基于 OpenGL 的等角插补明暗处理的软件实现

张 华
ZHANG Hua

中国工程物理研究院 计算机应用研究所,四川 绵阳 621900
Insitute of Computer Applications,China Academy of Engineering Physics,Mianyang,Sichuan 621900,China
E-mail:hzhang@caep.ac.cn

ZHANG Hua.OpenGL based software implementation of equal angle interpolation shading.Computer Engineering and Applications,2008,44(23):86-87.

Abstract: Equal angle interpolation shading not only has the same visual effect as that of Phong shading,but also can get more rendering efficiency than that of Phong shading.In this paper,based on the standard graphics rendering pipeline,hardware computing of OpenGL rendering pipeline,and equal angle interpolation shading are implemented by using C language.In the final,experiments are presented.This implementation can be referenced in the research and designing of rendering algorithm.
Key words: OpenGL;equal angle interpolation;shading

摘 要:等角插补明暗处理具有 Phong 明暗处理的视觉效果,而且具有较 Phong 明暗处理更快的渲染效率,基于标准图形渲染管道,用 C 语言实现了 OpenGL 渲染管道的硬件计算部分,同时用 C 语言实现了等角插补明暗处理,并给出了实验结果。该方法的实现可为渲染算法的研究和实现提供参考。
关键词:OpenGL;等角插补;明暗处理
DOI:10.3778/j.issn.1002-8331.2008.23.027 **文章编号**:1002-8331(2008)23-0086-02 **文献标识码**:A **中图分类号**:TP391.41

1 引言

等角插补明暗处理于 1989 年由 Kuijk 和 Blake 提出以来[1],由于其与 Phong Shading 有相同的视觉效果,且具有渲染效率高的特点,受到了不少的学者的关注。

但是在实现等角插补明暗处理的过程中,需要详细了解计算机图形学标准图形渲染管道和底层图形接口软件。本文采用 C 语言和 OpenGL 的方式来实现整个标准图形渲染管道,同时,用 C 语言来实现等角插补明暗处理算法。

等角插补明暗处理是采用等量插补两向量之间的角度,得到各插补点向量,然后根据相应的光照模型和光源,计算出各插补点的光照颜色值。

如图 1 所示, 向量 ***OA*** 和向量 ***OB*** 分别是线段 AB 的法向量,其夹角为 θ,点 I 是线段 AB 中的一插补点,向量 ***OC*** 是点 I 的插补法向量,角度 $i\theta$ 是点 I 的插补角度,计算向量 ***OC*** 的原理如下[2]。

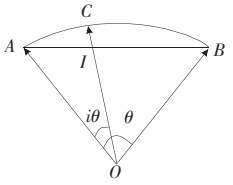

图 1 等角插补

(1)计算点 A 和点 B 之间像素个数;
(2)计算线段 AB 与扫描转换线的交点;
(3)由(2)中交点坐标,逐一取插补值 i;
(4)计算插补角度 $i\theta$;
(5)给定一光照模型和光源;
(6)由等角插补明暗处理公式,计算出插补点的光强颜色值;
(7)以此类推,计算出线段 AB 上的各插补点的光强颜色值;
(8)插补运算结束。

2 渲染管道

OpenGL 是通过设置各变换矩阵, 由硬件来完成各矩阵的运算,本文采用手动设置各变换矩阵,然后由软件来逐步进行各矩阵的变换运算。

标准图形渲染管道的软件实现步骤如下:(1)设置模型变换矩阵,依次为视点变换矩阵、透视投影变换矩阵,透视除法变换矩阵和视口变换;(2)用标准 C 编写各变换的运算代码。其中,前一级的变换结果是下一级变换的输入数据,第一级变换的输入数据是顶点的齐次坐标。

3 软件实现

首先通过顶点与各变换矩阵的运算,逐级计算出变换结果

基金项目:中国工程物理研究院科学技术基金(Science & Technology Funds of China Academy of Engineering Physics under Grant No.20060653)。
作者简介:张华(1971-),男,高工,博士,主要研究方向:计算机图形学的基本渲染算法及其在计算可视化、虚拟现实和网络渲染中的应用。
收稿日期:2007-10-18 **修回日期**:2007-12-06

坐标,进而完成渲染管道中所有的变换运算。然后在伪像素级利用等角插补公式计算等角插补明暗处理。

3.1　渲染管道的计算

由于渲染管道的各级运算类似，故本文举透视投影变换为例,设前一级变换的结果坐标为 $A(Xv,Yv,Zv,1)$,根据文献[3],透视变换投影矩阵为:

$$\boldsymbol{M}=\begin{pmatrix} \frac{2N}{R-L} & 0 & \frac{R+L}{R-L} & 0 \\ 0 & \frac{2N}{T-B} & \frac{T+B}{T-B} & 0 \\ 0 & 0 & -\frac{F+N}{F-N} & \frac{-2FN}{F-N} \\ 0 & 0 & -1 & 0 \end{pmatrix} \tag{1}$$

其中:F、N 分别是远近裁剪平面的坐标;T、B 分别是上下裁剪平卖的坐标,R、L 分别是右左裁剪平面的坐标。那么透视投影变换的变换运算为 MA:

$$\begin{pmatrix} \frac{2N}{R-L} & 0 & \frac{R+L}{R-L} & 0 \\ 0 & \frac{2N}{T-B} & \frac{T+B}{T-B} & 0 \\ 0 & 0 & -\frac{F+N}{F-N} & \frac{-2FN}{F-N} \\ 0 & 0 & -1 & 0 \end{pmatrix} \begin{pmatrix} Xv \\ Yv \\ Zv \\ 1 \end{pmatrix} = \begin{pmatrix} \frac{2NXv}{R-L}+\frac{(R+L)Zv}{R-L} \\ \frac{2NYv}{T-B}+\frac{(T+B)Zv}{T-B} \\ \frac{-(F+N)Zv}{F-N}+\frac{-2FN}{F-N} \\ -Zv \end{pmatrix} \tag{2}$$

其 C 代码如下所示。

```
void perspectiveProjection(double*matrix,double*point){
  /* 输入参数分别透视投影变换矩阵，前一级变换的结果坐标,输出参数存入 point*/
    int i=0;
    double temp[4]={ 0,0,0,1};
    //临时变量,暂存变换结果
    temp[0]=matrix[0]*point[0]+matrix[4]*point[1]+
            matrix[8]*point[2]+matrix[12]*point[3];
    temp[1]=matrix[1]*point[0]+matrix[5]*point[1]+
            matrix[9]*point[2]+matrix[13]*point[3];
    temp[2]=matrix[2]*point[0]+matrix[6]*point[1]+
            matrix[10]*point[2]+matrix[14]*point[3];
    temp[3]=matrix[3]*point[0]+matrix[7]*point[1]+
            matrix[11]*point[2]+matrix[15]*point[3];
    for(i=0;i<4;i++)
            point[i]=temp[i];//输出结果
}
```

3.2　等角插补明暗处理

给定一光照模型和光源,然后用扫描线和等角插补公式来计算等角插补明暗处理。

3.2.1　光照模型及光源

本文采用简单的 Lambert 光照模型[4],其模型示意图如图 2 所示,$\boldsymbol{L}$ 是光源方向,$\boldsymbol{V}$ 为法向量。

其光照计算公式为:

$$I=K_d\times(\boldsymbol{V}\cdot\boldsymbol{L}) \tag{3}$$

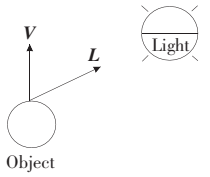

图 2　Lambert 光照模型

其中 K_d 为漫反射系数,为简化起见,本文假设 K_d 为常数 1,且设光源为平行光(如太阳光)。

3.2.2　计算

由第 1 章的方法,可以得到相应的插补角,那么据文献[5]的等角插补明暗处理公式,得到插补点的法向量:

$$\boldsymbol{OC}=\frac{\sin(\theta-i\theta)}{\sin\theta}\boldsymbol{OA}+\frac{\sin(i\theta)}{\sin\theta}\boldsymbol{OB} \tag{4}$$

因此 $\boldsymbol{I}=\boldsymbol{OC}\cdot\boldsymbol{L}$。

4　实验

由于三角形是 OpenGL 渲染的基本单元,本文以三角形的三个顶点 $A(0,0,0)$,$B(40,0,0)$和 $C(40,10,0)$为例,通过上述渲染管道和等角插补明暗处理的运算,得到如图 3 所示的实验结果。另外由 Phong Shading 得出的实验结果如图 4 所示,可以看出两图视觉效果相同。

图 3　等角插补明暗处理效果图

图 4　Phong 明暗处理效果图

5　结语

本文通过分析 OpenGL 渲染管道和等角插补明暗处理,用 C 语言实现了 OpenGL 渲染管道的硬件计算部分,同时用 C 语言实现了等角插补明暗处理,并给出了实验结果。本文的实现可为渲染算法的研究和实现提供参考。

参考文献:

[1] Kuijk A A M,Blake E H.Faster phong shading via angular interpolation[J].Computer Graphics Forum,1989,8(4):315-324.

[2] Barrera T,Hast A,Bengtsson E.Faster shading by equal angel interpolation[J].IEEE Trans on Visualization and Computer Graphics,2004,10(2):217-223.

[3] Foley J D,van Dam A,Feiner S K,et al.Computer graphics:principles and practice[M].2nd ed.MA:Addison-Wesley,1996.

[4] Blinn J F.Models of light reflection for computer synthesized pictures[C]//Proc SIGGRAPH,1977:192-198.

[5] Glassner A.Situation normal[J].IEEE Computer Graphics and Applications,1997,17(2):83-87.

(上接 76 页)

[2] 高兴宝.线性约束非线性规划的新神经网络[J].西安电子科技大学学报:自然科学版,2002,29(1):52-55.

[3] Kennedy M P,Chua L O.Neural networks for nonlinear programming[J].IEEE Trans on Circuits and Systems,1988,35(5):554-562.

[4] 张国平,王正欧,袁国林.一个新的线性规划通用神经网络[J].天津大学学报,2001,34(4):459-462.

[5] Perez M J.Convergence analysis of a discrete-time recurrent network to perform quadratic real optimization with bounded constraints[J].IEEE Trans Neural Networks,1998,9(6):1344-1351.

[6] 胡铁松,郭元裕.多目标动态规划的神经网络方法[J].电子学报,1999,10:70-73.