Wavelet-Based Image Coder with Entropy-Constrained Lattice Vector Quantizer (ECLVQ)

Won-Ha Kim, *Member, IEEE*, Yu-Hen Hu, and Truong Q. Nguyen, *Senior Member, IEEE*

***Abstract*— This paper presents an algorithm that jointly optimizes a lattice vector quantizer (LVQ) and an entropy coder in a subband coding at all ranges of bit rate. Estimation formulas for both entropy and distortion of lattice quantized subband images are derived. From these estimates, we then develop dynamic algorithm optimizing the LVQ and entropy coder together for a given entropy rate. Compared to previously reported min–max approaches, or approaches using asymptotic distortion bounds, the approach reported here quickly designs a highly accurate optimal entropy-constrained LVQ. The corresponding wavelet-based image coder also has better coding performance compared to other subband coders that use entropy-constrained LVQ, especially at low bit rates.**

***Index Terms*— Entropy coding, image coding, lattice vector quantization, wavelet.**

I. Introduction

In a coding system with entropy coding after quantization, a quantizer produces a stream of codeword indices which are entropy coded. Since an entropy coder uses fewer bits to identify the more frequent indices, the bit length required to represent an index is determined by its frequency of occurrence, and is thus variable. This indicates that the final bit rate is determined by the entropy of the index stream, not by the fixed bit rate of the quantizer. Therefore, a quantizer must be designed to take both probability of codeword index and distortion of quantization into account. Previous works [1]–[3] also indicated that a coder with good performance is a result from matching of an entropy coding to a quantization scheme rather than from sophisticated quantization algorithm. As a result, a key to making efficient use of energy compaction by subband coding is matching the quantizer-entropy coder combination according to information distribution at each subband.

Among various quantization approaches, the *Lattice Vector Quantizer* (LVQ) has been successfully used in a practical coding system, because of its low complexity and superb performance [1]–[3]. By using a lattice, or set of regularly spaced points, LVQ predetermines encoding and decoding rules, and so does not require a codebook. Consequently, a lengthy training process accompanied by a heavy computation burden is not necessary when using LVQ. There also exist fast encoding and decoding algorithms making use of simple rounding operations for the regular lattice structures [4].

The initial use of a lattice quantizer for transform coding of images was made by Sayood *et al.* [5]. Their lattice quantizer was designed for the memoryless uniform source. Fischer [2] developed the cubic lattice quantizer, motivated by the geometric properties of the Laplacian source. The lattice quantizer was restricted to large vector dimension, and used the cubic lattices lying on only the most probable pyramidal surface for a given quantizer bit rate. So it is named pyramid vector quantizer (PVQ). For applying any type of lattice to quantizing wavelet transform coefficients, Barlaud *et al.* [6] used all lattice points within the outermost pyramid by counting the number of lattice points on each surface. They performed bit allocation by the formula derived from the high quantizer bit rate distortion bound. Thus, their coder does not contain a procedure for matching LVQ and entropy coding. Recently, Yusof *et al.* [3] developed a subband coder with entropy-constrained LVQ. They allocated entropy rate (that is, the bit rate after entropy coding) to each subband by using a high quantizer bit rate formula which is basically equivalent to the one used by Barlaud *et al.* Subsequently, the entropy is estimated for high rate uniform threshold quantize (UTQ) and used in the design of an LVQ yielding allocated entropy. Since the coder in [3] assumes quantization with high bit rate while performing entropy allocation, the LVQ's in [3] are not well matched to entropy coding at low bit rates.

This paper presents an algorithm that jointly optimizes both the LVQ and entropy coder in subband coding so that the LVQ and entropy coder are well matched at all bit rates. The approach in this paper differs from previous works in the following ways. First, we derive estimation formulas for both entropy and distortion of LVQ-quantized subband images. The novel approaches of those estimation formulas are as follows. 1) We have developed an estimation formula for codeword entropy as a function of the scaling factor, which has not been done in previous works. 2) The partition of vector space according to the geometric structure of the source probability density function (*pdf*) yields a more accurate estimation of the probability that a source vector is in a Voronoi cell. 3) In estimating the probability that a source vector is assigned to a codeword, a maximum-likelihood (ML)

Manuscript received October 30, 1996; revised December 26, 1997.

W.-H. Kim is with Los Alamos National Laboratories, Computer Research/Applications (CIC-3), Los Alamos, NM 87545 USA.

Y.-H. Hu is with the Department of Electrical and Computer Engineering, University of Wisconsin-Madison, Madison, WI 53706 USA.

T. Q. Nguyen is with the Department of Electrical and Computer Engineering, Boston University, Boston, MA 02215 USA.

Publisher Item Identifier S 1057-7130(98)04669-2.

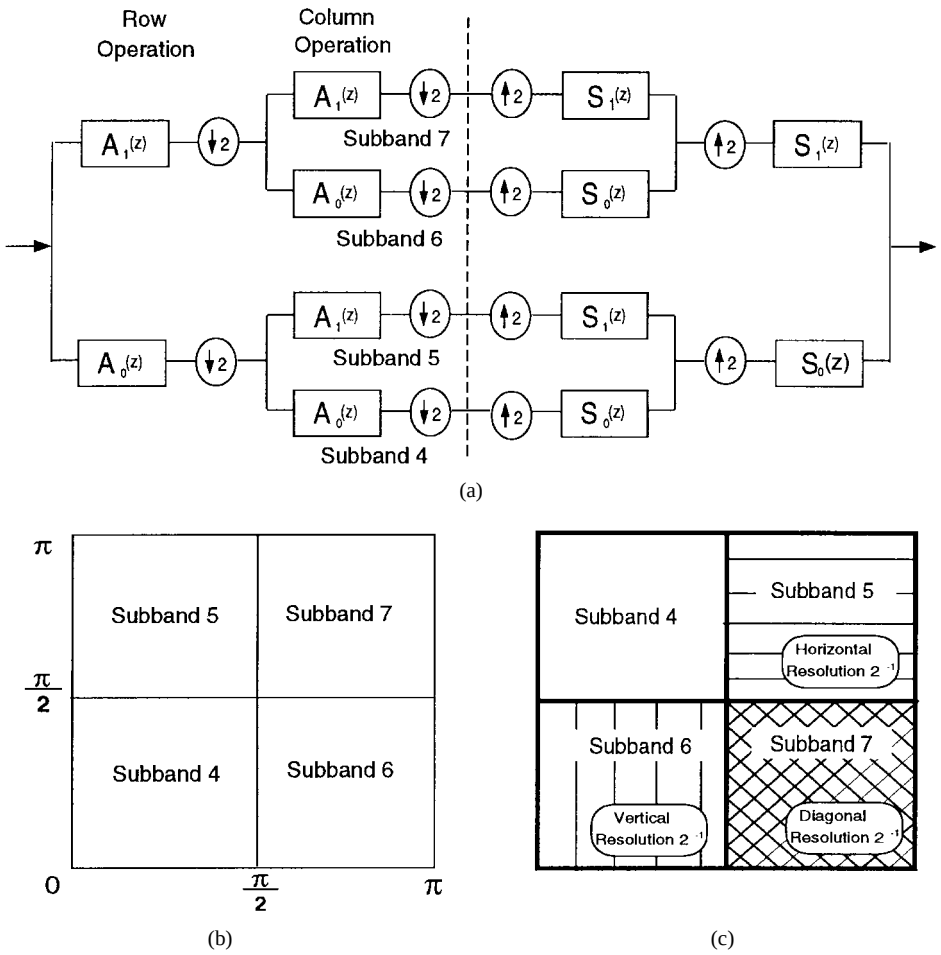


Fig. 1. (a) A tree-structured connections of separable two-dimensional QMF's, (b) its subband decomposition map, and (c) multiresolution map. The leaf node set of this wavelet tree is $L(S) = \{4, 5, 6, 7\}$.

estimator is developed, which offers a better estimation when the number of source vectors is less than the number of lattice points. The estimates are shown to be accurate enough to be practically used for image coding even at low entropy rate. Second, from these two expressions, we develop an entropy allocating algorithm for subbands having different variances. Consequently, unlike previous approaches using high quantizer bit rate formulas, the proposed algorithm adaptively and quickly designs the entropy constrained LVQ. The results match reasonably well to the optimal ones across the whole range of entropy rates. When the proposed algorithm is applied to wavelet based image coding, the coding performance also surpasses that of any previously reported subband coder with an entropy-constrained LVQ, especially at low bit rates.

The remainder of this paper is organized as follows. In Section II, a brief review of wavelet based subband coding which adopts the combination of the lattice vector quantization and entropy coder is presented with corresponding notations and definitions. In Section III, formulas to approximate LVQ distortion and entropy are derived as functions of the scaling factor, and the accuracy of these formulas is discussed. In Section IV, an optimization algorithms to determine the optimal scaling factors for entropy-constrained LVQ is developed. In Section V, the application of the proposed algorithm to image coding is discussed.

II. Wavelet-Based Subband Coding (SBC)

A. Tree-Structured Quadrature Filter (QMF) Banks

Fig. 1 illustrates a separable two-dimensional dyadic subband decomposition and its multiresolution analysis, where the analysis and synthesis filters are realized by the tree-structured connections of one-dimensional quadrature mirror filter (QMF) banks. The analysis bank transforms the input signal into subsequences, each representing a distinct subband. If the QMF is designed from a wavelet basis, the QMF tree realizes the multiresolution analysis with *perfect reconstruction* (PR) system, in which the reconstructed signal is a delay version of the input signal. The perfect reconstruction property ensures that the reconstruction error of the subband coding (SBC) is mainly due to quantization.

We denote the set of nodes on a QMF tree by S. Nodes $2i$ and $2i+1$ are the child nodes of node i. Node 1 is the root node. The branch from node i to node $2i$ corresponds to an analysis low-pass filter $A_0(z)$ followed by a downsampler or, in synthesis banks, a synthesis low-pass filter $S_0(z)$ following an upsampler. The branch from node i to node $(2i+1)$ corresponds to either an analysis high-pass filter $A_1(z)$ followed by a down-sampler or a synthesis high-pass filter $S_1(z)$ following an upsampler. The *depth* of node i is denoted by $d_i = \lfloor \log_2 i \rfloor$,

where $\lfloor \cdot \rfloor$ is the floor function, and then the corresponding level for two-dimensional wavelet decomposition is $d_i/2$. Let $L(S)$ denote the set of leaf nodes for the tree S. For example, the set of leaf nodes of the conventional 3-level 10-band wavelet tree is $L(S) = \{5, 6, 7, 17, 18, 19, 64, 65, 66, 67\}$, and the subband decomposition of node 65 is *lllllh* with tree depth of $d_i = 6$, where h and l denote high-pass filtering and low-pass filtering, respectively.

B. Design of Wavelet-Based Subband Coding (SBC)

Let H_i be the coding source (entropy) for encoding the signals at subband i. The transmission bit rate of this subband is $H_i/2^{d_i}$ bits/s because the subband signals are sampled at a rate of $1/2^{d_i}$ samples/s. Since the lattice codewords are already known, the lattice quantization needs no side information. Thus, the total signal transmission bit rate is $R_H = \sum_{i \in L(S)} H_i/2^{d_i}$. We use D_i to denote the mean squared distortion of subband i, which comes from the quantization distortion. The subband distortion D_i is inherently decreased by the coding source H_i. The reconstruction error of the SBC with respect to wavelet tree S is obtained as $D(S) = \sum_{i \in L(S)} (D_i/2^{d_i})$ [7].

Consequently, for wavelet tree S and an entropy budget R_H, the task of designing wavelet based subband coder (SBC) is a constrained optimization problem stated as follows:

$$\text{minimize}\, D(S) = \sum_{i \in L(S)} \frac{D_i}{2^{d_i}} \quad \text{subject to} \sum_{i \in L(S)} \frac{H_i}{2^{d_i}} \le R_H. \tag{1}$$

C. Lattice Vector Quantization (LVQ)

Consider an n-dimensional real vector space $\boldsymbol{R}^n$ with an l^1 norm-based distance metric $d(\boldsymbol{X}, \boldsymbol{Y}) = ||\boldsymbol{X} - \boldsymbol{Y}||$, where $\boldsymbol{X}, \boldsymbol{Y} \in \boldsymbol{R}^n$. (The reason l^1 norm is used will be explained in Section III.) A lattice is a regular arrangement of points in $\boldsymbol{R}^n$. The codewords of a lattice quantizer are lattice points [8]. So the lattice codebook $C = \{\boldsymbol{Y}_1, \boldsymbol{Y}_2, \cdots, \boldsymbol{Y}_M\}$ whose bit rate is R bits/sample is formed by selecting $|C| = 2^{nR} = M$ lattice points about the origin.

The fundamental Voronoi cell V_0 is located at the origin. Since the partition of the Voronoi cells to lattice points obeys the nearest neighbor rule and the lattice points are regularly spaced, the Voronoi cells of the lattice quantizer are translations of the fundamental cell. So, the volume of the Voronoi cell V_i of the lattice point $\boldsymbol{Y}_i$ is $\text{Vol}(V_i) = \text{Vol}(V_0)$. If a lattice is c-time scaled, the norm of a lattice point is an integral multiple of the scaling factor; that is, $||\boldsymbol{Y}_i|| = c \cdot k$ where k is an integer. The volume of the scaled Voronoi cell is calculated by $\text{Vol}(cV_i) = c^n \text{Vol}(V_0)$ [8].

Define C_k as the set of code vectors whose norms are ck, that is, $C_k = \{\boldsymbol{Y}_i \in C \mid ||\boldsymbol{Y}_i|| = ck\}$. Let $C_s(k) = |C_k|$ be the number of codewords on the surface at a distance ck from the origin. Then the radius of the LVQ codebook whose bit rate is R bits/sample can be found as

$$m = \max \left\{ q \in \boldsymbol{Z}^+ \left| \sum_{k=0}^{q} C_s(k) \le 2^{nR} \right. \right\}. \tag{2}$$

The codebook radius m is the normalized distance to the farthest lattice points from the origin. Therefore,

$$C = \bigcup_{k=0}^{m} C_k \quad \text{and} \quad |C| = \sum_{k=0}^{m} |C_k| = \sum_{k=0}^{m} C_s(k).$$

Values of $C_s(k)$ have been calculated in [6] for popularly used lattice types. Then, the code radius truncates the vector space into the granular and overload regions such as

$$\begin{aligned} \text{Granular region:} \quad & B^{gr}(cm) = \{\boldsymbol{X} \in \boldsymbol{R}^n : ||\boldsymbol{X}|| \le cm\} \\ \text{Overload region:} \quad & B^{ov}(cm) = \{\boldsymbol{X} \in \boldsymbol{R}^n : ||\boldsymbol{X}|| > cm\}. \end{aligned}$$

For given bit rate R and lattice type, the encoding and decoding procedure of an LVQ is as follows.

- Encoding
  1) Determine code radius m from (2).
  2) Choose a proper scaling factor c.
  3) Normalize a source vector $\boldsymbol{X}$ by the scaling factor c, so that the normalized vector $\tilde{\boldsymbol{X}} = \boldsymbol{X}/c$.
  4) If $\tilde{\boldsymbol{X}}$ is in the granular region, that is, $||\tilde{\boldsymbol{X}}|| < m$, map $\tilde{\boldsymbol{X}}$ to the nearest lattice point; else if $\tilde{\boldsymbol{X}}$ is in the overload region, that is, $||\tilde{\boldsymbol{X}}|| > m$, $\tilde{\boldsymbol{X}}$ is rescaled to the outermost surface enclosing the granular region, and is then mapped to the nearest lattice point on the surface.
  5) Index the position of the mapping lattice, e.g., $\boldsymbol{Y}$, and transmit the index.
- Decoding
  1) Recover the position of $\boldsymbol{Y}$ from the received index.
  2) Obtain decoding vector $\hat{\boldsymbol{X}} = c\boldsymbol{Y}$.

An example of a lattice quantization scheme for l^1 norm is shown in Fig. 2.

As seen in the encoding procedure, once the code radius is determined, the size of each region is controlled by the scaling factor. If the lattice is scaled up, distortion in the granular area will predominate, since a greater proportion of vectors will appear in this area. Likewise, if the lattice is scaled down, distortion in the overload area will predominate. Therefore, designing the LVQ requires choosing the scaling factor which results in the best tradeoff between granular area and overload area. So (1) can be restated as the design problem of the entropy-constrained wavelet based subband coding using lattice vector quantizer in optimizing scaling factor:

$$\begin{aligned} \boldsymbol{c}^* = \arg \min_{\{c_i | i \in L(S)\}} & \left\{ \sum_{i \in L(S)} \frac{D_i(c_i)}{2^{d_i}} \right\} \\ \text{subject to} & \sum_{i \in L(S)} \frac{H_i(c_i)}{2^{d_i}} \le R_H. \end{aligned} \tag{3}$$

III. Estimations of Distortion and Entropy

In this section, estimations of the distortion and entropy of LVQ encoding wavelet coefficients are derived as functions of the scaling factor c. The advantage of the estimations is that the optimal scaling factor which minimizes the quantization

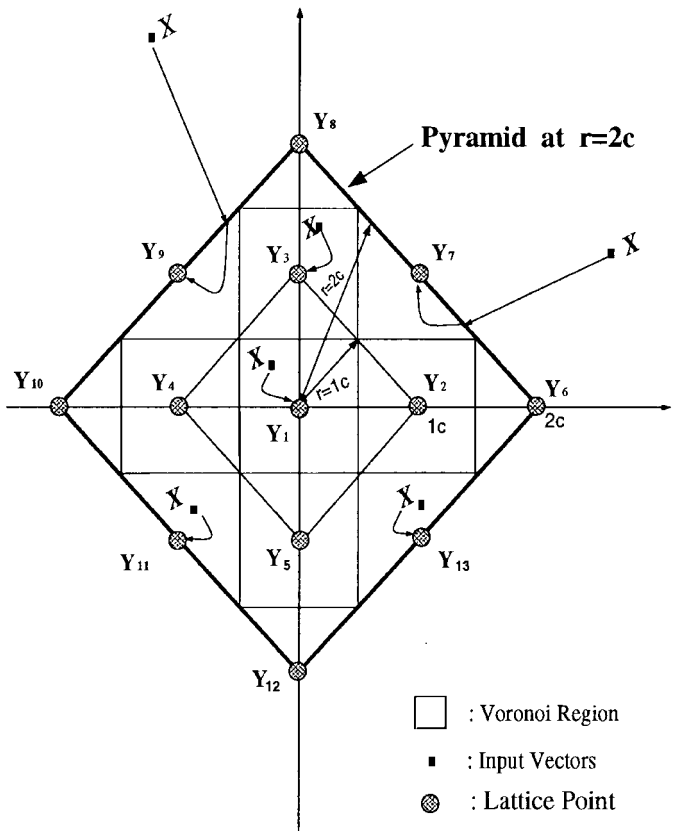


Fig. 2. Pyramidal lattice vector quantization scheme for two-dimen-sional cubic (Z^2) lattice. $B^{gr}(2c) = \{\boldsymbol{X}\colon \|X\|_1 \leq 2c\}$, $B^{ov}(2c) = \{\boldsymbol{X}\colon \|\boldsymbol{X}\|_1 > 2c\}$. $C = \{\boldsymbol{Y}_1, \boldsymbol{Y}_2, \cdots, \boldsymbol{Y}_{13}\}$, code radius $m = 2$, code size $|C| = 13$, bit rate $R = \log_2 |C|/\text{dimension} = 1.85$ bit/sample, lattice points on pyramid: $C_s(0) = 1$, $C_s(1) = 4$, $C_s(2) = 8$.

distortion for an entropy rate can be quickly determined without any measurements.

Previous research has verified that a Laplacian source was a suitable model for a wide class of subband transformed data including wavelet coefficients [6], [9], [10]. Therefore, a Laplacian source is the model used for the wavelet coefficients in this paper. Let x_i be a sequence of identical and independent distribution (i.i.d.) Laplacian random variables with zero mean and variance $2/\lambda^2$. Then, the joint *pdf* of an n-dimensional Laplacian random vector $\boldsymbol{X} = [x_1, x_2, \cdots, x_n]^T$ is transformed its gamma distribution in such way as

$$f(\boldsymbol{X}) = f(x_1, x_2, \cdots, x_n) = \prod_{i=1}^{n} f(x_i) = \left(\frac{\lambda}{2}\right)^n e^{-\lambda \sum_1^n |x_i|} = g(r), \qquad \text{where } r = \sum_1^n |x_i|.$$

Observing that $\|\boldsymbol{X}\|_1 = \sum_1^n |x_i|$, the Laplacian *pdf* is constant at the contour specified by l^1-norm distance. In order to be consistent with the source's geometric properties, the proper distance measure for a Laplacian source is l^1-norm. The equiprobability contour at r is $S_p(r) = \{\boldsymbol{X} \in S | \|\boldsymbol{X}\|_1 = r\}$, which forms a pyramid.

The $(n-1)$-dimensional pyramidal surface $S_p(r)$ encloses an n-dimensional hyperpyramid $V_p(r)$. The volumes of $V_p(r)$ and $S_p(r)$ are [11]

$$\text{vol}[V_p(r)] = \frac{2^n r^n}{\Gamma(n+1)} \quad \text{and vol}[S_p(r)] = \frac{2^n \sqrt{n} r^{n-1}}{\Gamma(n)}. \tag{4}$$

The Laplacian source and the gamma function are related as follows:

$$\int_{\boldsymbol{X} \in B^{gr}(cm)} f(\boldsymbol{X})\, d\boldsymbol{X} = \int_0^{cm} g(r)\, dr,$$
$$\int_{\boldsymbol{X} \in B^{ov}(cm)} f(\boldsymbol{X})\, d\boldsymbol{X} = \int_{cm}^{\infty} g(r)\, dr. \tag{5}$$

A lattice point can be identified by its coordinate; namely, by its norm and a label denoting its relative position on the surface specified by its norm. That is, $\boldsymbol{Y}_i \Longleftrightarrow \boldsymbol{Y}_{(r_i, l_i)}$ and $V_i \Longleftrightarrow V_{(r_i, l_i)}$, where $r_i = \|\boldsymbol{Y}_i\|$ and l_i is the l_i^{th} lattice point on the surface at a distance r_i from the origin. For example, in Fig. 2, the lattice points $\boldsymbol{Y}_1, \boldsymbol{Y}_2, \boldsymbol{Y}_3, \cdots, \boldsymbol{Y}_6, \boldsymbol{Y}_7$, and $\boldsymbol{Y}_8$, etc., are identified by $\boldsymbol{Y}_{(0,0)}, \boldsymbol{Y}_{(1,0)}, \boldsymbol{Y}_{(1,1)}, \cdots, \boldsymbol{Y}_{(2,0)}, \boldsymbol{Y}_{(2,1)}$ and $\boldsymbol{Y}_{(2,2)}$, etc., respectively. Denote the codeword probability of $\boldsymbol{Y}_i$ as $p(\boldsymbol{Y}_i) = p[\boldsymbol{Y}_{(r_i, l_i)}] \equiv p(r_i, l_i)$. Since the lattice Voronoi cells are congruent and the Laplacian source is symmetric with respect to the origin, $P[\boldsymbol{X} \in V_{(r,l)}]$ depends only on r and is independent of l, which implies that $P[\boldsymbol{X} \in V_{(r,l)}] = P[\boldsymbol{X} \in V_{(r,l')}]$ for integer l, l'.

Then, the mean-square distortion per symbol can be expressed as

$$D(c) = \frac{1}{n} \sum_{k=0}^{m} \left\{ \sum_{\boldsymbol{Y}_{(r,l)} \in C_k} \int_{\boldsymbol{X} \in V_{(r,l)}} |\boldsymbol{X} - \boldsymbol{Y}_{(r,l)}|^2 f(\boldsymbol{X})\, d\boldsymbol{X} \right\} \quad \text{where} \quad r = ck \tag{6}$$

and the entropy per symbol is given by

$$H(c) = -\frac{1}{n} \sum_{k=0}^{m} \left\{ \sum_{\boldsymbol{Y}_{(r,l)} \in C_k} P[\boldsymbol{Y}_{(r,l)}] \log_2 P[\boldsymbol{Y}_{(r,l)}] \right\} \quad \text{where} \quad r = ck. \tag{7}$$

A. Maximum-Likelihood Estimation of Codeword Probability $p(\boldsymbol{Y}_i)\{= p[\boldsymbol{Y}_{(r_i, l_i)}]\}$

In order to estimate entropy of codewords, one must first calculate the probability that a codeword is used for a given set of source vector, that is, $p(\boldsymbol{Y}_i)\{= p[\boldsymbol{Y}_{(r,l)}]\} = p(r, l)$. Let N be the total number of source vectors. It would be more tractable to calculate $p(r, l) = p(l/r)p(r)$ instead of directly calculating $p[\boldsymbol{Y}_{(r,l)}]$.

From the empirical probability, $p(r, l) = p(l/r)p(r)$ is obtained by

$$p(r, l) = \frac{\text{number of vectors mapped to the lattice at}(r, l)}{\text{total number of vectors}} = \frac{\omega}{N} = p(l/r) \cdot p(r) \tag{8}$$

where

$$p(r) = \frac{\text{number of vectors mapped to lattice points at } r}{\text{total number of vectors}} = \frac{N \cdot C_s(k) \cdot P[\boldsymbol{X} \in V_{(r,l)}]}{N} = C_s(k) \cdot P[\boldsymbol{X} \in V_{(r,l)}] \tag{9}$$

is the empirical probability that an input vector is mapped to a lattice point on the surface at $r = ck$, and

$$p(l/r) = \frac{\begin{pmatrix}\text{Among the vectors mapped to the lattice}\\ \text{points at } r, \text{ the number of vectors}\\ \text{mapped to the lattice point at } (r,l)\end{pmatrix}}{\text{number of vectors mapped to lattice points at } r} = \frac{\omega}{N \cdot C_s(k) \cdot P[\boldsymbol{X} \in V_{(r,l)}]}$$

is the empirical probability that the lattice point $\boldsymbol{Y}_{(r,l)}$ encodes a vector among the source vectors mapped to the lattice points at r.

In lattice vector quantization, codewords which do not encode any source vectors will not be transmitted, and hence should be excluded in calculating $p(\boldsymbol{Y}_i)\{= p[\boldsymbol{Y}_{(r_i,l_i)}]\}$. For example, if there are 20 lattice points located at distance r, and only 11 source vectors are assigned to the Voronoi cells at the distance r, then at most 11 codewords will be transmitted rather than 20 codewords. Hence, $p(l/r)$ must be greater than $1/11$, and cannot be equal to $1/20$. In order to calculate $p(l/r)$, one must know the number of lattice points actually encoding source vectors instead of using the number of all lattice points at r. Since the numbers of lattice points encoding source vectors are associated with different probabilities, the number of transmitted codewords at r must be estimated.

Define $\tilde{C}_k(\subseteq C_k)$ as the set of code vectors whose norms are $r = ck$, and each of which encodes at least one input vector (we only consider the lattice points used in the encoding process). Since the *pdf* of a Laplacian source is constant on the surface at r, the source vectors must be uniformly distributed on the surface. So the probability that a given source vector is encoded by a lattice point among the lattice points on the surface should be equal to $1/C_s(k)$. Consequently, among the source vectors mapped to the lattice points on the surface, the number of vectors mapped to the lattice point $\boldsymbol{Y}_{(r,l)} \in \tilde{C}_k$ (that is, ω) obeys a binomial distribution. Denote $N_r = N \cdot C_s(k) \cdot P[\boldsymbol{X} \in V_{(r,l)}]$. Then, the probability that $\omega = j$ is given by

$$P(\omega = j) = \binom{N_r}{j}\left(\frac{1}{C_s(k)}\right)^j\left(\frac{C_s(k)-1}{C_s(k)}\right)^{N_r - j}, \quad \text{for } \boldsymbol{Y}_{(r,l)} \in \tilde{C}_k.$$

By setting the likelihood function $L_j(\omega) = P(\omega)$, the maximum likelihood (ML) estimate for ω is given by [12]

$$\hat{\omega} = \arg\max_{\omega} L_j(\omega) = \arg\max_{\omega} P(\omega) = \begin{cases} 1, & \text{if } N \cdot P[\boldsymbol{X} \in V_{(r,l)}] < 1 \\ \dfrac{N_r}{C_s(k)} = N \cdot P[\boldsymbol{X} \in V_{(r,l)}], & \text{if } N \cdot P[\boldsymbol{X} \in V_{(r,l)}] \geq 1 \end{cases} = \max\{1,\, N \cdot P[\boldsymbol{X} \in V_{(r,l)}]\}.$$

Consequently, the estimated probability for $p(l/r)$ is

$$\hat{p}(l/r) = \frac{\hat{\omega}}{N \cdot C_s(k) \cdot P[\boldsymbol{X} \in V_{(r,l)}]} = \frac{\max\{1, N \cdot P[\boldsymbol{X} \in V_{(r,l)}]\}}{N \cdot C_s(k) \cdot P[\boldsymbol{X} \in V_{(r,l)}]}. \tag{10}$$

From (8)–(10), the estimated codeword probability is

$$\hat{p}[\boldsymbol{Y}_{(r,l)}] \equiv \hat{p}(r,l) = \frac{\max\{1,\ N \cdot P[\boldsymbol{X} \in V_{(r,l)}]\}}{N} = \max\left\{\frac{1}{N},\, P[\boldsymbol{X} \in V_{(r,l)}]\right\}. \tag{11}$$

Since the source vectors are uniformly distributed on the surface over r, each of lattice points of the encoding set is most likely to encode the same number of source vectors. The size of encoding set $|\tilde{C}_k|$ is estimated to be $|\tilde{C}_k| \cdot \hat{\omega} = N_r$, and it is given by

$$|\tilde{C}_k| = \frac{N_r}{\hat{\omega}} = \frac{N \cdot C_s(k) \cdot P[\boldsymbol{X} \in V_{(r,l)}]}{\max\{1, N \cdot P[\boldsymbol{X} \in V_{(r,l)}]\}} = C_s(k) \cdot \min\{1,\, N \cdot P[\boldsymbol{X} \in V_{(r,l)}]\}. \tag{12}$$

Example: Assume there are 20 lattice points on the surface at $r = ck$ and the Voronoi cell probability $P[\boldsymbol{X} \in V_{(r,l)}] = 0.01$.

When the size of input vectors $N = 200$,

$$\begin{aligned} N \cdot P[\boldsymbol{X} \in V_{(r,l)}] &= 200 \cdot 0.01 = 2 \\ |\tilde{C}_k| &= C_s(k) \cdot \min\{1,\, N \cdot P[\boldsymbol{X} \in V_{(r,l)}]\} \\ &= 20 \cdot \min\{1,\, 2\} = 20, \\ \hat{p}(l/r) &= \frac{\max\{1,\, N \cdot P[\boldsymbol{X} \in V_{(r,l)}]\}}{N \cdot C_s(k) \cdot P[\boldsymbol{X} \in V_{(r,l)}]} = \frac{\max\{1,\, 2\}}{200 \cdot 20 \cdot 0.01} \\ &= 0.05, \\ \hat{p}[\boldsymbol{Y}_{(r,l)}] &= \frac{\max\{1,\, N \cdot P[\boldsymbol{X} \in V_{(r,l)}]\}}{N} \\ &= \frac{\max\{1,\, 200 \cdot 0.01\}}{200} = 0.01. \end{aligned}$$

When the size of input vectors $N = 50$

$$\begin{aligned} N \cdot P[\boldsymbol{X} \in V_{(r,l)}] &= 50 \cdot 0.01 = 0.5 \\ |\tilde{C}_k| &= C_s(k) \cdot \min\{1,\, N \cdot P[\boldsymbol{X} \in V_{(r,l)}]\} \\ &= 20 \cdot \min\{1,\, 0.5\} = 10, \\ \hat{p}(l/r) &= \frac{\max\{1,\, N \cdot P[\boldsymbol{X} \in V_{(r,l)}]\}}{N \cdot C_s(k) \cdot P[\boldsymbol{X} \in V_{(r,l)}]} = \frac{\max\{1,\, 0.5\}}{50 \cdot 20 \cdot 0.01} \\ &= 0.1, \\ \hat{p}[\boldsymbol{Y}_{(r,l)}] &= \frac{\max\{1,\, N \cdot P[\boldsymbol{X} \in V_{(r,l)}]\}}{N} \\ &= \frac{\max\{1,\, 50 \cdot 0.01\}}{50} = 0.02. \end{aligned}$$

B. Approximation of Voronoi Cell Probability $P(\boldsymbol{X} \in V_i)$ for Laplacian Source

The approximations derived here are different from previous approximations [9], [13]. First, the random vector space is separated into four regions according to the geometric shape of the source *pdf*; and second, the statistical centroids of the

Voronoi cells are used to approximate the probability in a Voronoi cell instead of the geometrical centroids, i.e., the lattice points.

1) At $r = 0$: At the origin, a Laplacian source has the highest density and decays most rapidly, so it is improper to assume that a Laplacian *pdf* is constant inside the Voronoi cell at $r = 0$ (as is done in previous studies [9], [13]). A better estimation is to approximate $P(\boldsymbol{X} \in V_i)$ by the probability that a vector resides in the hyperpyramid which has the same volume as the Voronoi cell at the origin. From (4), the hyperpyramid radius a is determined by $2^n a^n/\Gamma(n+1) = c^n \text{vol}(V_0)$, or $a = c/2[\text{vol}(V_0)\Gamma(n+1)]^{1/n}$. Thus,

$$P(\boldsymbol{X} \in cV_0) = \int_{\boldsymbol{X} \in cV_0} f(\boldsymbol{X})\, d\boldsymbol{X} \approx \int_0^a g(r)\, dr = \left[1 - e^{-\lambda a} \sum_{k=0}^{n-1} \frac{(\lambda a)^k}{k!}\right]. \tag{13}$$

2) At $c \le r \le c(m-1)$: For $\boldsymbol{X} \in B^{gr}(r)$, the *pdf* is assumed to be approximately uniform or constant in a Voronoi cell. The lattice points are geometric centroids of Voronoi cells, but not the centroids in the statistical sense. Let $\tilde{\boldsymbol{Y}}_i$ be the statistical center of the Voronoi cell V_i. Then, by the Mean Value Theorem, the statistical center $\tilde{\boldsymbol{Y}}_i$ must satisfy [14]

$$P(\boldsymbol{X} \in V_i) = \int_{\boldsymbol{X} \in V_i} f(\boldsymbol{X})\, d\boldsymbol{X} = f(\tilde{\boldsymbol{Y}}_i) \cdot \text{vol}(V_i), \quad \text{where } \tilde{\boldsymbol{Y}}_i \in V_i.$$

We approximate the integration over a polytope V_i by the integration over the cubic having the same volume as the polytope. Then, for the polytope V_i with edge-length L (the edge length L of Z_n lattice, $D_n (n \ge 4)$ lattice and E_8 lattice are c, $2^{1/n}c$, and c, respectively [4])

$$\int_{\boldsymbol{X} \in V_i} f(\boldsymbol{X})\, d\boldsymbol{X} \approx \left(\frac{\lambda}{2}\right)^n \int_{\boldsymbol{Y}_i - (L/2, \cdots, L/2)}^{\boldsymbol{Y}_i + (L/2, \cdots, L/2)} e^{-\lambda |\boldsymbol{X}|}\, d\boldsymbol{X} = \left(\frac{\lambda}{2}\right)^n \prod_{j=1}^{n} \int_{y_j - (L/2)}^{y_j + L/2} e^{-\lambda |x_j|}\, dx_j.$$

Suppose that a lattice point $\boldsymbol{Y}_i$ has p zero components, then

$$\left(\frac{\lambda}{2}\right)^n \prod_{j=1}^{n} \int_{y_j - (L/2)}^{y_j + L/2} e^{-\lambda |x_j|}\, dx_j = \frac{1}{2^{(n-p)}} (1 - e^{-(L/2)\lambda})^p (e^{(L/2)\lambda} - e^{-(L/2)\lambda})^{n-p} \cdot e^{-\lambda \|\boldsymbol{Y}_i\|}. \tag{14}$$

At a certain distance, the Voronoi cell probabilities at the lattice points with fewer zero components dominate those at the lattice points with more zero components. The Voronoi cell probabilities are then approximated by the probability of the Voronoi cell whose lattice point is at the same distance and has the smallest number of zero components. The lattice point at $r = ck (k \ge 1)$ has at least $\max\{(n-k), 0\}$ zero components. Therefore, by replacing p of (14) with $\max\{(n-k), 0\}$, the approximation of Voronoi cell probability is

$$P(\boldsymbol{X} \in V_i) = \int_{\boldsymbol{X} \in V_i} f(\boldsymbol{X})\, d\boldsymbol{X} \approx f(\tilde{\boldsymbol{Y}}_i) \cdot \text{vol}(V_i) \approx \frac{1}{2^{\min\{k,n\}}} (1 - e^{-(L/2)\lambda})^{\max\{(n-k), 0\}} \cdot (e^{(L/2)\lambda} - e^{-(L/2)\lambda})^{\min\{k, n\}} e^{-\lambda r} \tag{15}$$

where $r = ck$, and $1 \le k \le m-1$.

3) At $r = cm$: About half the volume of the Voronoi cells at the outermost surface is included in the granular region. Hence,

$$P(\boldsymbol{X} \in V_{cm}) \approx \frac{1}{2^{\min\{m,n\}+1}} (1 - e^{-(L/2)\lambda})^{\max\{(n-m), 0\}} \cdot (e^{(L/2)\lambda} - e^{-(L/2)\lambda})^{\min\{m, n\}} e^{-\lambda cm} \tag{16}$$

where V_{cm} is a Voronoi cell whose lattice point is at $r = cm$.

4) At $r > cm$: A random vector in the overload area is rescaled to the pyramid $S_p(cm)$ at $r = cm$ that encloses the granular area, and each rescaled vector is mapped to its nearest lattice points on $S_p(cm)$. Ignoring the edge effect of $S_p(cm)$, random vectors are uniformly distributed around $S_p(cm)$, and the Voronoi cells on $S_p(cm)$ are congruent. Consequently, $P[\boldsymbol{X} \in B^{ov}(cm)] \approx C_s(m) P(\boldsymbol{X} \in V_{cm}^{ov})$ where V_{cm}^{ov} is a Voronoi cell covering the overload area. From (5),

$$P(\boldsymbol{X} \in V_{cm}^{ov}) \approx \frac{P[\boldsymbol{X} \in B^{ov}(cm)]}{C_s(m)} = \frac{\displaystyle\int_{cm}^{\infty} g(r)\, dr}{C_s(m)} = \frac{1}{C_s(m)} \cdot e^{-\lambda a} \sum_{k=0}^{n-1} \frac{(\lambda a)^k}{k!} \tag{17}$$

where $a = (cm/2)[\text{vol}(V_0)\Gamma(n+1)]^{1/n}$.

C. Distortion Estimate $D(c)$

The distortion is also calculated at four regions.

1) At $r = 0$: From (13), the distortion at the origin (i.e., $\boldsymbol{Y}_{(r,l)} = \mathbf{o}$) will be

$$D(c|r=0) = \frac{C_s(0)}{n} \int_{\boldsymbol{X} \in cV_0} |\boldsymbol{X}|^2 f(\boldsymbol{X})\, d\boldsymbol{X} \approx \frac{1}{n} \int_0^a r^2 g(r)\, dr = \frac{(n+1)}{\lambda^2} \left\{1 - e^{-\lambda a} \sum_{k=0}^{n+1} \frac{(a\lambda)^k}{k!}\right\} \tag{18}$$

where $a = (c/2)[\text{vol}(V_0)\Gamma(n+1)]^{1/n}$.

2) At $c \le r \le c(m-1)$: As in (15), the *pdf* is assumed to be constant in a Voronoi cell. Then

$$\frac{1}{n} \int_{\boldsymbol{X} \in V_i} |\boldsymbol{X} - \boldsymbol{Y}_i|^2 f(\boldsymbol{X})\, d\boldsymbol{X} \approx \frac{1}{n} \frac{P(\boldsymbol{X} \in V_i)}{\text{vol}(V_i)} \int_{\boldsymbol{X} \in V_i} |\boldsymbol{X} - \boldsymbol{Y}_i|^2\, d\boldsymbol{X} = f(\tilde{\boldsymbol{Y}}_i) \cdot G_n \cdot c^{n+2} \cdot [\text{vol}(V_i)]^{1+2/n}$$

where G_n is the dimensionless second moment for an n-dimensional polytope V_i. It is interpreted as the normalized

mean-square quantization error to uniformly distributed vector source [4]. G_n for several lattices have been calculated in [4].

Recalling that $\boldsymbol{Y}_i \Longleftrightarrow \boldsymbol{Y}_{(r_i,l_i)}$, $P[\boldsymbol{X} \in V_{(r,l)}] = P[\boldsymbol{X} \in V_{(r,l')}]$, the distortion at $c \le r \le c(m-1)$ is obtained as follows:

$$\begin{aligned} D[c|c \le r \le c(m-1)] &= \sum_{k=1}^{m-1} D(r=ck) \\ &= \frac{1}{n}\sum_{k=1}^{m-1}\left\{\sum_{\boldsymbol{Y}_{(r,l)}\in C_k}\int_{\boldsymbol{X}\in V_{(r,l)}} |\boldsymbol{X}-\boldsymbol{Y}_{(r,l)}|^2 f(\boldsymbol{X})\,d\boldsymbol{X}\right\} \\ &\approx \frac{1}{n}\sum_{k=1}^{m-1}\left\{\sum_{\boldsymbol{Y}_{(r,l)}\in C_k}\frac{P[\boldsymbol{X}\in V_{(r,l)}]}{\mathrm{Vol}[V_{(r,l)}]}\int_{\boldsymbol{X}\in V_{(r,l)}} \cdot\,|\boldsymbol{X}-\boldsymbol{Y}_{(r,l)}|^2\,d\boldsymbol{X}\right\} \\ &= \sum_{k=1}^{m-1} C_s(k)\cdot f[\tilde{\boldsymbol{Y}}_{(r,l)}]\cdot G_n\cdot\{\mathrm{Vol}[V_{(r,l)}]\}^{1+2/n} \end{aligned} \tag{19}$$

where $f[\tilde{\boldsymbol{Y}}_{(r,l)}]$ is derived in (15).

3) At $r = cm$: Using the same rationale as in (16), we have

$$D(c/r=cm) \approx \tfrac{1}{2}C_s(m)f(\tilde{\boldsymbol{Y}}_{cm})\cdot G_n\cdot[\mathrm{Vol}(V_{cm})]^{1+2/n} \tag{20}$$

where V_{cm} is a Voronoi cell whose lattice point is at $r = cm$.

4) At $r > cm$: The overload distortion is due to rescaling and quantization. Let $\hat{\boldsymbol{X}}$ be the rescaled vector of $\boldsymbol{X}[\in B^{ov}(cm)]$ on the pyramid $S_p(cm)$. Then $\hat{\boldsymbol{X}} = cm\boldsymbol{X}/\|\boldsymbol{X}\|$. The overload distortion will be

$$\begin{aligned} &D(r>cm) \\ &\quad = \text{rescaling distortion} + \text{quantization distortion} \\ &\quad = \frac{1}{n}\int_{\boldsymbol{X}\in B^{ov}(cm)} |\boldsymbol{X}-\hat{\boldsymbol{X}}|^2 f(\boldsymbol{X})\,d\boldsymbol{X} \\ &\qquad + \frac{1}{n}\int_{\boldsymbol{X}\in B^{ov}(cm)}\left(\int_{\hat{\boldsymbol{X}}\in S_p(cm)} \|\hat{\boldsymbol{X}}-\boldsymbol{Y}(\hat{\boldsymbol{X}})\|\,d\hat{\boldsymbol{X}}\right) \\ &\qquad \cdot f(\boldsymbol{X})\,d\boldsymbol{X} \\ &\quad \approx \frac{1}{n}\int_{\boldsymbol{X}\in B^{ov}(cm)} |\boldsymbol{X}-\hat{\boldsymbol{X}}|^2 f(\boldsymbol{X})\,d\boldsymbol{X} \end{aligned} \tag{21}$$

where $\boldsymbol{Y}(\hat{\boldsymbol{X}}) = \{\boldsymbol{Y}_i | \|\boldsymbol{Y}_i - \hat{\boldsymbol{X}}\| \le \|\boldsymbol{Y}_j - \hat{\boldsymbol{X}}\| \text{ for all } i \ne j\}$, that is, $\boldsymbol{Y}(\hat{\boldsymbol{X}})$ is the nearest lattice point to $\hat{\boldsymbol{X}}$. The last step of (21) is based on the following observation. For small value of c, the second term of (21) is dominated by the first term, and for large value of c, the granular distortion overwhelms the overload distortion itself. So the effect of the second term of (21) is negligible. Therefore, the overload distortion is directly integrated via a gamma function

$$\begin{aligned} &\frac{1}{n}\int_{\boldsymbol{X}\in B^{ov}(cm)} |\boldsymbol{X}-\hat{\boldsymbol{X}}|^2 f(\boldsymbol{X})\,d\boldsymbol{X} \\ &\quad = \frac{1}{n}\int_{cm}^{\infty}(r-cm)^2 g(r)\,dr \\ &\quad = \frac{e^{-\lambda cm}}{\lambda^2 n}\sum_{k=0}^{n-1}\frac{(n-k)(n+1-k)}{k!}(\lambda cm)^k. \end{aligned} \tag{22}$$

Combining the distortions (18)–(22) in each region into (6), and remembering that $\mathrm{Vol}(V_i) = c^n\mathrm{Vol}(V_0)$, for a Laplacian source, the LVQ distortion as a function of the scaling factor c can be estimated by the following formula

$$\begin{aligned} D(c) \approx{}& \frac{2}{\lambda^2}\left\{1 - e^{-\lambda a}\sum_{k=0}^{n+1}\frac{(a\lambda)^k}{k!}\right\} + G_n\cdot c^2\cdot[\mathrm{Vol}(V_0)]^{2/n} \\ &\cdot\sum_{k=1}^{m} C_s(k)\frac{(1-0.5\delta_{km})}{2^{\min\{k,n\}}}(1-e^{-(L/2)\lambda})^{\max\{(n-k),0\}} \\ &\cdot(e^{(L/2)\lambda} - e^{-(L/2)\lambda})^{\min\{k,n\}}e^{-\lambda ck} \\ &+ \frac{e^{-\lambda cm}}{\lambda^2 n}\sum_{k=0}^{n-1}\frac{(n-k)(n+1-k)}{k!}(\lambda cm)^k \end{aligned} \tag{23}$$

where $a = (c/2)[\mathrm{vol}(V_0)\Gamma(n+1)]^{1/n}$, and δ_{km} is the Kronecker delta: $\delta_{km} = 1$ if $k = m$, $\delta_{km} = 0$ if $k \ne m$.

D. Entropy Estimate $H(c)$

After plugging (11) into (7), the entropy per symbol at scaling factor c is given by

$$\begin{aligned} H(c) &= -\frac{1}{n}\sum_{k=0}^{m}\left[\sum_{\boldsymbol{Y}_{(r,l)}\in\tilde{C}_k}\frac{\max\{1, N\cdot P[\boldsymbol{X}\in V_{(r,l)}]\}}{N}\right. \\ &\qquad \left.\cdot\log_2\left(\frac{\max\{1, N\cdot P[\boldsymbol{X}\in V_{(r,l)}]\}}{N}\right)\right] \\ &= -\frac{1}{n}\sum_{k=0}^{m}\left[|\tilde{C}_k|\cdot\frac{\max\{1, N\cdot P[\boldsymbol{X}\in V_{(r,l)}]\}}{N}\right. \\ &\qquad \left.\cdot\log_2\left(\frac{\max\{1, N\cdot P[\boldsymbol{X}\in V_{(r,l)}]\}}{N}\right)\right]. \end{aligned}$$

From (12),

$$\begin{aligned} H(c) &= -\frac{1}{n}\sum_{k=0}^{m}\left[C_s(k)\cdot\min\{1, N\cdot P[\boldsymbol{X}\in V_{(r,l)}]\}\right. \\ &\qquad \cdot\frac{\max\{1, N\cdot P[\boldsymbol{X}\in V_{(r,l)}]\}}{N} \\ &\qquad \left.\cdot\log_2\left(\frac{\max\{1,\ N\cdot P[\boldsymbol{X}\in V_{(r,l)}]\}}{N}\right)\right] \\ &= -\frac{1}{n}\sum_{k=0}^{m}\left[C_s(k)\cdot P(\boldsymbol{X}\in V_r)\right. \\ &\qquad \left.\cdot\left(\log_2\max\left\{\frac{1}{N}, P(\boldsymbol{X}\in V_r)\right\}\right)\right]. \end{aligned} \tag{24}$$

As usual, N is the total number of vectors, and V_r is a Voronoi cell whose lattice point is on the surface at r.

For convenience, define a subfunction $\mathrm{Ent}[\alpha(x)]$ as

$$\mathrm{Ent}[\alpha(x)] = \alpha(x)\cdot\log_2\left(\max\left\{\frac{1}{N}, \alpha(x)\right\}\right).$$

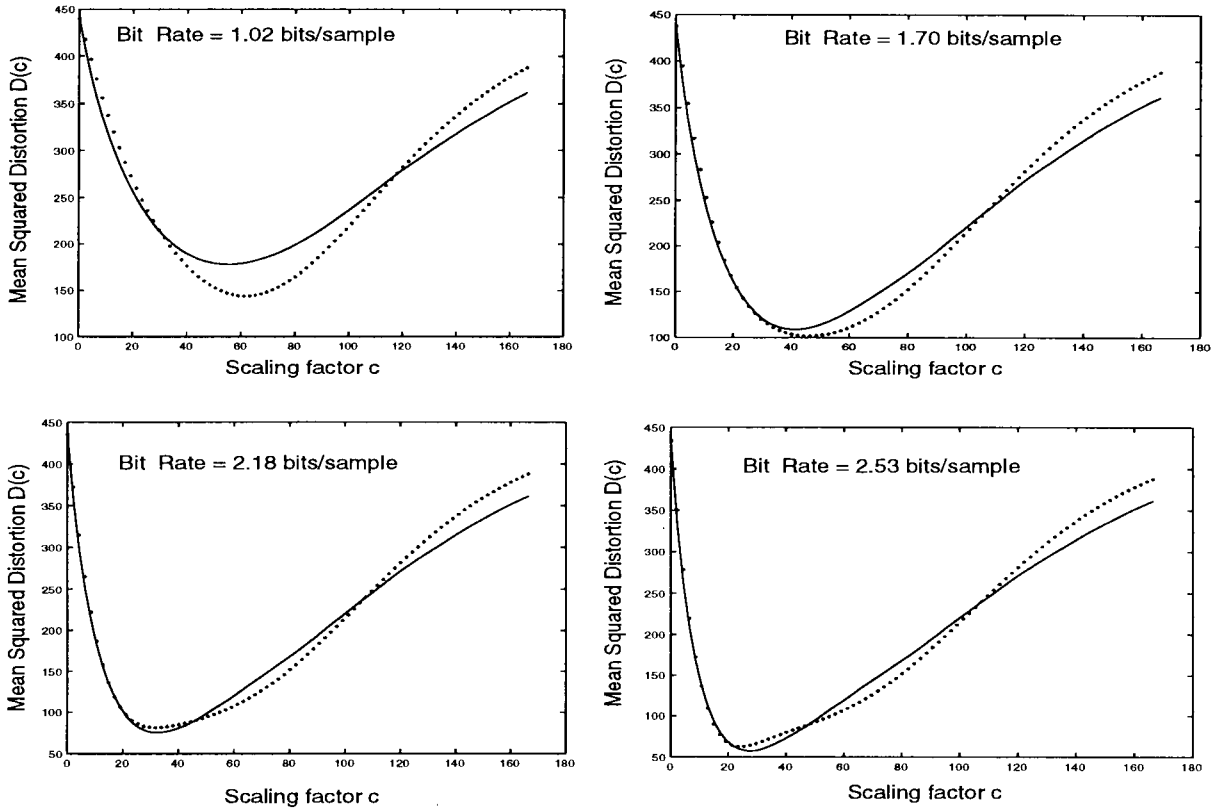


Fig. 3. Real distortion (solid line) and estimated distortion (dotted line) for subband 17 $(lllh)$ of 512 × 512 "Lena." Lattice type is D_4.

TABLE I
MINIMUM DISTORTION $D(c^*)$ AND ITS SCALING FACTOR c^*, AND AVERAGE ESTIMATION ERROR (%) FOR SUBBAND 17 $(lllh)$ OF 512 × 512 "LENA." D_4 LATTICE

	Real		Estimated		Average
Rate (bits/sample)	Minimum distortion $D(c^*)$	Factor of $D(c*)$	Minimum distortion $D(c^*)$	Factor of $D(c^*)$	Estimation error (%)
1.70	143	61.2	178	54.9	8.4
1.95	102	46.5	109	42.3	6.7
2.42	82	32.7	75.4	31.7	6.3
2.78	63	23.3	58	27.5	6.2

By applying (13)–(17) to (24), the entropy estimation as a function of scaling factor c becomes

$$\begin{aligned} H(c) \approx & -\text{Ent}[1-G(a)] - \sum_{k=1}^{m} C_s(k)\cdot \text{Ent}\left[\frac{(1-0.5\delta_{km})}{2^{\min\{k,n\}}}\right. \\ & \cdot (1-e^{-(L/2)\lambda})^{\max\{(n-k),0\}}(e^{(L/2)\lambda}-e^{-(L/2)\lambda})^{\min\{k,n\}} \\ & \left.\cdot e^{-\lambda c k}\right] - C_s(m)\cdot \text{Ent}\left[\frac{G(cm)}{C_s(m)}\right] \end{aligned} \quad (25)$$

where

$$G(x) = e^{-\lambda x}\sum_{k=0}^{n-1}\frac{(\lambda x)^k}{k!} \quad \text{and} \quad a = \frac{c}{2}[\text{Vol}(V_0)\Gamma(n+1)]^{1/n}.$$

E. Estimation Accuracy

In this subsection, the accuracy of the estimations is examined. We performed LVQ to subband 17 $(lllh)$ of 512 × 512 8 bit gray scale "Lena" at various scaling factors, and measured the distortion and the entropy for each scaling factor. Fig. 3 compares the real distortion and the distortion estimated from (23). Table I compares real and estimated values for minimum distortion at different LVQ bit rates, and calculates the average estimation error thus:

$$\frac{1}{T}\int_0^T \left|\frac{\text{Real } D(c) - \text{Estimated } D(c)}{\text{Real } D(c)}\right| dc, \qquad T = 5\cdot\sigma$$

where σ is the standard deviation of source data.

The error checking range $c = [0, T]_{T=5\cdot\sigma}$ was chosen because more than 97% of source vectors lie within that range. At more than 2 bits/sample, the estimations are fairly accurate for practical use. Furthermore, the overall accuracy of distortion estimation presented in this work is superior to any other reported work [13].

In Fig. 4, we also compare the real entropy and the entropy estimated from (25), for various LVQ bit rates at fixed input vector size ($N = 4096$). In Fig. 5, we do the same for various sizes of input vectors and fixed bit rate (LVQ rate = 2.1

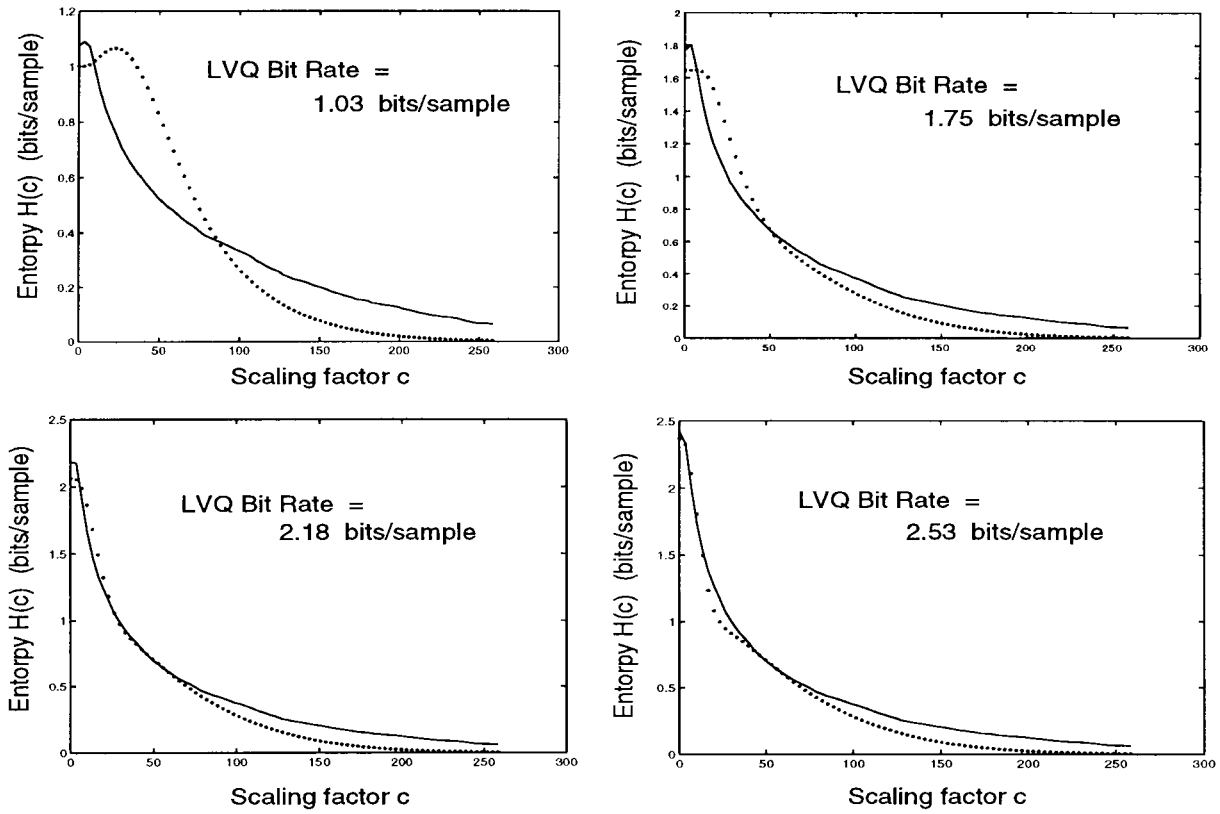


Fig. 4. Real entropy (solid line) and estimated entropy (dotted line) for subband 17 $(lllh)$ of 512 $\times$ 512 "Lena." Lattice type is D_4. The vector size is $N = 4096$.

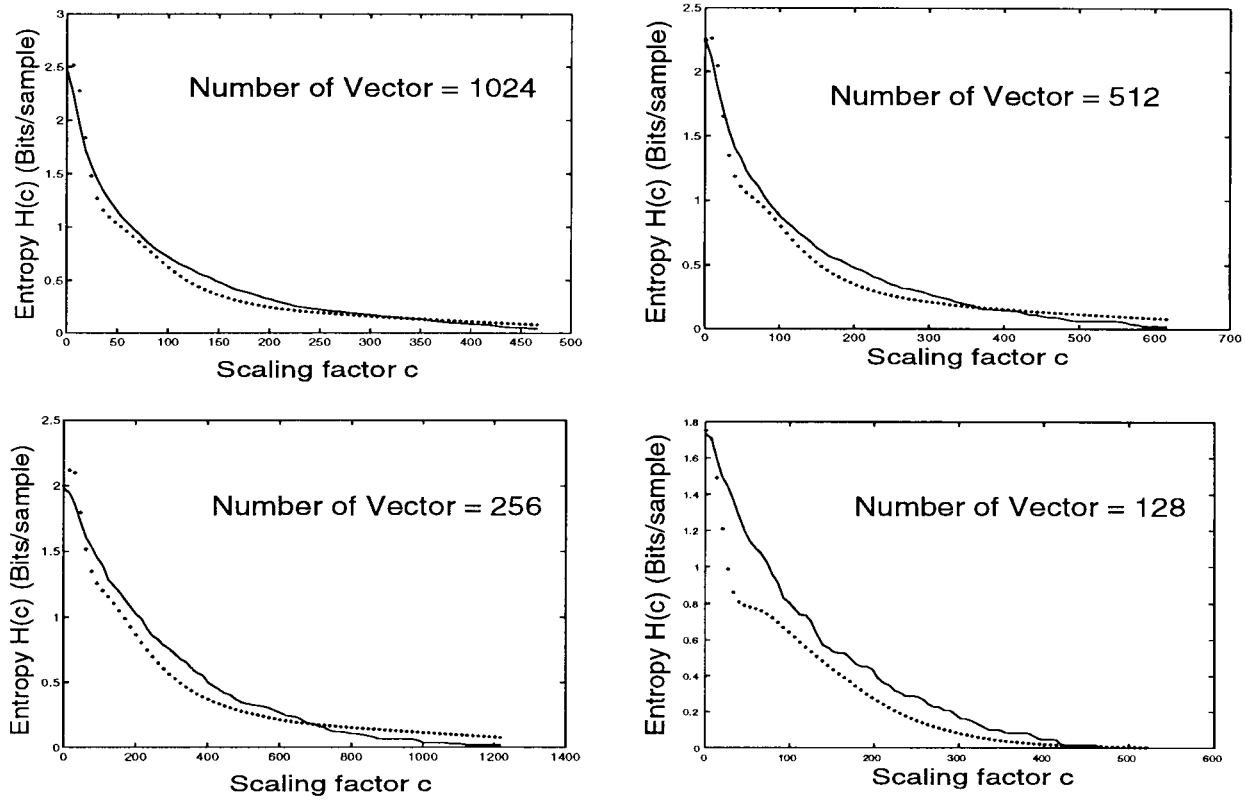


Fig. 5. Real entropy (solid line) and estimated entropy (dotted line) for various input vector size. From clockwise, subband 131, 260, 523, and 1044 of 512 $\times$ 512 "Lena." The LVQ bit rate $= 2.18$ bits/sample.

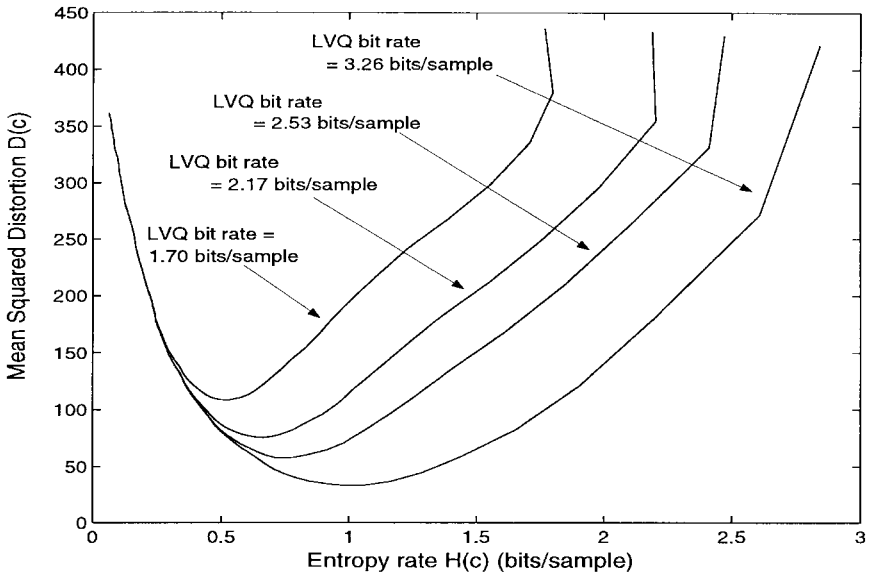


Fig. 6. Distortion versus Entropy curves for various LVQ bit rates. The test data is subband 17 $(lllh)$ of 512 × 512 "Lena."

TABLE II
AVERAGE ENTROPY ESTIMATION ERROR (%) FOR 512 × 512 "LENA." D_4 LATTICE

LVQ bit rate (bits/sample)	Entropy estimation error (%)
1.70	11.0
1.95	7.6
2.42	6.2
2.78	6.1

(a)

Subband Number	Entropy approximation error (%)
1044 (N=64)	9.3
523 (N=128)	7.5
260 (N=256)	6.9
131 (N=512)	6.4

(b)

bits/sample). Table II contains the average estimation error for various bit rates and input vector sizes. For the LVQ bit rate of more than 2 bits/sample, the average entropy estimation error is about 10%, which is acceptable for most cases. Since the entropy estimation has not been derived previously, it is not possible to compare our entropy estimation to other work.

IV. DESIGNING ENTROPY-CONSTRAINED LVQ FOR SUBBAND CODING

While subband signals are independently coded, the coding source (entropy) should be allocated to each subband in accordance with the information distribution among subbands. In this section, an operational algorithm to design optimal entropy constrained LVQ's for an SBC is developed from distortion and entropy estimates of (23) and (25).

Fig. 6 shows a plot of distortion versus entropy. When the scaling factor is large (at the left of the curve in Fig. 6), the distortion is large since the granular area dominates, and the entropy is small because most vectors are mapped to only one lattice point at the origin. As the scaling factor decreases, the distortion decreases because the scaling factor offers a better tradeoff between the granular area and the overload area; and the entropy increases because the source vectors are mapped to more lattice points. When the scaling factor becomes so small that the effect of overload area dominates over that of the granular area, the distortion increases sharply. In overload scaling, most of the source vectors are mapped to the lattice points on the outermost surface enclosing the granular area, and the entropy is mainly generated by those lattice points. Since the codeword probabilities of the lattice points on the outermost surface are uniform, the amount of generated entropy is only affected by the number of those lattice points. Thus, when the scaling factor becomes so small for the overload area to dominate, the entropy will saturate even if the distortion increases sharply, as indicated by the upward region at the right side of each curve in Fig. 6. Consequently, one should not choose a scaling factor in which overload region will dominate. The admissible range for the scaling factor can be determined from the distortion estimate in such a way that $I = [c_i^0, \infty]$ where $c^0 = \arg\min D(c)$ as seen in Fig. 7(a). In the admissible range, the distortion is a monotonically decreasing function of entropy.

In the case where a Laplacian source is a good model for the subband images, the Voronoi cell probability decreases exponentially with distance from the origin; therefore, the Voronoi cell probabilities close to the origin dominate. Consequently, before the break point where the overload area begins to dominate, the dominant Voronoi cell probabilities at a given scaling factor are nearly independent of LVQ bit rate. This implies that the LVQ bit rate determines only break point locations and the entropy-distortion curves with different LVQ bit rates are equivalent before the break points,

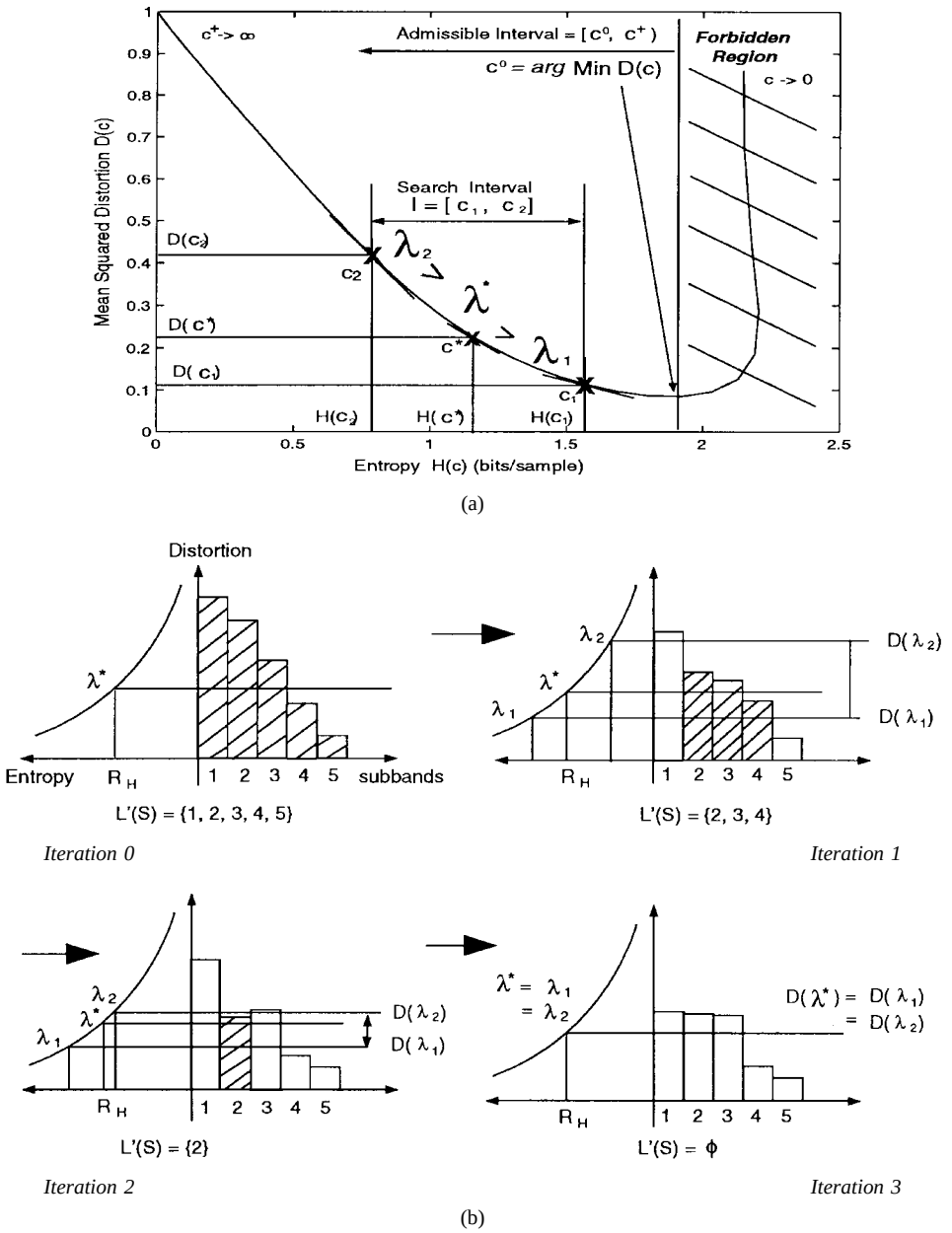


Fig. 7. (a) Search procedure for the optimal scaling factor c^* at a subband and (b) entropy allocation procedure.

as seen in Fig. 6. Based on this observation, we fix the LVQ bit rate as 3.00 bits/sample, to ensure reasonable accuracy of estimations.

The constrained problem in (3) can be reformulated as an unconstrained problem using a Lagrange multiplier approach [15]:

$$\min_{\{c_i \in I_i | i \in L(S)\}} \left\{ \sum_{i \in L(S)} \frac{D_i(c_i)}{2^{d_i}} + \lambda \sum_{i \in L(S)} \frac{H_i(c_i)}{2^{d_i}} \right\} = \sum_{i \in L(S)} \min_{c_i \in I_i} \left\{ \frac{D_i(c_i)}{2^{d_i}} + \lambda \frac{H_i(c_i)}{2^{d_i}} \right\}$$

$$\boldsymbol{I} = \{I_i = [c_i^0, \infty) | c_i^0 = \arg\min_{c_i} D_i(c),\ i \in L(S)\}. \quad (26)$$

The problem posed in (26) is different from the conventional source allocation problem solved in for the following reasons.

1) The cost $D(c)$ is not a direct function of source $H(c)$; instead, $D(c)$ and $H(c)$ are related through the scaling factor c.
2) Since $D(c)$ and $H(c)$ are continuous, there are no singular points.
3) $D(c)$ is an increasing function for c, and $H(c)$ is a decreasing function for c.

From these observations, a dynamic entropy allocation algorithm based on the Lagrange multiplier method [15] is developed. Fig. 7(a) depicts the procedure to find the optimal scaling factor for a subband. In the figure, the sweeping range of the Lagrange multiplier is $I_\lambda = [\lambda_1, \lambda_2]$ and the corresponding search range of the scaling factor is $I_c = [c_1, c_2]$. If $\{H(c, \lambda), D(c, \lambda)\}$ is the solution set of $\min\{D(c)+\lambda H(c)\}$ and $\lambda_2 > \lambda_1 > 0$, then $c_2 > c_1$, $H(c_2, \lambda_2) < H(c_1, \lambda_1)$ and $D(c_2, \lambda_2) > D(c_1, \lambda_1)$. As the sweeping range of

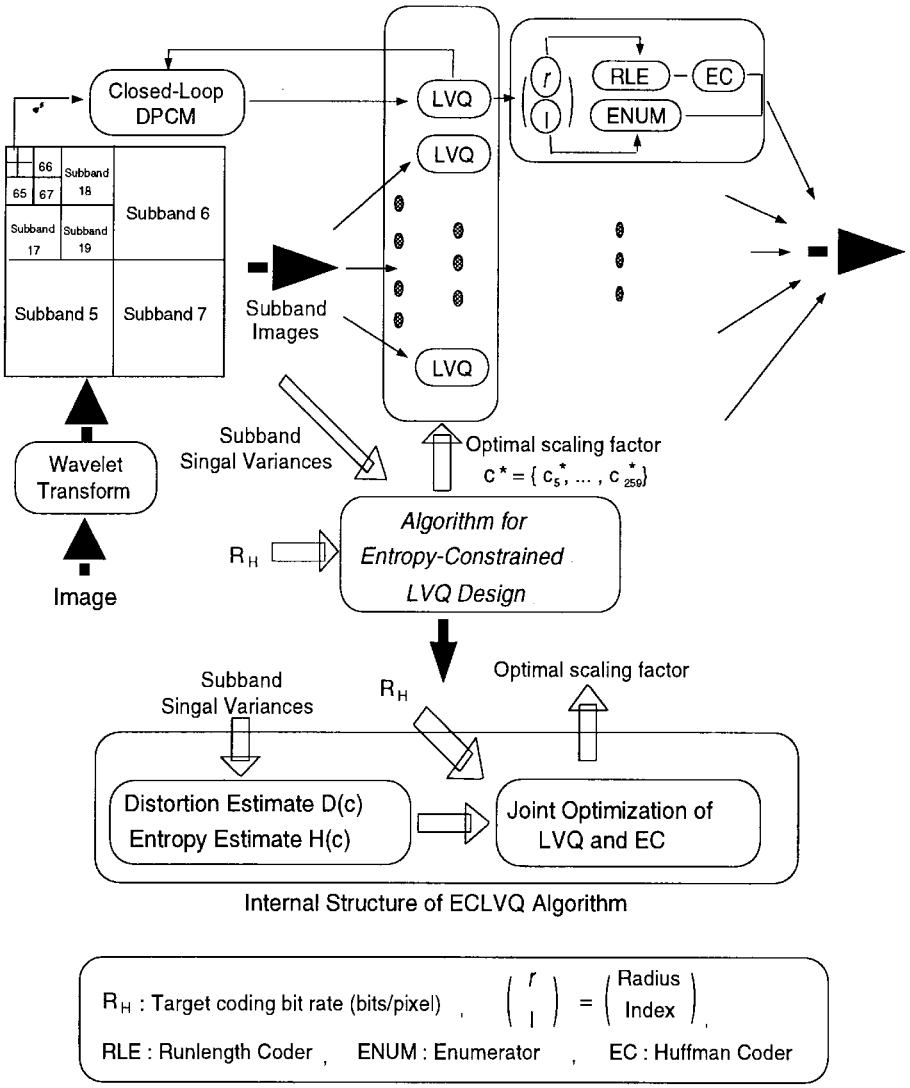


Fig. 8. Wavelet-based image coding system with entropy constrained LVQ designed by proposed algorithm.

λ decreases, the search range of the scaling factor is also dynamically reduced, and finally reaches the optimal scaling factor $c_1 = c_2 = c^*$ when $I_c = \emptyset$.

Since subband variances are different, at any particular λ, the optimal scaling factors for some subbands will already have been found, and they should be removed from the search space. Let $I_D = [D(\lambda_1), D(\lambda_2)]$ be the distortion range corresponding to $I_\lambda = [\lambda_1, \lambda_2]$; further, let $L'(S)[\subset L(S)]$ be the set of subbands whose optimal scaling factors are not yet found, that is, $L'(S) = \{i \in L(S) : I_{c_i} \neq \emptyset\}$. If the distortion of a subband is outside of I_D, the optimal scaling factor for that subband has already been found and the subband must be removed from $L'(S)$ to reduce the search space. Consequently, the distortions of the subbands belonging to $L'(S)$ must be within I_D. As λ is swept, I_D will determine $L'(S)$. Note that $L'(S)$ also decides the sweeping range of λ, and so $L'(S)$ and I_D are mutually dependent. To resolve this mutual dependency, $L(S)$ is initially set to $L(S)$ and the corresponding I_D is calculated. The subbands of which distortions are not in I_D will be removed from $L'(S)$. The new $L'(S)$ leads to new I_λ and I_D. This iteration continues until $L'(S) = \emptyset$. Fig. 7(b) illustrates the procedure for determining optimal scaling factors among subbands while optimally allocating entropy among subbands.

In order to check the accuracy of the proposed algorithm, the wavelet based image coding is performed as seen in Fig. 8. From subband variances, the algorithm estimates the optimal scaling factors for each LVQ. Table III compares the estimated values and measured values resulting from the estimated optimal scaling factor set. For example, with given entropy rate $R_H = 0.5$ bits/pixel for the "Lena" image, the estimated PSNR is 36.83 dB, while the measured PSNR and entropy rate are 36.02 dB and 0.48 bits/pixel, respectively. The entropy is measured by $H(\boldsymbol{Y}) = -\sum P(\boldsymbol{Y}_i) \log_2 P(\boldsymbol{Y}_i)$, where $P(\boldsymbol{Y}_i)$ is the codeword probability obtained from the frequency of the code $\boldsymbol{Y}_i$ over the size of codeword sequences. As shown in Table III, the estimation error is within 10% for all range of rate, which implies that the proposed algorithm controls the coder performance bound with more than 90% accuracy at all bit rates.

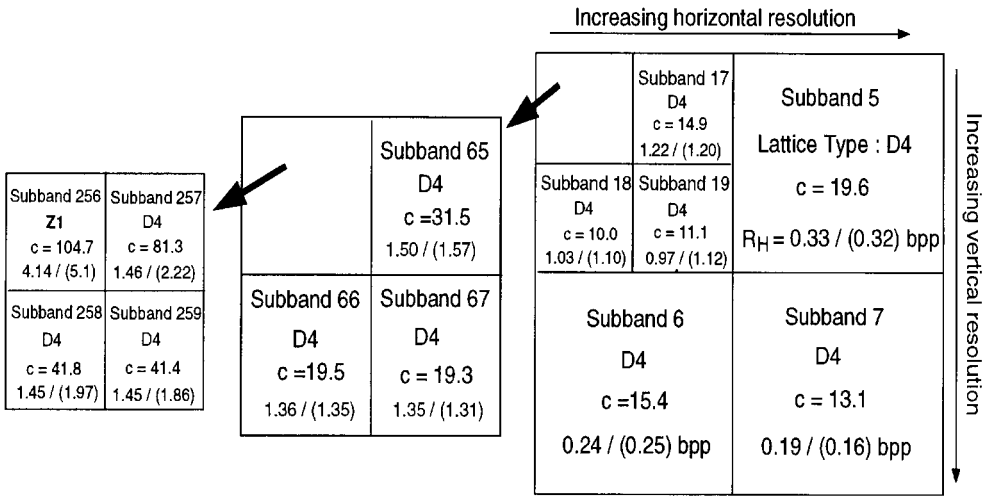
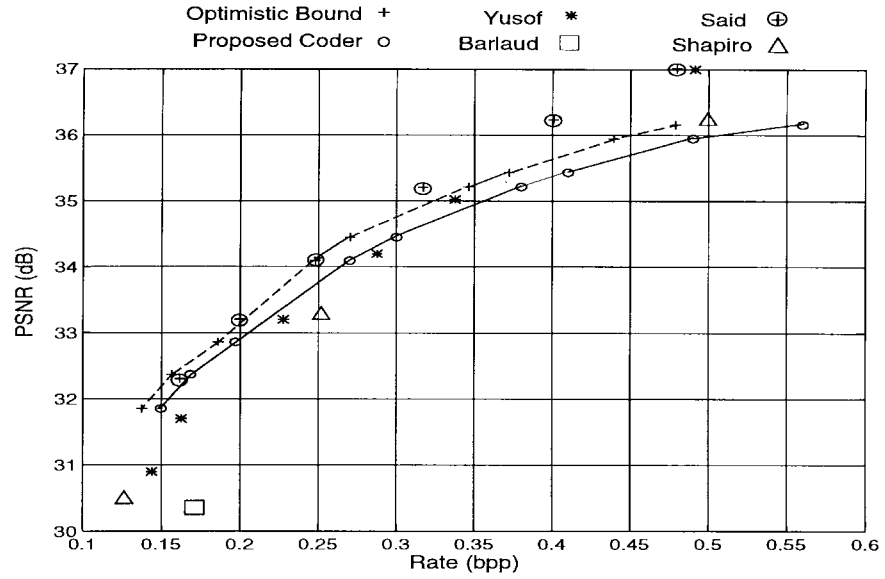
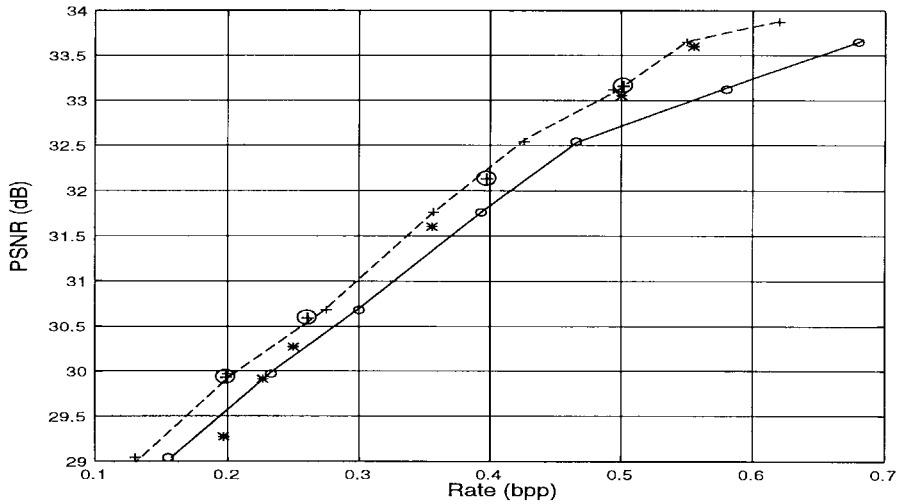


Fig. 9. 0.5 bits/pixel coding strategy for "Lena." c is the estimated optimal scaling factors at each subband. R_H is allocated entropy rate. The bit rate in parenthesis is actually obtained from binary file.



(a)



(b)

Fig. 10. Comparison of coding performances.

TABLE III
COMPARISONS OF ESTIMATED PERFORMANCE AND ACTUALLY MEASURED PERFORMANCE. R_H: ENTROPY RATE (BITS/PIXEL), PSNR: PEAK SNR (DECIBELS)

Lena (512X512X8)			
Estimated		Measured	
R_H	PSNR	R_H	PSNR
0.5	36.83	0.48	36.02
0.4	35.35	0.39	35.41
0.3	35.90	0.29	34.68
0.2	32.90	0.19	32.61

(a)

Goldhill (512X512x8)			
Estimated		Measured	
R_H	PSNR	R_H	PSNR
0.6	34.86	0.62	33.87
0.5	34.00	0.49	33.12
0.35	32.64	0.36	31.76
0.2	30.92	0.20	29.97

(b)

Fig. 11. Perceptual performances: Original "Lena" (upper left), PSNR = 34.45 dB at 0.32 bpp (upper right), PSNR = 32.37 dB at 0.18 bpp (lower left), PSNR = 29.62 dB at 0.09 bpp (lower right).

V. APPLICATION TO IMAGE CODING

The proposed algorithm is applied to a wavelet based image coder with 4-level wavelet decomposition using the (9, 7)-tap wavelet and symmetric extension [6]. Since the lowest frequency subband image is strongly correlated with the original image and does not fit a Laplacian source, a closed-loop DPCM is applied to the lowest frequency subband. Fig. 8 depicts the coding system. Fig. 9 illustrates a coding strategy at a bit rate of $R_H = 0.5$ bits/pixel. The allocated entropy and the scaling factors at each subband are decided by the proposed algorithm. The bit rates in parentheses are actually obtained from binary file. For the lowest subband, Z_1 lattice (the uniform scalar quantizer) is used, and for other subbands, D_4 lattice is used.

As explained in Section III, the codeword sequences produced by each LVQ are in coordinate form, that is, (radius, position index) $\Longrightarrow$ (r, l). Runlength coding and Huffman coding are applied to radius r. Index l is enumerated with the minimum bit accuracy required to cover the index range. For example, at $r = 2$ in the D_4 lattice, there are 64 lattice points [6], which means that the index range goes from 0 to 63 and so $6 = \lceil \log_2 63 \rceil$ bits are required to enumerate the indices at $r = 2$, where $\lceil \cdot \rceil$ is the ceiling function. Since index l is 0 if radius r is 0, the runlength code already contains the information of zero indices and they are not enumerated. The enumerating bits linearly increase with the radius that grows exponentially with vector dimension and LVQ bit rate [6]. To prevent the need for too many enumerating bits, large indices are adaptively shifted down, and the shifting information is carried on radii of those indices. The enumerating bits are adjusted to 4–8 bits [16]–[18]. Although the shift process increases the entropy of the radius sequence, it decreases the number of enumeration bits and saves more than 0.7 bits/pixel in a subband whose allocated entropy rate is more than 1.5 bits/pixel.

Fig. 10 compares the performances of up-to-date coding methods to that of the proposed coder. The PSNR versus coding rate plots are obtained from luminance of 8 bpp 512 $\times$ 512 "Lena" and "Goldhill." Test images are available at the University of Waterloo image archives [19]. Coding results of the proposed coder are calculated from the actual sizes of binary files, not from entropy measurement. Figs. 11 and 12 illustrate the perceptual performances of the proposed coder.

The optimistic bound is obtained from entropy measurements, as in Table III. The actual performance of the proposed coder is close to the optimistic bound at low rates, but deviates more from that bound as the bit rate increases. This phenomenon is due to the increase of enumerating bits, as explained above. Thus, subband coders with lattice vector quantizers are no better than subband coders with uniform scalar quantizer at high rates (more than 0.4 bits/pixel in the D_4 lattice). Even if the shifting process saves the redundant bits, it is still necessary to make a vector entropy coder work closely to the entropy bound at all ranges of coding rate.

The numerical performance of the proposed coder surpasses that of most subband coding methods previously reported in the literature, particularly at bit rates less than 0.25 bits/pixel. When compared to the recently reported subband coder using entropy-constrained LVQ [13], the performance of the proposed coder is better than the coder in at bit rates lower than 0.25 bits/pixel, as seen in Fig. 10. The competitive performance of the proposed coder at low bit rates is a

Fig. 12. Perceptual performances: Original "Goldhill" (upper left), PSNR $= 31.73$ dB at 0.41 bpp (upper right), PSNR $= 29.94$ dB at 0.24 bpp (lower left), PSNR $= 27.42$ dB at 0.1 bpp (lower right).

benefit of the joint optimization of LVQ and entropy coding. However, the performance of the proposed coder is 0.5–0.8 dB lower than the progressive subband coder reported by Said [20]. Since the coder in [20] exploits pixel-to-pixel correlations across different subbands while a classical subband coder takes the average correlation of each subband, the progressive subband coder should be better than a classical subband coder. We expect that the application of our entroy-constrained Lattice Vector Quantizer (ECLVQ) to the zero-tree based subband coders [20], [21] will result in further notable improvement.

VI. Conclusion

The algorithm presented in this paper achieves the joint optimization of an LVQ and an entropy coding in a subband coding. Estimation formulas for distortion and entropy for a lattice vector quantizer encoding a Laplacian source were derived as functions of the scaling factor. The optimization algorithm based on these estimation formulas quickly calculates highly accurate scaling factors for entropy-constrained LVQ's in a subband coding. A novel runlength coder for an LVQ was also developed in order to save the redundant bits in enumerating indices of LVQ code words. The wavelet-based image coder with entropy-constrained LVQ's designed by the proposed algorithm is competitive with previously reported subband coders with entropy-constrained LVQ at bit rates less than 0.3 bits/pixel.

References

[1] T. Senoo and B. G. Girod, "Vector quantization for entropy coding of image subbands," *IEEE Trans. Image Processing*, vol. 1, pp. 526–533, Oct. 1992.

[2] T. R. Fischer, "A pyramid vector quantizer," *IEEE Trans. Inform. Theory*, vol. 32, pp. 568–583, July 1986.

[3] Z. M. Yusof and T. R. Fischer, "An entropy-coded lattice vector quantizer for transform and subband image coding," *IEEE Trans. Image Processing*, vol. 5, pp. 289–298, Feb. 1996.

[4] J. H. Conway and N. J. Sloane, "Voronoi regions of lattices, second moments of polytopes and quantization," *IEEE Trans. Inform. Theory*, vol. 28, pp. 211–226, Mar. 1982.

[5] K. Sayood, J. D. Gibson, and M. C. Rost, "An algorithm for uniform vector quantizer design," *IEEE Trans Inform. Theory*, vol. 30, pp. 805–814, Nov. 1984.

[6] M. Barlaud, P. Solé, M. Antonini, and P. Mathieu, "Pyramidal lattice vector quantization for multiscale image coding," *IEEE Trans. Image Processing*, vol. 3, pp. 367–380, July 1994.

[7] Y. H. Kim and J. W. Modestino, "Adaptive entropy coded subband coding of images," *IEEE Trans. Image Processing*, vol. 1, no. 1, pp. 31–48, Jan. 1992.

[8] J. Gibson and K. Sayood, "Lattice quantization," *Advances Electron. Electron Phys.*, vol. 72, pp. 259–330, June 1988.

[9] P. F. Swaszek, "A vector quantizer for the Laplacian source," *IEEE Trans. Inform. Theory*, vol. 37, pp. 1355–1365, Sept. 1991.

[10] K. A. Birney and T. R. Fischer, "On the modeling of DCT and subband image data for compression," *IEEE Trans. Image Processing*, vol. 4, pp. 186–193, Feb. 1995.

[11] I. S. Grandshteyn and I. M. Ryzhik, *Table of Integrals, Series, and Products.* New York: Academic, 1980.

[12] H. J. Larson, *Introduction to Probability Theory and Statistical Inference*, 3rd ed. New York: Wiley, 1982.

[13] D. G. Jeong and J. D. Gibson, "Uniform and piecewise uniform lattice vector quantization for memoryless Gaussian Laplacian sources," *IEEE Trans. Inform. Theory*, vol. 39, pp. 786–804, May 1993.

[14] H. J. Thomas and R. L. Finney, *Calculus and Analytic Geometry*, 9th ed. Reading, MA: Addison-Wesley, 1992.

[15] 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.

[16] G. K. Wallace, "The JPEG still picture compression standard," *Communi. ACM*, vol. 34, no. 4, pp. 30–44, Apr. 1991.

[17] W. B. Pennebaker and J. L. Mitchell, *JPEG Still Image Data Compression Standard.* New York: Van Nostrand Reinhold, 1993.

[18] W. H. Kim and Y. K. Hong, "Adaptive runlength coder for lattice vector quantization," in preparation.

[19] Univ. Waterloo image archives, http://links.uwaterloo.ca/bragzone.base.html.

[20] A. Said and W. A. Pearlman, "A new fast and efficient image codec based on set partitioning in hierachical trees," *IEEE Trans. Circuits Syst. Video Technol.* vol. 6, pp. 243–250, June 1996.

[21] J. M. Shapiro, "Embedded image coding using zerotrees of wavelets coefficients," *IEEE Trans. Signal Processing*, vol. 41, pp. 3445–3462, Dec. 1993.

Won-Ha Kim (S'90–M'97) received the B.S. degree in electronic engineering from Yonsei University, Seoul, Korea. He received the M.S. and Ph.D. degrees in electrical engineering from the University of Wisconsin-Madison in 1988 and 1997, respectively.

He worked for Motorola Software Enterprise at Schamburg, IL from January 1996 to July 1996, as a Technical Consultant. He is now with Los Alamos National Laboratory, Los Alamos, NM. His research interests are source coding and error correct coding for multimedia applications.

Yu-Hen Hu received the B.S.E.E. degree from National Taiwan University, Taipei, Taiwan, R.O.C., in 1976. He received the M.S.E.E. and Ph.D. degrees in electrical engineering from the University of Southern California, Los Angeles, CA, in 1980 and 1982, respectively. From 1983 to 1987, he was an Assistant Professor of the Electrical Engineering Department of Southern Methodist University, Dallas, TX. He joined the Department of Electrical and Computer Engineering, University of Wisconsin, Madison, WI, in 1987 as an Assistant Professor (1987–1989) and is currently an Associate Professor. His research interests include multimedia signal processing, artificial neural networks, fast algorithms, and design methodology for application-specific micro-architectures, as well as computer-aided design tools for VLSI using artificial intelligence. He has published more than 150 journal and conference papers in these areas.

He is a former Associate Editor (1988–1990) for the IEEE TRANSACTION ON ACOUSTIC, SPEECH, AND SIGNAL PROCESSING in the areas of system identification and fast algorithms. He is currently Associate Editor of *Journal of VLSI Signal Processing*. He is a founding member of the neural network signal processing technical committee of IEEE Signal Processing Society and served as chair from 1993 to 1996. He is a former member of VLSI signal processing technical committee of the Signal Processing Society. Currently he is serving as the secretary of the IEEE Signal Processing Society.

Truong Q. Nguyen (S'85–M'90–SM'95), for a biography, see this issue, p. 931.