

# Computation-rate-distortion in transform coders for image compression

Michael J. Gormish<sup>1,2</sup> and John T. Gill<sup>2</sup>

<sup>1</sup>Ricoh California Research Center, Menlo Park, CA 94025

<sup>2</sup>Electrical Engineering Department, Stanford University, CA 94305

## ABSTRACT

We consider the computational complexity of block transform coding and tradeoffs among computation, bit rate, and distortion. In particular, we illustrate a method of coding that allows decompression time to be traded with bit rate under a fixed quality criteria, or allows quality to be traded for speed with a fixed average bit rate. We provide a brief analysis of the entropy coded infinite uniform quantizer that leads to a simple bit allocation for transform coefficients. Finally, we consider the computational cost of transform coding for both the discrete cosine transform (DCT) and the Karhunen-Loève transform (KLT). In general, a computation-rate-distortion surface can be used to select the appropriate size transform and the quantization matrix for a given bandwidth/CPU channel.

## 1. INTRODUCTION

Transform coding has become the most commonly used method of image compression with the advent of the JPEG and MPEG standards. At the same time, general purpose CPUs have increased in capability to the point where some desktop workstations are capable of real time decompression of video, at least at small sizes. Desktop video on general purpose CPUs is more challenging than in other video environments because both bandwidth and available CPU time change. On the desktop it may be desirable to degrade video quality when network bandwidth becomes limited or when the CPU is needed for additional tasks. Typically, in these situations the frame rate is decreased, freeing up both bandwidth and computational resources. In this paper, we will look at changing the bandwidth and computational requirements independently in an attempt to maintain the highest quality.

Rate-distortion theory answers the question: given a fixed average number of bits per symbol, what is the smallest distortion achievable when encoding a source? Unfortunately, the proofs of achievability depend on arbitrarily long block lengths. Real encoding systems limit themselves to rather small block sizes in the interest of both computational complexity and encoder delay. For example, JPEG does transform coding and quantization on 8 by 8 blocks of pixels. Because of these limitations, data compression systems like JPEG cannot reach the rate-distortion limit. Any transform coder or class of coders is capable of coding a source to a specific bit rate in some number of operations and incurring some distortion. For a given random source and limited number of operations and bit rate, there is a best possible expected distortion. Any specific coding system provides an achievable computation-rate-distortion point that provides an upper bound on the theoretical minimum.

Some systems, such as transform coders, can be described by sets of parameters. Each parameter set yields a rate, a distortion, and a computation cost when applied to a specific source. If all parameters of a system can be effectively analyzed, it is possible to determine the best possible operation of a system. This can allow a more general comparison of systems than is possible by checking a few data sources with several coder settings.

In this paper we consider the discrete cosine and Karhunen-Loève transforms of various sizes applied to our image model. By considering the best possible performance of both systems, we observe that for a large class of images there is no advantage to using the “optimal” KLT. A larger DCT can provide the same performance as a smaller KLT at a lower computational cost.

## 2. TRANSFORM IMAGE CODING

A typical transform coder operates on blocks of data as shown in Figure 1. The transform block takes a vector  $X$  of pixels and performs a matrix multiplication to yield  $Y$ , a vector of coefficients, that is,  $Y=TX$ . In the JPEG and MPEG standards, quantization is performed independently on each transformed coefficient, then the result is entropy coded with either Huffman or arithmetic coding. The goal of the transform is twofold: 1) compact the signal energy into only a few transform coefficients which can be quantized accurately, and 2) use the transform domain to apply a frequency sensitive distortion measure. We will ignore the second goal in this paper, although it is often more important for human image observers. For image coding the discrete cosine transform has become the transform of choice. It is often stated that the DCT performs close to the optimal Karhunen-Loève transform and is computationally more efficient. We will make such statements more precise and show the relationship as a surface of achievable computation-rate-distortion points.

The transform systems we consider use uniform quantization of coefficients with optimal bit allocation. We have also assumed perfect entropy coding is possible for the transform coefficients. Before presenting results we need to discuss the image model and quantization.

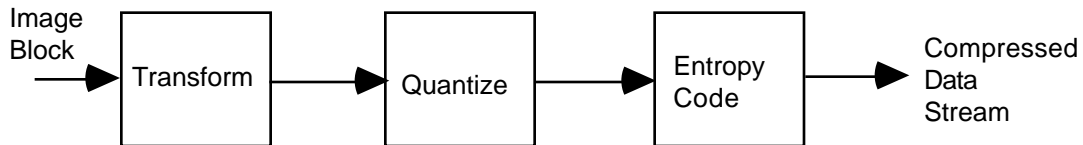


Figure 1. Transform coding system

### 2.1. Image Model

In order to efficiently analyze all parameter settings of a transform coder it is necessary to have an image model. The model we will use is well matched to images in terms of the final bit rate and distortion for many test images, so it allows us not only to analyze the transform coder, but also to predict the rate and distortion for a specific image before compression is performed. This is important because it will allow parameters of the compression algorithm to be changed as the bandwidth and computational resources change.

A common model of pixel statistics is the assumption that the mean-removed pixel intensity has covariance function given by  $r(m,n) = \sigma^2 \begin{matrix} |m| \\ x \end{matrix} \begin{matrix} |n| \\ y \end{matrix}$ , where  $m$  and  $n$  are the horizontal and vertical distances between pixels,  $\sigma^2$  is the mean square pixel intensity,  $\rho_x$  and  $\rho_y$  are correlational coefficients close to 0.9 for images, and  $\sigma^2$  is the mean square pixel intensity. For one-dimensional encoding this can be simplified to  $r(m) = \sigma^2 |m|$ . If the pixel intensities are assumed to have a Gaussian probability density function, then this is a first-order Gauss-Markov model and the model is completely specified. Although a Gauss-Markov model is often used as an approximation for images, it is not the best model when transform coding. Transform coefficients for images are known to be modeled more accurately as Laplacian random variables.<sup>1</sup> Rather than make an assumption about the distribution of pixel intensities, we will make an assumption about the distribution of the transform coefficients. We will assume that after the block transformation the coefficients have a Laplacian distribution with variance given by the diagonal elements of the transform covariance matrix  $R_{yy} = TR_{xx}T^t$ . If we had assumed the pixel intensities had Gaussian distributions, the coefficients would necessarily have had a Gaussian distribution and our results would not have matched real images as closely.

Images are described with three parameters—the variance,  $\sigma^2$ , and the horizontal and vertical correlation coefficients,  $\rho_x$  and  $\rho_y$ . If a one-dimensional transform is performed,  $\sigma^2$  and  $\rho_x$  describe the image. Although

this model does not capture the nonstationarity of images, it is effective for coding at a given average bit rate. After analyzing uniform quantization, we will use the image parameters to perform optimal bit allocation.

### 3. QUANTIZATION

Both JPEG and MPEG quantize transform coefficients uniformly. Each coefficient  $y_{ij}$  is divided by a different value  $q_{ij}$  and these quantized coefficients are entropy coded. It is sometimes thought that the best quantization of transform coefficients is the Lloyd-Max quantizer and that uniform quantization is a convenient although inferior method. In fact, when the quantized coefficients are entropy coded, the uniform quantizer has less distortion for a given bit rate than the entropy coded Lloyd-Max quantizer. More importantly, the Lloyd-Max quantizers operate only at fixed points corresponding to an integer number of reconstruction levels, while the entropy coded uniform quantizer can operate at any desired rate by changing the quantizer values,  $q_{ij}$ . This is extremely valuable in a transform coder where rates of 1/2 bits per pixel are common. The quantizers generated by the Lloyd-Max algorithm have no operational rates between 0 and 1 bits per pixel.<sup>2</sup> Use of uniform quantizers for random variables is well discussed by Farvardin and Modestino.<sup>3</sup> Farvardin and Lin<sup>4</sup> discuss the use of entropy constrained uniform quantizers with transform coders, particularly for Gauss-Markov sources. These papers discuss asymptotic high rate performance extensively, while we are concerned with exact low rate performance.

Most coefficients will be coded with substantially less than one bit. We analyze a quantizer with infinitely many possible reconstruction points. Although this could cause a slight implementation problem, the actual images have bounded pixel intensities, so this is not a problem. The probability of the extreme values is small so our analysis remains usable.

It is easy to compute the rate and distortion of a uniform quantizer from the quantizer step size and input distribution. We consider only uniform quantizers that have a reconstruction value at 0, because symmetric quantizers without a reconstruction point at zero have a minimum rate of one bit per coefficient. By assigning a reconstruction point at 0 it is possible for more than half of the outcomes to be quantized to zero and thus rates under one bit per sample are possible. For example, the simplest quantizer divides the sample by  $Q$  and rounds to the nearest integer. Reconstruction points are at the midpoints of each quantization range, that is,  $\hat{y}_i = iQ$  for all integers  $i$ .

The Laplacian probability density function is given by

$$f_y(y) = \frac{1}{2} e^{-|y|} \quad (1)$$

where  $\sigma = \sqrt{2}$ . The probability that a sample from this or any distribution will be quantized to  $\hat{y}_i = iQ$  is simply the probability  $p_i$  that the sample is between  $Q(i-1/2)$  and  $Q(i+1/2)$ . This is given by

$$p_i = \int_{Q(i-\frac{1}{2})}^{Q(i+\frac{1}{2})} f_y(y) dy. \quad (2)$$

If perfect entropy coding is used, the resulting rate to encode the quantized outcome is given by

$$R(Q) = - \sum_{i=-\infty}^{+\infty} p_i \log p_i. \quad (3)$$

A closed form expression for the rate as a function of the quantizer step size for the Laplacian distribution is given by

$$R(Q) = -\log \left( 1 - e^{-\frac{Q}{2}} + e^{-\frac{Q}{2}} \log \frac{2}{1 + e^{-\frac{Q}{2}}} + \frac{Q}{2 \sinh \frac{Q}{2}} \right). \quad (4)$$

Again by considering each quantization range separately, it is possible to compute the expected distortion as a function of  $Q$ . For midpoint reconstruction,  $\hat{y}_i = iQ$ , the expected distortion for any distribution is

$$D(Q) = \int_{i=-\frac{Q}{2}}^{+\frac{Q}{2}} (y - Qi)^2 f_y(y) dy, \quad (5)$$

and again for the Laplacian random variable there is a closed form,

$$D(Q) = \frac{2}{3} + \frac{Q}{2} e^{-\frac{Q}{2}} - \frac{2Q \cosh \frac{Q}{2}}{(1 - e^{-\frac{Q}{2}})}. \quad (6)$$

Equations (4) and (6) together give a parametric description of the rate-distortion possible using an entropy coded uniform quantizer. Note that when the rate is considered a function of distortion rather than of step size, the  $R(D)$  curve is convex for a single Laplacian random variable. This makes bit allocation much simpler than is often discussed for transform coding bit allocation. Rather than a Gaussian rate-distortion approximation or integer bit allocation algorithms<sup>5</sup>, gradient search methods can be used to allocate bits between several Laplacian transform coefficients.

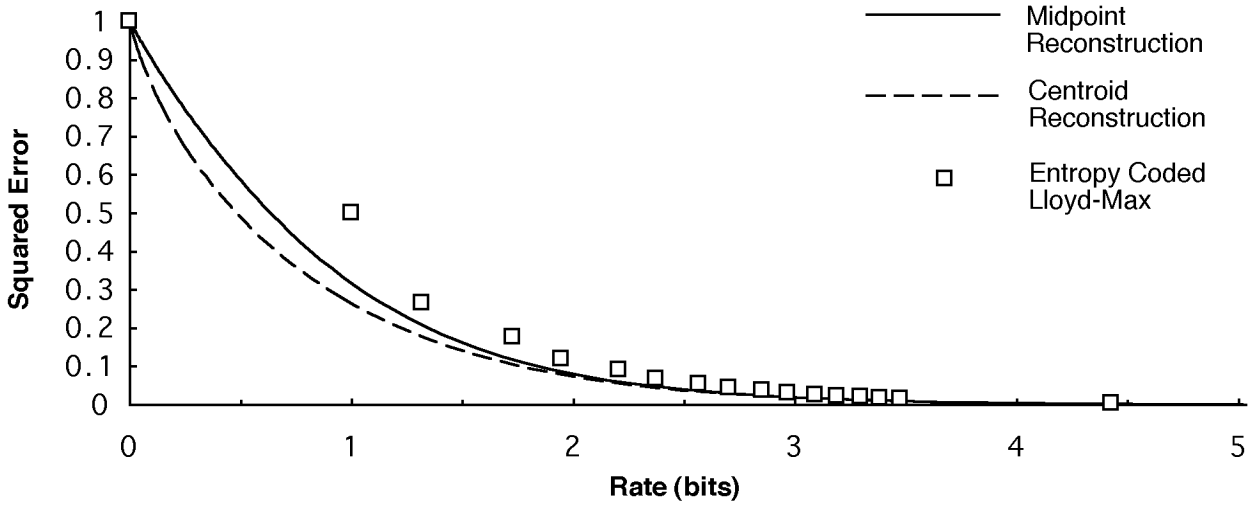


Figure 2. Quantization of a single Laplacian random variable

One addition can be made to the uniform quantizer to improve performance. Instead of reconstructing the quantized outcome at  $iQ$ , the midpoint of the quantization interval, it can be reconstructed at the centroid. That is,

$$\hat{y}_i = E[y | (i - \frac{1}{2})Q < y < (i + \frac{1}{2})Q] = \frac{\int_{(i - \frac{1}{2})Q}^{(i + \frac{1}{2})Q} y f_y(y) dy}{F_y((i + \frac{1}{2})Q) - F_y((i - \frac{1}{2})Q)}. \quad (7)$$

Once again, the Laplacian distribution has a useful and simple form:

$$\hat{y}_i = \begin{cases} \frac{1}{2} - \frac{Q}{2} - \frac{Q}{e^{\frac{Q}{2}} - 1} + iQ & i > 0 \\ 0 & i = 0 \\ -\hat{y}_{-i} & i < 0 \end{cases} \quad (8)$$

This involves no additional work for the encoder. And for the Laplacian distribution, decoding can be done by adding a constant that depends only on the coefficient variance to the usual midpoint reconstruction. Figure 2 shows achievable (R,D) pairs for the Lloyd-Max quantizer and uniform quantization with midpoint reconstruction and uniform quantization with centroid reconstruction. It has been shown that asymptotically uniform quantizers on Gaussian random variables are 0.255 bits worse than the theoretical rate-distortion limit.<sup>4</sup> Fortunately, this gap is *smaller* at low bit rates, since in a transform coder for images most coefficients get substantially less than one bit.

### 3.1. Bit Allocation

Given the statistics and from an image and a fixed transform, it is desirable to find the best quantization for each coefficient. As stated in the image model discussion, we assume the pixel covariance matrix is given by  $[R_{xx}]_{ij} = \frac{1}{2^{|i-j|}}$ , and the coefficient matrix is therefore  $R_{yy} = TR_{xx}T^t$ . We are concerned only with the diagonal elements of  $R_{yy}$ , which give the coefficient variances. If each coefficient is assigned a large initial quantizer,  $q_i$ , then it is possible to compute a rate and distortion as in the previous section. Initially, the average rate per coefficient will be close to zero and the distortion close to  $\frac{1}{2}$ . Decreasing the quantizer step size,  $q_i$ , for any coefficient increases the average bit rate required to encode the outcome and decreases the distortion. Because  $R(D)$  is convex for each coefficient quantizer, the overall best bit allocation can be found by decreasing the quantization of the coefficient with the largest negative value of  $dD/dR$ . This is repeated until the desired rate or distortion is obtained.

## 4. COMPLEXITY

Several operations are necessary for transform coding: transformation, quantization, and entropy coding. The cost of uniform quantization is clearly linear in the number of coefficients or equivalently in the number of pixels. Computation to quantize with the Lloyd-Max quantizer depends on the number of reconstruction levels. Huffman and arithmetic coding and decoding can certainly be done in time linear in the number of coefficients. Only the transform cannot be done in time linear in the number of coefficients or pixels. Much effort has gone into fast methods for transforms, particularly for the most common 8-point transform, but these are still slower than 4-point or 2-point transforms would be. Thus the speed of a transform coder is largely dependent on the transform type and transform size.

Regardless of the transform size the bit rate can be controlled with the quantizer step size. Clearly quality is dependent on bit rate; rate-distortion analysis indicates this tradeoff exactly. However, a larger transform produces higher quality with the same number of bits, because the energy is more concentrated. Thus quality is dependent on both bit rate and computation. Use of a larger transform allows better quality at the same bit rate. Use of smaller quantization intervals allows higher quality at the same complexity.

We have stated that transform cost is not linear in the number of pixels. The exact cost depends on the transform. For the DCT, the most commonly used image transform, fast algorithms operate in time proportional to  $N \log N$ , where  $N$  is the transform size. The "optimal" transform, the KLT, does allow better energy compaction and hence better quality at a given bit rate, but does not have a fast transform in general.<sup>6</sup>

Thus the cost could be as high as  $O(N^2)$ . The computation may be less than  $N^2$  because not all coefficients will need to be computed at low bit rates.

#### 4.1. Computation-rate-distortion surface

Assuming that the computational cost of quantization and entropy coding is constant and equal regardless of the specific transform or size, we can easily plot computation-rate-distortion surfaces for the described coders using the image model we have discussed. This has been done for  $N^2 = 2000$  and  $\alpha = 0.8$  and DCTs of several sizes in Figure 3. This figure is clipped at the left edge where the distortion reaches  $\alpha^2$  at rate zero regardless of the computational effort. The computation axis, which does not include entropy coding or quantization costs, measures cost per pixel. A computation value of  $n$  indicates a size  $2^n$  discrete cosine transform. Although we consider only transforms of size  $2^n$ , we can still regard the achievable points as a surface because some fraction of the blocks could be encoded with a size four transform and the remaining blocks could be coded with a size eight transform. This time sharing makes it possible to use any desired amount of computation and also insures that the computation-rate-distortion surface is convex.

A similar surface is possible for the KLT. At this scale the difference in distortion between a  $2^n$  point KLT and a  $2^n$  point DCT would not be visible. The computation required for the KLT is much higher for sizes above 2 points. In fact, there is less computation required for an 8-point DCT than for a 4-point KLT and the distortion for the 8-point DCT is also less than for the 4-point KLT.

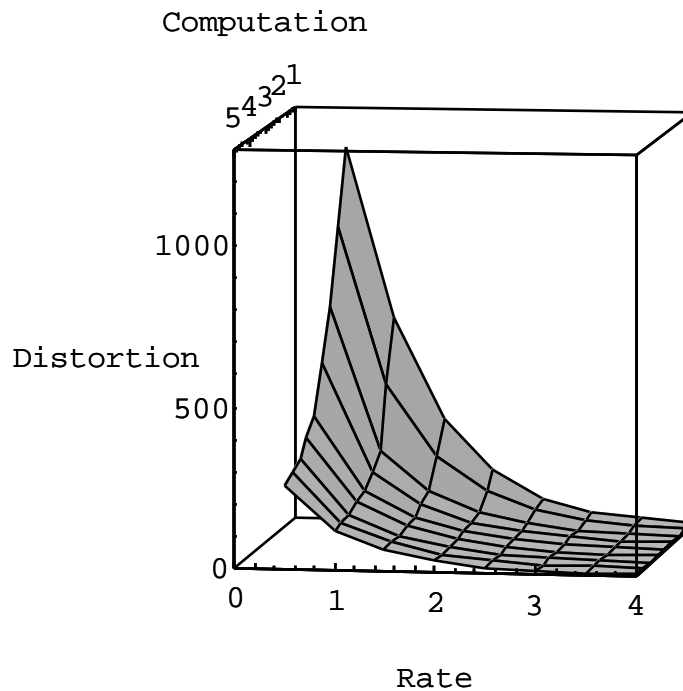


Figure 3. Computation-rate-distortion for DCT

## 5. CONCLUSIONS

We have discussed a method of independently choosing computation and bit rate to maximize quality. The method uses a gradient descent bit allocation for entropy coded uniform quantizer. The image model provides a practical way to rapidly predict the final bit rate of the encoding method. By examining the best parameterization we have been able to compare different transforms in a computation-rate-distortion sense rather than a purely rate-distortion manner.

## 6. ACKNOWLEDGEMENTS

Michael Gormish is supported by an Office of Naval Research Graduate Fellowship.

## 7. REFERENCES

1. R. C. Reininger and J. D. Gibson, "Distributions of the two-dimensional DCT coefficients for images," *IEEE Trans. on Communications*, Vol. COM-31, pp. 835–839, June 1983.
2. J. Max, "Quantizing for minimum distortion," *IRE Trans. on Information Theory*, Vol. IT-6, pp. 7–12, March 1960.
3. N. Farvardin and J. W. Modestino, "Optimum quantizer performance for a class of non-Gaussian memoryless sources," *IEEE Trans. Information Theory*, Vol. IT-30, pp. 485–497, May 1984.
4. N. Farvardin and F. Y. Lin, "Performance of entropy-constrained block transform quantizers," *IEEE Trans. Information Theory*, Vol. 37, pp. 1433–1439, Sept. 1991.
5. Paul Michael Farrelle, *Recursive Block Coding for Image Data Compression*, Springer-Verlag, New York, 1990.
6. W. D. Ray and R. M. Driver, "Further decomposition of the Karhunen-Loève series representation of a stationary random process," *IEEE Trans. Information Theory*, vol. IT-16, pp. 663–668, Nov. 1970.