Advances in Computer Science Research (ACSR), volume 73
7th International Conference on Education, Management, Information and Computer Science (ICEMC 2017)

Research on Collision Detection in 3D Games

Li-na Ye
Wuhan Institute of Shipbuilding Technology Wuhan China
504292159@qq.com

Keywords: Collision detection; AABB; OBB; K-DOP

Abstract. This paper first introduces the basic principle of collision detection and collision of several commonly used detection method, and make comparison of them, and point out the advantages and disadvantages, finally put forward a fast collision detection algorithm based on optimal bounding sphere, and on this basis, the realization of a simple application of collision detection technology is an instance of Demo.

Introduction

The advances in technology and the constant demands and upgrades of the game experience pose even greater challenges for the game environment, especially in 3d scenarios. In real life, collisions always happen naturally, such as playing football and playing basketball, we always need to control the direction and force of the ball. Since the advent of 3D games, programmers have been working hard to simulate more accurate real-world realities. To more realistically simulate the real environment, scholars experience to develop more high quality games, will have to be to various physics phenomena existing in reality into the game development process, in which most of the game without collision detection, which whether the collision detection between the object and scene, and collision detection between object and object, collision detection technology is one of the most critical problems. When we fly in the game through the wall to wall balls, or is driving the car and obstacles appeared interference, will greatly reduce the gaming experience reality, thus, collision detection technology really become an important topic in the study of game scenario development, and aimed at the technology abroad many famous game developers have embarked on the exploration of the related work.

The Basic Principle of Collision Detection

Collision detection is to detect whether interference occurs between the virtual objects, is one of the key issues in the virtual reality technology, it is judgment graph together with the virtual scene, to meet the space-time law of the consistency of the conditions, namely the game in the scene to identify the relative relationship between object to decide whether to trigger new corresponding game events, simply put, is the intersection test between a polyhedron. The collision detection in 3D games is much more complicated than the collision detection in 2D games.

The most common situation in 3D game development is one or some objects moving in a static environment. The following three conditions are typically described by collision detection:

- Model the collision join points and the collision response
- Detect the collision between two objects
- Test the accuracy of collision and join points

From the rendering perspective, the first description requires a dynamic model, where the two objects overlap, and the collision occurs. The latter two are geometric operations.

A Collision Detection Method Based on a Hierarchy Surround Box

The Basic Principle of the Axial Surround Box (AABB) Detection Method

A 3D AABB is simple hexahedron, each side parallel to a coordinate plane, and the bounding box are not necessarily the cube, the length, width and height can be different from each other. This method is relatively simple, from two-dimensional collision detection to one dimensional space operation, when two AABB bounding box intersect, if and only if they are on the X axis and Y axis and Z axis are overlapping, the projection of the projection as long as there is a direction not overlap, then they don't intersect. The intersection test between AABB bounding volume and update speed is the fastest, so the most widely used, especially suitable for multiple objects motion mass environment and deformation of collision detection.

The Basic Principle of the Compass Box (OBB)

The OBB's intersecting test is based on the theory of the separation axis, which is defined as a positive hexahedron that contains the object and is arbitrary relative to the axis. Its best direction must ensure that the box is the smallest in the direction of the box, thus forming a division within it. Because of its arbitrary direction, good tightness, the execution of collision query, may only need to traverse less hierarchy tree, its overlapping detection speed faster, greatly improving the efficiency of the algorithm.

The Basic Principle of the Discrete Directional Polyhedron (K-DOP) Test

The concept of K-DOP was first proposed by Kajiya and Kay, using the parallel plane of k / 2 to surround the object, which is the extension of AABB. he advantage of this approach is to increase the encirclement efficiency of AABB while retaining its advantage over the OBB. By adjusting the value of k, can reach a certain in the simplicity and compactness of compromise, so as to improve the efficiency of collision detection, so it can satisfy a efficient hierarchical structure should have the conditions of the scene.

AABB, OBB, K-DOP Detection Methods are Compared

AABB method is used as a model of bounding box, while the intersection test is simple but tightness is poor, and follow the sex difference, computation is large, it will significantly slow the progress of the application, in addition of the problem of the precision of this method still faces. The OBB method also requires a large amount of computation and is slow to be used for dynamic object detection. Another drawback of using the OBB approach is that a large number of matrix operations are required. K-DOP method between the interfaces of the foregoing two methods of a kind of method, its characteristic is as long as the plane parallel to the number of selected reasonably and direction, can detect the simplicity of the flexible trade-off between and the compactness of wrapped object, but is still large amount of calculation, and the parallel to the plane of the selection is a difficult problem. The problem of collision detection between multiple virtual objects in complex virtual environments in dynamic environments is inefficient.

Algorithm Optimization and its Operation Result

In view of the above all sorts of bounding box test methods of comparison, this paper is on polyhedron collision detection using a sphere package every part of the object or objects, and then test whether these spheres intersect. The cost of this method is lower. Although surrounded by ball method algorithm is simple, but complicated virtual game in dynamic environment of collision detection problem between virtual objects in the scene when the same brought considerable computational load to the processor. This article makes the following optimization for this approach:

1）Filtration method

This optimization is logical, that is, the concept of using a test range. Based on the development of engine code for different polyhedron surrounded the ball when designing the data structure in which joined a Boolean flag bit, surrounded by first detected between frame and frame the sphere of space position is changed, the test just surrounded by calculating the sphere of 3D coordinate is changed, only involves the addition and subtraction calculation will not burden to larger additional processors, but it is very effective.

2）Formula optimization method

In 2D graphics, for example, as shown in figure1 in the outside of the object to draw a circle, A test for the collision of the sideline, whether any object to its collision happens, are judged by the circle and the distance of the center of the circle, as the two center distance is equal to or less than the sum of the radius of the two objects collide circumference, means that the two objects collision happened. Two objects such as B and C, when the distance of the center of the circle is equal to two radius and, that is two collision detection of tangent circle, then we can judge that two objects collide.

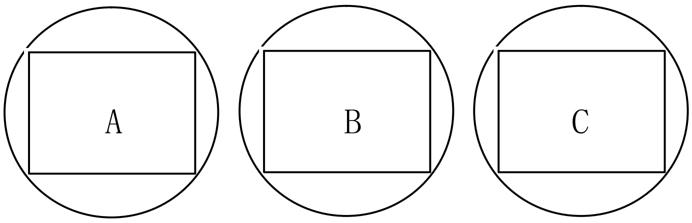

Fig. 1 Objects A, B, and C are surrounded by A circle

The code that usually applies the two points is written as follows in the code implementation of collision detection:

Distance=sqrt((x2－x1)*(x2－x1)+(y2－y1)*(y2－y1)) (1)

As the writing of the code, there is no error, but the SQRT function occupies about 70 CPU cycles, consumed CPU time very much, if it is a simple collision detection between several objects can afford, but when the object is very long, game frames will drop dramatically, cannot satisfy the real-time requirement. In response to this, this paper proposes to simplify this operation by using the Taylor expansion approximation, which is the following:

```
#deifne  min(a,b)((a<b) ? a:b)
#deifne  max(a,b)((a>b) ? a:b)
#deifne  swap(a,b,t){t=a;a=b;b=t;}
int  Fast_Distance_2D(int x,int y)// To calculate the distance between two points in 2D
{
x=abs(x);
y=abs(y);
int mn=min(x， y);
retum (x+y-(mn>>1)-(mn>>2)+(mn>>4));
}
float  Fast_Distance_3D(float fx , float fy , float fz)// To calculate the distance between two points in 3D
{
int  temp;
int  x,y,z;
x=fabs(fx) * 1024;
y=fabs(fy) * 1024;
z=fabs(fz) * 1024;
if(y<x)swap(x,y,temp);
if(z<y) swap(y,z,temp);
if(y<x)SWAP(x,y,temp);
int  dist=(z+11 * (Y>>5) + (x>>2));
return((float)(dist>>10));
}
……
```

Based on the algorithm of collision detection optimization described above, this paper presents a simple demo of 3D real-time collision detection. Run results below Fig2 left of the current frame only car position has changed, the coordinates of two obstacles are stationary, only two tests, the right for a collision is detected after the collision response effect of screenshots.

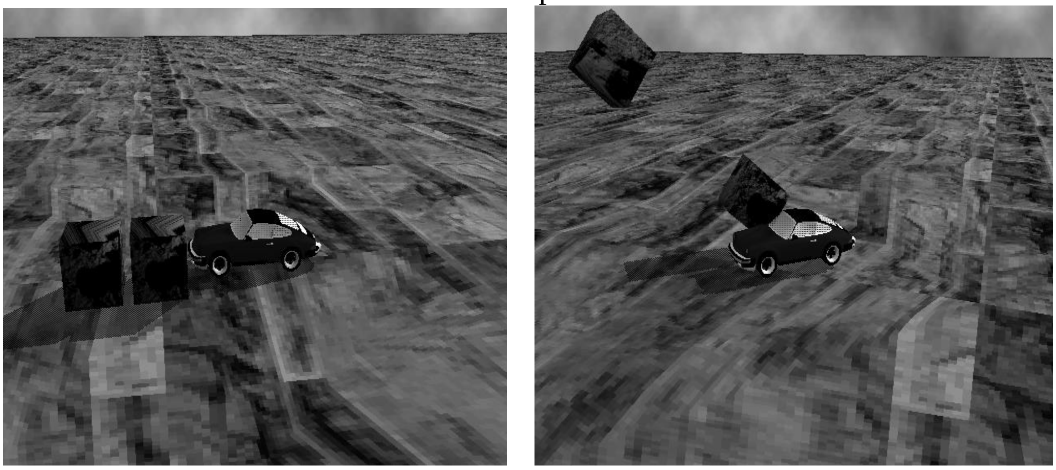

Fig. 2 3DSimulation of multi-object collisions in real time（From the demo in this paper）

Reference

[1] Earnshaw R A, Gigante M A, Jones H: Virtual Reality Systems, Academic Press Limited 24/28, Oval Road London NW1 7DX,1993,135-237.
[2] Z.Q Wang: Research review of collision detection, Journal of software, 1999, Act(5), 545-551.
[3] Christer Ericson: Real-time collision detection algorithm technology (Beijing tsinghua university press 2010).
[4] Wei Zhao, P.R. Tan and Yong Li: Real-time collision detection algorithm in complex virtual environment, System simulation journal, 2010.