ELSEVIER

Pattern Recognition Letters 18 (1997) 1253–1259

Pattern Recognition Letters

Image compression and encryption using tree structures [1]

Xiaobo Li *, Jason Knipe, Howard Cheng

Computing Science Department, University of Alberta, Edmonton, Alberta, Canada T6G 2H1

Abstract

This paper outlines our project on image coding for mobile wireless environments. The mobile, or even portable, system has a limited computation power, but deals with a large quantity of image data and a public-accessible transmission task. Thus the system must be simple, and perform compression and encryption at the same time. We investigate hierarchical data structures and related algorithms. The preliminary results, both theoretical and experimental, are encouraging. © 1997 Elsevier Science B.V.

Keywords: Image compression; Data encryption; Quadtree; Wavelet; Vector quantization

1. Introduction

The basic objective of this project is the development of an efficient and secure image storage and transmission system for mobile wireless environments. The research deals with image compression and encryption methods. In a wireless system, the radio transmission is open to public access. However, a high degree of security is required in many applications and this may be addressed through encryption. The low communications bandwidth and the large quantity of image data transmissions demand a high compression ratio. With a mobile system, compression and encryption must be handled by limited hardware power and limited storage space, thus the *simplicity* and *efficiency* of our system are also crucial. The fundamental requirement of our project is to develop a system which

- is simple (in terms of hardware and software);
- is efficient (in terms of time and space);
- performs image compression, and
- performs data encryption.

We have investigated several image compression algorithms and focused on two schemes using tree structures. A spatial domain method is a modified version of an existing simple quadtree algorithm. A transform domain method combines wavelet transform and lattice vector quantization. These algorithms show certain advantages over previous methods while remaining simple. We also studied a partial encryption scheme based on these tree data structures involved in compression. Experimental results are outlined.

2. Spatial domain compression method: Quadtree

Quadtree image compression is known as one of the more computationally simple compression algo-

* Corresponding author.

[1] This research is supported in part by Motorola Wireless Data Group and the Canadian Natural Sciences and Engineering Research Council under Grant OGP9198.

0167-8655/97/$17.00 © 1997 Elsevier Science B.V. All rights reserved.
PII S0167-8655(97)00099-8

rithms in existence (Clarke, 1995). Strobach describes a very successful compression method based on the quadtree (Strobach, 1991). Another method, developed by Shusterman and Feder (1994), was particularly appealing because of the ease with which it could be modified to test possible combinations with other compression methods, as well as its computational simplicity.

Instead of using a 3×3 reconstruction filter (Strobach, 1991; Shusterman and Feder, 1994), we use a 2×2 filter, which involves only the immediate neighbors, as the other values may influence the result in a negative way. For example, to determine a value for the north-west child-node (black) of the center, we only take the value of the parent node, as well as the north-west, north, and west neighbors (gray), as shown in Fig. 1.

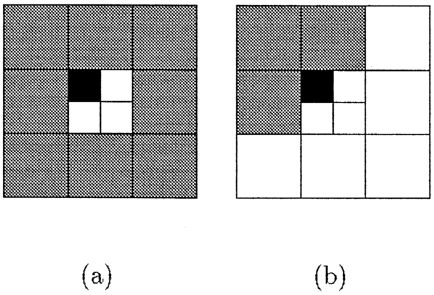

Fig. 1. Filter construction. The black box shows the child leaf to be reconstructed, and the gray boxes show the neighbor leaves influencing the reconstruction result.

We conducted a systematic search and found the following ''optimal'' (in terms of mean squared error for the testing images) filters for different levels of the reconstruction process:

$$F_1 = \begin{bmatrix} 0.0133 & 0.0887 \\ 0.0887 & 0.8093 \end{bmatrix},$$

$$F_2 = \begin{bmatrix} 0.0040 & 0.0784 \\ 0.0784 & 0.8391 \end{bmatrix},$$

$$F_3 = \begin{bmatrix} 0.0186 & 0.0375 \\ 0.0375 & 0.9064 \end{bmatrix},$$

$$F_{4+} = \begin{bmatrix} 0.0000 & 0.0000 \\ 0.0000 & 1.0000 \end{bmatrix}.$$

A single filter for all levels is also derived:

$$F' = \begin{bmatrix} 0.00 & 0.13 \\ 0.13 & 0.74 \end{bmatrix}.$$

The multiple filter gives reconstructed images with slightly higher Peak Signal to Noise Ratio (PSNR) values, but the single filter gives better visual quality.

We also employ a ''reconstruction threshold''. If the difference between a leaf and its neighbors is above a certain threshold, the filter is not used. This simple test effectively avoids ''bleeding'' of dark regions onto light regions, and vice versa.

Fourteen test images of different types (outdoor scene, indoor scene, portrait, car, building, etc.) are used. Some of the test images are given in Fig. 2. More detailed results can be seen in (Knipe and Li, 1996, Knipe and Li, in review).

Fig. 3(a) and (b) report the comparison between our methods and the original quadtree algorithm ''SF'' (Shusterman and Feder, 1994) on two images. Our methods (''MF'' for multiple filters and ''SR'' for single filter) give slightly higher PSNR for all compression ratios. Fig. 3(c) and (d) report the comparison between our method and JPEG on those two images. At compression rates of 0.2 bit-per-pixel (bpp) or below (less than 0.2 bpp), our method outperforms (in terms of PSNR) JPEG for image ''Lena''. For image ''Io'', the intersection point is around 0.33 bpp. For most test images, the intersection point is between 0.25 bpp and 0.31 bpp, except for two outdoor scene images, the intersection points are 2.26 and 3.20, where our method is obviously better for a large range.

3. Transform domain compression method: wavelet

Table 1 gives the approximate number of operations required by four algorithms: JPEG, ''RPR'' (Strobach, 1991), ''SF'' and ''SR'', and shows that ''SR'' requires less than half as many operations as JPEG in the compression phase.

Wavelet compression schemes have become powerful image compression methods. The EZW method (Shapiro, 1993) and SPIHT (Said and Pearlman, to appear) utilize prediction of wavelet coefficient significance to effectively and efficiently compress images. We propose another compression scheme which

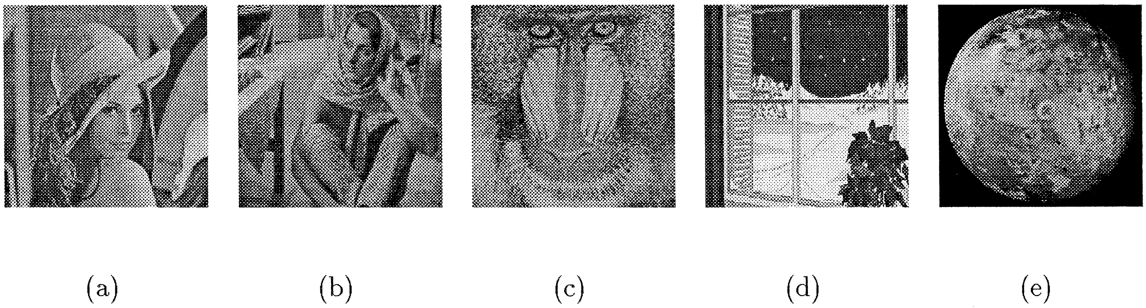

Fig. 2. Some test images: Lena, Barbara, Mandrill, Window, Io.

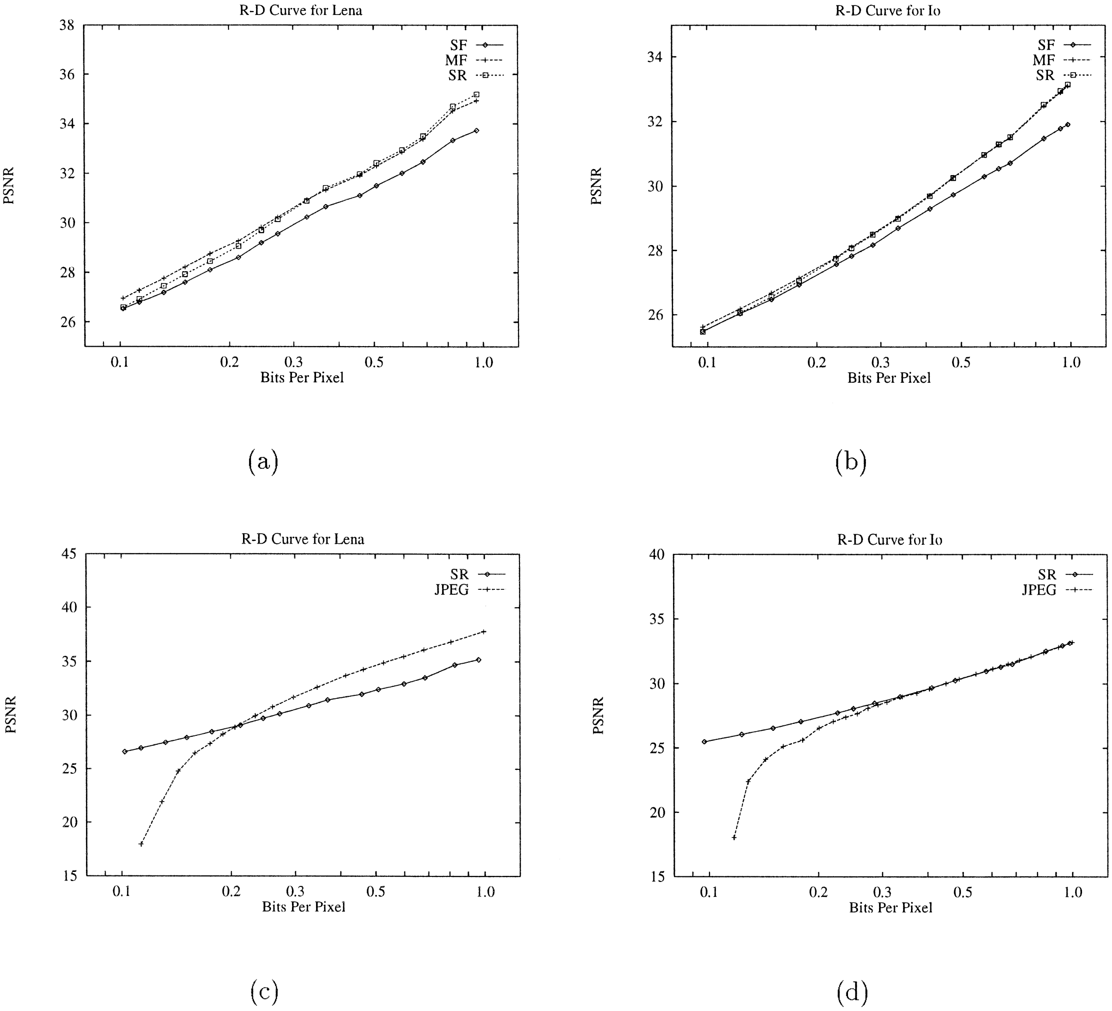

Fig. 3. Comparison with ‘‘SF’’ and JPEG.

Table 1
Average number of operations

	JPEG	RPD	SF	SR
	1988	1991	1994	1996
Compression				
+	7.25	8.00	1.48	1.48
×	1.25	3.33	0.03	0.03
/	1.00	0	0	0
Decompression				
+	7.25	–	10.63	6.69
×	2.25	–	6.97	2.99

is based on SPIHT and includes the following modifications: a modified lattice vector code book, a more efficient coder, effective timing of the vector coding, and elimination of a minor inefficiency in the original algorithm.

The SPIHT algorithm is based on a hierarchical structure which considers single coefficient elements. This structure can be easily extended to handle multidimensional elements, assuming that the elements are *square*. That is, the algorithm can accommodate the use of 2 × 2 or 4 × 4 blocks (or vectors) of coefficients. The use of the 16-dimensional Barness–Wall lattice (Λ_{16}), based in part on the Reed–Muller code (Conway and Sloane, 1988; MacWilliams and Sloane, 1978), produces the best results, and will be the lattice of choice for our new algorithm.

We added additional lattice and non-lattice vectors to the lattice code book. Vectors will consist of a single *dominant component* which causes the vector to become significant. Observation of the coefficient vectors confirms this hypothesis. In such a case, the best vector form which would match this would be $\langle \pm 1^1, 0^{15} \rangle$ and all 32 permutations for 16-dimensional vectors.

The final form of the new Λ'_{16} vectors include 32 vectors of form $\langle \pm 1^1, 0^{15} \rangle$, 480 vectors of form $\frac{1}{\sqrt{2}} \langle \pm 2^2, 0^{14} \rangle$, and 7680 vectors of form $\langle \pm 1^8, 0^8 \rangle$. In total, using this new scheme, Λ'_{16} contains 8192 vectors.

In order to refine wavelet coefficients, one should minimize the residual by coding a vector at the correct time. It may be desirable to code a given vector at a different pass than it otherwise would be

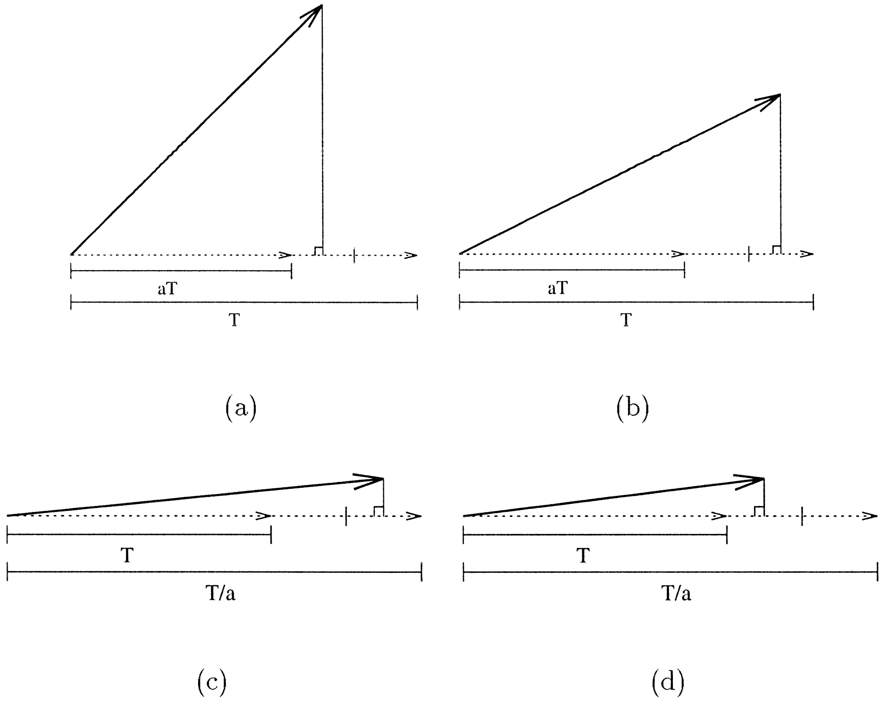

Fig. 4. Different cases of vector matching.

Table 2
PSNR for two test images

Bits per pixel	Lena image		Barbara image	
	SPIHT	MSPIHT	SPIHT	MSPIHT
1.0	40.42	40.47	37.45	37.52
0.5	37.22	37.02	32.10	32.50
0.25	34.12	33.99	28.13	28.95
0.125	31.10	30.81	25.37	26.16
0.0625	28.38	28.20	23.77	24.20
0.03125	25.97	25.92	22.71	22.79
0.015625	23.97	24.01	21.79	21.67
0.0078125	22.08	22.17	20.62	20.62

coded: (1) In the case of *short projection* (when an input vector is poorly matched by a code vector, see Fig. 4(a) and (b)), the residual would become smaller if the vector were coded using the same code vector in the *next* refinement. (2) In the case of *long projection* (if an input vector is very well matched by a code vector, see Fig. 4(c) and (d)), the residual would become smaller if the vector were coded using the same code vector in the *previous* refinement. Therefore, if the projection of the initial vector onto the code vector is greater in length than the midpoint of aT and T, it should be coded at the current threshold. Otherwise, the coding is postponed until a subsequent pass with a smaller threshold.

The experimental results are encouraging, both numerically and subjectively, especially on busy images (which are more difficult to compress) and complex portions of images (Knipe, 1996; Knipe and Li, in review). A comparison of the reconstructed image quality of SPIHT and our modified version, say MSPIHT, for the Lena and Barbara images is shown in Table 2, where the values in the table are PSNR in dB. Note that the Lena image is relatively smooth while the Barbara image contains considerably more detail and is more difficult to compress (by any algorithm).

4. Partial encryption

To protect sensitive data in wireless communication, we propose a partial encryption scheme which can be easily implemented on a mobile or even portable system. The image is divided into two parts: a small but crucial part is encrypted, and the remaining part can be sent without encryption.

In situations where encryption keys must be exchanged between the sender and the recipient, the crucial part can be encrypted directly by a public-key algorithm such as RSA (Rivest et al., 1978). Secret-key encryption algorithms such as IDEA (Lai, 1992) are not required, simplifying both the implementa-

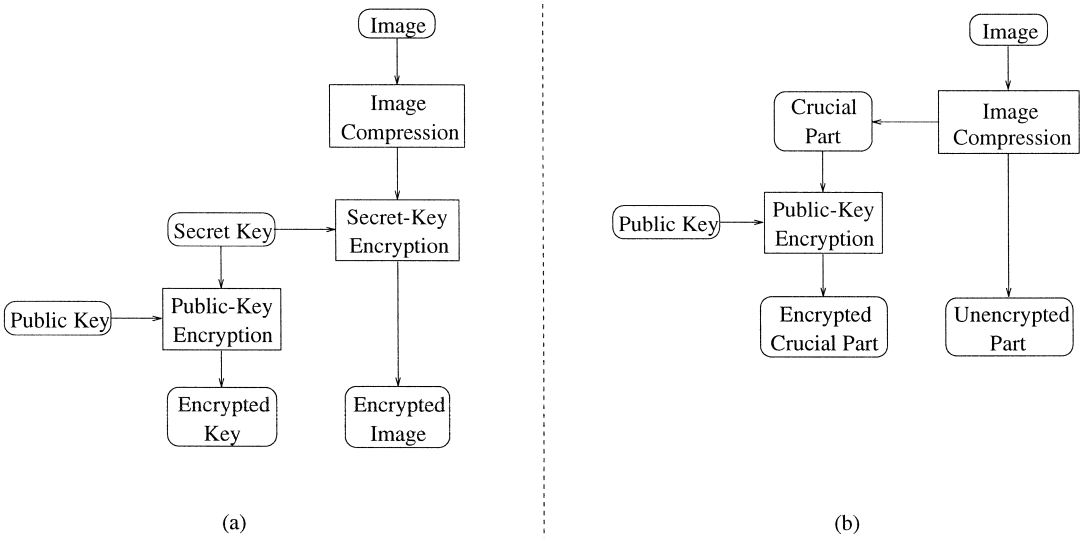

Fig. 5. A comparison between (a) the traditional approach to image compression and encryption and (b) our new approach.

tion and the operation of the communication network. This is illustrated in Fig. 5. The crucial part takes the role of the secret key in the traditional approach. Existing network facilities can be used without the additional cost for the implementation of cryptographic protocols.

We examined the security issue of partial encryption for different compression schemes, and the results indicate:

1. Spatial domain decomposition (quadtree) compression can divide an image into two parts with the tree structure being less than 20% of the entire data volume. Encrypting only the tree structure is secure, which is straightforward. Brute force attacks are infeasible.
2. In a frequency domain decomposition (e.g., DCT) compression method, encrypting only low frequency coefficients is not secure.
3. In wavelet based compression methods, such as SPIHT, tree structures are used to identify the location of significant coefficients. The trees are not transmitted explicitly, but are determined implicitly by several lists available to both the encoder and the decoder. The states of the lists must be identical in both the encoder and the decoder. If the initial changes to the lists are hidden, the decoder cannot determine the correct states of the lists to reconstruct the image. Only the significance of the coefficients, as well as the significance of their descendant sets, of the two highest levels of the pyramid are stored in the crucial file, which will be encrypted.

Table 3 gives the size of the crucial file for some test images. The details of the use of these files are given in (Cheng and Li, in review).

Table 3
The sizes of the crucial files produced by the modified SPIHT algorithm at various bit rates

Image	Crucial file					
	0.80 bpp		0.60 bpp		0.40 bpp	
	Bits	%	Bits	%	Bits	%
Lena (512×512)	1674	0.80%	1671	1.06%	1661	1.58%
Io (512×512)	1534	0.73%	1528	0.97%	1524	1.45%
Cameraman (256×256)	1586	3.03%	1558	3.96%	1491	5.69%
Bay (256×256)	1505	2.87%	1467	3.73%	1386	5.29%

5. Conclusions

In summary, we have investigated modifications to two image compression schemes. One scheme uses a quadtree to decompose the image in the spatial domain. Our modifications include the use of a simple 2 × 2 reconstruction filter and a reconstruction threshold. This algorithm is simple and outperforms the original quadtree algorithm (Shusterman and Feder, 1994). Its performance is similar to JPEG, but requires about half as many operations as JPEG in the compression phase. The second scheme MSPIHT uses a wavelet transform to decompose the image in the transform domain and is a modification of the SPIHT algorithm. Lattice vector quantization is used, and the coding *time* is carefully determined. MSPIHT works better than SPIHT for busy images which are generally harder to compress. Both compression schemes use tree data structures. We investigated a partial encryption method which takes advantage of the tree structure and simplifies, or even eliminates, the need for secret-key encryption. These algorithms provide a useful alternative for secure image transmission in wireless mobile environments.

Discussion

Kamel: I just want to get some understanding of the components of your compression techniques. You have the quadtree and then you do the transformation using wavelets?

Li: No, they are two separate algorithms. You either do this or that, not both.

Kamel: How do you do the coding itself?

Li: The quadtree coding is designed for a very simple hand-held portable unit. It is, however, not nearly as powerful as the wavelet method.

Kamel: You compared with the SPIHT algorithm. In fact there are two versions: A and B. Which of the two versions did you use?

Li: The first version is the only version we have.

Kamel: But the second version (version B) is much faster.

Li: We did not calculate how fast our method is as compared to SPIHT.

Kamel: There is also an improved version of JPEG. Is your comparison again with the basic JPEG algorithm, or with the improved one?

Li: With the basic version. We have not tried the improved version yet. I do not know how much it is improved. Because we think that the wavelet method has a much better performance than JPEG, we do not spend too much energy on JPEG comparisons.

Kanal: Many people are using wavelets. So you're saying basically that the only difference is that you have an LVQ quantization added to the wavelet?

Li: Yes, only the vector quantization part of it changed.

References

Cheng, H., Li, X., in review. Image compression and partial encryption based on tree structures. IEEE Trans. Commun.

Clarke, R.J. Digital Compression of Still Images and Video. Academic Press, New York.

Conway, J.H., Sloane, N.J.A. Sphere Packings, Lattices and Groups. Springer, Berlin.

Knipe, J. Improved spatial and transform domain compression schemes. Master thesis, University of Alberta.

Knipe, J., Li, X., in review. On the reconstruction of quadtree data. IEEE Trans. Image Process.

Knipe, J., Li, X. A new quadtree decomposition reconstruction method. In: Proc. 13th Internat. Conf. on Pattern Recognition, pp. B364–B369.

Knipe, J., Li, X., Han, B., in review. An improved lattice vector quantization based scheme for wavelet compression. IEEE Trans. Signal Process.

Lai, X. On the Design and Security of Block Ciphers. ETH Series in Information Processing, vol. 1. Hartung-Gorre Verlag, Konstanz.

MacWilliams, F.J., Sloane, N.J.A., 1978. The Theory of Error Correcting Codes. North-Holland, AMsterdam.

Rivest, R., Shamir, A., Adleman, L., 1978. A method for obtaining digitial signatures and public key cryptosystems. Comm. ACM 21 (2), 120–126.

Said, A., Pearlman, W.A., to appear. A new fast and efficient image codec based on set partitioning in hierarchical trees. IEEE Trans. Circuits and Systems for Video Tech.

Shapiro, J.M., 1993. Embedded image coding using zerotrees of wavelet coefficients. IEEE Trans. Signal Process. 41 (12), 3445–3462.

Shusterman, E., Feder, M., 1994. Image compression via improved quadtree decomposition. IEEE Trans. Image Process. 3 (2), 207–215.

Strobach, P., 1991. Quadtree-structured recursive plane decomposition coding of images. IEEE Trans. Signal Process. 39 (6), 1380–1397.