An Approach to Fractal image compression based on

Annapurna Misra, Nalini Singh, Boshisottva Dash

Abstract: *The conception of digital image coding techniques is of great interest in various areas concerned with the storage or transmission of images. Fractal-based. Image coding or fractal image coding is a new method of image compression. In this method, similarities between different scales of the image are used for compression. Intuitively, fractals are sets that reveal details at every scale. These sets are in contrast to regular sets like lines, curves and planes that are typically studied in Euclidean geometry and become smooth when sufficiently magnified. In the early implementations of the theory, these similarities were found by human interaction and hence, the images were encoded by interactive computer programs. This resulted in codes for images that were extremely compact in size, but the decoded images had very low quality. In this paper, we propose an independent and novel approach to image coding, based on a fractal theory of iterated transformations.*

Keywords: *Euclidean geometry, Fractal, transmission, human interaction*

I. Introduction

Fractal dimension has been used as a tool in different aspects of image compression algorithms. Fractal dimension was used in a fractal image coder for adjusting error threshold. Also fractal dimension was used for image segmentation. After segmentation, fractal dimension was used as a measure of the complexity of the segment to determine how the segments should be coded. To the human eye, many fractal curves and surfaces, look very similar to natural curves and surfaces, and for this reason they are extensively used in computer graphics. In model based coding, this similarity has the potential to be used for coding of natural images by modeling the processes that generate parts of these Images. In the context of image coding, fractal dimension was also used for selecting the optimal scale parameter in an edge detector. Experiments have shown that the fractal dimension of a curve or set is closely related to human perception of its roughness .Although fractal dimension alone is not enough for generating a visually good approximation of a set, it may be used as one of the parameters for its representation. It is a known fact that the human visual system's sensitivity to details -in any part of the image is dependent on the amount of activity in the background surrounding that part. Fractal dimension of image regions has been used as an objective measure of this activity.

II. Literature Review

1. Arnaud E.Jacquin in his paper*" Image Coding based on a Fractal Theory of Iterated Contractive Image Transformations* "described about the contractive image transformation and the coding techniques.
2. Geoffrey Davis of Department of Mathematics, Bradley Hall, Dartmouth College, Hanover, in his paper "Adaptive Self-quantization of Wavelet subtrees: A Wavelet based theory of Fractal image Compression "has described about the methods of Fractal image coding of wavelet transformed images.

III. Fractal Image Coder

A. Encoding Steps

First ,the image is partitioned into a number of blocks of size 2B x 2B pixels called domain blocks, which are used as building blocks, and a number of blocks of size B x B pixels referred to as range blocks. For each range block the aim is to find the domain block and transformation which is the best match of the original range block[1,2]. The transformation τ_i is defined as the transformation from domain block D_i to range block R_i and is written as the composition of two transformations where S_i and T_i are the so- called geometric and massic parts of τ_i, respectively as in equation(1).

$$\tau_i = T_i \circ S_i \quad (1)$$

A pool of domain blocks D is defined and consists of all image blocks of size 2B x 2B which can be extracted from the original image, and a pool of massic transformations T, made of all block transformations T_i. The problem of finding the best match is then equal to the problem of finding the "best pair" (D_i, T_i) ε D x T, which complies with the minimum distortion given in equation (2)

$$\delta_{l2}\ (\mu\tau\ R_i\ , T_i o S_i\ (\mu\tau\ D_i\ )) \quad (2)$$

The transformation consists of two parts. The two parts are Geometric part and Massic part.

Geometric Part : A down scaling of the domain block in size from 2B x 2B pixel to B x B pixels and a geometric displacement from the location of the domain block to the location of the range block.

Massic Part : Two transformations which manipulates the grey levels of the block and an isometric transformation which simply shuffles pixels within the block.

Contrast scaling by α :

$$(\sigma\mu)_{i,j} = \alpha\mu_{i,j} \quad (3)$$

Luminance shift by Δg :

$$(\tau\mu)_{i,j} = \mu_{i,j} + \Delta g \quad (4)$$

Isometries by I_i :The following transformations simply shuffle pixels within a range block, in a deterministic way- we call them isometries[3]. We list in the following a list of the eight canonical isometries of a square block as shown in the figure1.

- **Identity :**

$$(I_0\mu)_{i,j} = \mu_{i,j} \quad (5)$$

- **Orthogonal reflection about mid-vertical axis(j =(B-1)/2) of block:**

$$(I_1\mu)_{i,j} = \mu_{i,B-1-j} \quad (6)$$

- **Orthogonal reflection about the mid-horizontal axis (i= (B-1)/2) of block:**

$$(I_2\mu)_{i,j} = \mu_{B-1-i,j} \quad (7)$$

- **Orthogonal reflection about first diagonal (i=j) of block:**

$$(I_3\mu)_{i,j} = \mu_{j,i} \quad (8)$$

- **Orthogonal reflection about second diagonal (i + j = B -1) of block :**

$$(I_4\mu)_{i,j} = \mu_{B-1-j,B-1-i} \quad (9)$$

- **Rotation around center of block, through + 90^0 :**

$$(I_5\mu)_{i,j} = \mu_{j,B-1-i} \quad (10)$$

- **Rotation around center of block, through + 180^0:**

$$(I_6\mu)_{i,j} = \mu_{B-1-i,B-1-j} \quad (11)$$

- **Rotation around center of block, through – 90^0 :**

$$(I_7\mu)_{i,j} = \mu_{B-1-i,j} \quad (12)$$

In effect, massic transformations allow us to generate, from a single block, a whole family of geometrically related transformed blocks, which provides a pool in which matching blocks will be looked for during the encoding.

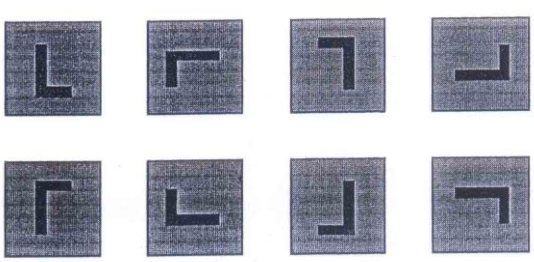

Figure 1. Different forms of Image using Isometries.

The total transformation from a domain block $\mu_{\tau Di}$ to a range block $\mu_{\tau Ri}$ is given in equation (12)

$$T_i(S_j(\mu_{\tau D_j})) = l_i(\alpha_i(S_j(\mu_{\tau D_i})) + \Delta g_i) \quad (12)$$

With the δ_{l2} metric we are able to calculate the contractivity of a transformation for example the contrast scaling on the block S (io, j_o, B) as given in equation(13)

$$\delta_{l2}(\sigma(\mu_{\tau s}),\sigma(v_{\tau s})) = \sum_{0\le i,\ J<B} (\alpha\mu_{i_o+i,\ j_o+j} - \alpha v_{i_o+i,\ j_o+j})^2 = \alpha^2 \delta_{l2}(\mu_{\tau s}) \quad (13)$$

So the contractivity of this transformation is α^2.Let μ is an image or a block and $\tilde{\mu}$ is the approximation of μ. A square block of size B x B pixels located at position (i_0, j_0) in μ is denoted by $s(i_0, j_0, B)$. To measure the distortion between the two blocks μ and $\tilde{\mu}$ of sizes B x B pixels, the following peak signal-to-noise ratio is given in equation (14) :

$$PSNR = 10\log_{10}\left(\frac{255*255}{\delta_{l_2}\ (\mu,\tilde{\mu})\ /\ B^2}\right) \quad (14)$$

The $\delta_{l_2}(\mu,\tilde{\mu})$ is the squared distortion between the two blocks is given in equation (15):

$$\delta_{l_2}(\mu,\tilde{\mu}) = \sum_{0\le i,\ j<B} (\mu_{ij} - \tilde{\mu}_{ij})^2 \quad (15)$$

B.Image Coding Algorithm Suppose that the image is segmented into blocks of size 4 X 4 pixels called Ranges. Each range block R must be approximated as R≈ sD + 01, where D is a 4 X 4 block from the shape codebook._Consider any domain block of size 8 X 8 in the image [4, 11]. Then shrink the block by pixel averaging to the desired size of 4 X 4 pixels. All such blocks are added to the shape codebook. For an image of size 512 X 512 pixels this process yields a huge codebook with 255025 blocks. In order to reduce the number of blocks to a more manageable size one may consider only those blocks that have their upper left corner pixel on a regular square grid with a spacing 1:1. For example, with 1=8 we would obtain 4096 adjacent domain blocks, which is often used in practice.

The encoder has to solve the following problem. For each range block the best approximation ≈sD + 01 needs to be found. In fractal encoding the coefficients s and o are called scaling and offset. To obtain optimal s, o and D, a scan of all the codebooks_blocks D should be performed. For each codebook block D the best coefficients s and o need to be determined. However, for all but the smallest scalar codebooks for s and o this is computationally infeasible. It takes too long. Fortunately, there exists a shortcut. If we work with the Euclidean norm when making the selection of the best coefficients i.e., finding the best coefficient.

$$E(D,R) = \min_{i_0, j_0} \| R - (sD + 01) \| \quad (16)$$

Given the two blocks R and D with n pixel intensities, $\Upsilon_1 \ldots \Upsilon_n$ and $d_1 \ldots d_n$ we have to minimize the quantity as in equation (17)

$$\sum_{i=1}^{n} (s.d_i + o - r_i)^2 \quad (17)$$

The best coefficients are given by equation (18)

$$s = \frac{n(\sum_{i=1}^{n} d_i r_i) - (\sum_{i=1}^{n} d_i)(\sum_{i=1}^{n} r_i)}{n(\sum_{i=1}^{n} d_i^2) - (\sum_{i=1}^{n} d_i)^2} \quad (18)$$

And

$$o = \frac{1}{n}\left(\sum_{i=1}^{n} r_i - s\sum_{i=1}^{n} d_i\right) \quad (19)$$

With s and o given the square error is given in equation (20)

$$E(D,R)^2 = \frac{1}{n}\left[\sum_{i=1}^{n} r_i^2 + s(s\sum_{i=1}^{n} d_i^2 - 2\sum_{i=1}^{n} d_i r_i + 2o\sum_{i=1}^{n} d_i) + o(on - 2\sum_{i=1}^{n} r_i)\right] \quad (20)$$

This procedure yields two real numbers s and o. For the encoding we can only use the quantized values from the scalar codebooks [5, 12]. Usually, one employs uniform scalar quantization amounting to a rounding operation.

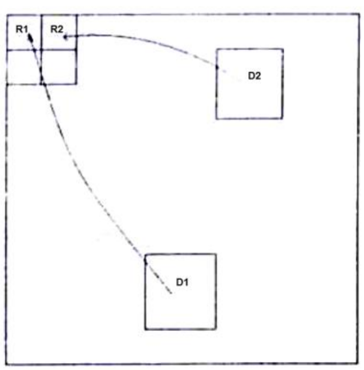

Figure 2. Example of a range and domain block

In summary the baseline encoder using a fixed block operates in the following steps.

1. *Image segmentation:* Segment the given image using a fixed block size, e.g., 4 X 4. The resulting blocks are called the ranges R_i.
2. *Domain pool and shape code book:* By stepping through the image with a step size of I pixels horizontally and vertically create a list of domain blocks from the image, which are twice the range size. By averaging four pixels each, shrink the domain blocks to match the size of the ranges. This produces the codebook of blocks *Di*
3. The search for each range block R an optimal approximation $R \approx sD + o1$ is computed as follows. For each codebook block D_i compute an optimal approximation $R \approx s\, D_i = o1$ in three steps:

- Perform the least squares optimization using the given formulae, yielding a real coefficient s and an offset o.
- Quantize the coefficients using a uniform quantizer.
- Using the quantized coefficients s and o compute the error $E(R, D_i)$.

Among all the codebook blocks D_i find the block D_k with the minimum error [6, 7]. Output the code for the current range block consisting of indices for the quantized coefficients sand 0 and the index k identifying the optimal codebook block D_k .As already mentioned the output code of this baseline encoder is not a code with which the decoder can directly recover an approximation of the original. Instead we have a description of an operator.

Given the original image along with its partitioning in square ranges replace each range R by the corresponding blocks SD + 01 as specified by the code. The resulting image, called collage, is an approximation of the original. Thus the code is nothing but the description of an image operator. Given any image 90 one can carry out the operation iteratively to arrive at the reconstructed image. This is exactly what is done at the decoder. Usually the domain blocks are chosen to be twice the size as the corresponding range blocks. The contract activity condition for the image operator does not require a geometric contraction of domain blocks. Therefore the domain blocks can be of any size. It seems, however, that the error propagation at the decoder is generally worse when the geometric scaling factor is too small [8]. Therefore, shrinking 'the domains to half their original size is practical from the computational point of view and seems to produce the best looking results. It is common practice to enlarge the domain pool by including blocks obtained by rotation and by reflection.

IV. RESULTS AND DISCUSSIONS

In fractal coding usually a square block of size 2^r X 2^r is approximated by another image block size of size 2^{r+1} X 2^{r+1} under an affine mapping. Thus, one tries to find similar structures at two different scales. For a given 512 X 512 grey scale image, partitioned into non overlapping 16 X 16 blocks, a fractal code C is determined in the standard way, considering the domain pool non overlapping 32 X 32 blocks. C contains the information of 32.32 transformations [9,10]. In the decoding, C is used to compute the attractor A_l of size 512 X 512. But C can also be iteratively applied to an arbitrary 256 X 256 image, partitioned into 8 X 8 blocks, gaining an attractor A_2, or to an 128 X 128 image, partitioned into 4 X 4 blocks, giving an attractor A_3, and so on. Thus, one ends up with a pyramid A_1, $A2 \ldots A5$, describing different resolutions of the attractor.

(a)

(b)

(c)

(d)

(e)

(f)

(g)

Figure 3. Output of the coder showing results for different iterations(a,b,c,d,e,f,g) for iterations(1,2,3,5,7,10,20)

No. of iterations of Lena image	***Encoding Time in sec***	***Decoding Time in sec***	***MSE***	***PSNR***
1	62.891	1.437	521.64	69.156
2	62.078	2.86	289.68	71.71
3	62.922	4.297	159.07	74.313
5	62.719	7.109	107.95	75.997
7	62.672	9.953	107.95	75.997
10	63.328	14.265	107.95	75.997
20	63.156	28.516	107.95	75.997

Figure 4. Output of fractal coder of Lena image for one iteration

The result shows applying iterations on the image, for a less no. of iterations the MSE is more and for more iteration MSE is less. After certain iterations we get a constant MSE and PSNR value.

V. CONCLUSION AND FUTURE WORK

When the above theory is used for image compression, it is implemented in a discrete setting; however, the fractal code generated by encoding a digital image describes relationships, in the form of affine functions, between various segments of the image and is independent of the resolution of the image. In other words, the fractal code is a resolution independent representation of the image and theoretically represents a continuous image approximating the original image. A decoder may decode the code to generate a digital image at any resolution. The resolution of the decoded image may as well be higher than the resolution of the original image. The increase in resolution is sometimes referred to as the fractal zoom. The higher resolution obtained is not created by a simplistic technique such as repeating the pixels of the image, but more detail is actually generated in the decoded images. When an image is reconstructed at the same resolution as the original, in the decoding process the domain blocks are shrunk, which eliminates some of the details of the domain blocks. However, if the image is reconstructed at a higher resolution, in the shrinking of the domain block, the details of the domain block are only shrunk to generate the extra resolution in the range block. In fact, details of the domain blocks are used for missing details of the range blocks. The details in the domain block are also generated to some extent from details of other domain blocks used for encoding each part. In other words, it is implicitly assumed that if the range block is similar to its corresponding domain block, then the details of the range block, which are beyond the resolution of the original image, are also similar to the details of the domain block, which are within the resolution of the encoded image. This assumption is a typical property of self similarity of fractal sets at different scales and the resolution in dependence is a property of the code generated by the fractal based methods. There are a number of ways in which the designed codec can be improved. Ongoing research around the world has suggested combination of more than one scheme for achieving fast, deep and less distortion compression. Among some of these schemes are a combination of wavelets and vector quantization, fractals with vector quantization and even wavelets with neural networks. The coder was designed for grey scale images only. The same algorithm could be used for color images too with modifications. Fractal coding is also being tried for video coding and has proved to be promising. Further work in this direction should consider efficient compression using zero trees, compression of color images, and deep video compression.

ACKNOWLEDGMENT

The authors thank to Dr. R. N. Pal, retired Professor IIT Kharagpur, India and his team for their help in conducting experiments with different face images.

REFERENCES

1. Fractals and the human visual system," in Proceedings of IEEE ICASSP-90, Albuquerque, NM, Apr.3 {6,1990, pp. 1957-1960}.
2. J. E. Hutchinson, Fractals and self-similarity," Indiana University Mathematics Journal, vol.30, pp. 713{747, Sept.-Oct.1981}.
3. M.F. Barnsley, Methods and apparatus for image compression by iterated function systems." United States Patent Number 4, 941, 193, 1990.
4. C.K. Lee and W.K. Lee, "Fast fractal image block coding based on local variances", IEEE Trans. Image Processing, Vol 7, No.6, pp 888-891, June 1998.
5. Y. Fisher, Ed., Fractal Image Compression: Theory and Applications;, New York: Springer- Verlag 1995.
6. A. Jacquin, "Image coding based on a fractal theory of iterated contractive image transformations", IEEE Trans. Image Processing, Vol.1 No.1, pp 18-30, Jan 1992.
7. G.M. Davis, "Self-quantization of wavelet subtrees: A wavelet-based theory of fractal image compression. "in Proc. Data Compression Conf. Snowbird. UT. Mar, 1995. pp. 232-241.
8. H. Krupnik. D. Malah and E. Kamin., "Fractal representation of images via the discrete wavelet transforms." In IEEE 18th Conv. Electrical Engineering in Israel. Tel-Aviv. Israel, Mar 1995.
9. A. van de Walle. "Merging fractal image compression and wavelet transform methods." In Fractal Image Coding and Analysis: A NATO ASI Series Book. Y. Fisher. Ed. New York: Springer-Verlag. 1996.
10. J. Shapiro, "Embedded image coding using zerotrees of wavelet coefficients." IEEE Trans. Signal Processing. Vol.41. pp. 3445-3462. Dec., 1993.
11. Y. Fisher, B. Jacobs, and R. Boss, "Fractal image compression using iterated transforms." In Image and Text Compression. J. Storer. Ed. Boston, MA: Kluwer, 1992. pp. 35-61.
12. Fractals and the human visual system," in Proceedings of IEEE ICASSP-90, Albuquerque, NM, Apr.3 {6, 1990, pp. 1957-1960}.