QUADTREE BASED CLASSIFICATION WITH ARITHMETIC AND TRELLIS CODED QUANTIZATION FOR SUBBAND IMAGE CODING

Farshid Golchin and Kuldip K. Paliwal*
School of Microelectronic Engineering
Griffith University
Brisbane, QLD 4111, Australia
F.Golchin, K.Paliwal@me.gu.edu.au

ABSTRACT

In this a paper a quadtree based method is proposed for classifying blocks of samples in image subbands. Classification of blocks of subband samples according to their energy and variable bit allocation within the subsequent classes has demonstrated considerable gains in coding efficiency. The gains due to classification increase as smaller blocks are used; however, so do the overheads for transmitting the classification information. The quadtree based method proposed in this paper allows for more efficient classification by using variable-sized blocks in order to maximize the classification gain, while maintaining a limit on the classification overheads. Using an efficient quantization scheme such as ACTCQ [5] (Arithmetic and Trellis Coded Quantization), we have been able to demonstrate competitive coding results at low bit-rates.

1. INTRODUCTION

Subband coding [1] has proven to be an efficient method of coding images at low bit-rates. In subband coding, the image is first decomposed into a number of critically sampled subbands and then quantized and transmitted to the decoder. In a subband decomposed image, the different subbands usually contain vastly different amounts of energy. This property of subbands is utilized in coding. The bands which contain more energy are quantized using a finer quantizer and those bands which contain less energy are quantized more coarsely.

The choice of fine and coarse quantizers corresponds with the number of bits used by each quantizer. Usually an optimization algorithm [3] is used to allocate bits to each subband according to the energy in that subband and the rate-distortion characteristics of the quantizers being used. In effect, the optimization algorithm minimizes the Mean-Squared-Error (MSE) of the reconstructed image for a given overall bit-rate. In this fashion the non-uniform distribution of energy across the subbands is used to achieve compression. However, a close examination of a typical subband decomposed image reveals that the spatial distribution of energy within the subbands is also far from uniform.

Most of the energy within subbands is confined to areas corresponding to edges and strong textures in the original image. This non-uniformity within the subbands can be exploited to make coding more efficient. Chen and Smith [2] proposed such a scheme for coding of images using the discrete cosine transform (DCT). In their scheme, the image is divided into a number of equal-sized square blocks which are classified according to their energy. Each DCT coefficient within each class is then assigned a number of bits according to the average energy of the particular transform coefficient in that class and the overall bit budget.

The Chen-Smith type of classification can be easily adapted for use in a subband coder. Each subband is divided into a number of equal-sized blocks which are classified according to their energies. An optimization algorithm is then used to select an appropriate quantizer for each class of each of the subbands.

In the Chen-Smith type classification, the classes are chosen such that they are equally populated. However, this is non-optimal and recently better classification schemes have been devised [6], [8]. Joshi et al. [4], provide a thorough study of a number of classification schemes for subband coding.

Although classification provides considerable coding gain, this gain comes at a cost. The decoder needs to be made aware of the classification information. This is normally done by transmitting a classification table which indicates the classes to which the blocks of subband samples belong. Using smaller sized blocks results in a higher coding gain, but it also increases the amount of classification information which needs to be transmitted.

The choice of an appropriate block size is a trade-off between the coding gain resulting from classification and the amount of classification information. Several methods for the reduction of classification information have been proposed in the literature [4]. However, at low bit-rates, the classification information can still amount up to 20% of the total bit budget.

The scheme proposed in this paper aims to provide more efficient classification by using smaller blocks where required (in areas of high activity) and larger blocks in other areas.

2. THE PROPOSED SCHEME

As mentioned previously, the energy in the subbands is not distributed uniformly. In a typical subband decomposed image, there are small areas of high activity which correspond to edges and strong textures, and large areas with little activity corresponding to the smoother areas in the original image. In this paper we attempt to exploit this property by allocating smaller block sizes over the non-uniform (high activity) areas of the subbands and larger block sizes in the areas of uniformity (low activity). This added degree of adaptivity allows for more efficient classification of the subband samples for a given classification bit budget.

In the case of non-uniform block sizes, the decoder also needs to be made aware of the sizes and the locations of the blocks used. The quadtree [7] was selected as an efficient method of encoding the blocking scheme. In the following sections, we will describe the algorithms for generating the quadtrees and the methods of encoding the quadtrees.

2.1. Generating the Quadtrees

Similar to the binary-tree, the quadtree is a tree structure. However, instead of nodes branching off to two children as

*The first author is supported in part by a CSIRO Division of Telecomm. and Industrial Physics postgraduate scholarship.

is the case for a binary tree, the quadtree's nodes branch off to four children.

In our application, the root of the quadtree corresponds to the entire image (or a particular subband) and each node which descends from the root corresponds to a square block within that image. The quadtrees used in this paper are not balanced and hence, to encode them one bit must be sent along for each node in the tree to indicate whether or not that node is split.

Now we will examine how the quadtrees are generated. The aim of our quadtree generation process is to split an image (or subband) into small subblocks, each of which have roughly uniform properties. The algorithm used in this case is based on growing the quadtree one step at a time, while splitting a block in each step of the growth. The choice of which block is to be split is made on the basis of an objective criterion, which depends on the degree of uniformity within the block as well as the size of the block. We will refer to this criterion as Splitting Gain (SG).

In this paper, we define the Splitting Gain in two different ways. In the first definition, we use the notion of the classification gain for a non-stationary source [4] and define the Splitting Gain as follows:

$$SG = \frac{N_p \sigma_p^2}{\prod_{i=1}^{4} (\sigma_i^2)^{1/4}}, \tag{1}$$

where N_p is the number of samples in the parent block and σ_p^2 the sample variance of the parent block. The σ_i^2's in this equation represent the variances of the four blocks which would be formed as a result of splitting the parent block. In subsequent sections, we will refer to this definition of the splitting gain as *Definition A*.

Looking at the problem from a slightly different perspective, we may also define the Splitting Gain based on how well the energy of the sub-blocks is represented by their parent. That is, if the energy (standard deviation) of all sub-blocks is similar to that of the parent, we may wish to leave that particular block unsplit. On the other hand if some subblocks have energies which differ greatly from that of the parent block, we would wish to split the block. This definition is closer in line with that of conventional quadtree based image coding. In this case, the Splitting Gain is defined as follows:

$$SG = \frac{N_p}{4} \sum_{i=1}^{4} (\sigma_p - \sigma_i)^2. \tag{2}$$

We will refer to this definition as *Definition B*.

The algorithm for growing the quadtree is as follows:

1 Initialize the quadtree root to the entire image (or subband).

2 Split the root into 4 equal sized blocks.

3 Calculate the Splitting Gain (SG) for each block.

4 Split the block with the largest Splitting Gain into 4 blocks.

5 Calculate Splitting Gain for the new blocks.

6 Repeat from Step 4 until a maximum number of blocks is reached.

We should note that when quadtrees are utilized in a coder, the structure of the quadtree needs to be made known to the decoder. Hence, the number of bits used to encode the quadtree can be important.

Once the quadtree has been generated using *Definition A* or *Definition B*, it is simply encoded using one bit for each node in the quadtree to indicate whether or not it has been split.

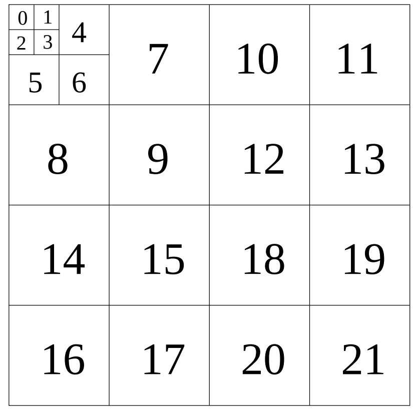

Figure 1. 22-band subband decomposition.

Since our algorithm for generating the quadtree is based on successive splitting of the blocks, it is appropriate to evaluate the length of the quadtree in terms of the number of block splits performed:

$$QuadtreeLength(bits) = 4N_s + 1. \tag{3}$$

where N_s is the number of splits performed on the quadtree root.

In cases where a minimum block size has been set, the leaf nodes of that size will not require an additional bit since they are always left unsplit. Therefore equation (3) only sets an upper limit on the number of bits required to encode the quadtree.

2.2. Using the Quadtrees in a Subband Coder

So far we have explained how quadtrees can be generated in order to split an image into blocks with uniform activity levels (as measured by variance or standard deviation). Now we will take a look at how this concept can be utilized in a classification based subband coder.

The subband coder used in this paper relies on a 22-band decomposition as used in [4], which is shown in Fig. 1.

Joshi et al. [4], have experimented with a number of classification schemes. The most successful of these schemes is based on classifying equal-sized blocks (2x2 for bands 0 to 6 and 4x4 for bands 7 to 21) into one of four classes within each subband. Two algorithms named Maximum Classification Gain and Equal Mean-Normalized Standard Deviation (EMNSD) are proposed for classification. They have shown that these algorithms perform almost equally well and that they outperform the Chen-Smith [2] type of classification. Unlike Chen-Smith classification, both of these algorithms result in classes with unequal populations.

Joshi et al. [4], have also devised methods of reducing the classification information which needs to be sent along by exploiting various dependencies both between and within the classification maps of the subbands. However, despite the reductions, the classification information still comprises a large portion of the total bit-rate. With the use of the quadtree structures described in the previous section, we aim to reduce this overhead through a better, adaptive choice of block sizes.

The simplest method of utilizing the quadtrees in this scheme would be to generate a quadtree for each subband and then perform the classification accordingly. However, in that case, the cost of encoding the quadtrees themselves can become prohibitive. A typical quadtree (with around 400-500 blocks) used for a subband can contribute around 0.002 bpp to the overall bit-rate for a 512x512 pixel image. Thus encoding 22 such quadtrees would take up around 0.04

bpp which is quite expensive when the total bit budget is around 0.5 bpp or less.

The smallest four subbands (subbands 0-3) require at most 512 bits to classify them into 4 classes of 2x2 blocks. This is equivalent to a contribution of approximately 0.002 bpp to the total bit-rate which is hardly worthwhile attempting to reduce. On the other hand, the higher frequency subbands 10-21 usually contain very little energy and in our range of target bit-rates are mostly quantized to zero. The subbands which interest us the most are subbands 4-9 where the majority of the classification information is required.

We examine 3 different methods for incorporating the quadtrees into the subbands classification:

- Method 1: Subbands 0-3 are divided into 2x2 sample blocks and classified. A single quadtree is generated on the original (512x512) image, and scaled down appropriately for use in subbands 4-21. The minimum block size in the quadtree is limited to 16x16 pixels which corresponds to 2x2 samples in subbands 4-6 and 4x4 samples in subbands 7-21 after appropriate down-scaling.
- Method 2: Design a quadtree for each of the subbands 4-9. Uniform sized blocks (2x2 samples) used for subbands 0-3. Subbands 10-21 will use the same quadtree as subbands 7,8 or 9 depending on their orientation. That is, the diagonal bands (subbands 18 and 21) use the quadtree generated for subband 9, The vertical bands (14,15,16,17 and 20) use the same quadtree as subband 8 and so on.
- Method 3: Design a quadtree for each of the subbands 4,5 and 6. Subbands 0-3 will be divided into 2x2 blocks as before. Subbands 7-21 will use scaled up versions of the quadtrees for subbands 4,5 or 6 depending on their orientation (determined as in Method 2).

Figure 2 is an example of a quadtree (of size 421 blocks) generated from the original (512x512) Lena image by Method 1 using *Definition A* of Splitting Gain. The quadtree has been superimposed onto the original image to show where the splits have been made.

In the following sections, we provide a more detailed description of the subband coder and compare the various methods of generating and using the quadtrees.

3. THE SUBBAND CODER

The subband coder used to demonstrate the quadtree based classification is very similar to the coders used in [4] and [5].

The quantizer used is the Arithmetic and Trellis Coded Quantizer (ACTCQ) described in [5]. At the heart of the ACTCQ system, lies a scalar quantizer with uniform thresholds. The codewords of the scalar quantizer are divided into a number of subsets corresponding to different states of the trellis. The Viterbi algorithm [12] is then used to choose the trellis path which minimizes the distance between the quantizer's inputs and its outputs. An arithmetic coder is used to encode the trellis codewords. For a more detailed description of the ACTCQ system, refer to [5].

ACTCQ and other similar quantizers such as ECTCQ [9], have demonstrated excellent rate-distortion performance for the quantization of Generalized Gaussian (GG) sources. The performance of these quantizers makes them ideal candidates for use in a subband coder. Operational rate-distortion curves for Generalized Gaussian sources with different shape parameters (in this case 0.5,0.6,0.7,0.8,0.9,1.0,2.0) are generated and stored for subsequent use by the bit allocation algorithm.

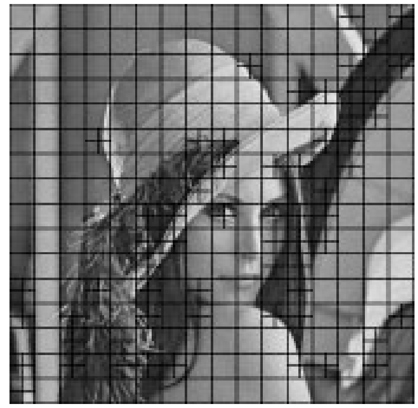

Figure 2. Quadtree generated using Method 1A and superimposed onto the Lena image

Once the subbands are divided into blocks (using quadtrees or equal-sized blocks), the blocks within each subband are classified into four classes. The classification algorithm used is the Equal Mean-Normalized Standard Deviation (EMNSD) classification as described in [4]. Each class in each subband is modeled as a realization of a Generalized Gaussian source whose variance and shape parameter are estimated.

An optimal bit-allocation algorithm is used to allocate the bit-budget among the classes in the different subbands. The bit allocation algorithm used in this paper is that of Westerink et al. [3]. This algorithm is a greedy (gradient based) algorithm which starts by allocating zero bits to all sources and then increases the bit-rate of the sources one at a time until the bit budget has been exhausted.

Once the bit-allocation is completed the classification maps are encoded and transmitted to the decoder. As described in [4], a number of methods are used to reduce the classification information. These methods can be summarized into the following 3 points:

1 The classification maps of subbands where all classes are allocated zero bits, need not be transmitted.

2 If more than one class in a particular subband has been allocated zero bits then these classes can be combined together into one class.

3 The classification tables are entropy coded using conditional probabilities. The symbol probabilities are conditioned on the classification maps of other subbands as well as the class index of adjacent blocks. In this fashion, interband and intraband dependencies are exploited.

The final step in coding is the quantization of the subband samples. After the ACTCQ has completed the quantization of all subband samples, the encoded file sizes are measured and used to determine the bit rate of the coder.

4. RESULTS

The subband coder described in the previous section is used to compare the performance of the various quadtree schemes described in section 2 with the performance of the system using equal-sized blocks as in [4].

The filters used in the subband coder are Antonini et al.'s 7-9 tap perfect reconstruction filter pair [10] and Johnston's 32D (32-tap) filter [11]. For target bit-rates below 0.3 bpp, the 7-9 tap filter pair provides better performance both in perceptual and PSNR terms. The short filter lengths result in less noticeable ringing around the edges at low bit-rates.

	Method 1	Method 2	Method 3
SG Definition A	34.46	34.53	34.61
SG Definition B	34.31	34.38	34.44
Uniform Sizes		34.32	

Table 1. PSNR (dB) for various methods of encoding Lena at 0.25 bpp

However, the 32-tap filter gives slightly better results at around 0.3 bpp and higher. Thus, the subband filters are selected in each case depending on the target bit-rate.

Coding results at a bit-rate of 0.25 bpp for the Lena image (512x512 pixels, 256 grey levels) are listed in Table 1. We have observed that quadtrees of around 400-500 blocks lead to the best results in all methods and, hence, have used these sizes in this experiment. The allocation of the bit-rate between the classification and quantization has been "tweaked" so that the overall bit rate is as close as possible to the target bit-rate. Normally, this is not necessary; however, in this case, it is needed to enable meaningful comparisons to be made among the various methods.

The columns in Table 1 correspond to the different methods of using the quadtree (see section 2.2), while the rows correspond to the definition of Splitting Gain (SG) used in the generation of the quadtree (see section 2.1). It is clear that *Definition A* of the Splitting Gain produces better results regardless of the quadtree being used. This is not surprising, since this definition closely follows the concept of classification gain. It can also be observed that Method 3 of using the quadtree produces the best results. Method 3 is a good compromise between Method 1 (where one quadtree is generated for all subbands) and Method 2 (where a quadtree is generated for each of subbands 4,5,6,7,8 and 9).

Method 3A (quadtrees generated according to Method 3 and *Definition A* of the splitting gain) results in a PSNR improvement of almost 0.3 dB over the uniform block-size scheme of [4], which is a small but significant improvement. We should also note that in Method 3A the classification overhead (including the quadtrees) amounted to slightly over 0.02 bpp which is about half of that reported in [4]. This reduction in the classification information effectively leaves the quantizers with more available bits.

As the bit-rate increases the improvement due to the use of quadtrees becomes less significant. At a bit-rate of 0.5 bpp, the advantage of Method 3A over the uniform block sized scheme is around 0.08 dB which is almost insignificant (37.96 dB PSNR for Method 3A compared to 37.88 dB [4] for uniform block sizes). At this bit-rate other quadtree based methods demonstrate almost no advantage over the equal block-sized scheme. The classification overhead for Method 3A at 0.5 bpp is 0.042 bpp which is 0.01 bpp less than the overheads for equal sized blocks. At bit-rates above 0.5 bpp the gain due to quadtrees continues to diminish as the classification information comprises an increasingly smaller portion of the total bit-rate.

5. CONCLUSION

In this paper, we have used a quadtree based classification scheme for a subband coder. We have experimented with a number of different methods for the generation of quadtrees and incorporated them into a subband coder. It has been shown that *Definition A* of the Splitting Gain is an appropriate measure for use in generating the quadtree.

It has also been shown that Method 3 of using the quadtrees in the subband coder provides a good compromise between adaptivity of the quadtrees and the amount of information used in encoding individual quadtrees. The combination of Method 3 and *Definition A* of the Splitting Gain (referred to as Method 3A) results in an improvement of almost 0.3 dB PSNR over the uniform block based method of [4] at bit rates around 0.25 bpp.

This improvement is due to the reduction of the classification information which needs to be transmitted. For bit-rates below 0.3 bpp, this reduction ensures that a larger portion of the total bit budget is made available to the quantizers. As a result, improvement is quite noticeable in the quality of the coded image at these low bit-rates.

As the bit rate increases beyond 0.3 bpp, the gain due to the use of quadtrees becomes less significant. It is no longer noticeable at 0.5 bpp and beyond. This is due to the fact that the classification information comprises a smaller portion of the bit budget and, hence, has a smaller effect on the overall performance of the coder.

REFERENCES

[1] J.W. Woods and S.D. O'Neil, "Subband coding of images", *IEEE Trans. Acoust., Speech and Signal Proc.*, vol. ASSP-34, pp. 1278-1288, October 1986.

[2] W.H. Chen and C.H. Smith, "Adaptive coding of monochrome and color images", *IEEE Trans. on Commun.*, vol. COM-25, pp. 1285-1292, November 1977.

[3] P.H. Westerink, J. Biemond and D.E. Boekee, "An optimal bit allocation for subband coding", *Proc. IEEE Int. Conf. Acoust., Speech and Signal Proc.*, pp. 757-760, April 1988.

[4] R.L. Joshi, H. Jafarkhani, J.H. Kasner, T.R. Fischer, N. Farvardin, M.W. Marcellin and R.H. Bamberger, "Comparison of different methods of classification in subband coding of images", *Submitted to IEEE Trans. on Image Proc.*, available from: `HTTP://www.ee.umd.edu/~hamidj/publications.html`, October 1995.

[5] R.L. Joshi, V.J. Crump and T.R. Fischer, "Image subband coding using arithmetic coded trellis coded quantization", *IEEE Trans. Circuits and Systems for Video Technology*, vol. 6, pp. 515-523, December 1995.

[6] H. Jafarkhani, N. Farvardin and C.-C. Lee, "Adaptive image coding based on the discrete wavelet transform", *Proc. IEEE Int. Conf. on Image Proc.*, vol. 3, pp. 343-347, November 1994.

[7] J. Vaisey and A. Gersho, "Image compression with variable block size segmentation", *IEEE Trans. Signal Proc.*, vol. 40, pp. 2040-2060, August 1992.

[8] R.L. Joshi, T.R. Fischer and R.H. Bamberger, "Optimum classification in subband coding of images", *Proc. IEEE Int. Conf. on Image Proc.*, vol. 2, pp. 883-887, November 1994.

[9] M.W. Marcellin, "On entropy constrained trellis coded quantization", *IEEE Trans. Commun.*, pp. 14-16, January 1994.

[10] M. Antonini, M. Barlaud, P. Mathieu and I. Daubechies, "Image coding using the wavelet transform", *IEEE Trans. Image Proc.*, vol. IP-1, pp. 205-220, April 1992.

[11] J.D. Johnston, "A filter family designed for use in quadrature mirror filter banks", *Proc. IEEE Int. Conf. Acoust., Speech and Signal Proc.*, pp. 291-294, April 1980.

[12] G.D. Forney, Jr., "The Viterbi algorithm", *Proc. IEEE*, vol. 61, pp. 268-278, March 1973.