Computer Graphics 11: 3D Object Representations – Octrees & Fractals

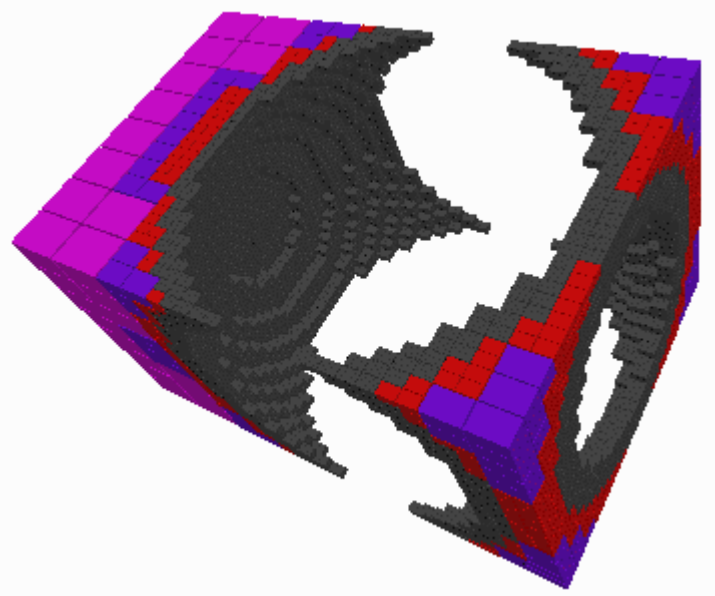
Contents

In today's lecture we would like to continue on from the last day and look at some more modelling techniques

- Octrees
- Fractals

Octrees

Octrees are hierarchical tree structures used to represent solid objects



Octrees are particularly useful in applications that require cross sectional views – for example medical applications

Octrees are typically used when the interior of objects is important

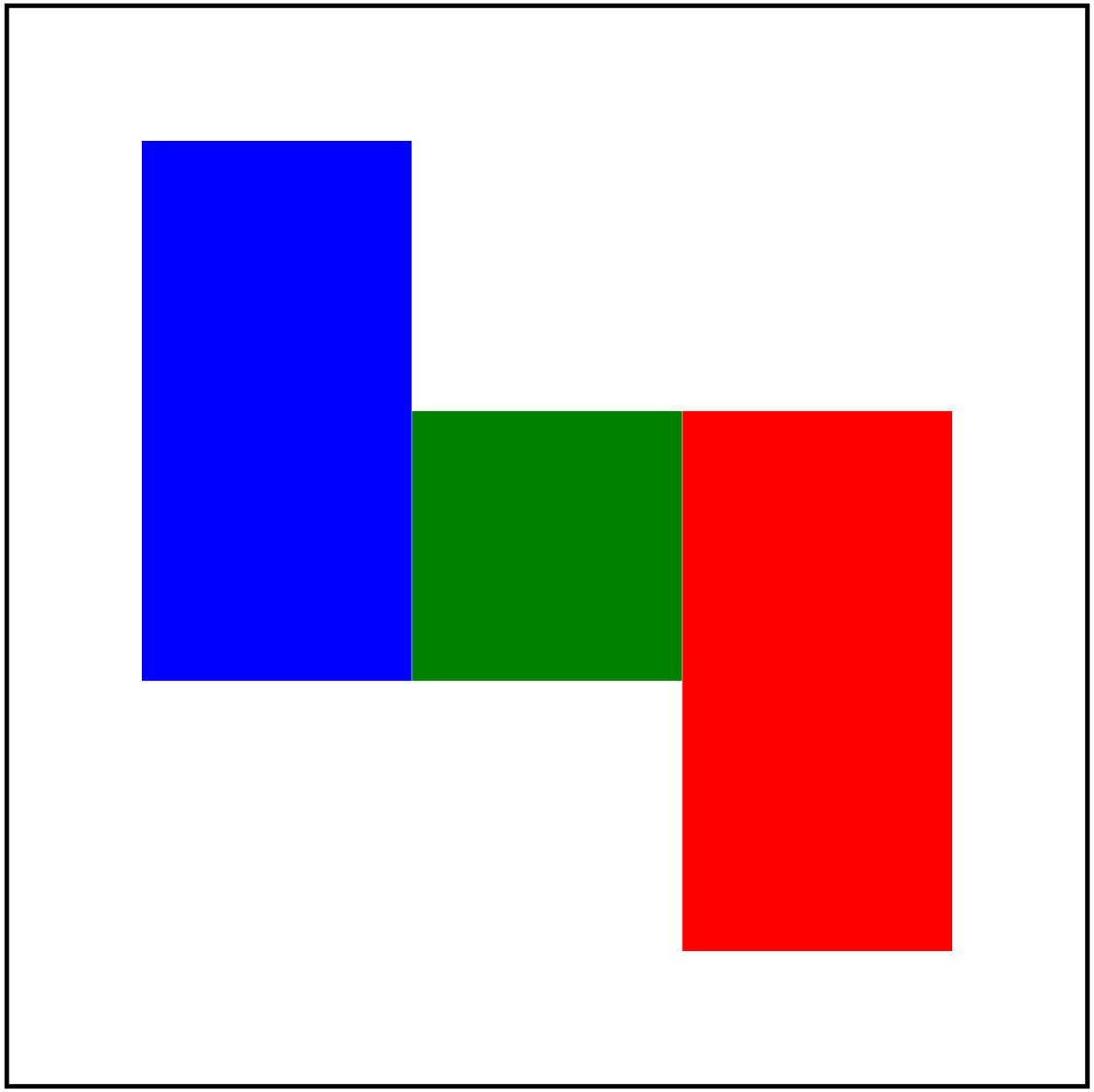
Octrees & Quadtrees

Octrees are based on a two-dimensional representation scheme called **quadtree** encoding

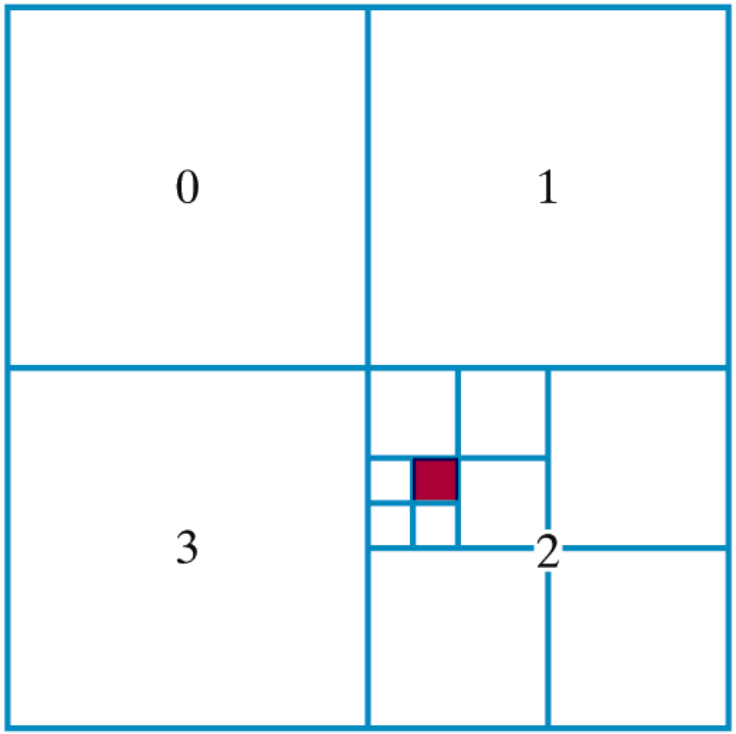
Quadtree encoding divides a square region of space into four equal areas until *homogeneous regions* are found

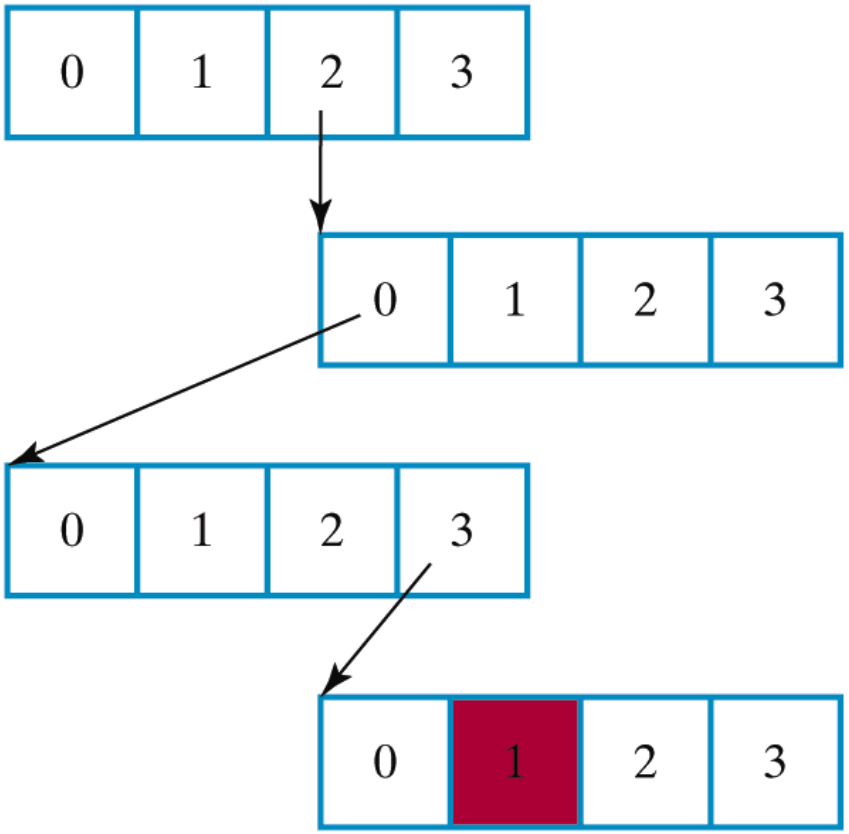
These regions can then be arranged in a tree

Quadtree Example 1



Quadtree Example 2





Images taken from Hearn & Baker, "Computer Graphics with OpenGL" (2004)

Octrees

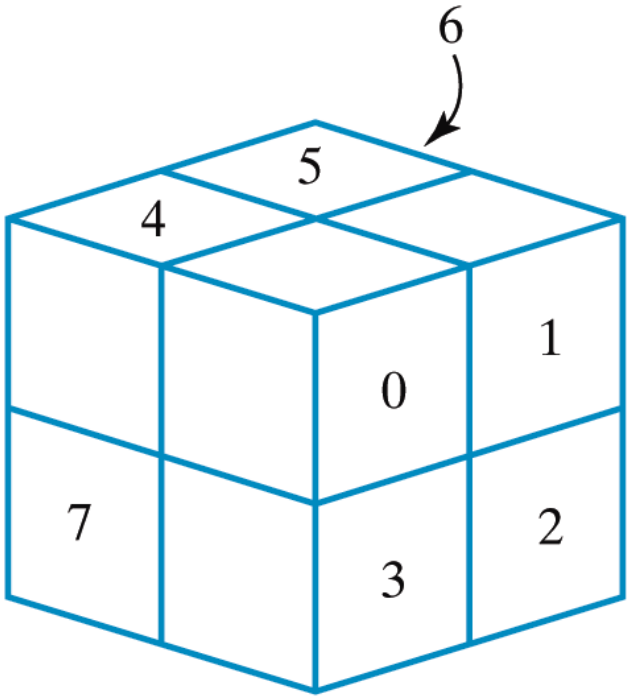
Quadtree encodings provide considerable savings in storage when large colour areas exist in a region of space

An octree takes the same approach as quadtrees, but divides a cube region of 3D space into octants

Each region within an octree is referred to as a **volume element** or **voxel**

Division is continued until homogeneous regions are discovered

Octrees (cont…)



Region of a
Three-Dimensional
Space

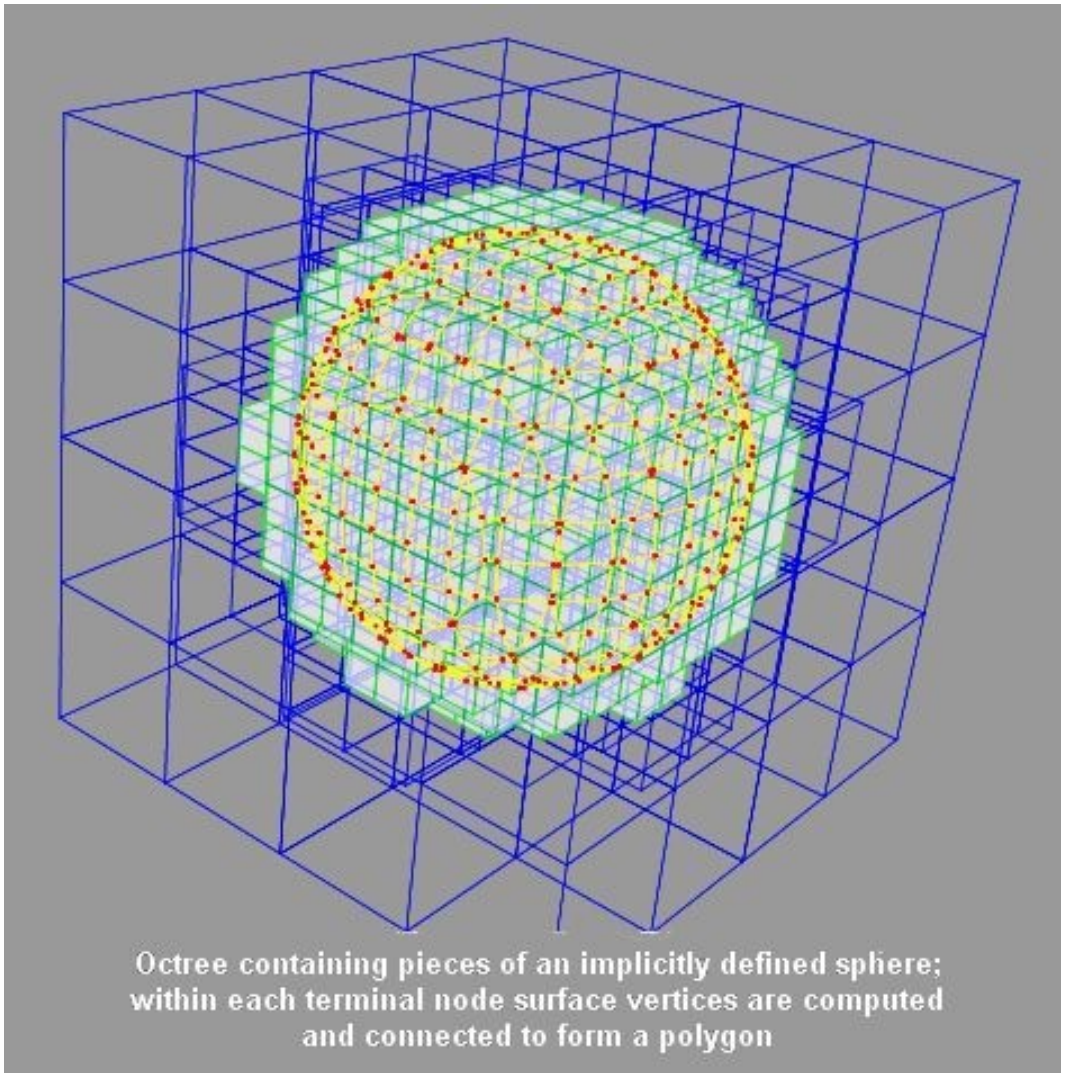
Data Elements
in the Representative
Octree Node

Octrees (cont…)

In 3 dimensions regions can be considered to be homogeneous in terms of colour, material type, density or any other physical characteristics

Voxels also have the unique possibility of being *empty*

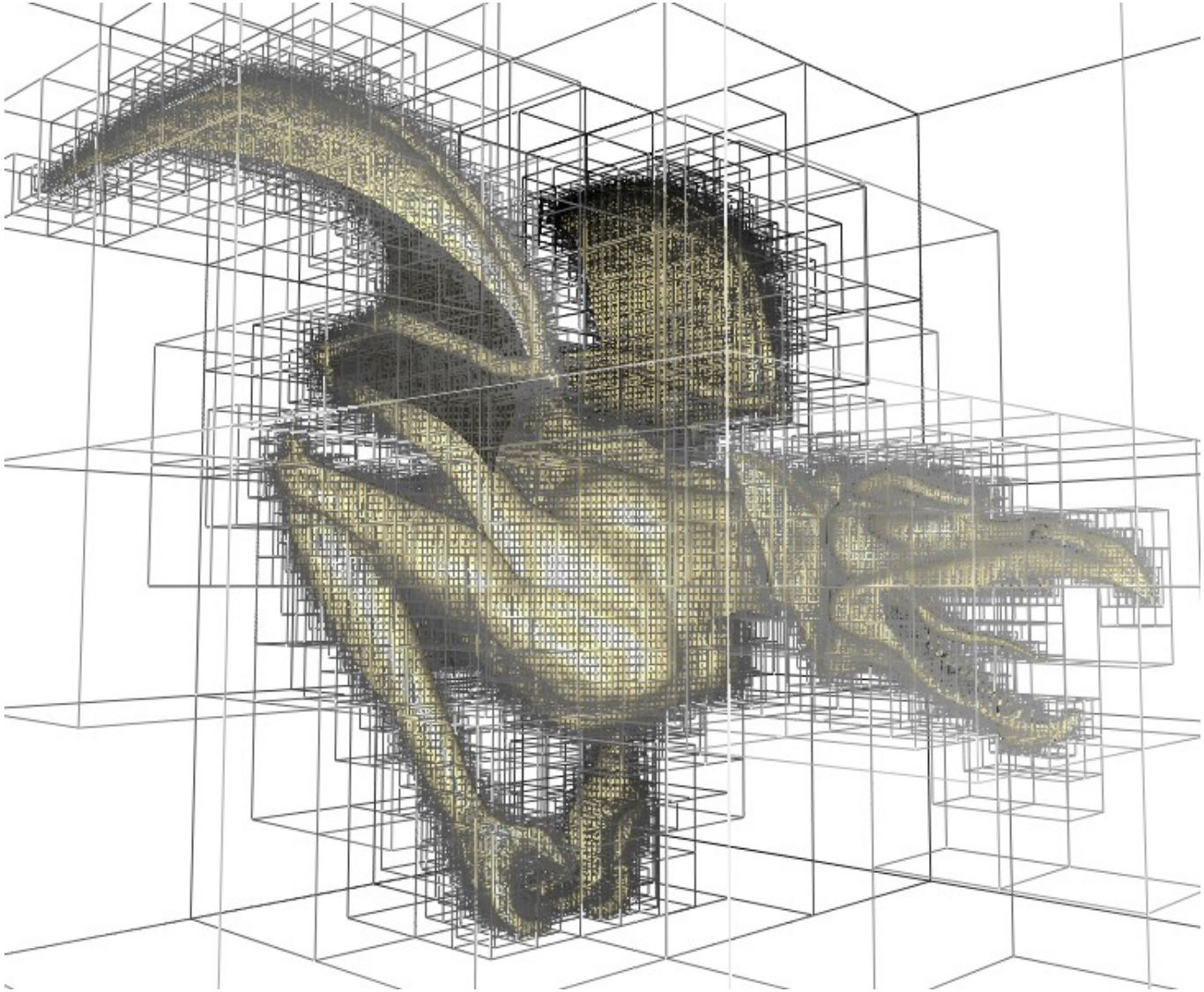
Octree Examples



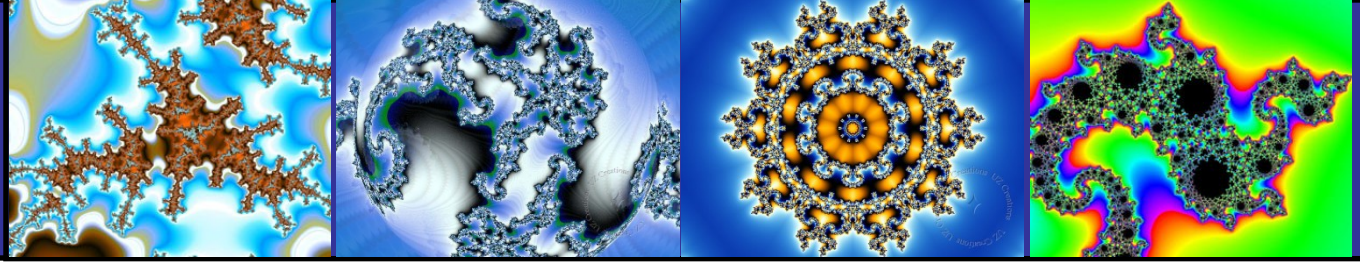
Octree containing pieces of an implicitly defined sphere; within each terminal node surface vertices are computed and connected to form a polygon

Taken from http://www.unchainedgeometry.com/jbloom/images.html

Octree Examples (cont…)



Taken from http://www-evasion.inrialpes.fr/Membres/Sylvain.Lefebvre/these/



Fractals

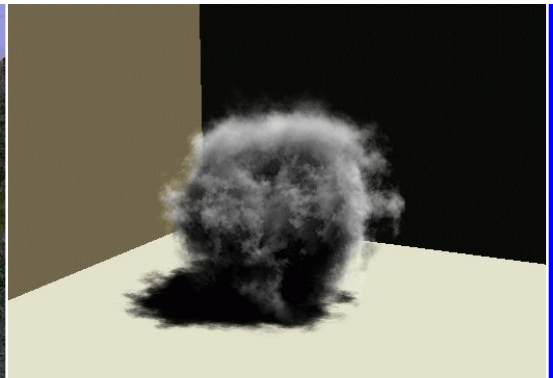
All of the modelling techniques covered so far use Euclidean geometry methods

- Objects were described using equations

This is fine for manufactured objects

But what about natural objects that have irregular or fragmented features?

- Mountains, clouds, coral…



Fractal Geometry Methods & Procedural Modelling

Natural objects can be realistically described using **fractal geometry methods**

Fractal methods use procedures rather than equations to model objects - **procedural modelling**

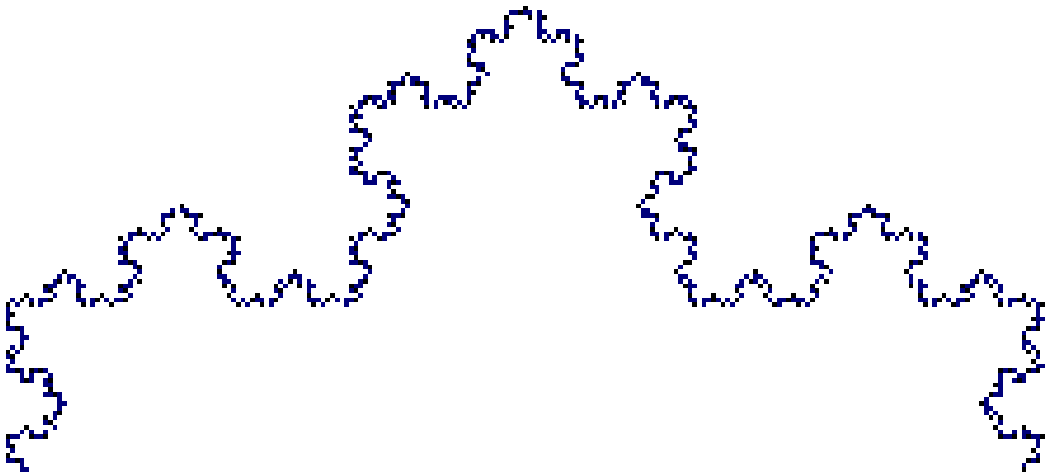
The major characteristic of any procedural model is that the model is not based on data, but rather on the implementation of a procedure following a particular set of rules

Modelling On The Fly!

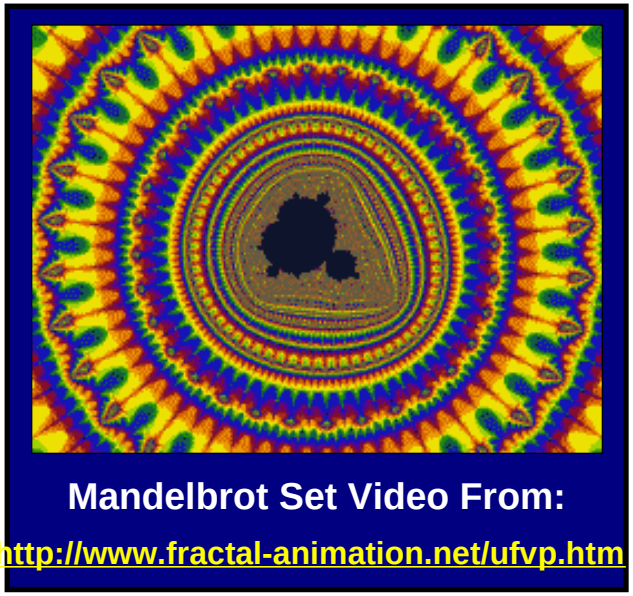
Fractals

A fractal object has two basic characteristics:

- Infinite detail at every point
- A certain self similarity between object parts and the overall features of the object



The Koch Curve



Mandelbrot Set Video From:

http://www.fractal-animation.net/ufvp.htm

Generating Fractals

A fractal object is generated by repeatedly applying a specified transform function to points in a region of space

If $P_0 = (x_0, y_0, z_0)$ is a selected initial position, each iteration of a transformation function F generates successive levels of detail with the calculations:

$$P_1 = F(P_0), \quad P_2 = F(P_1), \quad P_3 = F(P_2), \quad \ldots$$

In general the transformation is applied to a specified point set, or to a set of primitives (e.g. lines, curves, surfaces)

Generating Fractals (cont…)

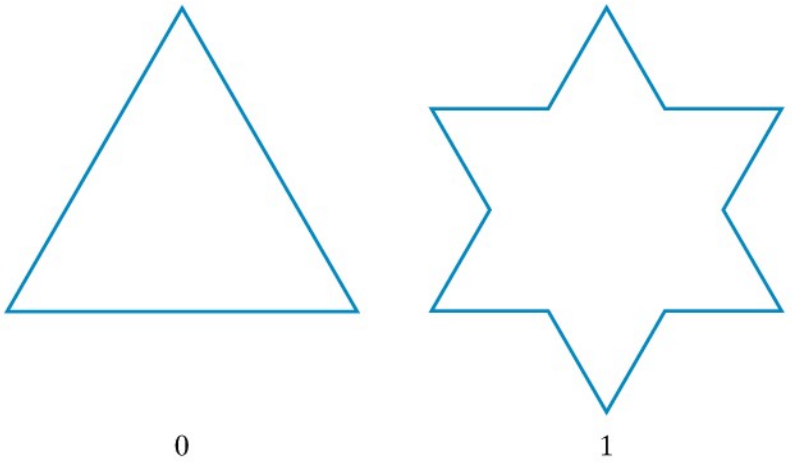
Although fractal objects, by definition have infinite detail, we only apply the transformation a finite number of times

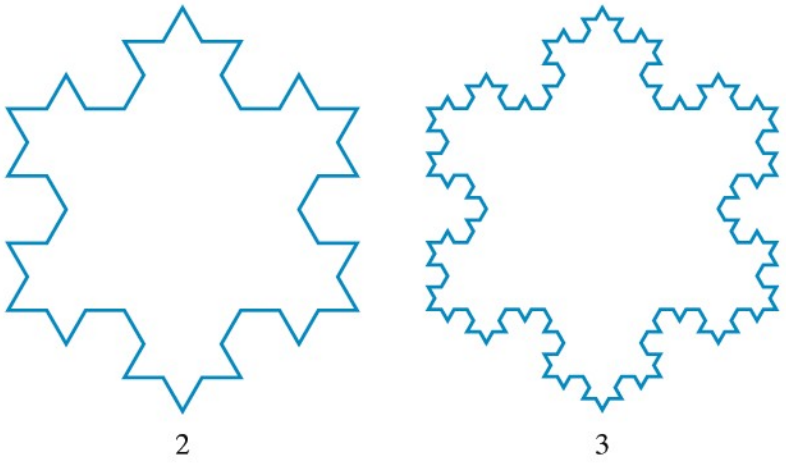
Obviously objects we display have finite dimension – they fit on a page or a screen

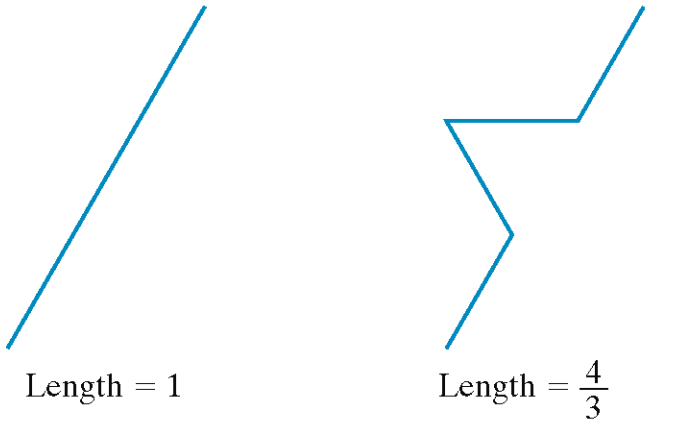
A procedural representation approaches a *true* representation as we increase the number of iterations

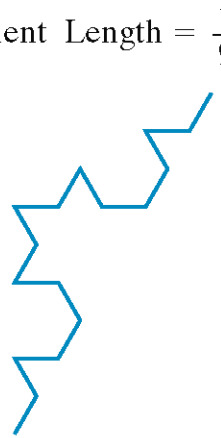
The amount of detail is limited by the resolution of the display device, but we can always *zoom in* for further detail

Example: The Koch Snowflake





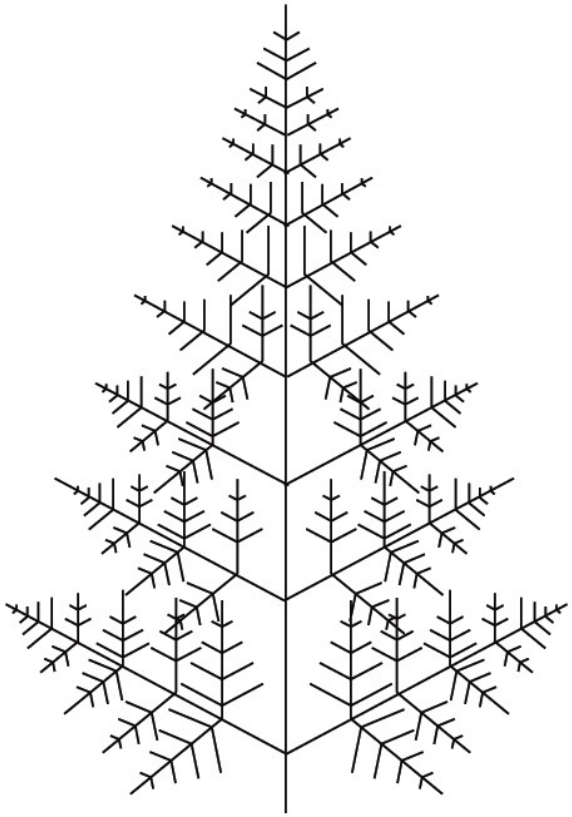


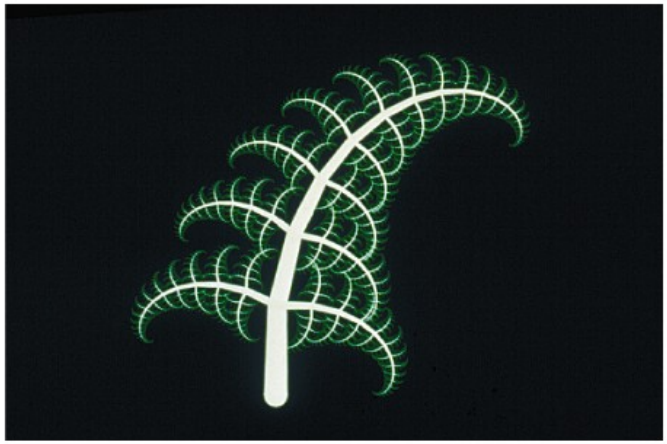


Images taken from Hearn & Baker, "Computer Graphics with OpenGL" (2004)

Example: Ferns

Very similar techniques can be used to generate vegetation





Fractal Dimension

The amount of variation in the structure of a fractal object is described as the **fractal dimension**, D

- More jagged looking objects have larger fractal dimensions

Calculating the fractal dimension can be difficult, especially for particularly complex fractals

We won't look at the details of these calculations

Types Of Fractals

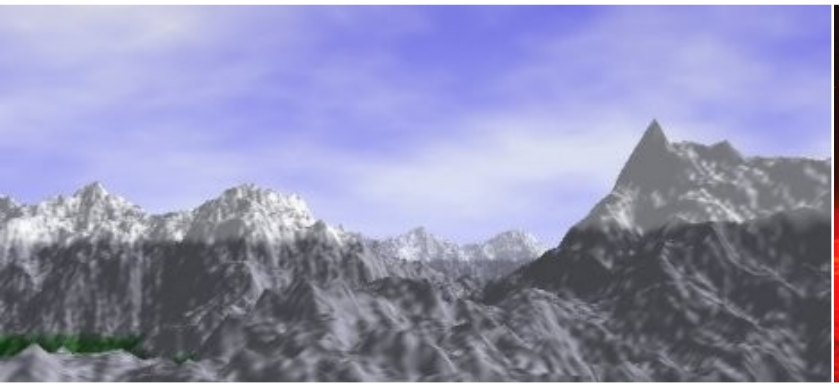
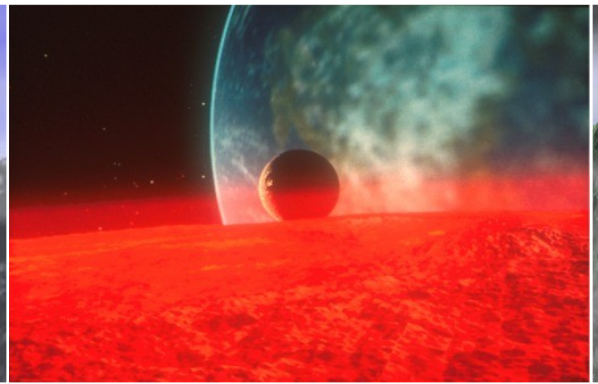
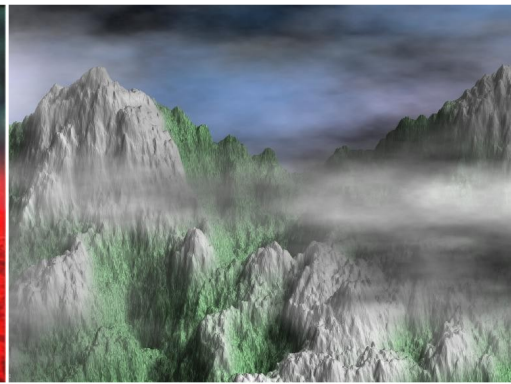
Fractals can be classified into three groups

- Self similar fractals
  - These have parts that are scaled down versions of the entire object
  - Commonly used to model trees, shrubs etc
- Self affine fractals
  - Have parts that are formed with different scaling parameters in each dimension
  - Typically used for terrain, water and clouds
- Invariant fractal sets
  - Fractals formed with non-linear transformations
  - Mandelbrot set, Julia set – generally not so useful

Random Midpoint Displacement Methods For Topography

One of the most successful uses of fractal techniques in graphics is the generation of landscapes

One efficient method for doing this is **random midpoint displacement**

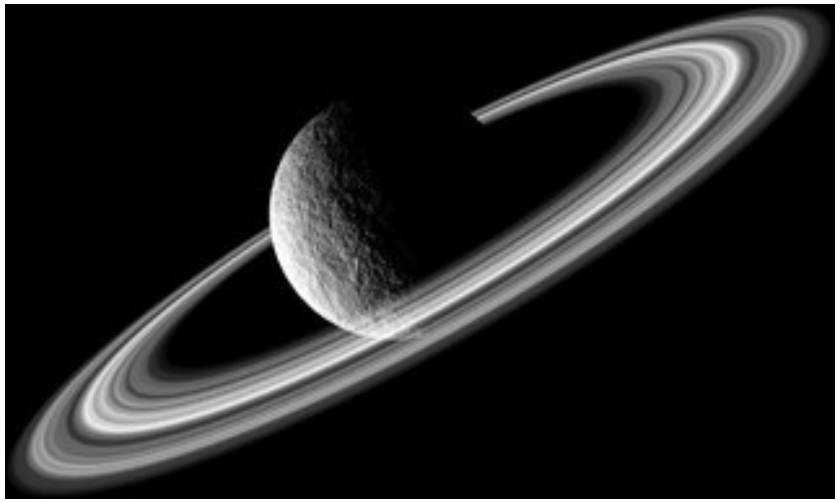




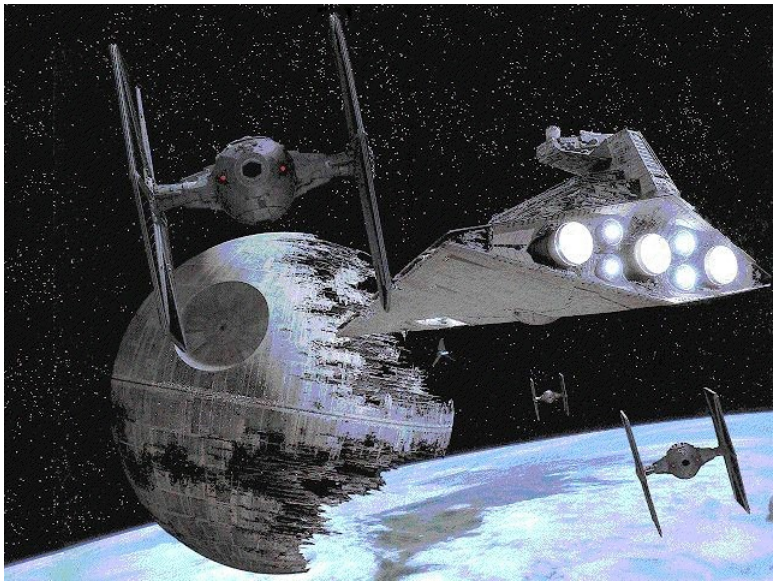
Random Midpoint Displacement Methods For Topography (cont…)

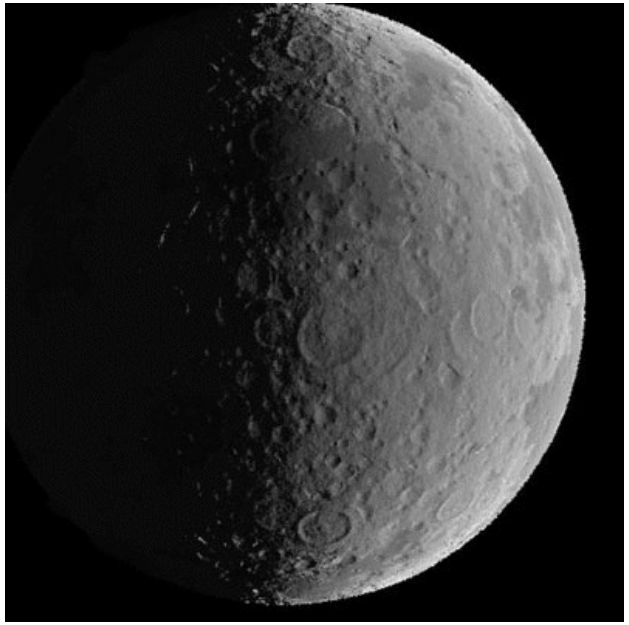
- Easy to do in two dimensions
- Easily expanded to three dimensions to generate terrain
- Can introduce a roughness factor H to control terrain appearance
- Control surfaces can be used to start with a general terrain shape

Terrain generation demo:
http://world.std.com/~bgw/applets/1.02/MtFractal/MtFractal.html

Fractals In Film Special Effects







Summary

In today's lecture we looked at how octrees and fractals are used in modelling

Fractals in particular are a fairly exotic modelling technique, but can be extremely effective

Next time we will look at curved surfaces which are extremely important