EFFICIENT BLOCK CODING OF IMAGES USING GAUSSIAN MIXTURE MODELS

Stephen So and Kuldip K. Paliwal

School of Microelectronic Engineering,
Griffith University, Brisbane, Australia, 4111.
s.so@me.gu.edu.au, k.paliwal@me.gu.edu.au

ABSTRACT

An efficient method for block coding of speech was presented by Rao and Subramaniam in [7]. An adaptation of this method for the use in image coding is presented in this paper. The probability density function (PDF) of the image blocks is estimated and modelled as multivariate Gaussian mixtures using the k-means and Expectation-Maximisation (EM) algorithms. This parametric model is incorporated into the block transform coder proposed in [7], where the complexity is dependent on the number of mixtures rather than the bit rate. The block coder, being rate-independent, calculates the required bit allocations based on the parametric model for the desired bit rate, making it bit scalable and thus adaptable for Internet applications. The results show a PSNR improvement of 5–10 dB at high bit rates and 2–4 dB at medium to low bit rates over the optimal block transform coder of Huang and Schultheiss [3].

1. INTRODUCTION

The performance of any quantiser depends on the probability density function (PDF) of the source. Therefore it is necessary to either make assumptions of or to estimate the PDF of the source in order to design efficient quantisers. Lloyd [4] and Max [5] independently developed methods for designing non-uniform scalar quantisers to efficiently code Gaussian sources. Huang and Schultheiss in [3] extended the Lloyd-Max scalar quantisers in their proposed block quantiser scheme where blocks of correlated Gaussian random variables are efficiently coded. They showed that the use of an orthogonal transform or Karhunen-Loeve Transform (KLT) to decorrelate the blocks improved the efficiency of the scalar quantisers. This led to the development of block-based transform coders where the bit allocation is either calculated (using the formula given in [3]) or determined via perceptual experiments (as in the JPEG image standard) [9].

These block coders proposed in [3] perform best on vectors that are generated from a Gaussian source but become suboptimal when the source PDF does not match a Gaussian. Performance improvements can therefore be realised

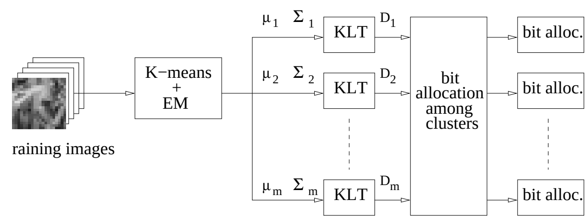

Figure 1: PDF Estimation and Bit Allocation from Training Images

by estimating the PDF of the source and designing quantisers that match this PDF. A method based on non-parametric estimation of PDFs was proposed by Ortega and Vetterli in [6] where the PDFs were estimated using a piecewise linear histogram. Subramaniam and Rao in [7] also proposed a block coder for speech where the PDF was modelled by a Gaussian Mixture Model (GMM). This parametric method has many advantages which include low complexity, rate independence, bit scalability, etc. [7]. Therefore, we have adapted this parametric coder for the application of images.

2. THE GMM-BASED BLOCK CODER

The block coder consists of two parts: *PDF estimation/bit allocation* and the *GMM block quantiser*. Figures 1 and 2 show schematics of the PDF estimation and bit allocation procedure and GMM block quantiser respectively which are similar to those that appear in [8]. Like the JPEG standard, images are operated on as blocks of pixels with dimension $8 \times 8 = 64$ since most of the correlation generally is contained within the vicinity of 8 pixels (assuming an isotropic source model).

2.1. PDF Estimation using Gaussian Mixture Models

GMMs can be used for modelling any arbitrary distribution using multivariate Gaussians as basis functions. The PDF model, P, to be estimated is represented by a mixture of

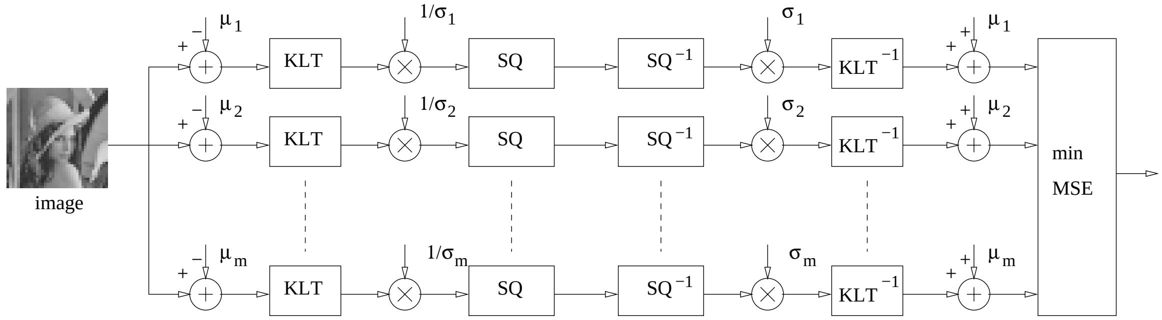

Figure 2: The GMM-based Block Quantiser

Gaussians $\mathcal{N}(\boldsymbol{x}; \boldsymbol{\mu}; \boldsymbol{\Sigma})$

$$P(\boldsymbol{X}|\lambda) = \sum_{i=1}^{m} c_i \mathcal{N}_i(\boldsymbol{x}; \boldsymbol{\mu}_i; \boldsymbol{\Sigma}_i) \quad (1)$$

$$\lambda = [m, c_i, \ldots, c_m, \mu_1, \ldots, \mu_m, \Sigma_1, \ldots, \Sigma_m] \quad (2)$$

$$\mathcal{N}(\boldsymbol{x}; \boldsymbol{\mu}; \boldsymbol{\Sigma}) = \frac{1}{(2\pi)^{\frac{n}{2}} |\boldsymbol{\Sigma}|^{\frac{1}{2}}} e^{-\frac{1}{2}(\boldsymbol{x}-\boldsymbol{\mu})^T \boldsymbol{\Sigma}^{-1}(\boldsymbol{x}-\boldsymbol{\mu})} \quad (3)$$

where $\boldsymbol{X}$ are the source vectors, m is the number of mixtures, and n is the dimension of the vectors. λ is the set of model parameters consisting of $c_i, \boldsymbol{\mu}_i, \boldsymbol{\Sigma}_i$ which are the weight, mean, and covariance matrix of the ith mixture respectively. Note the words 'mixture' and 'cluster' will be used interchangeably in this paper.

The parametric model, λ, is initialised by applying the K-means algorithm on the training image vectors where m clusters are produced, each represented by a mean or centroid, $\boldsymbol{\mu}$, a covariance matrix, $\boldsymbol{\Sigma}$, and a cluster 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 and a final set of means, covariance matrices, and weights are produced.

Since the GMM parameters for the estimated PDF are properties of the source and thus independent of the bit rate, this procedure only needs to be done once and stored for later use.

2.2. Karhunen-Loeve Transform

The Karhunen-Loeve transform (KLT) decorrelates vectors, $\boldsymbol{x}$, by multiplying with an orthogonal matrix, $\boldsymbol{K}$, which is known to diagonalise the covariance matrix of those vectors, $\boldsymbol{\Sigma}$. That is, $\boldsymbol{D} = \boldsymbol{K}^T \boldsymbol{\Sigma} \boldsymbol{K}$, where $\boldsymbol{D}$, being the covariance matrix of the output, is diagonal. Huang and Schultheiss in [3] showed that the efficiency of a block of scalar quantisers is optimum when operating on decorrelated data. In essence, the KLT orthogonal matrix acts as a rotation matrix, aligning the axis of the source PDF that has the largest variance with the first dimension or principal component. This has the effect of removing correlation across dimensions as well as 'compacting' variance or energy into the principal component [1].

A KLT is performed on the classified vectors of each GMM cluster to find the set of eigenvalues that represent the variances of the decorrelated vectors. These will be used in the bit allocation procedure of the scalar quantiser design. The orthogonal matrix and eigenvalues are saved for use in the block quantiser.

Since the orthogonal matrices and eigenvalues are properties of the source model and thus independent of the bit rate, this procedure only needs to be done once and the parameters are stored for later use.

2.3. Bit Allocation

There are two types of bit allocation that are required: *intercluster bit allocation* and *intracluster bit allocation*. Since the bit allocation is not a computationally expensive procedure, it can be done 'on-the-fly' depending on the chosen bit rate. Hence scalability is a feature of the GMM-based block coder [8].

2.3.1. Intercluster Bit Allocation

With intercluster bit allocation, bits or quantiser levels need to be assigned to each of the m clusters. Following the derivation given in [7], for a fixed-rate quantiser, the total number of quantiser levels is fixed:

$$2^{b_{tot}} = \sum_{i=0}^{m-1} 2^{b_i} \quad (4)$$

where b_{tot} is the total number of bits in the bit budget, b_i is the number of bits assigned to cluster i, and m is the number of clusters. Since the cluster weights can be thought of as probabilities of occurrence of each cluster [8], the average distortion is approximated by

$$D_{tot} = \sum_{i=0}^{m-1} c_i D_i(b_i) \quad (5)$$

Using the high resolution approximation for distortion of a Lloyd-Max scalar quantiser operating on Gaussian sources,

$$D_i(b_i) = Kn\Lambda_i 2^{-2\frac{b_i}{n}} \quad (6)$$

$$\Lambda_i = \left[\prod_{k=0}^{n-1} \lambda_{i,j}\right]^{\frac{1}{n}} \quad (7)$$

$$\text{for } i = 0, 1, \ldots, m-1$$

where n is the dimension of the vectors, m is the number of clusters, $\lambda_{i,j}$ is the jth eigenvalue of cluster i, and K is a constant which is approximately equal to $\frac{\pi\sqrt{3}}{2}$ for Gaussian sources.

Using Lagrange multipliers, the average distortion can be minimised under the fixed rate constraint of equation (4), and the following bit allocation formula is derived:

$$2^{b_i} = 2^{b_{tot}} \frac{(c_i\Lambda_i)^{\frac{n}{n+2}}}{\sum_{i=0}^{m-1}(c_i\Lambda_i)^{\frac{n}{n+2}}}, \quad (8)$$

$$\text{for } i = 0, 1, \ldots, m-1$$

where c_i is the weight of cluster i.

2.3.2. Intracluster Bit Allocation

After the bits are allocated to each cluster, further bit allocation is performed to assign to each of the n components. Following the derivation presented in [3], the total number of bits is fixed:

$$b_i = \sum_{j=0}^{n-1} b_{i,j}, \quad \text{for } i = 0, 1, \ldots, m-1 \quad (9)$$

where $b_{i,j}$ is the number of bits assigned to component j of cluster i. Again, using the high resolution approximation for the distortion of a Lloyd-Max scalar quantiser, the average distortion of cluster i is given by:

$$D_i = \frac{1}{n}\sum_{j=0}^{n-1} \lambda_{i,j} K 2^{-2b_{i,j}} \quad (10)$$

$$\text{for } i = 0, 1, \ldots, m-1$$

Using Lagrange multipliers, the average distortion is minimised under the fixed rate constraint in equation (9) and the following bit allocation formula is derived:

$$b_{i,j} = \frac{b_i}{n} + \frac{1}{2}\log_2 \frac{\lambda_{i,j}}{|\mathbf{\Sigma}_i|} \quad (11)$$

$$\text{for } i = 0, 1, \ldots, m-1$$

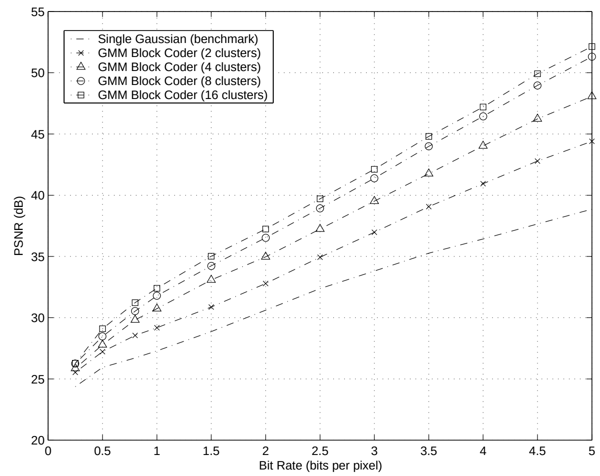

Figure 3: Distortion-Rate Characteristics for boat512.raw (512 x 512)

2.4. GMM Block Quantiser

To quantise an image vector, $\boldsymbol{x}$, using a particular cluster i, the cluster mean, $\boldsymbol{\mu}_i$ is first subtracted and decorrelated using the orthogonal matrix for that cluster. It is then divided by the standard deviation (square root of the eigenvalues) to make it unity variance and quantised using a set of n Lloyd-Max scalar quantisers as described in [3] with their respective bit allocations. The indicies from the quantiser are decoded, multiplied by the standard deviation and correlated again by an inverse KLT. The cluster mean is then added back to give the approximated vector, $\widehat{\boldsymbol{x}}_{\boldsymbol{i}}$. The distortion between this quantised vector and original is then calculated, $d(\boldsymbol{x} - \widehat{\boldsymbol{x}}_i)$. For image applications, the most popular distortion measure is mean squared error (MSE). The above procedure is performed for all clusters in the system. The cluster which gives the least distortion is chosen and the indices as well as the cluster number is transmitted or stored, $i = \arg_i \min d(\boldsymbol{x} - \widehat{\boldsymbol{x}}_i)$.

It is noted that the search complexity of the GMM block quantiser is a function of the number of clusters, m rather than the bit rate [7]. Thus it is a computational advantage over unstructured vector quantisers where the search complexity is an exponential function of the bit rate [2].

3. RESULTS

A training set of 14 images was used. Using 8×8 blocks, this translates to a training set of 28928 vectors. Lloyd-Max scalar quantisers of up to 8 bits/256 levels were used. The number of clusters was varied (2,4,8) and 20 iterations of the EM algorithm were performed.

Figure 3 shows the distortion-rate characteristic of the

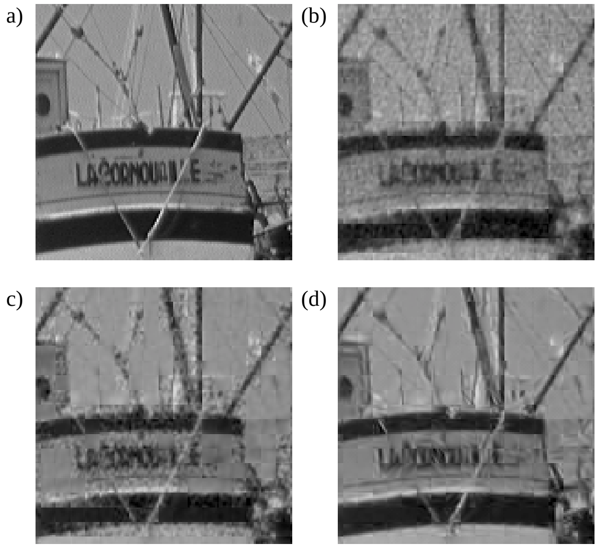

Figure 4: (a) Original image (boat512.raw) (b) KLT Block Coder at 0.5 bpp (single Gaussian) (c) GMM-based KLT at 0.5 bpp (2 Gaussians) (d) GMM-based at 0.5 bpp (16 Gaussians)

GMM-based block coder on the image 'boat', which is an 8-bit greyscale image of size 512×512. Also this image is not part of the training set. The results for a single Gaussian coder are also presented as a benchmark. This benchmark coder is based on the one proposed in [3] where a KLT is applied to decorrelate the vectors which are coded by a set of n Lloyd-Max Gaussian quantisers. It can be seen from the results that the PSNR of the GMM-based coder is at least 5 to 10 dB better at higher bit rates and at least 2 to 4 dB better at medium bit rates than the typical transform coder which makes the Gaussian PDF assumption. As the number of clusters is increased, considerable performance gains are realised since with more mixtures, the true PDF of the training data set is estimated better than with a lesser number of mixtures. The amount of intrinsic side information that is needed by the decoder is determined by the number of clusters.

Figure 4 shows a comparison of the subjective quality of the decompressed images. It can be seen that noticeable improvement is seen going to 2 clusters in 4(c) from the single Gaussian case 4(b), especially in the smoother areas of the top right part of the image. Increasing the clusters to 16 in 4(d), there is considerably less distortion in the lower part of the image, below the letters.

4. CONCLUSION

The GMM-based block coder for images presented in this paper has shown that accurate modelling of the source using GMMs can lead to significant performance gains (PSNR of 2 to 6 dB better at medium to high bit rates) over current transform coders which assume the PDF to be a standard function (eg. Gaussian). The GMM estimation procedure captures the statistics of the source PDF which are common to all bit rates and can be performed offline. The bit allocation involves simple computations based on the GMM parameters and can be performed online depending on the chosen bit rate, thus making the coder scalable. The number of clusters, being a critical design parameter, has been shown to incur a relatively small overhead. Also the search complexity is independent of the bit rate and thus is considerably less than that of unstructured vector quantisers.

5. REFERENCES

[1] V.K. Goyal, "Theoretical Foundations of Transform Coding", *IEEE Signal Processing Magazine*, Vol. 18, No. 5, Sept. 2001.

[2] R.M. Gray & D.L. Neuhoff, "Quantization", *IEEE Transactions on Information Theory*, Vol. 4, No. 6, Oct 1998.

[3] J.J.Y. Huang & P.M. Schultheiss, "Block Quantization of Correlated Gaussian Random Variables", *IEEE Trans. Commun. Syst.*, Vol. CS-11, pp. 289–296, Sept. 1963.

[4] S.P. Lloyd, "Least Square Quantization in PCM", *IEEE Transaction on Information Theory*, Vol. IT-28, No. 2, pp. 129–137, Mar 1982.

[5] J. Max, "Quantising for Minimum Distortion", *IRE Trans. Inform. Theory*, Vol. IT-6, pp. 7–12, Mar 1960.

[6] A. Ortega & M. Vetterli, "Adaptive Scalar Quantization Without Side Information", *IEEE Trans. Image Proc.*, 1997.

[7] A.D. Subramanian & B.D. Rao, "PDF Optimized Parametric Vector Quantization with Applications to Speech Coding", *34th Asilomar Conference on Signals, Systems and Computers*, Pacific Grove, CA, Nov 2000.

[8] A.D. Subramanian & B.D. Rao, "Speech LSF Quantization with Rate Independent Complexity, Bit Scalability and Learning", *Proc. IEEE Int. Conf. Acoustics, Speech, and Signal Processing*, Vol. 2, pp. 705–708, 2001.

[9] G.K. Wallace, "The JPEG Still Picture Compression Standard", *Communications of the ACM*, Vol. 34, No. 4, pp. 30-44, Apr 1991.