A Novel Fingerprint Image Compression Technique Using Wavelets Packets and Pyramid Lattice Vector Quantization

Shohreh Kasaei, Mohamed Deriche, *Senior Member, IEEE,* and Boualem Boashash, *Fellow, IEEE*

***Abstract*—A novel compression algorithm for fingerprint images is introduced. Using *wavelet packets* and *lattice vector quantization*, a new vector quantization scheme based on an accurate model for the distribution of the wavelet coefficients is presented. The model is based on the generalized Gaussian distribution.**

We also discuss a new method for determining the *largest radius* of the lattice used and its *scaling factor*, for both uniform and piecewise-uniform pyramidal lattices. The proposed algorithms aim at achieving the best rate-distortion function by adapting to the characteristics of the subimages. In the proposed optimization algorithm, no assumptions about the lattice parameters are made, and no training and multi-quantizing are required. We also show that the wedge region problem encountered with sharply distributed random sources is resolved in the proposed algorithm.

The proposed algorithms adapt to variability in input images and to specified bit rates. Compared to other available image compression algorithms, the proposed algorithms result in higher quality reconstructed images for identical bit rates.

***Index Terms*—Compression, fingerprints, generalized Gaussian distribution, pyramid lattice vector quantization, wavelet packets.**

I. Introduction

THE fundamental goal of image compression is to obtain the best possible quality, for a given storage or communication capacity. One of the main applications where compression is crucial is in fingerprint analysis for forensic applications. The increasing amount of fingerprints collected by law enforcement agencies has created an enormous problem in storage and transmission. Fingerprints are digitized at a resolution of 500 pixels/inch with 256 grey-levels.[1] Although there are many image compression techniques currently available, there still exists a need to develop faster and more robust algorithms adapted to fingerprints.

Manuscript received May 23, 1997; revised November 11, 1999. This work was supported by Queensland University of Technology, King Fahd University, and the Australian Research Council. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Christine Podilchuk.

S. Kasaei is with Sharif University of Technology, Iran (e-mail: skasaei@sharif.edu).

M. Deriche is with King Fahd University, KSA, on leave from the Signal Processing Research Centre at Queensland University of Technology, Australia (e-mail: m.deriche@qut.edu.au).

B. Boashash is with the Signal Processing Research Centre at Queensland University of Technology, Australia.

Digital Object Identifier 10.1109/TIP.2002.802534

[1]Grey-level images have a more *natural* appearance to human viewers than do black/white images and allow a higher level of subjective discrimination by fingerprint examiners (who make the final decision in identifying fingerprints).

One of the main difficulties in developing compression algorithms for fingerprints resides in the need for preserving the *minutiae* (ridges endings and bifurcations) which are subsequently used in identification. To achieve high compression ratios while retaining these fine details, wavelet packets are considered in this work. To save on computation, transmission, and storage costs, a *fixed* decomposition structure, tailored to fingerprint images, is designed.

To obtain high compression ratios, we follow the above transformation with vector quantization. In this work, we chose to use *lattice vector quantization* (LVQ) given its superior performance over other types of vector quantizers; such as the LBG [26].

To apply LVQ, one needs to determine the *truncation level* and *scaling factor* of the lattice. By truncation, the largest radius of the lattice, r, is chosen determining the codebook size, $|c|$. By scaling, the density of lattice points is changed to minimize the *expected total distortion* (ETD) caused by quantization. In previous lattice-based compression algorithms, the lattice parameters are commonly predetermined, leading to nonoptimized quantization schemes. In this paper, a new approach for estimating these parameters is proposed. The proposed quantizers aim at minimizing the quantization distortion by considering the characteristics of each subimage data. The design is based on a *precise* model for the distribution function of the wavelet coefficients and results in a superior performance compared to other available algorithms [2], [19], [29]–[32], [21], [24].

Since LVQ is a multidimensional generalization of uniform quantization, it produces minimum distortion for inputs with *uniform* distributions. In order to take advantage of the properties of LVQ and its fast implementation while considering the i.i.d. nonuniform distribution of the wavelet coefficients, we use a *piecewise-uniform pyramid LVQ* (PU-PLVQ) algorithm. The proposed algorithm results in the quantization of most source vectors without the need to project these on the lattice outermost shell, while properly maintaining a small codebook size. It also resolves the *wedge* region problem caused by pyramidal lattice point shells. The above represent some of the drawbacks of the algorithm proposed by Barlaud [12]. The proposed algorithm handles all types of lattices, not only the cubic lattices, as opposed to the algorithms developed by Fischer [14], [35] and Jeong [17]. Moreover, no training and multi-quantizing (to determine the lattice parameters) are required, as opposed to Powell's algorithm [28].

The rest of the paper is organized as follows. The wavelet transform and wavelet packets are briefly introduced in

Section II. Subband image coding, statistical properties of subbands, and distribution of the wavelet coefficients for different subbands are described in Section III. LVQ, lattice truncation and scaling, and previous work on pyramid LVQ are presented in Section IV. In Section V, the proposed PLVQ algorithms, design of lattice parameters for different subbands, and the lossless compression algorithms used for certain subimages are presented. Experimental results are discussed in Section VI, followed by a conclusion in Section VII.

II. Brief Introduction to the Wavelet Transform and Wavelet Packets

Fingerprint images can be seen as texture patterns of flow orientations with sharp discontinuities. Given this particular nature of fingerprint images and the need for retaining ridge fine details and their inter-relation, the wavelet decomposition is considered here. Some advantages of using the wavelet transform include: providing a multiscale representation of a given image, flexibility in time-frequency tiling, attractive properties in extracting features from nonstationary signals, good energy compaction property, ability to match human vision spectral properties, low computational load $(O(N))$, possibility of using short filters (while reducing the Gibbs ringing effects), absence of block artifacts, and providing a sparser representation than the discrete cosine transform (higher lossy compression gains at low bit rates).

In addition to the above, wavelet packets provide a more flexible decomposition by allowing decompositions at any node of the decomposition tree. This can result in a structure that matches the characteristics of specific data of interest. To obtain the best decomposition structure, in the *entropy-based best basis selection* (EBBBS) algorithm [4], a library of orthonormal basis is designed providing a projection with the lowest information cost. Using the Shannon entropy as the cost function, for a given image (or family of images), a decomposition structure which results in the minimum entropy is obtained using the EBBBS algorithm.

Extensive work on power spectral density estimates from numerous fingerprints has shown that natural frequencies of ridges in fingerprint images are in the portion of spectrum contained roughly in the range of $\pi/8$–$\pi/4$ (corresponding to the third wavelet decomposition level). In this work, we opted to use the fixed 73-Subband decomposition which we developed earlier [22], as opposed to the FBI's 64-Subband [2]. This structure was chosen based on experiments carried on the distribution of energies over different subbands and their effect on the quality of the reconstructed images and by analyzing the best trees obtained for a variety of fingerprints, taking into account the spectral sensitivities in the human vision system.[2] Typical decomposition structures, examined in this work, are depicted in Fig. 1.

As shown in Fig. 1, for the 73-Subband, we do not apply further decomposition to the diagonal subimage of the first wavelet level ($1d$), as opposed to the 64-Subband. This is due to the low energy content, low variance, and low effect of this subimage ($1d$) on the reconstructed image. Note that, this also reduces the computational load. Furthermore, given the importance of the second wavelet level, we apply further decompositions at this level. As a result, the 73-Subband is computationally less expensive than the 64-Subband while it outperforms the 64-Subband as shown in Section VI. The 73-Subband decomposition is *prefixed* and is found to provide comparable results to those obtained using the EBBBS method, which requires an extremely high computational power (see Section VI).

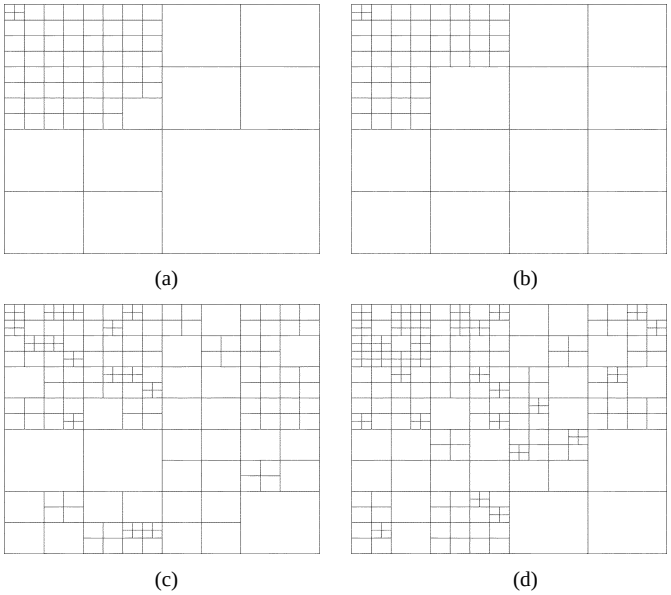


Fig. 1. Five-level decomposition structures: (a) proposed 73-Subband; (b) WSQ's 64-subband; (c) EBBBS using Johnston-32D filter; and (d) EBBBS.

III. Subband Image Coding

Data compression is the process of reducing the amount of bit rate required to represent a given set of data. Compression can be applied using lossless and/or lossy techniques. In lossless compression, there is no loss of information and data is totally preserved, however the coding gain can not be very high. On the other hand, in lossy compression, part of information is lost but a higher coding gain can be achieved.

After transforming the data, one needs to decide on the quantization scheme to be used. One way to improve quantization performance is to adjust the quantizer to the characteristics of different subbands in the case of subband coding. To do so, the quantizer should be tailored to the *probability density function*s (PDF) of the each of the subbands. Consequently, in order to have an efficient compression algorithm, we firstly need to investigate the statistical properties of the different subbands.

A. Statistical Properties of Subbands

Table I displays the results obtained from the statistical analysis of different subimages of a typical fingerprint image. The table shows the statistics of the wavelet packet coefficients for different orientations (subbands). For each orientation, it gives the minima, maxima, mean (μ), standard deviation (σ), and the mean-removed normalized values of the minima and maxima. Here, the Johnston-32D filter [18] is used.

[2]It is shown that the human vision is more sensitive to contrasts at low frequencies while details at high resolution and diagonal directions are less visible to human eye [3], [33].

TABLE I
STATISTICS OF WAVELET PACKET COEFFICIENTS FOR DIFFERENT SUBBANDS, FOR A MEAN-REMOVED FINGERPRINT IMAGE [LD: LEVEL-DIRECTION, μ: MEAN, σ: STANDARD DEVIATION, d: DIAGONAL, v: VERTICAL, h: HORIZONTAL, AND ℓ: LOWPASSED]

LD	Min	Max	μ	σ	$\frac{Min-\mu}{\sigma}$	$\frac{Max-\mu}{\sigma}$
$1d$	-13.67	15.13	-0.01	2.42	-5.65	6.26
$1v$	-52.27	62.30	-0.05	5.73	-9.11	10.88
$1h$	-82.58	58.07	-0.02	6.75	-12.23	8.60
$2d$	-118.30	122.44	-0.16	12.17	-9.71	10.07
$2v$	-282.33	295.59	-0.06	25.30	-11.16	11.69
$2h$	-204.39	253.02	-0.09	24.44	-8.36	10.35
$3d$	-500.41	507.91	-0.18	50.63	-9.88	10.03
$3v$	-435.21	413.29	0.67	70.44	-6.19	5.86
$3h$	-517.80	640.32	0.65	75.79	-6.84	8.44
$4d$	-248.10	286.52	0.31	58.06	-4.28	4.93
$4v$	-352.67	376.88	-2.71	76.47	-4.58	4.96
$4h$	-608.78	716.44	5.51	113.96	-5.39	6.24
4ℓ	-1483.50	3707.70	1.39	667.21	-2.23	5.55

From Table I, we notice the following:

- the range of the coefficients (Max–Min) and their variances grow as the resolution becomes coarser (which shows the importance of higher level subimages);
- after obtaining the standardized subimages, the range for the resulting coefficients becomes smaller and comparable in all orientations at each level (which enables us to quantize different subbands with similar schemes).

Based on the above observations and extensive experiments on a variety of fingerprint images, we chose to use

- lossless predictive coding for the 4ℓ and $4h$ subbands;
- lossy zero-memory compression algorithms for remaining subbands.

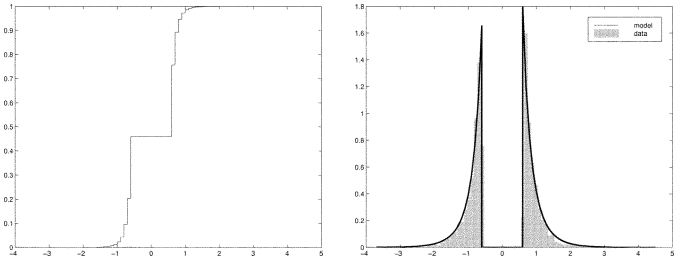
B. Shape of Distribution

The shape of the distribution of the wavelet coefficients for each subimage is important in order to understand the behavior of the coefficients and properly characterize these at different resolutions. In [1], [13], it was shown that the PDF of the wavelet coefficients can be closely approximated by the *generalized Gaussian distribution* (GGD). This GGD law is defined as:

$$f_{j,d}(x) = a_{j,d}\exp(-|b_{j,d}x|^{\eta_{j,d}})$$

$$a_{j,d} = \frac{b_{j,d}\eta_{j,d}}{2\Gamma\left(\frac{1}{\eta_{j,d}}\right)}, \quad b_{j,d} = \frac{1}{\sigma_{j,d}}\frac{\Gamma\left(\frac{3}{\eta_{j,d}}\right)^{1/2}}{\Gamma\left(\frac{1}{\eta_{j,d}}\right)^{1/2}} \qquad (1)$$

where $\eta_{j,d}$ is a *shape* parameter describing the exponential rate of decay, $\sigma_{j,d}$ is the standard deviation of the subimage at resolution level j and direction d, and $\Gamma(\cdot)$ is the usual Gamma function. The GGD with $\eta_{j,d} = 1$ coincides with the Laplacian PDF while, for $\eta_{j,d} = 2$ it becomes a Gaussian PDF. To match the real PDF, $\eta_{j,d}$ is usually computed using the χ^2 test. As the χ^2 test is computationally expensive, we developed here a nonlinear regression procedure for estimating the parameter $\eta_{j,d}$ using



(a) (b)

Fig. 2. (a) Empirical cumulative distribution function of a subimage (with rounded and discarded data) and (b) histogram of the subimage and its model.

$$\left(\frac{\text{mean deviation}}{\text{standard deviation}}\right)^2 = \frac{E[|x|]^2}{E[x^2]} = \frac{\Gamma(2/\eta)^2}{\Gamma(1/\eta)\Gamma(3/\eta)} = g(\eta)$$

$$\eta = g^{-1}\left(\frac{E[|x|]^2}{E[x^2]}\right). \qquad (2)$$

In our previous work [10], [24], we modeled the wavelet coefficients, with discarded zero elements, using the generalized Gaussian described above. In this work, a more accurate PDF model is obtained based on the GGD law while *considering* the *discarded* coefficients, as shown in Fig. 2. Note that the discarded (zero-probable) coefficients result in a flat segment in the *cumulative distribution function* (CDF).

Using the model above, we found that a more accurate bit allocation scheme can be designed. Furthermore, considering this model, no bit budget needs to be allocated for the "zero-probable" values of the data, which effectively improves the design of the lattice. In fact, based on this model, a more dedicated quantization scheme is developed which effectively reduces the quantization distortion for a given bit rate.

IV. LATTICE VECTOR QUANTIZATION

Let $\boldsymbol{x}$ be an n-dimensional source random vector with an arbitrary joint probability density function $f_X(\boldsymbol{x}) = f_X(x_1, \ldots, x_n)$. Using an n-D vector quantizer with output points $\{\boldsymbol{y}_1, \boldsymbol{y}_2, \ldots, \boldsymbol{y}_L\} \in \mathbb{R}^n$, the input point $\boldsymbol{x} \in \mathbb{R}^n$ gets mapped into a *closest* output point $\boldsymbol{y}_i$. The optimal encoder, selects the codeword $\boldsymbol{y}_i$ (for a given source vector $\boldsymbol{x}$) if: $d(\boldsymbol{x}, \boldsymbol{y}_i) \le d(\boldsymbol{x}, \boldsymbol{y}_j), \quad \forall j$ (where d denotes the Euclidean norm L^2) therefore: $Q(\boldsymbol{x}) = \boldsymbol{y}_i$. The optimal encoder, thus, operates on a nearest-neighbor or minimum-distortion basis.

A lattice Λ_n in n-D space $\mathbb{R}^n$ is composed of all integer combinations of a set of linearly independent vectors that span the space: $\Lambda_n = \{\boldsymbol{y} \in \mathbb{R}^m \,|\boldsymbol{y} = u_1\boldsymbol{a}_1 + \cdots + u_n\boldsymbol{a}_n\}$, where $\{\boldsymbol{a}_1, \ldots, \boldsymbol{a}_n\}$ are linearly independent vectors in m-D real Euclidean space $\mathbb{R}^m$ with $m \ge n$, and $\{u_1, \ldots, u_n\}$ are arbitrary integers.

Around each lattice point, the region consisting of all points of the underlying space which are closer to that point than any other point is the so-called *Voronoi* region. For each cell of the lattice, the reproduction vector is taken as the lattice vector lying in that cell. For each particular lattice, the Voronoi cells have

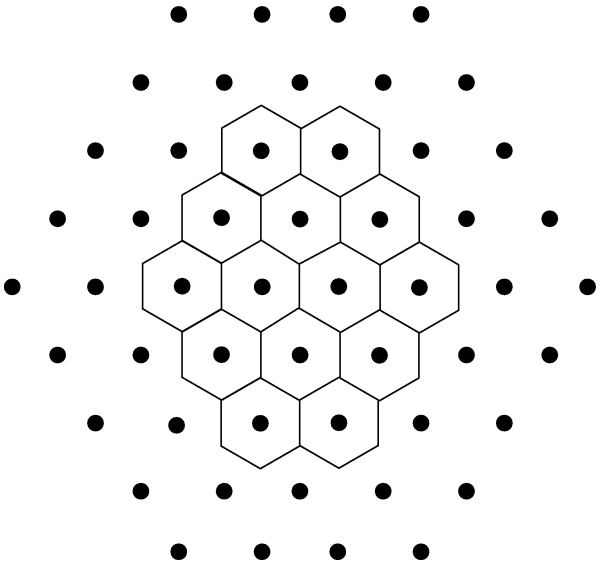


Fig. 3. Portion of the 2-D hexagonal lattice A_2.

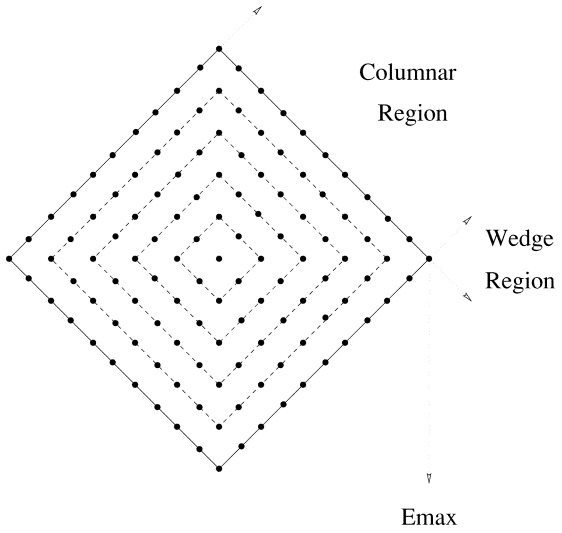


Fig. 4. Pyramid lattice vector quantizer in 2-D.

identical shape and size, regardless of the input to the quantizer. Given the regular structure of lattice codebooks, training as well as designing and transmitting the codebook is not necessary, as opposed to other quantizers such as the LBG. The best known lattices for several dimensions, as well as fast quantizing and indexing algorithms were developed by Conway and Sloane in [5], [6]. Some of the most frequently used lattices are the *root* lattices $A_n(n \geq 1)$, $D_n(n \geq 2)$, $E_n(n = 6, 7, 8)$, the Coxeter–Toble K_{12}, the Barrnes–Wall Λ_{16} and the Leech Λ_{24}. Fig. 3 illustrates a portion of the 2-D hexagonal lattice A_2 and the hexagonal Voronoi regions around some of the lattice points.

A. Lattice Truncation and Scaling

The two critically important issues with LVQ are truncation and scaling of the lattice. A truncation region is the subset of the lattice that will actually be encoded. By truncation, the largest radius of the lattice, r, is chosen to determine the number of concentric shells of lattice points within the lattice volume. Consequently, by truncation we determine the codebook size $|c|$.

Since for a particular lattice, the Voronoi cells have identical shape and size, it is also desirable to scale the lattice. By scaling, the density of the lattice points can be changed, and consequently the ETD caused by quantizing source vectors, can be minimized. For a scaled lattice, a scaling factor $s < 1$, squeezes the lattice, increases the density of lattice points, and therefore reduces the quantization distortion. However, the volume enclosed by the truncated surface and the probability that a source vector falls within that region decreases as well. An n-D vector falls into a truncated (by r-radius) and scaled (by s) lattice, if its L^1 norm is less than a predetermined value of the maximum energy; $E_{\max}$

$$\sum_{i=1}^{n} |x_i| \leq E_{\max} = sr. \tag{3}$$

The value of $E_{\max}$ is usually determined experimentally and is prefixed. In the proposed algorithm, the $E_{\max}$ is automatically determined within the algorithm.

As the wavelet transform coefficients have i.i.d. generalized Gaussian source vectors, equal probability surfaces will have a pyramidal shape [14], [17]. Fig. 4 illustrates PLVQ in 2-D. Pyramidal LVQs are far less dense than spherical LVQs, for sufficiently large radii, hence results in smaller pyramidal codebooks for equivalent radii. However, with pyramidal lattice point shells, wedge region problem arises when input vectors falling into each wedge region are all projected to lattice points with at least one coordinate being zero. These vectors will then be reconstructed with 1 to $n - 1$ degrees of freedom. Additionally, as n increases the number of different shapes of the wedge regions increases too, and therefore the portion of total space contained in these regions increases. As a result, the total distortion due to the quantizing vectors falling in these wedge regions also increases. Since the wedge regions contain high-energy edge information, the wedge problem can have an obvious distortion effect on the reconstructed image.

B. Previous Work Using PLVQ

In 1986, Fischer investigated Laplacian sources and the corresponding pyramidal surfaces of equal probability [14]. Due to the nonexistence of explicit relationship between the radius of pyramid and the codebook size,[3] he limited his work to cubic lattices. In his approach, only the vectors on the outermost shell are taken as codewords. His quantizer provided good results (in MSE sense) for input sequences with high-dimensions. For lower dimension quantizers, he suggested concentric lattice point shells.

Still with the lack of an explicit ν_Λ function, Jeong in [17] considered cubic lattices only. His algorithm uses constant PDFs inside each Voronoi cell and inside each concentric lattice. Using several-density lattice regions, his algorithm uses experimentally determined values for the two main parameters (scale factor, c_0, and scale factor ratio, s), using *Monte Carlo* simulations.

[3]The relationship between the radius of pyramidal volume and the codebook size is known as ν function which was introduced by Barlaud in [12].

One of the best image compression techniques using PLVQ was proposed by Barlaud *et al.* in 1994 [12]. However, their technique had also several drawbacks. The lattice truncation energy, m, and the maximum energy considered for the source, $E_{\max}$, were determined experimentally. Also, the entropy measure used, was not achievable because the codebook size can be orders of magnitude greater than the number of quantizer source vectors [9], [28]. Furthermore, their algorithm does not consider wedge region problem encountered with pyramidal shells.

To overcome these difficulties, a concentric double-density PLVQ was discussed by Powell in [28] in which several assumptions about the variables were made. In his work, the ratio of the scaling factors of the concentric lattices, the ratio of the distortions in each of the concentric lattices, and the maximum lattice height (radius) of one of the lattices, are determined experimentally. Additionally, training and multi-quantizing procedures were still needed.

V. Proposed Algorithms

In the proposed algorithm, the wavelet packets associated with the 73-Subband decomposition structure [23], are first applied to mean-removed input source image. The wavelet representations are then separated as $1d$ to $4l$ subimages. In the $1d$ to $4v$ subimages (which are chosen to be lossy encoded) first, a *hard thresholding* scheme is applied where part of the coefficients with an energy less than a predetermined threshold are set to zero. The threshold (same for all subbands) is based on the required overall bit rate and the reconstructed image quality.

The $1d$ subimage is usually discarded in existing techniques (including the FBI's WSQ). Since this subimage has high-frequency content discarding this subband leads to a slightly blurred reconstructed image. In this work, the $1d$ subimage is quantized using a simple *positive–negative mean* (PNM) scheme. In the PNM scheme, the zero coefficients mostly caused by the hard thresholding scheme are first discarded. The remaining coefficients are then mapped to a binary data containing the mean of the positive and the mean of the negative data. The algorithm is simple (no need for quantizing and indexing), while the quality of the reconstructed image is mostly preserved. The resulting center dead zone reduces picture noise contained in high-frequency subimages.

The $1v$ to $2d$ subimages are quantized using the 8-D lattice E_8 and the $2v$ to $4v$ subimages are quantized using the 4-D lattice D_4. To do so, each subimage is first mapped to the desirable n-D vectors. To form these n-D vectors, the nonzero coefficients of the $1v$ to $3h$ subimages as well as all of the coefficients of the $4d$ and $4v$ subimages are used. In this work, in forming these vectors, the nonneighboring coefficients are selected in a predetermined manner and the last vector is zero padded when necessary. Experiments showed that using the above strategy, we can achieve a better performance (see [24] for details).

To obtain a better trade-off between the rate and distortion, *soft thresholding* is also applied. First, the $1v$ to $4v$ subimages are standardized to have zero-mean and unit-variance; this leads to similar PDFs with *narrow* main lobe. Consequently, the resulting standardized subimages can be quantized using similar algorithms. Furthermore, to save on computational cost decimals of resulting coefficients are discarded. This actually reduces the number of different symbols to be encoded and results in a much simpler entropy encoding procedure, which speeds up the algorithm efficiently with only about 0.04 dB reduction on the *peak signal-to-noise ratios* (PSNR) [23]. Then, optimal bit allocations $R_{j,\,d}$ and the corresponding distortion measures $\mathcal{D}_{j,\,d}$ (at level j and direction d) are determined.

For LVQ with any arbitrary PDF, the expected distortion in $(j,\ d)$th subimage is given by [11], [15]

$$\mathcal{D}_{j,\,d} = \mathcal{C}_{j,\,d} 2^{-2R_{j,\,d}} \tag{4}$$

with,

$$\mathcal{C}_{j,\,d} = \rho_{\dot{\Lambda}} \left[\int_{-\infty}^{\infty} \left\{ [f_{j,\,d}(x)]^{n_{j,\,d}/(2+n_{j,\,d})} \right\} dx \right]^{(2+n_{j,\,d})} \tag{5}$$

where $\rho_{\dot{\Lambda}}$ is the upper bound of the MSE for $n_{j,\,d}$-D lattice, listed in [7], and $R_{j,\,d}$ is the bit rate. To compute the optimal bit assignment for each subimage, we have to minimize

$$\min_{R_{j,\,d}} \left[\dot{\mathcal{D}}_T = \frac{1}{4^J} \mathcal{D}_{4\ell} + \frac{1}{n_T^2} \sum_{i=1}^{3J} \mathcal{B}_{j,\,d} \mathcal{D}_{j,\,d}(i) n_{j,\,d}(i) y_{j,\,d}(i) \right] \tag{6}$$

subject to

$$R_T = \frac{1}{4^J} R_{4\ell} + \frac{1}{n_T^2} \sum_{i=1}^{3J} R_{j,\,d}(i) n_{j,\,d}(i) y_{j,\,d}(i) \tag{7}$$

where $R_{4\ell}$ is the bit requirement for the lowest resolution subimage 4ℓ, and $\mathcal{B}_{j,\,d}$ is a parameter used to control the degree of noise shaping across subimages, given by [11]: $\mathcal{B}_{j,\,d} = \gamma^j \log_{10}(\sigma_{j,\,d}^{2\zeta_{j,\,d}})$, where values of γ and $\zeta_{j,\,d}$ are chosen experimentally to match human vision.

The minimization problem can be solved using Lagrangian multipliers as

$$\frac{\partial}{\partial R_{j,\,d}} \left[\dot{\mathcal{D}}_T - \dot{\lambda} \left(R_T - \frac{1}{4^J} R_{4\ell} - \frac{1}{n_T^2} \cdot \sum_{i=1}^{3J} R_{j,\,d}(i) n_{j,\,d}(i) y_{j,\,d}(i) \right) \right] = 0 \tag{8}$$

where $\dot{\lambda}$ is a Lagrangian multiplier. Taking the partial derivative with respect to $R_{j,\,d}$, we get

$$R_{j,\,d} = \frac{1}{2} \log_2 \left(\frac{\ln(4) \mathcal{B}_{j,\,d} \mathcal{C}_{j,\,d}}{\dot{\lambda}} \right). \tag{9}$$

As mentioned above, the $1d$ and $4h$ subimages are not vector quantized and the wavelet coefficients in $1d$ subimage are hard thresholded and packed; which leads to a negligible bit requirement for $1d$. Therefore, following the derivation, the expression of λ obtains as

$$\lambda = \ln(4) \left\{ 2^{-2[R_T - ((R_{4\ell}/4^J) + (R_{4h}/4^J))]} \mathcal{E}_{j,\,d} \right\}^{4^J/(4^J-2)} \tag{10}$$

with

$$\mathcal{E}_{j,\,d} = \prod_{i=2}^{3J-1} \left\{ [\mathcal{B}_{j,\,d}(i) \mathcal{C}_{j,\,d}(i)]^{n_{j,\,d}(i) y_{j,\,d}(i)/n_T^2} \right\}. \tag{11}$$

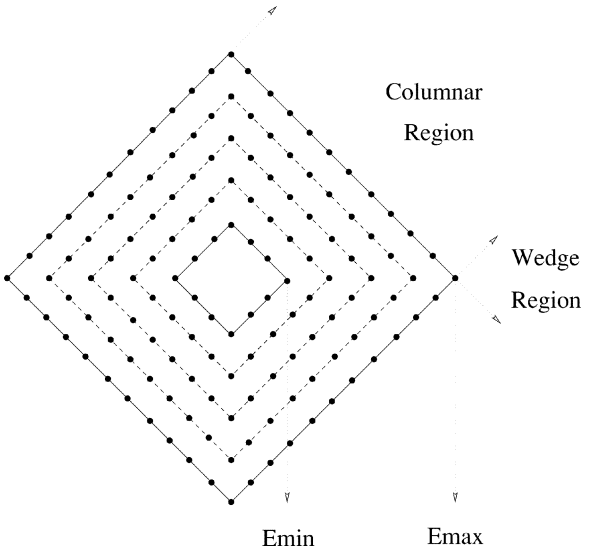


Fig. 5. Proposed uniform pyramid lattice vector quantizer (U-PLVQ) in 2-D.

Finally, substituting (10) into (9), the optimal bit allocation, for each subimage to be quantized by the proposed PLVQ algorithms, is given by

$$R_{j,\,dopt} = \frac{4^J R_T - (R_{4\ell} + R_{4h})}{4^J - 2} + \frac{1}{2}\log_2\left(\frac{\mathcal{B}_{j,\,d}\mathcal{C}_{j,\,d}}{\mathcal{E}_{j,\,d}^{4^J/(4^J-2)}}\right). \tag{12}$$

Using (12) in (4), the optimal expected distortion of the quantizer for each subimage is obtained.

A. Design of Lattice Parameters for U-PLVQ

In the proposed *uniform* PLVQ (U-PLVQ) algorithm, for the $4d$ and $4v$ subimages (with no discarded data) two lattice densities and for the $1v$ to $3h$ subimages (with discarded data) three lattice densities are considered. For each case, the lattice truncation level, r, and lattice scaling factor, s, are designed in order to give the smallest possible ETD, using the allocated bit budget. The computation of the lattice parameters, for each case, is presented below.

For the $1v$ to $3h$ subimages, considering the PDF of the wavelet coefficients with discarded data, three concentric lattice vector quantizers are used where the second lattice is assumed to be as dense as possible and the first and third lattices are empty, as shown for 2-D case in Fig. 5.

Considering the discarded data within these subimages, the allocated bit budget $R_{j,\,d}$ and the corresponding distortion measure $\mathcal{D}_{j,\,d}$ are computed by using (12) and (4), respectively, while using the distribution model with discarded data discussed in Section III-B.

The proposed optimization algorithm starts with the best possible condition under which all input vectors are contained inside the second lattice which has the smallest possible scaling factor. The ETD of the scaled lattice is defined as

$$ETD = P\rho_\Lambda s^2 \tag{13}$$

where P is the probability of the input vectors lying within the second lattice and ρ_Λ is the upper MSE bound, given in [7].

Since the vectors within the first lattice are discarded and the vectors within the third lattice are projected to the outermost shell of the second lattice, we have $P = 1$. We aim at computing the scaling factor and the truncation level so that the obtained ETD becomes smaller than the predetermined distortion measurement $\mathcal{D}_{j,\,d}$, consequently

$$ETD = \alpha\mathcal{D}_{j,\,d} \tag{14}$$

where α is a variable with $\alpha \ll 1$. Using (13) in (14), the scaling factor is obtained as

$$s = \sqrt{\frac{\alpha\mathcal{D}_{j,\,d}}{\rho_\Lambda P}}\,. \tag{15}$$

The truncation levels are obtained from (3) as

$$r_0 = \left\lceil \frac{E_{\min}}{s} \right\rceil, \qquad r_1 = \left\lceil \frac{E_{\max}}{s} \right\rceil \tag{16}$$

where $E_{\min}$ is the minimum energy of input vectors, $E_{\max}$ starts with the maximum energy (L^1 norm) of input vectors, and $\lceil a \rceil$ denotes the smallest integer greater than or equal to a.

Using the ν_Λ function, the codebook size can be determined as

$$|c| = [\nu_\Lambda(r_1) - \nu_\Lambda(r_0)] \tag{17}$$

the required bit rate is then

$$\tilde{R}_{j,\,d} = \frac{\log_2(\nu_\Lambda(r_1) - \nu_\Lambda(r_0))}{n_{j,\,d}} \tag{18}$$

where $n_{j,\,d}$ is lattice dimension in the $(j,\,d)$th subimage.

The optimization algorithm firstly computes the CDF of the L^1 norm of the input vectors. Starting with a small value for α (e.g., 0.3) and a maximum value for $E_{\max}$, the corresponding values for s, r_0, r_1, and $\tilde{R}_{j,\,d}$ are obtained using (15), (16), and (18), respectively.

If $\tilde{R}_{j,\,d} \le R_{j,\,d}$ bpp, the condition is met. Otherwise, P (and consequently $E_{\max}$) is reduced leading to a higher s and hence a smaller r_i and $\tilde{R}_{j,\,d}$.

In fact, in this situation the algorithm allows a small number of vectors to be projected on the outermost shell of the second lattice. However, since α is as small as possible (and consequently the lattice is as dense as possible), the ETD has the smallest possible value. If the current condition cannot be met, even with the smallest appropriate $E_{\max}$, the algorithm allows a slightly higher α. Continuing with the procedure, the algorithm converges for most sources in few iterations to the best lattice structure; which gives the smallest quantization distortion for the allocated bit budget to the subimage. Additionally, at each iteration, if the value of α tends to a bit rate which is not affordable, the algorithm does not continue the loop and checks for a higher ETD.

Notice that no assumptions about the different parameters are made and there is no need to go through training and multi-quantizing procedures. Furthermore, since the range of desired values of $E_{\max}$ is very small, the algorithm converges in few iterations. A flow chart of the algorithm is depicted in Fig. 6.

The algorithm above presents several advantages:

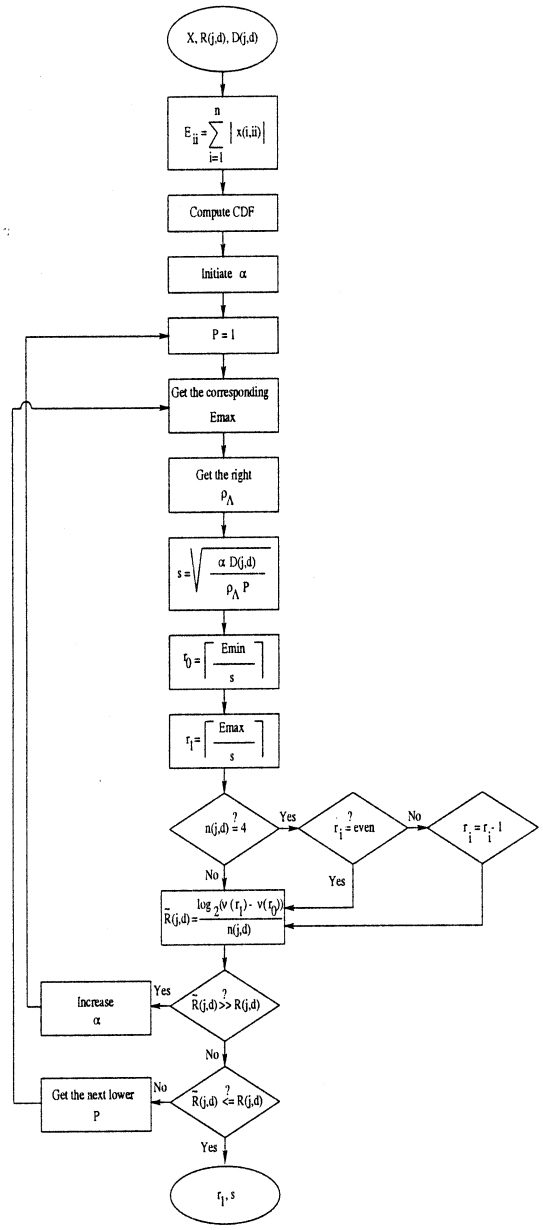


Fig. 6. Flow chart for computation of truncation level, r_1, and scaling factor, s, for proposed U-PLVQ algorithm (with discarded data).

- bit budget distributed more accurately among the subimages;
- no bit rate needs to be allocated to the zero-probable discarded data, which reduces the required bit rate $\tilde{R}_{j,\,d}$ (and hence the codebook size);
- with less bit requirement, the optimization algorithm can converge while still checking for a small α (and hence a small ETD);
- the algorithm designs the lattice structures in a way that gives less quantization distortion for the allocated bit rate.

Following the above procedure, the lattice codewords are then indexed, using the fast encoding method in [6]. A schematic block diagram of the proposed compression/decompression algorithm is shown in Fig. 7.

Indices of source vectors projected to the outermost shell of the lattice should be transmitted as side information. Since the

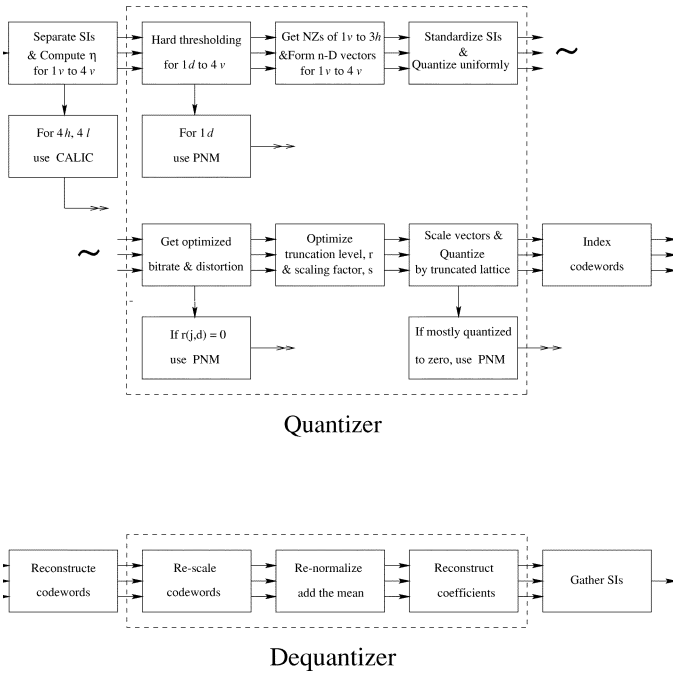


Fig. 7. Schematic block diagram of proposed compression algorithm.

algorithm is designed in a way to have most of source vectors contained in the lattice, the amount of corresponding side information is negligible [e.g., $(610 \times 8 \times 2)/(512 \times 512) = 0.0372$ bpp for a fingerprint image with hard thresholding of 0.1].

The average information of codebook, zeroth order entropy, is measured as

$$\mathcal{R}_{j,\,d} = -\frac{1}{n_{j,\,d}} \sum_{i=1}^{L} p(v_i) \log_2 p(v_i) \ \text{bpp} \tag{19}$$

where $p(v_i)$ is the probability of selecting the n-D index vector v_i, belonging to the obtained indices at level j and corresponding to the orientation d, during coding of (j, d)th subimage, and L is the number of different index vectors in that subimage. The total estimated entropy, $\mathcal{R}_T$, is obtained by

$$\mathcal{R}_T = \frac{\sum_{j=1}^{J} \sum_{d=1}^{3} \mathcal{R}_{j,\,d} n_{j,\,d} y_{j,\,d}}{n_T^2} \ \text{bpp} \tag{20}$$

where J is the depth of the wavelet decomposition, $n_{j,\,d} \times y_{j,\,d}$ is the codebook size at the $n \times y$ subimage, and n_T^2 is the size of the original image.

B. Design of Lattice Parameters for PU-PLVQ

The lattice vector quantizer produces minimum distortion for inputs with uniform distributions. For an arbitrary joint PDF $f_X(\boldsymbol{x}) \in \mathbb{R}^n$, the "optimal" lattice point density should be proportional to $f_X(\boldsymbol{x})^{n/(n+2)}$ [8]. This equation shows that an optimal LVQ must have a denser clustering of lattice points for more probable vectors. To do this, the bit budget must be actually allocated proportionally to the probability of vectors. One way to expand on this idea, is to assume that the source distribution is piecewise uniform between a few surfaces of constant probability. This results in a quantizer that places lattices of

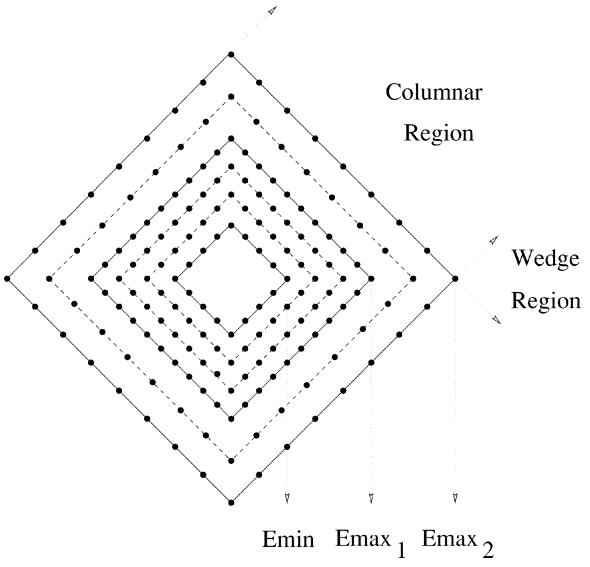


Fig. 8. Proposed piecewise-uniform pyramid lattice vector quantizer (PU-PLVQ) in 2-D.

constant density between concentric surfaces of constant probability. This may not result in the minimum entropy output, but it does result in a lower achievable bit rate after entropy coding a finite amount of quantizer outputs with a complexity constrained system [28].

To do so, in our proposed *piecewise-uniform* PLVQ (PU-PLVQ) algorithm, for the $4d$ and $4v$ subimages (with no discarded data) three lattice densities and for the $1v$ to $3h$ subimages (with discarded data) four lattice densities are considered. Our proposed algorithm for computing the lattice parameters is presented below.

For the $1v$ to $3h$ subimages, considering the joint PDF of input sources with discarded data, four concentric lattice vector quantizers are applied which are separated by three surfaces of constant probabilities. Having a highly nonuniform and sharp PDF, the second lattice is assumed to be as dense as possible, the third lattice is sparse, and the first and fourth lattices are empty, as shown for 2-D case in Fig. 8. The third lattice with a sparse density, is designed to include *less probable* high-energy vectors which mostly fall into wedge regions. In fact, the third lattice is actually used to quantize the edge information more accurately. By allowing the density of lattice points in the third volume to be much lower than the second, the truncation of the third lattice does not need to be restricted. Consequently, the third lattice can actually contain a large portion or all of the volume that a high-energy source vector might fall into. Note that since the third lattice is a truncated sparse lattice it does not lead to a significant increase in the codebook size.

For a PU-PLVQ, the ETD is defined as

$$ETD = \rho_\Lambda \left(P_1 s_1^2 + P_2 s_2^2\right) \tag{21}$$

where P_i is the probability of input vectors lying within the $(i+1)$th lattice, s_i is the scaling factor of the $(i+1)$th lattice.

Since the vectors within the first lattice are discarded and the vectors within the fourth lattice are projected to the outermost shell of the third lattice, we have

$$P_2 = 1 - P_1. \tag{22}$$

The ratio of scaling factors is

$$k = \frac{s_1}{s_2} \tag{23}$$

where $k < 1$. Using (14), (22), and (23) in (21), we get

$$s_1 = \sqrt{\frac{k^2 \alpha \mathcal{D}_{j,\,d}}{\rho_\Lambda (P_1(k^2-1)+1)}}\,. \tag{24}$$

The total codebook size is

$$|c| = \left\{ \left[\nu_\Lambda(r_1) - \nu_\Lambda(r_0)\right] + \left[\nu_\Lambda(r_2) - \nu_\Lambda\left(\frac{s_1}{s_2} r_1\right)\right] \right\} \tag{25}$$

where the truncation levels are given as

$$r_0 = \left\lceil \frac{E_{\min}}{s_1} \right\rceil, \quad r_1 = \left\lceil \frac{E_{\max_1}}{s_1} \right\rceil, \quad r_2 = \left\lceil \frac{k E_{\max_2}}{s_1} \right\rceil \tag{26}$$

where $E_{\min}$ is the minimum energy of input vectors and $E_{\max_i}$ starts with maximum energy of input vectors. Thus, the required bit rate becomes

$$\tilde{R}_{j,\,d} = P_1 R_1 + (1 - P_1) R_2 \text{ bpp} \tag{27}$$

with,

$$R_1 = \frac{\log_2(\nu_\Lambda(r_1) - \nu_\Lambda(r_0))}{n_{j,\,d}},$$

$$R_2 = \frac{\log_2\left(\nu_\Lambda(r_2) - \nu_\Lambda\left(\frac{s_1}{s_2} r_1\right)\right)}{n_{j,\,d}}.$$

Considering the discarded data within the subimages, the allocated bit budget $R_{j,\,d}$ and the corresponding distortion measure $\mathcal{D}_{j,\,d}$ are computed using the distribution model discussed in Section III-B. Computing the CDF of the L^1 norm of the input vectors, the optimization algorithm initially starts with the best possible condition under which all input vectors are contained inside the second and third lattices where they have the smallest possible scaling factors. Starting with a small value for α and the maximum energy for $E_{\max_i}$, the corresponding values of the s_1, r_i, and $\tilde{R}_{j,\,d}$ are computed. The condition is examined using Eq. (27). If $\tilde{R}_{j,\,d} \leq R_{j,\,d}$ the condition is met. Otherwise, first k increases leading to a smaller difference between two scales. If this is not enough, P_1 decreases leading to a smaller $E_{\max_1}$. If the bit budget is still less than the required $\tilde{R}_{j,\,d}$, $E_{\max_2}$ decreases and allows a small number of vectors to be projected on the outermost shell of the third lattice r_2. If this condition can not be met, the algorithm checks a slightly higher α, and iterations continue. The experimental results showed that with $\alpha \ll 1$ which leads to a small codebook size (requiring less bits), the proposed algorithm always computes the truncation levels and scaling factors which lead to a small ETD measure in few iterations. Furthermore, in each iteration if the value of the $E_{\max_2}$ or $E_{\max_1}$, or k tends to a very high bit rate, the algorithm does not continue the loop and checks the next condition. A flow chart of the proposed algorithm is depicted in Fig. 9.

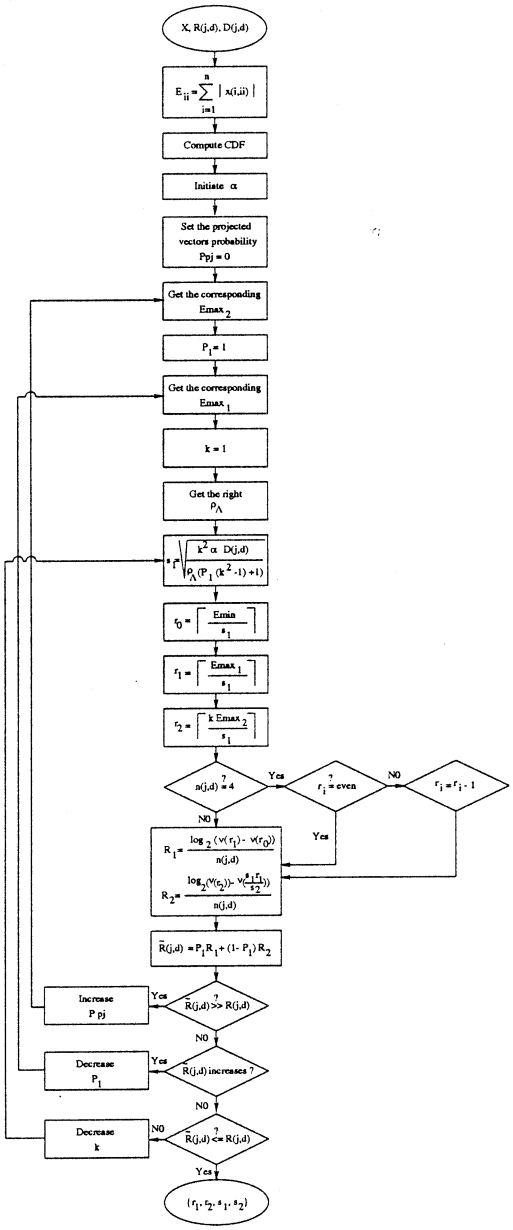


Fig. 9. Flow chart for computation of truncation levels, r_i, and scaling factors, s_i, for proposed PU-PLVQ algorithm (with discarded data).

Despite a small increase in the computation of the lattice parameters, the PU-PLVQ algorithm is preferred over the U-PLVQ algorithm given the following advantages: It considers the (nonuniform) joint PDF of input sources, resolves the wedge region problem encountered with pyramidal lattices, and therefore results in a better performance (see performance comparisons in Section VI).

For the PU-PLVQ case, the average information of codebook (zeroth order entropy) is

$$\mathcal{R}_{j,\,d} = P_1 \mathcal{R}_{1,\,j,\,d} + (1 - P_1)\mathcal{R}_{2,\,j,\,d} \ \text{bpp} \tag{28}$$

where for each lattice $\mathcal{R}_{i,\,j,\,d}$ is obtained using

$$\mathcal{R}_{i,\,j,\,d} = -\frac{1}{n_{j,\,d}} \sum_{ii=1}^{L_i} p(v_{i,\,ii}) \log_2 p(v_{i,\,ii}) \tag{29}$$

TABLE II
PERFORMANCE OF DIFFERENT LOSSLESS CODERS [*LD*: LEVEL-DIRECTION, *Ad*: ADAPTIVE, *Ar*: ARITHMETIC, *Hu*: HUFFMAN, AND *Ca*: CALIC]

LD	Bytes	bpp	AdAr	AdHu	CaAr	CaHu
$4h$	2048	11	1601	1580	1178	1128
4ℓ	2048	13	1926	1865	1405	1312

TABLE III
PERFORMANCE OF DIFFERENT LOSSLESS CODERS REGARDING LOCATIONS OF NONZERO ELEMENTS [*LD*: LEVEL-DIRECTION, *NZ*: NONZERO ELEMENTS, *Ad*: ADAPTIVE, *Ar*: ARITHMETIC, AND *Hu*: HUFFMAN]

LD	Bytes	NZ	AdAr	AdHu	JBIG	8:1
$1v$	65536	948	462	8254	546	8192
$2v$	16384	6690	1369	2469	1554	2048
$2h$	16384	6608	1420	2464	1569	2048
$3d$	4096	3374	506	729	646	512
$3v$	4096	4590	511	740	646	512
$3h$	4096	4566	509	742	649	512

where $p(v_{i,\,ii})$ is the probability of selecting the n-D index vector $v_{i,\,ii}$, during the coding of the ith concentric lattice in that subimage. The total estimated entropy, $\mathcal{R}_T$, is computed as

$$\mathcal{R}_T = \frac{\sum_{j=1}^{J} \sum_{d=1}^{3} \mathcal{R}_{j,\,d} n_{j,\,d} y_{j,\,d}}{n_T^2} \ \text{bpp} \tag{30}$$

where J is the depth of the wavelet decomposition, $n_{j,\,d} \times y_{j,\,d}$ is the codebook size for each $n \times y$ subimage, and n_T^2 is the size of original image.

C. Lossless Compression Algorithms

Lossless compression is chosen for the part of the data which has high effect on the reconstructed image quality. For the $4h$ and 4ℓ subimages, we examined several lossless coding schemes. To adapt to nonstationarity in the data, adaptive *arithmetic*, adaptive *Huffman*, and the *CALIC* (context-based adaptive lossless image codec) [34] schemes were examined. Table II displays some of the results obtained for a typical fingerprint image using Johnston-32D filter.

Given the coarse resolution data (lowpassed), the CALIC with its Huffman version was found to give the best performance.

As described in Section V, the nonzero coefficients of the $1v$ to $3h$ subimages are quantized. Consequently, the locations of nonzero coefficients must be transmitted as side information. To reduce the required bit rate, the following procedure is proposed. First, indices of nonzero elements are mapped to a binary image with ones in the nonzero locations. The obtained binary images have few nonzero elements which are weakly correlated. To encode these binary images, different lossless techniques were examined. These included: the JBIG (Joint Bilevel Image Experts Group) [25], adaptive arithmetic coding [16], adaptive Huffman coding [27], squeezing 8 binary bits into 1 compressed grey-level byte, etc.

Table III illustrates the performance of different methods (in bytes) for some of the subimages, for a fingerprint image with hard thresholding level of 0.067.

From the results, we note that

- due to the low correlation in the data, a simple zero-memory encoder should be satisfactory;

TABLE IV
PSNRs AND CPU COSTS FOR DIFFERENT DECOMPOSITION STRUCTURES, SHOWN IN FIG. 1, USING DIFFERENT FILTERS (5% OF BEST COEFFICIENTS ARE USED WITH NO QUANTIZATION) [*EBBBS*: ENTROPY-BASED BEST BASIS SELECTION, *OWT*: ORDINARY WAVELET TRANSFORM]

Structure	*Johnst-32D*		*Symmet-16*		*Coif-18*	
	PSNR	*CPU*	*PSNR*	*CPU*	*PSNR*	*CPU*
EBBBS	30.77	108.3	30.66	100.2	30.57	107.5
73Subband	30.77	28.1	30.66	23.0	30.59	23.8
64Subband	30.71	28.8	30.60	23.5	30.53	24.2
OWT	30.26	7.8	30.07	5.0	29.96	5.4

TABLE V
SEVERAL QUADRATURE MIRROR FILTER COEFFICIENTS. IN JOHNSTON FILTER, COEFFICIENTS ARE LISTED FROM CENTER TO END [J_{32D}: JOHNSTON-32D, S_{16}: SYMMETRIC-16, C_{18}: COIFMAN-18 , D_{12}: DAUBECHIES-12 TAP]

Filter	*Quadrature Mirror Filter Coefficients*			
J_{32D}	0.463674	0.132972	-0.099338	-0.044524
	0.054812	0.019472	-0.034964	-0.007961
	0.022704	0.002069	-0.014228	0.000842
	0.008181	-0.001969	-0.003971	0.002245
S_{16}	0.002672	-0.000428	-0.021145	0.005386
	0.069490	-0.038493	-0.073462	0.515398
	1.099106	0.680745	-0.086653	-0.202648
	0.010758	0.044823	-0.000766	-0.004783
C_{18}	-0.002683	0.005503	0.016583	-0.046507
	-0.043220	0.286503	0.561285	0.302983
	-0.050770	-0.058196	0.024434	0.011229
	-0.006369	-0.001820	0.000790	0.000329
	-0.000050	-0.0000243		
D_{12}	0.111540	0.494623	0.751133	0.315250
	-0.226264	-0.129766	0.097501	0.027522
	-0.031582	0.000553	0.004777	-0.001077

- the adaptive Arithmetic coding gave the best performance;
- using the proposed technique, the bit requirement for the side information to be transmitted has decreased significantly.

In addition, the indices of lattice codewords as well as the PNM values are adaptive Arithmetic coded.

VI. EXPERIMENTAL RESULTS

Both the U-PLVQ and the PU-PLVQ algorithms were implemented and tested on a database of 512×512 grey-level fingerprint images.

First, the performance of the proposed 73-Subband decomposition structure was investigated. Using a typical fingerprint image, the performance of different types of five-level decomposition structures, shown in Fig. 1, were examined. To remove the effect of quantization on the performance of different structures, the quantization process was not applied. Table IV shows the resulting *peak signal-to-noise ratio* (PSNR), in decibels, and the required *central processing unit* (CPU) time, in seconds, for each structure (the CPU times are recorded to give a relative measure of the computational cost of each structure). The actual filter coefficients are listed in Table V.

For grey-level 8 *bits per pixel* (bpp) images, the PSNR, in decibels, is computed as

$$PSNR = 20 \log_{10} \frac{255}{RMSE} \text{ dB} \tag{31}$$

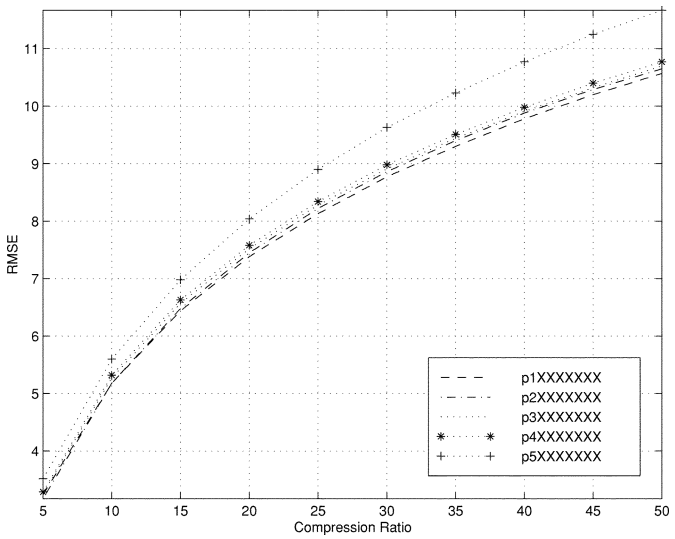


Fig. 10. RMSE of different decomposition structures, using different filters (no quantization).

with *root mean square error* (RMSE) defined as

$$RMSE = \sqrt{\frac{1}{NM} \sum_{n=1}^{N} \sum_{m=1}^{M} [f(n, m) - \hat{f}(n, m)]^2} \tag{32}$$

where N and M are the lengths of image, and f and $\hat{f}$ are the original and reconstructed intensities.

The RMSE performance of different structures is displayed in Fig. 10. Also, the S_{16} family was found to slightly outperform the *Daubechies 12*-tap orthonormal wavelet. Additionally, different types of biorthogonal filters were used with results showing the superior performance of the Johnston-$32D$ filter.

The structures shown in Fig. 1 were also evaluated when used in conjunction with the complete coding process. Implementing the different structures associated with the Johnston filter, the obtained wavelet representations were compressed using the proposed PU-PLVQ algorithm. In this case, as discussed in Section V, the $1d$ to $4h$ subimages were hard thresholded such that only 5% of the wavelet coefficients (in energy sense) were retained. Table VI shows the resulting PSNRs.

Tables IV and VI show that despite the very high computation complexity of the EBBBS method, compared to that of the proposed fixed 73-Subband decomposition, their performances are very similar.

Furthermore, by comparing the performance of the proposed 73-Subband and the FBI's 64-Subband in these tables, one can see that in all cases the 73-Subband outperforms the 64-Subband while it is computationally less expensive. These Tables also show the efficiency of the wavelet packets compared to the ordinary wavelet transform.

Subsequently, the performance of the proposed uniform and piecewise-uniform compression algorithms, at different bit rates, was compared to that of other prominent image compression algorithms.

First, the average performances of the proposed algorithms when using the root lattices E_8 and D_4, and the integer lattices $\mathbb{Z}^8$ and $\mathbb{Z}^4$ were tested. The algorithms were run on a set of 24

TABLE VI
PERFORMANCE OF DIFFERENT DECOMPOSITION STRUCTURES, SHOWN IN FIG. 1, USING THE PROPOSED PU-PLVQ ALGORITHM (5% OF BEST COEFFICIENTS ARE USED) [*EBBBS*: ENTROPY-BASED BEST BASIS SELECTION, *OWT*: ORDINARY WAVELET TRANSFORM]

Structure	*PSNR (dB)*	*Bit Rate (bpp)*
EBBBS	30.765	0.1537
73–Subband	30.761	0.1542
64–Subband	30.703	0.1542
OWT	30.199	0.1564

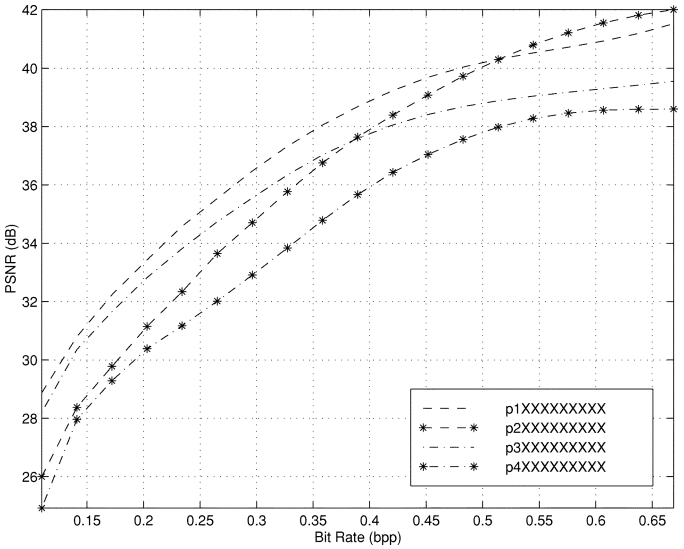


Fig. 11. Average performance of different standard lattices, for fingerprint images [ED: E_8&D_4 root lattices, $\mathbb{Z}$: integer lattices].

fingerprint images of size 512 $\times$ 512. The results are shown in Fig. 11.

Fig. 12 and Table VII shows the average performances of the proposed algorithms, the standard JPEG [32], and the FBI's standard WSQ [2]. The same set of images were used for the proposed algorithms as well as the JPEG algorithm. For the WSQ algorithm, the rate-distortion performance reported in [32] for 512 $\times$ 512 fingerprint images was used. The results obtained from different experiments showed that at low bit rates the proposed algorithms with the root lattices E_8 and D_4 outperform the case with the integer lattices $\mathbb{Z}^8$ and $\mathbb{Z}^4$ by about 2 dB in most cases (see Fig. 11).

To give a relative measure of the computational cost of the proposed algorithms, Table VIII lists the required CPU costs, in minutes, for a sample fingerprint image (using MATLAB on an UNIX platform). The reported CPU costs comprise all stages of the compression/decompression algorithms, including the wavelet packets.

Fig. 12 and Table VII show the superior performance of the proposed image compression algorithms compared to that of the standard JPEG and the standard WSQ. These results also show that the proposed PU-PLVQ algorithm outperforms the U-PLVQ algorithm, while the computational cost of the former technique is not significantly higher than the latter, as shown in Table VIII. Note that the original images are compressed only once, while they will be decompressed frequently in most applications. As can be seen from Table VIII, even with MATLAB codes, the CPU cost

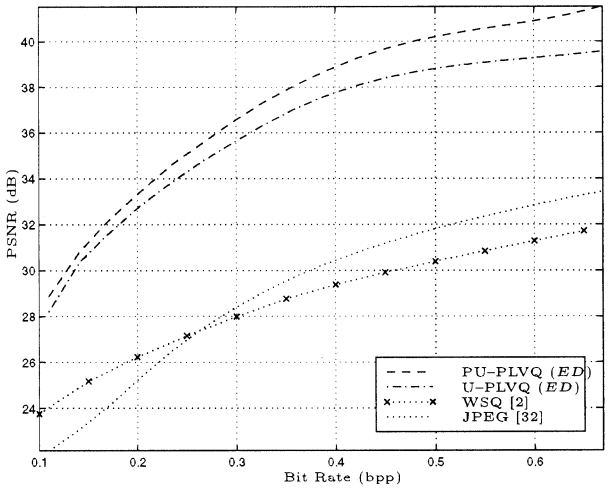


Fig. 12. Average performance of the proposed algorithms as well as the JPEG and the WSQ algorithms, for various bit rates.

TABLE VII
RESULTING AVERAGE PSNRS FROM DIFFERENT ALGORITHMS

Bit Rate	*JPEG [32]*	*WSQ [2]*	*PU–PLVQ*	*U–PLVQ*
0.60	32.84	31.28	40.88	39.27
0.45	31.18	29.91	39.67	38.40
0.30	28.40	27.99	36.58	35.65
0.15	23.40	25.18	31.26	30.78

TABLE VIII
CPU COSTS OF A SAMPLE FINGERPRINT IMAGE

Bit Rate	*PU–PLVQ*		*U–PLVQ*	
	com.	*decom.*	*com.*	*decom.*
0.60	4.16	1.63	2.87	1.32
0.45	3.49	0.98	2.41	0.92
0.30	3.76	0.59	2.16	0.57
0.15	3.44	0.34	2.03	0.33

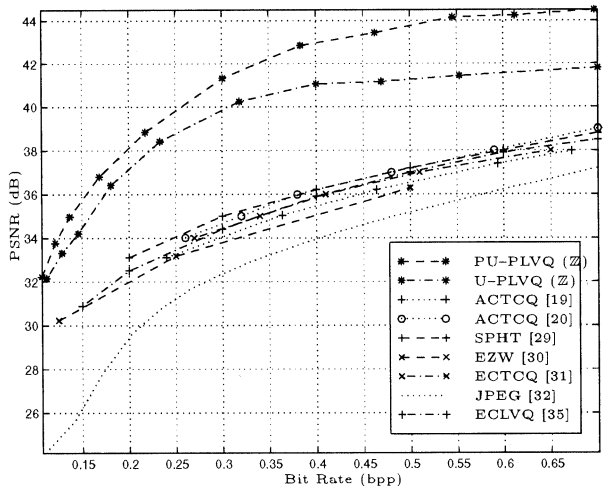


Fig. 13. Performance of different image compression algorithms, for "Lena" image [$\mathbb{Z}$: integer lattices].

of the PU-PLVQ decompression algorithm is comparable with that of the U-PLVQ decompression algorithm.

To compare the performance of the proposed algorithm with that of other existing algorithms using other images, the ubiquitous "Lena" image 512 $\times$ 512 was also encoded. Using this

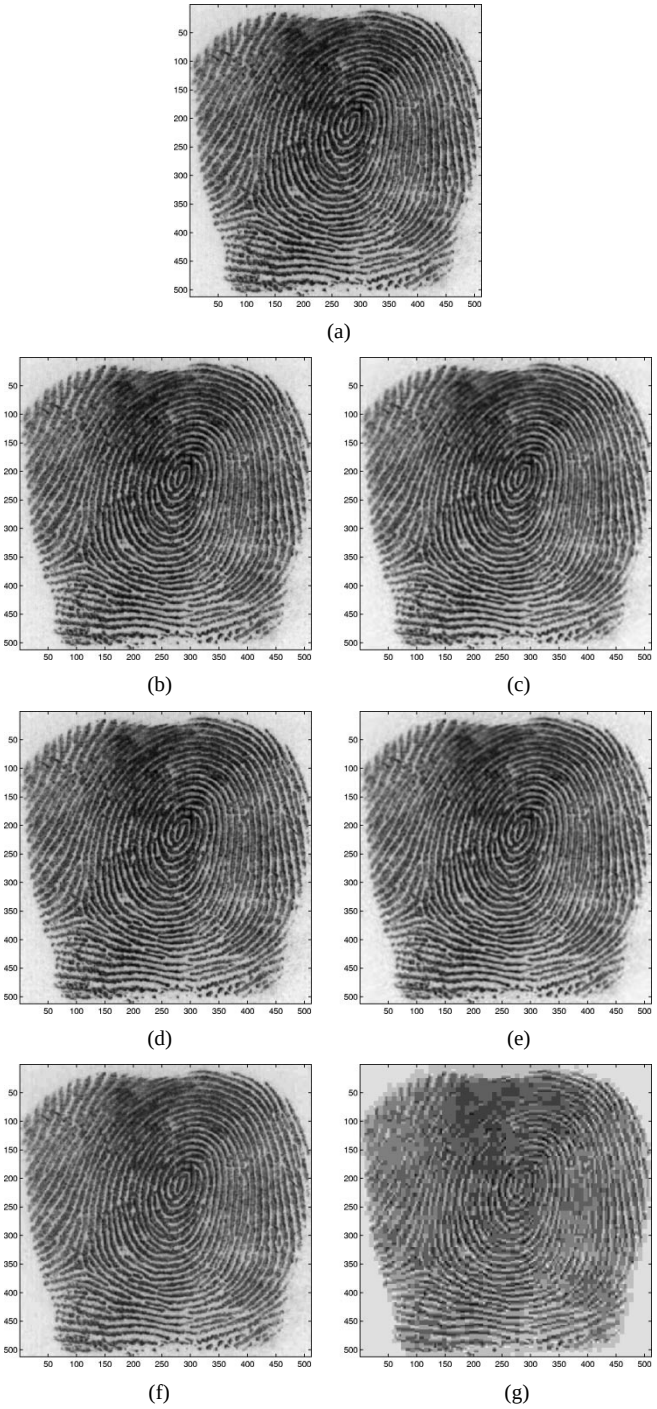


Fig. 14. (a) Original image; PSNRs for 12 : 1 (0.65 bpp) and 53 : 1 (0.15 bpp); (b) PU-PLVQ: 41.28 dB; (c) PU-PLVQ: 29.96 dB; (d) U-PLVQ: 39.26 dB; (e) U-PLVQ: 29.58 dB; (f) JPEG: 30.95 dB; and (g) JPEG: 21.50 dB.

image, the PSNRs obtained from the JPEG, Barlaud's [12], and the proposed PU-PLVQ and U-PLVQ algorithms for 0.17 bpp, were 27.72 dB, 30.3 dB, 36.95 dB, and 36.26 dB, respectively. Fig. 13 shows the performance of the proposed algorithms as well as that of several prominent image compression algorithms [19], [20], [29]–[32], [35] when encoding this image.

Figs. 14 and 15 demonstrate the quality of the reconstructed images obtained using different algorithms, at various bit rates.

Also, to show that the fidelity of the minutiae is preserved during the proposed compression/decompression algorithm,

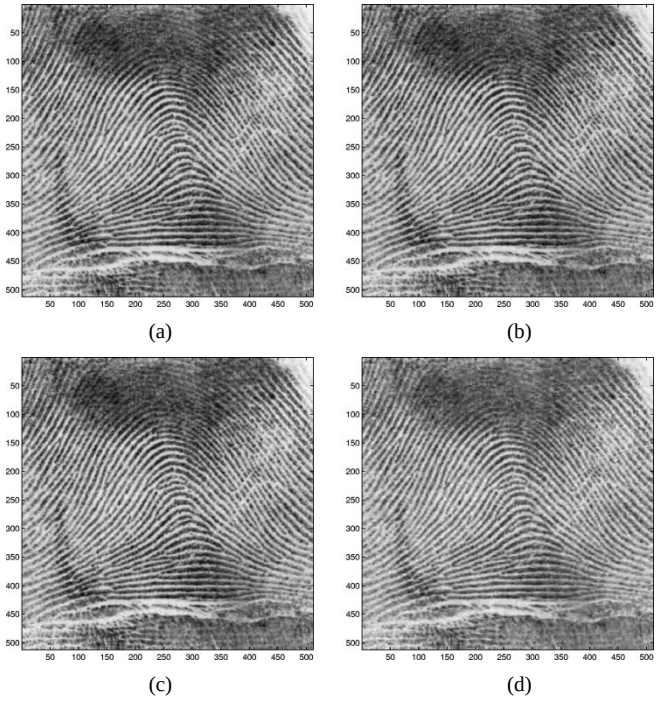


Fig. 15. (a) Original image. Reconstructed images with 20 : 1 compression (0.4 bpp); (b) proposed PU-PLVQ (40.10 dB); (c) WSQ (31.41 dB); and (d) JPEG (29.52 dB).

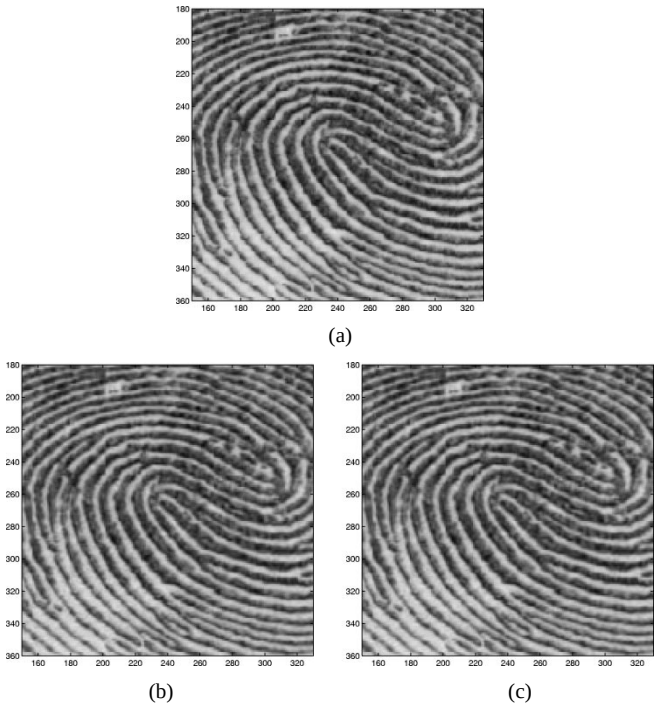


Fig. 16. (a) Original image. PSNRs for 20 : 1 (0.40 bpp) and 53 : 1 (0.15 bpp), using proposed PU-PLVQ algorithm: (b) 34.98 dB; and (c) 28.07 dB.

Fig. 16 shows an enlarged portion of an original and a reconstructed image of a typical fingerprint image, at compression ratios of 20 : 1 and 53 : 1.

VII. Conclusion

A new compression algorithm (with two variations) adapted to fingerprint images is introduced. A modified wavelet packet

scheme is developed to decorrelate image pixels, which uses the proposed fixed decomposition structure (matched to fingerprint images).

A statistical analysis of the subbands, led to an optimization of the design of different subband coders and the bit allocation among subbands. The algorithm uses both hard and soft thresholding schemes to make the process fast and robust.

We presented a new design method for optimizing the lattice parameters, for both uniform and piecewise-uniform PLVQs, with no need for preassumptions. The design is based on a precise distribution model of the wavelet coefficients in each subimage and aims at achieving the best rate-distortion function. In the proposed piecewise-uniform PLVQ, the wedge problem encountered with pyramidal lattice point shells is also resolved.

The proposed algorithm adapts to variability in input data and to overall allocated bit budgets. At very low bit rates, the developed PNM method improves the quality of the reconstructed image while keeping the algorithm simple and fast. A method for reducing bit requirement of necessary side information has also been introduced.

In addition to the proposed algorithm, the performance of other prominent algorithms is also discussed. Experimental results clearly show that the proposed compression technique results in higher quality reconstructed images compared to that of other prominent algorithms operating at similar bit rates. The work developed here will be of substantial benefit in forensic applications where high compression ratios while preserving ridge details would help in the storage and communication problems. Experiments carried in conjunction with the Queensland Police showed that our algorithms, even when operating at $50:1$ compression, can be used in subsequent identification stages based on reconstructed images.

REFERENCES

[1] M. Abramowitz and I. A. Stegun, *Handbook of Mathematical Functions*. New York: Dover, 1965.

[2] J. N. Bradley and C. M. Brislawn. (1993) Proposed first-generation WSQ bit allocation procedure. [Online]. Available: ftp.c3.lanl.gov in /pub/WSQ/documents/bit.alloc.ps.

[3] F. W. Campbell and J. Kulikowski, "Orientation selectivity of the human visual system," *J. Physiol.*, vol. 197, pp. 437–441, 1966.

[4] R. R. Coifman and M. V. Wickerhauser, "Entropy-based algorithm for best basis selection," *IEEE Trans. Inform. Theory*, vol. 38, pp. 713–718, Mar. 1992.

[5] J. H. Conway and N. J. A. Sloane, "Fast quantization and decoding algorithm for lattice quantizers and codes," *IEEE Trans. Inform. Theory*, vol. IT-28, pp. 227–232, Mar. 1982.

[6] ——, "A fast encoding method for lattice codes and quantizers," *IEEE Trans. Inform. Theory*, vol. IT-29, pp. 820–824, Nov. 1983.

[7] ——, "A lower bound on the average error of vector quantizers," *IEEE Trans. Inform. Theory*, vol. IT-31, pp. 106–109, Jan. 1985.

[8] ——, *Sphere Packings, Lattices and Groups*. New York: Springer-Verlag, 1988.

[9] P. C. Cosman, R. M. Gray, and M. Vetterli, "Vector quantization of image subbands: A survey," *IEEE Trans. Image Processing*, vol. 5, pp. 202–225, Feb. 1996.

[10] M. Deriche, S. Kasaei, and A. Bouzerdoum, "Fingerprint compression using a piecewise-uniform pyramid lattice vector quantization," in *Proc. ICIP 99*, Oct. 1999.

[11] M. Antonini *et al.*, "Image coding using wavelet transform," *IEEE Trans. Image Processing*, vol. 1, pp. 205–220, Apr. 1992.

[12] M. Barlaud *et al.*, "Pyramid lattice vector quantization for multiscale image coding," *IEEE Trans. Image Processing*, vol. 3, pp. 367–381, July 1994.

[13] N. Farvardin and J. W. Modestino, "Optimum quantizer performance for a class of non-Gaussian memoryless sources," *IEEE Trans. Inform. Theory*, vol. IT-30, pp. 485–497, May 1984.

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

[15] A. Gersho, "Asymptotically optimal block quantization," *IEEE Trans. Inform. Theory*, vol. IT-25, pp. 373–380, July 1979.

[16] R. C. Gonzalez and R. E. Woods, *Digital Image Processing*. Reading, : Addison-Wesley, 1992.

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

[18] J. D. Johnston, "A filter family designed for use in quadrature mirror filter banks," in *Proc. ICASSP*, Denver, CO, Apr. 1980, pp. 291–294.

[19] R. L. Joshi, V. J. Crump, and T. R. Fischer, "Image subband coding using arithmetic coded trellis coded quantization," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 5, pp. 515–523, Dec. 1995.

[20] R. L. Joshi, T. R. Fischer, and R. H. Bamberger, "Optimum classification in subband coding of images," in *IEEE Int. Conf. Image Processing*, vol. 2, Austin, TX, Nov. 1994, pp. 883–887.

[21] S. Kasaei and M. Deriche, "Fingerprint compression using a piecewise-uniform pyramid lattice vector quantization," in *Proc. ICASSP'97*, Apr. 1997, pp. 3117–3120.

[22] S. Kasaei, M. Deriche, and B. Boashash, "Fingerprint compression using a modified wavelet transform and pyramid lattice vector quantization," in *Proc. IEEE Region Ten Conf., Digital Signal Processing Applications, TENCON*, Perth, Australia, Nov. 1996, pp. 798–803.

[23] ——, "Performance analysis of fingerprint compression using an efficient wavelet transform algorithm," in *Proc. IEEE Conf. 4th Int. Symp. Signal Processing and Its Applications*, Gold Coast, Australia, Aug. 1996, pp. 433–436.

[24] ——, "Fingerprint compression using wavelet packet transform and pyramid lattice vector quantization," *IEICE Special Section on Digital Signal Processing*, vol. E80-A, no. 8, pp. 1446–1452, Aug. 1997.

[25] M. Kuhn. (1996) Joint Bi-level Image experts Group (JBIG). *Resources, Source Code: Markus Kuhn's JBIG Implementation* [Online]. Available: http://www.internz.com/compression-pointers.html.

[26] Y. Linde, A. Buzo, and R. M. Gray, "An algorithm for vector quantizer design," *IEEE Trans. Commun.*, vol. COM-28, pp. 84–95, Jan. 1980.

[27] M. Nelson, *The Data Compression Book*. New York: M & T Books, 1992.

[28] W. C. Powell and S. G. Wilson, "Lattice quantization in the wavelet domain," *Proc. SPIE*, vol. 2034, pp. 218–229, 1993.

[29] 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.

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

[31] P. Sriram and M. W. Marcellin, "Image coding using wavelet transforms and entropy-constrained-trellis coded quantization," *IEEE Trans. Image Processing*, vol. 4, pp. 725–733, June 1995.

[32] G. K. Wallace, "The JPEG still picture compression standard," *Commun. ACM*, vol. 34, no. 4, pp. 31–44, Apr. 1991.

[33] A. B. Watson, *Digital Images and Human Vision*. Cambridge, MA: MIT Press, 1993.

[34] X. Wu, N. Memon, and K. Sayood. (1996) Context-based Adaptive Lossless Image Codec (CALIC). [Online]. Available: ftp.csd.uwo.ca in /pub/from_wu/v.huff/ and v.arith/.

[35] Z. Mohd-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.

Shohreh Kasaei received the B.S. degree in electronic engineering from Isfahan University of Technology (IUT), Iran, in 1986. She then joined the University of Ryukyus, Okinawa, Japan, from which she received the M.S. degree in 1994. In 1998, she received the Ph.D. degree in image processing from the Signal Processing Research Centre at Queensland University of Technology, Brisbane, Australia.

Since then, she has been an Assistant Professor with Sharif University of Technology, Iran. Her research interests are image processing with primary emphasis on content-based video compression, content-based image retrieval, video restoration, motion estimation, virtual studios, fingerprint authentication/identification, and multidimensional signal modeling and prediction.

Dr. Kasaei was awarded the Best Ph.D. Student Studying Overseas Award in 1998.

Mohamed Deriche (S'90–M'91–SM'99) graduated from the National Polytechnic School of Algeria in 1984 with an Engineer's Degree. He then joined the University of Minnesota, where he completed his M.S. and Ph.D. in 1988, and 1992 respectively.

He was a Post Doctorate Fellow with the Radiology Department, University of Minnesota, Minneapolis, in the area of MRI. He then joined the Queensland University of Technology as a Lecturer in 1994, Senior Lecturer (1997), and Associate Professor (2000). He has published over 100 papers. His research interests are multiscale signal processing, wavelets, fractals, spectral estimation, with particular emphasis on Multimedia compression applications and biomedical applications.

Dr. Deriche was awarded the Best Electrical Engineering Student in 1984. He delivered a tutorial on wavelets at ISSPA '99. He received the IEEE Third Millennium Medal in 2000.

Boualem Boashash (M'82–SM'89–F'99) received the Diplome d'ingenieur-Physique - Electronique degree from ICPI University, Lyon, France, in 1978, and the M.S. and Doctorate degrees from the Institut National Polytechnique de Grenoble, France, in 1979 and 1982, respectively.

He is the Director of the Signal Processing Research Centre at QUT. In 1984, he joined University of Queensland, Australia, as a Lecturer, then Senior Lecturer (1986), and Reader (1989). In 1991, he joined the Queensland University of Technology as the Foundation Professor of Signal Processing and Director of the Signal Processing Research Centre. B. Boashash was Technical Chairman of ICASSP 1994 and is currently the Chairman of the International Symposium on Signal Processing and Its Applications, organized regularly in Australia since 1987. He is the editor of two books including *Time-Frequency Signal Analysis: Methods and Applications* (White Plains, NY: Longman-Cheshire, 1991). He has written over 200 technical publications. His research interests are time-frequency signal analysis, spectral estimation, signal detection and classification, and higher-order spectra, mobile communications, and biomedical signal processing.

Dr. Boashash is a Fellow of IE Australia and IREE.