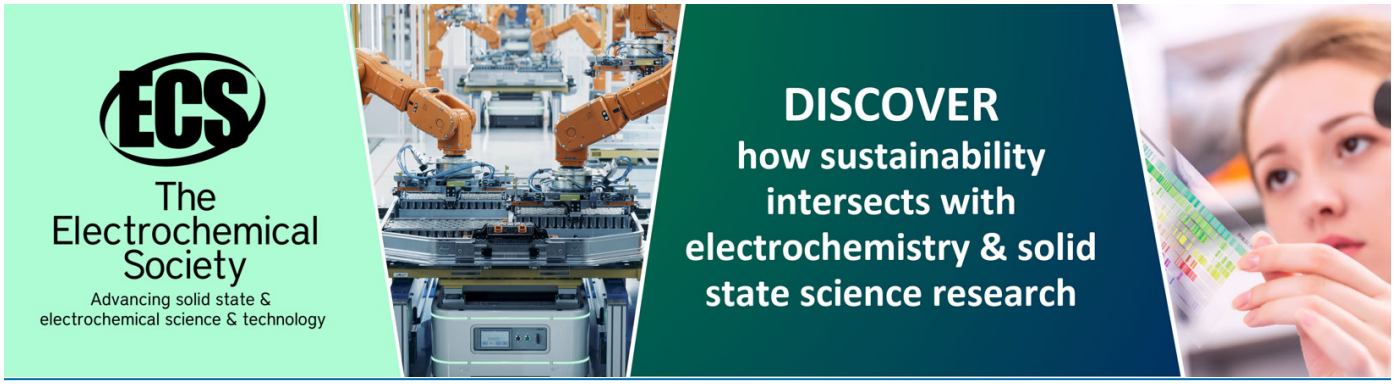
The application of 3D technology in video games



IOP Conf. Series: Journal of Physics: Conf. Series **1087** (2018) 062024 doi:10.1088/1742-6596/1087/6/062024

The application of 3D technology in video games

Yuexiang SU

College of Software Engineering, Beijing University of Technology, Beijing100124, China

494797152@qq.com

Abstract. With the growing maturity of computer technology and the rapid development of the video game industry, players have higher needs for the game display technology. They are no longer satisfied with the 2D game, yet hope to experience a more authentic 3D scene in the 2D display. Therefore, in the research and development process of game design, game designers continue to pursue a higher quality of the game and the real effect. And implementing 3D technology, it will have the 3D effect on the 2D display. In this context, 3D games have gained extensive research and rapid development. This paper analyzes and discusses the application of 3D technology in video games. By building up the 3D game engine and game client-side based on 3D technology, this paper will also describe the entire 3D game development in the direction of process development.

1. Introduction

3D technology is one of the most sophisticated technologies in high-tech today, and it is used in many fields of people's life. Reasonable integration of 3D technology into video games can make qualitative change to the game. Current video games have evolved from the previous 2D effect to 3D effect, which enables video games to have vivid scenes like movies. At present, the 3D video game industry has become a representative of high and new technology in the entertainment industry, and gradually tends to mature. With the development of game technology, game content and game mode, 3D games will become an important part of people's lives.

2. Concept of 3D technology

For the current games, they can be divided into three categories according to the visual effect of them: the first is the 2D game, in which all the graphic elements are in graphic form, the ground fields and buildings are composed of single map elements; second one is 3D game, that is, all the graphic elements are composed of a number of geometric polygons of the three-dimensional graphics, the game screen is more real; the third is the pseudo 3D game, also known as the 2.5D game, which meets the various needs of the video game players through the combination of 2D technology and 3D technology.

3D virtual imaging technology, also known as 3D technology, is a branch of virtual reality disciplines, which is now widely used in virtual teaching and virtual games. This technology can visually make the color more hierarchical, forming a powerful visual effect. When 3D technology is used in video games, it can mainly bring stereoscopic visual perception. By creating three-dimensional coordinates in the game, it can design separate spaces in the virtual world, hence there is a distinct difference between the past and the 2D game. [1]

3D technology is commonly used in four aspects of video games, that is, scenario modeling, role

modeling, particle system and 3D engine.

In scenario modeling, a game scenario requires the creation of a scene in a game, including virtual indoor scenes and outdoor scenes. In virtual reality, scene generation is very demanding for real time, and it needs to simulate human vision, hearing, touching and movement in natural environment. Which needs to be applied to computer graphics, multimedia, artificial intelligence, human-computer interface technology, digital image processing, network technology, sensor technology, parallel computing technology and other information technology branches, and 3D technology can effectively solve these problems.

During the process of role modeling, with the virtual reality technology matures, people are no longer satisfied with the virtual environment that only contains the scenery, buildings and other general visual elements, but are eager to add living objectives in the virtual environment. Therefore, the role modelling and motion research of human beings in virtual environment has caused a stir in video game industry. The virtual character is the core of video games, which consists of two branches of research. One is about the facial expression animation, hand deformation, and virtual human appendages such as clothing, hair, etc.; another branch is about the human body modeling and deformation, such as motion capture, human animation tool development, human balance control and other parts of the movement control system.

In the particle system, due to the need for some special effects in the virtual scene to attract the players, such as brilliant light effect, waterfalls, snow falling, flashing flame, flame spray, bullets flying out of the barrel and effect of objects explosion, which can be perfectly achieved with the use of particle system. Special eye-catching effects produced by particle system are able to give a refreshing feeling and a strong visual shock effect. Its fast, convenient features and interesting, diverse effects, and the creation of visual effects are breath-taking.

3. The Application of 3D Technology in Game Engine

Game engine is the core of the game development, as well as the main content of the game screen display. In the game engine, the data-level function design quality will directly influence the game R&D and the normal operation of the program. The composition of the game engine is complex, whose main part includes a renderer, scene structure, movement system and collision system.

The 3D engine abstracts the material in reality into polygons or curves. It is a collection of algorithms for calculating and outputting the final image in a computer. The 3D engine is divided into immediate 3D engines and offline 3D engines based on whether they can be instantly computed by mainstream computers. Instant 3D images on PC and game consoles are generated using instant 3D engine operations.

3.1. The Renderer

The renderer design is the most crucial part of a 3D game engine, by which the 3D game can display the 3D images on the 2D screen. Therefore, the renderer is mainly divided into two parts, including renderer on the software and the one on the hardware. These two types of renderers are implemented in different fields, where the hardware renderer is mainly used for real-time rendering, such as games and virtual reality; while the software renderer is mainly used for off-line rendering, for instance, the renderings and film level and product level rendering. This is mainly because that the renderer has high speed yet low flexibility. Although the speed of the software renderer is not ideal enough, it can use very complex rendering algorithm to achieve the photo level of authenticity and effect. In fact, 3D game is to simulate the real world, so as to get a three-dimensional effect of the virtual world. Hence the virtual world can be seen as a discrete sample of vertices connected to the data flow, whereas the data flow can be processed by the renderer in order to present the 3D effect on the 2D display.

3.2. Scene System Design

In the 3D game design, all 3D objects have to be in the same 3D space, which is the scene system who has a 3D structure, and usually is known as the scene graph. In the scene graph, many interconnected

nodes eventually form a tree graph, where there will be a parent structure and a few sub structure at each node. In the design, a node with sub structure is called a non-leaf node, while the node without the sub structure is called leaf node. The space in the scene graph is a hierarchical structure, which complies with the 3D space.

In the design of scene graph, attributes relatively change, among which the most important attribute is the one of position coordinates that needs to change in the scene graph. Each node in the scene will have a separate spatial coordinates and the function of conversion to the world coordinates, where the node in the sub-structure coordinates will be established on the parent structure, so the transition of coordinates of the substructure to the world coordinates need to be achieved by its parent structure [2].

3.3. Motion system design

Pictures in the 3D game are usually dynamic, hence each node of the 3D space in the scene graph is likely to change at any time, and these changes and movement are managed by a time control program on the abstraction level. Usually, the time control program requires two core tasks. The first is to record the appearance of the movement time, that is, the node marks the movement time according to the appearance of the application, or according to its own start or end. Secondly, there is a circular classification of movements, including cyclic motion, movement during a fixed period of time, and reverse motion. In addition to the time control program, other corresponding programs are also needed in the scene map for other changes, where the motion control program is the key frame for controlling the movement of the nodes in the scene. Moreover, a particle system is also included in the moving system, which is an application structure formed by combining multiple groups of points.

3.4. Collision system design

The collision of objects in the scene are also detected by the node. In the scene graph, all the nodes are designed to be surrounded by an outer structure, when two objects collide, surrounding structure will appear to overlap. From the view of performance, the collision system is divided into dynamic collisions and static collisions. Static collisions are collisions of surface shapes, while time factors are added to dynamic collisions. At the same time, the surrounding structures around the object also need to be designed. First of all, it is necessary to design the boundary for the surrounding structure. Secondly, to set a direction, time and location attribute for each object. Once again, an independent tree structure is needed to connect it. In addition, in the collision system, it is also needed to design the pick-up program of objects, which can be divided into two ways. The first one is the implementation of pick-up on the object pixel in the renderer, which has higher precision, yet higher consumption of the system. The second is achieved by the collision system. 3D objects can interact with each other through the mouse movement, so as to complete the pick-up action. For instance, on the display, the coordinates of the mouse and orientation of the camera are the ray direction, and it finally interact with the object to realize the pick-up action [3].

4. Application of 3D technology in game client

With the support of the perfect 3D game engine, it is necessary to build a game body framework that can contain the windows, state machines, and game caching mechanisms. Generally speaking, game designers will define the game display logic part as the game client. Therefore, it can be seen that not only the online game need the design of the client, there is also a client in the console game. That is, in the design of the highest level of abstraction, it is needed to distinguish between the logical part of the game display and the core logical part. So the client in 3D game design will be defined in broad sense.

4.1. Program window design

3D games need to be run through the operating system, so the first step in the design of the client requires to build a game window for application on the operating system, and the operating system can be WINDOWS or any other types of systems. The game window should have the following functions: first of all, the program window needs to receive the transferring data and process it. Secondly, it has

to control the input and output of system, as well as completing the interaction with the operating system [4].

4.2. State machine design

State machine is a concept widely used in every step of computer programming. In the 3D game client, there will be a lot of program logics, along with a lot of different state. Therefore, state machines need to be used to assist the design on the abstract level. In other words, the client itself is a state machine whose function is to receive the information data from the system server and then to make a corresponding state.

From the logical design of the abstract level, the switching logic is equal to the switching state. If the client is seen as a state machine, when the 3D game starts running, it means the server will continue to send the game protocol data to the client, so that it constantly switches the state. At the same time, the client responds to the server data through switching states.

In the program design, most of the clients can be the form of state machine. However, it is necessary to classify the logical abstraction in concrete design. For some lower level of non-abstract logic, simple status flag can be applied, while the state machine can be implemented on higher level of abstraction. At the same time, the state machine design has another huge advantage, that since the state machines in the game are purely responsive programs, it is possible to avoid the plug-ins in games. From the perspective of design, the plug-in is only designed based on the client side, so avoiding the plug-in is to reduce the client to send instructions to the server in the core game logic. When using the state machine, it is the server who commands to control the switch state, which makes the client to become the state machine waiting for the response and command information. Therefore, the plug-in can be effectively avoided in the 3D game.

4.3. Cache mechanism design

The switch of the state machine in the client is not an instant act, but requires a transitional stage for processing the server information and to save resources. In some large 3D games, a large number of states will be waiting to be switched, which requires the design of a buffer mechanism on the client's side of the game to wait for the handoff.

4.3.1 Server Information

In the process of game design, the server is the core logic of the 3D game, while the game client belongs to the display logic part. And the ideal state is when the state of the client and the instructions issued by the server are always synchronized. However, due to the constraints of the speed of the network transmission and the computer’s own system operating efficiency, the core part of the game logic and display logic can not always stay synchronized. Therefore, in the process of running the game, the two will keep their own running states. Usually in the game design, the server sends instruction information to the client in two ways. One is the unconditional execution of the client's immediate request. The second is to add instruction information to the cache sequence of the handoff state, waiting to execute the previous state switching instructions and then execute the new command. At the same time, the instruction information sent by the server belongs to an irregular data packet, so that a certain amount can be cached in the client and be processed sequentially according to the order of instructions.

4.3.2 Resource data cache

Since many of the hard disks read slowly, it results in slower loadings of resources, especially larger resources. This requires asynchronous reading in game design. The main design process is as follows:

There are main logic and side logic in the game logic reading. When reading the target data resources, the main logic will not stagnate due to the slow speed but remain the current operating state and to mark the target resource so that it can be read by the side logic. When the reading is completed, the main logic switches the state according to the status flag. Same with the processing server

instruction information, the resource reading also needs to be conducted in the appropriate order, and a sequence for resource reading is established in the cache mechanism.

5. Conclusion

All in all, the application of 3D technology in video games has become more and more extensive. 3D games have become the current mainstream of game development, hence 3D game design should be based on changes in the times and the needs of gamers. Innovation is the key to the R&D process, where data rendering, scene design, motion modules and game collision detection can contribute to build the structure of the game engine and 3D model, and to realize the designs of program window, state machine and cache mechanism in the game client, which consist of the basic framework of 3D games. Although 3D games have been widely sought after by the players, and its technology has caused a stir among game designers, it also requires higher computer configuration at the same time. However, the author still believes that with the continuous development of 3D technology, 3D games will be equipped with higher picture quality and better simulation effect so that players can get better gaming experience.

References

[1] Yu X. 3D virtual imaging technology based on the game design and implementation [D]. Central China Normal University, 2013.
[2] Gao N.J. Game Design and Realization Based on 3D Virtual Imaging Technology [J]. Fine Arts Education Research, 2014,19: 78.
[3] Cao Y., Guo L.P., Du H.Y, etc. Cartoon style 3D game scene design production technology [J]. Software, 2015,03: 22-25.
[4] Zhang S., Zhou H.J., Zhang L.T. Design and Realization of Key Technology for Body - building Design of Body Piece Legacy Game Based on Unity3D and Kinect [N]. Journal of Sanming University, 2015,06: 32-36.