

On-Line Computer Graphics Notes

BOUNDING BOXES

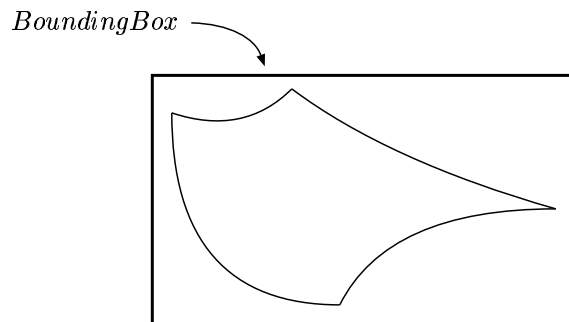
Kenneth I. Joy
Visualization and Graphics Research Group
Department of Computer Science
University of California, Davis

Overview

In implementing computer graphics algorithms, we frequently need a low-cost method to “keep track of” an object without having to continually reference the object’s mathematical complex definition. In many cases, a *bounding box* can be placed about the object and the algorithms can refer to the box when necessary, rather than the object.

What is a Bounding Box?

A bounding box for an object is just a rectangular box in three-dimensional space, with sides parallel to the coordinate planes, that contains (or surrounds) the object. This illustration below shows a two-dimensional box surrounding a curved object.



How Do We Construct Bounding Boxes?

If we are given an object represented by a set of points

$$\mathbf{P}_0 = (x_0, y_0, z_0),$$

$$\mathbf{P}_1 = (x_1, y_1, z_1),$$

⋮

$$\mathbf{P}_n = (x_n, y_n, z_n)$$

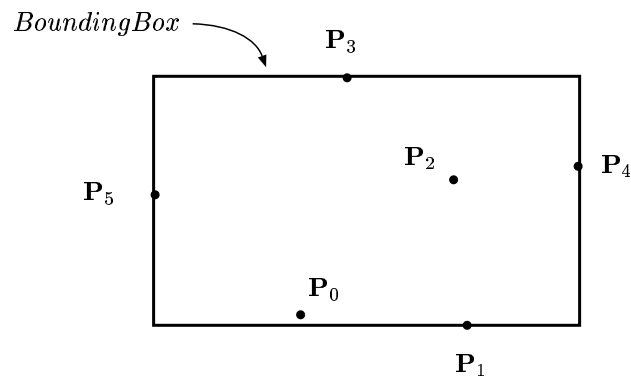
(e.g. the vertices of a polygon, or the control points of a Bézier Patch), we can build a bounding box about the object simply by defining the box to be

$$\min_{0 \leq i \leq n} \{x_i\} \leq x \leq \max_{0 \leq i \leq n} \{x_i\}$$

$$\min_{0 \leq i \leq n} \{y_i\} \leq y \leq \max_{0 \leq i \leq n} \{y_i\}$$

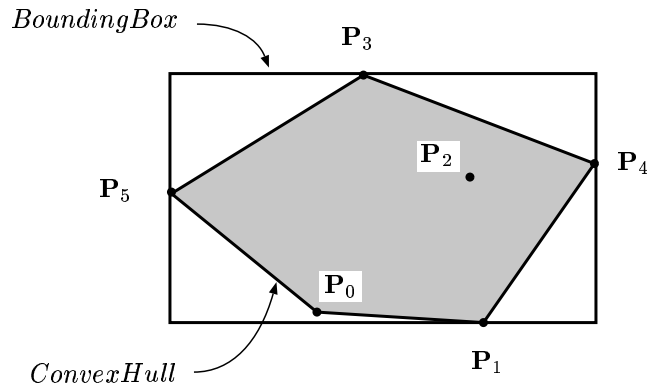
$$\min_{0 \leq i \leq n} \{z_i\} \leq z \leq \max_{0 \leq i \leq n} \{z_i\}$$

Of course, many other boxes may be defined which contain the object, but this is a minimal box that has sides parallel to the coordinate axes.



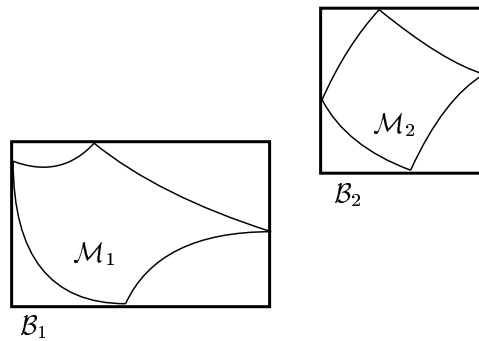
In Relation to the Convex Hull ...

If we consider the above points $\mathbf{P}_0, \mathbf{P}_1, \dots, \mathbf{P}_n$, the bounding box is a convex set containing the points. Since we know that the convex hull is the smallest convex set that contains all the points, it must also be contained in the bounding box. That is, a bounding box contains the convex hull.



A Simple Intersection Test

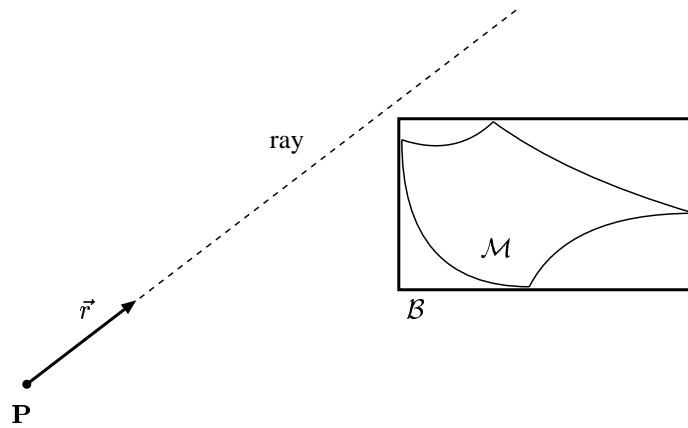
If we have two complex models \mathcal{M}_1 and \mathcal{M}_2 and we wish to see if these models do not intersect, we can use a “bounding-box test” to give a quick initial answer. If \mathcal{B}_1 and \mathcal{B}_2 are bounding boxes containing \mathcal{M}_1 and \mathcal{M}_2 respectively, it is easily seen that \mathcal{M}_1 and \mathcal{M}_2 cannot intersect if the two bounding boxes do not intersect.



This test can only be used to see if the objects do not intersect, it cannot be used to test for intersection. If the bounding boxes do intersect, we can only say that it is possible that the objects intersect. We cannot say it with certainty.

A Ray/Object Intersection Test

Suppose we have a complex models \mathcal{M} and we wish to see if a ray intersects \mathcal{M} . This is normally a complex operation, and we can simplify it somewhat by using a simple “bounding-box test” to see if the ray misses \mathcal{M} .



By placing a bounding box \mathcal{B} around \mathcal{M} , we first see if the ray hits \mathcal{B} , and if not, we know that the ray does not hit the model \mathcal{M} . Of course, if the ray hits the bounding box, we then must test it against \mathcal{M} for intersection which may be expensive. But by testing first against the bounding box, we can eliminate a number of complex expensive calculations.

All contents copyright (c) 1996, 1997, 1998, 1999
Computer Science Department, University of California, Davis
All rights reserved.