Space-Frequency Quantization for Wavelet Image Coding

Zixiang Xiong, *Member, IEEE,* Kannan Ramchandran, *Member, IEEE,* and Michael T. Orchard, *Member, IEEE*

***Abstract*— Recently, a new class of image coding algorithms coupling standard scalar quantization of frequency coefficients with tree-structured quantization (related to spatial structures) has attracted wide attention because its good performance appears to confirm the promised efficiencies of hierarchical representation [1], [2]. This paper addresses the problem of how spatial quantization modes and standard scalar quantization can be applied in a jointly optimal fashion in an image coder. We consider *zerotree quantization* (zeroing out tree-structured sets of wavelet coefficients) and the simplest form of scalar quantization (a single common uniform scalar quantizer applied to all nonzeroed coefficients), and we formalize the problem of optimizing their joint application. We develop an image coding algorithm for solving the resulting optimization problem. Despite the basic form of the two quantizers considered, the resulting algorithm demonstrates coding performance that is competitive, often outperforming the very best coding algorithms in the literature.**

I. Introduction

ALL IMAGE coding algorithms can be viewed as being based on some model for the class of natural images, and can be seen to exploit dependencies characterized by that model. Typical transform and subband coding algorithms model images as a composition of statistically distinct narrowband processes. Basic vector quantization (VQ) algorithms are based on low-dimensional, but very general, models for small blocks of image data, though they provide no model for dependencies between blocks [3]. A variety of more complex coding algorithms have been proposed that are based on models of block interdependence (e.g., finite-state VQ [4], lapped-orthogonal transform coding [5]), composite source models (e.g., classified VQ [6], [7]), image segmentation models, etc. The performance of each of these approaches has been optimized through careful consideration of quantization strategy (e.g., optimal bit-allocation among bands [8], vector quantizers [9], [10], optimal nonuniform scalar quantizers [11], trellis-coded quantization [12], etc.). It has been a general trend that more complex models are needed to achieve improved coding efficiency. One could easily draw the conclusion that the promising directions in image coding involve higher modeling complexity designed to better capture the complex character of natural images.

This paper takes a step in the opposite direction, proposing a very efficient image coding algorithm based on a remarkably simple image model. Drawing on the experience of several recent wavelet-based coding algorithms [1], [2], we suggest that natural images are well characterized as a linear combination of energy concentrated in both frequency and space; i.e., most of the energy of typical images is concentrated in low-frequency information, and of the remaining high-frequency components of the image, most energy is spatially concentrated around edges (we view texture as a dense clustering of edges). Efficient transform coding of such a source model calls for a transform that compacts energy into few low-frequency coefficients, while also representing high-frequency energy in a few spatially clustered high-frequency coefficients. The wavelet transform provides exactly these desired features [13]–[15].

Since the introduction of wavelets as a signal processing tool in the late 1980's, considerable attention has focused on the application of wavelets to image compression [1], [2], [8], [16]–[18]. The hierarchical signal representation given by the dyadic wavelet transform provides a convenient framework both for exploiting the specific types of statistical dependencies found in images, and for designing quantization strategies matched to characteristics of the human visual system. Indeed, before the introduction of wavelets, a wide variety of closely related coding frameworks had been extensively studied in the image coding community, including pyramidal coding [19], transform coding [20], and subband coding [21]. Viewed in the context of this prior work, initial efforts in wavelet coding research concentrated on the promise of more effective compaction of energy into a small number of low-frequency coefficients. Following the design methodology of earlier transform and subband coding algorithms, initial "wavelet-based" coding algorithms [8], [16]–[18] were designed to exploit the energy compaction properties of the wavelet transform by applying quantizers (either scalar or vector) optimized for the statistics of each frequency band of wavelet coefficients. Such algorithms have demonstrated modest improvements in coding efficiency over standard transform-based algorithms.

Contrasting with those early coders, this paper proposes to exploit both the frequency and spatial compaction property of the wavelet transform through the use of two very simple

Manuscript received October 8, 1995; revised January 13, 1997. This work was supported in part by the Ricoh Company and by the National Science Foundation under Grant NSF 92-122 (RIA-94). Part of this work was presented at the International Conference on Acoustics, Speech and Signal Processing, Minneapolis, MN, USA, April 1993, and at the 27th Annual Asilomar Conference on Signals, Systems, and Computers, Pacific Grove, CA, November, 1993. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Nasser M. Nasrabadi.

Z. Xiong and M. T. Orchard are with the Department of Electrical Engineering, Princeton University, Princeton, NJ 08544 USA (e-mail: zx@ee.princeton.edu).

K. Ramchandran is with the Beckman Institute and Department of Electrical and Computer Engineering, University of Illinois at Urbana-Champaign, Urbana, IL 61801 USA.

Publisher Item Identifier S 1057-7149(97)03081-9.

quantization modes. To exploit the spatial compaction properties of wavelets, we define a symbol that indicates that a spatial region of high-frequency coefficients has value zero. We refer to the application of this symbol as zerotree quantization, because it will involve setting to zero a tree-structured set of wavelet coefficients. In the next section, we explain how a spatial region in the image is related to a tree-structured set of coefficients in the hierarchy of wavelet coefficients. Zerotree quantization can be viewed as a mechanism for pointing to the locations where high-frequency coefficients are clustered. Thus, this quantization mode directly exploits the spatial clustering of high-frequency coefficients predicted by our image model.

For coefficients that are not set to zero by zerotree quantization, we propose to apply a common uniform scalar quantizer, independent of the coefficient's frequency band. The resulting scalar indices are coded with an entropy coder, with probabilities adapted to the statistics of each band. We select this quantization scheme for its simplicity. In addition, though we recognize that improved performance can be achieved by more complicated quantization schemes (e.g., vector quantization, scalar quantizers optimized for each band, optimized nonuniform scalar quantizers, entropy-constrained scalar quantization [22], etc.), we conjecture that these performance gains will be limited when coupled with zerotree quantization. When zerotree quantization is applied most efficiently, the remaining coefficients will be characterized by distributions that are not very peaked near zero. Consequently, uniform scalar quantization followed by entropy coding provides nearly optimal coding efficiency, and achieves nearly optimal bit allocation among bands with differing variances. The coding performance of our proposed algorithm provides some experimental evidence in support of this conjecture.

Though zerotree quantization has been applied in several recent wavelet-based image coders, this paper is the first to address the question of how to jointly optimize the application of spatial quantization modes (zerotree quantization) and scalar quantization of frequency bands of coefficients. In [1], Lewis and Knowles apply a perceptually based thresholding scheme to predict zerotrees of high-frequency coefficients based on low-valued coefficients in a lower frequency band corresponding to the same spatial region. While this simple *ad hoc* scheme exploits interband dependencies induced by spatial clustering, it often introduces large error in the face of prediction errors. Shapiro's embedded zerotree approach [2] applies the zerotree symbol when all coefficients in the corresponding tree equal zero. While this strategy can claim to minimize distortion of the overall coding scheme, it cannot claim optimality in an operational rate and distortion sense (i.e., it does not minimize distortion over all strategies that satisfy a given rate constraint).

This paper focuses on the problem of optimizing the application of zerotree and scalar quantization in order to minimize distortion for a given rate constraint. The image coding algorithm described in the following sections is an algorithm for optimally selecting the spatial regions (from the set of regions allowed by the zerotree quantizer), for applying zerotree quantization, and for optimally setting the scalar quantizer's stepsize for quantizing the remaining coefficients. We observe that, although these two quantization modes are very basic, an image coding algorithm that applies these two modes in a jointly optimal manner can be competitive with (and perhaps outperform) the best coding algorithms in the literature. Consequently, we claim that the joint management of space- and frequency-based quantizers is one of the most important fundamental issues in the design of efficient image coders.

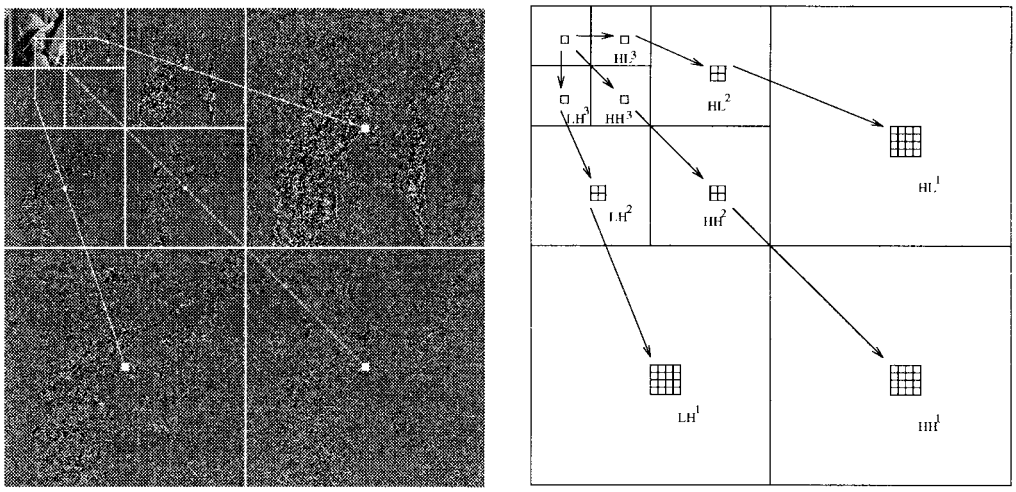
Section II-A begins by defining the tree-structured hierarchy of wavelet coefficients relating trees of wavelet coefficients to spatial regions in the image, and defines zerotree quantization in the context of this tree. The remainder of Section II formally defines the objective function upon which our image coding algorithm is based, and outlines our proposed approach for minimizing this objective function. Section III describes the image coding algorithm developed from the formulation of Section II. Section III-B details our scheme for predicting the information defining where zerotree quantization is applied. It can be viewed as an extension of the Lewis and Knowles scheme [1] intended to exploit interband dependencies. Finally, Section IV presents simulation results of the new algorithm on standard images.

II. Background and Problem Statement

A. Defining the Tree

A wavelet image decomposition can be thought of as a tree-structured set of coefficients, providing a hierarchical data structure for representing images, with each coefficient corresponding to a spatial region in the image. Fig. 1(a) shows a three-level wavelet decomposition of the Lena image, together with a spatial wavelet coefficient tree structure representing the eye region of Lena. A spatial wavelet coefficient tree is defined as the set of coefficients from different bands that represent the same spatial region in the image. Arrows in Fig. 1(b) identify the parent–children dependencies in a tree. The lowest frequency band of the decomposition is represented by the root nodes (top) of the tree, the highest frequency bands by the leaf nodes (bottom) of the tree, and each parent node represents a lower frequency component than its children. Except for a root node, which has only three children nodes, each parent node has four children nodes, the 2×2 region of the same spatial location in the immediately higher frequency band.

Define a *residue tree* as the set of all descendants of any parent node in a tree. (Note: a residue tree does not contain the parent node itself.) Zerotree spatial quantization of a residue tree assigns to the elements of the residue tree either their original values or all zeros. Note the semantic distinction between residue trees and zerotrees: The residue tree of a node is the set of all its descendants, while a zerotree is an all-zero residue tree. A zerotree node refers to a node whose descendants are all set to zero. Note that zerotrees can originate at any level of the full spatial tree, and can therefore be of variable size. When a residue tree is zerotree quantized, only a single symbol is needed to represent the set of zero-quantized wavelet coefficients—our coder uses a binary zerotree map indicating the presence or absence of zerotree nodes in the spatial tree.



(a) (b)

Fig. 1. Wavelet decomposition offers a tree-structured representation. (a) A three-level wavelet decomposition of the Lena image. (b) A spatial wavelet coefficient tree consisting of coefficients from different bands that coresond to the same spatial region of the original image (e.g., the eye of Lena). Arrows identify the parent-children dependencies.

The motivation for applying zerotree spatial-quantization of a residue tree is the observation that the coefficients of any residue tree represent the energy above some fixed frequency (varies with the residue tree size) over some spatial region of the image. Thus, if the spatial region associated with any node of the tree has no energy at frequencies greater than or equal to its assigned frequency, the entire residue tree of coefficients descending from that node should be zero. The zerotree data structure is a convenient way to deal with *sets* of coefficients and therefore to characterize the entropy of sets or vectors of insignificant coefficients without approximating them as the sum of individual entropies, i.e., without assuming independence. An important distinction between our proposed framework and that of earlier use of the zerotree data structure [2] is that in our case, the zerotree criterion does *not* necessarily require that *all* coefficients of the residue tree be insignificant with respect to a set of quantization thresholds. Thus, although scalar quantization can also result in a residue tree of zero values, our zerotree quantization framework is a vector operation that is more general.

B. Motivation and High-Level Description

The underlying theme of the space-frequency quantization (SFQ) is that of efficiently coupling the spatial and frequency characterization modes offered by the wavelet coefficients by defining quantization strategies that are well matched to the respective modes. The paradigm we invoke is a combination of simple uniform scalar quantization to exploit the frequency characterization, with a fast tree-structured zerotree quantization scheme to exploit the spatial characterization.

Our proposed SFQ coder has a goal of jointly finding the best combination of spatial zerotree quantization choice and the scalar frequency quantizer choice. The block diagram of the new coder is shown in Fig. 2. The SFQ paradigm is conceptually simple: Throw away, i.e., quantize to zero, a subset of the wavelet coefficients, and use a single simple uniform scalar quantizer on the rest. Given this framework, the key questions are obviously

1) What (spatial) subset of coefficients should be thrown away?
2) What uniform scalar (frequency) quantizer stepsize should be used to quantize the survivor set, i.e. the complementary set of 1)?

This paper formulates the answers to these questions, invoking an operational rate-distortion optimality criterion. While the motivation is simple, this optimization task is complicated by the fact that the two questions posed above are interdependent. The reason for this is easily seen. The optimal answer to 1), i.e., the optimal spatial subset to throw away depends on the scalar quantizer choice of 2) since the zerotree pruning operation involved in 1) is driven by rate-distortion tradeoffs induced by the quantizer choice. Conversely, the scalar quantizer of 2) is applied only to the complementary subset of 1), i.e., to the population subset that survives the zerotree pruning operation involved in 1). This interplay between these modes necessitates an iterative way of optimizing the problem, which will be described in detail in the following sections.

Note that the answers to 1) and 2) are sent as side information ("map" bits) to the decoder in addition to the quantized values of the survivor coefficients ("data" bits). Since a single scalar quantizer is used for the entire image, the quantizer stepsize information of 2) is negligible and can be ignored. The side-information of 1) is sent as a binary zerotree map indicating whether or not tree nodes are zerotree quantized. This overhead information is not negligible and is optimized jointly with the "data" information in our SFQ coder, with the optimization being done in a rate-distortion sense. At this point we will not concern ourselves with the details of how this map is sent, but we will see later that much of this zerotree map information is actually predictable and can be inferred by the decoder using a novel prediction scheme based on the known data field of the corresponding parent band.

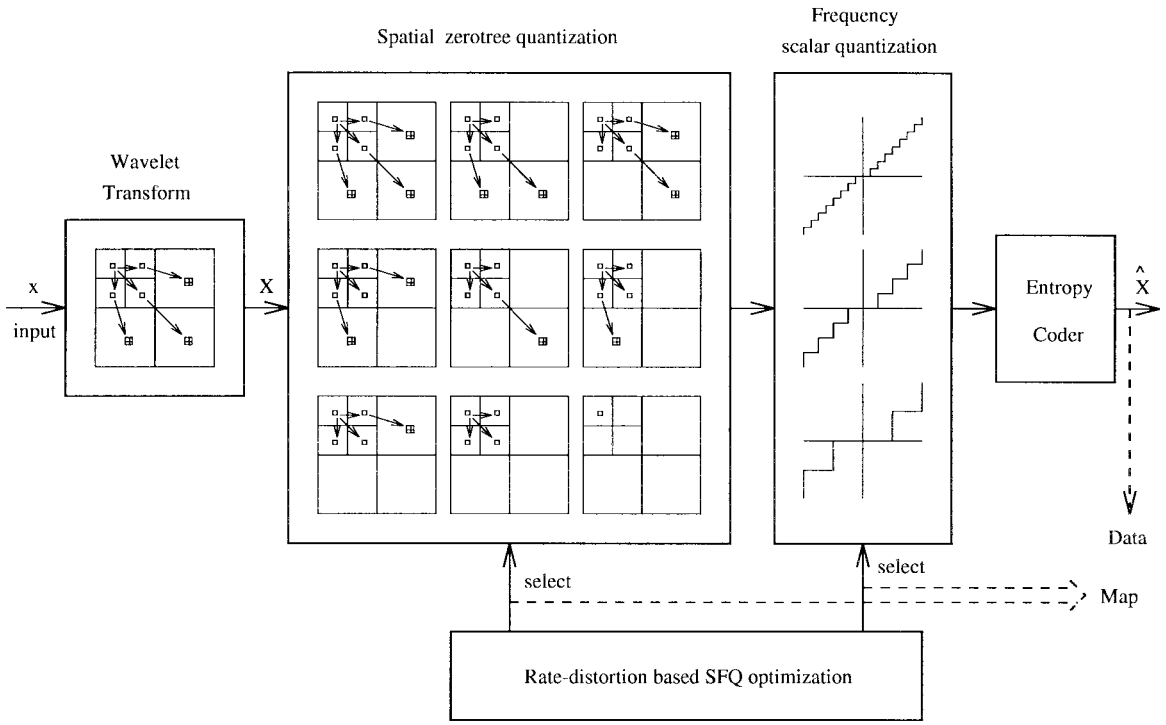


Fig. 2. Block diagram of the proposed SFQ coder (assuming a two-level wavelet decompostion). All nine possible complementary subsets of (I) for each depth-2 spatial wavelet coefficient tree are listed in the spatial zerotree quantization block, three possible scalar quantizer choices of (II) in the frequency scalar quantization block.

Finally, while it appears counterintuitive at first glance to expect high performance from using a single uniform scalar quantizer, further inspection reveals why this is possible. The key is to recall that the quantizer is applied only to a subset of the full wavelet data, namely the survivors of the zerotree pruning operation. This pruned set has a distribution which is considerably less peaked than that of the original full set (see Fig. 9), since most of the samples populating the zero and low-valued bins are discarded during the zerotree pruning operation. Thus, the spatial zerotree operation effectively "flattens" the residue density of the "trimmed" set of wavelet coefficients, endorsing the use of a single stepsize uniform quantizer. In summary, the motivation of resorting to multiple quantizers for the various image subbands (as is customarily done) is to account for the different degrees of "peakiness" around zero of the associated histograms of the different image subbands. In our proposed scheme, the bulk of the insignificant coefficients responsible for this peakiness are removed from consideration, rendering the bands with near-flat distributions and justifying a simple single stepsize scalar quantizer. Experimental evidence verifying these claims will be given in Section IV.

We will now proceed to give a quantitative description of the problem and its solution using our framework. To this end, it is necessary to introduce some notation to aid in the analysis.

C. Notation and Problem Statement

Let T denote the balanced (full) spatial tree, i.e., the tree grown to full depth (in a practical scenario, this may be restricted to four to six levels typically). Letting i denote any node of the spatial tree, T_i signifies the full balanced tree rooted at node i. Note that T is shorthand notation for the balanced full-depth tree (i.e., rooted at the origin). In keeping with conventional notation, we define a pruned subtree $S_i \preceq T_i$ as any subtree of T_i that shares its root i. Again for brevity, $S \preceq T$ refers to any pruned subtree of the full depth tree T. Note that the set of all $S \preceq T$ corresponds to the collection of all possible zerotree spatial-quantization topologies. We also need to introduce notation for residue trees. A *residue tree* U_i (corresponding to any arbitrary parent node i of T) consists of the set of all descendants of i in T but does not include i itself, i.e., $U_i = T_i - \{i\} = \cup_{j \in C_i} T_j$, where C_i is the set of children or direct descendants of i (this is a 2×2 children set for all parent nodes except the root nodes that contain only three children). See Fig. 3.

Let us now address quantization. Let Q represent the (finite) set of all admissible scalar frequency quantizer choices. Thus, the quantization modes in our framework are the spatial tree-structured quantizer $S \preceq T$ and the scalar frequency quantizer $q \in Q$ (used to quantize the coefficients of S). The unquantized and quantized wavelet coefficients associated with node i of the spatial tree will be referred to by w_i and $\hat{w}_i(q)$, with the explicit dependency on q of $\hat{w}_i$ being dropped where obvious. In this framework, we seek to minimize the average distortion subject to an average rate constraint. Let $D(q, S)$ and $R(q, S)$ denote the distortion and rate, respectively, associated with quantizer choice (q, S). We will use a squared-error distortion measure. The rate $R(q, S)$ consists of two components: tree data rate $R_{(data)}(q, S)$, measured by the first-order entropy; and tree map rate $R_{(map)}(S)$, where the superscripts will be dropped where obvious.

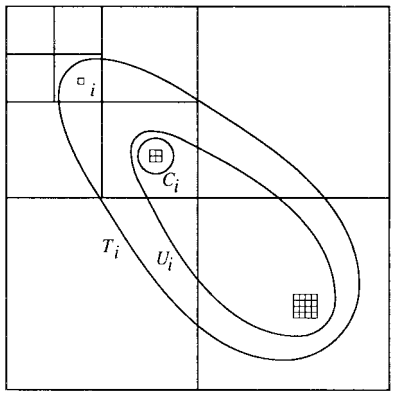


Fig. 3. Definitions of C_i, U_i and T_i for a node i in a spatial tree. C_i is the set of children or direct descendants of i; U_i the residue tree of i; T_i the full balanced tree rooted at node $i (T_i = i \cup U_i)$.

Then, our problem can be stated simply as

$$\min_{\{q \in Q;\, S \preceq T\}} D(q, S) \qquad \text{subject to } R(q, S) \le R_b \quad (1)$$

where R_b is the coding budget.

Stated in words, our coding goal is to find the optimal combination of spatial subset to prune (via zerotree spatial quantization) and scalar quantizer stepsize to apply to the survivor coefficients (frequency quantization) such that the total quantization distortion is minimized subject to a constraint on the total rate. Qualitatively stated, scalar frequency-quantizers trade off bits for distortion in proportionality to their stepsizes, while zerotree spatial-quantizers trade off bits for distortion by zeroing out entire sets of coefficients but incurring little or no bit-rate cost in doing so. We are interested in finding the optimal tradeoff between these two quantization modes.

D. Proposed Approach

The constrained optimization problem of (1) can be converted to an unconstrained formulation using the well-known Lagrange multiplier method. That is, it can be shown [23], [24] that the solution to (1) is identical to the solution to the following equivalent unconstrained problem for the special case of $R(q, S) = R_b$:

$$\min_{\{q \in Q,\, S \preceq T\}} [J(q, S) = D(q, S) + \lambda R(q, S)] \quad (2)$$

where $J(q, S)$ is the Lagrangian (two-sided) cost including both rate and distortion, that are connected through the Lagrange multiplier $\lambda \ge 0$, which is the quality-factor trading off distortion for rate ($\lambda = 0$ refers to the highest attainable quality and $\lambda = \infty$ to the lowest attainable rate). Note that the entropy only or distortion only cost measure of [2] become special cases of this more general Lagrangian cost measure corresponding to $\lambda = \infty$ and $\lambda = 0$, respectively. The implication of (2) is that if an appropriate λ can be found for which the solution to (2) is (q^*, S^*) and further $R(q^*, S^*) = R_b$, then (q^*, S^*) is also the solution to (1). The solution of (2) finds points that reside on the convex-hull of the rate-distortion function, and sweeping λ from 0 to ∞ traces this convex hull. In practice, for most applications (including ours), a convex-hull approximation to the desired rate R_b suffices, and the only suspense is in determining the value of λ that is best matched to the bit budget constraint R_b. Fortunately, the search for the optimal rate-distortion slope λ^* is a fast convex search that can be done with any number of efficient methods, e.g., the bisection method [24].

Our proposed approach is therefore to find the convex-hull approximation to (1) by solving

$$\overset{(c)}{\max_{\lambda \ge 0}} \; \overset{(b)}{\min_{q \in Q}} \; \overset{(a)}{\min_{S \preceq T}} \{D(q, S) + \lambda [R(q, S) - R_b]\} \quad (3)$$

where the innermost minimization (a) involves the search for the best spatial subtree S for fixed values of q and λ, the second minimization (b) involves search for the best scalar quantizer q [and associated $S(q)$] for a fixed value of λ, and finally the outermost optimization (c) is the convex search for the optimal value of λ that satisfies the desired rate constraint (see [24]). The solution (λ^*, q^*, S^*) to (3) is thus obtained in three sequential optimization steps.

Minimization (a) involves optimal tree-pruning to find the best S for a fixed q and λ, and is by far the most important of the three optimization operations of (3). This will be described in Sections III-A and III-B. Given our global framework of data + tree-map, minimization (a) can be written as

$$\overset{(a)}{\min_{\{S \preceq T\}}} \{D(q, S) + \lambda [R_{(data)}(q, S) + R_{(map)}(q, S)]\} \quad (4)$$

where $R_{(map)}(q, S)$ is the tree-map rate and $R_{(data)}(q, S)$ the tree-data rate. We seek an efficient way to jointly code the data + map information using a novel way of predicting the map information from the known data information (see Section III-B). As this method dictates a strong coupling between the data and map field components, the chicken-and-egg problem is solved using a two-phase approach. In the first phase, (4) is optimized assuming that $R_{(map)}(q, S)$ is zero, or more generally that it is a fixed cost[1] independent of the choice of q and S.

The optimal S from the phase I tree-pruning operation is $S^*_{(data)}$, i.e.,

$$S^*_{(data)} = \arg \min_{\{S \preceq T\}} [D(q, S) + \lambda R_{(data)}(q, S)]. \quad (5)$$

In the second phase (tree-map prediction phase), the true data-map dependencies are taken into account, and the solution of phase I, $S^*_{(data)}$, is modified to reflect the globally correct choice $S^*_{(data+map)}$. Details are provided in Section III.

At the end of phase II, for each space-frequency quantization choice, we identify a single point on the operational R–D curve corresponding to a choice of q, λ, and their best matched S. In order to find the best scalar quantizer, we search for the $q \in Q$ [in minimization (b)], which "lives" at absolute slope λ on the convex hull of the operational R–D curve. This defines the optimal combination of q and S for a fixed λ. Finally, the "correct" value of λ, λ^* that matches the rate constraint R_b is found using a fast convex search in optimization (c).

[1] Assuming that $R_{(map)}(q, S)$ is a constant rather than zero simply changes the optimal operating slope λ^* for the same target bit budget R_b.

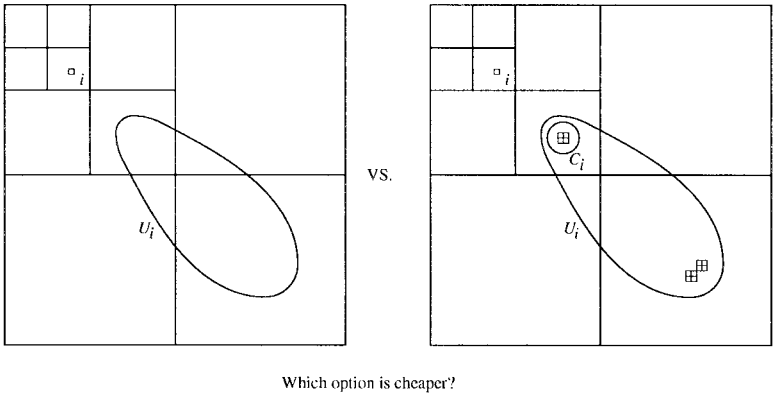


Fig. 4. Zerotree pruning in Step 2) of Algorithm I. (a) Zero out or prune U_i. (b) Send C_i and the best residue tree representations of nodes in C_i. Prune residue tree U_i if that is the cheaper option: i.e., $\sum_{j\in U_i} w_j^2 \leq \sum_{j\in C_i}(J^{(k)}{}_j + J^*_{U_j})$. This pruning process starts from the leaf nodes and proceeds toward the root nodes.

III. The SFQ Coding Algorithm

A. Tree Pruning Algorithm: Phase I (For Fixed Quantizer q and Fixed λ)

The proposed algorithm is designed to minimize an unweighted mean-squared error distortion measure, with distortion energy measured directly in the transform domain to reduce computational complexity. In this tree-pruning algorithm, we also approximate the encoder output rate by the theoretical first-order entropy, which can be approached very closely by applying adaptive arithmetic coding. Our (phase I) tree-pruning operation assumes that the cost of sending the tree map information is independent of the cost of sending the data given the tree map, an approximation that will be accounted for and corrected later in phase II. Thus, there will be no mention of the tree-map rate in this phase of the algorithm, where the goal will be to search for that spatial subtree $S^*_{data} \preceq T$ whose *data* cost is minimum in the rate-distortion sense.

The lowpass band of coefficients at the coarsest scale cannot (by definition) be included in residue trees, since residue trees refer only to descendants. The lowpass band quantizer operates independently of other quantizers. Therefore, we code this band separately from other highpass bands. The quantizer applied to the lowpass band is selected so that the operating slope on its R–D curve matches the overall absolute slope λ on the convex hull of the operational R–D curve for the "highpass" coder; i.e., we invoke the necessary condition that at optimality both coders operate at the same slope on their operational rate-distortion curves, else the situation can be improved by stealing bits from one coder to the other until equilibrium is established.

The following algorithm is used for a fixed value of q and λ to find the best $S \preceq T$. Note that the iteration count k is used as a superscript where needed. $n_i^{(k)}$ refers to the binary zerotree map (at the kth iteration of the algorithm) indicating the presence $[n_i^{(k)} = 0]$ or absence $[n_i^{(k)} = 1]$ of a zerotree associated with node i of the tree. Recall that $n_i^{(k)} = 0$ implies that all descendants of i (i.e., elements of U_i) are set to zero at the k iteration. $S^{(k)}$ refers to the (current) best subtree obtained after k iterations, with $S^{(0)}$ initialized to the full tree T. We will drop the "data" suffix from S to avoid cluttering. C_i refers to the set of children nodes (direct offspring) of node i. $J^*_{U_j}$ refers to the minimum or best (Lagrangian) cost associated with the residue tree U_j of node j, with this cost being set to zero for all leaf nodes of the full tree. $J_j^{(k)}$ is the (Lagrangian) cost of quantizing node j (with w_j and $\hat{w}_j$ denoting the unquantized and quantized values of the wavelet coefficient at node j, respectively) at the kth iteration of the algorithm, with $D_j^{(k)}$ and $R_j^{(k)}$ referring to the distortion and rate components, respectively. $p_j^{(k)}$ is the probability at the kth iteration, i.e., using the statistics from the set $S^{(k)}$, of the quantization bin associated with node j. We need to superscript the tree rate (and hence the Lagrangian) cost with the iteration count k, because the tree changes topology (i.e., gets pruned) at every iteration, and we assume a global entropy coding scheme. Finally, we assume that the number of levels in the spatial tree is indexed by the scale parameter l, with $l = 0$ referring to the coarsest (lowest frequency) scale.

Algorithm 1:

- *Step 0 (Initialization)*: Set $S^{(0)} \leftarrow T$; set the iteration count $k \leftarrow 0$. For all leaf nodes j of T, set $J^*_{U_j} = 0$.

$$\begin{aligned} S^{(0)} &\leftarrow T, \\ k &\leftarrow 0, \\ J^*_{U_j} &\leftarrow 0, \qquad \forall j \in \text{leaf nodes of } T. \end{aligned}$$

- *Step 1 (Probability Update—Needed Due to the Use of Entropy Coding)*: Update the probability estimates for all nodes in $S^{(k)}$, i.e., update $p_i^{(k)}$, $\forall i \in S^{(k)}$, where

$$p_i^{(k)} = \frac{\text{no. of coeffs. quantized to bin no.} \left\lfloor \left(\frac{w_i}{q} + 0.5 \right) \right\rfloor}{\text{no. of coeffs. in } S^{(k)}}.$$

- *Step 2 (Zerotree Pruning, see Fig. 4)*: Set tree-depth count $l \leftarrow$ maximum depth of $S^{(k)} - 1$. For every node i at current tree-depth l of $S^{(k)}$, determine if it is cheaper to zero out or to keep its best residue tree U_i in a rate-distortion sense. Zeroing out or pruning U_i incurs a cost

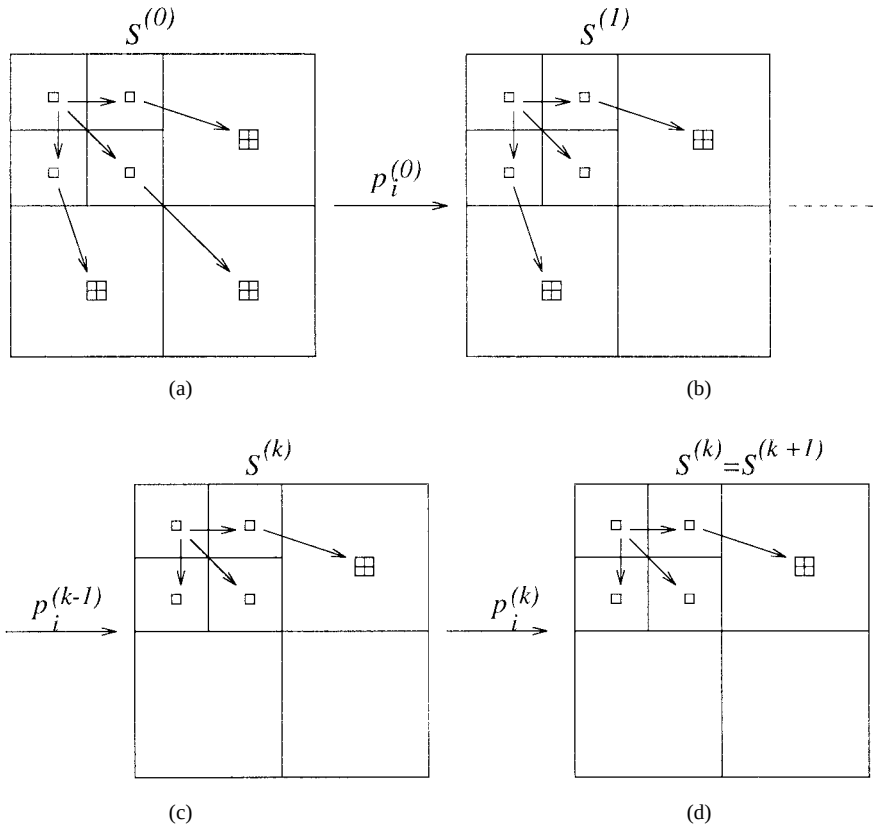


Fig. 5. The iterative Algorithm I to finding S^*_{data} for a fixed value of q and λ. (a) $S^{(0)}$ is initialized as the full tree T. (b) $S^{(1)}$ is obtained by running Step 2) (zerotree pruning) using statistics $p^{(0)}$. (c) At the kth iteration, $S^{(k)}$ is obtained by running Step 2) using $p_i^{(k-1)}$. (d) The algorithm stops at the $(k+1)$th iteration when $S^{(k+1)} = S^{(k)}$, i.e., no new nodes get pruned in the $(k+1)$th iteration, and $S^{(k+1)}$ is the targeted optimal S^*_{data}.

equal to the energy of residue tree U_i [left-hand side of inequality (6)], while keeping U_i incurs the cost of sending C_i and the best residue tree representations of nodes in C_i [right-hand side. of inequality (6)]. That is

$$l \leftarrow \text{maximum depth of } S^{(k)} - 1$$
$$\forall i \in \text{depth } l \text{ of } S^{(k)}.$$

If

$$\sum_{j \in U_i} w_j^2 \leq \sum_{j \in C_i} [J_j^{(k)} + J^*_{U_j}]$$

then

$$\left\{ n_i^{(k)} = 0;\ J^*_{U_i} = \sum_{j \in U_i} w_j^2 \right\}$$

else

$$\left\{ n_i^{(k)} = 1;\ J^*_{U_i} = \sum_{j \in C_i} [J_j^{(k)} + J^*_{U_j}] \right\} \tag{6}$$

where

$$J_j^{(k)} = D_j^{(k)} + \lambda R_j^{(k)} = [w_j - \hat{w}_j^{(k)}]^2 + \lambda\{-\log_2 [p_j^{(k)}]\}. \tag{7}$$

- *Step 3 (Loop Bottom-Up Through All Tree Levels):* Set $l \leftarrow l - 1$ and go to Step 2) if $l \geq 0$.
- *Step 4 (Check for Convergence, Else Iterate)*: Using the values of $\{n_i^{(k)}\}$ for all $i \in S^{(k)}$ found by optimal pruning, carve out the pruned subtree $S^{(k+1)}$ for the next iteration. If $S^{(k+1)} \neq S^{(k)}$ (i.e., if some nodes got pruned), then increment the iteration count $k \leftarrow (k+1)$ and go back to Step 1) to update statistics and iterate again. Else, declare $S^*_{data} \leftarrow S^{(k+1)}$ as the converged pruned spatial tree associated with scalar quantizer choice q and rate-distortion slope λ. This uniquely defines the (locally) optimal zerotree map $\{n_i\}$ for all nodes $i \in T$. See Fig. 5 for a pictorial explanation of this algorithm.

Discussion:

1) Scalar frequency quantization (using stepsize q) of all the highpass coefficients is applied in an iterative fashion. At each iteration, a fixed tree $S^{(k)}$ specifies the coefficients to be uniformly quantized, and the pruning rule of Step 2) is invoked to decide whether coefficients are worthy of being retained or if they should be killed. As the decision of whether or not to kill the descendants U_j of node j cannot be made without knowing the best representation (and associated best cost) for residue tree U_j, the pruning operation must proceed from the bottom of the tree (leaves) to the top (root).
2) Note that in Step 2, we query whether or not it is worthwhile to send any of the descendants of node i. This is done by comparing the cost of zeroing out *all* descendants of node i (assuming that zerotree quantized

data incurs zero rate cost) to the best alternative associated with not choosing to do so. This latter cost is that of sending the children nodes of i together with the best cost of the residue trees associated with each of the children nodes. Since processing is done in a bottom-up fashion, these best residue tree costs are known at the time. The cheaper of these costs is used to dictate the zerotree decision to be made at node i, and is saved for future reference involving decisions to be made for the ancestors of i.

3) As a result of the pruning operation of Step 2, some of the spatial tree nodes are discarded. This affects the histogram of the surviving nodes, which is recalculated in Step 1 (initially the histogram associated with the full tree is used) whenever any new node gets pruned out.
4) The above algorithm is guaranteed to converge to a local optimal choice for S^*_{data}.

Proof: See Appendix.

A plausible explanation for the above proposition is the following: Recall that the motivation for our iterative pruning algorithm is that as the trees get pruned, the probability density functions (pdf) of the residue trees change dynamically. So, at each iteration we update to reflect the "correct" pdf's till the algorithm converges. The above Proposition shows that the gain in terms of Lagrangian cost comes from better representation of the pdf's of the residue trees in the iterations. In the above tree-pruning algorithm, the number of nonpruned nodes is monotonically decreasing before it converges, so the above iterative algorithm converges very fast! In our simulations, the above algorithm converges in less than five iterations in all our experiments.

Proposition 1: The above tree pruning algorithm converges to a local minimum.

5) In addition to being locally optimal, our algorithm can make claims to having global merits as well. While the following discussion is not intended to be a rigorous justification, it serves to provide a relative argument from a viewpoint of image coding. After the hierarchical wavelet transform, the absolute values of most of the highpass coefficients are small, and they are deemed to be quantized to zero. The pdf of the wavelet coefficients is approximately symmetric, and sharply peaked at zero [see Fig. 9(c)]. Sending those zero-quantized coefficients is not cost effective. Zerotree quantization efficiently identifies those zerotree nodes. A larger portion of the available bit budget is allocated for sending the larger coefficients that represent more energy. So, the resulting pdf after zerotree quantization is considerably less peaked than the original one [see Fig. 9(d)]. Suppose the algorithm decided to zerotree quantize residue tree U_i at the kth iteration, i.e., deemed the descendants of i to be not worth sending at the kth iteration. This is because the cost of pruning U_i, which is identical to the energy of U_i [left-hand side of inequality (6)], is less than or equal to the cost of sending it [right-hand side of inequality (6)], or

$$\sum_{j\in U_i} w_j^2 \le \sum_{j\in C_i} [J_j^{(k)} + J^*_{U_j}].$$

The decision to zerotree quantize the set U_i is usually because U_i consists mainly of insignificant coefficients (with respect to the quantizer stepsize q). Then, in the $(k+1)$th iteration, due to the trimming operation, the probability of "small" coefficients becomes smaller, i.e., the cost of sending residue tree U_i becomes larger at the $(k+1)$th iteration, $J_j^{(k+1)}$ goes up [since $D_j^{(k+1)} = D_j^{(k)}$ while $R_j^{(k+1)} > R_j^{(k)}$ in (7)], thus reinforcing the wisdom in killing U_i at the kth iteration. That is, if inequality (6) holds at the kth iteration, then with high probability it is also true at the $(k+1)$th iteration. Thus, our algorithm's philosophy of "once pruned it is pruned forever," which leads to a fast solution, is likely to be very close to the globally optimal point as well. Of course, for residue trees that only have a few large coefficients, zerotree quantization in an early iteration might affect the overall optimality. However, we expect that the probability of such subtrees to be relatively small for natural images. So, our algorithm is efficient globally.

B. Predicting the Tree: Phase II

Recall that our coding data structure is a combination of a zerotree map indicating which nodes of the spatial tree have their descendants set to zero, and the quantized data stream corresponding to the survivor nodes. The side information needed to send the zerotree map is obviously a key issue in our design. In the process of formulating the optimal pruned spatial-tree representation, as described in Section III-A, we did not consider the cost needed to send the tree description $\{n_i\}$. This is tantamount to assuming that the tree map is free, or more generally that the tree-map cost is independent of the choice of tree (i.e., all trees cost the same regardless of choice of tree). While this is certainly feasible, it is not necessarily an efficient coding strategy. In this section, we describe a novel way to improve the coding efficiency by using a prediction strategy for the tree *map* bits of the nodes of a given band based on the (decoded) *data* information of the associated parent band. We will see that this leads to a way for the decoder to *deduce* much of the tree-map information from the data field (sent top-down from lower to higher frequency bands), leaving the encoder to send zerotree bits only for nodes having unpredictable map information.

Due to the tight coupling between data and map information in our proposed scheme, the "best" tree representation, as found through Algorithm I (assuming that the map bits are decoupled from the data bits), needs to be updated to correct for this bias, and zerotree decisions made at the tree nodes need to be reexamined to check for possible reversals in decision due to the removal of this bias. In short, the maxim that "zerotree maps are not all equally costly" needs to be quantitatively reflected in modifying the spatially pruned

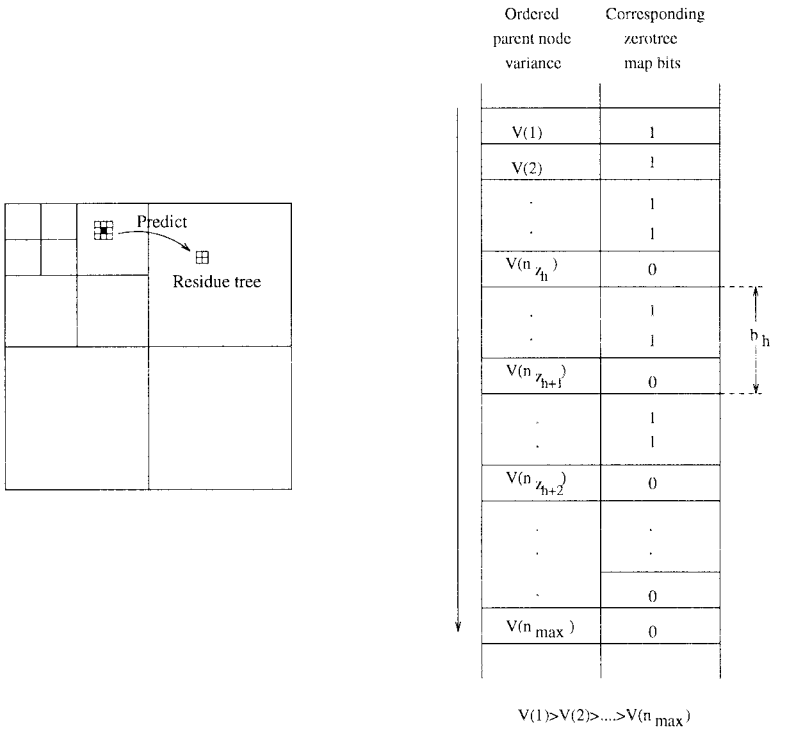


Fig. 6. Predicting the tree. (a) Basic idea of predicting the tree. The energy of a node is used to predict the significance/insignificance of its residue tree. (b) The optimal design of threshold T_h. The variances of the parent nodes of each band are ordered in decreasing magnitude, and the zerotree map bits corresponding to these nodes n_i are listed in the same order. T_h should be at least as small as the variance of the first node from the top of the list for which $n_i = 0$. The best T_h is designed in the rate-distortion sense to reflect the global data + map optimality.

subtree obtained from Algorithm I to a tree description that is the best in the global (data + map) sense. In this subsection, we describe how to accomplish this within the framework of predictive spatial-quantization while maintaining overall rate-distortion optimality. The basic idea is to predict the significance/insignificance of a residue tree from the energy of its parent [see Fig. 6(a)].

The predictability of subtrees depends on two thresholds output from the spatial quantizer. This is a generalization of the prediction scheme of Lewis and Knowles [1] for both efficiently encoding the tree and modifying the tree to optimally reflect the tree encoding. The Lewis and Knowles technique is based on the observation that the variance of a parent block centered around node i usually provides a good prediction of the energy of coefficients in the residue tree U_i. Their algorithm eliminates tree information by completely relying on this prediction. In order to improve performance, we incorporate this prediction in our algorithm by using it to represent the overall spatial tree information in a rate-distortion optimal way, rather than blindly relying on it to completely avoid sending tree-map information, which is, in general, suboptimal.

First, the variance of each (parent) node i is calculated as the energy of a 3×3 block centered at the corresponding wavelet coefficient of the node.[2] Note also that, for decodability or closed-loop operation, all variances should be calculated based on the *quantized* wavelet coefficients. We assume zero values for zerotree quantized coefficients.

Then, the variances of the parent nodes of each band are ordered in decreasing magnitude, and the zerotree map bits corresponding to these nodes, n_i, are listed in the same order [see Fig. 6(b)]. Two thresholds T_h and T_l are sent per band as (negligible) overhead information to assist the encoding.[3] Nodes whose parents have variances above T_h are assumed to be significant (i.e., n_i is assumed to be 1), thus requiring no tree information. Similarly, nodes with parents having energy below T_l are assumed to be insignificant (i.e., n_i is assumed to be 0), and they too require no side information. Tree-map information is sent only for those nodes whose parents have a variance between T_h and T_l. The algorithm is motivated by the potential for the Lewis and Knowles predictor to be fairly accurate for nodes having very high or very low variance, but to perform quite poorly for nodes with variance near the threshold.

This naturally leads to the question of optimization of the parameters T_h and T_l within the pruned spatial-tree quantization framework. We now address their optimal design. Clearly, T_h should be at least as small as the variance of the highest insignificant node (or the first node from the top of the list for which $n_i = 0$), since setting T_h any higher would require

[2] This "lowpass" filtering is needed to more accurately reflect the level of activity at the node.

[3] T_h and T_l are actually sent indirectly by sending the coordinates of two parent nodes, which have variances T_h and T_l respectively.

sending redundant tree information for residue tree, which could be inferred via the threshold. Likewise, T_l should be at least as large as the variance of the smallest significant node (or the first node from the bottom of the list for which $n_i = 1$).

Now let us consider if T_h should be made smaller in an optimal scenario. Let z_h denote the index of the node with variance equal to T_h (n_{z_h} must be 0), and suppose the number of 0 nodes down to n_{z_h} in the variance-ordered list is h. Note, if we shall reduce T_h at all, then, $h > 1$, since, T_h should made at least as small as the variance of the next node $n_{z_{h+1}}$ with $n_{z_{h+1}} = 0$. Let b_h be the position difference between n_{z_h} and $n_{z_{h+1}}$, then, this change to T_h saves us $\sum_{i=1}^{h} b_i$ bits, equal to the number of positions we move T_h down in the list (we assume that these binary map symbols have an entropy of one bit per symbol). Thus, changing n_{z_i} from 0 to 1 for all i from $i = 1$ to h decreases the map rate by

$$\Delta R_{map,\,h} = \sum_{i=1}^{h} b_i. \tag{8}$$

Of course, we know that reversing the map bits for the n_{z_i} nodes from 0 to 1 increases the *data* cost (in the rate-distortion or Lagrangian sense) as determined in the pruning phase of Algorithm I. So in performing a complete analysis of data + map, we reexamine and, where globally profitable, reverse the "data-only"-based zerotree decisions output by Algorithm I. The rule for reversing the decisions for the nodes n_{z_i} is clear: weigh the "data cost loss" versus the "map cost gain" associated with the reversals and reverse only if the latter outweighs the former. As we are examining rate-distortion tradeoffs, we need to use Lagrangian costs in this comparison.

It is clear that in doing the tree-pruning of Algorithm I, we can store (in addition to the tree map information $\{n_i\}$) the winning and losing Lagrangian costs corresponding to each node i, where the winning cost corresponds to that associated with the optimal binary decision n_i (i.e., $J^*_{U_i}$) and the losing cost corresponds to that associated with the losing decision $\overline{n_i}$ (i.e., the larger side of inequality (6). Denote by $\Delta J_{data,\,i}$ the magnitude of this Lagrangian cost difference for node i (note the inclusion of the data subscript for emphasis); i.e., $\Delta J_{data,\,i}$, for every parent node $i \in T$ is the absolute value of the difference between the two sides of the inequality (6) after convergence of Algorithm I. Then, the rule for reversing the n_{z_i} decisions from 0 to 1 for all nodes z_i from $i = 1$ to h is clearly

$$\text{If } \lambda \Delta R_{map,\,h} = \lambda \sum_{i=1}^{h} b_i > \sum_{i=1}^{h} \Delta J_{data,\,z_i}. \tag{9}$$

$$\text{Then reverse phase I decision : } n_{z_i} \leftarrow 1. \tag{10}$$

If inequality (9) is not true, no decision shall be made until we try to move T_h to the next 0 node. In this case, h is incremented until inequality (9) *is* satisfied for some larger h, whereupon n_{z_h} is reversed to 1 for all i from $i = 1$ to h. Then h is reset to 1 and the whole operation repeated until the entire list has been exhausted.

We summarize the design of T_h as follows.

Algorithm 2:

- *Step 1*: Order the variance of each parent node in decreasing magnitude, and list the zerotree map bits associated with these nodes in the same order.
- *Step 2*: Identify all the **zero** nodes n_{z_h} in the list, and record b_h, the difference in list position entry between the hth and the $(h+1)$th **zero** nodes.
- *Step 3*: Set $h = 1$.
- *Step 4*: Check if inequality (9) is satisfied for this value of h. If it is not, increment h, if possible, and go to Step 4). Else, reverse the tree-map bits n_{z_i} from 0 to 1 for all i from $i = 1$ to h, and go to Step 2).

T_h will point to the first zero node on the final modified list. It is obvious that Algorithm II optimizes the choice of T_h, using a global data + map rate-distortion perspective. A similar algorithm is used to optimize the choice of T_l. As a result of the tree prediction algorithm, the optimal pruned subtree S^*_{data} output by Algorithm I (based on data only) is modified to $S^*_{data+map}$.

C. Joint Optimization of Space-Frequency Quantizers

The above fast zerotree pruning algorithm tackles the innermost optimization (a) of (3), i.e., finds $S^*(q)$ for each scalar quantizer q (and λ, which is implied). As stated earlier, for a fixed quality factor λ, the optimal scalar quantizer is the one with stepsize q^* that minimizes the Lagrangian cost $J[q, S^*(q)]$, i.e., lives at absolute slope λ on the composite distortion-rate curve. That is, from (3), we have

$$\begin{aligned} q^* &= \arg \min_{q \in Q} \{J[q, S^*(q)]\} \\ &= \arg \min_{q \in Q} \{D[q, S^*(q)] + \lambda R[q, S^*(q)]\}. \end{aligned}$$

While faster ways of reducing the search time for the optimal q exist, in this work, we exhaustively search for all choices in a finite admissible list. Finally, the optimal slope λ^* is found using the convex search bisection algorithm as described in [23]. By the convexity of the pruned-tree rate-distortion function [23], starting from two extreme points in the rate-distortion curve, the bisection algorithm successively shrinks the interval in which the optimal operating point lies until it converges. The convexity of the pruned-tree rate-distortion function guarantees the convergence of the optimal space-frequency quantizer.

IV. Simulation Results

Experiments are performed on standard 512 × 512 grey-scale Lena, Barbara, and Goldhill images to test the proposed SFQ algorithm at several bit rates. Although our analysis of scalar quantizer performance assumed the use of orthogonal wavelet filters, simulations showed that little is lost in practice from using "nearly" orthogonal wavelet filters that have been reported in the literature to produce better perceptual results. We use the 7–9 biorthogonal set of linear phase filters of [18] in all our experiments. We use a 4-scale wavelet decomposition with the coarsest lowpass band having dimension 32 × 32.

(a) (b)

(c) (d)

Fig. 7. Original and decoded 512 × 512 Goldhill images. (a) Original image. (b) Bit rate = 1.0 b/pixel, PSNR = 36.70. (c) Bit rate = 0.5 b/pixel, PSNR = 33.37 dB. (d) Bit rate = 0.25 b/pixel, PSNR = 30.71 dB.

This lowest band is coded separately from the remaining bands, and the tree node symbols are also treated separately. For decodability, bands are scanned from coarse to fine scale, so that no child node is output before its parent node. The scalar quantization stepsize q takes values from the set $\{q : q = 7.5 + 0.1^{*}k,\ k = 1, 2, \cdots, 245\}$. An adaptive arithmetic coding [25], [26] is used to entropy code the quantized wavelet coefficients. All reported bit rates, which include both the data rate and the map rate, are calculated from the "real" coded bitstreams. About 10% of the total bit rate is spent in coding the zerotree map. The original Goldhill image is shown in Fig. 7(a), while the decoded Goldhill images at bit rates of 1.0 b/pixel, 0.5 b/pixel and 0.25 b/pixel, are shown in Fig. 7(b)–(d), respectively. The corresponding peak signal-to-noise ratios (PSNR's), defined as $10 \log_{10} [255^2/\text{mse}]$ at different bit rates for all three images, are tabulated in Table I.

We compare the performance of our SFQ algorithm in Fig. 8 with some of the high-performance image compression algorithms by Shapiro [2], Said and Pearlman [27], and Joshi, Crump, and Fisher [28]. Our simulations show that the SFQ algorithm is competitive with the best coders in the literature. For example, our SFQ-based coder outperforms 0.2 dB and 0.7 dB better in PSNR over the coder in [27] for Lena and Barbara, respectively.

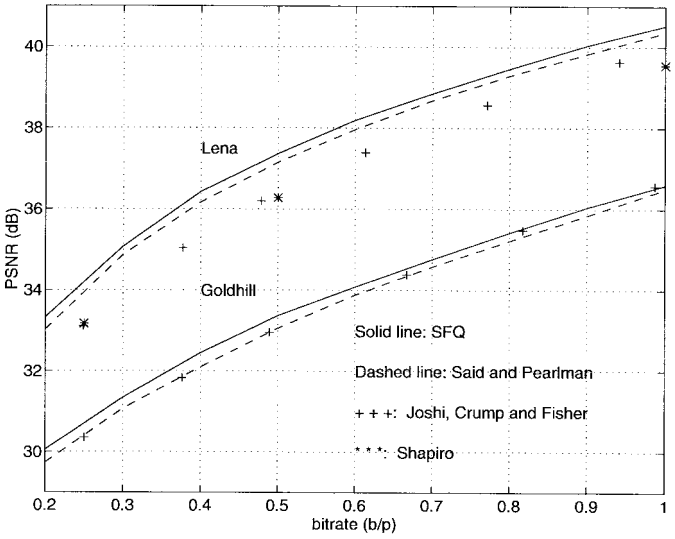
To illustrate how zerotree pruning in our proposed SFQ algorithm changes the statistics of the set of wavelet coefficients to be scalar quantized, we compare the probability distributions of the highpass wavelet coefficients of the Lena

TABLE I
CODING RESULTS OF THE SFQ ALGORITHM AT VARIOUS BIT RATES FOR THE STANDARD 512 × 512 LENA, BARBARA, AND GOLDHILL IMAGES

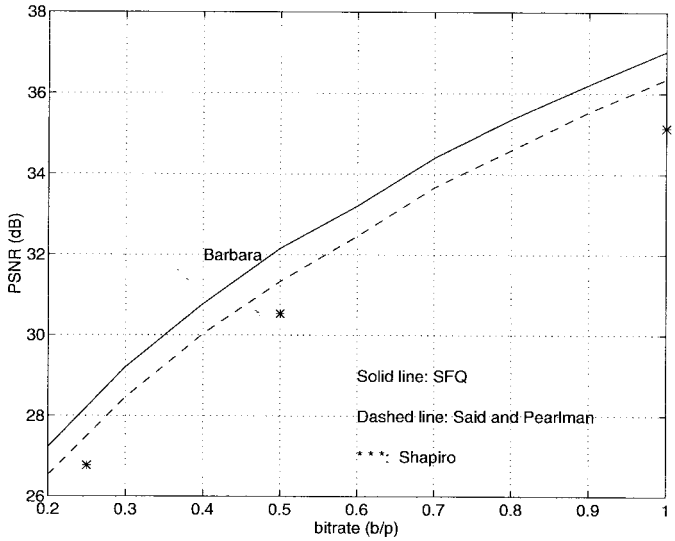
	Lena	Barbara	Goldhill
Rate (b/p)	PSNR (dB)	PSNR (dB)	PSNR (dB)
0.20	33.32	27.23	29.86
0.25	34.33	28.29	30.71
0.30	35.07	29.21	31.34
0.40	36.43	30.77	32.45
0.50	37.36	32.15	33.37
0.60	38.19	33.21	34.08
0.70	38.85	34.40	34.76
0.80	39.46	35.36	35.42
0.90	40.04	36.21	36.05
1.00	40.52	37.03	36.70

image with and without zerotree pruning. The comparison results are based on the output bit rate of 1.0 b/pixel. Fig. 9(a) and (b) display the scalar and space-frequency quantized four-level wavelet decompositions of the Lena image, respectively. A scalar quantization stepsize of $q = 7.8$ is applied to all highpass coefficients in Fig. 9(a) and 9(b). White regions in Fig. 9(b) represent pruned nodes by zerotree quantization. Histograms (after scalar quantization) of the full set of highpass coefficients in Fig. 9(a) and the pruned set in Fig. 9(b) are plotted in Fig. 9(c) and 9(d). The probability of zero index decreases from 0.7181 in the full set of highpass coefficients to 0.3755 in the pruned set, so zerotree pruning effectively flattens the probability density of the pruned set of highpass wavelet coefficients.

Another simulation explores the justification of using a single scalar quantization stepsize for all highpass bands of our decomposition. Earlier subband and wavelet coders have confirmed the importance of optimizing scalar quantization stepsizes to match the distribution of coefficients in each band. However, we observe that the distribution of coefficients in different bands are not nearly so different after zerotree pruning than before pruning. In particular, while the percentage of very small coefficients (i.e., those quantized to 0 or -1, 1) in each band differs significantly before pruning, it is very similar in most bands after pruning. Consequently, we conjecture that the slope of the operational rate-distortion functions for uniform scalar quantizers operating with the same stepsize in different bands will be approximately equal after zerotree pruning. Fig. 10 shows two sets of plots testing this conjecture for two collections of bands. In all cases, we fix the pruned tree produced by the SFQ algorithm at 1 b/pixel [see Fig. 9(b)], and we show the operational rate-distortion curves of uniform scalar quantizers applied to the coefficients remaining in the tree. The marks indicated on each curve show the operating points of the quantizers with the SFQ coding at 1 b/pixel. The first set of three curves in Fig. 10(a) shows the rate distortion (RD) curves for the three highest frequency bands (LH^1, HL^1, and HH^1 in Fig. 1) for the Lena image. The three slopes shown for this first set match very closely. The second set of curves in Fig. 10(b) shows the RD curves for three different frequency bands at the same orientation



(a)



(b)

Fig. 8. Comparisons of SFQ and other high-performance coders. (a) Comparisons for the Lena and Goldhill images. (b) Comparisons for the Barbara image.

(HH^1, HH^2,, and HH^3 in Fig. 1). Though the slopes of these three curves do not match as closely as for the first set, they are close enough to suggest that overall performance will not suffer significant degradation by using a common stepsize for all bands.

The SFQ algorithm developed in this paper is based on minimizing a squared-error objective function. This formulation can be naturally adapted to incorporate other distortion measures modeling perceptual sensitivity to error, though such variations are beyond the scope of the current work. For example, the rate-distortion relations considered throughout this paper could use a distortion measure incorporating both frequency and spatial weightings in characterizing visual sensitivity to errors; e.g., errors in higher frequency bands could be weighted less than those in lower bands, and errors in textured regions could be weighted less than those in smoothly varying

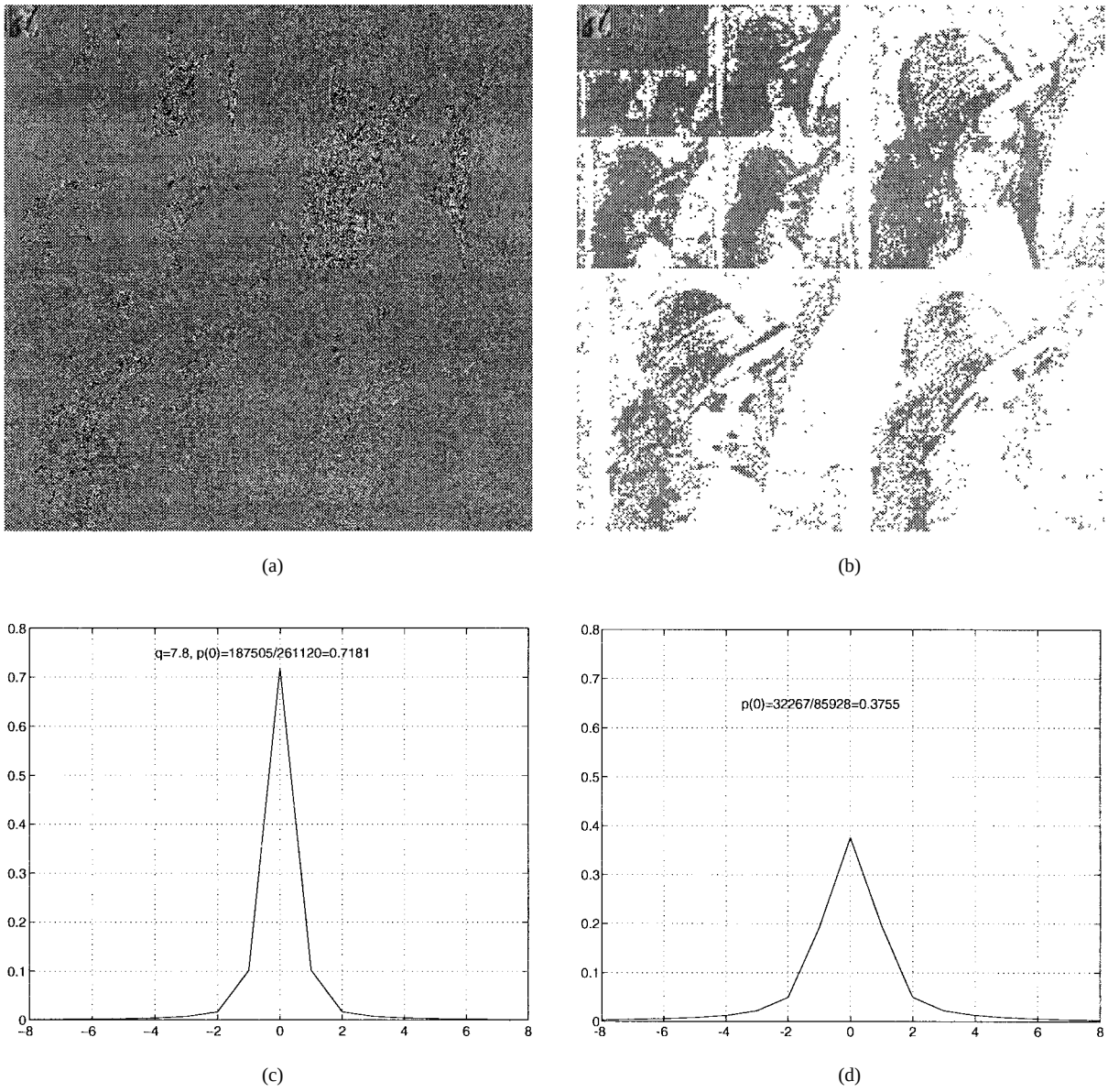


Fig. 9. Histograms of the quantized Lena images before and after zerotree pruning for $q = 7.8$ targeted at 1.0 b/pixel. (a) Scalar quantized four-level wavelet decomposition of the Lena image. A scalar quantization step size of $q = 7.8$ is applied to all highpass coefficients. All highpass quantized coefficients are shifted by 128 for display. (b) The space-frequency quantized four-level wavelet decomposition of the Lena image (white regions represent pruned nodes). (c) Histogram of the full set of highpass bands in (a). The probability of zero index is 0.7181. (d) Histogram of the pruned set of highpass bands after SFQ of (b). The probability of zero index is 0.3755.

regions. Although the current algorithm is not optimized for subjective quality, we felt that some subjective testing of the SFQ-coded images could provide insight into how the higher PSNR measurements of the SFQ algorithm relate to noticeable improvements in picture quality, and could hint at the important issues in designing an SFQ algorithm optimized for subjective quality. Thus, we report on several simple subjective comparisons of the SFQ results with those of the Shapiro's algorithm and the Joint Photographers Expert Group (JPEG) standard at various bit rates.

In direct comparison of SFQ- and JPEG-coded images, the SFQ-coded images show dramatically higher picture quality at all bit rates. While it has been noted that wavelet-based coding algorithms eliminate annoying blocking artifacts, the superior SFQ picture quality is much too pronounced to be accounted for by the different types of artifacts. The differences are more likely attributable to the approximately 2.5 dB higher PSNR of the SFQ results across the range of bit rates. A very dramatic example of the perceptual differences of SFQ and JPEG images coded at 0.25 b/pixel is shown in Fig. 11.

It would be tempting to conclude that SFQ and JPEG images having similar PSNR measurements would have comparable picture quality. However, in comparisons of SFQ and JPEG images coded to have identical PSNR measurements, the SFQ-coded images appear to have lower overall picture quality at almost all PSNR levels (at very low PSNR levels, it is difficult to make meaningful comparisons between *extremely* blocky JPEG images and *extremely* blurry SFQ images). Thus,

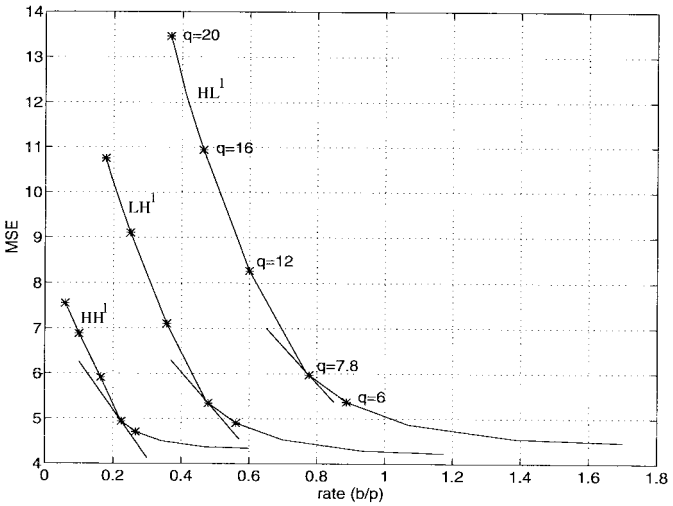
as an example, our tests show that the SFQ image coded at 0.5 b/pixel is clearly superior in picture quality to the JPEG image coded at 0.5 b/pixel, but inferior to the JPEG image coded at 0.95 b/pixel. In a crude attempt to identify coded images with similar subjective quality, we compared SFQ coded images at a fixed bit rate to JPEG images, allowing the JPEG bit rate to increase until the picture quality seemed roughly comparable. The results are plotted in Fig. 12.

In our final subjective tests, we compared the SFQ and Shapiro coding results at various bit rates, and we found little or no noticeable differences in overall picture quality. Actually, the differences between the two algorithms appeared mostly to reflect differences in spatial rate allocation across the picture. Since the control of this rate allocation does not consider subjective picture quality in either algorithm, the subjective difference between the two algorithms appears quite random (i.e., each algorithm looks better in some places and worse in others). There appeared to be little subjective evidence of the approximately 1 dB higher PSNR of the SFQ results. Further interpretation of the subjective test results are found in the discussions in the following section.

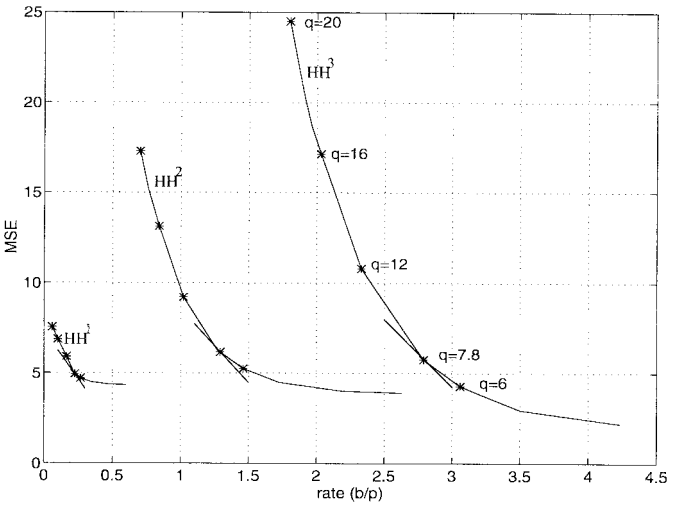
V. Discussion and Conclusions

The complexity of the SFQ algorithm lies mainly in the iterative zerotree pruning stage of the encoder, which can be substantially reduced with fast heuristics based on models rather than actual R–D data, which is expensive to compute. A good complexity measure is the running time on a specific machine. On a Sun SPARC 5, it takes about 20 s to run the SFQ algorithm for a fixed q and λ pair. On the same machine, Said and Pearlman's coder [27] takes about 4 s to run on encoding. Although our SFQ encoder is slower than that of Said and Pearlman's, the decoder is much faster because there are only two quantization modes used in the SFQ algorithm with the classification being sent as side-information. Our coder is suitable for applications such as image libraries, CD-ROM's, and centrally stored databases where asymmetric coding complexity is preferred.

The SFQ algorithm developed in this paper tests the hypothesis that high performance coding depends on exploiting both frequency and spatial compaction of energy in a space-frequency transform. The two simple quantization modes used in the SFQ algorithm put a limit its overall performance. This can be improved by introducing sophisticated schemes such as trellis-coded quantization and subband classification [12], [29] to exploit "packing gain" in the scalar quantizer, a type of gain quite separate and above anything considered in this paper. Zerotree quantization can be viewed as providing a mechanism for spatial classification of wavelet coefficients into two classes: i) "zerotree" pruned coefficients and ii) nonpruned coefficients. The tree-structure constraint of the SFQ classification permits us to efficiently implement RD-optimization, but produces suboptimal classification for coding purposes. I.e., if our classification procedure searched over a richer collection of possible sets of coefficients (e.g., including some nontree-structured sets), algorithm complexity would be increased, but improved results could be realized. In fact,



(a)



(b)

Fig. 10. Operational rate-distortion curves generated by quantizing a fixed pruned tree structure (bsed on the one obtained for Lena at 1.0 b/pixel) with different scalar quantization step sizes. (a) Operational rate-distortion curves of the three bands in the lowest scale (LH^1, HL^1 and HH^1 in Fig. 1. (b) Operational rate-distortion curves of the diagonal bands of three different scales (HH^1, HH^2, and HH^3 in Fig. 1). The slopes at the operation point ($q = 7.8$ are approximately equal for all bands. This justifies the choice of a single uniform scalar quantization step size for all bands.

performance improvements in other zerotree algorithms have been realized by expanding the classification to consider sets of coefficients other than purely tree-structured sets [27].

We list these possible improvements of the SFQ algorithm to highlight the fact that the SFQ algorithm makes no claim to establishing any limits to performance of image coding algorithms, even within the general paradigm of linear transform coding within which it is defined. In fact, it is built from relatively straightforward components, suggesting extensions that would almost certainly improve performance. Though complexity is one reason for not considering such extensions, our primary reason for not doing so in this paper is to provide a clear and direct test of our hypothesis that the optimal allocation of bit rate among space- and frequency-compacted signal energy is one of the most important characteristics of

Fig. 11. SFQ (left) and JPEG (right) decoded Lena images at 0.25 b/pixel.

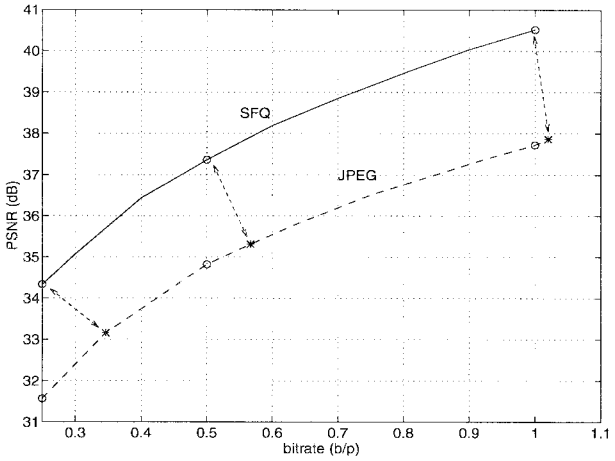


Fig. 12. Comparison of subjective qualities between SFQ and JPEG coded images. Arrows indicate the RD points at which both coders achieve comparable image quality.

high-performance image coding algorithms. Without zerotree quantization, the SFQ algorithm reduces to a most trivial wavelet coder; i.e., a wavelet transform followed by scalar quantizer with a common stepsize applied to all highpass bands. By providing this trivial coder with a limited and simple spatial quantization (zeroing out tree-structure sets of coefficients), and optimally allocating bitrate between spatial quantization and scalar quantization, we are able to achieve among the best coding results in today's literature.

We attribute the excellent performance of our coder to two important characteristics. First, the SFQ is built around a linear transform that allows signal energy to be compacted both in frequency and space, and quantization modes designed to match this characterization. Second, the SFQ provides a framework for optimizing (in the rate-distortion sense) the application of the quantization modes available to it. Insights into the importance of these characteristics are offered by comparing the SFQ to three other coding approaches from the literature. The algorithms of [17] use subband transforms with very sophisticated scalar and vector quantizers optimized for the statistics of each band. However, these algorithms lack a mechanism for efficiently identifying locations of compacted energy in the highpass bands (i.e., spatial quantization), thus limiting coding performance (the best results obtained from these algorithms are 1.1 dB below SFQ for Lena). We conclude that the SFQ achieves superior coding performance with a much less sophisticated scalar quantizer because its use of zerotree quantization allows it to better exploit the spatial compaction of high-frequency energy. The algorithm of [2] uses the wavelet transform as well as zerotree quantization in an embedded coding algorithm. However, zerotree quantization is applied in [2] to minimize distortion rather than to optimize rate-distortion performance, and we believe this difference accounts for most of the PSNR advantage of the SFQ algorithm seen in Fig. 8. We should note that the embedded structure of the algorithm of [2] makes direct comparison of these two algorithms difficult. Finally, it is interesting to compare the SFQ algorithm with the rate-distortion optimized version of JPEG proposed in [30]. These two algorithms are built around very similar rate-distortion optimization frameworks, with the algorithms of [30] using block discrete cosine transforms (DCT's) instead of the wavelet transform, and using runlengths of zeros instead of zerotrees. The R–D optimization provides a large gain over standard JPEG (0.7 dB at 1 b/pixel for Lena), but the final PSNR results (e.g., 39.6 dB at 1 b/pixel for Lena) remain about 0.9 dB below SFQ for the Lena image at a bit rate of 1 b/pixel. We interpret these results as reflecting the fact that the block DCT is not as effective as the wavelet transform at compacting high-frequency energy around edges (i.e., blocks containing edges tend to spread high-frequency energy among many coefficients.)

To complete our conclusions, we offer some final comments on the subjective quality of SFQ coded images. Although the SFQ offers noticeably improved picture quality compared with the JPEG algorithm at 1 b/pixel, reflecting a difference of nearly 3 dB in PSNR, comparisons with the algorithm of [2] showed no noticeable improvement in picture quality, despite being 1 dB higher in PSNR. This result is a direct consequence of our rate-distortion optimization procedure. The key difference between these two algorithms lies in how bit rate is distributed spatially across the image. In [2], this spatial allocation of bit rate is dictated by when coefficients fall below a threshold. In the SFQ algorithm, spatial allocation of bit rate is governed by the rule that each bit is applied to the location where it gives the biggest reduction in distortion. While the second strategy must (and does) produce higher PSNR, it may lead to bit allocations that do not give higher picture quality. For example, bits invested in highly textured areas often give greater reduction in distortion than bits invested in smooth areas. However, it is widely recognized that distortion in smooth areas can yield much lower picture quality than the equivalent distortion in textured areas. We interpret the disappointing subjective quality of the SFQ coded images as reflecting a mismatch between the squared-error distortion measure that governs our spatial allocation of rate, and the subjective distortion as perceived by the viewer. We conclude that optimum spatial allocation of available bit rate is an important feature of high-performance image coding, but we emphasize that the optimization criterion should reflect as closely as possible the true coding objective—typically, subjective picture quality. Modification of the SFQ algorithm to incorporate such criterion remains a topic for future research.

APPENDIX
PROOF OF PROPOSITION 1: ALGORITHM I CONVERGES TO A LOCAL OPTIMUM

We will show that $J[S^{(k+1)}] \leq J[S^{(k)}]$, thereby establishing the proposition, since the number of iterations is guaranteed to be finite, given that the tree is always being pruned.

Since $S^{(k+1)}$ is a pruned version of $S^{(k)}$, let us refer to the set of tree coefficients pruned out in the $(k+1)$th iteration as $\Delta S^{(k+1)}$, i.e.,

$$\Delta S^{(k+1)} = S^{(k)} \backslash S^{(k+1)}$$

or

$$S^{(k)} = S^{(k+1)} \cup \Delta S^{(k+1)}. \tag{11}$$

Let us now evaluate the costs $J[S^{(k+1)}]$ and $J[S^{(k)}]$, as follows:

$$\begin{aligned} J[S^{(k+1)}] &\overset{(a)}{=} \sum_{i \in S^{(k+1)}} J_i^{(k+1)} + \sum_{j \in T \backslash S^{(k+1)}} w_j^2 \\ &\overset{(b)}{=} \sum_{i \in S^{(k+1)}} J_i^{(k+1)} + \sum_{j \in \Delta S^{(k+1)}} w_j^2 + \sum_{j \in T \backslash S^{(k)}} w_j^2 \end{aligned} \tag{12}$$

where (a) above follows from the cost of $S^{(k+1)}$ being the sum of the Lagrangian cost $J_i^{(k+1)}$ of all nodes in $S^{(k+1)}$ plus the cost of zerotree quantizing the set $T \backslash S^{(k+1)}$, and (b) follows from (11).

Similarly, we have

$$\begin{aligned} J[S^{(k)}] &\overset{(a)}{=} \sum_{i \in S^{(k)}} J_i^{(k)} + \sum_{j \in T \backslash S^{(k)}} w_j^2 \\ &\overset{(b)}{=} \sum_{i \in S^{(k+1)}} J_i^{(k)} + \sum_{j \in \Delta S^{(k+1)}} J_j^{(k)} + \sum_{j \in T \backslash S^{(k)}} w_j^2 \end{aligned} \tag{13}$$

where (a) above follows from the definition of $J[S^{(k)}]$ and (b) above follows from (11).

We therefore have

$$\begin{aligned} J[S^{(k)}] - J[S^{(k+1)}] &= \sum_{i \in S^{(k+1)}} [J_i^{(k)} - J_i^{(k+1)}] \\ &\quad + \sum_{j \in \Delta S^{(k+1)}} [J_j^{(k)} - w_j^2] && (14) \\ &\geq \sum_{i \in S^{(k+1)}} [J_i^{(k)} - J_i^{(k+1)}] && (15) \\ &= \lambda \sum_{i \in S^{(k+1)}} [R_i^{(k)} - R_i^{(k+1)}] && (16) \\ &= \lambda |S^{(k+1)}| \sum_{bin=-\infty}^{+\infty} \\ &\quad - p^{(k+1)}(bin) \log_2 \left[\frac{p^{(k)}(bin)}{p^{(k+1)}(bin)} \right] && (17) \\ &\geq 0 && (18) \end{aligned}$$

where (14) follows from (12) and (13). (15) follows because the second summation of (14) is greater than or equal to zero [else $\Delta S^{(k+1)}$ would not have been pruned out in the $k+1$th iteration, by hypothesis]. (16) follows, since $J_i = D_i + \lambda R_i$, with $D_i^{(k)} = D_i^{(k+1)} = (w_i - \hat{w}_i)^2$. (17) expresses the rates for the kth and $(k+1)$th iterations in terms of their first-order entropies induced by the distributions $\{p^{(k)}(bin)\}$ and $\{p^{(k+1)}(bin)\}$, respectively [note that bin is the histogram bin number and $|S^{(k+1)}|$ refers to the cardinality or number of elements in $S^{(k+1)}$]. (18) follows from the fact that the summation in (17) is the Kullback–Leibler distance or relative entropy between the distributions $\{p^{(k)}(bin)\}$ and $\{p^{(k+1)}(bin)\}$, which is nonnegative.

REFERENCES

[1] A. Lewis and G. Knowles, "Image compression using the 2-D wavelet transform," *IEEE Trans. Image Processing*, vol. 1, pp. 244–250, Apr. 1992.

[2] J. M. Shapiro, "Embedded image coding using zerotrees of wavelet coefficients," *IEEE Trans. Signal Processing*, vol. 41, pp. 3445–3463, Dec. 1993.

[3] A. Gersho and R. M. Gray, *Vector Quantization and Signal Compression.* Boston, MA: Kluwer, 1992.

[4] G. D. Forney, R. M. Gray, and M. Ostendorf Dunham, "Finite-state vector quantization for waveform coding," *IEEE Trans. Inform. Theory*, vol. 31, pp. 348–359, May 1985.

[5] H. S. Malvar, *Signal Processing with Lapped Transforms.* Norwood, MA: Artech House, 1992.

[6] B. Ramamurthi and A. Gersho, "Classified vector quantization of images," *IEEE Trans. Commun.*, vol. 34, pp. 1105–1115, Nov. 1986.

[7] E. A. Riskin, "Optimal bit allocation via the generalized BFOS algorithm," *IEEE Trans. Inform. Theory*, vol. 37, pp. 400–402, Mar. 1991.
[8] J. W. Woods and S. O'Neil, "Subband coding of images," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. 34, pp. 1278–1288, Oct. 1986.
[9] Y. Linde, A. Buzo, and R. M. Gray, "An algorithm for vector quantizer design," *IEEE Trans. Commun.*, vol. 28, pp. 84–95, 1980.
[10] P. H. Westerink, D. E. Boekee, J. Biemond, and J. W. Wood, "Subband coding of images using vector quantization," *IEEE Trans. Commun.*, vol. 36, pp. 713–719, June 1988.
[11] J. Max, "Quantization for minimum distortion," *IEEE Trans. Inform. Theory*, pp. 7–12, Mar. 1960.
[12] M. W. Marcellin and T. R. Fischer, "Trellis coded quantization of memoryless and Gaussian–Markov sources," *IEEE Trans. Commun.*, vol. 38, no. 1, pp. 82–93, Jan. 1990.
[13] I. Daubechies, "Orthonormal bases of compactly supported wavelets," *Commun. Pure Appl. Math.*, vol. XLI, pp. 909–996, 1988.
[14] S. Mallat, "A theory for multiresolution signal decomposition: The wavelet representation," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 11, pp. 674–693, July 1988.
[15] M. Vetterli and C. Herley, "Wavelets and filter banks: Theory and design," *IEEE Trans. Signal Processing*, vol. 40, pp. 2207–2232, Sept. 1992.
[16] P. Westerink, "Subband coding of images," Ph.D. dissertation, Delft Univ. Technol., Oct. 1989.
[17] Y. H. Kim and J. W. Modestino, "Adaptive entropy coded subband coding of images," *IEEE Trans. Image Processing*, vol. 1, pp. 31–48, Jan. 1992.
[18] M. Antonini, M. Barlaud, P. Mathieu, and I. Daubechies, "Image coding using wavelet transform," *IEEE Trans. Image Processing*, vol. 1, p. 205–221, Apr. 1992.
[19] P. J. Burt and E. H. Adelson, "The Laplacian pyramid as a compact image code," *IEEE Trans. Commun.*, vol. 31, pp. 532–540, 1983.
[20] R. J. Clarke, *Transform Coding of Images.* New York: Academic, 1985.
[21] J. W. Woods, Ed., *Subband Image Coding.* Boston, MA: Kluwer, 1991.
[22] N. Farvardin and J. W. Modestino, "Optimum quantizer performance for a class of non-Gaussian memoryless sources," *IEEE Trans. Inform. Theory*, vol. 30, pp. 485–497, May 1984.
[23] Y. Shoham and A. Gersho, "Efficient bit allocation for an arbitrary set of quantizers," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. 36, pp. 1445–1453, Sept. 1988.
[24] K. Ramchandran and M. Vetterli, "Best wavelet packet bases in a rate-distortion sense," *IEEE Trans. Image Processing*, vol. 2, pp. 160–175, Apr. 1993.
[25] G. Langdon, "An introduction to arithmetic coding," *IBM J. Res. Develop.*, vol. 28, no. 2, pp. 135–149.
[26] I. Witten, R. Neal, and J. Cleary, "Arithmetic coding for data compression," *Commun. ACM*, vol. 30, no. 6, pp. 520–540.
[27] A. Said and W. A. Pearlman, "A new, fast, and efficient image codec based on set partitioning in hierarchical trees," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 6, pp. 243–250, June 1996.
[28] R. L. Joshi, V. J. Crump, and T. R. Fisher, "Image subband coding using arithmetic and trellis coded quantization," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 5, pp. 515–523, Dec. 1995.
[29] R. L. Joshi *et al.*, "Comparison of different methods of classification in subband coding of images," *IEEE Trans. Image Processing*, submitted for publication, 1995.
[30] M. Crouse and K. Ramchandran, "Joint thresholding and quantizer selection for decoder-compatible baseline JPEG," in *Proc. ICASSP'95*, Detroit, MI.
[31] Z. Xiong, N. P. Galatsanos, and M. T. Orchard, "Marginal analysis prioritization for image compression based on a hierarchical wavelet decomposition," in *Proc. ICASSP'93*, Minneapolis, MN, vol. 5, pp. 546–549.
[32] Z. Xiong, K. Ramchandran, and M. T. Orchard, "Joint optimization of scalar and tree-structured quantization of wavelet image decomposition," in *Proc. 27th Annu. Asilomar Conf. Signal, Systems, and Computers*, Pacific Grove, CA, Nov. 1993, pp. 891–895.

Zixiang Xiong (S'92–M'96) received the B.S.E.E. degree in 1987 from Wuhan University, Wuhan, China, the M.A. degree in mathematics in 1991 from the University of Kansas, Lawrence, the M.S.E.E. degree in 1992 from the Illinois Institute of Technology, Chicago, and the Ph.D. degree in electrical engineering in 1996 from the University of Illinois at Urbana-Champaign. Currently, he is a post-doctoral fellow at Princeton University, Princeton, NJ.

His main research interest lies in image and video compression.

Kannan Ramchandran (M'93) received the B.S. degree from the City College of New York, and the M.S. and Ph.D. degrees from Columbia University, New York, all in electrical engineering, in 1982, 1984, and 1993, respectively.

From 1984 to 1990, he was a Member of the Technical Staff at the Telecommmunications Research and Development Area, AT&T Bell Laboratories, involved in optical fiber carrier systems. From 1990 to 1993, he was a Graduate Research Assistant at the Center for Telecommunications Research, Columbia University. Since 1993, he has been with the University of Illinois, Urbana-Champaign, where he is currently an Assistant Professor in the Electrical and Computer Engineering Department and a Research Assistant Professor at the Beckman Institute and the Coordinated Science Laboratory. His research interests include multirate signal processing and wavelets, image and video compression, and robust image communication.

Dr. Ramchandran was the recipient of the 1993 Elaihu I. Jury Award at Columbia University for the best doctoral dissertation in the area of systems, signal processing, or communications. He received an NSF Research Initiation Award in 1994, an Army Research Office Young Investigator Award in 1996, and an NSF Career Award in 1997. He received the 1997 Senior Best Paper Award from the IEEE Signal Processing Society for a paper (with M. Vetterli) that appeared in the IEEE TRANSACTIONS ON IMAGE PROCESSING in April 1993.

Michael T. Orchard (M'88) was born in Shanghai, China, and grew up in New York. He received the B.S. and M.S. degrees in electrical engineering from San Diego State University, San Diego, CA, in 1980 and 1986, respectively, and the M.A. and Ph.D. degrees in electrical engineering from Princeton University, Princeton, NJ, in 1988 and 1990.

He worked at the Government Products Division of Scientific Atlanta from 1982 to 1986, developing passive sonar DSP applications. Since 1988, he has consulted with the Visual Communications department of AT&T Bell Laboratories From 1990 to 1995, he was an Assistant Professor with the Department of Electrical and Computer Engineering, University of Illinois, Urbana-Champaign, where he served as Associate Director of the Image Laboratory of the Beckman Institute. Since summer, 1995, he has been an Associate Professor with the Department of Electrical Engineering, Princeton University, Princeton, NJ. His research interests include image and video coding with emphasis on motion estimation and compensation, wavelet representations of images and video, and image display.

Dr. Orchard received the National Science Foundation Young Investigator Award in 1993 and the Army Research Office Young Investigator Award in 1996.