Dynamics of Mobius Transformations and Image Compression

Presented By:-

Piyush Keshri
Y5303
3rd Year, B.Tech (EE), IIT Kanpur

Outline

- **Introduction to Mobius Transformations**
- Introduction to Fractal Image Compression
- Mathematical Foundations of Image Compression
- Fundamental Principles of Image Compression
- Design of Digital Image encoder
- Design of Fractal Block coding System
- Encoding of Digital Image using Fractals
- Output File
- Stochastic Image compression
- Decoding of Image from fractal code
- Results obtained for Lena & Baboon Images
- Comparison between Stochastic & Non stochastic Algorithms
- Differences between different Fractal Coding Schemes
- Specifications/ Parameters
- References

Mobius Transformations

- A **Möbius transformation** is a function, $f : \hat{\mathbb{C}} \to \hat{\mathbb{C}}$ defined by,

$$f(z) = \frac{az+b}{cz+d}$$

where z, a, b, c, d are complex numbers satisfying $ad - bc \neq 0$.

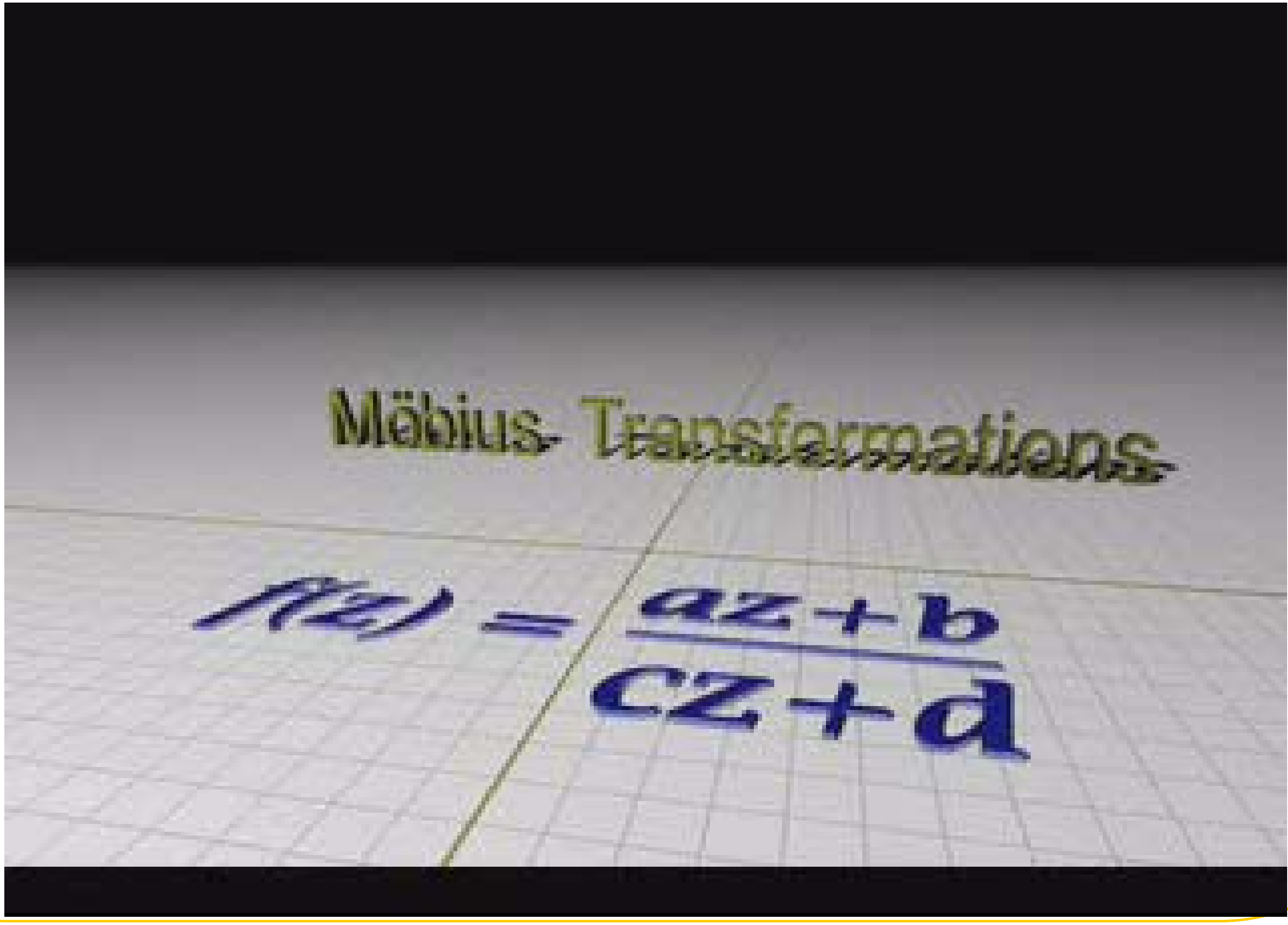
- It maps the whole plane $\mathbb{C}$, together with the the point at infinity, onto the sphere $\hat{\mathbb{C}}$. Hence, it maps circles in the Riemann sphere into circles in the Riemann sphere, or, equivalently, that the image of a line or circle in the plane is another line or circle.

A Möbius transformation is composition of elementary transformations:

- Translation, $f_1(z) = z + d/c$
- Inversion and reflection, $f_2(z) = 1/z$
- Dialation and Rotation, $f_3(z) = \left(-\frac{ad-bc}{c^2}\right).z$
- Translation, $f_4(z) = z + a/c$

Möbius transformation, $\Longleftrightarrow$ $f_4 \circ f_3 \circ f_2 \circ f_1(z) = \frac{az+b}{cz+d}$

Mobius Transformation



Outline

- Introduction to Mobius Transformations
- **Introduction to Fractal Image Compression**
- **Mathematical Foundations of Image Compression**
- Fundamental Principles of Image Compression
- Design of Digital Image encoder
- Design of Fractal Block coding System
- Encoding of Digital Image using Fractals
- Output File
- Stochastic Image compression
- Decoding of Image from fractal code
- Results obtained for Lena & Baboon Images
- Comparison between Stochastic & Non stochastic Algorithms
- Differences between different Fractal Coding Schemes
- Specifications/ Parameters
- References

Introduction to Fractal Image Compression

- *Fractal compression is a lossy image compression method using fractals to achieve high levels of compression.*
- *Its a comparatively new technique which has gained considerable attention in the popular technical press, and in the research literature.*

- *The most significant advantages claimed are:*
  - *high reconstruction quality at low coding rates*
  - *rapid decoding*
  - *"resolution independence" ⇨ an encoded image may be decoded at a higher resolution than the original.*

Introduction….

- The main characteristics of this approach are :
  - It relies on the assumption that image redundancy can be efficiently exploited through self-transformability on a blockwise basis.
  - It approximates an original image by a fractal image.

- Fractal compression represents an image by the parameters of a set of affine transforms on image blocks under which the image is approximately invariant.

Mathematical Foundations of Fractal Compression

- Fractal image compression is based on a result of metric space theory known as **Banach's Fixed Point Theorem**, which guarantees that an image may be reconstructed from its representation as a contractive transform of which it is a fixed point.

- **Banach Fixed Point theorem.** *A contraction mapping $T: X \Rightarrow X$ on a complete metric space (X,d) has precisely one fixed point.*

- *A mapping $T: X \Rightarrow X$ on a metric space (X,d) is a contraction mapping if*

$$\exists \alpha \in \mathbb{R}, 0 < \alpha < 1 \text{ such that } \forall x, y \in X$$

$$d(T_x, T_y) \leq \alpha d(x, y)$$

- The mathematical foundation of the technique is a general theory of iterated contractive transformations in metric spaces.

Mathematical foundations…

- Banach's theorem proves that the fixed point of any contraction mapping in a complete metric space may be approximated to arbitrary accuracy by iterated application of the contraction mapping to an arbitrary initial element of the metric space.
- To determine the inverse problem of finding a contraction mapping having a given point as its fixed point, the Collage Theorem is used.

- **Collage theorem.** *If (X,d) is a complete metric space, T :X ⇨ X is a contraction mapping with contractivity 0≤ α <1 and fixed point* x_T *, then*

$$d(x, x_T) \leq (1-\alpha)^{-1}\, d(x, Tx) \quad \forall x \in X$$

Methods for Image Compression using Fractals

- Mainly deals with solving the inverse problem of finding such transformations corresponding to a given image.
- Method is based on constructing contractive affine transformations for which the given image is a fixed point.
- The theories of Iterated Function System (IFS) and Recurrent Iterated Function System form the basis for fractal image compression techniques .
- Each transform operates only on a subregion of the image, called "**domain blocks**".
- The image subregions to which the domain blocks are mapped are called "**range blocks**".

Outline

- Introduction to Mobius Transformations
- Introduction to Fractal Image Compression
- Mathematical Foundations of Image Compression
- **Fundamental Principles of Image Compression**
- Design of Digital Image encoder
- Design of Fractal Block coding System
- Encoding of Digital Image using Fractals
- Output File
- Stochastic Image compression
- Decoding of Image from fractal code
- Results obtained for Lena & Baboon Images
- Comparison between Stochastic & Non stochastic Algorithms
- Differences between different Fractal Coding Schemes
- Specifications/ Parameters
- References

Fundamental Principles of fractal coding

- **Step1.** Tile the image by nonoverlapping range blocks (e.g. 8 * 8) and larger (e.g. 16 * 16), possibly overlapping domain blocks.

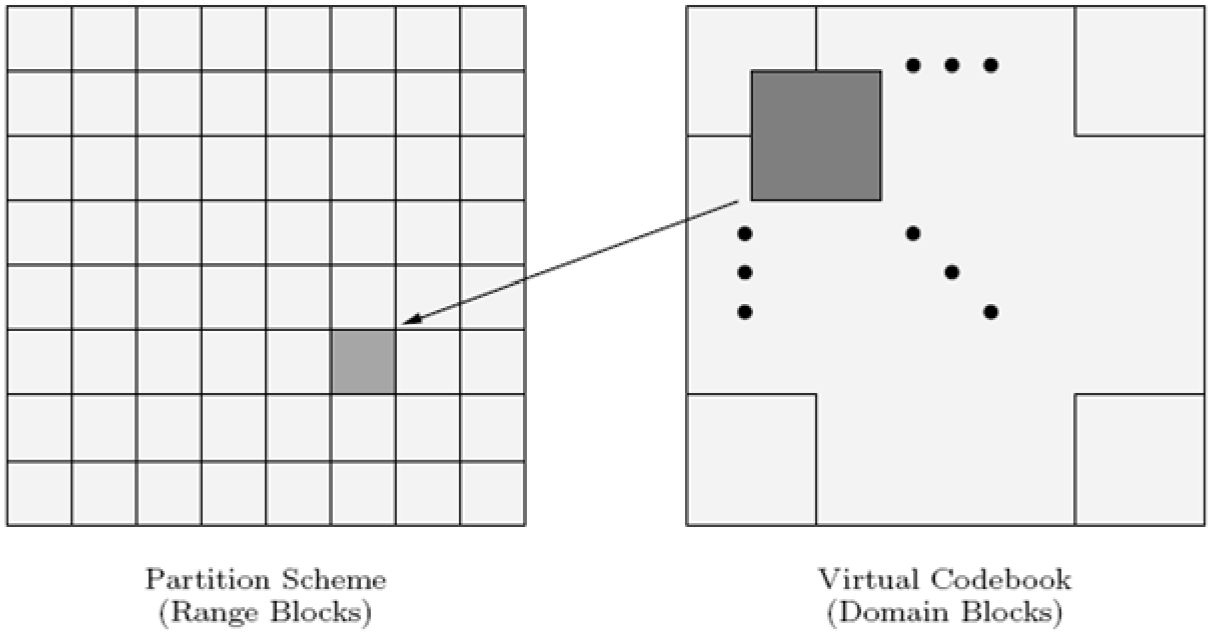


Fig. Domain & Range blocks in PIFS coding.

Fundamental Principles of fractal coding

- **Step 2.** A set of admissible block transforms is defined, consisting of a contraction of the block support by a factor of two on each side by averaging neighbouring pixels.

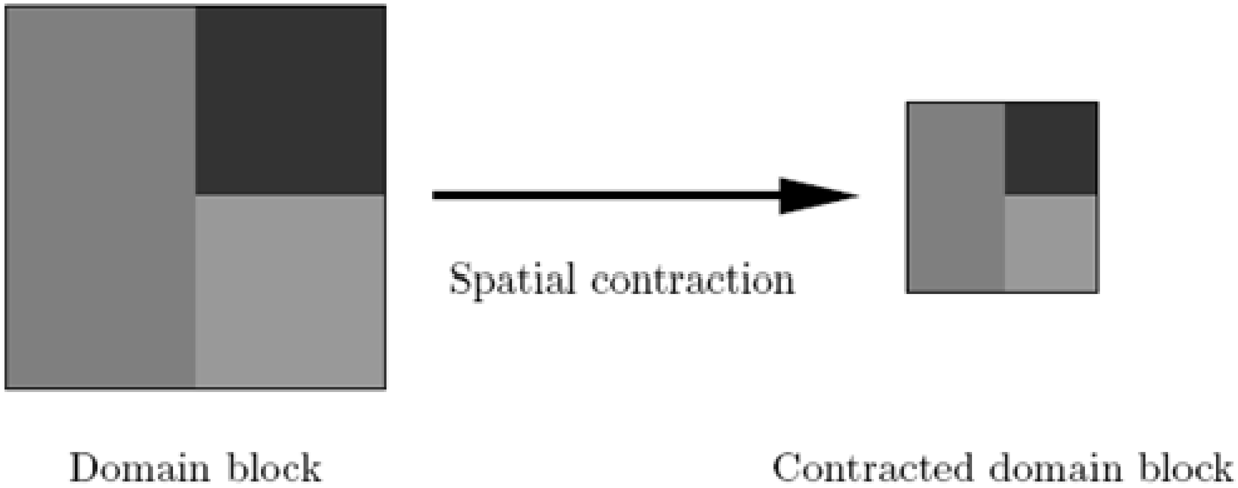


Fig. Spatial contraction of a Domain block.

Fundamentals...

- **Step 3.** Process is followed by the application of one of the eight rotations and reflections making up the isometries of a square.

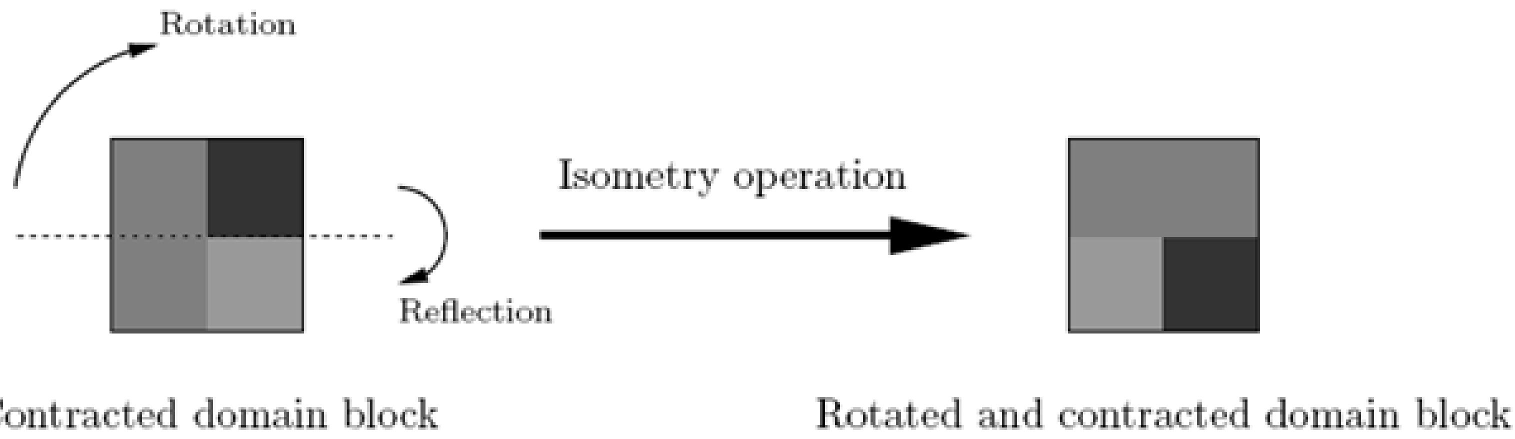
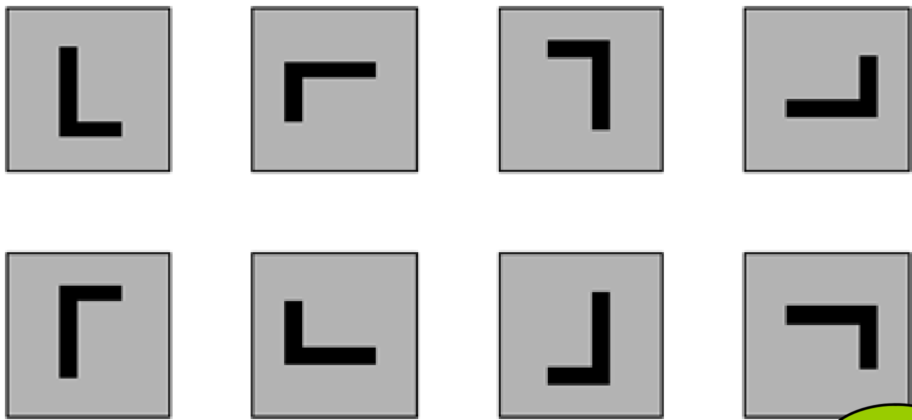


Fig. An isometry applied to domain block.

Fig. The Square isometries.



Fundamentals…

- **Step 4.** Finally an affine transform on the pixel intensities of the range block (contracted Domain block) is applied.

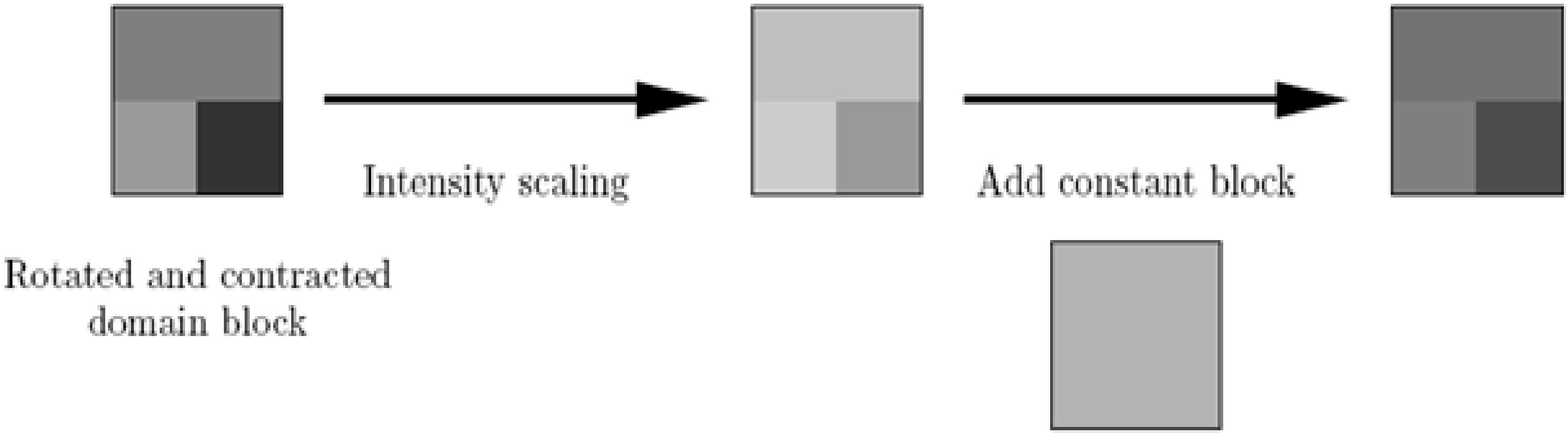


Fig. An Affine transformation applied to the domain block.

Fundamentals…

- **Step 5.** The ***Encoding Phase*** (utilising the *Collage Theorem*) consists of finding for each range block a domain block for which the pixel values can be made close to those of the range block by the application of an admissible transform.
- **Step 6.** Once encoding is complete, the image is represented by a list containing the selected domain block and transform parameters for each range block.
- **Step 7.** The image is decoded by iteratively transforming an arbitrary initial image using the transform consisting of the union of the transforms for each range block.

Outline

- Introduction to Mobius Transformations
- Introduction to Fractal Image Compression
- Mathematical Foundations of Image Compression
- Fundamental Principles of Image Compression
- **Design of Digital Image encoder**
- Design of Fractal Block coding System
- Encoding of Digital Image using Fractals
- Output File
- Decoding of Image from fractal code
- Results obtained for Lena & Baboon Images
- Comparison between Stochastic & Non stochastic Algorithms
- Differences between different Fractal Coding Schemes
- Specifications/ Parameters
- References

Issues in Design of Digital Image coder

- Practical issues in the design of a digital image coder based on iterated transformation theory are presented in next section. This covers:
- The construction of a partition of an image support.
- The selection of a distortion measure.
- The specification of a class of discrete contractive image transformations defined blockwise from which fractal codes are selected.

Theoretical Foundations

- **The Inverse Problem of Iterated Transformation Theory**

Let $(\mathcal{M}, d)$ $\Rightarrow$ denote a metric space of digital images.

d $\Rightarrow$ is a given metric-distortion measure.

μ_{orig} $\Rightarrow$ is an original image (fixed point).

τ $\Rightarrow$ contractive image transformation.

$\mathcal{F}$ $\Rightarrow$ set of allowed transformations.

The requirements on the transformation τ are formulated as:

$$\exists s < 1 \text{ such that } \forall \mu \in \mathcal{M}, d(\tau(\mu), \tau(\nu)) \leq s d(\mu, \nu),$$

$d(\mu_{orig}, \tau(\mu_{orig}))$ is as "small" as possible.

(app. fixed point)

S $\Rightarrow$ contractivity of τ.

Construction Of Fractal Codes

Structure Of Transformation

- Constructing a single transformation τ for whole image with some distortion is difficult.
- Image partitioned into small blocks.
- For each block a transformation is constructed.
- Set of transformations serves as the transformation for whole image.
- Thus, the transformation τ in the following form:

$$\forall \mu \in M, \ \tau(\mu) = \sum_{0 \le i < N} (\tau\mu)_{R_i} = \sum_{0 \le i < N} \tau_i(\mu_{D_i})$$

- Where, $\wp = \{R_i\}_{0 \le i < N}$ denotes the non overlapping partition of image in N cells, usually squares.
- τ_i denotes transformation from domain cell D_i ⇨ Range cell R_i.
- $\tau_i = \Im \, o \wp$, where $\Im$ is the **massic part** & $\wp$ is the **geometric part.**

Outline

- Introduction to Mobius Transformations
- Introduction to Fractal Image Compression
- Mathematical Foundations of Image Compression
- Fundamental Principles of Image Compression
- Design of Digital Image encoder
- **Design of Fractal Block coding System**
- Encoding of Digital Image using Fractals
- Output File
- Stochastic Image compression
- Decoding of Image from fractal code
- Results obtained for Lena & Baboon Images
- Comparison between Stochastic & Non stochastic Algorithms
- Differences between different Fractal Coding Schemes
- Specifications/ Parameters
- References

Design Of Fractal Block coding System

Three main issues involved in the design & implementation of a fractal block coding system are:

1. The partition of image(image dependent).
2. The choice of a measure of distortion between two images.
3. The specification of a class of discrete contractive image transformations defined consistently with a partition, and of a scheme for the quantization of their parameters.

1. Image Partition

- Square support S of the original digital image μ_{orig} is partitioned into non-overlapping square range cells of two different sizes, forming two-level square partition.
- The larger cells- of size BxB known as (range) parent cells.
- The smaller ones- of size B/2 x B/2 known as (range) child cells.
- A parent cell can be split into upto 4 non overlapping child cells.
- Decisions about the splitting of a parent cell are made during the encoding of the image block over this cell.

- Thus a partition constructed is image dependent. It allows encoder to:

1. To use larger bocks to take advantage of smoothly varying image areas.
2. To use small blocks to capture details in complex areas.

Small blocks vs. Large block image cells

- **Small image blocks :**

1. Easy to analyze and to classify geometrically.
2. Allow fast evaluation of interblock distances.
3. Easy to encode correctly.
4. Lead to a robust encoding system.

- **Large blocks :**

1. Allow exploitation of the redundancy in smooth image areas.
2. Lead theoretically to high compression ratios.

2. Distortion Measure

- A distortion measure between digital images is constructed from an interblock distortion measure.
- Let $S(i_o,j_o,B)$ denote the square cell of size BxB, with the bottom left corner at (i_o,j_o).
- Let μ be an r x r image, and $\widehat{\mu}$ be an approximate image of μ.
- Let $\mu\rceil_S$ and $\widehat{\mu}\rceil_S$ denote their restrictions to the cell $S(i_o,j_o,B)$.the L_2 or Mean Squared (MS) distortion b/w image blocks $\mu\rceil_S$ and $\widehat{\mu}\rceil_S$ is defined as sum over the cell s, of the difference of pixel values, i.e. :

$$d_{L_2}(\mu\rceil S, \nu\rceil S) = \sum_{0 \le i,j < B} (\mu_{i_0+i,j_0+j} - \nu_{i_0+i,j_0+j})^2.$$

- Peak-to-peak signal-to-noise ratio (SNR) is given by:

$$\mathrm{SNR} = 10 \log_{10} \left(\frac{dr(\mu)^2}{d(\mu, \tilde{\mu})/r^2} \right)$$

Where $dr(\mu)$ denotes dynamic range of μ.

dynamic range(dr)= (highest pixel value – lowest pixel value +1).

3. Class of Discrete Image Transforms

- The class $\mathcal{F}$ of discrete contractive affine transformations defined blockwise that coder uses are:

  - Geometric Part
  - Massic Part

$$\mu_{\rceil D_i} \xmapsto{\mathcal{S}_i} \mathcal{S}_i(\mu_{\rceil D_i}) \xmapsto{\mathcal{T}_i} (\mathcal{T}_i \circ \mathcal{S}_i)(\mu_{\rceil D_i}) = (\tau\mu)_{\rceil R_i}$$

$$\downarrow \Sigma$$

$$\tau(\mu)$$

Fig. Application of an image transformation τ defined blockwise to an image μ.

Image Transforms…

- **Geometric Part** $\wp$

Domain cell of size 2B (2B x 2B) is mapped by geometric transformation onto a range cell of size B (B x B).

Pixel value of the contracted image on range block are average of 4 pixel values of domain block:

$$(\wp\mu)_{i,j} = (\mu_{2i,2j} + \mu_{2i+1,2j} + \mu_{2i+1,2j+1})/4$$

Where i,j ϵ {0,……..B-1}

Massic Transformations…

Massic Part $\Im$

These transformations effect affect pixel values of the transformed domain blocks. The transformations are:

- **Absorption at constant gray level g:**
  $(\theta\mu)_{i,j} = g,\ g \in \{0,\ldots\ldots\ldots\ldots.255\}$
- **Luminance shift by:**
  $\Delta g : (\zeta\mu)_{i,j} = \mu_{i,j} + \Delta g,\ \Delta g \in \{0,\&\ldots 255\}$
- **Contrast scaling** by $a \in [0,1]$: $(\sigma\mu)_{i,j} = a\mu_{i,j}$
- **Color reversal :** $(\rho\mu)i,j = 255 - \mu_{i,j}$
- **Isometries**

Massic Transformations...(Isometries)

Isometry: *In mathematics, an **isometry** is a distance-preserving isomorphism between metric spaces.*
Let X and Y be metric spaces with metrics d_X and d_Y. A map $f : X \to Y$ *is called **distance preserving** if for any* $x, y \in X$ *one has* .

$$d_Y\left(f(x), f(y)\right) = d_X(x, y)$$

- **The isometries are (for block size = B) :**

1. Identity

$$(\iota_0 \mu)_{i,j} = \mu_{i,j}.$$

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

$$(\iota_1 \mu)_{i,j} = \mu_{i,B-1-j}.$$

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

$$(\iota_2 \mu)_{i,j} = \mu_{B-1-i,j}.$$

Isometries...

- **The isometries are (for block size = B) :**

4. Orthogonal reflection about first diagonal (i=j) of block:

$$(\iota_3 \mu)_{i,j} = \mu_{j,i}.$$

5. Orthogonal reflection about another diagonal (i+j=B-1) of block:

$$(\iota_4 \mu)_{i,j} = \mu_{B-1-j,B-1-i}.$$

6. Rotation around centre of block by (+90) degrees.

$$(\iota_5 \mu)_{i,j} = \mu_{j,B-1-i}.$$

7. Rotation around centre of block by (+180) degrees:

$$(\iota_6 \mu)_{i,j} = \mu_{B-1-i,B-1-j}.$$

8. Rotation around centre of block by (-90) degrees

$$(\iota_7 \mu)_{i,j} = \mu_{B-1-i,j}.$$

Massic Transformations...(Contractivity)

- Massic transformations allow to generate a whole family of geometrically related transformed blocks, which provides a pool in which matching blocks will be looked for during the encoding.
- The L_2 contractivities of discrete block transformations can be computed easily :

Massic Transformation	L_2-Contractivity
Absoption at g_0	$s = 0$
Luminance shift by Δg	$s = 1$
Contrast scaling by α	$s = \alpha^2$
Color reversal	$s = 1$
Isometries $\{\iota_n\}_{0 \leq n \leq 7}$	$s = 1$

Outline

- Introduction to Mobius Transformations
- Introduction to Fractal Image Compression
- Mathematical Foundations of Image Compression
- Fundamental Principles of Image Compression
- Design of Digital Image encoder
- Design of Fractal Block coding System
- **Encoding of Digital Image using Fractals**
- Output File
- Stochastic Image compression
- Decoding of Image from fractal code
- Results obtained for Lena & Baboon Images
- Comparison between Stochastic & Non stochastic Algorithms
- Differences between different Fractal Coding Schemes
- Specifications/ Parameters
- References

Encoding of digital Image using fractals

- This describes the procedure for the encoding of any monochrome digital image, given a specific fractal block coder. The procedure consists of:

  - The organized search of a **"virtual codebook"** obtained from a pool of domain blocks and a pool of block transformations.

- The result of the search is a discrete image transformation defined blockwise, built so as to leave the original image approximately invariant and known as a fractal code.

Class of Domain Blocks

- Range block of size B.
- Maximal pool of domain blocks is “huge” set of all possible blocks of size 2B.
- To trim this pool, only those blocks as domain blocks are considered which fall under a sliding window of size 2B, shifted over image horizontally and vertically by a fixed number of pixel values.
- The pool so obtained is further divided into:
- ✓ Shade Blocks
- ✓ Midrange Blocks
- ✓ Edge Blocks

Domain blocks…

- Original r x r digital image μ is given as input to the encoder.
- $\{R_i\}_{0 \le i < N}$ ⟹ image partition made of range cells
- Image transformation τ has the form:

$$\tau = \sum_{0 \le i < N} \tau_i, \quad \text{With} \quad \tau_i = \mathcal{T}_i \circ \mathcal{S}_i.$$

- Construction of a transformation τ_i maps onto the range cell R_i (of size B X B) is broken into two distinct steps corresponding to the transformation $\mathcal{S}_i$ & $\mathcal{T}_i$ respectively.
- A pool of domain blocks D, is formed extracted from original image.
- A pool of massic transformations $\mathcal{T}$, made of discrete block transformations $\mathcal{T}_i$.
- The encoding of range block $\mu_{\rceil R_i}$ consists in finding a "best" pair $(D_i, \mathcal{T}_i) \in \mathfrak{D} \times \mathcal{T}$, Such that the distortion: $d(\mu_{\rceil R_i}, \mathcal{T}_i \circ \mathcal{S}_i(\mu_{\rceil D_i}))$ is minimum.

Virtual Codebook

- The product $\mathfrak{D} \times \mathscr{T}$ is called a **global pool** or **virtual codebook**.

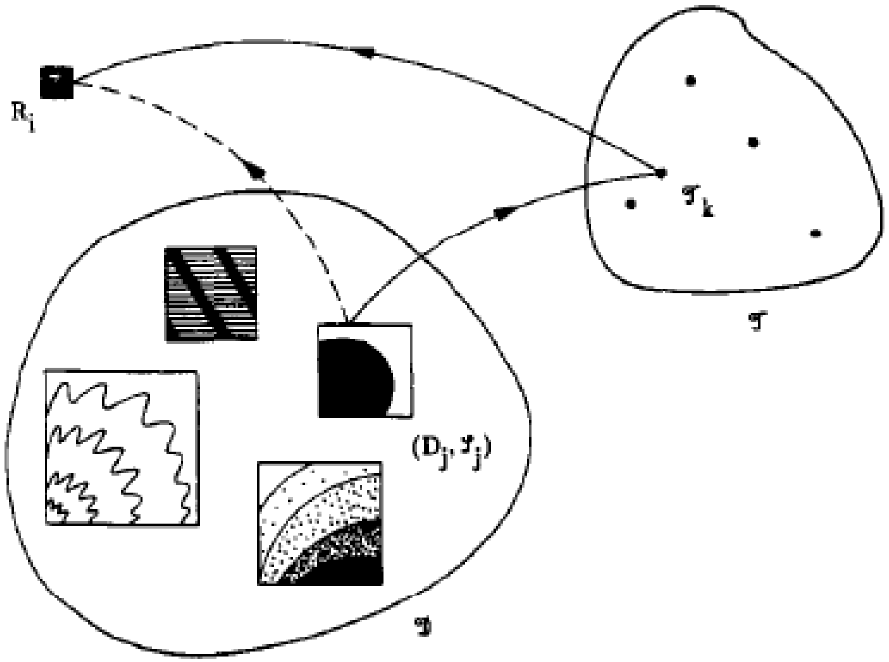


Fig. Encoding of the range block R_i.

Classification of Domain Blocks

- Domain blocks can be classified into:

- **Shade Blocks :** is "smooth" with no significant gradient.

- **Edge Blocks :** Presents a strong change of intensity across a curve-often a piece of object boundary, which runs through the block.

- **Midrange Blocks :** Has moderate gradient but no definite edge. Finely textured blocks belong to this category.

- Shade blocks are not used as domain blocks:
- A shade block remains a shade block under any of the block transformations.
- Hence, not included in the domain block.

Transformation Pool

- For each block μ, there is transformation which depends on whether μ is a shade, midrange or edge block.

If we have a domain block v, then if μ is a

- ✓ **Shade Block:** Approximate it by uniformly gray block whose gray level is average of pixel values of μ. For these blocks only single value is stored.

Transformation Pool…

- **Midrange Block:** It is composition of contrast scaling and luminance shift:

$$\Im(\wp\, \nu) = \alpha(\wp\, \nu) + \Delta g$$

Where α is contrast scaling factor that takes in the set {0.5, 0.6, 0.7, 0.8, 0.9, 1.0} and Δg so that average gray levels of range block and scaled domain blocks are the same.

Δg_i is computed so that the average gray levels of the range block and the scaled domain block are approximately the same, i.e.:

$$\Delta g_i = \overline{\mu}_{\rceil R_i} - \alpha_i \overline{\mu}_{\mid D_j}.$$

Transformation pool...

- **Edge Block:** It is composition of contrast scaling, luminance shift and an isometry

$$\Im(\wp\, v) = i(\alpha(\wp\, v)) + \Delta g$$

A is chosen such that μ and $\wp(v)$ have same dynamic range.

$$\alpha = \min[\frac{dr(\mu)}{dr(\wp\, v)}, \alpha_{\max}]$$

Where, dynamic range(dr)= (highest pixel value – lowest pixel value +1) of the block under consideration.

A is computed so that μ and ρv have the same dynamic range.

α so computed is quantized to nearest value in the set {0.5, 0.6, 0.7, 0.8, 0.9, 1.0}. Then Δg is computed so that average gray levels of range block and scaled domain blocks are the same.

- **The one among the eight isometries {i} $_{0 \le n \le 7}$ which minimizes the distortion is selected.**

Algorithm to search for Edge, midrange & shade block

- **Edge block:** For classification of the edge block, let f(x,y) be the grey level of image at (x,y). The gradient of at the point (x,y) is

$$\nabla f = (G_x, G_y) = \left(\frac{\partial f}{\partial x}, \frac{\partial f}{\partial y}\right).$$

Let, $F = mag(\nabla f) = [G_x^2 + G_y^2]^{1/2}$, denote maximum rate of increase in f(x,y) per unit distance in the direction of ∇f . We approximate F by $[|G_x| + |G_y|]$ as this is simple to implement. The quantities G_x and G_y are computed as:

$G_x = (Z_7+2Z_8+Z_9) - (Z_1+2Z_2+Z_3)$

$G_y = (Z_3+2Z_6+Z_9) - (Z_1+2Z_4+Z_7)$

Where, Z_1........Z_9 are grey levels of 3*3 part of the image.

Z_1	Z_2	Z_3
Z_4	Z_5	Z_6
Z_7	Z_8	Z_9

And G_x and G_y are derivatives of Z_5.

Algorithm for Edge Block

- Whenever, the ∇f crosses a threshold value (30: found through experimentations)) a counter 'k' is incremented. If k> the block size, the block is designated as an edge block.

Algorithm for Midrange Block

- **Approach based on the variance of block under consideration.**

- **Algorithm**

Var = variance of block

r= 1- 1/(1+var)

If (r<0.7), block is classified as midrange.

- The bound 0.7 on r is found by experimentation.

Algorithm for Shade Block

- Dynamic range 'dr' of the block is computed.
- For (dr < 15) block classified as Shade block.
- (dr < 15) is determined after experimentation with images, which gave large number of shade blocks.

Shade Block takes minimum time for classification & requires minimum storage space ⇒ Only average grey level of block is to be stored.

Outline

- Introduction to Mobius Transformations
- Introduction to Fractal Image Compression
- Mathematical Foundations of Image Compression
- Fundamental Principles of Image Compression
- Design of Digital Image encoder
- Design of Fractal Block coding System
- Encoding of Digital Image using Fractals
- **Output File**
- Stochastic Image compression
- Decoding of Image from fractal code
- Results obtained for Lena & Baboon Images
- Comparison between Stochastic & Non stochastic Algorithms
- Differences between different Fractal Coding Schemes
- Specifications/ Parameters
- References

Output file

- Has 1 line of code for each range block of in the image.
- Number of integers in o/p file decide the type of block.

- **Shade block:** Only average grey level of block is stored.

- **Midrange block:**
  - Luminance shift
  - Contrast-scaling factor
  - Co-ordinates of the upper left corner of domain block are stored.

- **Edge block:**

  Consists of several lines of codes where each line consists of:
  - Luminance shift
  - Contrast-scaling factor
  - Co-ordinates of the upper left corner of the domain block
  - Isometry number

Output File Format

- File generated as the following format:

Line 1 {Image width, Image Length, Block size}
Line 2 {code of block 1}
……… Line n+1 {code of block n}

- Code of edge block ⇨ x, y, i, α, β
- Code of midrange block ⇨ x, ,y, α, β
- Code of Shade block ⇨ avg. of the block

Position of the block in original image is as follows:

1	2	3	4
5	6	7	8
…	…	…	…
n-3	n-2	n-1	n

Outline

- Introduction to Mobius Transformations
- Introduction to Fractal Image Compression
- Mathematical Foundations of Image Compression
- Fundamental Principles of Image Compression
- Design of Digital Image encoder
- Design of Fractal Block coding System
- Encoding of Digital Image using Fractals
- Output File
- **Stochastic Image compression**
- Decoding of Image from fractal code
- Results obtained for Lena & Baboon Images
- Comparison between Stochastic & Non stochastic Algorithms
- Differences between different Fractal Coding Schemes
- Specifications/ Parameters
- References

Drawbacks with Fractal Image Compression

- The large time taken for the compression.
- (e.g. for a 2x2 block of baboon time of compression was 54 minutes 07 sec).
- This large amount of time is due to:-
- Checking the eight isometries for each edge block.
- Reducing the number of isometries resulted in distortion of the image.

Stochastic Image Compression

- To overcome this difficulty, Use probabilistic measures .
- 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.

Highly successful approach-

- Without distorting the image, able to reduce the compression time by up to 80% .
- Further reduction in image size resulting in small storage requirement.

Probabilistic approach in O/p file

- Using probabilistic approach, only isometry number for certain number of (eg. 100) blocks need to be written.

Example:

Suppose, **4096** ⟹ **edge blocks** are present in a particular image,

Able to **reduce size of image further by** ⟹ **(4096 – 100) integer values.**

If, each integer takes **2 bytes** ⟹ file of image reduced by **7992 bytes**.

Outline

- Introduction to Mobius Transformations
- Introduction to Fractal Image Compression
- Mathematical Foundations of Image Compression
- Fundamental Principles of Image Compression
- Design of Digital Image encoder
- Design of Fractal Block coding System
- Encoding of Digital Image using Fractals
- Output File
- Stochastic Image compression
- **Decoding of Image from fractal code**
- **Results obtained for Lena & Baboon Images**
- Comparison between Stochastic & Non stochastic Algorithms
- Differences between different Fractal Coding Schemes
- Specifications/ Parameters
- References

Decoding of Image from Fractal Code

- This section addresses:
  - The decoding-reconstruction of an image from a fractal code.
  - Present and analyze coding simulations.

- Output file is read line by line.
- Process is repeated till all the lines of the file are processed.
- Whole process completes one iteration of the decoding algorithm.
- Generally, after completing 8 -22 such iterations, the sequence of image generated at each iteration finally converges to the required stable image.

  - Very promising results obtained for coding of 256 X 256 and 256 X 256 pixel digital images such as "Lena."

Results obtained for Lena with initial “face” image

Fig. Original Image

Fig. Three iterations of a code for “Lena,” applied on the initial “face” image.

Results obtained for Lena with initial grey image

Fig. Original Image

Fig. Three iterations of a code for "Lena," applied on the initial uniform grey image.

Results obtained for Lena

Fig. Original Image

Fig. First eight iterations for Lena Image

No.of Itr.	1	2	3	4	5	6	7	8
SNR	36.1	35.7	35.7	37.0	41.4	41.9	42.4	42.1

Table1. SNR (signal to Noise ratio) values for Lena Image.

Table2. Image specification (Lena)

Image type	Resolution	size (in bytes)	Gray levels	Range Blocks	compression ratio
.Pgm	256 x 256	266312	256	4 x 4	1:8

Results obtained for Baboon

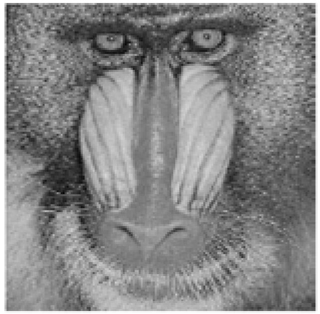


Fig. Original Image

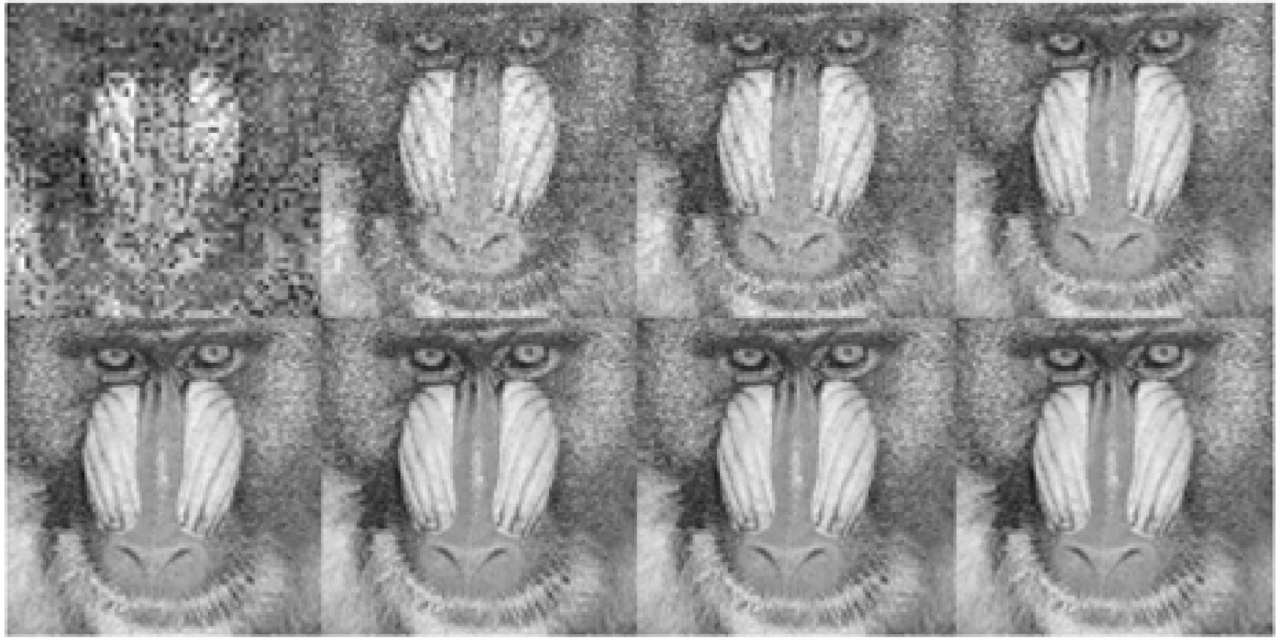


Fig. First few iterations for Baboon Image

No.of Itr.	1	4	10	14	16	20	24	30
SNR	35.0	36.3	37.9	38.1	38.2	38.3	38.5	38.6

Table1. SNR (signal to Noise ratio) values for Baboon Image.

Table2. Image specification (Baboon)

Image type	Resolution	size (in bytes)	Gray levels	Range Blocks	compres ratio
.Pgm	256 x 256	266312	256	4 x 4	1:6

Results... (Baboon)

- Baboon image has more number of Edge blocks → Requires Large Number of Iterations.
- Above decompressed images of Lena & Baboon show extremely good reproduction of edge blocks and shade blocks.
- Some blockiness left can be completely removed by using size of range block as 2 x 2.
- However, then Compression is ~ 50% of Original image.
- Time of compression also increases.

Outline

- Introduction to Mobius Transformations
- Introduction to Fractal Image Compression
- Mathematical Foundations of Image Compression
- Fundamental Principles of Image Compression
- Stochastic Image compression
- Design of Digital Image encoder
- Design of Fractal Block coding System
- Encoding of Digital Image using Fractals
- Output File
- Decoding of Image from fractal code
- Results obtained for Lena & Baboon Images
- **Comparison between Stochastic & Non stochastic Algorithms**
- **Differences between different Fractal Coding Schemes**
- **Specifications/ Parameters**
- **References**

Comparison b/w Stochastic & non-stochastic algorithms

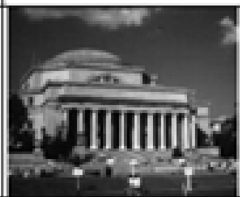
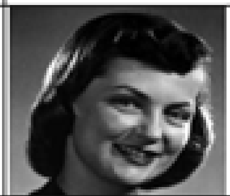
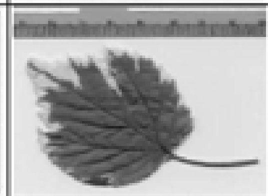
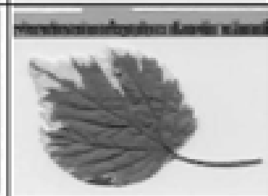
Images	Peppers	Columbia	Face	Madhu	Leaf	Cat
ORIGINAL IMAGE						
D1						
D2						
SNR VALUE	S1= 42.70; ITR = 10	S1= 43.35; ITR=10	S1= 43.73; ITR=10	S1= 43.95; ITR = 10	S1= 41.00; ITR=10	S1= 46.86; ITR = 12
	S2= 43.20; ITR =10	S2 = 43.58; ITR=10	S2= 46.94; ITR = 8	S2= 43.74; ITR =10	S2= 43.07; ITR = 10	S2= 46.64; ITR = 8

Fig. comparison b/w SNR values.

*__D1__,: *Final decoded image without using probabilistic theory* *__D2__: *Final decoded image using probabilistic theory.*
*__S1__: *SNR value of image without using probabilistic theory.* *__S2__: *SNR value of image using probabilistic theory.*
*__ITR__: *No of Iterations used to decode the image.*

Comparison in terms of Compression time & compression Ratio

IMAGE 266132 bytes	Block Size	Image1	Image2	T1	T2	PR
Lena	4 x 4	39795 bytes	35280 Bytes	3 Min 32 s	1 Min 23 s	62.95%
	2 x 2	136984 bytes	123618 bytes	15 Min 43 s	4 Min 38 s	71.6%
Baboon	4 x 4	56433 bytes	48988 Bytes	8 Min 29 s	2 Min 42 s	70.8%
	2 x 2	209574 bytes	182703 bytes	54 Min 7 s	19 Min 23s	64.4%
Peppers	4 x 4	35374 bytes	31934 Bytes	2 Min 30 s	1 Min 00 s	56.52%
	2 x 2	121001 bytes	111065 Bytes	10 Min 07 s	3 Min 44 s	65.83%
Columbia	4 x 4	39055 Bytes	34639 Bytes	3 Min 26 s	1 Min 21 s	62.88%
	2 x 2	137436 bytes	124160 Bytes	17 Min 17 s	5 Min 04 s	70.65%
Face	4 x 4	32742 bytes	29766 Bytes	1 Min 55 s	0 Min 40 s	74.19%
	2 x 2	114592 bytes	104200 Bytes	8 Min 15 s	2 Min 31 s	71.65%
Madhu	4 x 4	41504 bytes	37001 Bytes	4 Min 16 s	1 Min 18 s	71.63%
	2 x 2	141655 bytes	128759 Bytes	19 Min 19 s	5 Min 39 s	71.91%
Leaf	4 x 4	33327 bytes	30000 Bytes	2 Min 15 s	0 Min 41 s	80.93%
	2 x 2	114911 bytes	105893 Bytes	7 Min 46 s	2 Min 20 s	70.50%
Cat	4 x 4	49767 bytes	43908 Bytes	5 Min 57 s	1 Min 56 s	72.00%
	2 x 2	169147 bytes	152458 Bytes	30 Min 43 s	8 Min 54 s	71.90%

***T1**: *Time taken for compression of image without using probabilistic theory.*
***T2**: *Time taken for compression of image using probabilistic theory.*
***Image1**: *compression of image (in bytes) without using probabilistic theory.*
***Image2**: *compression of image (in bytes) using probabilistic theory.*
***PR**: *Percentage reduction in time using the probabilistic theory as compared to the non- probabilistic the*

Advantage of Stochastic Process over Non-stochastic Processes

- Stochastic Image Compression able to reduce time taken for compression by 55%-80% as compared to non-stochastic algorithm.
- Compression Ratio ranges between 60%-80%, too greater than compression ration achieved using non-stochastic algorithm.

Achieves same or better SNR values while reducing Compression time and storage space significantly.

Fractal image compression has been assessed suitable for spacecraft images. Relevant aspects from spacecraft to ground station has been surveyed using this.

Difference b/w various Fractal Coding Schemes

Difference between various existing Fractal coding schemes can be classified into following categories:

- The partition imposed on the image by the range blocks.
- The composition of pool domain blocks, which is restricted to some extend by the range partition.
- The class of transformations applied to the domain blocks.
- The type of search used in locating suitable domain blocks.
- The quantization of the transform parameters and any subsequent entropy coding.

Specifications/ Parameters

- **Encoding specifications:**
  - The partition of Image support.
  - The pool of Domain blocks.
  - The choice of distortion measure used for block matching.
  - The pool of transformations.

- **System Performance:**
  - The global proportions of Shade, Edge and Midrange blocks in the original image.
  - The Bit rate.
  - The SNR between the original and the decoded image.
  - Subjective remarks about the fidelity of the decoded image to the original.

References

Books

- "*Fractal Image Compression and the Self-Affinity Assumption: A Stochastic Signal Modelling Perspective*", B. Wohlberg, PhD Thesis, University of Cape Town, August 1996.
- "*Fractals Everywhere*", M.F. Barnsley, A.P. Professional, 2nd Edition, Cambridge, MA, 1993.

Research Papers

- A.E. Jacquain, "*Image coding based on a Fractal Theory of Iterated Contractive Image Transformations*", IEEE transactions on Image Processing, vol. 1, Jan. 1992, pp. 18-30.
- A. Kapoor, K. Arora, A. Jain, G.P. Kapoor, "*Stochastic Image compression using Fractals*", Proceedings of International conference on Information Technology (ITCC '03), Apr. 28-30th , pp. 574-579.

Questions & Queries !!!

Thank You!!!