Image Compression Using Wavelet Transform and Multiresolution Decomposition

Amir Averbuch, Danny Lazar, and Moshe Israeli

Abstract—**Schemes for image compression of black-and-white images based on the wavelet transform are presented. The multiresolution nature of the discrete wavelet transform is proven as a powerful tool to represent images decomposed along the vertical and horizontal directions using the pyramidal multiresolution scheme. The wavelet transform decomposes the image into a set of subimages called shapes with different resolutions corresponding to different frequency bands. Hence, different allocations are tested, assuming that details at high resolution and diagonal directions are less visible to the human eye. The resulted coefficients are vector quantized (VQ) using the LGB algorithm. By using an error correction method that approximates the reconstructed coefficients quantization error, we minimize distortion for a given compression rate at low computational cost.**

Several compression techniques are tested. In the first experiment, several 512 × 512 images are trained together and common table codes created. Using these tables, the training sequence black-and-white images achieve a compression ratio of 60–65 and a PSNR of 30–33.

To investigate the compression on images not part of the training set, many 480 × 480 images of uncalibrated faces are trained together and yield global tables code. Images of faces outside the training set are compressed and reconstructed using the resulting tables. The compression ratio is 40; PSNR's are 30–36. Images from the training set have similar compression values compression and quality.

Finally, another compression method based on the and vector bit allocation is examined. The idea is based on allocating different numbers of bits to the vectors, depending on their values, encoding the "type" of each vector (large or small) on a bit map. The vectors in each shape are grouped and trained together according to the magnitude of their variances. A vector that has higher variance is quantized using longer tables. Hence, in each shape, the vector coefficients are quantized using several tables: each vector by the appropriate table. The relation of each vector to its quantization table is saved in a map file. The major improvement is achieved by making the process more efficient and fast since smaller tables are used, and fewer comparisons for locating the closest vector in the table have to be made. The bottleneck of searching large tables, which is very inefficient in all VQ's, is eliminated.

The compression ratio and the quality of the reconstructed faces outside the training set have similar results as the previous compression method—compression ratio of 35–36 and PSNR of 35–37—although faces reconstructed from the training set are slightly better. Distinct wavelet filters are tested, and the best results are achieved by applying the biorthogonal filters. The results presented here are comparable with the best results published recently in terms of PSNR.

Manuscript received August 17, 1993; revised April 14, 1995. The associate editor coordinating the review of this paper and approving it for publication was Prof. Michel Barlaud.

A. Averbuch and D. Lazar are with the School of Mathematical Sciences, Tel-Aviv University, Tel-Aviv, Israel.

M. Israeli is with the Faculty of Computer Science, Technion—Israel Institute of Technology, Haifa, Israel.

Publisher Item Identifier S 1057-7149(96)00146-7.

I. Introduction

IMAGE compression means reducing the volume of data needed in order to represent an image. Image compression techniques are divided into two main techniques: transforms (DCT, JPEG, FFT, Wavelet) and nontransforms (PCM, DPCM). The PCM and DPCM methods enable us to compress the data without transforming it based on the signal's model. Compression can be achieved by transforming the data, projecting it on a basis of functions, and then encoding the resulted coefficients. One encoding approach is based on quantizing the coefficients using scalar or vector quantization (VQ). The subband coding method even splits the coefficients according to distinct areas of behavior, whereas each is quantized separately according to its own special parameters and predetermined compression ratio. From the comparison between different compression methods that are described in [10], it appears that while achieving an image representation of 0.5 b/pixel, the accompanied PSNR is usually 27–29 dB. A compression based on the wavelet transform resulted in a PSNR of 29.11–32.10, and compression ratios of 10–38 are reported in [3].

The compression method, which is being used in this paper, is based on applying a VQ on the wavelet coefficients resulting from the wavelet transform of the trained images, using the pyramidal multiresolution architecture (Mallat [11]). The multiresolution tree decompositions based on the wavelet transform are given in Fig. 1.

The possibility of compression by quantizing and encoding the wavelet coefficients relies on the assumption that details at high resolution are less visible to the eye and therefore can be eliminated or reconstructed with low-order precision. The wavelet coefficients are coded according to their location (in the sense of pyramidal level and texture shape) and their importance for the quality of the final reconstructed image. Higher bit rate per pixel and lengths of vectors are allocated for shapes that are found to be more important, and the error correction method, which approximates the error after the quantization of the reconstructed coefficients, is applied.

The compression of images using quantization of the wavelet coefficients is demonstrated in this work by several experiments.

A predetermined compression ratio to be used for each shape in the pyramidal structure is given, and different lengths of vectors are tested, in order to find the best lengths that enable both better quality and lower CPU computational cost. After determining the appropriate allocations for each level and shape, several images are trained together, and common table codes are created. In order to verify our approach, we

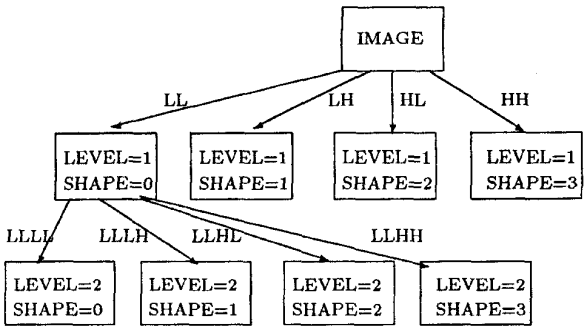


Fig. 1. Tables associated with image decomposed into two levels.

start with processing images that were taken from the training set. Using these tables, the black-and-white images that were part of the training sequence achieve a compression ratio of 60–65 and a PSNR of 30–33. The decomposition has four levels, and the coefficients, representing the texture of the image at the lower decomposition level, were not quantized at all.

Additional experiments based on the same quantization approach are tested in order to compress images taken outside the training set. Many images of faces (not calibrated) are trained together and yield a global tables code. The lengths of the vectors are different from the local training, as well as the number of the levels (usually three) in the decomposition. In addition, since we have only three levels, the compression is reduced, and the coefficients at the finest scale are uniformly quantized (using 8 b per coefficient). Images of faces outside the training set are compressed and reconstructed using the resulted tables. The compression ratio is 40, and the PSNR is 30–36. Images taken from the training set achieve similar values for compression and quality.

In the previous methods, the length of the vectors in the tables for each shape is established experimentally, and all the vectors in the shape have to be quantized according to the same large table. At the second part of this work, another approach for VQ, based on vector bit allocation, is described. The vectors in each shape are separated according to their variances. Using the bit allocation formula, each vector gets a number of bits by which it should be represented. Vectors with a similar number of bits are trained together and create an appropriate table code. It is obvious that vectors with higher variances (which implies that they are more important to the image reconstruction) will be quantized using larger tables, whereas unimportant vectors will be approximated by shorter tables. The vector coefficients are quantized in each shape using several and smaller tables, which makes the compression process faster and more efficient, and the decision on the vectors length is computed mathematically and not determined by heuristic experiments. The compression and quality of faces compressed and reconstructed outside the training set have similar results to the previous ones (compression ratio of 35–36 and PSNR of 35–37), although faces that are reconstructed from the training set are slightly better when compared with the previous methods. It is interesting to observe that in spite of the differences between the two approaches, the quality of the results are pretty similar.

In the paper, Section II describes how the energy distribution and the statistical properties of the wavelet coefficients yield the bit allocations for each shape and determines the number of decomposition levels that should be used. Obtaining the desired bit allocation is considered with the lengths of the training tables and their vectors. In addition, the improvement of the VQ in producing sufficient quality with shorter vector and tables by approximating the quantization error is explained. The results of the comparison of the different uses of wavelet filters are summarized in this section.

Section III brings the experimental results that determine the lengths of the training tables and their vectors and demonstrates the coding process on images taken within the training set. The changes needed for creating global tables are also considered as are the experimental results for compressing and restoring faces taken outside the training set. Section IV explains the compression method and its quantization based on the vector bit allocation.

II. Quantization

Applying the wavelet transform on images does not reduce the amount of the data to be compressed. A common way to reduce the number of bits required for the compression phase is to quantize the coefficients and apply some lossless compression such as Huffman or arithmetic coding on the quantized coefficients or their representative indices.

A. Principle of VQ

The encoding and decoding scheme in Fig. 2 demonstrates the use of wavelet transform together with VQ.

The encoding is performed by approximating the sequence to be encoded by a vector belonging to the codebook. Several techniques and methods are known for creating these codebooks. One of them is the LGB algorithm [12]. The codebook is created and optimized using classification based on a training set comprised of vectors belonging to different images; it converges iteratively to a locally optimal codebook.

1) The LGB Algorithm:

0) Initialization: Given N = number of levels, a distortion threshold $\epsilon \geq 0.0$, an initial N-level reproduction alphabet $\hat{A}_0$, and a training sequence $\{x_j;\ j = 1, \ldots, n\}$. Set $m = 0$ and $D_{-1} = \infty$.
1) Given $\hat{A}_m = \{y_i;\ i = 1, \ldots, N\}$, find its minimum distortion partition $\mathcal{P}(\hat{A}_m) = \{S_i; i = 1, \ldots, N\}$ of the training sequence: $x_j \in S_i$ if $d(x_j, y_i) \leq d(x_j, y_l)$ for all l.
   Compute the resulting average distortion

$$D_m = D(\{\hat{A}_m, \mathcal{P}(\hat{A}_m)\}) = \frac{\sum_{j=1}^{n} \min_{y \in \hat{A}_m} d(x_j, y)}{n}.$$

2) If $(D_{m-1} - D_m)/D_m \leq \epsilon$, halt with $\hat{A}_m$ and $\mathcal{P}(\hat{A}_m)$ describing the final quantizer. Otherwise, continue.
3) Find the optimal reproduction alphabet $\hat{x}(\mathcal{P}(\hat{A}_m)) = \{\hat{x}(S_i); i = 1, \ldots, N\}$ for $\mathcal{P}(\hat{A}_m)$. Set $\hat{A}_{m+1} \triangleq \hat{x}(\mathcal{P}(\hat{A}_m))$. Replace m by $m + 1$, and go to 1).

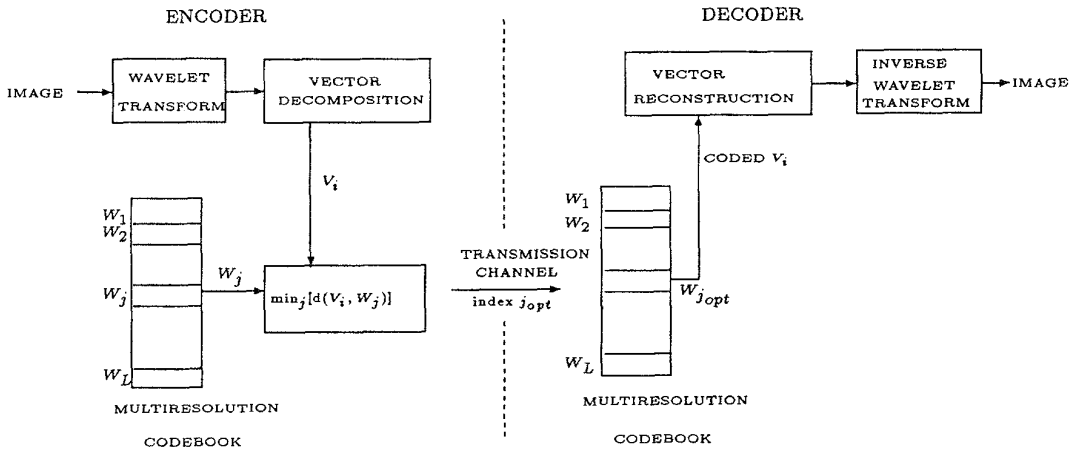


Fig. 2. Encoding/decoding scheme.

Observe above that while designing the quantizer, only partitions of the training sequence (the input alphabet) are considered. Once the final codebook $\hat{A}_m$ is obtained, however, it is used on new data outside the training sequence with the optimum nearest-neighbor rule, that is, an optimum partition of K-dimensional Euclidean space.

2) Choice of $\hat{A}_0$: There are several ways to choose the initial reproduction alphabet $\hat{A}_0$ required by the algorithm. The splitting technique is useful when one wishes to design quantizers of successively higher rates until achieving an acceptable level of distortion. This method considers an M-level quantizer with $M = 2^R, R = 0, 1, \ldots$, and continues until it achieves an initial guess for an N-level quantizer as follows:

0) Initialization: Set $M = 1$, and define $\hat{A}_0 = \hat{x}(A)$, which is the centroid of the entire alphabet (the centroid of the training sequence if a sample distribution is used).
1) Given the reproduction alphabet $\hat{A}_0(M)$ containing M vectors $\{y_i; i = 1, \ldots, M\}$, "split" each vector y_i into two close vectors $y_i + \epsilon$ and $y_i - \epsilon$, where ϵ is a fixed perturbation vector. The collection $\tilde{A}$ of $\{y_i + \epsilon, y_i - \epsilon, i = 1, \ldots, M\}$ has $2M$ vectors. Replace M by $2M$.
2) Does $M = N$? If so, set $\hat{A}_0 = \tilde{A}(M)$ and halt. $\hat{A}_0$ is then the initial reproduction alphabet for the N-level quantization algorithm. If not, run the LGB algorithm for an M-level quantizer on $\tilde{A}(M)$ and then return to step 1).

Comparing different methods yielded that choosing $\hat{A}_0$ by "splitting" performs better and thus was chosen to be the initial alphabet method in this work.

B. Scalar Quantization versus VQ

The wavelet coefficients can be quantized, among other methods, by scalar quantization (SQ) or VQ.

Comparing SQ versus VQ should take into consideration two aspects: compression ratio achieved and its quality.

Table I compares between SQ and VQ. For SQ, we used the values that were computed by Max's quantizer [13]. The quantizer uses the best values for achieving 1.0–4.0 b/pixel representation while quantizing samples that have a probability density function (pdf) of Laplacian, Gaussian, or Gamma functions. According to our analysis, the Gamma function represents best the wavelet pdf.

TABLE I
SCALAR AND VECTOR QUANTIZATIONS RESULTS APPLYING ON LENNA'S COEFFICIENTS IN SHAPE 1 LEVELS 3 AND 4. N IS THE NUMBER OF THE VECTORS IN THE TABLE, AND K IS THE LENGTH OF EACH VECTOR. FOR THE LGB CASE, THE COMPRESSION R BPP IS CALCULATED BY $N = 2^{KR}$. MAX SNR BRINGS THE SNR OF THE COEFFICIENTS GENERATED BY THE MAX QUANTIZER

bpp	Level	Shape	N	K	MAX SNR	LGB SNR
1.0	3	1	2	1	1.68	2.32
1.0	3	1	4	2	-	3.75
1.0	3	1	16	4	-	5.47
1.0	4	1	2	1	2.10	2.45
1.0	4	1	4	2	-	3.88
1.0	4	1	16	4	-	6.38
2.0	3	1	4	1	6.40	6.82
2.0	3	1	16	2	-	8.67
2.0	4	1	4	1	6.83	7.33
2.0	4	1	16	2	-	9.93
3.0	3	1	8	1	7.89	11.85
3.0	3	1	64	2	-	13.75
3.0	4	1	8	1	9.09	12.48
3.0	4	1	64	2	-	16.80

When using $K = 1$, VQ becomes SQ, but as shown in Table I, SQ that is achieved by LGB gives better results than by the usage of Max's approximations for the Gamma function since the pdf of the wavelet coefficients is slightly different from the Gamma function, and the LGB algorithm adapts to the wavelet's pdf very easily. In addition, it is observed that by using VQ, a much higher SNR is achieved in comparison with the results using SQ for the same compression ratio. Therefore, the current application uses VQ.

C. Statistical Distribution of the Wavelet Coefficients

The pdf of the wavelet coefficients in each shape (direction) is important in order to understand the behavior at different

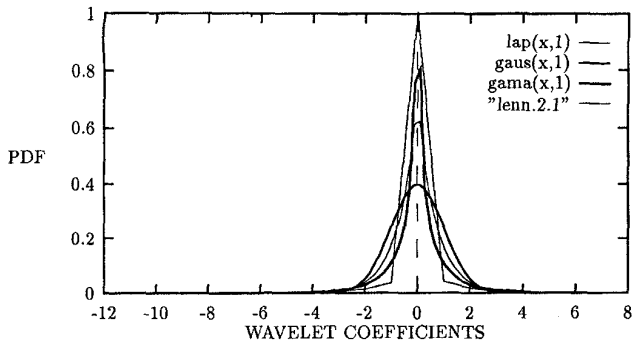


Fig. 3. Distribution of wavelet coefficients and their approximations. The wavelet coefficients were sampled from shape 1 in level 2 of Lenna's decomposition and were scaled by the local variance. The graph of lenn.2.1 demonstrates the wavelet pdf distribution. Similar results were given using other images. The graphs of lap(Laplacian), gaus(Gaussian), and gama(Gamma) demonstrates the approximations.

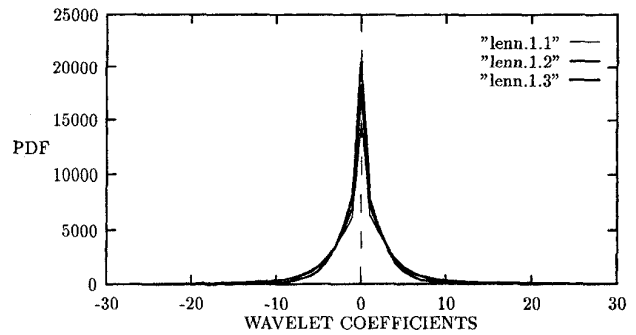


Fig. 4. Distribution of wavelet coefficients: Shapes 1, 2, 3 at level 1 of Lenna. Similar results were obtained by other images.

TABLE II
WAVELET COEFFICIENTS FOR LENNA

LEVEL	SHAPE	MIN	MAX	AVR	VAR	(Min-Avr)/Var	(Max-Avr)/Var
1	1	-100.8	106.3	0.13	8.090	-12.4	13.1
1	2	-53.28	81.37	0.03	5.057	-10.5	16.0
1	3	-34.07	41.33	0.00	3.136	-10.3	12.4
2	1	-343.3	207.8	0.23	26.65	-12.8	7.78
2	2	-198.4	157.6	0.16	15.65	-12.6	10.0
2	3	-113.9	137.7	-0.14	12.17	-9.34	11.3
3	1	-479.6	587.8	-2.16	68.05	-7.01	8.66
3	2	-324.2	257.8	-0.34	38.74	-8.36	6.66
3	3	-256.5	315.1	0.13	35.85	-7.15	8.78
4	1	-931.6	1070	-4.46	196.1	-4.72	5.48
4	2	-593.8	682.2	-3.61	98.13	-6.01	6.98
4	3	-484.5	721.0	0.39	89.85	-5.39	8.02
4	0	-1743	1609	-155	780.5	-2.03	2.26

resolutions. The results are demonstrated using the wavelet coefficients of Lenna. Other images yielded similar results.

Fig. 3 shows the wavelet pdf for level 2 and direction shape = 1 (horizontal edges) together with three approximations: Gaussian, Laplacian, and the Gamma function.

$$\text{Gaussian}(x,\sigma_x) = \frac{1}{\sqrt{2\pi\sigma_x^2}}\exp[-x^2/2\sigma_x^2] \quad (1)$$

$$\text{Laplacian}(x,\sigma_x) = \frac{1}{\sqrt{2}\sigma_x}\exp[-\sqrt{2}|x|/\sigma_x] \quad (2)$$

$$\text{Gamma}(x,\sigma_x) = \frac{\sqrt[4]{3}}{\sqrt{8\pi\sigma_x|x|}}\exp[-\sqrt{3}|x|/2\sigma_x]. \quad (3)$$

TABLE III
ENERGY DISTRIBUTION FOR THE WAVELET COEFFICIENTS OF LENNA AND JOHN. FOR EACH LEVEL AND SHAPE, THE RELATIVE ENERGY IS COMPUTED BY ENERGY OF THE COEFFICIENTS IN THAT SHAPE/TOTAL ENERGY OF THE COEFFICIENTS.

LEVEL	SHAPE	LENNA	JOHN
1	1	0.563	0.173
1	2	0.220	0.126
1	3	0.094	0.034
Sum of Level 1		0.877	0.333
2	1	1.528	0.227
2	2	0.527	0.161
2	3	0.318	0.083
Sum of Level 2		2.373	0.471
3	1	2.492	0.230
3	2	0.807	0.157
3	3	0.691	0.075
Sum of Level 3		3.99	0.462
4	1	5.179	0.401
4	2	1.298	0.441
4	3	1.088	0.093
Sum of Level 4		7.565	0.935
4	0	85.19	97.79

TABLE IV
IMPORTANCE OF LEVEL 1 IN THE DECOMPOSED LENNA. OTHER IMAGES PRODUCED SIMILAR RESULTS, YIELDING THE CONCLUSION THAT THE COEFFICIENTS IN LEVEL 1 ARE NOT NECESSARY. ALL THE COEFFICIENTS THAT ARE LEFT (AT LEVELS 2–4) ARE KEPT UNCHANGED.

Reconstruction Type	PSNR	ERMS
without shape 1 in level 1	35.52	4.26
without shape 2 in level 1	39.69	2.63
without shape 3 in level 1	43.09	1.78
without shapes 1,2 in level 1	34.13	5.00
without shapes 1,2,3	33.60	5.32

Examining the coefficients of the black-and-white images shows that the distribution of the wavelet coefficients remains similar for all the shapes at the same level (see Fig. 4) or at different levels, excluding the lower resolution.

Thus, we may conclude that the pdf of the wavelet coefficients is closely approximated by the Gamma function for all levels and shapes used in our work. Another fact that can be observed in the graphs is that 90% of the coefficients are very small and lie in a very narrow dynamic range around the origin.

Table II demonstrates the behavior of the coefficients. It shows the statistics of each level and shape and describes the smallest and maximal coefficient (MIN and MAX) at each shape, the average and variance (AVR and VAR) of the coefficients in each shape, and the range of each shape after normalizing by the average and variance.

The importance of each level is determined by Table II:

1) The range of the coefficients (which is [MIN, MAX]) and their variances grow as the resolution becomes coarser, which proves that the coefficients at the lower levels have higher importance.
2) After subtracting the average and scaling by the variance, the range of the coefficients becomes smaller and similar to all shapes in each level (see the last two columns in Table II). This fact (which is demonstrated

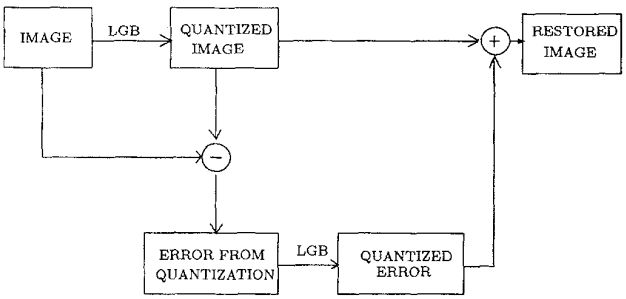


Fig. 5. Error correction.

TABLE V
APPLYING DIFFERENT WAVELET FILTERS ON LENNA. THE COMPRESSION ACHIEVED IS DESCRIBED IN THREE WAYS: ENTROPY MEASURE, THE COMPRESSION ACHIEVED AFTER APPLYING HUFFMAN CODING ON THE RESULTED VECTOR TABLES, AND WITHOUT USING THE HUFFMAN CODING. HUFFMAN CODING, WHICH IS A LOSSLESS COMPRESSION, INCREASES THE COMPRESSION BY 30%. CR DENOTES THE COMPRESSION RATIO.

Filter Type	Filter length	PSNR	ERMS	Compression					
				Entropy		Huffman coding		Without Huff	
				bpp	CR	bpp	CR	bpp	CR
Daubechies	2	29.87	8.18	0.129	61.80	0.129	61.58	0.168	47.51
Daubechies	6	32.06	6.35	0.126	63.14	0.127	62.87	0.168	47.51
Daubechies	8	32.38	6.12	0.127	62.51	0.128	62.11	0.168	47.51
Daubechies	10	32.70	5.90	0.128	62.24	0.129	61.94	0.168	47.51
Daubechies	12	33.10	5.64	0.127	62.59	0.128	62.28	0.168	47.51
Daubechies	16	33.10	5.64	0.128	62.44	0.128	62.09	0.168	47.51
Daubechies	20	32.97	5.72	0.128	62.03	0.129	61.67	0.168	47.51
Beylkin	18	33.41	5.44	0.128	62.05	0.129	61.74	0.168	47.51
Coiflet	6	31.98	6.41	0.128	62.43	0.128	62.15	0.168	47.51
Coiflet	18	33.09	5.64	0.128	62.41	0.128	62.11	0.168	47.51
Coiflet	30	32.83	5.81	0.127	62.99	0.127	62.68	0.168	47.51
Biorthog	9	32.84	5.81	0.128	62.29	0.129	61.99	0.168	47.51

in Fig. 4) enables us to quantize different shapes with similar considerations.

3) In the coarsest level, shape 0 (which contains the smoothed low passed coefficients) consists of a wider range and larger coefficients and variances than the other shapes. This shape contains the most important coefficients. Thus, they will be quantized using a SQ, which might result in lower compression but also in higher reconstruction accuracy.

D. Energy Distribution

From the behavior of the wavelet coefficients and their distribution, it can be seen that most of the coefficients lie in a narrow range around the origin, and the higher levels contain smaller coefficients and variances. Looking at Tables II and III, we see that in each level, shapes 1 and 2 contain more energy and larger coefficients and variances than shape 3. In addition, level 1 contains less energy and smaller coefficients than level 2, which contains less energy than level 3, etc. Tables II and III summarize the analysis results on the wavelet coefficients of Lenna and John. The same behavior is expected for the other images.

E. The Importance of Level 1

The importance of the levels was discussed in the previous sections. Lower levels proved to be more important,

TABLE VI
PROPERTIES OF DIFFERENT LEVELS IN LENNA: LOCAL QUANTIZATION OF THE WAVELET COEFFICIENTS THAT WERE DERIVED FROM THE APPLICATION OF THE BIORTHOGONAL FILTER OF LENGTH 9. FOR EACH LEVEL AND SHAPE, N AND K REPRESENT THE NUMBER OF VECTORS (N) AND THEIR LENGTH (K) IN THE QUANTIZATION TABLES. WHEN EC IS USED, SNR DIFF SHOWS THE QUALITY OF THE APPROXIMATED ERROR. THE COLUMN OF FINAL SNR BRINGS THE SNR OF THE COEFFICIENTS IN EACH SHAPE AFTER APPLYING THE EC. THE ENTROPY MEASURE IS COMPUTED, WHEREAS THE COLUMNS OF HUFFMAN AND WITHOUT HUFFMAN BRING THE EXPERIMENTAL RESULTS OF COMPRESSION BEFORE AND AFTER APPLYING HUFFMAN CODING

LEVEL	SHAPE		N	K	bpp			SNR	SNR Diff	Final SNR
					Entropy	Huffman	without Huff			
2	1		512	36	0.202	0.203	0.25	11.09		
		EC	512	36	0.122	0.122	0.25		8.53	19.22
2	2		512	36	0.176	0.177	0.25	10.16		
		EC	512	36	0.158	0.160	0.25		7.63	17.96
2	3		256	80	0.073	0.074	0.10	7.73		7.73
3	1		256	16	0.413	0.414	0.50	13.06		
		EC	256	16	0.267	0.267	0.50		12.09	25.87
3	2		256	16	0.369	0.371	0.50	12.39		
		EC	256	16	0.315	0.320	0.50		12.03	22.44
3	3		256	16	0.377	0.378	0.50	12.54		
		EC	256	16	0.316	0.321	0.50		12.82	24.13
4	1		64	24	0.210	0.214	0.25	10.76		
		EC	64	24	0.091	0.096	0.25		8.53	19.30
4	2		64	24	0.204	0.208	0.25	9.93		
		EC	64	24	0.097	0.101	0.25		11.69	24.38
4	3		64	24	0.209	0.213	0.25	11.59		
		EC	64	24	0.078	0.086	0.25		10.89	23.03

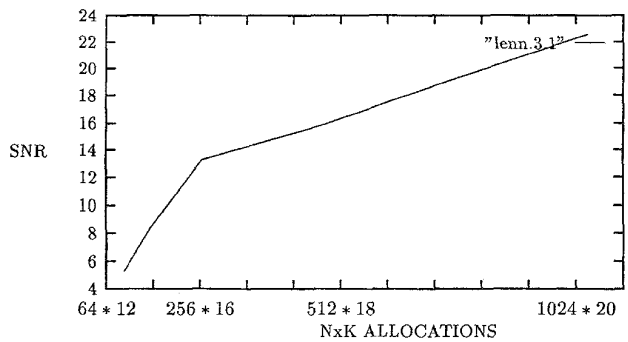


Fig. 6. Influence of different allocations on SNR: Different $N \times K$ that satisfy $N = 2^{RK}$ were tested for $R = 0.5$ b/pixel for shape 1 in level 3. SNR of 13 was reached by allocating $N = 256$ and $K = 16$.

TABLE VII
RESULTS OF TRAINING SEVERAL IMAGES TOGETHER AND RESTORING THEM FROM THE RESULTING TABLES. THE COMPRESSION ACHIEVED IS DESCRIBED IN THREE COLUMNS: THE ENTROPY, THE COMPRESSION ACHIEVED AFTER APPLYING THE HUFFMAN CODE ON THE RESULTING VECTOR TABLES, AND THE COMPRESSION RESULTING WITHOUT USING THE HUFFMAN CODE. CR BRINGS THE COMPRESSION RATIO

IMAGE	PSNR	ERMS	Big Err.	Max Diff.	Min Diff.	Compression					
						Entropy		Huffman Coding		Without H.C.	
						bpp	CR	bpp	CR	bpp	CR
LENNA	31.76	6.58	4112	76	-73	0.124	64.03	0.136	58.56	0.168	47.51
JOHN	32.28	6.19	3407	82	-64	0.124	64.50	0.134	59.31	0.168	47.51
CAROL	30.17	7.90	5869	101	-98	0.133	59.71	0.145	54.96	0.172	46.43
HAGIT	27.39	10.88	14867	111	-175	0.125	63.99	0.137	58.02	0.168	47.51

which is a fact that draws special attention to level 1, which holds for both the least important and the largest coefficient. Efficient quantization of this level, which contains 75% of the coefficients, is critical to the total compression ratio. In addition, during the reconstruction process, the error caused by quantizing its coefficients does not interfere with and affect the other reconstructed levels quality (see Table IV).

We can see that the coefficients in shape 3 are less important for the reconstruction since they are responsible for patterns

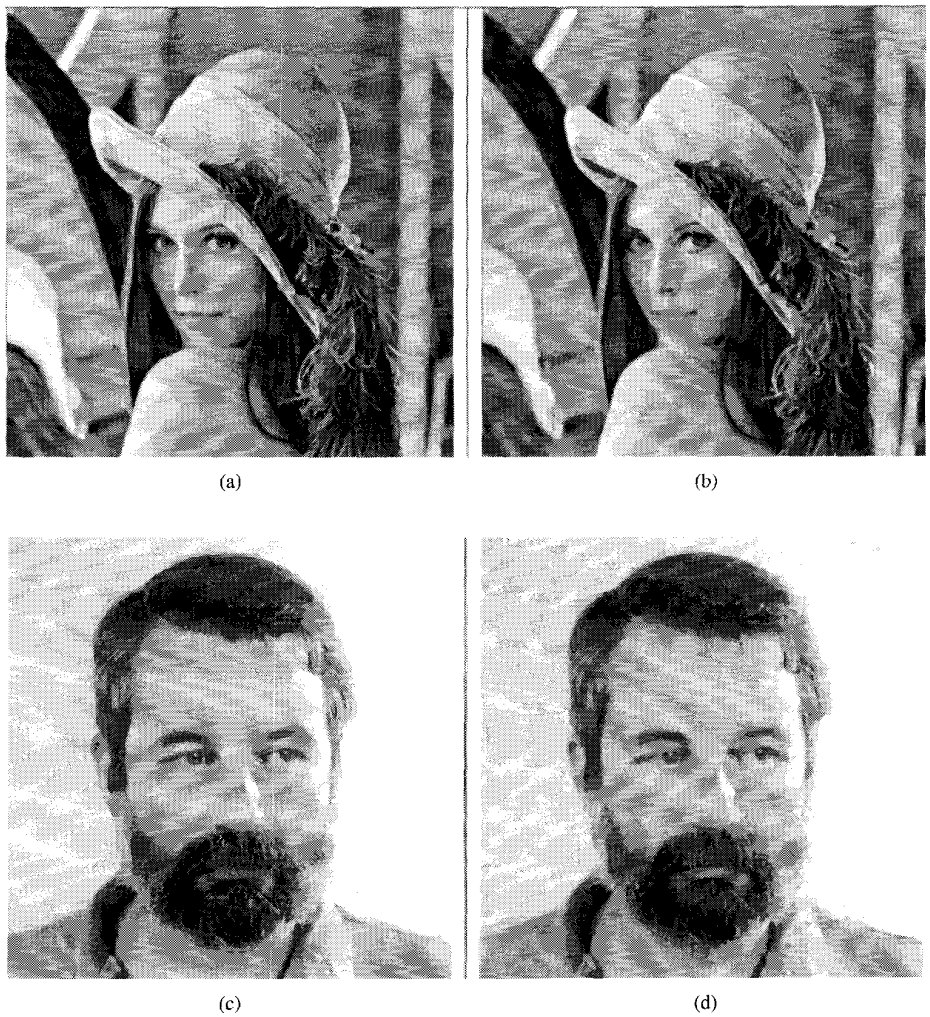


Fig. 7. Training several images together and restoring them from the resulting tables. "LENNA": (a) Original image; (b) compression of 0.136 b/pixel. "JOHN": (c) Original image; (d) compression of 0.134 b/pixel.

that can hardly be seen by the eye, especially their presence in the upper levels. By excluding them, we can still achieve PSNR of 43, whereas by removing the coefficients of shapes 1 or 2, the PSNR decreases to 34–39.

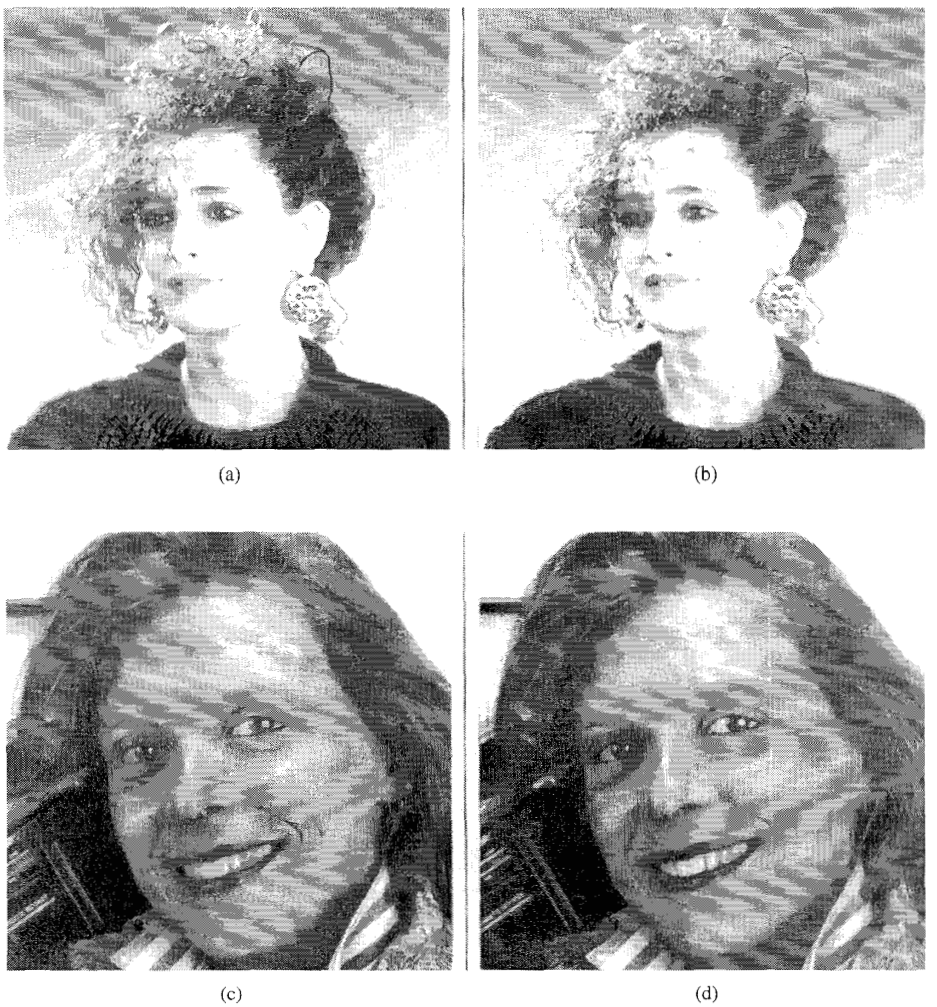
Table IV also directs us to decide whether to ignore all the coefficients in level 1 since it is still possible to achieve high PSNR (higher than 33).

F. Error Correction

Applying the LGB on the wavelet coefficients does not always produce satisfactory results. Better results can be obtained by allocating longer vectors and tables that necessitate heavy computational load related to searching and accessing the tables. The goal is to achieve higher SNR for the reconstructed coefficients with high compression ratios without allocating enormous tables that degrades the performance.

One way to improve the reconstructed results is by using the error correction (EC) method, which approximates the error of the quantization (Fig. 5). The quantization error of the reconstructed image is computed, and then, the LGB quantization is applied on the error. We get an approximation of the error and add it to the previous reconstructed image. This process reduces the differences between the original and the reconstructed image and thus yields better SNR for the quantized coefficients and higher PSNR for the reconstructed image, as well as several fine qualities for the compression:

1) If we decide, in advance, to use allocation of 0.5 b/pixel in a certain shape, it can be done either by allocating N, K (number of vectors in the table and their lengths) for the shape's coefficients such that $N = 2^{0.5K}$ or by allocating 0.25 b/pixel for the shape samples and 0.25 b/pixel for the error samples. The last choice yields higher SNR for the reconstruction. In Table VI, one can see that the final SNR is doubled when we use EC.
2) Splitting the quantization into two steps—quantizing the coefficients and quantizing the error—enables us to use shorter vectors; this is very important for faster computations as well as for achieving higher SNR.



(a) (b) (c) (d)

Fig. 8. Training several images together and restoring them from the resulting tables. "HAGIT": (a) Original image; (b) compression of 0.137 b/pixel. "CAROL": (c) Original image; (d) compression of 0.145 b/pixel.

TABLE VIII
RESULTS OF RESTORING SEVERAL IMAGES OF FACES. FACE 4 IS TAKEN FROM THE TRAINING SET. ALL OTHER FACES WERE TAKEN OUTSIDE THE TRAINING SET.

IMAGE	PSNR	ERMS	Big Err.	Max Diff.	Min Diff.	Compression bpp	Compression CR
FACE 1	35.05	4.50	699	96	-68	0.21	38.22
FACE 2	36.05	4.02	590	89	-58	0.20	39.23
FACE 3	30.73	7.40	6490	103	-71	0.22	36.34
FACE 4	35.60	4.22	727	139	-58	0.20	38.97

3) The values representing the approximated error are very small, and thus, its wavelet coefficients are small as well. This also enables fast computation with fine quality.

G. Comparing Different Wavelet Filters

Distinct QMF filters can be used, which raises the question of which filter is best for our compression methods. Table V presents the level of compression achieved by applying different types of filters. This work uses the biorthogonal wavelet of length 9. Although its PSNR is a little smaller than the best achieved by other filters, its visual quality, judged by several observers, is better.

TABLE IX
VECTOR ALLOCATIONS FOR GLOBAL TRAINING. THE EC LINE DENOTES THE ALLOCATIONS FOR THE EC STEP

Level	Shape		N	K
			128	28
2	1	EC	128	28
			128	28
2	2	EC	128	28
			128	28
2	3	EC	-	-
			128	14
3	1	EC	128	14
			128	14
3	2	EC	128	14
			128	14
3	3	EC	-	-

III. VECTOR QUANTIZATION OF WAVELET COEFFICIENTS: EXPERIMENTAL RESULTS

The sensitivity of the final reconstructed image quality to the quantization errors is more noticeable at lower levels, which necessitates larger allocation tables and higher reconstruction quality. This is true for all images because most of the energy

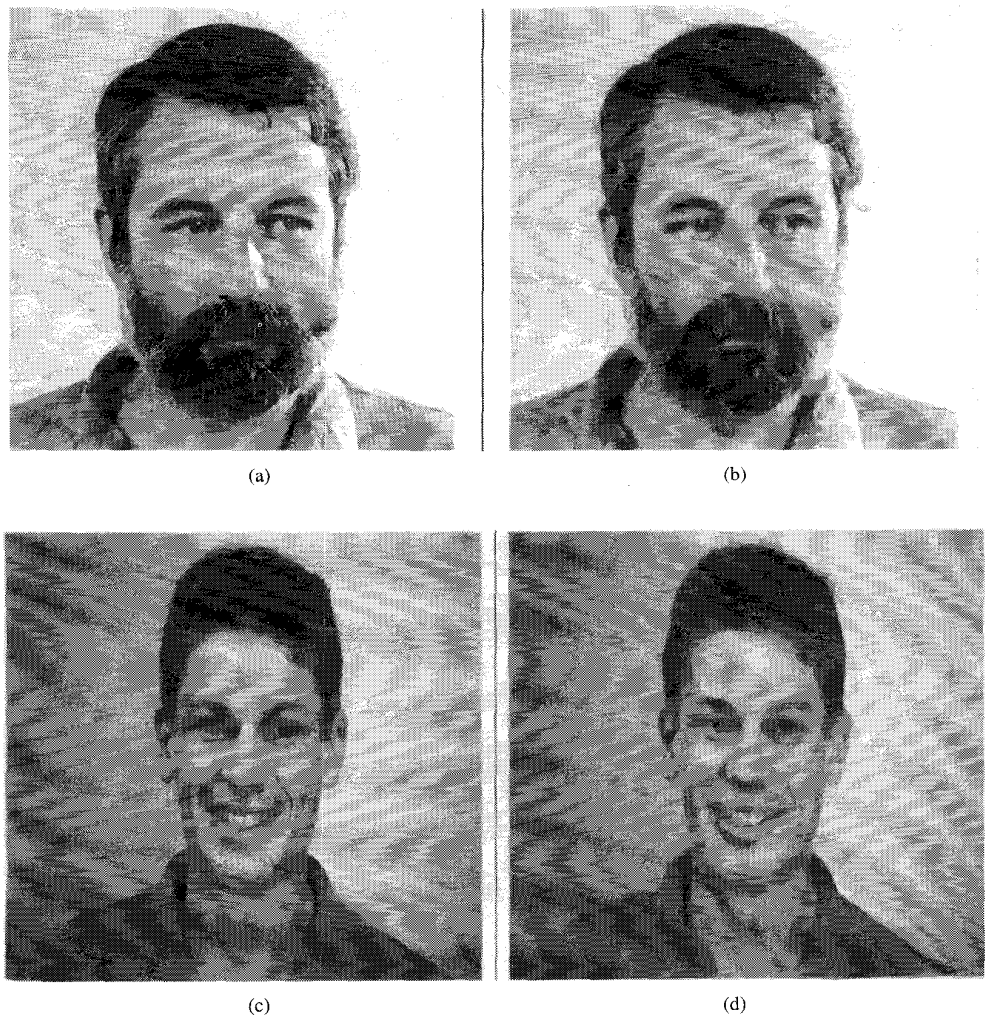


Fig. 10. Restoring faces taken outside and inside the training set. FACE 3: (a) Original image; (b) compression of 0.22 b/pixel. FACE 4: (c) Original image; (d) compression of 0.20 b/pixel.

The dependency between SNR and the vector length (with constant allocation $R = 0.5$) is given in Fig. 6.

Table VI presents the final allocations and its achieved SNR that result from applying similar operations to all shapes on Lenna's coefficients. Close results are obtained by applying these allocations on other images.

B. Allocating Local Tables

The previous section described the decision of how to determine the vector length for each shape. The selection of R, N, and K for each shape is based on this approach. In addition, the coefficients are divided into four levels. The coefficients in level 1 are ignored, and shape 0 (in level 4) remains untouched. Applying these allocations on Lenna yield the results of Table I as well as the PSNR of 31.76 and compression ratio of 58.56.

Several images were trained together. These images, which were part of the training sequence, were restored using the resulted tables and achieved a compression ratio of 60 and a PSNR of 30.17–32.28 (Table VII). The results are displayed in Figs. 7 and 8.

C. Allocating Global Tables

In order to generate global tables, where images outside the training sequence are being compressed, many images of faces (not calibrated) are trained together. In order to overcome problems created by quantization, the depth of the multiresolution decomposition is fixed to 3. This fact reduces the compression ratio: shape 0 in level 3 cannot be compressed at the same bit rate as level 4, whose compression is based on the decomposition to different shapes. In level 3, shape 0 is scalar quantized using 8 b per sample, contrary to the previous training, which keeps the coefficients in shape 0 of level 4 unquantized. Different allocations are used based on the same approaches which were mentioned previously. The new allocations, as well as the final results of the compressed images inside and outside the training set, are displayed in Tables VIII and IX.

The images are displayed in Figs. 9 and 10.

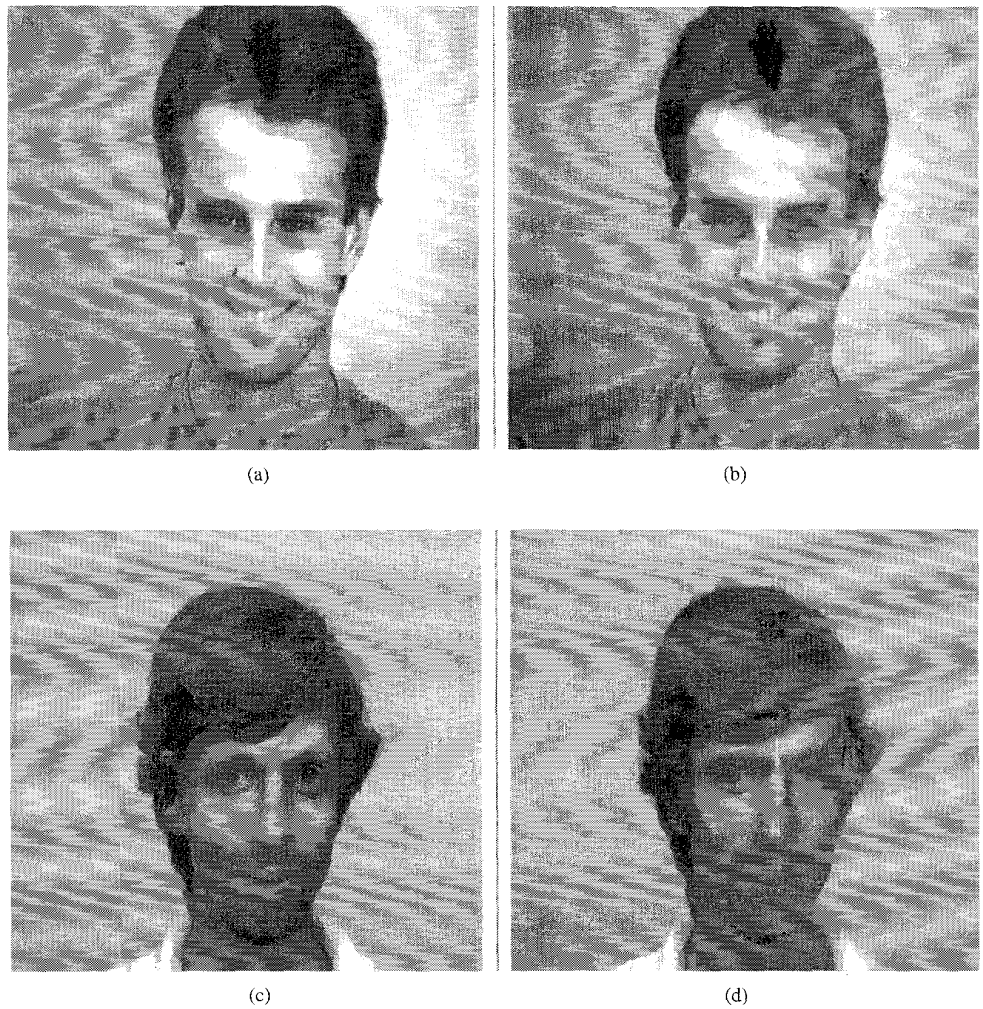


Fig. 9. Restoring faces taken outside the training set. FACE 1: (a) Original image; (b) compression of 0.21 b/pixel. FACE 2: (c) Original image; (d) compression of 0.20 b/pixel.

is generally concentrated in the lower levels. It is interesting to note that the human visual system is sensitive more to contrasts at low frequencies. The evolution of the contrast sensitivity with respect to the frequency of the stimuli is qualitatively the same for a human observer as for our quantized wavelet coefficients.

Thus, using the wavelet transform pyramidal scheme enables us to apply this approach and quantize the lower levels with larger allocations that yield better quality and SNR. The performance of the wavelet transform, as a subband coder that splits each level for three directions (horizontal, vertical, and diagonal), implies that shapes 1 and 2 (horizontal and vertical) should be quantized with extra care than shape 3 (diagonal). In the lower levels, error correction is used to improve the quality of the reconstructed coefficients. The coarser resolution (shape 0) is kept unquantized. This group of coefficients contains most of the energy of the image (see Table III). Quantization errors resulted when reconstructing these coefficients yield major disturbances in the image. Using VQ on the coefficients enables us to control the total compression ratio by altering N and K for each shape.

In the following section, we explain the idea of how to determine the length assigned to N and K along the wavelet decomposition.

A. Allocation Sizes and Vector Lengths

Let N be the number of vectors in the quantization table and K be its length. In order to reach a final predetermined compression ratio, different allocations (resulting in different N and K) were tested while coarser levels were assigned greater allocations and achieved a satisfactory SNR. After the bit allocations in each level was determined, different lengths of N, K for these allocations (according to $N = 2^{KR}$) were tested. The number of total coefficients $(N \times K)$ in the table should not exceed a bound that is level dependent; otherwise, the computational complexity will be overwhelming. Thus, this work uses N and K that achieved SNR of 12. If the shape is important for the final reconstruction, EC is applied, and thus, SNR of 20 is achieved without including longer vectors in the table codebook.

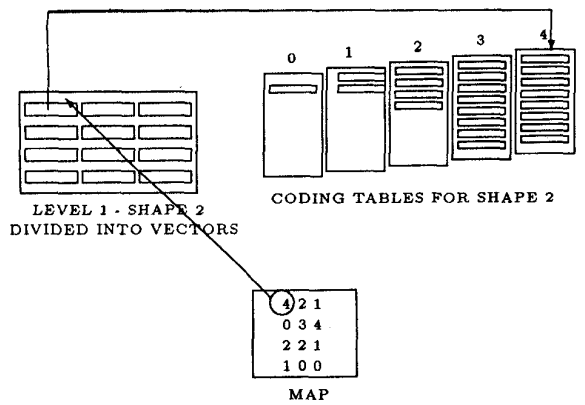


Fig. 11. Using a map file, the first vector in shape 2 in level 1 should be compressed by searching for the closest vector in Table Code number 4.

TABLE X
DIFFERENT ALLOCATIONS USING SEVERAL TABLES FOR LEVEL 3, SHAPE 2

R	K	*tablenumber*	SNR	bytes
0.25	8	8	3.13	171
0.25	8	9	3.31	187
0.25	8	10	3.31	198
0.25	8	11	3.37	197
0.25	8	12	3.43	180
0.5	8	8	3.69	205
0.5	8	9	3.94	227
0.5	8	10	3.99	241
0.5	8	11	4.16	249
0.5	8	12	4.19	239
1.0	8	8	7.35	260
1.0	8	9	8.10	301
1.0	8	10	8.88	333
1.0	8	11	8.70	354
1.0	8	12	8.56	360

IV. VECTOR BIT ALLOCATION USING SEVERAL TABLES FOR EACH SHAPE

In the previous sections, each shape was associated with one table that was generated by LGB, and all the vectors for this shape were quantized based on this table. Hence, before applying the Huffman coding, binary code words of equal length were used to represent the quantizer output. Here, we propose another approach, which takes into consideration the variance and the energy of the coefficients in each vector. Since different vectors (in each shape) have different energy concentrations, they should be quantized using different codebooks of different sizes. Obviously, the vectors with higher energy should be quantized using the larger codebook, whereas the vectors with lower energy may be quantized using the smaller codebook.

After determining the total number of bits that represent the shape, several distinct lengths for the vectors are examined, as was done in the previous method, but for each one of them, we examine a different number of tables.

A. Vector Bit Allocation

Assume that R is the targeted average rate in bits per pixel for a given shape, M and N are the dimensions of the shape, and K is the length of the vectors. Then, $[(M \times N)/K] = N_V$ is the number of vectors in the shape, and $(M \times N) \times R$ is the total number of bits available to code the shape.

If $\sigma_{i,k}^2$ is the variance of the ith component of the kth transformed vector, the variance of the vector is

$$\sigma_k^2 = \sum_{i=0}^{K-1} \sigma_{i,k}^2. \tag{4}$$

Let $b_{i,k}$ be the number of bits allocated for the ith component of this vector

$$b_{i,k} = R + 0.5\left(\log_2 \sigma_{i,k}^2 - \frac{1}{N \times M} \sum_{k=0}^{N_V-1} \sum_{i=0}^{K-1} \log_2 \sigma_{i,k}^2\right) \tag{5}$$

then the total number of bits b_k allocated for this vector is

$$b_k = \sum_{i=0}^{K-1} b_{i,k} = K \times R + 0.5 \times K \times \left(\log_2 \sigma_k^2 - \frac{1}{N_V} \sum_{k=0}^{N_V-1} \log_2 \sigma_k^2\right). \tag{6}$$

After the codebook size for each k in each shape has been determined, the LGB algorithm is used again to generate a new codebook. The vectors, which belong to the same codebook (and have the same b_k), are trained together, and a code table of size 2^{b_k} is created. During the compression, the map file associates each vector with its quantization table. This process is demonstrated by Fig. 11.

B. Implementation of the Vector Bit Allocation

In the implementation of vector bit allocation, a set of look-up tables are used to reproduce the quantized vectors. Several parameters are checked: R is the final predetermined average compression rate, K is the length of the vectors in the given shape, and *tables number* is the number of tables that are accompanied with each shape. The quality of the results is evaluated according to the values of the SNR of the reconstructed coefficients, the resulting compression after

TABLE XI
FINAL ALLOCATIONS WHILE CREATING GLOBAL TABLES

LEVEL	*SHAPE*	R	K	*tablesnumber*	SNR
2	1	1.0	8	8	4.66
2	2	0.5	8	16	4.56
2	3	-	-	-	-
3	1	1.0	16	12	7.29
3	2	1.0	8	12	5.83
3	3	1.0	8	8	4.03

TABLE XII
RESULTS FOR RESTORING SEVERAL IMAGES. FACE 3 IS TAKEN FROM THE TRAINING SET. ALL OTHER FACES WERE TAKEN OUTSIDE THE TRAINING SET.

IMAGE	PSNR	ERMS	Big Err.	Max Diff.	Min Diff.	Compression bpp	Compression CR
FACE 1	35.47	4.29	610	109	-78	0.22	36.00
FACE 2	36.49	3.81	549	85	-62	0.22	35.80
FACE 3	36.72	3.71	622	130	-63	0.22	35.23

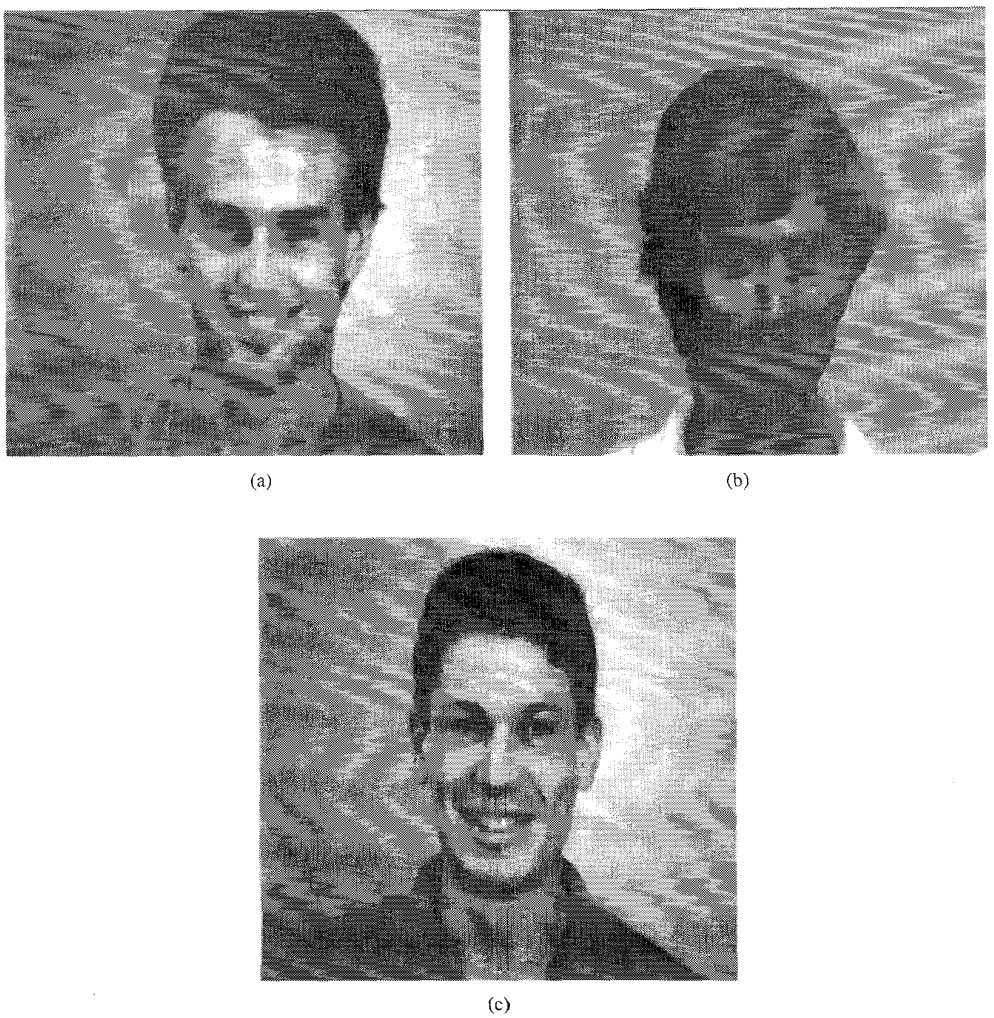


Fig. 12. Restoring faces taken (a), (b) outside and (c) inside the training set, using vector bit allocation. The compression is 0.22 b/pixel.

applying Huffman coding, and the PSNR of the reconstructed images—taken for images inside and outside the training set.

Table X displays the results for several tests applied on the coefficients of shape 2 in level 3. The parameter *table number* denotes the highest bit per pixel rate that is allocated for the most important vectors. *table number* $= n$ means that there are n tables that accompany the shape, and table number n has length of 2^n vectors and belongs to vectors that will be compressed to n b/pixel. The column *bytes* presents the number of bytes needed after the application of the Huffman coding to represent the shape. From this table, we can conclude that if we use more tables, then better quality of reconstruction is achieved, but larger tables (which belong to more important vectors) make the compression process slower since more vectors should be compared with in order to locate the index of the closest vector in the table. In addition, one can see that achieving a similar number of bytes is accompanied with achieving similar values of SNR that are not in relation to the allocations of R and K and *table number*. Hence, using the bit-allocation formula enables to use shorter allocations and fewer number of tables—without decreasing the resulted quality and compression. Observing the results of the other shapes yields similar conclusions.

C. Allocating Global Tables

Table XI brings the final allocations that were used while creating the training tables. The SNR column gives the quality of the reconstructed coefficients of an image taken outside the training set. EC is used in shape 1 of level 3. Table XII shows the final results of compressing images of faces taken within and outside the training set. Our new method, in addition to being faster, needs smaller tables and allocations and achieves finer quality results. The results are shown in Fig. 12.

V. Conclusion

In this paper, we have developed an image coding scheme combining the wavelet transform and the use of the LGB VQ. Applying the compression method on black-and-white images within the training set yielded a compression ratio of 60–65, and the PSNR was usually above 30. Using similar methods

on images of faces and decompressed faces taken outside the training sequence resulted in a compression ratio of 38–40 and a PSNR of 30–36. The compression, using vector bit allocation, produced similar results with lower computational cost.

REFERENCES

[1] N. M. Nasrabadi and R. A. King, "Image coding using vector quantization: A review," *IEEE Trans. Commun.*, vol. 36, no. 8, Aug. 1988.

[2] M. Barlaud, P. Mathieu, and M. Antonini, "Wavelet transform image coding using vector quantization," in *Proc. 6th Workshop MDSP*, Monterey CA, Sept. 1989.

[3] M. Antonini, M. Barlaud, P. Mathieu, and I. Daubechies, "Image coding using wavelet transform," *IEEE Trans. Image Processing*, vol. 1, no. 2, pp. 205–220, Apr. 1992.

[4] J. C. Feauveau, P. Mathieu, M. Barlaud, and M. Antonini, "Recursive biorthogonal wavelet transform for image coding," in *Proc. IEEE ICASSP 1991*, pp. 2649–2652.

[5] I. Daubechies, "Orthonormal bases of compactly supported wavelets," *Comm. Pure Appl. Math.*, vol. 41, pp. 909–996, 1988.

[6] A. Cohen, I. Daubechies, and J. C. Feauveau, "Biorthogonal bases of compactly supported wavelets," Tech. Rep. TM11217-900529-07, AT&T Bell Labs.

[7] M. Vetterli and C. Herley, "Wavelets and filter banks: Relationships and new results," in *Proc. IEEE ICASSP*, Albuquerque, NM, USA, Apr. 1990.

[8] P. H. Westerink, D. E. Boekee, J. Biemond, and J. W. Woods, "Subband coding of image using vector quantization," *IEEE Trans. Commun.*, vol. 36, pp. 713–719, 1988.

[9] ——, "Progressive transmission of images using subband coding," in *Proc. IEEE ICASSP 89*, pp. 1811–1814.

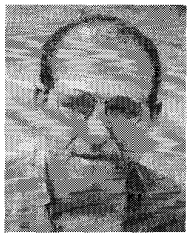
[10] P. H. Westerink, J. Biemond, and D. E. Boekee, *Subband Coding of Color Images*.

[11] S. Mallat, "A theory for multiresolution signal decomposition: The wavelet representation," *IEEE Trans. Patt. Anal. Machine Intell.*, vol. 11, no. 7, pp. 674–693, July 1989.

[12] Y. Linde, A. Buzo, and R. M. Gray, "An algorithm for vector quantizer design," *IEEE Trans. Commun.*, vol. COM-28, no. 1, pp. 84–95, Jan. 1980.

[13] J. Max, "Quantizing for minimum distortion," *IEEE Trans. Inform. Theory*, vol. IT-6, Mar. 1960.

[14] A. Averbuch, D. Lazar, and M. Israeli, "Image compression using wavelet transform and multiresolution decomposition," *DCC*, pp. 459, Mar. 1993.



Amir Averbuch was born in Tel-Aviv, Israel. He received the B.Sc. and M.Sc. degrees in mathematics from the Hebrew University, Jerusalem, Israel, in 1971 and 1975, respectively. He received the Ph.D. degree in computer science from Columbia University, New York, NY, USA, in 1982.

During 1976–1986, he was a research staff member at the Department of Computer Science at the IBM T. J. Watson Research Center, Yorktown Heights, NY USA. Since October 1987, he has been an assistant professor in the Department of Computer Science, School of Mathematical Sciences, Tel-Aviv University. His research interests include signal/image and parallel processing.

Danny Lazar was born in 1965 in Tel-Aviv, Israel. He received the B.Sc. and the M.Sc. degrees in mathematics and computer science from Tel-Aviv University, Tel-Aviv, Israel, in 1987 and 1993, respectively. He is currently pursuing the Ph.D. degree at Tel-Aviv University.

He is a research/teaching assistant in the areas of computer graphics and computer languages. He was a member of the university laboratory staff in computer graphics and worked as a system programmer for IBM Mainframe and in artificial intelligence projects. His current research activities include image processing, video coding and transmission, and multimedia communications.

Moshe Israeli was born in Bulgaria. He received the B.Sc. degree (suma cum laude) and the M.Sc. degree (with distinction) in aeronatical engineering from the Technion—Israel Institute of Technology, Haifa, Israel. He received the Ph.D. degree in applied mathematics from the Massachusetts Institute of Technology (MIT), Cambridge, USA, in 1971.

During 1971–1973 and 1977–1978, he was at MIT. During 1985–1987, he was with Princeton University, Princeton, NJ, USA. He has been a consultant to RAFAEL, Haifa, Israel, Flow Research Inc., Seattle, WA, USA, and Cambridge Hydrodynamics Inc., Princeton, NJ, USA. Since October 1987, he has held the Callner-Miller Chair of Computer Science at the Technion—Israel Institute of Technology.