Machine Intelligence Research Labs
MIR Labs Global Network for Innovation and Research Excellence

Compression Mechanism for Multimedia System in consideration of Information Security

Nileshsingh V. Thakur[1], Dr. O. G. Kakde[2]
[1]*Department of Computer Science and Engineering,* [2]*Department of Computer Science,*
[1]*Shri Ramdeobaba Kamla Nehru Engineering College,* [2]*Visvesvaraya National Institute of Technology, Nagpur, India*
thakurnisvis@rediffmail.com, ogkakde@cse.vnit.ac.in

Abstract

This paper presents the compression mechanism based on the fractal coding and spiral architecture for the color image data of multimedia applications or system. This compression mechanism can be used prior to the encryption in multimedia system security mechanism. Proposed mechanism falls in lossy compression category. Basically, the fractal coding is applied on the grey level images and encoding time is very high. In proposed mechanism, construction of one composite image and use of spiral architecture accelerates the encoding process in comparison with the use of three grey planes of RGB color image with the square architecture. Experimental results show that the encoding time, compression ratio and peak signal to noise ratio varies with the quadtree partitioning tolerance value. Using this compression mechanism, different encryption mechanism can be developed.

1. Introduction

With the development of multimedia technology and Internet technology, multimedia data are used more and more widely in human's life. Multimedia applications include the huge image data with critical risk of security, e.g. advertisements, pictures, maps, and the sensitive images related to politics or commerce, such as military maps, medical images or multi-resolution images, should be protected in order to avoid unauthorized users knowing it. Use of multimedia applications opened the new research domain where we can combine the image processing and security techniques. Security becomes a mandatory component of commercial multimedia applications providing access to images through public channels. Typical security mechanisms required by such applications include compression, encryption, digital signatures and fingerprinting.

This paper presents the compression mechanism which can be used prior to the encryption of the multimedia applications or systems color image data. Proposed mechanism falls in lossy compression methods. First, the composite grey level image is formed from the color image using homogeneously correlated blocks and then it is represented in the spiral architecture to accelerate the encoding. On this spiral architecture image, fractal coding is applied to get the encoded data. Later, encoded data can be encrypted by different encryption mechanisms. This paper is organized as follows: Section 2 discusses the possible multimedia system security mechanism. Section 3 discusses the key issues in fractal coding with related work. Representation of square architecture image in spiral architecture is explained in section 4. Section 5 presents the fractal encoding and decoding algorithms for color image with spiral architecture. Experimental setup and results are summarized in the section 6. Discussion and conclusion with future scope is presented in section 7.

2. Multimedia Systems

In multimedia systems, huge image data is available. Recently security has become one of the most significant and challenging problems for spreading multimedia technology.

Secured image data transmission is the open challenge. The following security requirements are essential for multimedia systems [1]. *Confidentiality:* Cipher systems are used to keep information secret from unauthorized entities; *Data integrity:* The alteration of data can be detected by means of one-way hash functions, message authentication codes, digital signatures (especially content-based digital signatures), fragile digital watermarking, and robust digital watermarking; *Data origin authenticity:* Message authentication codes, digital signatures, fragile digital watermarking, and robust digital watermarking enable the proof of origin; *Entity authenticity:* Entities taking part in a communication can be proven by authentication protocols. These protocols ensure that an entity is the one it claims to be. Numerous security mechanisms are reported in literature. Mainly are the digital watermarking and cryptography. The security measures mentioned above uses digital watermarking techniques and cryptographic mechanisms.

Security mechanisms, applied for multimedia systems are based on digital watermarking techniques as well as on cryptographic mechanisms. Digital watermarking techniques based on steganographic systems offer the possibility to embed information directly into the media data. Watermarking represents an efficient technology to ensure both data integrity and data origin authenticity. Watermarking techniques usually used for digital imagery and now also used for audio and 3D models, are relatively young and their amount is growing at an exponential rate. Copyright, customer or integrity information is embedded, using a secret key, into the media data as transparent patterns. Because the security information is integrated into the media data, one cannot ensure confidentiality of the media data itself, only the security information using a secret key. The most important cryptographic mechanisms are implemented by use of keys. Basically there are two types pf cryptography: Symmetric key cryptography and Asymmetric key cryptography. In symmetric key cryptography, the message or the data is encrypted and decrypted with the same key while in asymmetric key cryptography, the message or data is encrypted with the secret key and decrypted with the other key. Same can be stated in the following way also. In private-key cryptosystems, the encryption and decryption uses the same key which must strictly be kept secret, called the secret key or private key. The size of the key space must be large enough to make it hard to find the right key. Public-key cryptosystems are based on trapdoor one-way functions. Sender and receiver hold a key-pair of public key and secret key. This pair consists of a private key and a public key corresponding to secret key. The secret key must strictly be kept secret, and the public key may be made public. Given a public key, it is computationally infeasible to find the private key or secret key if the trapdoor information is unknown.

2.1 Encryption and Compression

Security risk in multimedia applications and multimedia systems activates the research on image compression and encryption. Image encryption technique [2-3] transforms the image into an unintelligible form under the control of a key. Only the user having the correct key can recover the image. Till now, various image encryption algorithms have been reported, which can be classified into two types, i.e., raw image encryption and compressed image encryption. In raw image encryption, the image is encrypted before compression. The simplest method is image permutation [4-5] that changes the pixels' position randomly. Recently, chaos-based ciphers are often used in image encryption. For example, the cascaded chaos maps are used to construct the stream cipher for image encryption [6], the discrete Kolmogorov flow map is used to design the parallel image encryption algorithm [7], two chaotic maps are combined to shuffle the image pixels [8], nonlinear chaotic algorithm is introduced to replace linear function in image encryption [9], discrete exponential chaotic maps' confusion and diffusion properties are improved and used to design the image encryption algorithm [10], the Chaotic Neural Network is used to design the stream cipher for images [11], the 2-dimensional Baker map is used to construct the block cipher for images [12-13], the 3-dimensional chaotic maps are used to make the block ciphers for images [14-15]. These ciphers encrypt the uncompressed images directly without considering of compression.

Encryption typically increases the entropy of the symbol stream. So it is disadvantageous to apply compression after encryption. The only way is to use the techniques in a combined algorithm, examples for such combinations can be found in [16-17]. The combination of encryption and compression is addressed in [16] and in [17] especially for text data. In [16] the two steps are combined to reduce processing time by adding a pseudo-random shuffle into the data compression process. In [17] the approach is to transform text data into an intermediate form which can be compressed more efficient. They apply a strategy called Intelligent Dictionary-based Encoding for preprocessing and encryption. Normally, compression is placed before encryption, because encryption transforms its input into a sequence that is hard to compress in a subsequent step. A trustworthy encryption algorithm should generate output data that can not be well compressed [18]. In practice, images or videos are often compressed in order to save the cost of storage space or transmission loading. Thus, it is more reasonable to encrypt the compressed data. Furthermore, considering that images or videos are often of large volumes, encrypting the compressed data completely will cost much time. Thus, it will reduce the computational cost if encrypting only part of the compressed data. For example, the DCT coefficients' signs are encrypted in DCT transformed blocks [19], the data blocks are permuted in frequency domain

[20,21], and both the coefficients' signs and block positions are encrypted [22,23]. These algorithms encrypt only some parameters in the image, reduce the encrypted data volumes, and thus, improve the encryption efficiency. The key problem is how to select the parameters. Considering that different compression method produces different parameters, different encryption algorithm should be designed for different compression method.

2.2 Possible Multimedia System Security Mechanism

Possible multimedia system security mechanism with the combination of compression and encryption is depicted in figure 1. Complete security mechanism is divided in two parts. First part is the compression and second is encryption. In compression, the multimedia system or application RGB color image is represented as the one composite plane image using quadtree partitioning with trichromatic coefficients for each pixel and variances of the blocks. Then composite plane image is represented in spiral architecture on which the fractal coding is applied to get the encoded data. In encryption, the encoded data can be encrypted by the stream ciphers and then the encrypted data can be transmitted. To encrypt the encoded data, different algorithms can be developed in consideration with the proposed compression mechanism.

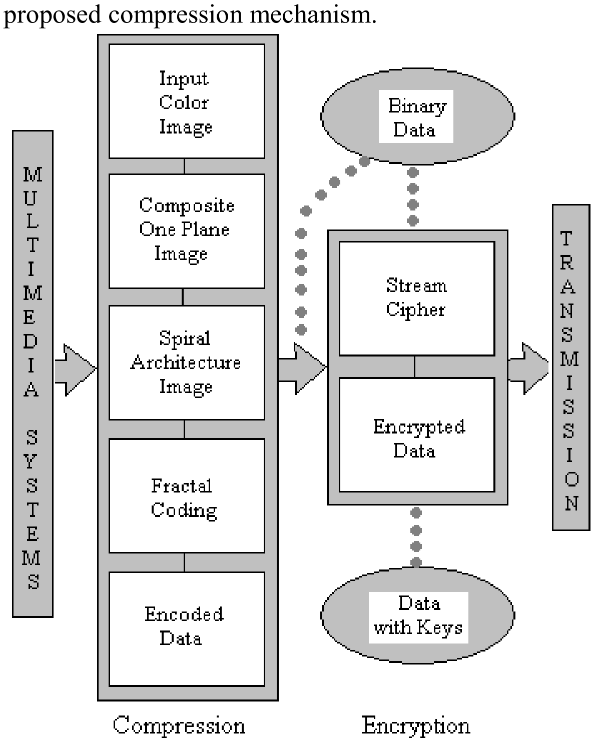


Figure 1. Possible multimedia system security mechanism

3. Fractal Coding

Recently, fractal image coding [24, 25] attracts more and more researchers, which adopts the self similarity in images to compress image data. Generally, in these compression methods, the image is partitioned into blocks, non-overlapping called range blocs and overlapping called domain blocks. According to the property of self-similarity, for each block, a fractal transformation can be obtained, which is determined by some fractal parameters. Thus, for each block, only the corresponding fractal parameters are stored instead of the block data themselves, which reduces the data size. Most of the recent works [26-28] focus on the method to obtain the suitable fractal transformation. Following are the key issues in the fractal coding: How to search the perfect domain block match for the range block? i.e. Speed Up; How to minimize the encoding time? i.e. Speed Up; How to optimize the domain pool? i.e. Speed Up; How to classify the domain pool? i.e. Speed Up.

In [29], speed up scheme for fractal image coding based on the classification methods are introduced where the domain pool is reduced which minimizes the number of operations for the similarity search. Nearest neighbor search approach is proposed in [30]. Computational complexity in the encoding step is minimized using the partitioning method [31]. In [32], a method of regional search for fractal image compression and decompression is proposed. In this method, the search for fractal codes is carried out in a region of the image instead of over the whole image. The method in [33] is particularly well suited for use with highly irregular image partitions. In this paper, the FFT-based cross correlation is used to reduce the time complexity of fractal image encoding. An improved formulation of approximate nearest neighbor search [34] based on orthogonal projection and pre quantization of the fractal transforms parameters. The approach in [35] reduces the memory requirement, and speeds up the reconstruction. The number of domain blocks searched to find the best match for each range block is reduced by eliminating the ineligible domain blocks using the law of cosines [36]. In [37], encoding complexity is reduced by minimizing the size of the domain pool based on the Entropy value of each domain block. In [38], a recursive scheme is proposed where feeding the coding results back to update domain pools during the coding process to improve the decoded image quality is discussed.

Parallel implementation of Fractal coding is discussed in [39-42]. In [39], parallel algorithms for fractal image compression on MIMD architectures are introduced, classified, and discussed. [40] introduce and analyze algorithms for fractal image compression on massively parallel SIMD arrays. [41] proposed a method for implementing fractal image compression on dynamically reconfigurable architecture. A parallel architecture for quadtree based fractal image coding is proposed in [42]. Some hybrid Fractal encoding schemes are discussed in [43-45]. In [43] proposed image coding scheme based on a unification of fractal and transform coding. The algorithm in [44] is based on a distance classification fractal coder with fixed cluster centers, decides whether to encode a range block by a cluster center or by a domain blocks. In [45], a hybrid fractal and Discrete Cosine transform (DCT) coder is developed using QT decomposition. Evolutionary and genetic approach to Fractal compression is discussed in [46]. In [47], functional programming technique in Haskell language is explained to model fractal image compression. In [48], a progressive fractal image coder in the spatial domain is proposed. Fractal image compression based on the theory of iterated function system (IFS) with probabilities is explained in [49]. A tree topology based approach is presented in [50] for parallel implementation of Fractal coding.

Some of the gray level Fractal image compression techniques are- Barthel et al. [51] shown that the coding performance can be greatly improved by applying an aliasing free codebook design, an enhanced luminance transformation combined with a vector quantization, and an adaptive geometrical search scheme. In [52], Barthel et al. described a generalization of the luminance transformation in the frequency domain where Fractal and transform coding are subsets of this transformation. Texture Based Image Compression [53], where the image compression carried out on the basis of texture, a block based approach consist of Fractal Compression by DCT is given in [54], Hybrid Wavelet-Fractal Coder [55] is a combination of the adaptive Fractal prediction and bit plane wavelet coding where the Fractal prediction is applied to regions where the Fractal rate saving justifies its overhead. A graph-theoretic interpretation of convergence of fractal encoding based on Partial Iterated Function System (PIFS) is presented in [56]. The concept leads to the development of a linear time fast decoding algorithm from the compressed image. It has been shown in [56] that the encoded image can be modeled as a *flow graph*, where the directed edges from a pixel q to p exist if the brightness value of the pixel p is determined from q. An approach based on relative fractal coding has been proposed in [57] which found to be suitable for coding multi-band satellite image.

Attempts have also been made to extend fractal coding beyond uniform square blocks in order to adapt the coding to local image characteristics and consequently increase coding performance. This has lead to either square or rectangular non-uniform partition schemes such as quadtree [58, 59] and horizontal-vertical partitioning [60]; the usual coarse-to-fine approach starts from a maximum-size range block and performs recursive partitioning of blocks according to a quality metric. In fine-to-coarse quadtree partitioning [61] small blocks are merged into larger blocks within a quadtree. An approach to fractal image coding that permits region-based functionalities is proposed in [62] where images are coded region by region according to a previously computed segmentation map. In Fractal image compression, the encoding time required is too large. This large amount of time is due to checking the eight isometries for each edge block. Reducing the number of isometries results in distortion of the image. To overcome this difficulty, [63] used probabilistic measures and found that the use of probability is quite efficient. For the first certain number of blocks, the isometry with highest probability is chosen and the same isometry is implemented on rest of the edge blocks. The domain blocks pool is divided in the shade, midrange and edge blocks. Some of the other work is available in Quadtree-based Fractal image compression [64].

A hybrid system where only some parts of the image are encoded using fractal techniques and the remaining parts are modeled using other methods, so that the merits of different approaches can be combined and the overall coding efficiency is improved. There have been some examples for such kind of hybrid systems. Laurencot and Jacquin compared FBC with LBG [65] based VQ scheme. Their results appeared that for sharp-edge blocks, FBC performs better than VQ, but the results are very similar for texture blocks, with perhaps a slight advantage for VQ coding of texture block [66]. It should be noted that all of the above mentioned hybrid systems assume that the fractal image coding algorithm is compatible with other methods. In [67], by introducing the concept of partial fractal mapping provides some mathematical explanations on how and why a hybrid fractal image coding system can work. A general framework of the hybrid system is then proposed. According to such a framework, many different hybrid image coding schemes can be derived.

The work on Fractal coding for grey level images on square architecture is extended recently by using the spiral architecture in place of a square architecture. In literature, research work in image processing on spiral architecture is reported, particularly on the image segmentation, image rotation, image compression, object recognition. Research work on image compression based on spiral architecture in [68], focused on the properties of the hexagonal pixel address labeling scheme. The property of interest was the physical proximity of the hexagonal pixels with neighboring addresses. The research reported in [69] used the properties of uniform image partitioning based on spiral architecture. On spiral architecture, an image can be partitioned into a few sub-images [70] each of which is a scaled down near copy of the original image. Fractal image compression on spiral architecture related work is also presented in [71]. Fractal image compression on spiral architecture related work presented in literature is particularly focused the grey level image compression.

4. Square and Spiral Architecture

The unit of the 3 x 3 pixel vision in the square and spiral architecture is represented in figure 2 where each square pixel is considered as the hexagonal pixel for spiral architecture. Labeling each of the individual hexagons with a unique address is the first step in spiral architecture formulation. This is initially applied to a collection of seven hexagons. Each of these seven hexagons are labeled consecutively with addresses 0, 1, 2, 3, 4, 5 and 6 as displayed in figure 3. In the Pseudo model, one rectangular pixel represents a hexagonal pixel. To find the Pseudo pixels, the central pixel is labeled with spiral address 0 and set the Cartesian coordinates of this pixel as (0, 0). Considering the length of edge length of each square pixel is unity. Then, the hexagon with spiral addresses 1, 2, 3, 4, 5 and 6 have corresponding Cartesian coordinates (0, -1), (-1, -1), (-1, 0), (0,1), (1, 1) and (1, 0) respectively. The basic cluster of seven pixels with spiral addresses 0~6 are represented in the Figure 3. To have more number of hexagons, dilate the structure so that six additional collections of seven hexagons can be placed about the addressed hexagons, and multiply each address by 10 as shown in Figure 4 [72]. For each new collection of seven hexagons, label each of the hexagons consecutively from the centre address as we did for the first seven hexagons.

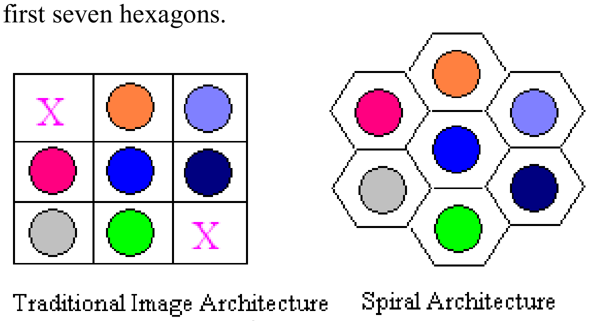


Figure 2. Unit of vision in the two image architectures

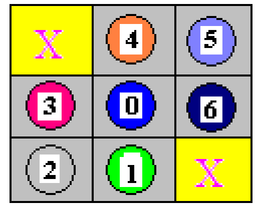


Figure 3. Distribution of 7 pixels constructed from rectangular pixels

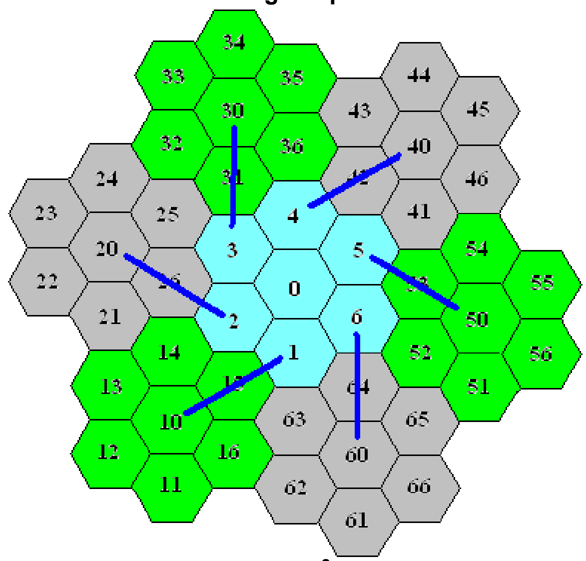


Figure 4. A collection of 7^2 = 49 hexagons with Addresses

5. Fractal Encoding and Decoding for Color Image with Spiral Architecture

In this section the fractal encoding and decoding algorithms are discussed. Fractal encoding algorithm for spiral architecture is as follows:

1. Read a color image CI with size N x N.
2. Use equation (1) to calculate the trichromatic coefficients [72] for each pixel.
3. Subdivide the image recursively by quad tree partitioning.

(a) Partition the image in 4 non-overlapped blocks with size $\frac{N}{2} \times \frac{N}{2}$. (b) For each such a block, calculate the average trichromatic coefficient and variances according to equations (2) and (3) [72]. (c) The process of division continues, if equation (4) (with a pre-defined tolerance) is not satisfied. When this occurs, the block will be divided into another four quadrant sub-blocks, and for each of them, the same steps 2), 3), is repeated again. The process of quarter halts if any one of the two conditions are satisfied: (i) The pre-defined maximum dividing level is reached; (ii) Or, equation (4) is satisfied. (d) If all of the blocks have

been processed (recursively returned), the whole process terminates.
4. One-plane image S is produced with a value of S_{ij} associated to each pixel.
5. The image S and the mean value of trichromatic coefficient ratio of each block are stored as the first part of compressed image.
6. The one-color plane image S is represented in spiral architecture to get image S'.
7. Compress S' by invoking the fractal coding algorithm, codebook blocks formed by taking median of the pixel intensities of each cluster of seven hexagons.
8. Store the indices for the quantized coefficient *s* and *o* and the index *l* identifying the optimal codebook block CB_l as the second part of compressed image.

Fractal decoding algorithm for spiral architecture is as follows:
1. Use the fractal decoding algorithm and the second part of compressed image to decode image S''.
2. By using first part of compressed image and equation (5) to get $R''_{ij}, G''_{ij},$ and B''_{ij}, where $R''_{ij} \approx R_{ij}, G''_{ij} \approx G_{ij}, B''_{ij} \approx B_{ij}$.
3. $R''_{ij}, G''_{ij},$ and B''_{ij} are used to reconstruct image CI'(spiral) where $CI \approx CI'$.

$$RR_{ij} = \frac{R_{ij}}{R_{ij} + G_{ij} + B_{ij}} \quad (1)$$

$$GG_{ij} = \frac{G_{ij}}{R_{ij} + G_{ij} + B_{ij}}$$

$$BB_{ij} = \frac{B_{ij}}{R_{ij} + G_{ij} + B_{ij}}$$

$$M_{rr}^{L} = \frac{1}{m^2}\sum_{i=L_x}^{m+L_x}\sum_{i=L_y}^{m+L_y} RR_{ij}$$

$$M_{gg}^{L} = \frac{1}{m^2}\sum_{i=L_x}^{m+L_x}\sum_{i=L_y}^{m+L_y} GG_{ij} \quad (2)$$

$$M_{bb}^{L} = \frac{1}{m^2}\sum_{i=L_x}^{m+L_x}\sum_{i=L_y}^{m+L_y} BB_{ij}$$

$$V_{rr}^{L} = \frac{1}{m^2}\sum_{i=L_x}^{m+L_x}\sum_{i=L_y}^{m+L_y} \left|RR_{ij} - M_{rr}^{L}\right|$$

$$V_{gg}^{L} = \frac{1}{m^2}\sum_{i=L_x}^{m+L_x}\sum_{i=L_y}^{m+L_y} \left|GG_{ij} - M_{gg}^{L}\right| \quad (3)$$

$$V_{bb}^{L} = \frac{1}{m^2}\sum_{i=L_x}^{m+L_x}\sum_{i=L_y}^{m+L_y} \left|BB_{ij} - M_{bb}^{L}\right|$$

$$V_{rr}^{L} \le tolerance \text{ and } V_{gg}^{L} \le tolerance \text{ and } V_{bb}^{L} \le tolerance \quad (4)$$

$$S_{ij} = R_{ij} RR_{ij} + G_{ij} GG_{ij} + B_{ij} BB_{ij} \quad (5)$$

6. Experimental Setup and Results

The proposed compression mechanism is implemented with MATLAB 6.5 and Windows operating system on the System Model P4i65GV, Intel HT, 256 MB RAM. An Uncompressed Color Image Database [73] is used for experimentation. Experimental results of the images house, airplane and sailboat are shown in the figure 6, figure 7 and figure 8. Encoding time required for the images house, airplane and sailboat in comparison with the straight fractal coding are represented in the figure 9 and figure 10 for the tolerance value 0.01 and 0.02 respectively. Compression ratio and peak signal to noise ratio with the tolerance value 0.01 and 0.02 for house, airplane and sailboat images are plotted in the figure 11 and figure 12 respectively.

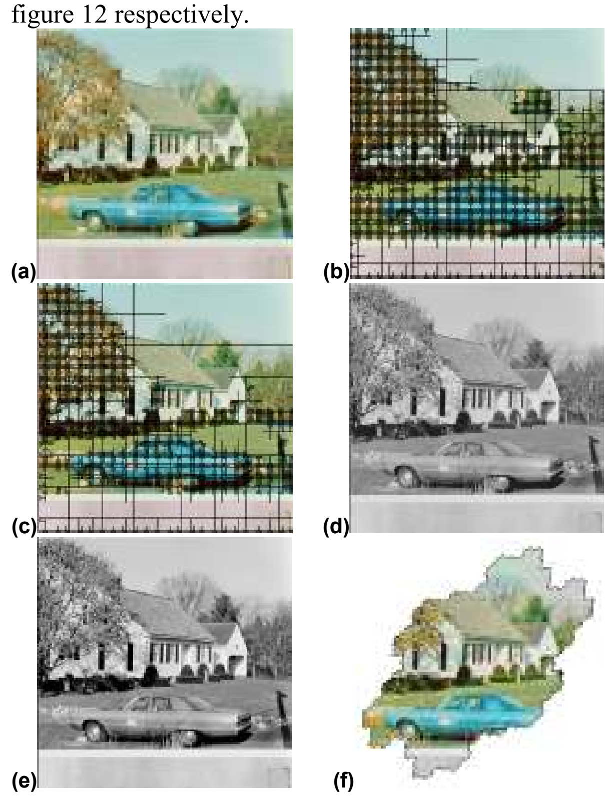
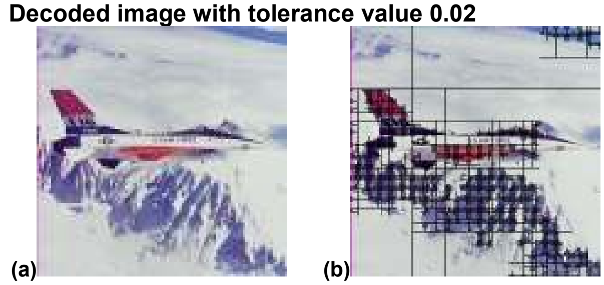


Figure 6. (a) Original House (512 x 512) image (b) and (c) image with tolerance value 0.01 and 0.02 respectively (d) and (e) One plane composite image with tolerance value 0.01 and 0.02 resply (f) Decoded image with tolerance value 0.02



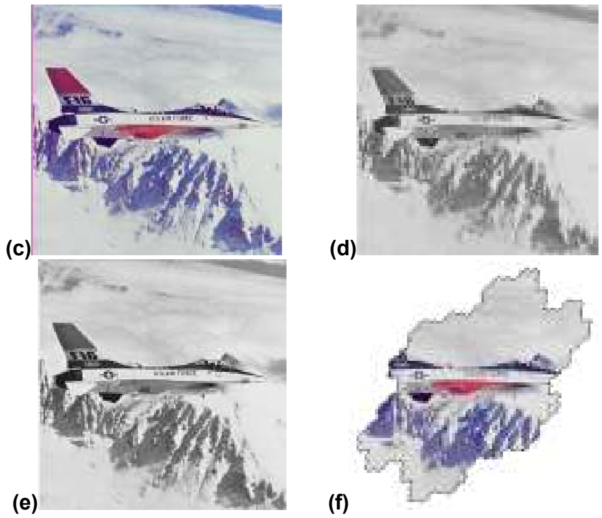


Figure 7. (a) Original Airplane (512 x 512) image (b) and (c) image with tolerance value 0.01 and 0.02 respectively (d) and (e) One plane composite image with tolerance value 0.01 and 0.02 resply (f) Decoded image with tolerance value 0.02

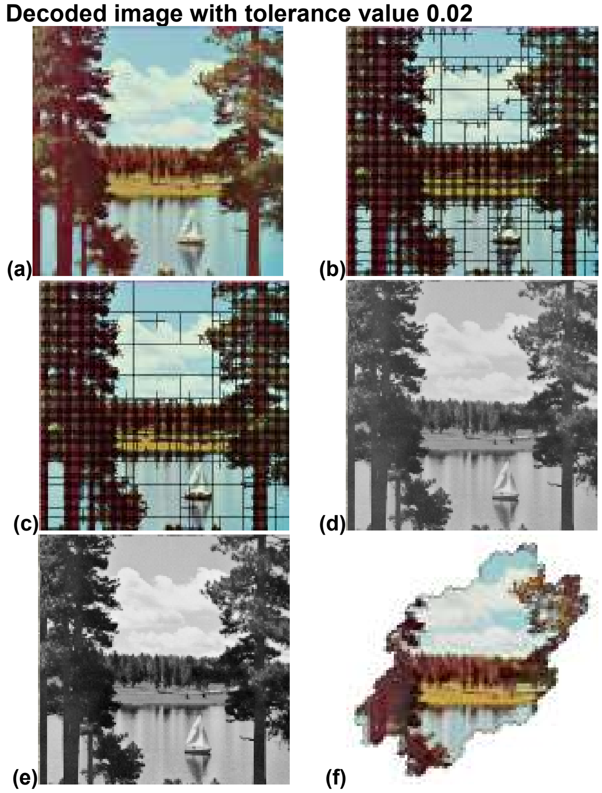


Figure 8. (a) Original Sailboat (512 x 512) image (b) and (c) image with tolerance value 0.01 and 0.02 respectively (d) and (e) One plane composite image with tolerance value 0.01 and 0.02 resply (f) Decoded image with tolerance value 0.02

Figure 9 and figure 10 justify that the proposed approach requires less encoding time in comparison with the straight fractal coding. Presently the constraint is that the whole image is not get encoded.

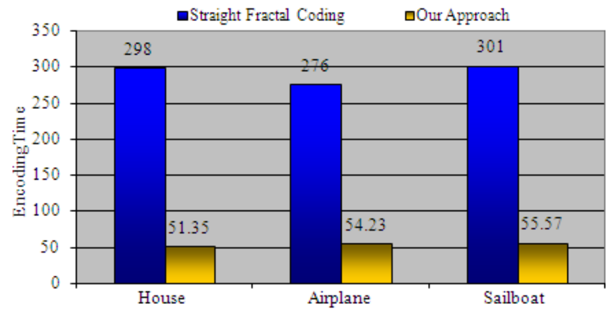


Figure 9. Encoding time required for the images house, airplane and sailboat in comparison with the straight fractal coding with tolerance value 0.01

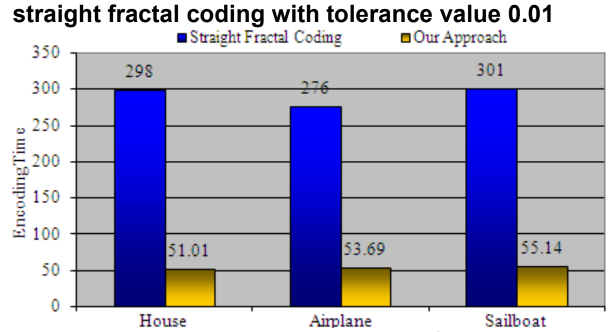


Figure 10. Encoding time required for the images house, airplane and sailboat in comparison with the straight fractal coding with tolerance value 0.02

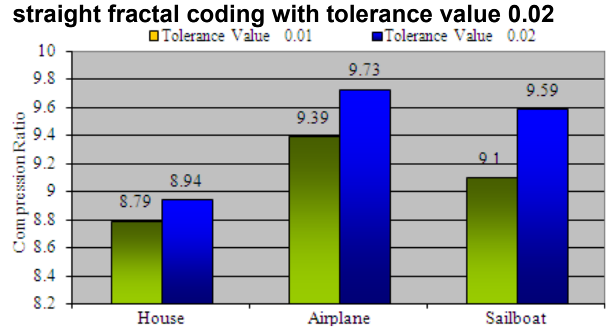


Figure 11. Compression ratio for the images house, airplane and sailboat with tolerance values 0.01 and 0.02

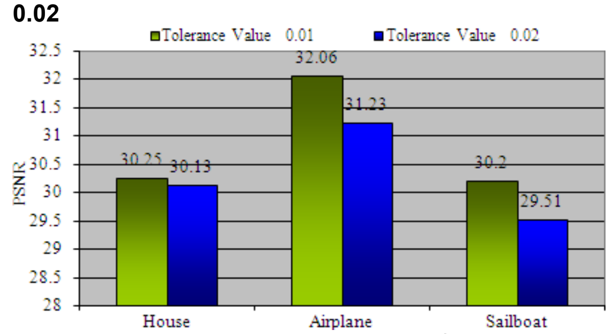


Figure 12. Peak signal to noise ratio for the images house, airplane and sailboat with tolerance values 0.01 and 0.02

Figure 11 and figure 12 present the results of the compression ratio and PSNR. It is clear that the compression ratio increases and image quality degrades as the tolerance value increases. One can go according to the requirement of image quality and size.

7. Discussion, Conclusion and Future Scope

The selection of compression method and encryption method varies in accordance with the kind of data and security policy. Encryption is counterproductive for compression if the two features aren't used together correctly. Generally, we want to compress first and then encrypt. This is the order that naturally happens when we compress at the encoding level and encrypt at the transport level. We tend to get disadvantageous results if we encrypt first and then compress. This order can happen when we encrypt early on, such as when we use message security with transport compression, or if we attempt to apply compression from outside the system after encryption has already taken place.

Proposed compression mechanism can be used prior to the encryption to save the whole processing time. As multimedia systems consist of the immense image data with the constraint of the Internet bandwidth, the proposed compression mechanism can save the encoding time and utilize the bandwidth in efficient way. Fractal coding is generally applied on the grey level images. This compression mechanism applies the fractal coding on the RGB color images without separating the three color planes and use of spiral architecture to represent the one composite plane image improvise the encoding time with little trade off in image quality and high compression ratio. Basic limitation of the compression mechanism is that no hardware is available to represent the given image into the spiral architecture so we have to simulate the spiral architecture using the square architecture.

Presented results justify that the compression ratio increases and the image quality decreases as the tolerance value increases. Though the whole image is not get compressed, but the obtained results justify the scope of improvement in encoding time. Presently we have considered that individual pixel as the hexagonal pixel. Proposed mechanism can become robust, once it is possible to represent the whole composite one plane image in spiral architecture. For this purpose, we may have to use the sub sampling so that we can able to mimic the spiral architecture on square architecture for whole image, only little distortion may be there at the boundaries of the image.

Though the encryption mechanism is not discussed in this paper, the general idea for encryption mechanism may be as follows. The ciphers are generally used for the plain text data, the encoded data created by the compression mechanism can be the input to the encryption mechanism. Encoded data is represented in bit string format, and later which can be encrypted by the known encryption algorithms. This may lead to the different encryption algorithms. This compression mechanism can be used to develop the encryption mechanism using self regressive function. Regression can also be used for the encryption. As encoded data consists of the various parameters as discussed in the section 5, partial encryption can also be possible where we will not encrypt the whole encoded data but the important part of the encoded data can be encrypted and the remaining encoded data can be passed directly. This paper provides the scope for further research in proposed compression mechanism as well as the possible encryption mechanism.

8. References

[1] J. Dittmann, P. Wohlmacher and K. Nahrstedt., “Approaches to Multimedia and Security: Cryptographic and watermarking algorithms”. Available at http://omen.cs.uni-magdeburg.de/itiamsl/cms/upload/lehre/sommer05/ACM-journal-paper-2001-1.pdf

[2] B. Furht and D. Kirovski (Ed.), Multimedia encryption and authentication techniques and applications, Boca Raton, Fla.: Auerbach Publications, First edition, May 3, 2006.

[3] S. Lian, J. Sun, D. Zhang and Z. Wang, “A Selective Image Encryption Scheme Based on JPEG2000 Codec”. 2004 Pacific-Rim Conference on Multimedia (PCM2004), Springer-Verlag Berlin Heidelberg LNCS, 3332, pp. 65-72, 2004.

[4] Y. Matias and A. Shamir, “A video scrambling technique based on space filling curves”. In proceedings of a Conference on the Theory and Applications of Cryptographic Techniques on Advances in Cryptology**,** Springer-Verlag London, UK LNCS 293, pp. 398-417, 1987.

[5] Access control system for the MAC/packet family: EUROCRYPT. European Standard EN 50094, CENELEC (European Committee for Electrotechnical Standardization), December 1992.

[6] H. S. Kwok and Wallace K. S. Tang, “A fast image encryption system based on chaotic maps with finite precision representation”, *Chaos, Solitons and Fractals*, Volume 32, Issue 4, pp. 1518-1529, May 2007.

[7] Q. Zhou, K. Wong, X. Liao, T. Xiang and Y. Hu, “Parallel image encryption algorithm based on discretized chaotic map”, *Chaos, Solitons and Fractals*, In Press, Corrected Proof, Available online 6 March 2007.

[8] T. Gao and Z. Chen, “Image encryption based on a new total shuffling algorithm”, Chaos, Solitons and Fractals, Vol. 38, No. 1, pp. 213-220, January 2007.

[9] H. Gao, Y. Zhang, S. Liang and D. Li, “A new chaotic algorithm for image encryption”, *Chaos, Solitons and Fractals*, Volume 29, Issue 2, July 2006, pp. 393-399.

[10] L. Zhang, X.g Liao and X. Wang, “An image encryption approach based on chaotic maps”, *Chaos, Solitons and Fractals*, Volume 24, Issue 3, May 2005, pp. 759-765.
[11] J. Yen and J. Guo, “A Chaotic Neural Network for Signal Encryption/Decryption and Its VLSI Architecture”, In Proceedings of 10th (Taiwan) VLSI Design/CAD Symposium, pp. 319-322, 1999.
[12] S. Lian, J. Sun and Z. Wang., “A Block Cipher Based on a Suitable Use of the Chaotic Standard Map”, *Chaos, Solitons and Fractals*, Vol. 26, Issue 1, pp. 117-129, 2005.
[13] S. Lian, J. Sun and Z. Wang, “Security Analysis of A Chaos-based Image Encryption Algorithm,” *Physica A: Statistical and Theoretical Physics*, Vol. 351, No. 2-4, 15 June 2005, pp. 645-661.
[14] G. Chen, Y. B. Mao and C. K. Chui, “A symmetric image encryption scheme based on 3D chaotic cat maps”, *Chaos, Solitons and Fractals*, 2004, 12, pp: 749-761.
[15] Y. B. Mao, G. Chen and S. G. Lian, “A novel fast image encryption scheme based on the 3D chaotic Baker map. *International Journal Bifurcat Chaos* 2004; 14(10): 3613-3624.
[16] C.-E. Wang, “Cryptography in Data Compression”, *Code- Breakers Journal, Security and Anti-Security - Attack and Defense*, 1, 2006.
[17] V. Govindan and B. Shajee-Mohan, “An Intelligent Text Data Encryption and Compression for High Speed and Secure Data Transmission over Internet”, In International Conference on Information Technology Coding and Compression, ITCC. IEEE Computer Society, April 2005.
[18] B. Schneier. *Applied Cryptography: Protocols, Algorithms, and Source Code in C.* John Wiley & Sons, Inc., New York, NY, USA, 1993.
[19] H. Cheng and X. Li, “Partial Encryption of Compressed Images and Videos”, *IEEE Transactions on Signal Processing*, Vol. 48, No. 8, Aug, 2000, p 2439-2451.
[20] W. Zeng and S. Lei, “Efficient frequency domain selective scrambling of digital video”, *IEEE Transactions on Multimedia*, Vol. 5, No. 1, March 2003, Page(s): 118-129.
[21] S. Lian, J. Sun and Z. Wang, “A Novel Image Encryption Scheme Based-on JPEG Encoding”, In Proceedings of the Eighth International Conference on Information Visualization (IV04), London, UK, July 2004, 217-220.
[22] S. Lian, J. Sun and Z. Wang, “Perceptual Cryptography on SPIHT Compressed Images or Videos”, IEEE International Conference on Multimedia and Expo (I) (ICME 2004), June 2004, v3, pp. 2195-2198.
[23] S. Lian, J. Sun, Z. Wang. “Perceptual Cryptography on JPEG2000 Encoded Images or Videos”, International Conference on Computer and Information Technology, 2004, pp. 78-83.
[24] Y. Zhou, C. Zhang and Z. Zhang, “An efficient fractal image coding algorithm using unified feature and DCT”, *Chaos, Solitons & Fractals*, In Press, Corrected Proof, Available online 7 August 2007.
[25] T. K. Truong, C. M. Kung, J. H. Jeng and M. L. Hsieh, “Fast fractal image compression using spatial correlation”, *Chaos, Solitons & Fractals*, Vol. 22, No. 5, December 2004, pp. 1071-1076.
[26] C. He, X. Xu and J. Yang, “Fast fractal image encoding using one-norm of normalized block”, *Chaos, Solitons & Fractals*, Vol. 27, No. 5, March 2006, pp. 1178-1186.
[27] K.-L. Chung and C.-H. Hsu, “Novel prediction- and subblock-based algorithm for fractal image compression”, *Chaos, Solitons & Fractals*, Vol. 29, No. 1, July 2006, pp. 215-222.
[28] C. He, G. Li and X. Shen, “Interpolation decoding method with variable parameters for fractal image compression”, *Chaos, Solitons & Fractals*, Vol. 32, No. 4, May 2007, pp. 1429-1439.
[29] B. Rejeb and W. Anheier, “A New Approach for the Speed Up of Fractal Image Coding”,13th International Conference on Digital Signal Processing, DSP 97, Vol. 2, pp. 853-856, 2-4 Jul 1997.
[30] D. Saupe, “Fractal Image Compression via Nearest Neighbor Search”, Y. Fisher (ed.), Springer-Verlag, New York, 1996.
[31] V. Sankaranarayanan, “Fractal Image Compression Project Report”, 1998.
[32] Pou-Yah Wu, “Fast Fractal Image Compression”, International Conference on Information Technology: Coding and Computing, pp. 54-59, 2000.
[33] H. Harenstein and D. Saupe, “ Lossless acceleration of fractal image encoding via the Fast Fourier transform”, Elsevier, 2000.
[34] C. S. Tong and M. Wong, “Adaptive Approximate Nearest Neighbor Search for Fractal Image Compression”, *IEEE Transactions on Image Processing*, 2000.
[35] R. Hashemian and S. Marivada, “Improved Image Compression Using Fractal Block Coding”, 2003.
[36] C. Wang and J. Kao, “A Fast encoding algorithm for fractal image compression”, *IEICE Electronics Express*, Vol. 1, No.12, pp. 352-357, 2004.
[37] M. Hassaballah, M. M. Makky and Youssef B. Mahdy, “A Fast Fractal Image Compression Method Based Entropy”, *Electronic Letters on Computer Vision and Image Analysis,* Vol. 5, No. 1, pp. 30-40, 2005.
[38] Z. Zhang and Y. Zhao, “Improving the Performance of Fractal Image Coding”, *International Journal of Innovative Computing, Information and Control IJICIC*, Vol. 2, No. 2, pp. 387-398, April 2006.
[39] A. Uhl and J. Hammerle, “Fractal Image Compression on MIMD architectures I: Basic Algorithms”, First International Conference on Visual Information Systems, Melbourne, 1996.
[40] C. Hufnagl and A. Uhl, “Algorithms for Fractal Image Compression on Massively Parallel SIMD Arrays”, International Picture Coding Symposium (PCS'97), Berlin, Germany, 1997.
[41] H. Nagano, A. Matsuura, and A.Nagoya, “An Efficient Implementation Method of Fractal Image Compression on Dynamically Reconfigurable Architecture”, 1998.
[42] S. Lee, S. Omachi and H. Aso, “A Parallel Architecture for Quadtree-based Fractal Image Coding”, 1999.
[43] K. U. Barthel , J. Schuttemeyer, T. Voyé and P. Noll, “A New Image Coding Technique Unifying Fractal and

Transform Coding", IEEE International Conference on Image Processing (ICIP' 94), 13-16 November, Austin Texas, pp. 112-116, 1994.

[44] R. Hamzaoui, M. Muller and D. Saupe, "Enhancing fractal Image Compression with Vector Quantization", DSPWS, 1996.

[45] G. Melnikov and A. K. Katsaggelos, "A Jointly Optimal Fractal/DCT Compression Scheme", *IEEE Transactions on Multimedia*, Vol. 4, No. 4, pp. 413-422, December 2002.

[46] L. Vences and I. Rudomin, "Genetic Algorithms for Fractal Image and Image Sequence Compression", Comptacion Visual, pp. 1-10, 1997.

[47] S. A. Curtis and C. E. Martin, "Functional Fractal Image Compression", 2003.

[48] Ivan Kopilovic, Dietmar Saupe, and Raouf Hamzaoui, "Progressive Fractal Coding", 2000.

[49] S. K. Mitra, C. A. Murthy, M. K. Kundu, B. B. Bhattacharya and T. Acharya, "Fractal Image Compression using Iterated Function System with Probabilities", IEEE, pp.191-195, 2001.

[50] Y. D. Sun and Y. Zhao, "A parallel implementation of improved fractal image coding based on tree topology", *Chinese Journal of Electronics*, Vol.2, No.2, 2003.

[51] K. U. Barthel and Thomas Voye, "Adaptive Fractal Image Coding in the Frequency Domain", In proceedings of International Workshop on Image Processing: Theory, Methodology, Systems, and Applications, Budapest and in *Journal on Communications*, Vol. XLV, pp. 33-37, 1994.

[52] K. U. Barthel, J. Schuttemeyer, Thomas Voye and Peter Noll, "A New Image Coding Technique Unifying Fractal and Transform Coding", IEEE International Conference on Image Processing, Austin Texas, pp. 112-116, 1994.

[53] R. A. Hobvis, Soundararajan Ezekiel, and Gary A. Winchester, "Texture Based Image Compression by using Quincunx Mask", 2000.

[54] B. Wohlberg and G. de Jager, "Fast image domain fractal compression by DCT domain block matching,", *Electronics Letters*, Vol. 31, pp. 869–870, May 1995.

[55] J. Li and C.-C. Jay Kuo, "Image Compression with a Hybrid Wavelet-Fractal Coder", *IEEE Transactions on Image Processing*, Vol. 8, No. 6, June 1999.

[56] J. Mukherjee, P. Kumar, and S. Ghosh, "A graph-theoretic approach for studying the convergence of fractal encoding algorithm", *IEEE Transaction on Image Processing*, 9, March 2000.

[57] S. K. Ghosh, J. Mukhopadhyay, V. M. Chowdary and A. Jeyaram, "Relative fractal coding and its application in satellite image compression", Proceedings of ICVGIP, Ahmadabad, India, pp. 469-474, Dec. 2002.

[58] Y. Fisher, "Fractal encoding with quadtrees," In Fractal Image Compression: Theory and Applications to Digital Images (Y. Fisher, ed.), pp. 55–77, Springer-Verlag, 1995.

[59] D. Saupe and S. Jacobs, "Variance-based quadtrees in fractal image compression", *Electronics Letters*, Vol. 31, No. 1, pp. 46–48, 1997.

[60] Y. Fisher and S. Menlove, "Fractal encoding with HV partitions", in Fractal Image Compression: Theory and Applications to Digital Images (Y. Fisher, ed.), pp. 119–136, Springer-Verlag, 1995.

[61] N. Lu, Fractal Imaging. New York: Academic Press, 1997.

[62] K. Belloulata and Janusz Konrad, "Fractal Image Compression with Region-Based Functionality", *IEEE Transactions on Image Processing*, Vol. 11, No. 4, April 2002.

[63] A. Kapoor, K. Arora, A. Jain, and G. P. Kapoor, "Stochastic Image Compression Using Fractals", IEEE Proceedings of the International Conference on Information Technology: Computers and Communications (ITCC' 03), 2003.

[64] B. Zalan-Peter, "Maximal Processor Utilization in Parallel Quadtree-based Fractal Image Compression on MIMD Architectures", *Studia Univ. Babes- Bolyai, Informatica*, Vol. XLIX, No. 2, 2004.

[65] T. Laurencot and A.E. Jacquin, "Hybrid image block coders incorporating fractal coding and vector quantization, with a robust classification scheme", AT&T Tech. Memo., 1992.

[66] Y. Linde, A. Buzo, and R.M. Gray, "An algorithm for vector quantizer design", *IEEE Transactions on Communication*, COM-28, pp. 84-95, 1980.

[67] Z. Wang, D. Zhang, and Y. Yu, "Hybrid Image Coding based on Partial Fractal Mapping", Signal Processing: Image Communication vol. 15, pp. 767-779, 2000.

[68] T. Hintz and Q. Wu, "Image compression on Spiral Architecture", Proceedings of The International Conference on Imaging Science, Systems and Technology, pp. 201-204, 2003.

[69] Q. Wu, X. He and T. Hintz, "Preliminary image compression research using uniform image partitioning on Spiral Architecture", The Second International Conference on Information Technology and Applications (IEEE), pp.216-221, 2004.

[70] X. He, T. Hintz and U. Szewcow, "Replicated shared object model for parallel edge detection algorithm based on Spiral Architecture", *Future Generation Computer Systems Journal*, Elsevier, Vol. 14, pp.341-350, 1998.

[71] H. Wang, M. Wang, T. Hintz, X. He and Q. Wu, "Fractal image compression on a pseudo Spiral Architecture", Australian Computer Science Communications, Australian Computer Society Inc, Vol.27, pp.201-207, 2005.

[72] N. V. Thakur, and O. G. Kakde, "Fractal Color Image Compression on a Pseudo Spiral Architecture", 2nd IEEE International Conference on Cybernetics and Intelligent Systems, CIS 2006, Bangkok, Thailand, pp. 228-233, 2006.

[73] G. Schaefer, and M. Stich, "UCID - An Uncompressed Color Image Database", Proc. SPIE, Storage and Retrieval Methods and Applications for Multimedia, San Jose, USA, pp. 472-480, 2004.