MULTI-FRAME GMM-BASED BLOCK QUANTISATION FOR DISTRIBUTED SPEECH RECOGNITION UNDER NOISY CONDITIONS

Stephen So

School of Engineering, Griffith University,
PMB 50, Gold Coast Mail Centre,
Gold Coast, QLD, Australia, 9726.
s.so@griffith.edu.au

Kuldip K. Paliwal

School of Microelectronic Engineering,
Griffith University, Brisbane,
QLD, Australia, 4111.
k.paliwal@griffith.edu.au

ABSTRACT

In this paper, we report on the recognition accuracy of the multi-frame GMM-based block quantiser for the coding of MFCC features in a distributed speech recognition framework under varying noise conditions. All experiments were performed using the ETSI Aurora-2 connected-digits recognition task. For comparison, we have also investigated other quantisation schemes such as the memoryless GMM-based block quantiser, the unconstrained vector quantiser, and non-uniform scalar quantisers. The results show that the rate-distortion effi ciency of the quantiser is a factor in determining the level of recognition accuracy at low to medium levels of additive noise. For high levels of additive noise, the influence of rate-distortion effi ciency diminishes and the recognition accuracy becomes dependent on the recognition features.

1. INTRODUCTION

With the increase in popularity of remote and wireless devices such as personal digital assistants (PDAs) and cellular phones, there has been a growing interest in applying automatic speech recognition (ASR) technology in the context of mobile communication systems. Speech recognition can facilitate consumers in performing common tasks, which have traditionally been accomplished via buttons or pointing devices, such as making a call through voice dialing or entering data into their PDAs via spoken commands and sentences. Some of the issues that arise when implementing ASR on mobile devices include: computational and memory constraints of the mobile device; network bandwidth utilisation; and robustness to noisy operating conditions.

Mobile devices generally have limited storage and processing ability which makes implementing a full on-board ASR system impractical. The solution to this problem is to perform the complex speech recognition task on a remote server that is accessible via the network. In *Distributed Speech Recognition* (DSR), shown in Figure 1, the ASR system is distributed between the client and server. Here, the feature extraction of speech is performed at the client. These ASR features are compressed and transmitted to the server via a dedicated channel, where they are decoded and input into the ASR backend. Studies have shown that DSR generally performs better than *network speech recognition* (NSR), where features are extracted from coded speech at the server end. This is because in NSR, speech coders aim for optimal perceptual quality and this does not necessarily correlate to optimal recognition performance [1].

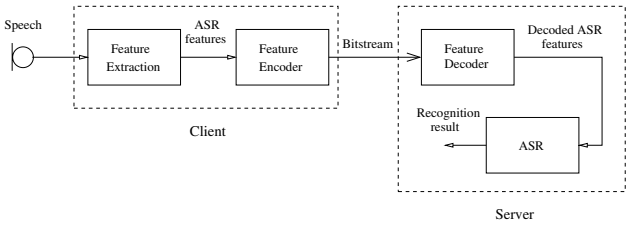

Fig. 1. A typical distributed speech recognition system

Various schemes for compressing the ASR features have been proposed in the literature. Digalakis *et al.* in [2] evaluated the use of uniform and non-uniform scalar quantisers as well as product code vector quantisers for compressing Mel frequency-warped cepstral coeffi cients (MFCCs) between 1.2 and 10.4 kbps. They concluded that split vector quantisers achieved word error rates (WER) similar to that of scalar quantisers while requiring less bits. Also, scalar quantisers with non-uniform bit allocation performed better than those with uniform bit allocation. Ramaswamy and Gopalakrishnan [3] investigated the application of ML searched multistage vector quantisers with one-step linear prediction operating at 4 kbps. Transform coding, based on the discrete cosine transform (DCT), was investigated in [4] at 4.2 kbps and in [5] which used a two-dimensional DCT. The ETSI DSR standard [6] uses split vector quantisers to compress the MFCC vectors at 4.4 kbps. Srinivasamurthy *et al.* in [1] exploited correlation across consecutive MFCC features by using a DPCM scheme followed by entropy coding.

In [9], we proposed the use of the multi-frame GMM-based block quantiser [8] for quantising MFCC vectors and showed that it achieved a more graceful degradation in recognition accuracy at lower bitrates when compared with other scalar quantiser-based schemes. The advantages of the multi-frame GMM-based block quantiser over the vector quantiser include: *bitrate scalability*, where the bitrate can be changed without the need to re-train the quantiser; and *fixed computational complexity* for all bitrates. However, those experiments were performed on clean speech from the ETSI Aurora-2 database only. The effect of additive noise on the recognition performance is important and relevant to DSR systems, as a mobile operator will be immersed in background environmental noise that will also be captured by his/her device. The Aurora-2 recognition task provides various types of back-

ground noise that are added to the clean speech at varying SNR levels. Therefore, in this paper, we extend the work presented in [9] by evaluating the multi-frame GMM-based block quantiser for quantising MFCCs with varying levels of noise added to the test speech. In addition to this, we compare its performance with other quantisation schemes, which include the memoryless GMM-based block quantiser, the unconstrained vector quantiser, and non-uniform scalar quantisers.

2. THE MULTI-FRAME GMM-BASED BLOCK QUANTISER

This coding scheme is based on the one proposed by Subramaniam and Rao in [7] for the coding of speech line spectral frequencies (LSF) , where a Gaussian mixture model (GMM) is used to parametrically model the probability density function (PDF) of the source and block quantisers are then designed for each Gaussian mixture component. In [8], we proposed a modified scheme which used vectors formed from p concatenated frames, in order to exploit interframe correlation. Therefore, if the length of the MFCC frame is n, then the dimensions of the vectors processed will be np. MFCCs are calculated frame-wise from speech and there is considerable overlap between successive frames. Generally, there will be high correlation between consecutive frames [1]. Therefore, we have chosen to use multi-frame GMM-based block quantisation ($p = 5$) to exploit this correlation. For more details on this coding scheme as well as its memory and computational requirements, the reader is referred to [7, 8].

2.1. Quantiser training

The PDF model and Karhunen-Loéve transform (KLT) orthogonal matrices are the only static and bitrate-independent parameters of the GMM-based block quantiser. These only need to be calculated once (training) and stored at the client encoder and server decoder.

The PDF model, which is in the form of a GMM, is initialised by applying the K-means algorithm on the training vectors where m mixture components are produced, each represented by a mean, $\boldsymbol{\mu}$, a covariance matrix, $\boldsymbol{\Sigma}$, and weight, c. These form the initial parameters for the GMM estimation procedure. Using the Expectation-Maximisation (EM) algorithm, the maximum likelihood estimate of the parametric model is computed iteratively until the log likelihood converges, where a final set of means, covariance matrices, and weights are produced.

2.2. Encoding process

Assuming that there are b_{tot} bits available for coding each vector (for an average bitrate of b_{tot}/np bps), these need to be allocated to the block quantisers of each mixture component, and then further allocated to the vector components within each block quantiser. Because of the use of closed-form expressions [7], this bit allocation can be done 'on-the-fly'.

To quantise a vector, $\boldsymbol{x}$, it is first coded and decoded by each of the m block quantisers to produce a series of candidate reconstructed vectors, $\{\widehat{\boldsymbol{x}}_i\}_{i=1}^{m}$. The distortions between these reconstructed vectors and original are then calculated, $\{d(\boldsymbol{x} - \widehat{\boldsymbol{x}}_i)\}_{i=1}^{m}$. The above procedure is performed for all mixture components in the system and the mixture component, k, which gives the *minimum distortion* is chosen:

$$k = \underset{i}{\operatorname{argmin}}\, d(\boldsymbol{x} - \widehat{\boldsymbol{x}}_i) \tag{1}$$

In the case of coding MFCC vectors, we use the mean-squared-error (MSE) as the distortion measure for selecting the appropriate block quantiser.

3. EXPERIMENTAL SETUP

We have evaluated the recognition performance of the memoryless and multi-frame GMM-based block quantiser, the unconstrained vector quantiser, and non-uniform scalar quantiser using the ETSI Aurora-2 connected digits recognition task [10]. The training speech set consists of 8440 utterances while the test set comprises 4004 utterances, with 1001 utterances assigned to each of the four noise types [10]. Quantiser codebook and ASR training was performed on the MFCCs derived from clean speech data while the quantisation and recognition was performed using MFCC vectors derived from the noisy speech of test set A. There are four types of noises (babble, car, subway, exhibition) provided for test set A in the Aurora-2 database. The amount of added noise is varied based on the signal-to-noise ratios (SNRs) of 20, 15, 10, 5, 0 dB.

The ETSI DSR standard Aurora frontend [6] was used for the MFCC feature extraction. As a slight departure from the ETSI DSR standard, we have used 12 MFCCs (excluding the zeroth cepstral coefficient, c_0, and logarithmic frame energy, $\log E$) as the feature vectors to be quantised. This was done to maintain consistency in the coding scheme as c_0 and $\log E$ are sensitive to changes in recording level of a speech utterance and are generally coded independently [6, 4, 3]. Cepstral liftering was applied to the MFCCs using the following window function, $w(n)$:

$$w(n) = 1 + \frac{L}{2}\sin\left(\frac{\pi n}{L}\right) \quad \text{where } n = 1, 2, \ldots, L \tag{2}$$

where L is the feature length. This was done to prevent a skewed bit allocation from occurring as bits are allocated on the basis of variance. These MFCC vectors are then quantised and transmitted. After decoding the 12 MFCC feature vectors, cepstral mean subtraction (CMS) is applied and they are then concatenated with their corresponding delta and acceleration coefficients. The final feature vector dimension for the ASR system is 36. Whole word HMMs are used for modelling the digits with the following parameters [10]:

- 16 states per word (with two dummy states at beginning and end);
- left-to-right topology without skips over states;
- three Gaussian mixtures per state; and
- diagonal covariance matrices

For the scalar quantisation experiment, each MFCC was quantised using a Gaussian Lloyd-Max scalar quantiser whose bit allocation was calculated using the high resolution formula given in [11]. We have chosen this method over the WER-based greedy algorithm of [2] because of its computational simplicity.

In the training of the GMM-based block quantiser, 20 iterations of the EM algorithm were used to generate a 16 mixture component GMM. For the multi-frame GMM-based block quantiser, 5 MFCC feature vectors were concatenated to form vectors of dimension 60. We use the following abbreviations to refer to each quantisation scheme: GMM-5 (5 frame GMM-based block quantiser), VQ (unconstrained vector quantiser), GMM-1 (memoryless GMM-based block quantiser), SQ (non-uniform scalar quantiser).

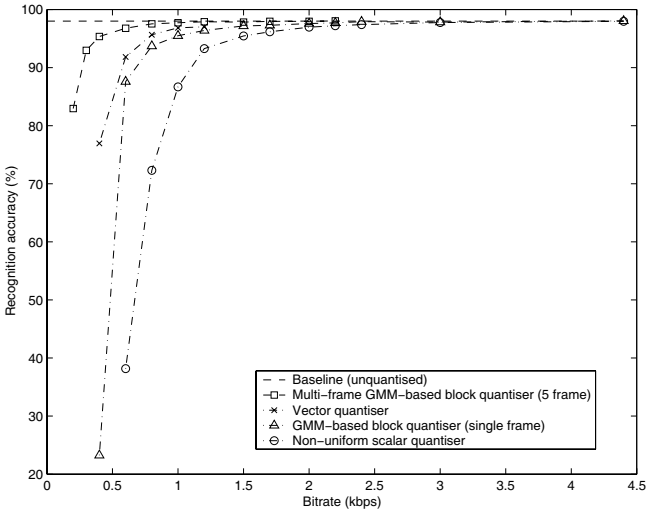

Fig. 2. Average word recognition accuracy on clean speech as a function of bitrate

4. RESULTS AND DISCUSSION

Figure 2 shows the average word recognition accuracy of all quantisation schemes as a function of bitrate for clean speech[1]. This is essentially similar to the fi gure in [8]. The average word recognition accuracy when using the original, unquantised MFCCs derived from clean speech is 98.01%. We observe that the multi-frame GMM-based block quantiser maintains acceptable recognition performance at very low bitrates. Because it exploits correlation across multiple frames, it is expected to perform better than the vector quantiser, which operates only on single frames.

Tables 1, 2, 3, and 4 show the word recognition accuracy at 0.6 kbps when speech is corrupted with subway, babble, car, and exhibition noise, respectively at varying SNRs for the different quantisation schemes. We can see that at low to medium levels of noise (20 and 15 dB), the multi-frame GMM-based block quantiser achieves the highest recognition accuracy, followed by the vector quantiser, memoryless GMM-based block quantiser, and then the scalar quantiser. Note that this is consistent with the rate-distortion effi ciencies of each scheme. For higher levels of noise (SNRs of 10, 5, 0 dB), the difference in word recognition accuracies between the various schemes becomes smaller. In this region, we note that the noise-robustness of the underlying features influences the recognition performance more than the distortion introduced by the quantiser. The differences in recognition accuracy between the various quantisation schemes can be visualised in Figure 3, which plots the recognition accuracy against the SNR.

5. CONCLUSION

In this paper, we have evaluated the multi-frame GMM-based block quantiser for quantising MFCC features in a DSR scenario, where speech has been corrupted by additive noise, and also compared this scheme with other quantisation schemes. From the results,

[1]The word recognition accuracies for each of the four subsets of test set A have been averaged.

Table 1. Word recognition accuracy for speech corrupted with subway noise at varying SNRs (in dB) at 0.6 kbps.

Quantisation scheme	Recognition accuracy (in %)					
	∞ dB	20 dB	15 dB	10 dB	5 dB	0 dB
Unquantised	98.07	94.14	86.67	66.17	38.62	23.43
GMM-5	96.38	88.73	74.33	48.17	26.93	18.73
VQ	94.40	82.22	71.29	48.30	26.44	15.84
GMM-1	84.41	77.56	64.14	44.24	25.15	16.43
SQ	8.32	8.29	8.29	8.26	8.14	8.11

Table 2. Word recognition accuracy for speech corrupted with babble noise at varying SNRs (in dB) at 0.6 kbps

Quantisation scheme	Recognition accuracy (in %)					
	∞ dB	20 dB	15 dB	10 dB	5 dB	0 dB
Unquantised	98.07	95.92	90.69	74.94	45.56	22.91
GMM-5	97.10	91.54	77.90	53.78	30.05	18.02
VQ	91.66	83.25	72.79	52.39	29.96	16.35
GMM-1	89.94	76.12	62.61	43.02	25.94	15.90
SQ	8.25	8.22	8.16	8.13	8.16	8.13

Table 3. Word recognition accuracy for speech corrupted with car noise at varying SNRs (in dB) at 0.6 kbps

Quantisation scheme	Recognition accuracy (in %)					
	∞ dB	20 dB	15 dB	10 dB	5 dB	0 dB
Unquantised	97.97	95.59	88.88	68.42	36.09	20.61
GMM-5	96.51	89.02	72.89	44.92	24.22	17.00
VQ	91.92	84.22	70.00	44.02	23.11	15.81
GMM-1	87.59	76.86	59.86	36.92	21.00	14.70
SQ	8.29	8.23	8.29	8.26	8.23	8.23

Table 4. Word recognition accuracy for speech corrupted with exhibition noise at varying SNRs (in dB) at 0.6 kbps

Quantisation scheme	Recognition accuracy (in %)					
	∞ dB	20 dB	15 dB	10 dB	5 dB	0 dB
Unquantised	97.93	93.34	85.56	62.79	33.42	19.01
GMM-5	97.13	89.60	73.25	43.04	24.13	16.94
VQ	92.35	84.60	70.01	42.58	22.80	14.47
GMM-1	87.41	78.96	59.86	34.16	20.18	12.74
SQ	7.93	7.87	7.87	7.84	7.81	7.81

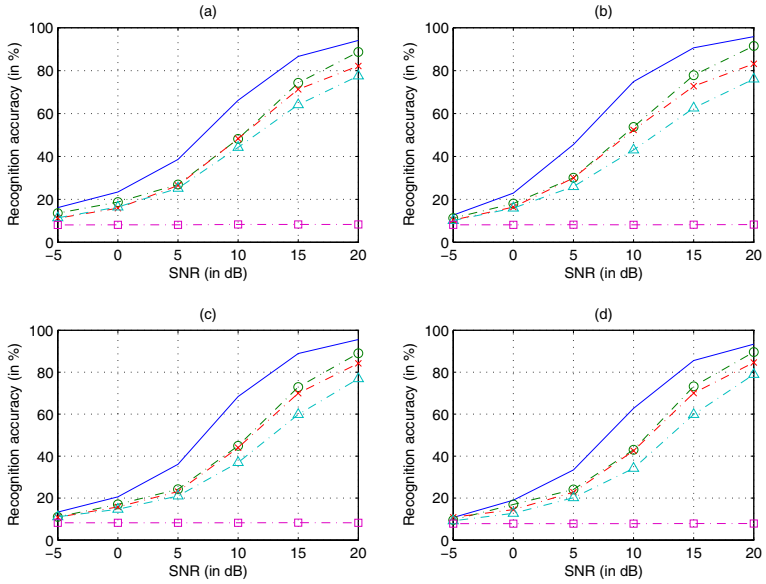

Fig. 3. Plot of recognition accuracy versus SNR for all quantisation schemes at 0.6 kbps: (a) subway noise; (b) babble noise; (c) car noise; and (d) exhibition noise. (Solid lines are unquantised, circles are GMM-5, crosses are VQ, triangles are GMM-1, squares are SQ)

it was observed that at low to medium levels of noise, the rate-distortion efficiency of the quantiser was an important factor in determining the recognition accuracy. However, when there was a high level of noise, the influence of the quantiser distortion diminished. That is, there was no difference between the multi-frame and memoryless schemes. This suggests that designing a DSR system for noisy environments requires both a noise-robust feature set and efficient quantisation scheme.

6. REFERENCES

[1] N. Srinivasamurthy, A. Ortega and S. Narayanan, "Efficient scalable encoding for distributed speech recognition", submitted to *IEEE Trans. Speech and Audio Processing*, 2003. Available: http://biron.usc.edu/~snaveen/papers/Scalable_DSR.pdf

[2] V.V. Digalakis, L.G. Neumeyer and M. Perakakis, "Quantization of cepstral parameters for speech recognition over the world wide web", *IEEE J. Select. Areas Commun.*, Vol. 17, No. 1, pp. 82–90, Jan 1999.

[3] G.N. Ramaswamy and P.S. Gopalakrishnan, "Compression of acoustic features for speech recognition in network environments", in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Processing*, 1998, pp. 977–980.

[4] I. Kiss and P. Kapanen, "Robust feature vector compression algorithm for distributed speech recognition", in *Proc. Eurospeech*, 1999, pp. 2183–2186.

[5] Q. Zhu and A. Alwan, "An efficient and scalable 2D DCT-based feature coding scheme for remote speech recognition", in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Processing*, Vol. 1, Aug 2001, pp. 113-116.

[6] "Speech processing, transmission and quality aspects (STQ); Distributed speech recognition; Front-end feature extraction algorithm; Compression algorithms", Tech. Rep. Standard ES 201 108, European Telecommunications Standards Institute (ETSI), April 11 2000.

[7] A.D. Subramaniam and B.D. Rao, "PDF optimized parametric vector quantization of speech line spectral frequencies", *IEEE Trans. Speech Audio Processing*, Vol. 11, No. 2, pp. 130–142, Mar. 2003.

[8] S. So and K.K. Paliwal, "Multi-frame GMM-based block quantisation of line spectral frequencies", *Speech Commun.*, vol. 47, pp. 265–276, 2005.

[9] K.K. Paliwal and S. So, "Scalable distributed speech recognition using multi-frame GMM-based block quantization", in *Proc. Int. Conf. Spoken Language Processing*, Jeju, Korea, 2004.

[10] H.G. Hirsch and D. Pearce, "The Aurora experimental framework for the performance evaluation of speech recognition systems under noisy conditions", *ISCA ITRW ASR2000*, Paris, France, Sept. 2000.

[11] J.J.Y. Huang and P.M. Schultheiss, "Block quantization of correlated Gaussian random variables", *IEEE Trans. Commun. Syst.*, Vol. CS-11, pp. 289–296, Sept. 1963.

[12] E.A. Riskin, "Optimal bit allocation via the generalized BFOS algorithm", *IEEE Trans. Inform. Theory*, 37(2), pp. 400–402, 1991.