11

Image Compression: Iterated Function Systems

This chapter can be covered in one or two weeks of classes. If only one week is available then you can briefly cover the introduction (Section 11.1) and then explain in detail the concept of an attractor of an iterated function system (Section 11.3) by concentrating on the Sierpiński triangle (Example 11.5). Demonstrate the theorem that constructs affine transformations mapping three points on the plane to three points on the plane and discuss the particular affine transformations that will be used often in iterated function systems (Section 11.2). Explain Banach's fixed-point theorem stressing the point that the proof on $\mathbb{R}$ can be transposed, nearly word by word, to complete metric spaces (Section 11.4). Finally, discuss the intuition behind the Hausdorff distance (beginning of Section 11.5). If you wish to spend a second week, then you can deepen the discussion of the Hausdorff distance and work through a few of the proofs of its various properties (Section 11.5). This leaves sufficient time to discuss fractal dimensions (Section 11.6) and to explain briefly the construction of iterated function systems that allow for the reconstruction of actual photographs (Section 11.7). Sections 11.5, 11.6, and 11.7 are almost independent, so it is possible to treat Section 11.6 or 11.7 without having gone through the more difficult Section 11.5.
Another option for a one-week coverage is to discuss Sections 11.1 to 11.3 and to jump to 11.7, which explains how to adapt the technique to compression of real photographs.

11.1 Introduction

The easiest way to store an image in computer memory is to store the color of each individual pixel. However, a high-resolution photograph (many pixels) with accurate color (many data bits per pixel) would require an enormous of amount of computer memory. And videos, with many such images per second, would required even more.

With widespread adoption of digital cameras and the Internet, people are storing an ever larger number of images on their computers. It is thus critical that these images be stored efficiently so as not to take up an inordinate amount of space. Images on the web can be of lower resolution than digital photographs or large posters. And we are

C. Rousseau and Y. Saint-Aubin, *Mathematics and Technology*,
DOI: 10.1007/978-0-387-69216-6_11,

very interested in keeping their sizes small; no doubt you have already experienced slow loading images while browsing the web, even if the images are compressed.

There exist many image compression techniques. The commonly used JPEG (Joint Photographic Experts Group) format makes use of discrete Fourier techniques and is explored in Chapter 12. In this chapter we will concentrate on another technique: image compression using iterated function systems.

There was a great deal of hope and excitement over the possibilities of this technique when it was first introduced in the 1980s, spurring considerable research. Unfortunately, formats based on these techniques have not seen much success because the compression algorithms and the compression ratios are not good enough. However, these techniques continue to be researched and might yet see improvements. We have decided to present these methods for several reasons. First, it is easy to show the underlying mathematics at work, which rely on Banach's powerful fixed-point theorem (the fixed point of the theorem referring to the attractor of an operator). Moreover, the method uses fractals, which we demonstrate how to construct in a very simple manner as fixed points of operators. That such complicated structures can be described through such simple constructions is a striking demonstration of the power and elegance of mathematics; it shows that if we look at an object from just the right point of view, everything is simplified, allowing us to understand its structure.

We stated above that the easiest way to store a picture is simply to store the color associated with each pixel, an approach that is far from efficient. How to do better? Suppose that we were to draw a profile of a city (Figure 11.1). Instead of storing the actual pixels, we could store the underlying geometric constructs, allowing us to reconstruct it:

- all line segments,
- all circular arcs,
- etc.

We have represented the image as a union of known geometric objects.

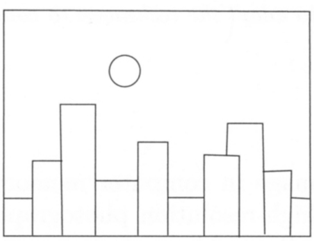


Fig. 11.1. A line drawing of a city.

To store a line segment it is more economical to store only its extremities and to create a program that can draw the line given these two points. Similarly, the arc of a circle may be specified by its center, radius, and starting and stopping angles. The underlying geometric objects form the *alphabet* with which we can describe an image.

How can we store a more complicated image, for instance, a photograph of a landscape or a forest? It may seem that the previous method cannot work, because our alphabet of geometric objects is too poor. We will discover that we can use the same technique, but with a larger and more advanced alphabet:

- we approximate our image with a finite number of fractal images. For example, consider the fern leaf in Figure 11.2;
- to store the image we create a program that draws the image using the underlying fractals. The fern leaf of Figure 11.2 can be drawn by a program of fewer than 15 lines! (A *Mathematica* program for drawing the fern can be found at the end of Section 11.3.)

In this process the resulting image is the "attractor" of an operator W (defined below) that maps a subset of the plane to a subset of the plane. Beginning from any initial subset B_0 we recursively construct the sequence $B_1 = W(B_0)$, $B_2 = W(B_1)$, $\ldots$, $B_{n+1} = W(B_n)$, $\ldots$. For sufficiently large n (in fact, $n = 10$ suffices if B_0 was carefully chosen), B_n will start to look like the fern leaf.

The technique may sound a little naive: can we really program a computer to approximate any photo using fractals? Indeed, some adaptation of the initial idea will be needed, but we will keep the fundamental idea that the reconstructed image is the attractor of some operator. Since constructing an arbitrary photo is quite advanced, we leave the discussion until the end of the chapter (Section 11.7). To start, we focus on constructing programs that can draw fractals.

11.2 Affine Transformations in the Plane

We start by explaining why we need affine transformations. Consider the fern leaf in Figure 11.2. It is the union of (see Figure 11.2)

- the stalk,
- and three smaller fern leaves: the bottom left branch, the bottom right branch, and the leaf minus the two lowest branches.

Each of these four pieces is the image of the entire fern leaf under an affine transformation. Knowing the four associated transformations allows us to reconstruct the entire image:

- the transformation T_1, which maps the entire leaf to the leaf minus the two lowest branches,
- the transformation T_2, which maps the entire leaf to the bottom left branch (marked L in Figure 11.2),
- the transformation T_3, which maps the entire leaf to the bottom right branch (marked R in Figure 11.2), and

Fig. 11.2. A fern leaf.

- the transformation T_4, which maps the entire leaf to the bottom part of the stalk.

Definition 11.1 *An affine transformation $T : \mathbb{R}^2 \to \mathbb{R}^2$ is the composition of a translation with a linear transformation. It can be written as*

$$T(x, y) = (ax + by + e, cx + dy + f). \tag{11.1}$$

This is the composition of the linear transformation

$$S_1(x, y) = (ax + by, cx + dy)$$

and the translation

$$S_2(x, y) = (x + e, y + f).$$

Linear transformations are often represented in matrix notation as

$$S_1 \begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} a & b \\ c & d \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix}.$$

We can also use this notation to represent affine transformations:

$$T\begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} a & b \\ c & d \end{pmatrix}\begin{pmatrix} x \\ y \end{pmatrix} + \begin{pmatrix} e \\ f \end{pmatrix}.$$

We see that the affine transformation is specified by the six parameters a, b, c, d, e, f. Thus, in order to uniquely determine a given affine transformation we require six linear equations.

Theorem 11.2 *There exists a unique affine transformation that maps three distinct noncollinear points P_1, P_2, and P_3 to three points Q_1, Q_2, and Q_3.*

PROOF: Let (x_i, y_i) be the coordinates of P_i and let (X_i, Y_i) be the coordinates of Q_i. The desired transformation is in the form of (11.1), and we must solve for a, b, c, d, e, f, knowing that $T(x_i, y_i) = (X_i, Y_i)$, $i = 1, 2, 3$. This gives us six linear equations in six unknowns a, b, c, d, e, f:

$$\begin{aligned} ax_1 + by_1 + e &= X_1, \\ cx_1 + dy_1 + f &= Y_1, \\ ax_2 + by_2 + e &= X_2, \\ cx_2 + dy_2 + f &= Y_2, \\ ax_3 + by_3 + e &= X_3, \\ cx_3 + dy_3 + f &= Y_3. \end{aligned}$$

The parameters a, b, e are solutions of the system

$$\begin{aligned} ax_1 + by_1 + e &= X_1, \\ ax_2 + by_2 + e &= X_2, \\ ax_3 + by_3 + e &= X_3, \end{aligned} \tag{11.2}$$

while c, d, f are solutions of the system

$$\begin{aligned} cx_1 + dy_1 + f &= Y_1, \\ cx_2 + dy_2 + f &= Y_2, \\ cx_3 + dy_3 + f &= Y_3. \end{aligned} \tag{11.3}$$

Both of these are systems over the same matrix A, whose determinant is

$$\det A = \begin{vmatrix} x_1 & y_1 & 1 \\ x_2 & y_2 & 1 \\ x_3 & y_3 & 1 \end{vmatrix}.$$

Note that this determinant is nonzero precisely when the points P_1, P_2, and P_3 are distinct and noncollinear. In fact, the three points are collinear if and only if the vectors $\overrightarrow{P_1P_2} = (x_2 - x_1, y_2 - y_1)$ and $\overrightarrow{P_1P_3} = (x_3 - x_1, y_3 - y_1)$ are collinear, which is the case if and only if the following determinant is zero:

$$\begin{vmatrix} x_2 - x_1 & y_2 - y_1 \\ x_3 - x_1 & y_3 - y_1 \end{vmatrix} = (x_2 - x_1)(y_3 - y_1) - (x_3 - x_1)(y_2 - y_1).$$

The determinant of a matrix does not change when we add to a row a multiple of another. Subtracting the first row from the second and the third yields

$$\begin{aligned} \det A &= \begin{vmatrix} x_1 & y_1 & 1 \\ x_2 - x_1 & y_2 - y_1 & 0 \\ x_3 - x_1 & y_3 - y_1 & 0 \end{vmatrix} \\ &= (x_2 - x_1)(y_3 - y_1) - (x_3 - x_1)(y_2 - y_1). \end{aligned}$$

We see that $\det A = 0$ precisely when the three points are aligned. On the other hand, if $\det A \neq 0$, then each of the systems (11.2) and (11.3) has a unique solution. □

Remark: We must use the technique of Theorem 11.2 to find the four transformations describing the fern leaf. For that we need to specify coordinate axes and measure the coordinates of the points P_i and Q_i. However, in many examples we can guess the affine transformations without having to measure the coordinates of the points P_i and Q_i and solving the associated systems. In these cases we use compositions of simple affine transformations.

Some simple affine transformations.

- Homothety with ratio r: $T(x, y) = (rx, ry)$.
- Reflection about the x axis: $T(x, y) = (x, -y)$.
- Reflection about the y axis: $T(x, y) = (-x, y)$.
- Reflection through the origin: $T(x, y) = (-x, -y)$.
- Rotation through angle θ: $T(x, y) = (x \cos\theta - y \sin\theta, x \sin\theta + y \cos\theta)$. To find this formula we use the fact that a rotation is a linear transformation. The columns of its matrix are the coordinates of the images of the base vectors $e_1 = (1, 0)$ and $e_2 = (0, 1)$ (Figure 11.3). The transformation matrix is therefore

$$\begin{pmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{pmatrix}.$$

- Projection onto the x axis: $T(x, y) = (x, 0)$.
- Projection onto the y axis: $T(x, y) = (0, y)$.
- Translation by a vector (e, f): $T(x, y) = (x + e, y + f)$.

11.3 Iterated Function Systems

Fractals that can be constructed using the technique described above will be *attractors* of *iterated function systems*. We define these terms more clearly.

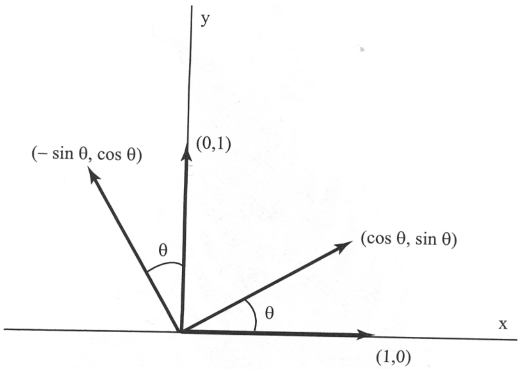


Fig. 11.3. *The images of base vectors under a rotation of angle θ.*

Definition 11.3 *1. An affine transformation is an* affine contraction *if the image of any segment is a shorter line segment.*
2. An iterated function system *is a set of affine contractions $\{T_1, \dots, T_m\}$.*
3. The attractor *of an iterated function system $\{T_1, \dots, T_m\}$ will be the unique geometric object A such that*

$$A = T_1(A) \cup \cdots \cup T_m(A).$$

Example 11.4 A fern leaf. *We consider the fern leaf from Figure 11.2. It is easy to see that each of the branches of the leaf resembles the entire leaf itself. Thus, the leaf is the union of the stalk and infinitely many smaller copies of the leaf. We want to avoid working with an infinite number of sets of transformations, so a little care is required. Call A the subset of the plane consisting of all points of the fern leaf. We introduce a coordinate system. Let T_1 be the transformation mapping P_i to Q_i, as labeled in Figure 11.4. The image $T_1(A)$ is a subset of A. Now consider $A \setminus T_1(A)$. It consists of the bottom portion of the stalk and the bottommost branches on either side, as outlined in Figure 11.2. We can choose points Q'_1, Q'_2, and Q'_3 to construct a transformation T_2 that maps the entire leaf to the bottommost left branch. (Exercise!) Similarly, we can choose points Q''_1, Q''_2, and Q''_3 describing a transformation T_3 that maps to the bottommost right branch. Thus $A \setminus (T_1(A) \cup T_2(A) \cup T_3(A))$ is simply the bottommost portion of the stalk. We wish to find another transformation T_4 that maps the entire leaf to this portion of the stalk. Such a transformation is simply a projection onto the y axis composed with a contraction (homothety with ratio $r < 1$) and a translation.*

We have constructed four affine transformations such that

$$A = T_1(A) \cup T_2(A) \cup T_3(A) \cup T_4(A). \tag{11.4}$$

Fig. 11.4. The points P_i and Q_i describing the transformation T_1.

We claim and will prove later that no other set than the fern satisfies (11.4). *The fern leaf will be the attractor of the iterated function system* $\{T_1, T_2, T_3, T_4\}$.

This example is relatively complicated. Thus, we present another easier example to help develop some intuition.

Example 11.5 The Sierpiński triangle. *To simplify the calculations we will consider a Sierpiński triangle with a base and height of* 1 *(see Figure 11.5).*

Here the triangle A is the union of three smaller copies of itself $A = T_1(A) \cup T_2(A) \cup T_3(A)$. In this case we can easily write the explicit equations of the affine contractions. In fact, if we suppose that the origin is situated at the bottom left corner of the triangle, then T_1 is the homothety with ratio $1/2$:

$$T_1(x, y) = (x/2, y/2),$$

and T_2 and T_3 are simply compositions of T_1 with translations. Since the base and height of the triangle are both 1, then T_2 is T_1 composed with a translation by $(1/2, 0)$, while T_3 is T_1 composed with a translation by $(1/4, 1/2)$:

$$\begin{aligned} T_2(x, y) &= (x/2 + 1/2, y/2), \\ T_3(x, y) &= (x/2 + 1/4, y/2 + 1/2). \end{aligned}$$

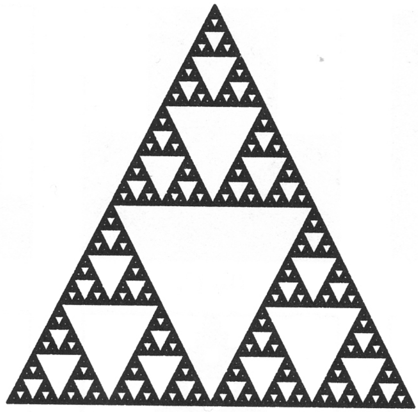


Fig. 11.5. The Sierpiński triangle.

The triangle lies within the square $C_0 = [0,1] \times [0,1]$. *We are interested in the sets*

$$\begin{aligned} C_1 &= T_1(C_0) \cup T_2(C_0) \cup T_3(C_0), \\ C_2 &= T_1(C_1) \cup T_2(C_1) \cup T_3(C_1), \\ &\vdots \\ C_n &= T_1(C_{n-1}) \cup T_2(C_{n-1}) \cup T_3(C_{n-1}), \\ &\vdots \end{aligned}$$

the first few of which are shown in Figure 11.6. Observe that for sufficiently large n *(even at* $n = 10$*), the set* C_n *already begins to resemble* A. *The set*

$$C_n = T_1(C_{n-1}) \cup T_2(C_{n-1}) \cup T_3(C_{n-1})$$

is called the n*th iteration of the initial set* C_0 *under the operator*

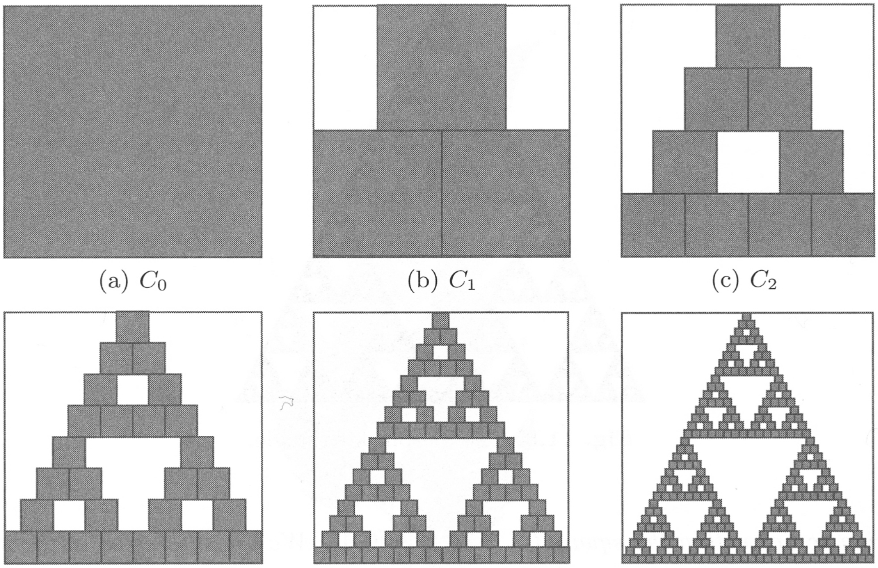
$$C \mapsto W(C) = T_1(C) \cup T_2(C) \cup T_3(C),$$

which maps a subset C *to another subset* $W(C)$.

It is for this reason we say that A *is an* attractor. *The remarkable thing is, had we started with any subset of the plane other than* C_0, *the limit of the process would still be the Sierpiński triangle (see Figure 11.7).*

The general principle. The Sierpiński triangle example allowed us to see the general process at work. Given an iterated function system $\{T_1, \ldots, T_m\}$ of affine contractions, we construct an *operator* W that acts on subsets of the plane. A subset C is mapped to the subset $W(C)$ as follows:

$$W(C) = T_1(C) \cup T_2(C) \cup \cdots \cup T_m(C). \tag{11.5}$$



(a) C_0 (b) C_1 (c) C_2

(d) C_3 (e) C_4 (f) C_5

Fig. 11.6. C_0 and the first five iterations C_1–C_5.

The fractal A that we wish to construct is a subset of the plane satisfying $W(A) = A$. We say that A is a *fixed point* of the operator W.

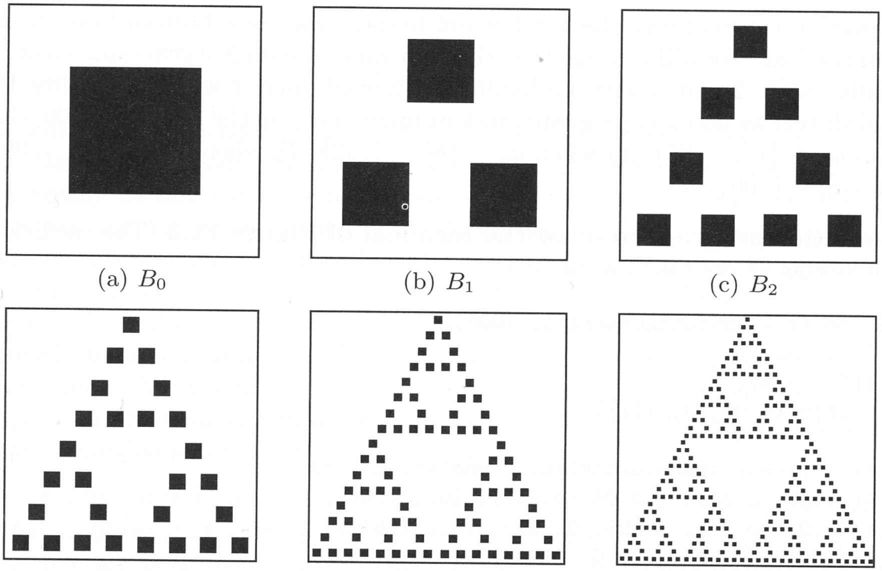
In the next section we will see that for all iterated function systems there exists a unique subset A of the plane that is a fixed point of the operator W. Moreover, we will show that for all nonempty subsets $C_0 \subset \mathbb{R}^2$, the subset A is the *limit* of the sequence $\{C_n\}$ defined by the recurrence

$$C_{n+1} = W(C_n).$$

The subset A is called the attractor of the iterated function system. Thus, if we know of a set B satisfying $B = W(B)$, then we know that B will be the limit of the sequence $\{C_n\}$.

In our Sierpiński triangle example we used the unit square $[0, 1] \times [0, 1]$ as our initial set C_0, and we constructed the sequence $\{C_n\}_{n\geq 0}$ using the recurrence $C_{n+1} = W(C_n)$. The experimental results of Figure 11.6 convinced us that the sequence $\{C_n\}_{n\geq 0}$ "converges" to the set A, the Sierpiński triangle. We could have performed this experiment with any initial set B_0, for example $B_0 = [1/4, 3/4] \times [1/4, 3/4]$. We would have obtained that the sequence $\{B_n\}_{n\geq}0$, where $B_{n+1} = W(B_n)$, again converges to A (Figure 11.7).

We can convince ourselves that we could have taken an initial set B_0 consisting only of a single point of the square C_0. In this case, the set B_n consists of 3^n points. If



(a) B_0 (b) B_1 (c) B_2 (d) B_3 (e) B_4 (f) B_5

Fig. 11.7. B_0 and the first five iterations B_1–B_5.

for each point in B_n we darken the corresponding pixel in a digitized image, then for sufficiently large n the image would resemble the Sierpiński triangle A.

In fact, traditional programs for drawing fractals function in a slightly different way, since it is simpler to draw a single point at each step than subsets of the plane consisting of 3^n points. We start by choosing a point P_0 in the rectangle R. At each step we randomly choose one of the transformations T_i and we calculate $P_n = T_{i_n}(P_{n-1})$, where T_{i_n} is the randomly chosen transformation at step n. If the point P_0 is already in the set A, then drawing the entire set of points from the sequence $\{P_n\}_{n\geq 0}$ will quickly begin to resemble A. If we are unsure whether P_0 is in A, then we discard the first M generated points $P_0, \ldots, P_{M-1}$, and draw the points $\{P_n\}_{n\geq M}$. The following section will show that there always exists a value for M that will ensure that we achieve a good approximation to A. In practice, M is often taken as small as 10, since convergence to the attractor usually occurs quite rapidly.

When drawing the Sierpiński triangle of Figure 11.5, at each step we randomly chose one of the transformations $\{T_1, T_2, T_3\}$. Thus, at step n we randomly chose a number $i_n \in \{1, 2, 3\}$ and applied the transformation T_{i_n}. Each time we generated 1 we applied T_1. If we generated 2 we applied T_2, and if we generated 3 we applied T_3. For the fern leaf this approach is not very efficient: we would spend too much time drawing points on the stalk and the bottom leaves and not enough time in the rest of the leaf. Let T_1 (respectively T_2, T_3, T_4) be the affine contraction that maps the leaf onto the upper portion (respectively the left bottom branch, the right bottom branch, and the stalk) of the leaf. We will arrange it so that our random-number generator yields 1 with probability 85%, 2 and 3 with probabilities 7% each, and 4 with probability 1%. To accomplish this we actually generate random-numbers $\bar{a}_n$ in the range 1 to 100, choosing T_1 when $\bar{a}_n \in \{1, \ldots, 85\}$, T_2 when $\bar{a}_n \in \{86, \ldots, 92\}$, T_3 when $\bar{a}_n \in \{93, \ldots, 99\}$, and T_4 when $\bar{a}_n \in \{100\}$.

Mathematica program to draw the fern leaf of Figure 11.2 (The coefficients for the transforms T_i are taken from [1].)

```
chooseT := (r = RandomInteger[{1, 100}];
    If[r <= 85, 1,
      If[r <= 92, 2,
        If[r <= 99, 3, 4]]])

t = { (* { linear transformation, translation } *)
      {{{0.85, 0.04}, {-0.04, 0.85}}, {0., 1.6}},
      {{{0.2, -0.26}, {0.23, 0.22}}, {0., 1.6}},
      {{{-0.15, 0.28}, {0.26, 0.24}}, {0., 0.44}},
      {{{0., 0.}, {0., 0.16}}, {0., 0.}}
      };

transfoAff[t_, pt_] := t[[1]].pt + t[[2]]

nIteration = 20000; A = {{0., 0.}};
Do[AppendTo[A, transfoAff[t[[chooseT]], Last[A]]], {nIteration}]

ListPlot[A, AspectRatio -> Automatic, Axes -> False]
```

11.4 Iterated Contractions and Fixed Points

A full reading of this section requires some familiarity with analysis, but the basic concepts can be understood without it.

We noted previously that for all iterated function systems $\{T_1, \ldots, T_m\}$ there exists a unique subset A of the plane that is a fixed point of the operator W defined by

$$W(B) = T_1(B) \cup \cdots \cup T_m(B). \tag{11.6}$$

This set, satisfying $W(A) = A$, is called the attractor of the iterated function system. We will now justify this claim.

The following theorem from real analysis provides the key.

Theorem 11.6 *Let $f : \mathbb{R} \to \mathbb{R}$ be a contraction. In other words, there exists some $0 < r < 1$ such that for all $x, x' \in \mathbb{R}$ we have that*

$$|f(x) - f(x')| \leq r|x - x'|.$$

Then f has a unique fixed point $a \in \mathbb{R}$ such that $f(a) = a$.

We will prove this theorem in order to understand exactly how it works. While working through the proof, note that we can replace $\mathbb{R}$ by any closed interval $[\alpha, \beta]$ and more generally by any complete metric space (an intuitive definition of this follows). However, we are unable to replace $\mathbb{R}$ by $\mathbb{Q}$, nor by any open interval (α, β). When generalizing this theorem we will replace the notion of a point in $\mathbb{R}$ with that of a closed and bounded subset of $\mathbb{R}^2$, and the function f by the operator W defined in (11.6). We will require the notion of a *distance* between two subsets (the equivalent of $|x - x'|$ in the above formulation) and we will need to show that W is a contraction with respect to this distance. We would like to be able to use the same argument as will be used in the proof of Theorem 11.6 in order to prove the existence of a unique attractor A, a closed and bounded subset of $\mathbb{R}^2$ that is the fixed point of W.

PROOF OF THEOREM 11.6: We start by showing that if f has a fixed point, then it must be unique. Suppose that $a_1 \neq a_2$ are two fixed points of f. Then $f(a_2) - f(a_1) = a_2 - a_1$ because they are both fixed points. However, since f is a contraction, we have that $|f(a_2) - f(a_1)| \leq r|a_2 - a_1|$, where $0 < r < 1$, a contradiction.

We must now prove the existence of a. To obtain a we will start with a point $x_0 \in \mathbb{R}$ and construct the sequence of its iterates $x_1 = f(x_0), x_2 = f(x_1), \ldots, x_{n+1} = f(x_n), \ldots$. If $x_1 = x_0$, then x_0 is a fixed point and we are done. Consider the case $x_1 \neq x_0$. Then

$$|x_{n+1} - x_n| = |f(x_n) - f(x_{n-1})| \leq r|x_n - x_{n-1}|.$$

By iterating we obtain

$$|x_{n+1} - x_n| \leq r^n |x_1 - x_0|.$$

We wish to show that the sequence $\{x_n\}$ converges to a point $a \in \mathbb{R}$ and that the limit a is a fixed point of f. A very powerful tool exists that permits us to show that a sequence of real numbers converges without having to guess a candidate for the limit: it suffices to show that it is a Cauchy sequence. (Recall that a sequence $\{x_n\}$ is a Cauchy sequence if $\forall \epsilon > 0\ \exists N \in \mathbb{N}$ such that if $n, m > N$ then $|x_n - x_m| < \epsilon$.) Suppose that $n > m$. Then

$$\begin{aligned}
|x_n - x_m| &= |(x_n - x_{n-1}) + (x_{n-1} - x_{n-2}) + \cdots + (x_{m+1} - x_m)| \\
&\leq |x_n - x_{n-1}| + |x_{n-1} - x_{n-2}| + \cdots + |x_{m+1} - x_m| \\
&\leq (r^{n-1} + r^{n-2} + \cdots + r^m)|x_1 - x_0| \\
&\leq r^m(r^{n-m-1} + \cdots + 1)|x_1 - x_0| \\
&\leq \tfrac{r^m}{1-r}|x_1 - x_0|.
\end{aligned}$$

For $|x_n - x_m|$ to be smaller than ϵ it suffices to take m sufficiently large, such that

$$\frac{r^m|x_1 - x_0|}{1 - r} < \epsilon,$$

or in other words, $r^m < \frac{\epsilon(1-r)}{|x_1-x_0|}$. Since $0 < r < 1$, we then take N large enough such that $\frac{r^N|x_1-x_0|}{1-r} < \epsilon$. Since $r^m < r^N$ for $N > m$ we have shown that the sequence $\{x_n\}$ is a Cauchy sequence.

Since every Cauchy sequence of real numbers converges to a real number, this yields that the sequence $\{x_n\}$ converges to some number $a \in \mathbb{R}$. We must now show that a is a fixed point of f. To do this we need to show that f is continuous. In fact, f is actually uniformly continuous on $\mathbb{R}$. Consider $\epsilon > 0$ and take $\delta = \epsilon$. Then if $|x - x'| < \delta$ we have that

$$|f(x) - f(x')| \leq r|x - x'| < r\delta = r\epsilon < \epsilon.$$

Since f is continuous, the image of the convergent sequence $\{x_n\}$ with limit a is itself a convergent sequence with limit $f(a)$. Thus

$$f(a) = \lim_{n\to\infty} f(x_n) = \lim_{n\to\infty} x_{n+1} = \lim_{n\to\infty} x_n = a.$$

□

We can generalize the statement of the previous theorem while maintaining the same proof. We can replace $\mathbb{R}$ by a general space K sharing certain properties with $\mathbb{R}$. In fact, we require only that K be a *complete metric space.* In order to keep the letters x and y for the Cartesian coordinates of a point we will denote points of K by the letters $v, w, \ldots$. Before we can elaborate on such spaces we must precisely define the notion of a *distance* $d(v, w)$ between two elements v, w of a space K. We will construct our definition of a distance so that it mirrors the properties of $|x - x'|$ in $\mathbb{R}$.

Definition 11.7 *1. A distance function $d(\cdot,\cdot)$ on a set K is a function $d : K \times K \to \mathbb{R}^+ \cup \{0\}$ that satisfies:*
(i) $d(v, w) \geq 0$;
(ii) $d(v, w) = d(w, v)$;
(iii) $d(v, w) = 0$ if and only if $v = w$;
(iv) Triangle inequality: for all v, w, z,

$$d(v, w) \leq d(v, z) + d(z, w).$$

2. *A set K equipped with a distance function d is called a* metric space.
3. *A sequence $\{v_n\}$ of elements in K is a Cauchy sequence if $\forall\epsilon > 0$, $\exists N \in \mathbb{N}$ such that for all $m, n > N$, we have that $d(v_n, v_m) < \epsilon$.*
4. *A sequence $\{v_n\}$ of elements of K converges to an element $w \in K$ if $\forall\epsilon > 0$, $\exists N \in \mathbb{N}$ such that for all $n > N$, we have that $d(v_n, w) < \epsilon$. The element w is called the limit of the sequence $\{v_n\}$.*
5. *A metric space K is* complete *if any Cauchy sequence of elements from K converges to a limit also in K.*

Example 11.8 *1. $\mathbb{R}^n$ with the Euclidean distance is a complete metric space.*

2. *Let K be the set of all closed and bounded subsets of $\mathbb{R}^2$: we call them* compact *subsets of $\mathbb{R}^2$. The distance we will use over this set of subsets is the Hausdorff distance, which will be defined and discussed in Section 11.5. Equipped with this distance, K will be a complete metric space (the proof of this fact can be found in [1]).*
3. *When moving from theory to practice in Section 11.7, we will consider a black and white photo on a rectangle R as a function $f : R \to S$, where S denotes the set of gray tones. We can then define a distance between two such functions f and f' through the use of the following definitions:*

$$d_1(f, f') = \max_{(x,y)\in R} |f(x, y) - f'(x, y)|$$

and

$$d_2(f, f') = \left(\iint_R (f(x, y) - f'(x, y))^2 \, dx\, dy\right)^{1/2}. \tag{11.7}$$

Equipped with these distances, the set of functions $f : R \to S$ is a complete metric space. We can replace the set $R = [a, b] \times [c, d]$ by a discrete set of pixels over the rectangle R by adapting slightly the above definitions. For example, the double integral in the distance function will be replaced by a discrete sum over the individual pixels. If x and y take the values $\{0, \ldots, h-1\}$ and $\{0, \ldots, v-1\}$) respectively, then the distance (11.7) *becomes*

$$d_3(f, f') = \left(\sum_{x=0}^{h-1}\sum_{y=0}^{v-1}(f(x, y) - f'(x, y))^2\right)^{1/2}. \tag{11.8}$$

We require that the operator W defined in (11.5) be a contraction with respect to the distance function over the space K. This leads us to the famous Banach fixed-point theorem: since we will apply it with the elements of K being compact subsets of $\mathbb{R}^2$, we will use capital letters for the elements of K.

Theorem 11.9 (Banach fixed-point Theorem) *Let K be a complete metric space and $W : K \to K$ a contraction. In other words, let W be a function such that for all $B_1, B_2 \in K$,*

$$d(W(B_1), W(B_2)) \le r\, d(B_1, B_2) \tag{11.9}$$

with $0 < r < 1$. Then there exists a unique fixed point $A \in K$ of W such that $W(A) = A$.

We will not give a proof of the Banach fixed-point theorem, since it is exactly the same as that of Theorem 11.6. We only need to replace $|x - x'|$ by $d(B, B')$.

The Banach fixed-point theorem is one of the most important theorems in mathematics. It has applications in many diverse areas.

Example 11.10 *We discuss a few applications of the Banach fixed-point theorem:*

1. *A first classical application of this theorem allows us to prove the existence and uniqueness of solutions to ordinary differential equations satisfying a Lipschitz condition. In this example the elements of K are functions. The fixed point is the unique function that is a solution to the differential equation. We will not go further into this example. However, we wish to point out that simple ideas often have important applications in seemingly unrelated fields.*
2. *The second application is of immediate interest. Let K be the set of all closed and bounded subsets of the plane, together with the Hausdorff distance. Equipped with this distance, K will be a complete metric space. Consider a set of affine contractions $T_1, \ldots, T_m$ forming an iterated function system. We define the operator of* (11.6), *and we will show that it is a contraction, satisfying* (11.9) *for some $0 < r < 1$. Theorem 11.9 immediately proves both the existence and uniqueness of the attractor A of such an iterated function system.*

Remark: The Banach theorem states that the fixed point A of a contraction W must be *unique*. Thus, if we are already aware of a set A satisfying this property (for example, the fern leaf), then we are sure that it is indeed the fixed point of the iterated function system we have constructed.

11.5 The Hausdorff Distance

The definition of this distance function is somewhat difficult. Thus, we will start by discussing the intuitive foundations on which it was built. The proof of the Banach fixed-point theorem uses the distance function only as a tool for discussing convergence and for discussing the closeness of two elements of K. When we talk of the convergence of a sequence of sets B_n in K to some set A, intuitively we wish to show that for sufficiently large n, the sets B_n strongly resemble A.

Thus, we wish to quantify the notion of closeness between two sets B_1 and B_2, such that we can say precisely when two sets are within some distance ϵ of each other. One way of doing this is to consider "inflating" the set B_1 by an amount ϵ. That is, we consider the set of all points within a distance ϵ of some point in B_1. If the distance between B_1 and B_2 is less than ϵ, then B_2 should be entirely contained in the inflated version of B_1. The ϵ-inflated set B_1 is given by

$$B_1(\epsilon) = \{v \in \mathbb{R}^2 | \exists w \in B_1 \text{ such that } d(v,w) < \epsilon\},$$

where $d(v,w)$ is the usual Euclidean distance between v and w, both points of $\mathbb{R}^2$. We require that $B_2 \subset B_1(\epsilon)$. However, this is not sufficient. The set B_2 could have a very different form and be much smaller than B_1. Thus, we also consider inflating B_2,

$$B_2(\epsilon) = \{v \in \mathbb{R}^2 | \exists w \in B_2 \text{ such that } d(v,w) < \epsilon\},$$