Available online at www.sciencedirect.com

SciVerse ScienceDirect

AASRI Procedia 1 (2012) 505 – 511

AASRI Procedia

www.elsevier.com/locate/procedia

2012 AASRI Conference on Computational Intelligence and Bioinformatics

A Study on Dynamic Collision Detection of Armored Vehicle in Visual Simulation

Meng Qinghua*, Gao Wei, Guo Xi, Li Qiuyan

Changchun Institute of Engineering Technology, Changchun, 130117, China

Abstract

Under the visual simulation, some cases often are happened such as an armored vehicle penetrating or penetrating through other entities. However, the various existed bounding box models can't resolve this trouble well. A bounding box method is introduced, the collision test concerning the box have been performed analysis under translation and rotation for an armored vehicle. In order to resolve the trouble existed, the author provides the method of dynamic collision detection in this paper, and set up the dynamic collision detection models for an armored vehicle accordingly. Utilizing OpenGVS through VC++ to create simulation. At the last, feasibility of those models are tested and verified by the contrast tests from simulation results.

Keywords: Visual Simulation; Armored Vehicle; Bounding Box; Dynamic Collision Detection

1. Introduction

Bounding box is a widely used method in collision detection algorithm, and has been thoroughly researched in many applied areas (such as virtual reality and so on) of computer graphics [1], whose basic idea is to use a bounding box with somewhat large volume and simple geometric characteristics to describe a

* *E-mail address:* mqhwolf@163.com.

doi:10.1016/j.aasri.2012.06.079

complex geometric object approximately, and then to approximate the object's geometric model as closely as possible by constructing tree structure until its geometric characteristics are almost fully obtained, and the collision detection process can also be accelerated through the pre-treatment process (bounding box) [2, 3]. This method can handle the collision detection problem of static entities well. However, it often appears such phenomena as armored vehicles penetrating or penetrating through other entities in a virtual scene, especially in high-speed motion in the wild. In order to solve this problem, the authors put forward a method of dynamic collision detection for armored vehicle.

2. Entity Motion and Its Collision Detection Analysis

In a virtual scene, the entity is given certain speed, with which the location of the entity at each frame is computed; then the graphics engine is refreshed at a speed no more than 30ms per frame; different locations are set up for the entity at each refreshing so that the entity is in motion visually. Collision detection is conducted through the bounding box; if the motion speed is slow, the ground object can be detected before the entity reaches it, and thus not penetrated (as in Fig. 1a); otherwise if the motion speed is fast, the ground object may be penetrated by the entity when the latter is at the location at time i+1 (as in Fig. 1b), creating such a distortion phenomenon.

In the figure, assume that the studied moving entity is a regular cuboid; against its exterior surface is a bounding box for collision detection. For the purpose of study, overlooking observation is conducted. The solid line represents the entity; the dotted line represents the face of the bounding box, perpendicular to the ground surface.

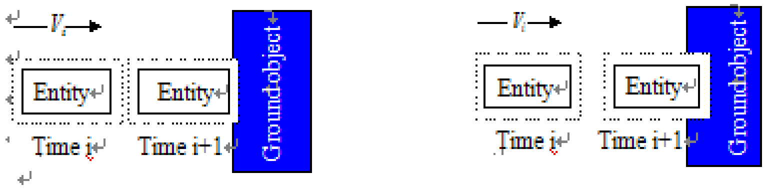

Fig. 1. (a) the ground object not penetrated by the entity; (b) sthe ground object penetrated by the entity

The motion pattern of a moving object is relatively complex; so the motion process of the entity is simplified for the sake of the study. The following study mainly concerns rigid entities. First, the collision detection is carried out when the entity is in translation on horizontal ground; then the collision detection of the entity in fixed-axis rotation is analyzed. Because complex motion of entities can all be reduced to the combination of these two basic motions [4].

2.1. Entity translation and its collision detection analysis

Translation is the most common movement in engineering and life, and is also the simplest movement [4]. Therefore, to study the motion of an entity in a virtual scene should also start from the study of translation. In translation, an entity can move in a straight line or along a curved path. But in a virtual scene, the motion of the entity is embodied by the change in its location at each frame. The location of the entity at the next frame is only associated with its location and speed at the current frame; so the translation between two frames can be regarded as linear motion.

The entity is in translation in the geodetic coordinate system (left-handed system) *oxyz.* The coordinate of the reference point o' of the entity at time i in the geodetic coordinate system is set as (x_i, y_i, z_i), the speed of the entity v_i, and its direction of motion (α_i, β_i, γ_i). α_i, β_i, γ_i refer to the angles formed in terms of the left-hand rule when the entity rotates around *X, Y, Z* axis at time *i* [5].

The coordinate of the reference point o' at time *i+1* in the geodetic coordinate system is (x_{i+1}, y_{i+1}, z_{i+1}), as is shown in Fig. 2.

The computation is as follows in terms of Euler method:

Where, Δt_i means the time of the frame at time *i*.

$$\begin{cases} x_{i+1} = x_i + v_i \Delta t_i \cos \alpha_i \cos \beta_i \\ y_{i+1} = y_i + v_i \Delta t_i \sin \alpha_i \\ z_{i+1} = x_i + v_i \Delta t_i \cos \alpha_i \sin \beta_i \end{cases} \tag{1}$$

Since the entity moves on the horizontal ground, the angle of pitch is 0, i.e. $\alpha_i = 0$. So the value of (x_{i+1}, y_{i+1}, z_{i+1}) is computed by x_i, y_i, z_i, v_i, Δt_i, β_i.

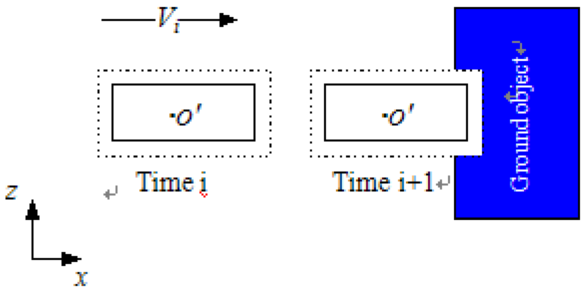

Fig. 2. the entity in translation in the coordinate system oxyz

When v_i is fast, the entity will penetrate the ground object at time i+1.

2.2. Fixed-axis rotation of entity and its collision detection analysis

Fixed-axis rotation is the most common movement in engineering. The entity rotates around a fixed axis in the geodetic coordinate system (left-handed system) oxyz. The coordinate of the reference point o' of the entity at time i in the geodetic coordinate system is set as (x_i, y_i, x_i); the entity rotates around the line that passes the point o' and is perpendicular to the ground surface. The current angle is φ_i and the angular speed is ω_i.

The angle of the entity at time i+1 is φ_{i+1}, as is shown in Fig. 3.

The computation is as follows in terms of Euler method [6]:

Where, Δt_i means the time of the frame at time i.

$$\varphi_{i+1} = \varphi_i + \omega_i \Delta t_i \tag{2}$$

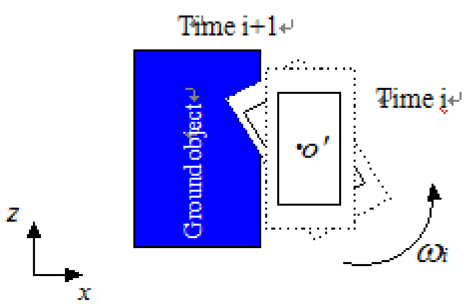

Fig. 3. the entity in fixed-axis rotation in the coordinate system oxyz

When ω_i is fast, the entity will penetrate the ground object at time i+1.

3. Dynamic Collision Detection Model of Armored Vehicle

In light of the above analysis, the static collision detection model (i.e. static bounding box) of armored vehicle entity (entity for short) is first constructed. This model is defined as :

$$M_s: \langle T, X, Q, Y, \lambda \rangle \tag{3}$$

Nomenclature

T	the time set, determining the time step length;
X	the space of the bounding box, through which the external environment acts on the model; for an entity, the space of its bounding box varies with the level requirements
Q	the status of the entity; it is the core of modeling and determines the boundaries of the bounding box; the bounding boxes of different entities are generally different
Y	the output set, through which the result of collision detection can be obtained
λ	the output function; when the status of the entity is Q, the input is X, Y can be detected by λ (T, X, Q)

For the purpose of analysis, the bounding box is also a cuboid, and the specifics are as follows:

$$Y = \lambda(T, X, Q) = \begin{cases} x_a \le x \le x_b \\ y_a \le y \le y_b \\ z_a \le z \le z_b \end{cases} \tag{4}$$

x_a , x_b , y_a , y_b , z_a , z_b refer to the boundary coordinates of each face of the bounding box in the entity’s coordinate system; after they are converted to the geodetic coordinates by coordinate transformation , the collision information can be obtained by this bounding box.

Since the output function changes with the purpose of the bounding box, it is required that users define this model according their own requirements.

3.1. Collision detection model of armored vehicle in translation

It can be found by analyzing entity motion and its collision detection that the essence of collision detection of an entity in motion is to detect through a model, and thus know in advance the location of the entity at the next frame and judge whether the entity at this location will overlap or intersect other entity. As is shown in Fig. 4, the ground object is not detected by the entity at time i through the static collision detection model, and since the speed of the entity v_i is fast, it penetrates the ground object at time i+1.

To avoid such a phenomenon, the dynamic collision detection model is thus introduced (see Fig. 2). At time i, the boundary of the entity's static collision detection model is extended by L_i, and the bounding box becomes one with the space encircled by the dotted line. Through this bounding box, the collision with the ground object at time i+1 can be detected by the entity at time i, and the location of the ground object can be obtained. Accordingly, the location and motion status of the entity at time i+1 can be set so as to avoid the collision with the ground object at time i+1, and thus avoid the occurrence of penetrating the ground object.

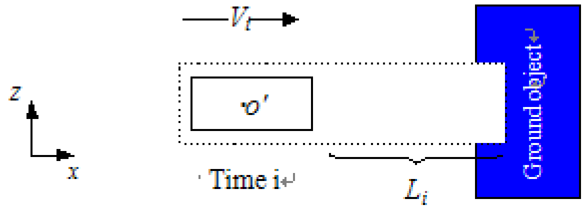

Fig. 4. Dynamic collision detection of entity in translation

The bounding box is extended in the direction of v_i, and the extended space is determined by L_i (related to v_i and Δt_i) and the maximum cross section perpendicular to v_i, that is the size of the bounding box is in dynamic change with the speed of the entity.

$$L_i = v_i \, \Delta t_i \tag{5}$$

The collision detection of the entity in translation is defined as:

$$M_{d1}: <T_{d1}, X_{d1}, Q_{d1}, Y_{d1}, V_{d1}, \lambda_{d1}> \tag{6}$$

Nomenclature

T_{d1} ;Y_{d1}	defined in the same way as T and Y in formula (3)
V_{d1}	the speed of the entity in translation
X_{d1}	the space of the bounding box of the entity in translation, affected by Q_{d1} and V_d and changeable
Q_{d1}	the status of the moving entity, affected by V_d and changeable
λ_{d1}	The output function

When the status of the entity is Q_{d1}, the input is X_{d1}, Y_{d1} can be detected by λ (T_{d1}, X_{d1}, Q_{d1}, V_{d1}); the specifics are as follows:

$$\lambda(T_{d1}, X_{d1}, Q_{d1}, V_d) = \begin{cases} x_a \le x \le x_b + v_i \Delta t_i \\ y_a \le y \le y_b \\ z_a \le z \le z_b \end{cases} \tag{7}$$

4. Simulation Realization and Verification in Virtual Scene

Based on the above models, the simulation of this model is realized by using OpenGVS as visual engine and VC++. The simulation process and results are judged and evaluated by many people, who all have considered that this model solves well the problem of armored vehicle penetrating or penetrating through other entity.

Model verification is conducted and completed by contrast test. First, a wheeled vehicle is used as the rigid entity in the visual scene; then the static bounding box and dynamic bounding box are used to carry out collision detection: passing the same location (relative to the same reference) in the visual scene at the same speed, the entity is made stop when each model detects the ground object; the locations of the entity are observed and the results are compared; the result of evaluation is finalized after a number of experiments. Fig. 5 and Fig. 6 show one contrast test.

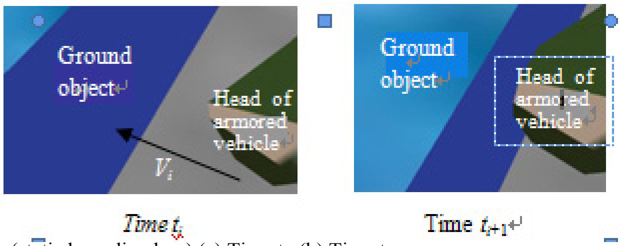

Fig. 5. Static collision detection (static bounding box) (a) Time t_i; (b) Time t_{i+1}

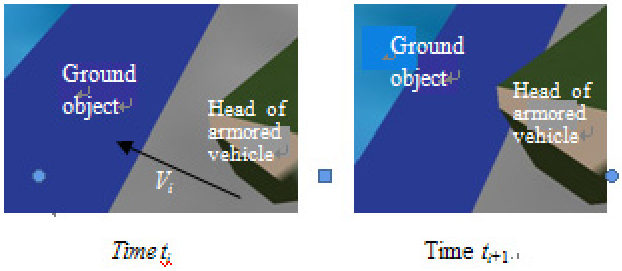

Fig. 6. Dynamic collision detection (a) Time t_i; (b) Time t_{i+1}

It can be seen from the above experiment results that the head of armored vehicle in Fig. 5 penetrates the ground object at time t_i+1; while the head of armored vehicle in Fig. 6 stops at the outside of the ground object.

5. Conclusion

In this paper, a dynamic collision detection method is put forward for the distortion phenomenon of armored vehicle penetrating the ground object in visual scenes, and a dynamic collision detection model is constructed. The model is verified to be of good detection effect of collision after simulation and a number of contrast tests, and thus capable of solving the problem mentioned above. The shortcoming of this model is its failure to incorporate the specific shape of armored vehicle in the model, which will be considered in future studies.

References

[1]Li Xiaoyan, Yin Hong and Gao Yunfei. A Fast Collison Detection Aglorithm of Tank In Virtual Battlefield [J]. Command Control and Simulation, 2006, 28(4): 78-81

[2]Hubbard P M.Collision Detection for Interactive Graphics Application[J]. IEEE Transactionon

Visualization and Computer Graphics, 1995, 1(3): 218-230
[3]Fabio Policarpo, Aura Conci. Real-Time Collision Detection and Response[J]. IEEE Computer Society (S0018-9162). 2001, 1:376-376
[4]University of Science and Technology Beijing. Engineering Mechanics [M]. Beijing: Higher Education Press, 1997