Fractals

Keshav Malpani 2017MT10213
Vivek Muskan 2017MT10755

Supervisor: Dr. Amit Priyadarshi

Project Objective

Pre Mid-Term

- Learn about the basics of Fractals, their formation and their properties.
- Approximating Fractals up to n iterations of formation
- Learn about the fractional nature of dimension in case of fractals
- Find the dimensions of various Fractals

Post Mid-Term

- Explore the theoretical and practical aspects of the Mandelbrot and Julia Sets
- Study and use Fractal Image compression

Topics Covered

(Till Mid-Term)

- Fractal Basics: Introduction and Motivations
- Prerequisite Study:
  - Metric Spaces
  - Measure Theory Basics
- Fractal Examples:
  - Middle Third Cantor Set
  - Sierpinski’s Triangle
  - Koch’s Curve…
- IFS and Chaos Game
- Plotting Fractals using IFS
- Fractal Dimensions
- Similarity Dimension Calculations
- Box Counting Dimension
  - Theoretical Formulation
  - Practical Aspect

The Julia and Mandelbrot Sets

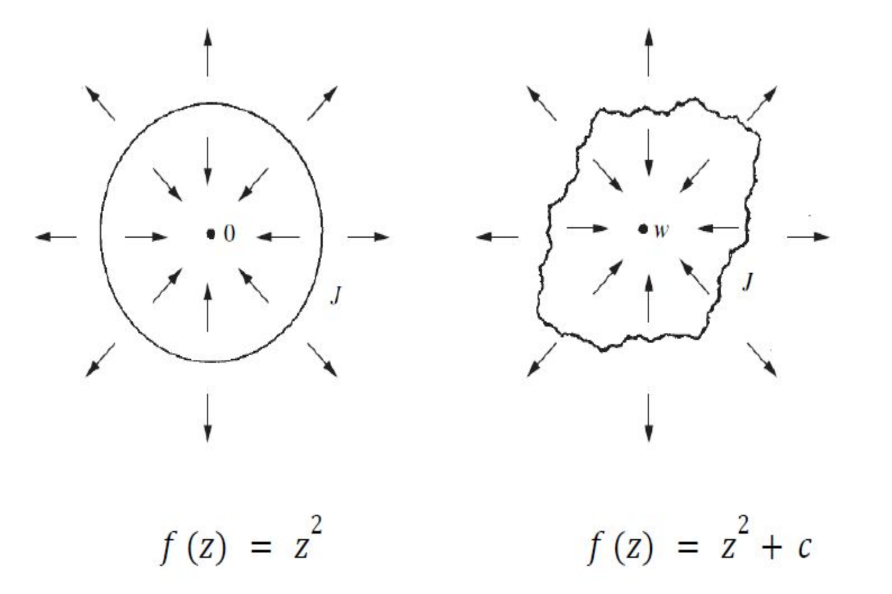
- Highly intricate sets generated using a very simple process.
- Explored Functions are of the form $\mathbf{f(z) = z^n + c}$.
- K-fold composition of the function after beginning at some starting point $\mathbf{f \circ f \circ \ldots \circ f}$.
- Julia Set accounts the behavior of these iterates for large *k* values for fixed Complex Constants *c*
- Filled-in Julia Set: $\boldsymbol{K(f) = \{z \in \mathbb{C} : |f^k(z)| \nrightarrow \infty \text{ as } k \to \infty\}}$
- Julia Set is the boundary of the Filled-in Julia Set.
- Mandelbrot Set M is the set of parameters for which the corresponding Julia Sets are connected.
- We will work on the Quadratic Functions to understand the properties of these Sets

Special Case of $f(z) = z^2$

- $f(z) = z^2 \quad \Rightarrow \quad f^k(z) = z^{(2^k)}$
- $|f^k(z)| \to 0$ as $k \to \infty$ if $|z| < 1$
- $|f^k(z)| \to \infty$ as $k \to \infty$ if $|z| > 1$
- But for $|z|=1$, $f^k(z)$ remains on the circle $|z|=1 \;\forall\; k$
- Filled in Julia Set = $K(f) = \{z \in \mathbb{C} : |z| \leq 1\}$
- Julia Set = Boundary of Filled in Julia Set = $J(f) = \{z \in \mathbb{C} : |z| = 1\}$
- Not a Fractal!

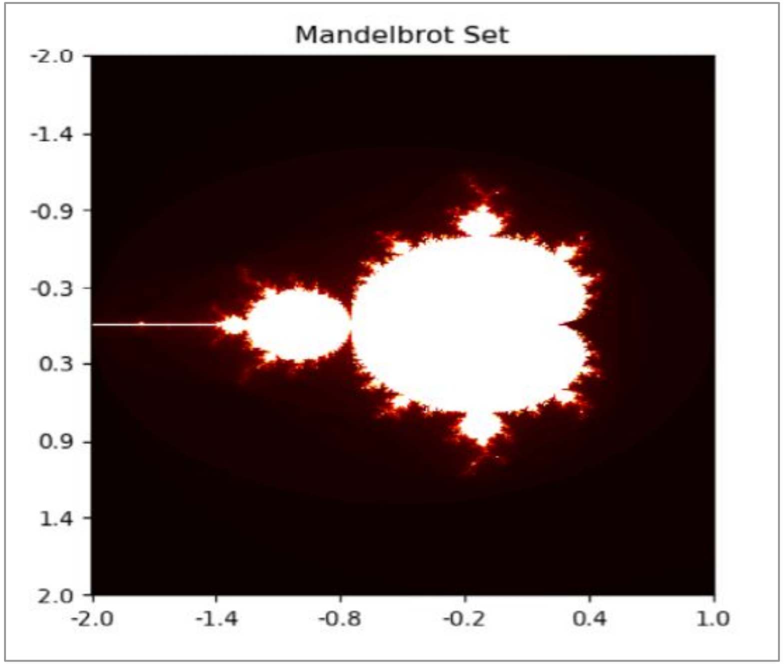
General Case: $f(z) = z^2 + c$

- $f(z) = z^2 + c$
- $f^k(z) \to w$ if z is small where w is a fixed point of f close to 0
- $|f^k(z)| \to \infty$ if z is large
- The Julia Set is again the boundary between the two types of behaviors
- In this case it comes out as a fractal!



The Mandelbrot Set

- $f_c(z) = z^2 + c$
- Mandelbrot Set $M = \{ c \in \not{C} : J(f_c) \text{ is connected} \}$
- ***Theorem:***
  - $M = \{ c \in \not{C} : J(f_c) \text{ is connected} \}$
  - $M = \{ c \in \not{C} : \{ \{f_c^{\,k}(0)\}_{k=0}^{\infty} \text{ is bounded} \}$
  - $M = \{ c \in \not{C} : \{ |f_c^{\,k}(0)| < \infty \text{ as } k \to \infty \}$
- *Idea for plotting:*
  - Use last definition
  - Compute iterates $f_c^{\,k}(0)$ for $\mathbf{n_0}$ number of iterations
  - Decide a bound r
  - If $|f_c^{\,k}(0)| > r$, $c \notin M$ otherwise $c \in M$
  - How to decide the bound???



$M^C = \{ c \in ¢ : \{ |f_c^k(0)| > 2 \text{ for some } k\}$

Lemma: If $\varepsilon > 0$ and $|z| > \max(2 + \varepsilon, |z|)$, then $|f_c(z)| \geq |z|(1 + \varepsilon)$, deducing that if $|c| > 2$ then $c \notin M$.

Proof:

- The first part can be proved using the inequality $|a+b| \geq (|a| - |b|)$
- Now $|c| > 2 \Rightarrow \exists\ \varepsilon$ s.t $|c| > 2 + \varepsilon$
- $f_c(0) = c$ and using part 1's inequality $|f_c^k(0)| \geq (1+\varepsilon)^k \to \infty \Rightarrow c \in M^C$

Theorem: $M^C = \{ c \in ¢ : \{ |f_c^k(0)| > 2 \text{ for some } k\}$

Proof: Let c be such that $f_c^k(0) > 2$ $for\ some\ k$.

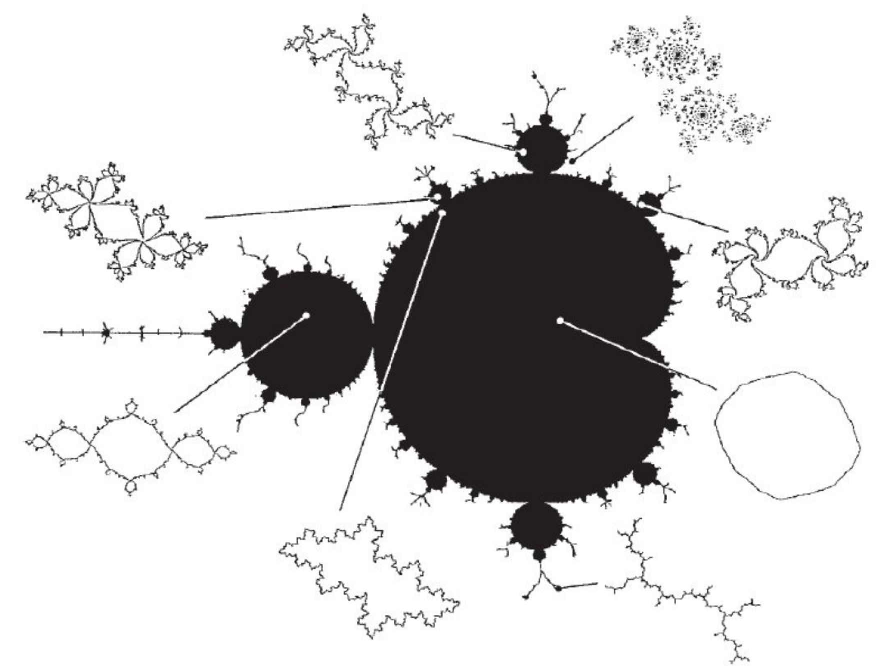
1. $|c| > 2 \Rightarrow c \notin M$ by previous result.
2. $|c| \leq 2 \Rightarrow \exists\, \epsilon > 0$such that $|f_c^k(0)| > 2 + \epsilon > |c|$

$\Rightarrow |f_c^{k+n}(0)| = |f_c^n(f_c^k(0))| \geq (1 + \epsilon)^n |f_c^k(0)| \to \infty$. So $c \notin M$.

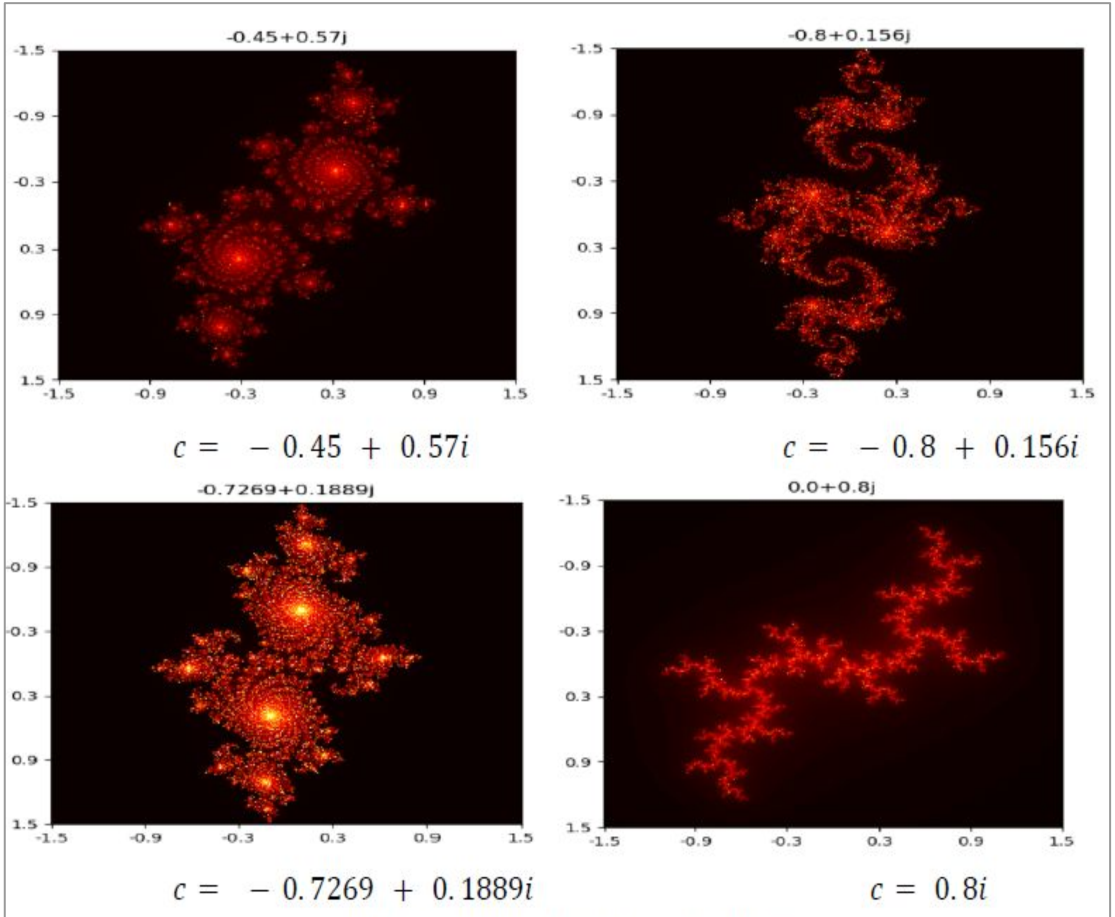
Conversely, if $c \notin M \Rightarrow |f_c^k(0)| \to \infty$, so $|f_c^k(0)| > 2$for some k.

Julia Sets for the Quadratic Poly. $f_c(z) = z^2 + c$

- Julia Sets explore the various parts of the Mandelbrot Set in detail.
- To understand the nature of the Julia Sets for various complex constants, the sets were plotted.
- Plotting algorithm very similar to the Mandelbrot Plotting.
- For Julia Sets, **c** was fixed and **z** took the value of pixel, instead of the initial value of **0.**



Julia Sets: Plotting Results



Julia Sets Plotted for various c values

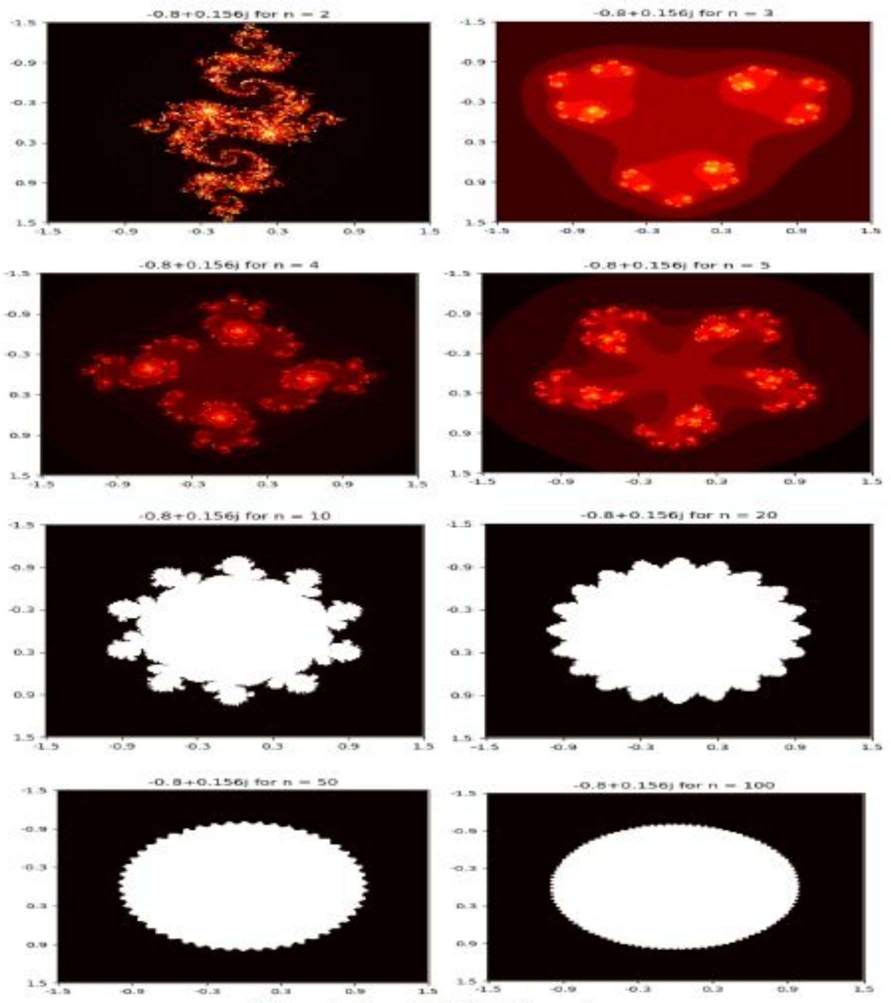
Higher Order Julia Sets: Plotting Results

$f_c(z) = z^n + c$

$c = -0.8 + 0.156i$

Observations:

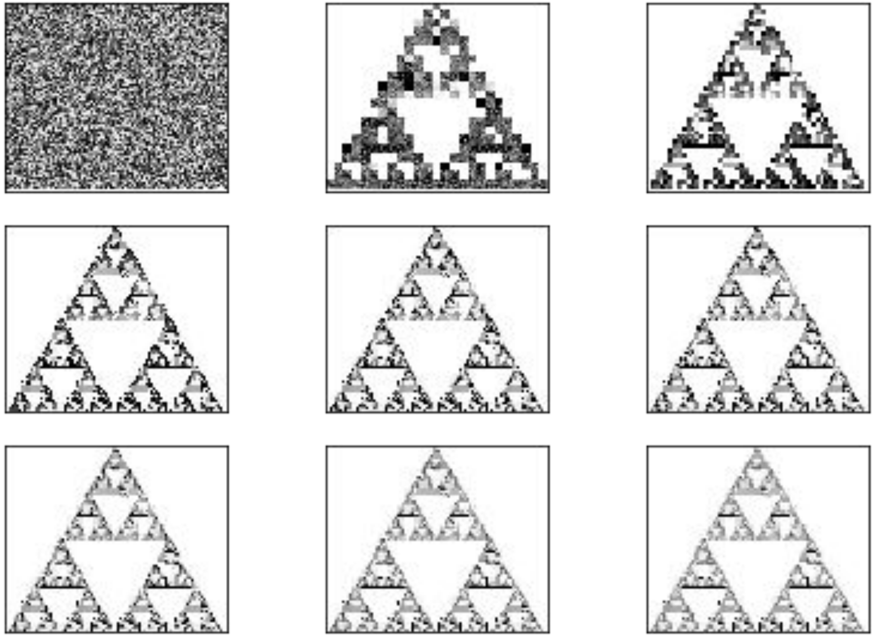
- The number of copies are varying with the degree of our polynomial.
- As $n \to \infty$, the Julia Set converges to Unit Circle.



Higher Order Julia Sets for various n

Fractal Image Compression

- It is a lossy compression
- It uses the concept of contraction mapping
- Uses the fact that in certain images, parts of the image resemble other parts of the same image



Obtaining the image of sierpinski triangle using its IFS

The Inverse Problem and Collage Theorem

The Inverse Problem

We know that every contractive function has one unique fixed point. But given a point x (image), does there exist a contractive function f such that x is its fixed point?

- NP Hard → Slow
- Incomplete space → Lossy

Collage Theorem

Let (X, d) be a complete metric space and $f : D \to X$, $D \subset X$, a contractive mapping with contractivity factor s, and a unique fixed point x_o . Then,

$$d(x, x_o) \leq d(x, f(x)) / (1-s)$$

This theorem tells us that if we find a contraction f such that f(x) is near to x then we are sure that the fixed point of f is also near to x

Mathematical Formulation of Problem

- Image $I \in R^{h \times w}$ & $f = \{f_1, f_2, \ldots, f_n\}$
- $f_i : R^{a \times b} \rightarrow R^{c \times d}$, with domain D_i & range R_i
- Ranges creates partition on image I
- Define metric (frobenius norm) for (*hxw*) pixel matrix

$$d(x, y) = \sqrt{\sum_{i=1}^{h} \sum_{j=1}^{w} (x_{ij} - y_{ij})^2}$$

So if $d(R_i, f_i(D_i)) < \epsilon \ \forall\, i$ then,

$$d(I, f(I)) = d\left(\bigcup_{i=1}^{n} R_i, \bigcup_{i=1}^{n} f_i(D_i)\right) = \sum_{i=1}^{n} d(R_i, f_i(D_i)) < n\epsilon$$

Using collage theorem will imply:

$$d(I, I_o) \leq \frac{d(I, f(I))}{1-s} = \frac{n\epsilon}{1-s}$$ where I_o is the fixed point of IFS f.

Domain blocks

D_1	D_2	D_3	D_4
D_5	D_6	D_7	D_8
D_9	D_{10}	D_{11}	D_{12}
D_{13}	D_{14}	D_{15}	D_{16}

Range blocks

R_1	R_2	R_3	R_4	R_5	R_6	R_7	R_8
R_9	R_{10}	R_{11}	R_{12}	R_{13}	R_{14}	R_{15}	R_{16}
R_{17}	R_{18}	R_{19}	R_{20}	R_{21}	R_{22}	R_{23}	R_{24}
R_{25}	R_{26}	R_{27}	R_{28}	R_{29}	R_{30}	R_{31}	R_{32}
R_{33}	R_{34}	R_{35}	R_{36}	R_{37}	R_{38}	R_{39}	R_{40}
R_{41}	R_{42}	R_{43}	R_{44}	R_{45}	R_{46}	R_{47}	R_{48}
R_{49}	R_{50}	R_{51}	R_{52}	R_{53}	R_{54}	R_{55}	R_{56}
R_{57}	R_{58}	R_{59}	R_{60}	R_{61}	R_{62}	R_{63}	R_{64}

Affine Transformation and Encoding

Affine Transformation

An affine transformation is any transformation that preserves collinearity

$$w_i \begin{pmatrix} x \\ y \\ z \end{pmatrix} = \begin{pmatrix} a_i & b_i & 0 \\ c_i & d_i & 0 \\ 0 & 0 & \alpha_i \end{pmatrix} \begin{pmatrix} x \\ y \\ z \end{pmatrix} + \begin{pmatrix} e_i \\ f_i \\ \beta_i \end{pmatrix}$$

- $a_i, b_i, c_i, d_i \in \{0, ½\} \rightarrow$ contraction ratio
- α_i and β_i are contrast and brightness
- e_i and f_i are translation parameters

Encoding of Contraction Map

- Contraction Ratio = ¼ ⟶ fixed
- D_i = 8x8 & R_i = 4x4
- 1 bit ⟶ flip {1, -1}
- 2 bit ⟶ rotation {0, 90, 180, 270}
- 5 bit ⟶ contrast ($-1.2 \leq \alpha_i \leq 1.2$)
- 8 bit ⟶ brightness ($0 \leq \beta_i \leq 256$)
- 8 bit ⟶ each translation x, y (0 to 256)

Total : 1+2+5+8+8+8 = 32 bit for each transformation

Algorithm

Compression Algorithm

Input : Image I, Domain size : 8x8, Range size : 4x4
Output : Encoded transformation
begin

    Segment I in Domain of 8x8
    Partition I in Range of 4x4

    Encoded transformation = []
    for D in Domain pool:
        Generate all possible transformations using :
            *f (D) = contrast * rotate(flip (reduce(D))) + brightness*

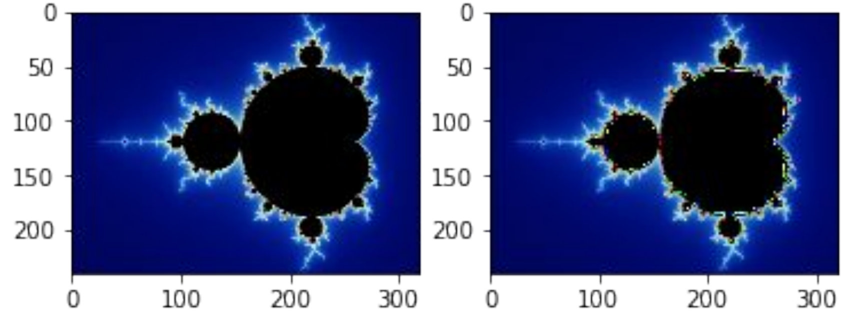
        for R in Range pool:
            find *f* with minimum *d(f(D), R)*
            Encode *f* and insert it Encoded transformation
    return Encoded transformation
end

Decompression Algorithm

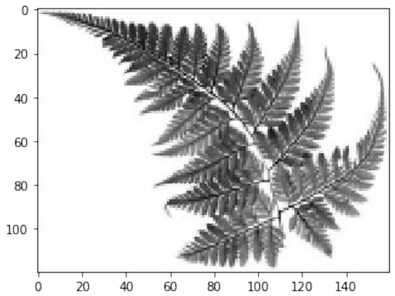
Input : Encoded Transformations, No of steps
Output : Image I
begin

    Decode the Encoded transformations
    Take any initial random image I
    for i in range (No of steps):
        for *f* in decoded transformations:
            I = f (I)
    return I
end

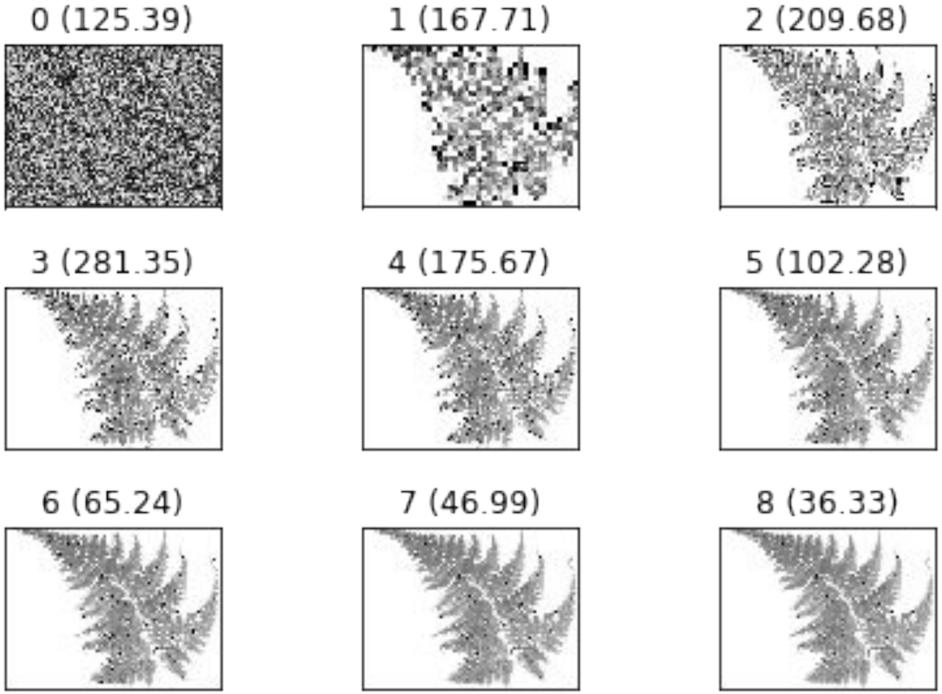
Result and Discussion I (Sharp Transition)



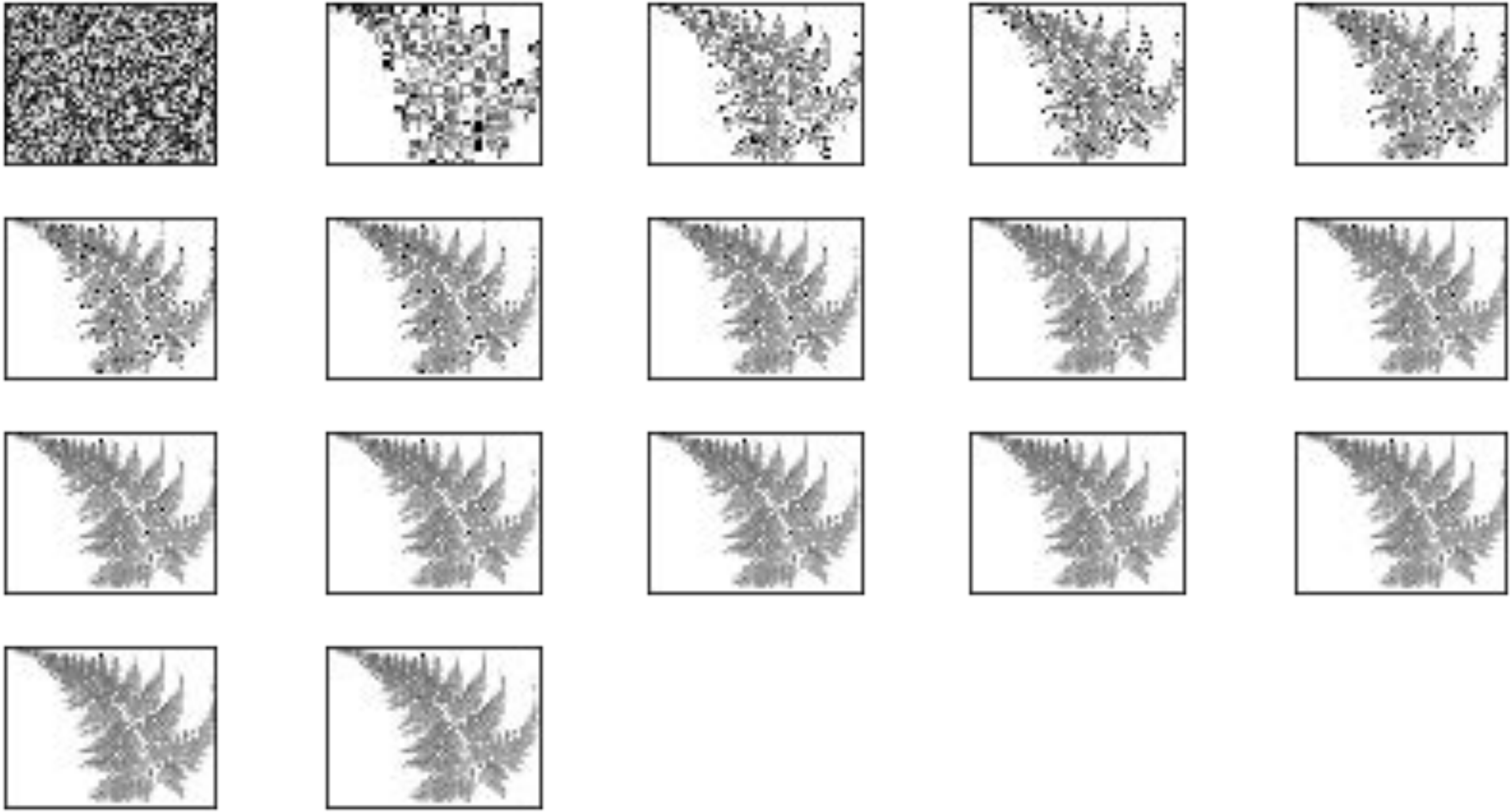
MANDELBROT : RGB input and output



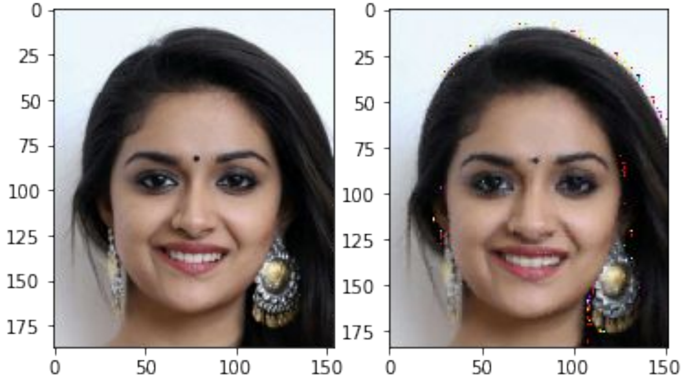
FERN : Greyscale Input



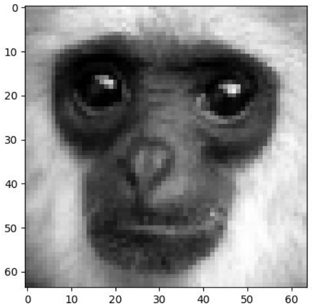
FERN : Grayscale output upto 8 iteration



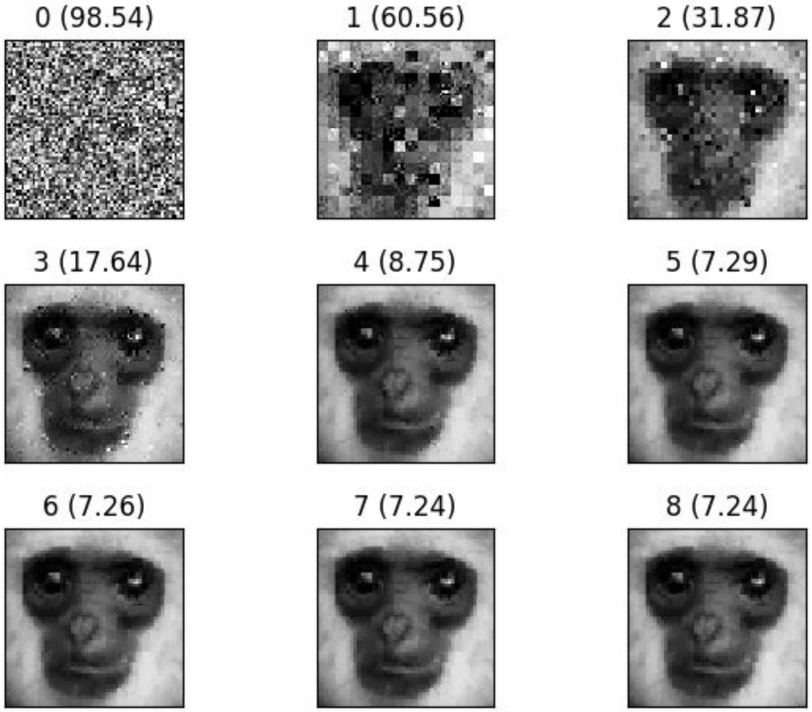
Result and Discussion II (Smooth Transition)



KEERTHY : RGB input and output



MONKEY : Greyscale Input



MONKEY : Grayscale output upto 8 iteration

Fractal Compression and Others

PNG	***JPEG***	***Fractal Compression***
Lossless	Lossy	Lossy
Low Compression Ratio	High Compression Ratio	High Compression Ratio
Uses DEFLATE algorithm and Huffman Coding, it supports palette based grayscale and RGB images	Uses visual effect that human are more sensitive to, like luminance over chrominance	Uses the fact that in certain images, parts of the image resemble other parts of the same image
Better Choice for images with text, line art or sharp transitions	It work better for all types of images	Better choice for images with repetitive patterns and natural scene
Quick compression and decompression	Quick compression and decompression	Compression is very slow and even decompression can be slow for specific type of images
Widely used	New version of JPEG are widely used	It is yet in a development phase

Conclusion

- We have learnt about the basics of Fractals, their nature and some of their properties.
- We have been successful in finding the dimensions of various fractals along with plotting them to have a visualization of things going on in an approximation.
- We have also delved a little into the theory of the Mandelbrot and Julia Sets and used it to plot them to further understand their structures
- To understand the application of fractals, we studied and implemented fractal image compression and compared it with existing image compression techniques.

Proof of Collage Theorem**

Proof : Note that,

$$
\begin{aligned}
d(f^{m-1}(x), f^{m}(x)) &\leq s \,.\, d(f^{m-2}(x), f^{m-1}(x)) \quad \text{(since } f \text{ has a contractivity factor } s\text{)} \\
&\leq s^2 \,.\, d(f^{m-3}(x), f^{m-2}(x)) \\
&\vdots \\
&\leq s^{m-1} \,.\, d(x, f(x))
\end{aligned}
$$

Then,

$$
\begin{aligned}
d(x, x_o) = d(x, \lim_{n\to\infty} f^n(x)) &= \lim_{n\to\infty} d(x, f^n(x)) \\
&\leq \lim_{n\to\infty} d(x, f(x)) + d(f(x), f^2(x)) + .. + d(f^{n-1}(x), f^n(x)) \\
&\leq \lim_{n\to\infty} d(x, f(x)) \,(1 + s + s^2 + \ldots + s^{n-1}) \\
&\leq \frac{d(x, f(x))}{1-s}
\end{aligned}
$$

Proof of Contraction Mapping Theorem**

Contraction Mapping Theorem: *A contraction mapping f on a complete non-empty metric space S has a unique fixed point.*

Proof. First, there is at most one fixed point. If x and y are both fixed points, then $d(x, y) = d(f(x), f(y)) \le rd(x, y)$. But $0 \le r < 1$, so this is impossible if $d(x, y) > 0$. So $d(x, y) = 0$, hence $x = y$.

Now let x_0 be any point of S. Then define recursively

$$x_{n+1} = f(x_n) \text{ for } n \ge 0.$$

We claim that (x_n) is a Cauchy sequence. Write $a = (x_0, x_1)$. It follows by induction that $d(x_{n+1}, x_n) \le ar^n$. But then, if $m < n$ we have

$$d(x_m, x_n) \le \sum_{j=m}^{n-1} d(x_{j+1}, x_j) \le \sum_{j=m}^{n-1} ar^n = (ar^m - ar^n) / (1-r)$$

$$= ar^m (1 - ar^{n-m}) / (1-r) \le ar^m / (1-r).$$

Therefore, if $\varepsilon > 0$ is given, choose N large enough that $ar^N / (1 - r) < \varepsilon$. Then, for $n, m \ge N$, we have $d(x_m, x_n) < \varepsilon$. Now S is complete, and (x_n) is a Cauchy sequence, so it converges. Let x be the limit. Now f is continuous, so from $x_n \to x$ follows also $f(x_n) \to f(x)$. But $f(x_n) = x_{n+1}$, so $f(x_n) \to$ x. Therefore the two limits are equal, $x = f(x)$, so x is a fixed point.

References

1. Fractal Geometry by Kenneth Falconer, Third Edition, Wiley Publications
2. Measure, Topology and Fractal Geometry by Gerald Edgar, Second Edition, Springer Publications
3. Prem Melville, Vinhthuy Phan (2000), “ Fractal Compression”, The University of Memphis
4. Pierre Vigier (2018), “ Fractal Image Compression | pvigier's “
5. Bekir Karlik (2015), “Comparison of Image Compression Techniques”, McGill University
6. Paula Aguilera, “Comparison of Different Image Compression Format”

Thank you !

Questions?