Design and Implementation of Human Computer Interaction in 3D Virtual Campus Based on Untiy3D

Li Wang
Department of education science and technology
Shanxi Datong University
Datong, China
e-mail: wangli523971@126.com

Abstract—The 3D virtual campus is the foundation of campus digitalization. The three-dimensional virtual campus can not only simulate the real campus scene, but also has a strong human-computer interaction function, can be used to control the virtual characters through the mouse and keyboard to roam the campus. This paper takes the virtual campus of Shanxi Datong University as the object, introduces the system development tools and the development process, and analyzes the realization method of the human computer interaction in virtual roaming.

Keywords-Untiy3D; virtual campus; human computer interactio; roam; C#

I. INTRODUCTION

Virtual campus roaming system is a virtual campus environment, which is based on realistic vision, hearing and touch. It is a real or virtual simulation of virtual space, the user with some equipment in virtual environment for virtual roaming, can be from any point of view to observe the virtual environment, thus resulting in the feeling of immersive [1]. Through the virtual campus, users can close to the campus construction, familiar with the campus environment, in-depth understanding of campus culture. Virtual campus can enhance the popularity of the school, promote the development of the school, it is also the basis for the realization of digital campus, can be used as an auxiliary tool for school planning.

The paper designs the virtual campus roaming system of Shanxi Datong University is a three-dimensional virtual campus system which has the visual, auditory, and has a strong interaction. This paper systematically analyzes the function realization of human-computer interaction in virtual ramble; the user can use the mouse and keyboard to choose to visit the building, can use the navigation map to locate the position, and can realize the scene change by clicking on the button. The user can walk and enjoy the campus scenery, but also into the room to understand the teaching environment, music commentary added, can ease the user's the dull mood of travel, so that users have a feeling of immersive.

II. RELATED TECHNOLOGIES TO ACHIEVE INTERACTION

The article uses Unity3D as the development platform and C# as the programming language to design the human computer interaction. Unity3D supports C# and JavaScript two programming languages, in which C# is more in line with the Unity3D’s programming habits. C# is an object-oriented programming language with simplicity and security. It has the advantages of Visual Basic, C++, Delphi, and Java and so on [2]. Unity3D is a standard commercial engine, its powerful; one of the notable features is cross platform development. Unity3D can run in the OS Mac or Windows operating system, in which the two operating systems, in addition to some differences between the client and the interface, the engine itself has no difference, for all types of developers to provide a convenient[3].

III. OVERALL DESIGN OF THE VIRTUAL CAMPUS

The development of the system uses the idea of software engineering, first of all, the campus geographic data is collected, and the camera is used to capture the scene; And then uses the accurate data of the campus, using AutoCAD to draw the map of the campus, so as to determine the space position of virtual campus; then uses 3ds Max to model, map and add lights on the campus, then the generated scene files are translated into Unity3D compatible format ".fbx"; And then in the Unity3D to carry out the campus roaming system design, including interface production, roaming interactive design and voice. Roaming interaction is the key point of this system, and it is also the focus of the article. Complete the above content, finally to optimize the system, packaged and released into executable file.

IV. DESIGN AND IMPLEMENTATION OF VIRTUAL CAMPUS HUMAN COMPUTER INTERACTION

Roaming interaction is the key to the design and implementation of virtual campus. Through human-computer interaction can be achieved through the mouse and keyboard to control the virtual human walking, make users choose to visit the building, can use the navigation map to locate the position, and can realize the scene change by clicking on the button, so as to achieve the purpose of autonomous roaming.

The objects in Untiy3D must bind the script to realize the logical judgment, complete the corresponding function. A script is equivalent to a component. The Inspector area of the object can be dragged only when the object is bound to the script. The programming language used in this article is C#. Specific design process is shown in figure 1.

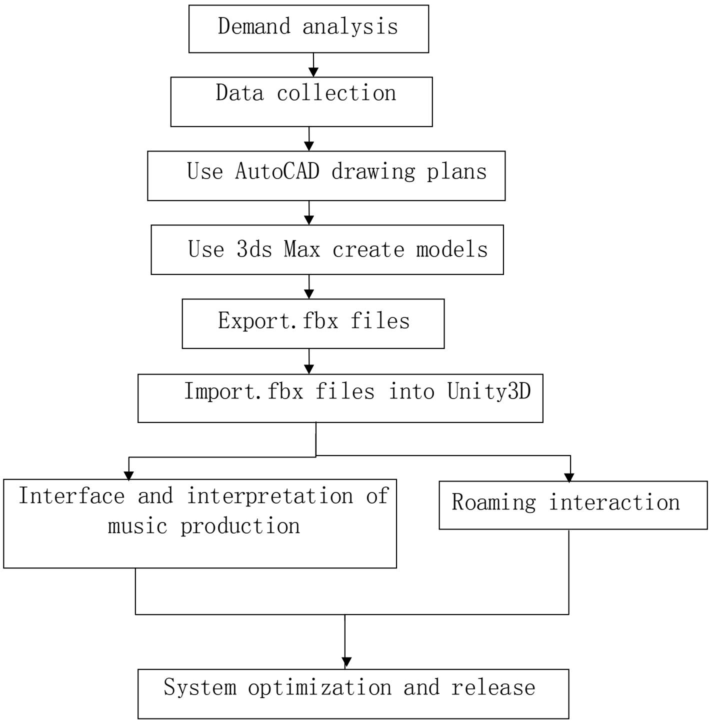
Demand analysis
Data collection
Use AutoCAD drawing plans
Use 3ds Max create models
Export.fbx files
Import.fbx files into Unity3D
Interface and interpretation of music production
Roaming interaction
System optimization and release

Figure 1. virtual campus design flow chart

A. Map navigation

The definition of map's width and height are 128 pixels，This map is a two-dimensional planar scaling map of the 3D scene. When visitors are in a strange three-dimensional scene, you can use it for navigation and positioning. This small map can show the location of the visitor in real time, it contains all the map of the scene; it will bring great convenience for visitors. The map navigation of this system is realized by NGUI. Steps are as follows:

Select the top view of the scene in the 3D view in unity, and then the interception of top view as the screenshot, the screenshot as small map template, zoom the screenshot, named map. To ensure that the mask and background image of equal size, its middle is a white circle, and the other parts of the alpha are transparent. The last is the small marks of roles.

1) Render Texture.

1 First create a “UI” under “MainCamera”, select “NGUI->Open the UI Wizard”, save these settings, click “Create Your UI”. Then the “Panel” under “Anchor” is changed to "MiniMapRenderPanel".

2 Create an “Atlas”, select “NGUI->Open the Atlas Maker”, to modify the name of “Atlas” on the new pop- up interface, click on the full map, then click “Create”.

3 Under “MiniMapRenderPanel” to build a “sprite”, select “NGUI->Open the Widget Wizard”, select the newly built Atlas, Select “Sprite” in “Template”, select “map” in “Sprite”, and other options are set to default.

4 Create a new “Textur Render” in the resource, select “Assets->Create->Render Texture”, named “MiniMapRenderTexture”.

5 Set the camera. Production materialCreate a Shader, select “Assets->Create->Shader”. The code is as follows: Shader "Transparent/Mask" // the shader’s path and name{ Properties // Sets up the rendering properties { _MainTex ("Base (RGB)", 2D) = "white" {} _Mask ("Culling Mask", 2D) = "white" {} _Cutoff ("Alpha cutoff", Range (0,1)) = 0.1 } SubShader { Tags {"Queue"="Transparent"} Lighting Off ZWrite Off Blend Off AlphaTest GEqual [_Cutoff] Pass { SetTexture [_Mask] {combine texture} SetTexture [_MainTex] {combine texture, previous} } }}Create a “material”, select “Assets->Create->Material”，then select “Shader”. At this time the interface will appear the Base（RGB）and the Culling Mask，Drag the Render Texture to the Base（RGB），Drag the white circle to the Culling Mask. Use NGUI to create a map at the top right corner of the interface. Create a “Panel” below the top right corner of the “Anchor”, named “MiniMapPanel”, Then add two “Sprite” and one “Texture”, One of the “Sprite” is a round box, and the other “Sprite” is a small arrow， “Texture” used to receive the material display graphs.

Create a “MiniMap” script code as follows: using UnityEngine； using System.Collections； public class MiniMap MonoBehaviour { // Use this for initialization

public GameObject point； // Player logo little arrow public GameObject map； // Map panel

private GameObject hero； // Control figures

//public GameObject terrain； private float miniMapScaleRatio； // Map and real terrain size ratio

void Start () { map.transform.localScale = new Vector3(Screen.heigh;Screen.height;1) hero = GameObject.Find("/Droid Girl u3")； GameObject terrain = GameObject.Find("DiMian")；

Terrain script = terrain.GetComponent<Terrain>()；

miniMapScaleRatio=float)map.transform.localScale.x/ script.terrainData.size.x； } // Update is called once per frame void Update () { if (hero && point && map) { point.transform.rotation=Quaternion.Euler（0；0；-hero.transform.rotation.eulerAngles.y)；

tor3() { x = -hero.transform.position.x * miniMapScaleRatio； y = -hero.transform.position.z * miniMapScaleRatio； z = 0； } }The autonomous roaming map is shown in Fig .2.

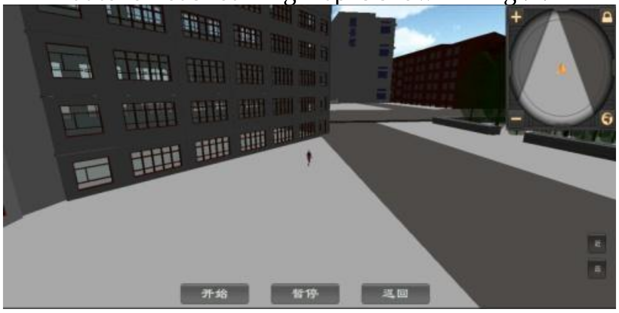

Figure 2. Autonomous roaming map

B. Collision detection

Collision Detection is also known as interference detection or contact detection, it is a kind of technology that can check the collision or interference that may occur during the movement of an object. The main purpose is to avoid collision or interfere with the occurrence of the original motion event. Collision detection is an essential technology to realize the virtual campus roaming. In order to guarantee real-time performance, the most commonly used detection method is AABB (Axially Aligned Bounding Box). The specific method is to use a cube or a sphere wrapped 3D object, and then according to the packing box distance and location information to calculate whether the collision occurred. The collider is divided into three categories: Static Collider, Rigid body Collider, Kinematic Rigid body Collider.

In Untiy3D, the role controller can detect the collision between the object. When using it, you need to call the parent class method OnControllerColliderHit() for detection. In the parent class, using the hit.gameObject reference, you can obtain the game object after the role of the controller component of the crash. Code for collision detection is as follows: using UnityEngine； using System.Collections； public class Test：MonoBehaviour

// Object name for active collisionpublic String castName = null； // the name of the object receiving the collisionpublic String receiveName = null； void Start(){ } void Update(){ } void OnGUI()

{ if(castName!=null&& receiveName !=null) {GUI.color = Color.black； // Set the color to black// Display names of active collision object and receiving collision objectGUI.Label(new Rect(100,100,200,30),"object name for active collision" + castName)； GUI.Label(new Rect(100,200,200,30),"the name of the object receiving the collision" + receiveName) ； }if(receiveName== "Cube1"){Application.LoadLevel("RenWuXingZou")； } }void OnControllerColliderHit(hit:ControllerColliderHit)

{public GameObject hitObject； hitObject = hit.collider.gameObject ；

if(!hitObject.name.Equals("Terrain")) {

castName = gameObject.name；

receiveName = hitObject.name ； }}Use the keyboard and mouse to interact In the process of autonomous roaming, the interaction between the mouse and the keyboard is a must. Rely on the mouse and keyboard to achieve the camera angle adjustment, the front, rear, left and right of the four directions of change. The realization of these functions provides a guarantee for the autonomous roaming.

In the autonomous roaming mode, the virtual human "forward" and "backward" are controlled by the "W" and "S" key, and the "A" and "D" key control the virtual human to rotate and rotate to the left. On this basis, user clicks the right mouse button to achieve the "circle of view"; user slides roller can make the visual angle closer or push away. Keyboard interaction code is as follows: void Update(){// forward

if(Input.GetKey(KeyCode.W)){ n.Translate(0,0,50 * Time.deltaTime,Space.Self)； } // backward

if(Input.GetKey(KeyCode.S)){ n.Translate(0,0,-50 * Time.deltaTime,Space.Self)； } // turn left

if(Input.GetKey(KeyCode.A)){ n.Rotate(0,-220 * Time.deltaTime,0,Space.Self)； } // turn right

if(Input.GetKey(KeyCode.D)) {n.Rotate(0,220* Time.deltaTime,0,Space.Self) ； }}System optimization and its publicationIn the system optimization, adjusts Clipping Planes of the Main Camera range small, so that the visual field of vision will be reduced, the processor pressure will be effectively reduced.

The establishment of a virtual campus system is to be placed on the Internet to allow users to experience, so it is necessary to publish the system. The specific process is as follows: In the Windows system, “File->Build Setting” is required to click on the menu bar, after clicking on it will appear a scene editing list, Click on the various scenes in the project panel and drag them to the editor's list, the figures behind scenes show the order of rendering. "0" is the first scene to be rendered. Click on “Build” system will be released.

V. CONCLUSIONS

Interactive design is the key to realize virtual roaming, and also the advantage of virtual reality technology. It can bring immersive experience for the user, and enhance the interaction of the system. The interactive technology of virtual roaming mainly uses the mouse, keyboard to interact, detect the collision, navigation map, etc... In this paper, the interaction between the mouse and the keyboard and the detection of the collision is analyzed in detail, which provides a reference for the research and practice of the virtual roaming technology.

Acknowledgment

This article belongs to Shanxi Province Education Science “ Twelfth Five Year Plan” --“ Research on construction technology of 3D virtual campus roaming simulation system based on Unity3D” (item number: GH-13092) stage research results, and is supported by the Shanxi Datong University youth project--“Design and implementation of virtual campus roaming system based on Virtools in Shanxi Datong University” (item number: 2013Q1).

REFERENCES

[1] X.C.Shen, T.Y.Xie, L.D.Huang, Design and implementation of virtual campus roaming system based on 3DSMax&3DVRI, J. Modern educational technology. 2012(10).

[2] Beida Jade Bird. C# and its characteristics, http://www.gdglc.com/training/29.html,2012.

[3] YuSong Xuan. Unity 3D game development. Beijing: People's Posts and Telecommunications Press, 2012:3.

[4] Q.Y.Lou, X.J.Cheng, K.Tan, Building 3D modeling based on AutoCAD and 3ds Max , J. Engineering Investigation. 2013(11)1-1.

[5] Y .Fan, M. Yang and W.C. Sun. 3ds Max three dimensional animation production base and operating instructions. Beijing: Tsinghua University Press. 2010.

[6] Li Wang, Researching of the three-dimensional virtual simulation campus scene’s construction technology, J. The Open Cybernetics & Systemic Journal. 2015(Volume 9)1056-1057.

[7] Li Wang, Construction of the three-dimensional virtual campus scene’s Problems and Solutions, J.The Open Cybernetics & Systemic Journal. 2015(Volume 9)1132-1133.

[8] Y.F. Ma,Y.T. Xue and H.S. Nan.Virtual reality technology and its application[M]. Beijing: China Railway Press, 2011: 2-2.

[9] Information on http://wenku.baidu.com.

[10] Information on http://baike.baidu.com.

[11] X.Y. Huang, M. Du, F.F. Dong and M.H. Guo. 3ds Max 2011 standard tutorial. Beijing: Tsinghua University Press. 2011.

[12] Q .Wang, Y.Q. Zhao, The application of educational games in primary school mathematics teaching, J. Software Guide. 20013.12(3):3-8.

[13] Y.S. Xuan, Unity3D game development, Beijing: People's Posts and Telecommunications Press, 2012:91-13.