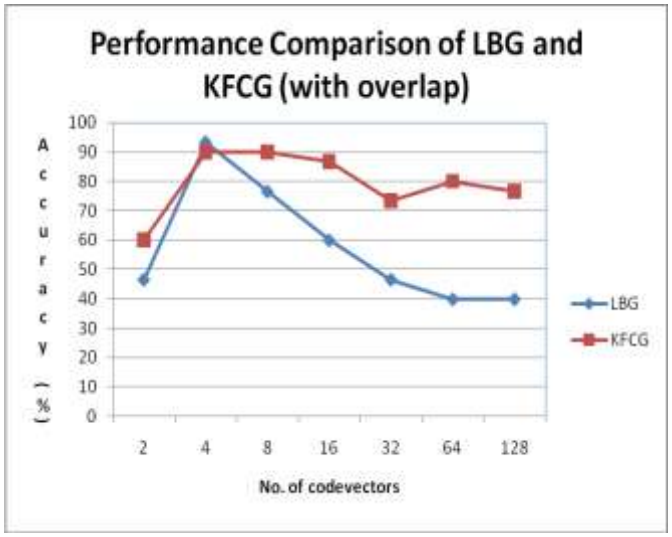

Fig 7 Performance comparison of LBG (distortion 0.005) and KFCG with overlap (30 speakers)

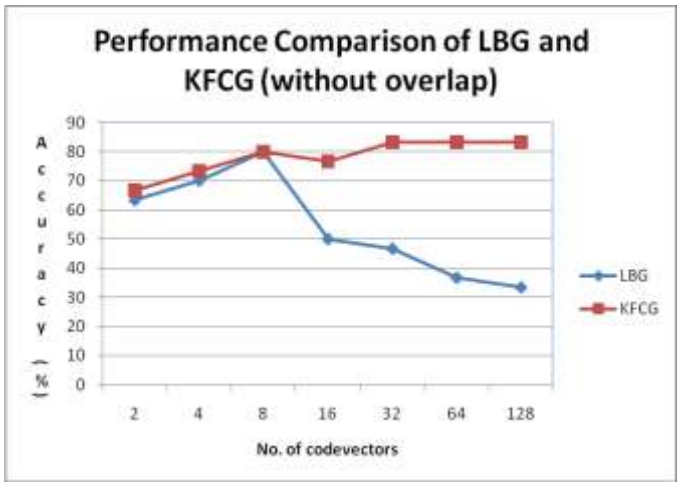

Fig 8 Performance comparison of LBG (distortion 0.01) and KFCG without overlap (30 speakers)

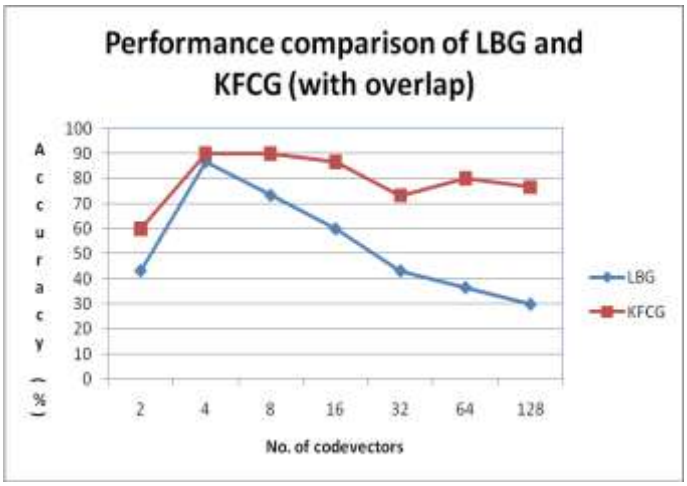

Fig 9 Performance comparison of LBG (distortion 0.01) and KFCG with overlap (30 speakers)

4. RESULTS

Basics of speech signal

The speech samples used in this work are recorded using Sound Forge 4.5. The sampling frequency is 8000 Hz (8 bit, mono PCM samples). Table 1 shows the database description. The samples are collected from different speakers. Samples are taken from each speaker in two sessions so that training model and testing data can be created. Twelve samples per speaker are taken. The samples recorded in one session are kept in database and the samples recorded in second session are used for testing.

Table 1 Database Description

Parameter	Sample characteristics
Language	English
No. of Speakers	30
Speech type	Read speech
Recording conditions	Normal. (A silent room)
Sampling frequency	8000 Hz
Resolution	8 bps

The algorithms are implemented using MATLAB 7.7.0. Fig. 4 shows the generation of 4 code vector codebook for the 16 dimensional vector space using the LBG algorithm for one of the speech sample. The code vectors thus formed are the feature

vectors used in the training phase. Fig. 4(A) shows the initial codebook which is the centroid of the entire set of training vectors. As can be seen the codebook size is 1 with a dimension of 16. Fig. 4(B) shows the two vector codebook obtained after splitting. Fig. 4(C) shows the final codebook of size 4. Fig. 5 shows the generation of 4 code vector codebook for the 16 dimensional vector space using the KFCG algorithm. The code vectors thus formed are the feature vectors used in the training phase. Fig. 5(A) shows the initial codebook which is the centroid of the entire set of training vectors. As can be seen the codebook size is 1 with a dimension of 16. Fig. 5(B) shows the two vector codebook obtained after splitting the first cluster and comparing Fig. 5(C) shows the final codebook of size 4.

The feature vectors of all the reference speech samples are stored in the database in the training phase. In the matching phase, the test sample that is to be identified is taken and similarly processed as in the training phase to form the feature vector. The stored feature vector which gives the minimum Euclidean distance with the input sample feature vector is declared as the speaker identified.

Fig. 6 shows the results obtained for text-dependent system by varying the number of feature vectors (code vectors) without overlap for a sample set of 30 speakers. As seen from the figure, for text-dependent samples, maximum accuracy is achieved with 4 feature vectors for LBG (distortion of 0.005). For LBG the accuracy decreases with the increase in the number of feature vectors. For KFCG the results are better and consistent. Accuracy does not drop as the number of feature vectors are increased. Fig. 7 shows the results obtained for text-dependent identification by varying the number of features for a sample set of 30 speakers with overlap. As seen from the figure, the results are better compared to without overlap. Again here also the performance of KFCG is better than LBG. Fig. 8 shows the performance comparison of LBG (distortion of 0.01) and KFCG without overlap. KFCG gives far better results than LBG. Fig. 9 shows the performance comparison of LBG (distortion of 0.01) and KFCG with overlap. As seen from the curves KFCG again gives far better results than LBG. As KFCG algorithm for codebook generation is based on comparison it is less complex and very fast compared to LBG which needs Euclidean distance calculations. For LBG the number of calculations required for generating the codevectors by Euclidean distance comparison for a 16-dimensional vector (16 additions + 16 Multiplications + 16 comparisons) are much more than KFCG (16 comparisons). This reduces computational time by a factor ten.

5. CONCLUSION

Very simple techniques based on the lossy compression using vector quantization have been introduced. The results show that accuracy decreases as the number of feature vectors are increased with or without overlap for LBG. For KFCG, the results are consistent and also accuracy increases with the increase in the number of feature vectors for without overlap approach. Also KFCG is simple and faster as only simple comparisons are required as against Euclidean distance calculations for LBG.

6. REFERENCES

[19] Jyoti Singhai, "Automatic Speaker Recognition :An Approach using DWT based Feature Extraction and Vector Quantization", IETE Technical Review, vol. 24, No 5, pp 395-402, September-October 2007

[1] Lawrence Rabiner, Biing-Hwang Juang and B.Yegnanarayana, "Fundamental of Speech Recognition", Prentice-Hall, Englewood Cliffs, 2009.

[2] S Furui, "50 years of progress in speech and speaker recognition research", ECTI Transactions on Computer and Information Technology, Vol. 1, No.2, November 2005.

[3] D. A. Reynolds, "An overview of automatic speaker recognition technology," *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process. (ICASSP'02)*, 2002, pp. IV-4072–IV-4075.

[4] Joseph P. Campbell, Jr., Senior Member, IEEE, "Speaker Recognition: A Tutorial", Proceedings of the IEEE, vol. 85, no. 9, pp. 1437-1462, September 1997.

[5] F. Bimbot, J.-F. Bonastre, C. Fredouille, G. Gravier, I. Magrin-Chagnolleau, S. Meignier, T. Merlin, J. Ortega-García, D.Petrovska-Delacrétaz, and D. A. Reynolds, "A tutorial on text-independent speaker verification," EURASIP J. Appl. Signal Process., vol. 2004, no. 1, pp. 430–451, 2004.

[6] D. A. Reynolds, "Experimental evaluation of features for robust speaker identification," *IEEE Trans. Speech Audio Process.*, vol. 2, no. 4, pp. 639–643, Oct. 1994.

[7] Tomi Kinnunen, Evgeny Karpov, and Pasi Fr¨anti, "Realtime Speaker Identification", ICSLP2004.

[8] Marco Grimaldi and Fred Cummins, "Speaker Identification using Instantaneous Frequencies", IEEE Transactions on Audio, Speech, and Language Processing, vol., 16, no. 6, August 2008.

[9] Zhong-Xuan, Yuan & Bo-Ling, Xu & Chong-Zhi, Yu. (1999). "Binary Quantization of Feature Vectors for Robust Text-Independent Speaker Identification" in *IEEE Transactions on Speech and Audio Processing, Vol. 7, No. 1*, January 1999. IEEE, New York, NY, U.S.A.

[10] R. M. Gray.: 'Vector quantization', *IEEE ASSP Marg.*, pp. 4-29, Apr. 1984.

[11] Y. Linde, A. Buzo, and R. M. Gray.: 'An algorithm for vector quantizer design," *IEEE Trans. Commun.*', vol. COM-28, no. 1, pp. 84-95, 1980.

[12] A. Gersho, R.M. Gray.: '*Vector Quantization and Signal Compression'*, Kluwer Academic Publishers, Boston, MA, 1991.

[13]F. K. Soong, et. al., "A vector quantization approach to speaker recognition", At & T Technical Journal, 66, pp. 14-26, 1987.

[14] A. E. Rosenberg and F. K. Soong, "Evaluation of a vector quantization talker recognition system in text independent and text dependent models", Computer Speech and Language 22, pp. 143-157, 1987.

[15] Jeng-Shyang Pan, Zhe-Ming Lu, and Sheng-He Sun.: 'An Efficient Encoding Algorithm for Vector Quantization Based on Subvector Technique', *IEEE Transactions on image processing*, vol 12 No. 3 March 2003.

[16] F. Soong, E. Rosenberg, B. Juang, and L. Rabiner, "A Vector Quantization Approach to Speaker Recognition", AT&T Technical Journal, vol. 66, March/April 1987, pp. 1426.

[17] Md. Rashidul Hasan, Mustafa Jamil, *Md. Golam Rabbani Md. Saifur Rahman* , "Speaker Identification using Mel Frequency Cepstral Coefficients", 3rd International Conference on Electrical & Computer Engineering ICECE held at Dhaka, Bangladesh , 28-30 December 2004.

[18] Poonam Bansal, Amrita Dev, Shail Bala Jain, "Automatic Speaker Identification using Vector Quantization", Asian Journal of Information Technology 6 (9): 938-942, 2007.

[20] H. B. Kekre, Tanuja K. Sarode, "Speech Data Compression using Vector Quantization", WASET International Journal of Computer and Information Science and Engineering (IJCISE), Fall 2008, Volume 2, Number 4, pp.: 251-254, 2008. http://www.waset.org/ijcise.

[21] H. B. Kekre, Tanuja K. Sarode, "New Fast Improved Codebook Generation Algorithm for Color Images using Vector Quantization," International Journal of Engineering and Technology, vol.1, No.1, pp. 67-77, September 2008.

[22] H. B. Kekre, Tanuja K. Sarode, "Fast Codebook Generation Algorithm for Color Images using Vector Quantization," International Journal of Computer Science and Information Technology, Vol. 1, No. 1, pp: 7-12, Jan 2009.

[23] H. B. Kekre, Tanuja K. Sarode, "An Efficient Fast Algorithm to Generate Codebook for Vector Quantization," First International Conference on Emerging Trends in Engineering and Technology, ICETET-2008, held at Raisoni College of Engineering, Nagpur, India, 16-18 July 2008, Avaliable at online IEEE Xplore.

[24] H B Kekre, Vaishali Kulkarni, "Speaker Identification by using Vector Quantization", International Journal of Engineering Science and Technology, May 2010 edition.

Author Biographies

Dr. H. B. Kekre has received B.E. (Hons.) in Telecomm. Engg. from Jabalpur University in 1958, M.Tech (Industrial Electronics) from IIT Bombay in 1960, M.S.Engg. (Electrical Engg.) from University of Ottawa in 1965 and Ph.D. (System Identification) from IIT Bombay in 1970. He has worked Over 35 years as Faculty of Electrical Engineering and then HOD Computer Science and Engg. at IIT Bombay. For last 13 years worked as a Professor in Department of Computer Engg. at Thadomal Shahani Engineering College, Mumbai. He is currently Senior Professor working with Mukesh Patel School of Technology Management and Engineering, SVKM's NMIMS University, Vile Parle(w), Mumbai, INDIA. He has guided 17 Ph.D.s, 150 M.E./M.Tech Projects and several B.E./B.Tech Projects. His areas of interest are Digital Signal processing, Image Processing and Computer Networks. He has more than 300 papers in National / International Conferences / Journals to his credit. Recently nine students working under his guidance have received best paper awards. Recently two of his students have completed Ph.D. Currently he is guiding eight Ph.D. students. He is member of ISTE and IETE.

Vaishali Kulkarni has received B.E in Electronics Engg. from Mumbai University in 1997, M.Tech (Electronics and Telecom) from Mumbai University in 2006. Presently she is pursuing Ph. D from NMIMS University. She has a teaching experience of more than 8 years. She is Assistant Professor in telecom Department in MPSTME, NMIMS University. Her area of interest include Speech processing: Speech and Speaker Recognition.