The Fractal Inverse Problem For Audio Compression

Martin Pham

December 2017

Introduction

Motivating Example: Barnsley Fern

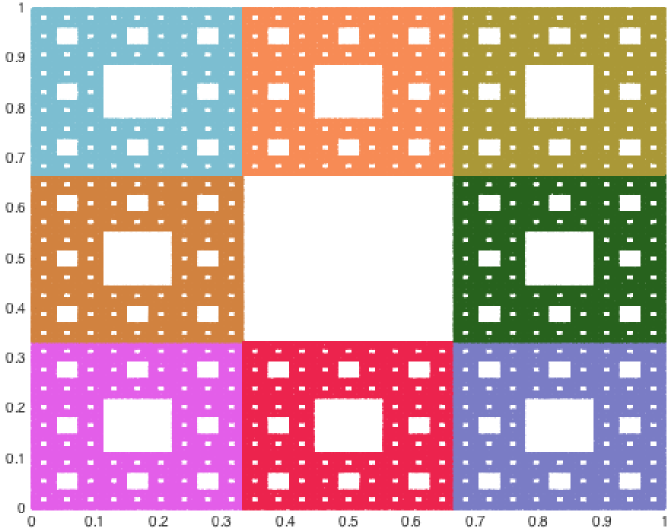
Fractal Inverse Problem as Compression

Further Directions

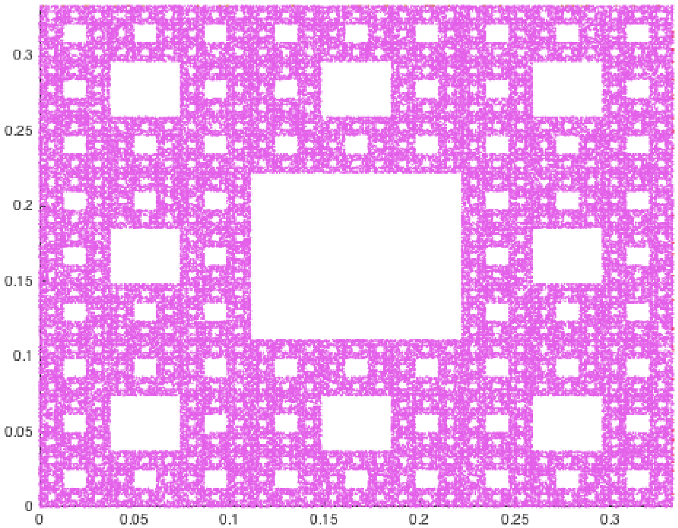
"Fractal geometry will make you see everything differently. You risk the loss of your childhood vision of clouds, forests, galaxies, leaves, feathers, flowers, rocks, mountains, torrents of water, carpets, bricks, and much else besides. Never again will your interpretation of these things be quite the same." -Michael Barnsley

Consider a fractal as an object that exhibits **self-similarity** and **scale-free complexity**

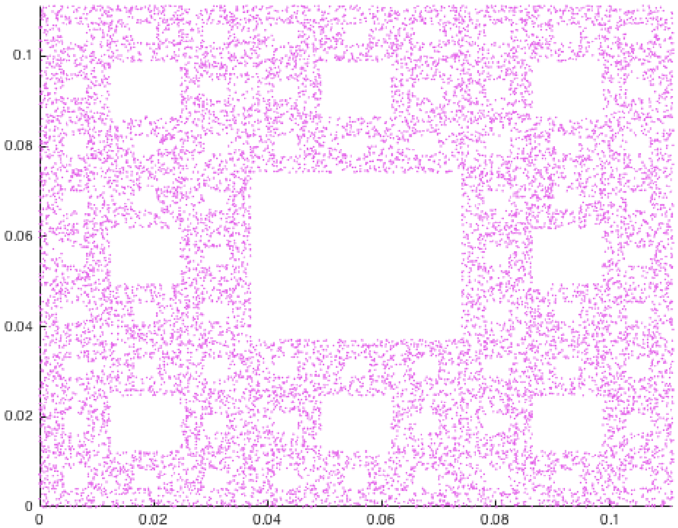
Sierpinski Carpet



Sierpinski Carpet



Sierpinski Carpet



Introduction: IFS

Michael Barnsley (1981) introduced iterated function systems (IFS) as a method for generating fractal objects

Introduction: IFS

Michael Barnsley (1981) introduced iterated function systems (IFS) as a method for generating fractal objects

A set of contractive mappings on a complete metric space defines an IFS

Introduction: IFS

Michael Barnsley (1981) introduced iterated function systems (IFS) as a method for generating fractal objects

A set of contractive mappings on a complete metric space defines an IFS

The IFS defines an associated Hutchinson operator

Introduction: Hutchinson operator

The operator provides a notion of "combining" the actions of the set of mappings

Introduction: Hutchinson operator

The operator provides a notion of "combining" the actions of the set of mappings

It has an attractive fixed point, called a fractal because it is invariant under the operator (i.e. self-similarity)

Within nature, nothing has entirely scale-free complexity

Within nature, nothing has entirely scale-free complexity

Can't expect the form of a tree to infinitely scale down, at some point it becomes a biological cell

Within nature, nothing has entirely scale-free complexity

Can't expect the form of a tree to infinitely scale down, at some point it becomes a biological cell

The same may be said for images and audio signals

Local IFS method is more forgiving, suspends scale-free complexity at your desired resolution

Local IFS method is more forgiving, suspends scale-free complexity at your desired resolution

Use larger subsections of a signal in order to approximate smaller subsections via a contractive affine transformation

Introduction: Inverse Problem

Fractal compression may be posed as an inverse problem:

What set of parameters defines an IFS with a fixed point that well approximates our compression target?

Introduction: Inverse Problem

Fractal compression may be posed as an inverse problem:

What set of parameters defines an IFS with a fixed point that well approximates our compression target?

This is different from other lossy compression, where an appropriate representation reduces redundancy of information

Introduction: Inverse Problem

Fractal compression may be posed as an inverse problem:

What set of parameters defines an IFS with a fixed point that well approximates our compression target?

This is different from other lossy compression, where an appropriate representation reduces redundancy of information

Here, we seek a *process* that will allow us to recover the target, not a *representation* with fewer degrees of freedom

Like looking for a treasure and having a map where no matter where you go you get to it

Introduction

Motivating Example: Barnsley Fern

Fractal Inverse Problem as Compression

Further Directions

Example: Barnsley Fern

Consider 4-map IFS with probabilities on $([-3,3] \times [0,10], \|\cdot\|)$

$$
\begin{aligned}
w_1(x) &= \begin{bmatrix} 0 & 0 \\ 0 & 0.16 \end{bmatrix} x, & p_1 &= 0.01 \\
w_2(x) &= \begin{bmatrix} 0.85 & 0.04 \\ -0.04 & 0.85 \end{bmatrix} x + \begin{bmatrix} 0 \\ 1.6 \end{bmatrix}, & p_2 &= 0.85 \\
w_3(x) &= \begin{bmatrix} 0.2 & -0.26 \\ 0.23 & 0.22 \end{bmatrix} x + \begin{bmatrix} 0 \\ 1.6 \end{bmatrix}, & p_3 &= 0.07 \\
w_4(x) &= \begin{bmatrix} -0.15 & 0.28 \\ 0.26 & 0.24 \end{bmatrix} x + \begin{bmatrix} 0 \\ 0.44 \end{bmatrix}, & p_4 &= 0.07
\end{aligned}
$$

The associated Hutchinson operator T is then

$$(Tu)(x) = \sum_{i=1}^{N} p_i u(w_i^{-1}(x))$$

This is a contractive operator that is guaranteed to have an attractive fixed point

$$\bar{u}(x) = (T\bar{u})(x)$$

It is this fractal object $\bar{u}$ that we will want to well approximate our target, we will seek the mappings that define T

The Barnsley Fern Attractor
(Demo)

So given some IFS we can repeatedly apply our operator to obtain an associated fractal object, but how can we construct such an IFS to give us what we want?

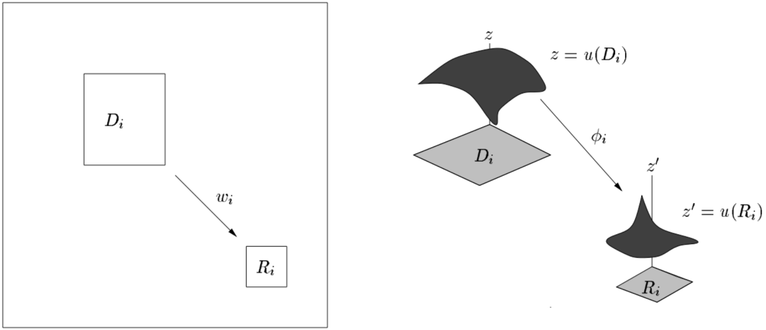
Introduction

Motivating Example: Barnsley Fern

Fractal Inverse Problem as Compression

Further Directions

Local IFS method (Jacquin, 1989) relaxes the global self-similarity constraint, requiring only a larger subsection of a signal to approximate a smaller subsection via a contractive transformation



Lena
(Demo)

Uncompressed

Compressed (15:1)

Can the same be performed for one-dimensional signals?

Can the same be performed for one-dimensional signals?

Yes

An Audio Fractal Encoder in Space

Encoding:

- Divide signal into two sets of range blocks with length N bits and domain blocks of length $2N$ bits
- For each range block, find the domain block and transformation that best approximates the current range block
- Save the domain block index and contractive mapping that best approximates that range block
- This defines the operator that is saved as an encoded file

An Audio Fractal Encoder in Space

Decoding:

- Start with *any* signal of appropriate length
- Divide signal into two sets of range blocks with length N bits and domain blocks of length $2N$ bits
- For each range block, get the associated encoded domain block index and contractive mapping
- Replace the range blocks with the approximations by the domain blocks
- Repeat application of operator until residual is below some desired threshold

Compression ratio of 2.7:1

or

What does a fractal sound like?

Further Directions: Wavelets

A wavelet multiresolution analysis (Mallat, 1989) of the signal provides a representation that is amenable to fractal methods

Further Directions: Wavelets

A wavelet multiresolution analysis (Mallat, 1989) of the signal provides a representation that is amenable to fractal methods

Wavelets have both a frequency and spatial component, and may be represented in the form of coefficient trees

Further Directions: Wavelets

- A wavelet multiresolution analysis (Mallat, 1989) of the signal provides a representation that is amenable to fractal methods
- Wavelets have both a frequency and spatial component, and may be represented in the form of coefficient trees
- These trees are self-similar and so compression may be formed as a fractal inverse problem on wavelet coefficients

Previous work (Wannamaker, 1997) has yielded a fractal-wavelet approach with up to $6 : 1$ compression ratio

Previous work (Wannamaker, 1997) has yielded a fractal-wavelet approach with up to $6:1$ compression ratio

Working in the wavelet domain opens up many more options for search and encoding strategies

Wavelet bases are usually constructed depending on application, perhaps a basis may be chosen depending on preliminary analysis of signal before encoding process

The question becomes:

In what wavelet basis up to what level of resolution do we need for fractal-wavelet methods to achieve certain compression ratios with acceptable audio fidelity?

Furthermore,

What kind of domain blocks are used to approximate the most number of range blocks?

Thanks