

硕 士 研 究 生 读 书 报 告



题目 纹理映射技术在三维游戏引擎中的应用

作者姓名 刘杰

作者学号 21651001

指导教师 李启雷

学科专业 软件工程

所在学院 软件学院

提交日期 二○一七年四月

Texture Mapping and Its Application in 3D Game Engine

A Dissertation Submitted to

Zhejiang University

in partial fulfillment of the requirements for

the degree of

Master of Engineering

Major Subject: Software Engineering

Advisor: Qilei Li

By

Jie Liu

Zhejiang University, P.R. China

2017

摘要

游戏引擎是游戏的心脏 ,它在后台控制着游戏中各个模块同时有序的工作 ,渲染模块是游戏引擎中最复杂 ,也是最能 体现游戏效果的模块。纹理映射技术的应用能增强场景绘制的真实感并能提高场景的渲染速度 。该文是对三维游戏引擎中的渲染模块的初步研究 ,首先介绍了三维游戏引擎系统以及纹理映射技术 ,使用 OpenGL 图形库实现了大环境天空盒 、三维地形的贴图效果、并加入了地面建筑物 ,再现了真实的三维场景。介绍了纹理映射技术在三维游戏引擎中的应用及作用。

**关键词**：移动互联网，终端技术，能源控制技术，混合开发模式

Abstract

The game engine is the heart of the game, which controls all the other models in the game to work orderly at one time. The rending model is the most complex and displays the quality of the game. In the rending, the application of texture mapping can strengthen the reality and increase the rendering speed of the scene. This paper is the elementary study of rending model in 3D game engine, first we introduce the system of 3D game engine and texture mapping, then we realize the texture mapping effect of environment-sky and the 3D terrain using OpenGL, and add the construction, reappearance the reality of the 3D scene. The paper also introduces the application and effect of texture mapping in 3D game engine.

**Keywords：**3D Game Engine, Texture mapping, Rending model

1三维游戏引擎

2003年，国家科技部首次将网络游戏技术纳入国家863科技发展计划，这标志着国家对年增长率近3倍、带动上下游经济增长超过10倍的网络游戏产业态度已经明朗化。在网络游戏研发、运营、管理等诸方面，与日韩、欧美等国相比，我国还存在较大差距。三维游戏中的关键就是游戏引擎的研发。研究三维游戏引擎中的关键技术成为当前的重点。

游戏引擎是游戏的心脏，它把游戏中的所有元素捆绑在一起，在后台指挥它们同时、有序地工作。简单地说，引擎就 是“用于控制所有游戏功能的主程序 ,从计算碰撞、物理系统 和物体的相对位置，到接受玩家的输入 ,以及按照正确的音量输出声音等等”。现在，三维游戏引擎已经发展为一套由多个子系统共同构成的复杂系统，从建模、动画到光影、粒子特效，从物理系统、碰撞检测到文件管理 、网络特性 ,还有专业的编辑工具和 插件，几乎涵盖了开发过程中的所有重要环节。比较优秀的 游戏引擎有 Quake 引擎、Unreal Tournament 引擎等。使用 Quake 引擎的游戏有 Counter Strike , Team Fortress ,等 ,而 Tac Ops 和 Strike Force 使用了 Unreal Tournament 引擎。引擎使得制作游戏快速简单起来 。

渲染是引擎最重要的功能之一，当 3D 模型制作完毕之 后 ,美工会按照不同的面把材质贴图赋予模型 ,这相当于为 骨骼蒙上皮肤 ,最后再通过渲染引擎把模型、动画、光影、特效等所有效果实时计算出来并展示在屏幕上。渲染引擎在引擎的所有部件当中是最复杂的，它的强大与否直接决定着最终的输出质量。通常，我们用 Open GL，Direct3D，Glide 或者软件成像来将图像显示在屏幕上。OpenGL提供了比较完整的纹理映射功能，使得再现真实感三维场景中的贴图效果变得容易起来。

2纹理映射技术

纹理映射技术是近几年来发展最快的技术之一，广泛应用于三维真实感图形的生成与显示中。运用纹理映射可以 方便地制作真实感图形而不花更多的时间去考虑物体的表面细节。

纹理映射技术最早是由Catmull 在1947年率先提出的，Catmull首先找到了以（u,v）表示的双变量实数空间(纹理空间)和以参数(s,t)表示的三维曲面之间的对应关系(映射关系)。为了计算三维曲面上每一点的彩色值，Catmull使用了一个二维矩形数组，该数组的位盆表示（u，v）的参数值,数组的值表示(u，v)点对应的彩色值；接着，应用上面找到的映射函数将每一点（u，v）及其对应的彩色值映射到相应的三维曲面上，代替了只用一种色彩表示曲面的情况，在三维曲面上得到了彩色图案。但是在Catmull的算法中，被映射的值仅限制为纹理图案的彩色值。为此，Blinn和Newell在1976年提出了“反射映射”技术, 他们在光照模型中考虑了纹理图案的镜面反射系数和高光值，从而大大改进了经纹理映射后的图形质量。在1978年，Blinn又提出了另一种纹理映射技术一一凹凸映射,利用一个扰动函数扰动物体的表面法矢来模拟有随机法矢的粗糙表面纹理，实现了在光滑的物体表面附上粗糙纹理的方法。

Gardner使用纹理函数控制物休的透明性,在模拟自然界景物方面取得了良好的效果。 Kajiya扩展了Blinn的方法,利用扰动函数扰动物体表面的其它参数，很好地模拟了头发、衣服之类的物体。

与此同时，国外大量的学者通过实验手段在多边形表面生成了许多逼真的纹理图案。但不幸的是，这些方法非常低效，只适用一些简单的场景。我们知道，要将一幅纹理图案映射到一张曲面上，需对纹理图案进行拉伸或压缩以适应曲面的形状。一般来说，随着纹理图案的压缩，增加了纹理图案的高频成分，在图象空间中容易引起图形混淆，这主要是由于欠采样引起的。因为图形显示器是由许多图象元素组成的,一个象素并不是理想中的无穷小量，而占据一定的区域，对纹理图案的采样频率取决于图形显示器的分辨率（即相邻两象素中心的间距），因此，对某个图形显示器来说，采用通常的显示方法，其采样频率是恒定的，随着纹理图案的压缩，增加了纹理图案的高频成分，容易产生图形混淆。

与此同时，国外大量的学者通过实验手段在多边形表面生成了许多逼真的纹理图案。但不幸的是，这些方法非常低效，只适用一些简单的场景。我们知道，要将一幅纹理图案映射到一张曲面上，需对纹理图案进行拉伸或压缩以适应曲面的形状。一般来说，随着纹理图案的压缩，增加了纹理图案的高频成分，在图象空间中容易引起图形混淆，这主要是由于欠采样引起的。因为图形显示器是由许多图象元素组成的,一个象素并不是理想中的无穷小量，而占据一定的区域，对纹理图案的采样频率取决于图形显示器的分辨率（即相邻两象素中心的间距），因此，对某个图形显示器来说，采用通常的显示方法，其采样频率是恒定的，随着纹理图案的压缩，增加了纹理图案的高频成分，容易产生图形混淆。 要从本质上了解图形混淆的实质，我价应该从数字信号处理理论上分析这个问题。在信号处理中，我们首先对原始的连续时间信号进行采样，得到一个等间隔的离散信号，然后，用该信号重建原始信号。为了避免混淆以使采样处理后仍有可能准确地恢复其原信号，采样频率必须大于最高频率的两倍，这就是众所周知的采样定理。这个理论用来处理图象时，把图象作为一个连续信号，对该信号的采样是在两个相邻的图象元素的间隔内进行的。根据采样定理知：在不产生混淆的情况下，被采样图象的最高频率应小于在一个采样周期内采样两个象素的采样频率。 在纹理映射过程中，混淆常发生在纹理图案压缩的情况下，为了避免图形混淆，人们已提出了许多反混淆方法。基本反混淆方法有两类：一类是过采样，这可以通过提高光栅的分辨率来实现。但不幸的是CRT光栅扫描设备目前实际能够达到的最高分辨率为每条扫描线约可显示2000个象素；另外，提高光栅分辨率也往往伴随着增加图形显示器的费用。由于这一限制使人们想到在较高分辨率上对光栅进行计算，然后采用某种求平均算法或加权平均算法得到较低分辨率的象素属性，显示在分辨率较低的显示器上。另一类反混淆方法就是采用滤波器，该方法把象素看成一个有限区域，而不是作为一个点来处理，使用某种类型的滤波器通过前重滤波和后置滤波来实现。滤波的主要用途就是滤掉被采样信号中的高频成分，从而达到反混淆的目的。

纹理映射的本质是对三维物体进行二维参数化，即先求得三维物体表面上任一点的二维 ( u , v) 参数值 ,进而得到该 点的纹理值，最终生成三维图形表面上的纹理图案。在光滑 曲面上添加纹理图案的核心问题是映射，因此纹理问题可以 简化为从一个坐标系到另一个坐标系的变换。其中至少涉及两个映射，如图 1 所示 。一个是从纹理空间到景物空间，有时也称为曲面参数化;第二个映射是从景物空间到图像(屏幕) 空间，即取景变换。通常，这两个变换被合成为一个变换。

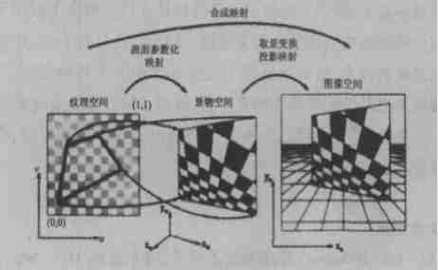


图 **1** 从纹理空间到图像空间的映射

如果纹理图案定义在纹理空间中一个正交坐标系 *( u , v)* 中，曲面定义在景物空间的正交坐标系*( x , y , z)*中，它在参数空间空间 的表示为那么在曲面上添加纹理将涉及在两曲面之间确定或指定一个映射函数。例如*，*从纹理空间到参数空间的映射为 *:*



从参数空间到纹理空间的逆映射为：



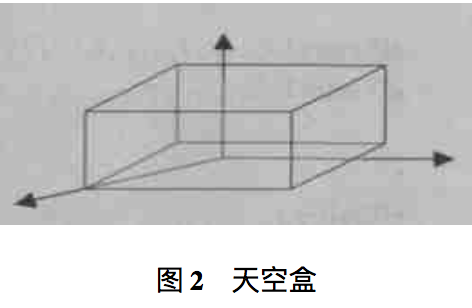
OpenGL 提供两类纹理 ,其中一类纹理图像的大小必须是几何级数的，另一类Mipmaps纹理可以为任意大小。

3纹理技术的应用

下面以一个室外游戏场景的真实感绘制为例讲述如何在绘制三维场景时使用纹理映射技术。

###### 3.1大环境——天空的绘制

室外三维游戏中，首先要构建大环境——天空。在一些三维建模软件如3D MAX或者Maya中，通常使用天空盒来表现天空景象，如晴朗的白天、灯光绚烂的夜晚等。天空盒的模型可以使用一个半球体，通过对半球体的内表面进行纹理贴图来表现天空。但这种模型在进行贴图时容易出现球体两极处的走样为了避免这种情况 ,我们使用立方体 (去掉底面) 模型来模拟天空天空盒的模型如图 2 所示 ,



晴朗天空的图片如图3 所示，当把映射函数取为一个线性函数 ,即取式 (1) 中的函数关系为 :θ= Au+B,<=Cv+D时,设定A=C=1,B=D=0,此时纹理空 间的(0,0)对应景物空间的(0,0) ,纹理空间的(1,1)对应景物 空间的 (1，1)，利用 OpenGL 的几何变换功能，将天空图片映 射到天空盒的内侧面，可以轻松地实现天空盒的贴图效果，效果图可以参见图 4。值得注意的是，这种方法必须使用来贴图的图片是连续的。观察图 3 中的前四幅图片，可以发现它们是连续的。

###### 3.2 真实感地形的绘制

真实感地形的绘制分为地形模型映射以及真实感地形生成两部分。地形模型映射就是把数字形式的地形数据转换成 OpenGL 的基本格式 ,并计算相关的参数 (顶点和法向量等) 的过程 。目前地形模型的数字表达方式主要有不规则三角 网 ( T I N ) 、规 则 格 网 ( D E M ) 和 等 值 线 ( V E C ) 3 种 , 但 这 3 种 格 式 的地形模型数据不能在 OpenGL 上直接使用 ,必须把数字地 形模型转成 OpenGL 原语 ,以生成 OpenGL 可识别的原语序 列 。OpenGL 不仅提供了点 、线 、多边形等建模原语 ,而且通 过这些原语还可把地形模型数据表示成典型原语序列 、线型 原语序列及多边形原语序列 。由于 OpenGL 要求多边形必须 是凸多边形 ,因此在软件实现中采用了三角面原语序列表达 地形模型 。

不同的数字地形模型表示形式 ,其三角面序列转换方式 是有差别的 。我们使用了 DEM 规则网格 ,所以下面主要介 绍 D E M 的 转 换 。 规 则 格 网 ( D E M ) 需 要 转 换 成 O p e n GL 的 三 角 面串序列 :

glBegin(GL-TRIANGLES-STRIP) ;

…glVertex(xi ,j ,yi ,j ,zi ,j) ; / / 第 i 行 j 列点

glVertex(xi+1,j ,yi+1,j ,zi+1,j) ; //第i+1行j列点

glVertex(xi,j+1 ,yi,j+1 ,zi,j+1) ; //第i行j+1列点

glVertex(xi+1,j+1,yi+1,j+1,zi+1,j+1); //第i+1行j+1列点

…

glEnd() ;

由于OpenGL不直接计算每个顶点的法向量,因此若需 获得较佳的视觉效果或模拟特殊的地景 ,在 DEM 转换成 OpenGL的三角面串序列时,就需为每一个顶点计算出它的 法向量 ,且所有顶点的法向量应保持一致 ,否则有的三角面 将不可见。顶点的法向量与顶点坐标的计算与设置步骤 如下:

glBegin(GL-TRIANGLES) ;

…

C a l c N o r m a l ( n x , n y , n z ) ; / / 计 算 顶 点 的 法 向 量

glNormal3f (nx ,ny ,nz) ; glVertex(x ,y ,z) ; / /设置当前顶点的法向量

glVertex(x ,y ,z) ; / / 顶点的坐标

…

glEnd() ;

我们取每个顶点的法矢量为围绕该顶点所有三角面的 法向量的平均值 ,这样可以实现地形的平滑效果 。

真实感地形是指通过对地形进行透视投影 、消隐等操作 后 ,生成像照片那样的黑白图像或彩色图像 ,产生出仿真效 果 。通过 OpenGL 的纹理映射技术将地貌图片与数字地形进 行融合 ,从而提高地形的表现力 。Mipmap 映射给出一系列成 比例的纹理图像 ,根据地形离视点的距离 ,分别赋予适当尺 寸的纹理 。使用

gluBuild2Dmipmaps(GL-TEXTURE-2D,3,w,h,GL-RGB,GL- UNSIGNED-BYTE,data) ;

函数可以生成二维纹理图像 。

若 O p e n GL 中 的 纹 理 坐 标 空 间 ( u , v ) 为 u : 0 ~ 1 . 0 ; v : 0 ~ 1 . 0 ,则地貌图片与数字地形融合的纹理映射过程 ,关键是计算 出各剖分点在纹理空间 (u ,v) 中的坐标 。计算数字地形剖分 点纹理坐标的过程如下 :

glBegin(GL-TRIANGLES) ;

…

CalcTexCoord(tx ,ty) ; / / 计算顶点的纹理坐标

GlTexCoord2f (tx ,ty) ; / / 设置当前顶点的纹理坐标

glVertex(x ,y ,z) ; / / 顶点的坐标

…

glEnd() ;



###### 3.3 地面建筑物

地面建筑物的绘制通常是先在 3dmax、MAYA 等三维建模软件中建立模型 ，包含顶点信息、纹理信息、材质信息。通过将模型文件转化为OpenGL能处理的格式导入到三维场景中再现建筑物。建筑物表面的纹理细节，如墙面、窗户、门、瓦片等的细节我们都可以用纹理贴图的形式来表现，使用贴图可以将建筑物 的细节表现的栩栩如生，活灵活现。效果如图 4 所示 。

4 小结

三维游戏引擎是游戏的心脏 ,它在后台控制着游戏各个模块同时有序的工作。其中渲染模块负责将场景显示在计算机屏幕上，所以三维游戏引擎的好坏在很大程度上取决于渲染模块的渲染质量。本文使用三维图形接口 OpenGL中的 纹理映射技术，逼真地再现了三维场景中的大环境天空、三维地形以及地面建筑物的真实感，取得了较好的渲染效果。

参考文献

［1］LukeHodorowicz著,龚敏敏译.游戏引擎的元素[DB].http://

www.chinagamedev.net/ newcgd/showarticle.php?id=194,2003.

[ 2 ] Lucifer著.游戏引擎系统[DB]. http://www.chinagamedev.net/

cgd/ develop/ 3D/ 200202/ EngineSys. htm , 2003．

［3］薛安,马蔼乃,李天宏. 基于OpenGL实现真实感地形表现的研

究[J ] . 中国图象图形学报 ,2001 ,6 (A) (8) :800 - 805.

［4］彭群生等.真实感图形的计算机生成[J].计算机学报,1989,(3):226-237.

［5］范波,吴慧中.多面体表面纹理映射方法的研究[J].计算机研究与发展,1999,36(4):446-450.

［6］DavidFRogers(美)著,石教英等译.计算机图形学的算法基础(第二版)[M].机械工业出版社,2002.

［7］Staadt O，Gross M，Gatti R． Fast multiresolution surface meshing［A］． In: Proceedings of IEEE Visualization’95 Conference［C］，Held in AtlantaGeorgia，October 1995．