In *Neural Networks for Signal Processing – Proceedings of the IEEE Workshop, IEEE, December 2000.*

Adaptive Transform Coding as Constrained Vector Quantization

Cynthia Archer and Todd K. Leen
Department of Computer Science and Engineering
Oregon Graduate Institute of Science & Technology
20000 N.W. Walker Rd, Beaverton, OR 97006-1000
e-mail: archer, tleen@cse.ogi.edu

Abstract. We investigate the application of local Principal Component Analysis (PCA) to transform coding for fixed-rate image compression. Local PCA transform coding adapts to differences in correlations between signal components by partitioning the signal space into regions and compressing signal vectors in each region with a separate local transform coder.

Previous researchers optimize the signal space partition and transform coders independently and consequently underestimate the potential advantage of using adaptive transform coding methods. We propose a new algorithm that concurrently optimizes the signal space partition and local transform coders. This algorithm is simply a constrained version of the LBG algorithm for vector quantizer design.

Image compression experiments show that adaptive transform coders designed with our integrated algorithm compress an image with less distortion than previous related methods. We saw improvements in compressed image signal-to-noise ratio of 0.5 to 2.0 dB compared to other tested adaptive methods and 2.5 to 3.0 dB compared to global PCA transform coding.

INTRODUCTION

By compressing vectors of arbitrarily large dimension, vector quantization can achieve code lengths arbitrarily close to the minimum possible length specified by the signal entropy [13]. However, the encoding complexity associated with large vector dimension makes vector quantizers impractical, especially for high-quality compression. Product coders overcome these complexity problems, by partitioning a vector into smaller dimension sub-vectors and separately coding each sub-vector.

Transform coding is a form of product coding where each scalar component of the signal vector is coded separately. A transform coder converts signal vectors to a new coordinate basis in order to reduce the statistical redundancy between vector components. Principal Component Analysis (PCA) is the classic statistical technique used to decorrelate components of vector signals, and hence is often used in transform coding [3, 11, 5]. The PCA transform converts a signal vector to transform coefficients and separate scalar quantizers code each transform coefficient. Quantization replaces each coefficient value with the best match from a small set of reproduction values (the codebook). The concatenated coefficient codes form the compressed vector representation.

Classic PCA transform coding assumes that the correlations between vector components are the same everywhere in the signal space. However, signals are typically not stationary; the high-variance directions and the distributions of coefficient values along those directions will be different in different regions of the signal space. Adaptive or local transform coding methods can capture these differences and so improve compression efficiency [14, 4, 1].

A local PCA transform coder partitions the signal space into disjoint regions and then separately transforms and codes the vectors in each region. To partition the signal space, similar signal vectors are clustered into regions according to some metric. Local PCA transform projects the signal vectors onto the leading local covariance matrix eigenvectors. Local scalar quantizers code the transform coefficients. The region designation and the concatenated coefficient codes form the compressed vector representation.

Previous Work with local PCA transform coding

Previous local PCA transform coding methods [2, 4, 15, 1] use sub-optimal methods to partition the signal space. Chen and Smith's activity classification method [2] partitions signal vectors into four regions according to the variance of the vector components. In our own previous work [1], we cluster the signal vectors using k-means clustering [9]. Dony and Haykin [4] cluster the signal vectors to minimize dimension reduction distortion, which is the distortion induced by projecting the data onto the leading m eigenvectors of the region's covariance matrix. This is the optimal clustering metric for *dimension reduction*, but not for compression where a scalar quantization operation follows the projection. Tipping and Bishop [15] cluster the signal vectors to maximize the data likelihood of a constrained mixture of Gaussians model. None of these methods minimizes compression-induced distortion.

Optimal local PCA transform coding

In this paper, we present a new algorithm that concurrently optimizes the signal space partition and the local transform coders. This algorithm is a constrained variant of the Linde-Buzo-Gray (LBG) algorithm for vector quantizer design [7]. In our case the quantizer reproduction values are constrained to form product codes. To optimize the partition, we cluster signal vectors so that each vector is assigned to the region whose transform coder compresses it with the least distortion. We then design each region's transform coder to compress the signal vectors assigned to it with minimal distortion.

To evaluate the compression performance of our algorithm, we give results from compressing gray-scale digital images. We compare compressed image quality using our local PCA transform coding to that of using global PCA transform coding and two other adaptive transform coding methods similar to previous work [1, 4].

LOCAL PCA TRANSFORM CODING ALGORITHM

The algorithm for an optimal local transform coder is a constrained version of the Linde-Buzo-Gray (LBG) algorithm [7] for vector quantizer design. The LBG algorithm is a descent-type optimization method that alternates between optimizing the data *partition* and optimizing the *codebook* of quantizer reproduction values. The partition is the assignment of data vectors to regions in the signal space. Assigning each data vector to the region whose transform coder compresses it with the least distortion optimizes the partition. For a transform coder, the reproduction vectors that form each region's codebook are constrained to lie at the vertices of a rectangular grid. Designing local transform coders that compress each region's vectors with minimal distortion optimizes the constrained codebook. The partition and codebook optimization steps form a constrained LBG algorithm that finds a local minimum of compression-induced distortion.

Partition Optimization

To optimize the partition, we recluster the example data vectors to minimize average compression-induced distortion

$$\mathrm{MSE} = \frac{1}{N}\sum_{i=1}^{N} \|x_i - Q(x_i)\|^2, \qquad (1)$$

where x are the signal vectors, their quantized representation is $Q(x)$, and N is the number of signal vectors. To minimize expected distortion, we assign a data vector, x, to the region R^α for which that vector is quantized with the least distortion

$$x \in R^\alpha \text{ if } \|x - Q_\alpha(x)\|^2 \leq \|x - Q^\beta(x)\|^2 \quad \forall\, \beta \neq \alpha. \qquad (2)$$

where $Q^\alpha(x)$ is the restoration of vector x after compression with the local transform coder for region R^α.

To assign a signal vector x to a region, we first convert x to each local covariance eigenbasis using the local region transform. The resulting transform coefficients are each coded with the corresponding region's *scalar* quantizers and the resulting distortion is calculated. We compare these distortions and assign x to the region that codes it with the least distortion. For M regions and B coding bits, this assignment requires order Md^2 multiply/add operations for transformation and MB compare operations for quantization. This assignment complexity is much less than the order $dM2^B$ multiply/add and $M2^B$ compare operations required for a conventional vector quantizer.

Codebook Optimization

Codebook optimization finds the quantizer reproduction values that minimize quantization distortion (1) while keeping the number of coding bits below some target value, B. The number of coding bits, B, determines the compression ratio. Region R^α's codebook includes the following parameters (for d-dimensional vectors): the $d \times d$ orthogonal transform U, the number of bits assigned to the quantizer for each coefficient $\{b_J\}$, $J = 1 \ldots d$, and the (scalar) quantizer reproduction values for each coefficient $\{q_{Ji}\}$, $J = 1 \ldots d$ and $i = 1 \ldots 2^{b_J}$. We find values for these parameters[1] that minimize the region distortion

$$D^\alpha = \mathsf{E}\left[\sum_{J=1}^{d} \min_{i \in \{1 \ldots 2^{b_J}\}} (\, x^T u_J - q_{Ji})^2\right] \tag{3}$$

where $\mathsf{E}[\cdot]$ denotes expectation over the signal vectors $x \in R^\alpha$ and u_J is the J^{th} column vector of U. Note that $\sum_J b_J = B$ and $u_J^T u_K = \delta_{JK}\ \forall J, K$.

This non-linear optimization problem consists of three *interdependent* operations. First, find the orthogonal matrix, U, that minimizes (3). Second, allocate the available coding bits, B among the transform directions to find quantizer sizes, $\{b\}$, that minimize (3). Third, design quantizers, $\{q\}$, by placing 2^{b_J} reproduction values along each transform direction, u_J, to minimize average coefficient quantization distortion

$$D_J^\alpha(b_J) = \mathsf{E}[\min_i (\, x^T u_J - q_{Ji})^2]. \tag{4}$$

We discuss each of these three steps in the following paragraphs.

Transform Definition Surprisingly, the transform U that minimizes (3) has apparently not been previously discussed in the literature. One *can* peform the minimization with respect to the group of orthogonal transformations U. That minimization will, of course, depend on the current values of the reproduction values $\{q\}$. We will discuss this optimization and the resulting transform in a future publication. As an *approximation* to the optimal coding transform we use the PCA transform, which minimizes the compression distortion for Gaussian variables in the high bit-rate limit [5]. We use singular value decomposition (SVD)[10] of the data vectors assigned to a region to find the local PCA transform, U. In global transform coding trials, using PCA instead of the optimal coding transform reduced compressed image signal-to-noise ratio by 0.1 dB or less.

[1] Each region has its own U, b, and q. We dropped the α superscript to simplify notation.

Bit Allocation Optimal bit allocation determines the number of reproduction values, $\{b\}$, in each quantizer by distributing the available coding bits[2], B, where they will reduce the quantization distortion the most [12]. However, finding this optimal distribution of coding bits requires designing a full set of quantizers (one through eight bits) for each coefficient, which makes this approach computationally prohibitive. Instead, our implementation uses the greedy algorithm described in [5]. This method allocates bits one at a time to the direction with the current largest quantization distortion (4), which significantly reduces the number of quantizers that must be designed[3]. In global PCA tranform coding trials, using this greedy algorithm instead of optimal bit allocation reduced compressed image signal-ro-noise ratio by approximately 0.1 dB.

Quantizer Design We design codebook quantizers in conjunction with bit allocation. The bit allocation process increases the bit-rate by one bit at a time and selects the direction with current largest measured quantization distortion to receive the additional bit. We then update placement of quantizer reproduction values for the selected direction, u_J. To determine the reproduction values, $\{q_{Ji}\}$ $i = 1 \ldots 2^{b_J}$, that minimize (4) we develop an empirical Lloyd quantizer [8] for the coefficient values. The allocation process stops when all available coding bits have been distributed, at which point designs for all quantizers are complete.

Algorithm Implementation

Our implementation of this constrained LBG (CBLG) algorithm for optimal transform coding design uses optimal partitioning. We assign each data vector to the local transform coder that codes it with the least error. However, we make two modifications to codebook optimization, as detailed above, to reduce computational requirements. First, instead of using the optimal coding transform, we use the PCA transform. Second, instead of performing a search for the optimal bit allocation, we use a greedy algorithm. Consequently, there is no guarantee that our implementation of the codebook optimization step will always decrease compression distortion. However, when working with image data, these modifications did *not* cause the procedure to diverge. The total quantization distortion decreased with nearly every[4] partition-codebook optimization step.

[2] The total number of representation bits is the sum of the coding bits, B and the bits required for the region designation, $log_2(M)$ for M regions.

[3] Riskin also presents a greedy algorithm in [12], which allocates bits one at a time where they decrease distortion the most. However, if the quantizer rate-distortion functions are not convex, this greedy algorithm can *discard* high-variance coefficients.

[4] In some training runs, we saw small *isolated* increases in total distortion as the transform coder parameters neared convergence.

EXPERIMENTAL METHODS

We evaluated the compression performance of our local PCA transform coding on gray-scale digital images. Our tests included compression of images from a 25MByte database consisting of 50 frames each from two video sequences of traffic moving through city street intersections[5]. Each image is divided into 8×8 pixel blocks; these blocks are the signal vectors. We concatenated eight frames from the first half of each sequence to form 4,065,132 64-dimensional training vectors. We use individual frames from the last half of each sequence (512×512 pixels = 267,144 vectors) for testing.

We find the adaptive transform coder for a set of images by applying our constrained LBG (CLBG) procedure to a training image. We then compress a test image using the resulting transform coder. We measure compressed *test* image quality with signal-to-noise ratio (SNR),

$$\mathrm{SNR} = 10 \log_{10}(\text{ signal variance / MSE }) \tag{5}$$

where the per pixel MSE is given by (1) divided by the vector dimension d. The signal variance is $\frac{1}{Nd}\sum_n \|x_n - \bar{x}\|^2$ where x are the N signal vectors (8×8 blocks) and $\bar{x}$ is the signal mean. The test image SNR values reported here are the average of eight trials. For each trial, we design a transform coder starting from a different random initialization of the parameters.

Classic global PCA transform coding is our baseline compression method. We also evalute the compression performance of two other adaptive transform coders that use different methods to partition the signal space. The first method, Euclidean Distance Partition (EDP), clusters image blocks into regions so that the Euclidean distance between the blocks and the region means is minimized [1]. The second method, Reconstruction Distance Partition (RDP), clusters image blocks into regions so that the *reconstruction distance* is minimized [6]. The reconstruction distance is the mean squared error between an image block and its dimension-reduced reconstruction. The RDP method is similar to that used by Dony and Haykin [4]. For the RDP method, we selected a target dimension of eight, since at 0.5 bits per pixel (bpp) this dimension gave us the best test image SNR.

IMAGE COMPRESSION RESULTS

We compared the compression performance of the three previously described adaptive transform coding methods (CLBG, EDP, and RDP) and global PCA transform coding. Figure 1 shows SNR for a single test image at compressed bit-rates of 0.375, 0.5, 0.625, and 0.75 bpp (compression ratios of 21:1, 16:1, 13:1, and 11:1 respectively). Compression of other test images yielded similar SNR results. Our CLBG transform coder improves compressed image SNR

[5] The traffic images are available from the Universität Karlsruhe website (i21www.ira.uka.de).

by 2.5 to 3.0 dB compared to using a global transform coder, by 1.3 to 1.8 dB compared to using the EDP method, and by 0.5 to 2.0 dB using the RDP method.

The SNR improvement seen with the RDP method rolls-off at higher bit-rates. This method requires that we select the number of retained dimensions *before* training. We retain only eight dimensions, consequently at higher bit-rates directions that should be coded are discarded.

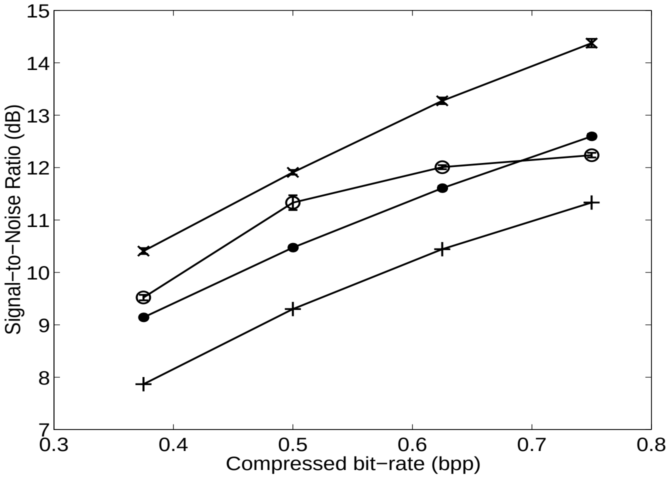

Figure 1: Compressed test image SNR. All adaptive transform coders have 32 regions. The + is global PCA, • is EDP, ∘ is RDP (8 dimension), and × is CLBG. Errorbars (where larger than the symbol size) indicate standard deviation of 8 trials.

The enhanced image quality resulting from CLBG transform coding is also evident in the restored images. Figure 2 shows sections from a test image compressed to 0.5 bpp. The figure includes the original image and restored images from global PCA transform coding and our CLBG transform coding. The global PCA transform coded image is significantly degraded compared to the original. For example, the trolley tracks, the lines on the road, and edges of the cars are blurred and broken. In addition, pixel block edges are readily apparent throughout much of the image. When the image is compressed with CLBG transform coding, the blocking effect is less severe and image details are less blurred.

We also evaluated the effect on compressed image SNR of changing the numbers of regions. Figure 3 shows test image SNR for compression to 0.5 bpp with 8, 16, and 32 region adaptive transform coders. For all three methods, SNR increases as the numbers of regions increase, assuming there is enough representative training data to prevent over-training. This is the expected result, since if all coding bits are used to represent the region designation, these algorithms produce an unconstrained LBG vector quantizer.

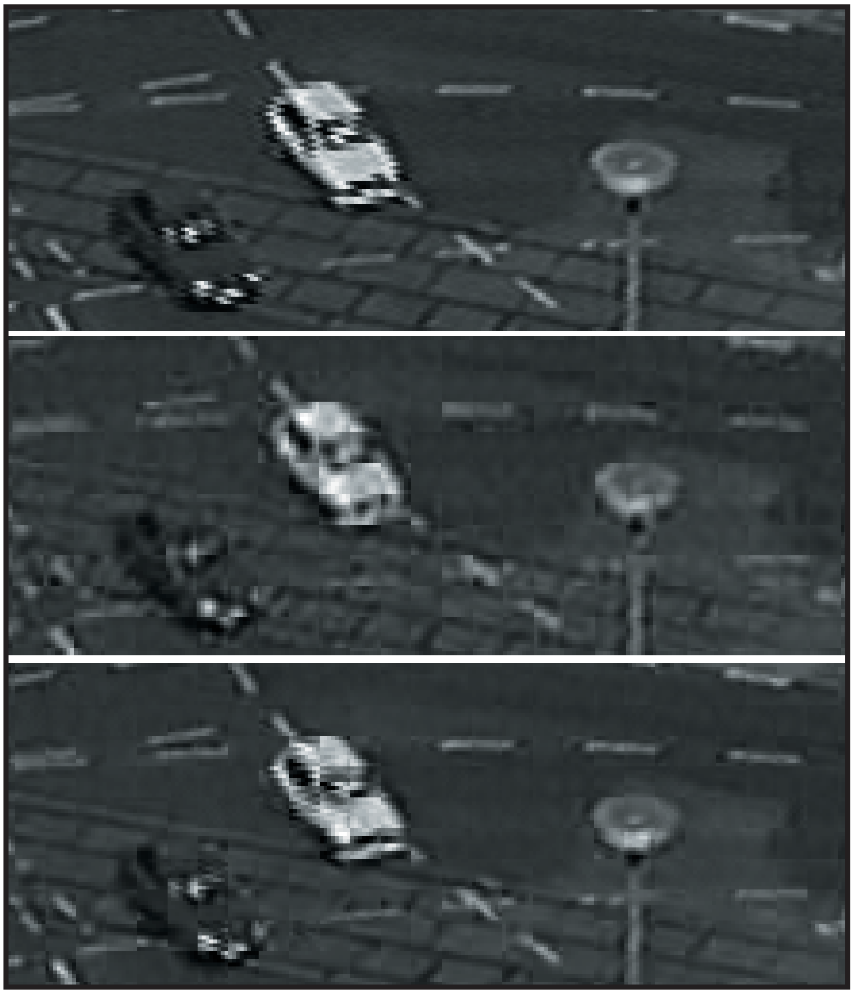

Figure 2: Sections from a test image compressed to 0.5 bpp. From top to bottom, original image, image compressed using global PCA transform coding, and CLBG transform coding with 32 regions.

DISCUSSION

In this paper we recast the construction of local transform codes as a constrained LBG algorithm. The algorithm alternates between optimizing the signal space partition and optimizing the local transform coders until compression distortion reaches a minimum. We optimize the partition by clustering signal vectors into regions so that distortion is minimized. Optimizing the local transform coders involves 1) finding the optimal coding transform, 2) optimally allocating the coding bits among the coefficient quantizers, and 3) developing empirical Lloyd quantizers to code each region's coefficient values

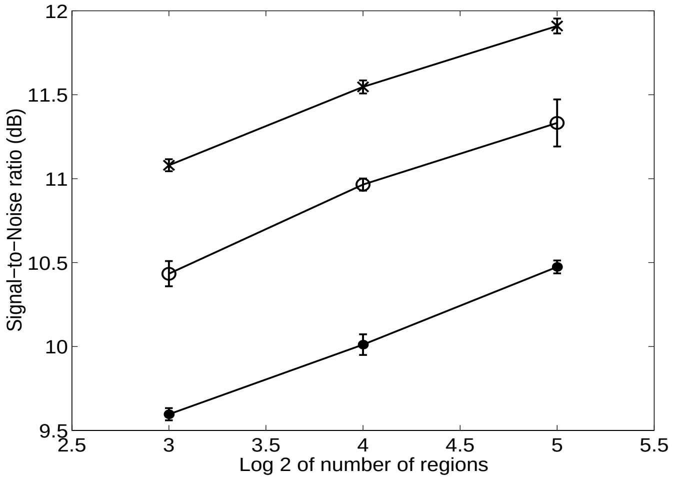

Figure 3: Test image compressed to 0.5 bpp. The • is EDP transform coding, ○ is RDP transform coding, × is CLBG transform coding. Errorbars indicate standard deviation of eight trails.

with minimal distortion.

We evaluated our CLBG algorithm by using it to compress digital grayscale images. To reduce computational requirements, our implementation approximates optimal local transform coder design by using the PCA transform and a greedy bit allocation procedure. At compression ratios in the range of 10:1 to 20:1, tests using our method demonstrate compressed image signal-to-noise ratios up to 3.0 dB higher than global PCA transform coding. When the same images were compressed with adaptive transform coders similar to previously implemented systems [4, 1], the resulting image SNRs are 0.5 to 2.0 dB lower than those obtained with our system. Our integrated algorithm produces more efficient transform coders than previous local PCA methods that design the signal space partition and coefficient quantizers separately.

Acknowledgement. The authors thank Institute für Algorithmen und Kognitive Systems, Universität Karlsruhe for making their traffic images available. This work was supported by NSF under grants ECS-9704094 and ECS-9976452.

REFERENCES

[1] C. Archer and T. Leen, "Optimal dimension reduction and transform coding with mixture principal components," in **Proceedings of International Joint Conference on Neural Networks**, July 1999.

[2] W. Chen and C. Smith, "Adaptive Coding of Monochrome and Color Images," **IEEE Transactions on Communications**, vol. 25, no. 11, pp. 1285–1292, 1977.

[3] R. Clarke, **Transform Coding of Images**, Academic Press, 1985.

[4] R. D. Dony and S. Haykin, "Optimally Adaptive Transform Coding," **IEEE Transactions on Image Processing**, vol. 4, no. 10, pp. 1358–1370, 1995.

[5] A. Gersho and R. Gray, **Vector Quantization and Signal Compression**, Kluwer Academic, 1992.

[6] N. Kambhatla and T. K. Leen, "Fast Non-Linear Dimension Reduction," in Cowan, Tesauro and Alspector (eds.), **Advances in Neural Information Processing Systems 6**, Morgan Kauffmann, Feb 1994, pp. 152–159.

[7] Y. Linde, A. Buzo and R. Gray, "An algorithm for vector quantizer design," **IEEE Transactions on Communications**, vol. 28, no. 1, pp. 84–95, January 1980.

[8] S. Lloyd, "Least Square Optimization in PCM," **IEEE Transactions on Information Theory**, vol. 28, no. 2, pp. 129–137, 1982.

[9] J. MacQueen, "Some methods for classification and analysis of multivariate observations," in **Proc. 5th Berkeley Symp. Math. Stat. Prob.**, 1967, vol. 3, p. 281.

[10] W. Press, B. Flannery, S. Teukolsky and W. Vetterling, **Numberical Recipes in C: the Art of Scientific Computing**, Cambridge University Press, 1988.

[11] K. Rao and P. Yip, **Discrete Cosine Transform - Algorithms, Advantages, Applications**, Academic Press, 1990.

[12] E. A. Riskin, "Optimal Bit Allocation via the Generalized BFOS Algorithm," **IEEE Transactions on information Theory**, vol. 37, no. 2, pp. 400–402, 1991.

[13] C. E. Shannon, "Coding Theorems for a discrete source with a fidelity criterion," in **IRE National Convention Record, Part 4**, 1959, pp. 142–163.

[14] A. Tescher, "Transform Image Coding," in W. Pratt (ed.), **Image Transmission Techniques**, Academic Press, 1979, pp. 113–156.

[15] M. Tipping and C. Bishop, "Mixture of Probabilistic Principal Component Analyzers," **Neural Computation**, vol. 11, no. 2, pp. 443–483, 1999.