Fractals

Ming-Hwa Wang, Ph.D.
COEN 148/290 Computer Graphics
COEN 396 Interactive Multimedia and Game Programming
Department of Computer Engineering
Santa Clara University

Introduction

A self-similar structure that occurs at different levels of magnification can be modeled by a branch of mathematics called fractal geometry

- a fractal is a dynamical system, and it can be generated by a computer recursively or iteratively with a repeating pattern
- to model plants, weather, fluid flow, geologic activity, planetary orbits, human body rhythms, socioeconomic patterns, and music

Iterated Function System (IFS)

- an IFS includes a number of kinds of operations:
  - contraction mapping operation: a function from a closed, bounded region to a closed bounded region with small area; if it is applied often enough, the original region will be mapped into a region of arbitrarily small area, the attractor; usually generate a large number of random points and apply the contraction mapping repeatedly to each
    - Sierpinski gasket: 4 functions map the tetrahedron into 4 tetrahedra with equal probability; the linear contraction function $f_i(p) = (p + p_i)/2$, for $\{p_i\}$ the 4 vertices of the tetrahedron
    - fern leaf:
      - $f_0(x, y) = (0, 0.16y)$ $p_0 = 0.1$
      - $f_1(x, y) = (0.85x+0.04y, -0.04x+0.85y+1.6)$ $p_1 = 0.85$
      - $f_2(x, y) = (0.2x-0.26y, 0.23x+0.22y+1.6)$ $p_2 = 0.7$
      - $f_3(x, y) = (-0.15x+0.28y, 0.26x+0.24y+0.44)$ $p_3 = 0.7$
  - generating functions: defines a geometric structure recursively, applying a recursive generating function to each component of the geometry; the *everywhere continuous* property – the final function is the limit of a converging sequence of uniformly continuous functions; the *nowhere differentiable* property – no matter how close we choose a pair of values, there will be a line segment between them for some iteration of this process
    - blancmange function: starting with a single line segment and replacing each segment by a pair of lines where the center of the pair is offset by ¼ the length of the line segment
    - dragon curve: replacing a simple line segment by 2 segments but putting the 2 segments offset to alternating sides of the line; a fascinating property is the way dragon curves fill the space around a given point
    - Koch curve (see below)

Koch Curves

- to construct a Koch curve
  - begin with a straight line and call it K_0
  - divide each segment of K_n into 3 equal parts
  - replace the middle part by the two sides of an equilateral triangle of the same length as the part being removed
- characteristics of a Koch curve
  - each segment is increased by a factor of 4/3, therefore, K_{n+1} is 4/3 as long as K_n, and K_i has the total length of $(4/3)^i$
  - when n is getting large, the curve still appears to have the same shape and roughness (with finer resolution)
  - when n becomes infinite, the curve has an infinite length, while occupying a finite region in the plane
- joining 3 Koch curves together, we get a Koch snowflake, the length of a Koch snowflake is $3 * (4/3)^i$ for the i^{th} generation
- a logo programming language, turtle graphics, using the concept of a turtle crawling over the drawing space with a pen attached to its underside, the drawing is always relative to the current position and direction of the turtle
- fractal landscape (midpoint subdivision): begins with a triangular patch, each of the edge of the patch is divided in half, and the midpoints are randomly displaced to add some irregularity to the surface; after one subdivision, the surface is now a collection of 4 triangular patches; this process is repeated

String Grammars

- L-systems: a parallel-grammar-based systems for drawing fractal curves, each character in the string serves as a command to perform an atomic operation
  - *F*: move forward the distance *D* while drawing in the current direction
  - *f*: move forward the distance *D* without drawing a line
  - +: turn right through the angle α
  - -: turn left through the angle α
  - [: store the current state of the turtle
  -]: restore the turtle's previously stored state
- the 1st generation of Koch curve K_1 is a string production rule $F \rightarrow F\text{-}F\text{++}F\text{-}F$ and angle 60°; the initial string is the axiom, which is recursively applying the production rule to produce strings of increasing length; the right-side of the rule is the F-string; and the grammar of the curve is (*F*, *F-F++F-F*, 60)
- a general grammar template with 6 parameters as (axiom, F-string, f-string, X-string, Y-string, angle), where X-string is used to replace every occurrence of X when producing the next generation string; Y-string is used to replace every occurrence of Y
  - Dragon curve: (*X, F, nil, X+YF+, -FX-Y,* 90)
  - Hilbert curve : (*X, F, nil, -YF+XFX+FY-, +XF-YFY-FX+,* 90)
  - Sierpinski arrowhead : (*YF, F, nil, YF+XF+Y, XF-YF-X,* 60)
  - islands : (*F+F+F+F, F+f-FF+F+FF+Ff+FF-f+FF-F-FF-Ff-FFF, ffffff, nil, nil,* 90)

- tree1 : (*F, F[+F]F[-F]F, nil, nil, nil,* 25.7)
- tree2 : (*X, FF, nil, F[+X]F[-X]+X, nil,* 20)
- tree3: (*F, FF-[-F+F+F]+[+F-F-F], nil, nil, nil,* 22.5)

- the save and store state can generate branches, and if the line thickness of each branch is set in proportion to its distance from the root and also a small fraction of randomness is applied to the turning angle, more realistic looking trees would be produced

Graftals

- graftals are another parallel grammar (L-systems) that are interpreted to give a graphic representation
- for each 0 or 1 symbol, a line of particular length is drawn; when a "[" symbol is reached, the current location and direction are noted and drawing continues at an angle of 45°; when a "]" symbol is reached, drawing moves back to the most recently noted location and direction; the direction of angle alternate between going to the left and right

Mandelbrot Sets

- the Mandelbrot set is infinitely complex; although it is self similar at different scales, the small-scale details are not identical to the whole; yet the process of generating it is based on an extremely simple equation involving complex numbers
- the Mandelbrot set is a set $\boldsymbol{M}$ of complex numbers defined as $\boldsymbol{M} = \{c \in \mathrm{C} \mid \lim_{n\to\infty} z_n \neq \infty\}$ and for a constant c, the sequence z_0, z_1, .. is defined as: $z_0 = 0$, $z_{n+1} = z_n^2 + c$; i.e., if the sequence does not approach infinite, then c belongs to the set; given a value c, the system generates a sequence of values called the *orbit* of the start value 0; as soon as an element of the sequence is at a distance greater than 2 from the origin, it is certain that the sequence tends to infinity; the computation of the square root operation can be saved by just checking whether $|z|^2 > 4$; the real part $x_{n+1} = x_n^2 - y_n^2 + Real(c)$, and the imaginary part $y_{n+1} = 2x_n y_n + Imaginary(c)$

Julia Sets

- a Julia set uses the same iteration $z_{n+1} = z_n^2 + c$, where c is the *index* into the Julia set, but with a starting value z_0 derived from the coordinates (x_{pix}, y_{pix}) of the pixel displayed on the screen; choose points near the boundaries of the Mandelbrot set as starting value z_0 gives interesting Julia sets
- a Julia set is either connected or disconnected; for values of z_0 chosen from within the Mandelbrot set, we obtains connected Julia set; conversely, those values of z_0 outside the Mandelbrot set give disconnected Julia sets or *dusts*

This document was created with Win2PDF available at http://www.win2pdf.com.
The unregistered version of Win2PDF is for evaluation or non-commercial use only.
This page will not be added after purchasing Win2PDF.