

文章编号: 1000-5013(2010)02-0170-04

面向实时虚拟游戏的 GPU 真实感草地绘制

陈柏生, 周谐, 陈锻生

(华侨大学 计算机科学与技术学院, 福建 泉州 362021)

摘要: 在满足虚拟游戏实时性要求的前提下, 绘制具有一定视觉真实感的草地, 以增强游戏场景的真实感. 将草地抽象成一块均匀、连续的透明体, 提出一种简化层状算法, 利用图形处理器(GPU) 支持功能, 计算穿过草丛的光的能量衰减, 高效地模拟草的自阴影效果, 并通过加入 Perlin 噪声进一步增加草地阴影杂乱交错的效果. 结果表明, 该方法能以极小的性能代价绘制草地阴影, 实现具有一定真实感的大面积草地的实时绘制.

关键词: 虚拟游戏; 绘制算法; 草地阴影; Perlin 噪声

中图分类号: TP 391. 9; TP 391. 41

文献标识码: A

随着现代个人计算机图形显示硬件的发展, 计算机游戏等实时图形交互应用的视觉真实感有了长足的进展. 由于图形运算资源日渐充足, 原来游戏中尽力避免出现的大面积植被, 现在也成为了游戏中的常见场景. 草地是自然场景的一个重要元素, 由于草具有丰富的细节, 并且数量巨大, 因此高度真实感的大面积草地的实时绘制成为一个难点. 现在的草地渲染技术主要采用 Lengyel 等^[1]提出的快速毛发渲染的思想, 利用多层镂空纹理来模拟草的片状多边形外观. 虽然采用真实的多边形来构成每棵草, 并使用光线跟踪的方式渲染可以达到非常好的视觉效果, 但是由于计算量过大, 不宜在实时交互应用中使用. Boulanger 等^[2]采用了近处使用真实草植株, 中远处使用多层镂空贴图, 远处使用地形贴图的方式, 来弥补多层纹理光照运算简单的不足. 这种方法虽然可以获得很好的视觉效果, 但是资源的耗费依然较大, 并不适合还有大量其他对象需要渲染的游戏场景. 对于 Lengyel 算法的改进一直在进行, 2006 年, 杨刚等^[3]提出层状阴影算法, 利用现代图形卡的图形处理器(GPU) 性能, 提高了 Lengyel 方法的自阴影表现. 本文论述一种用于增强游戏场景大面积草地绘制效果的方法.

1 算法概述

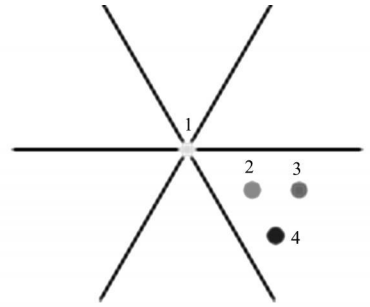
使用多边形表现草的叶子, 把一些草叶组合起来, 表示在一个纹理中. 图 1(a) 是一幅融合彩色通道信息的草纹理贴图. 该图被映射到按一定方式组合的一些多边形上, 使模拟的草丛显得茂密.

多边形的排列应尽量保证从不同视角看起来草都显得密集. 采用 Kurt 和 Piranha 提出的方法^[4], 在关闭背面剔除的情况下绘制成星形排列的六片正方形. 图 1(b) 显示的是该方法的俯视效果.

为了增加密度, 草多边形在绘制时进行适当偏移并多次绘制, 总共绘



(a) 草纹理贴图



(b) 草多边形布局俯视图

图 1 草和草丛模型

Fig. 1 Model of grass and hassock

收稿日期: 2009-04-25

通信作者: 陈柏生(1980-), 男, 讲师, 主要从事图像处理与模式识别的研究. E-mail: samchen@hqu.edu.cn.

基金项目: 华侨大学科研基金资助项目(08H ZR17)

制 4 遍, 取得比较好的覆盖率. 图 1(b) 中, 点 1 是首次绘制时使用的原点, 点 2, 3, 4 分别是 3 次偏移绘制使用的原点.

在 CPU 中生成草的多边形列表, 并将它导入显存中, 后续操作则充分利用现代显卡的 GPU 支持来实现, 如图 2 所示. 图 2 的算法流程有以下几个步骤: (1) 生成草多边形列表; (2) 绘制多边形列表; (3) 计算草多边形各点照度值; (4) 照度值与噪声值相乘; (5) 将步骤(4)的结果叠加到草纹理贴图.

首先, 在 Vertex Shader 中使用简化层状阴影算法, 计算穿过草丛的光的能量衰减, 获得草多边形中各点的照度值. 为了方便后续叠加操作的实现, 同时计算多边形各点的几何位置相对草纹理的坐标值.

然后, 向 Fragment Shader 输入一幅 Perlin 噪声纹理和草纹理贴图. 将之前在 Vertex Shader 中计算出来的草多边形各点的照度值与噪声纹理上对应点的强度值相乘, 再将结果叠加到草纹理贴图, 最终输出具有渐变自阴影效果和离散互遮挡阴影效果的真实感草地.

2 简化层状阴影

自阴影现象是光线照射到草地表面时, 由于草体的遮挡作用, 会使光线发生衰减, 在草丛空间形成阴影. 它对于真实地表现草地非常重要. 考虑到草地作为场景的一部分, 通常不是用户关注的焦点. 因此, 对该场景元素的绘制目标是, 在满足实时性要求的前提下获得全局性的视觉真实感效果, 并不需要关注草的细节和对草地进行细致绘制.

设想草地是一块均匀、连续的巨大半透明体, 由于草的遮挡作用, 会使入射光线发生衰减, 从而在草丛中形成自阴影效果. 光线穿过半透明长方体到达当前点的光照强度值, 由光源的位置、入射角度和半透明体的透光率决定, 如图 3 所示. 则有

$$l = \frac{(H - h) \times \sqrt{x^2 + y^2}}{y}, \tag{1}$$

各点的照度值等于该点到入射点距离(l)与半透明体透光率(α)的乘积.

计算草多边形各点的照度值, 将照度值叠加到草纹理贴图, 便可生成带有自阴影效果的草地. 有自阴影和无阴影绘制效果的对比, 如图 4 所示. 显然, 带有自阴影效果的草地看上去更为逼真.

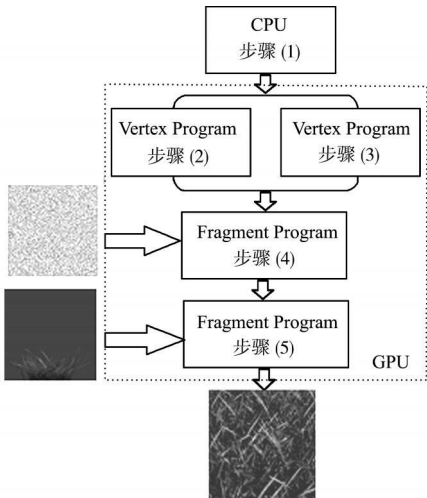


图 2 算法流程
Fig.2 Algorithm flow

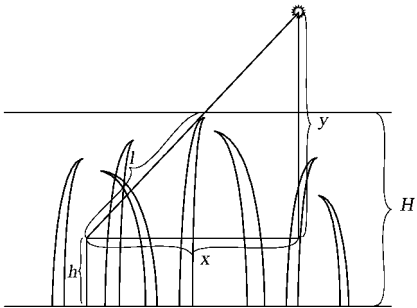
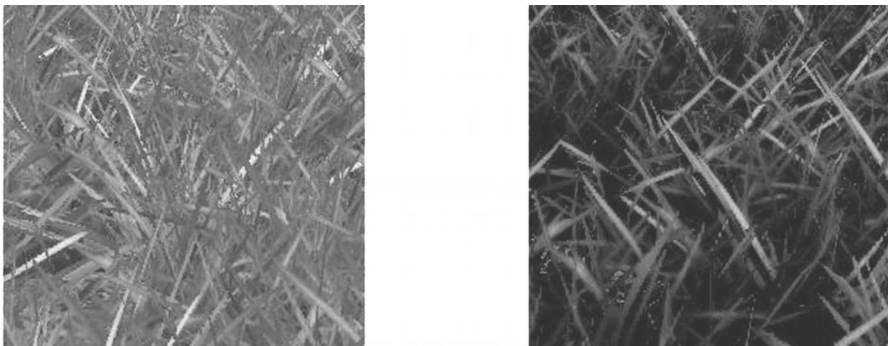


图 3 光线的衰减
Fig.3 ray attenuation



(a) 无自阴影效果 (b) 有自阴影效果
图 4 简化层次阴影算法绘制效果

Fig.4 Effect of simplified layered shadow rendering algorithm

3 Perlin 噪声阴影

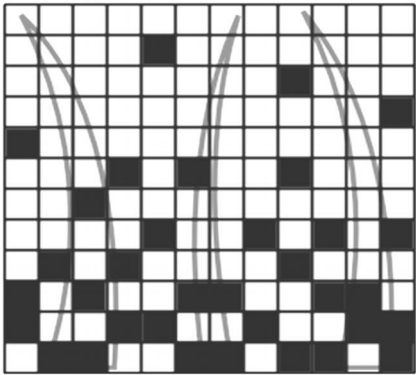
草具有正面宽大, 侧面狭窄的几何特征, 在宽大的正面会清楚地留下另一植株遮挡造成的投射阴影. 因此, 在草丛空间会形成杂乱交错的离散阴影、互阴影. 通过加入 Perlin 噪声模拟草的互阴影效果. Perlin 噪音由一系列噪声函数叠加而成, 其一般表达式为^[5]

$$t(x) = \sum_{i=0}^k \left| \frac{n(2^i x)}{2^i} \right|. \tag{2}$$

式中: x 输入 1 个三维空间点, k 是满足 $1/2^{k+1}$ 小于 1 个像素的尺寸的最小整数, $n(x)$ 为三维噪声函数. 在实际运用中, 现场计算 Perlin 噪音是不经济的, 一般直接使用预先计算好的 Perlin 噪音贴图. 使用一幅标准 Perlin 噪音贴图, 与从简化层状阴影算法步骤输出的阴影图做相乘叠加, 来增强草之间相互遮挡的互阴影效果. 具体实现时, 将阴影图上的阴影值和噪音纹理贴图上对应坐标点的灰度值相乘生成, 所产生的互阴影效果如图 5(b) 所示.

由图 5(b) 可见, 阴影由原来的平滑过渡变得相对不均匀, 即一些像素的色调变得较浅, 而另一些像素的变化不大, 同时暗点的分布仍然大致遵循原来阴影过渡的趋势.

为了防止最终结果出现明显的噪声特征——暗点过于细小, 阴影在叶片上呈网状, 需要视情况缩放噪声纹理的分辨率, 使得在绝对大小相同时, 噪声纹理上 3 个像素能够接近草纹理上大部分叶片的宽度. 这样, 暗点在草纹理上就会形成如图 5(a) 所示的横向贯通条纹的视觉感.



(a) Perlin 噪声阴影示意图



(b) 带有自阴影和互阴影效果的草地

图 5 Perlin 噪声阴影绘制效果

Fig. 5 Shadow rendering effect after multiplying perlin noise

4 实验与讨论

利用 OpenGL 2.0, 在 Linux 平台上实现. OpenGL 2.0 支持的 GLSL (OpenGL Shading Language) 可以充分发挥现代显卡上 GPU 的几何渲染器和片元渲染器的性能. 实验所用微机配置为 Athlon X2 3800⁺, 内存为 2 GB, 显卡是 Geforce 8600 GTS, 显存为 256 MB. 测试所用的草模型包含约 3 万个几何三角形, 纹理分辨率为 256 px × 256 px.

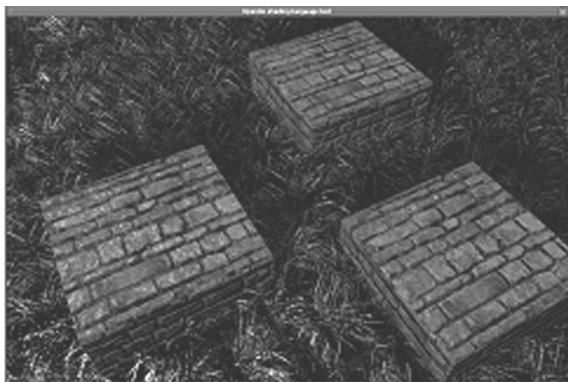
测试结果表明, 使用简化层状算法计算草多边形各点光照值的时间开销为 6.4 ms, 直接使用 Perlin 噪声贴图模拟草之间的互阴影效果, 叠加噪声操作的时间开销为 2.2 ms.

由于充分利用了 GPU 的加速性能, 整个草地绘制算法仅耗时 14.7 ms, 绘制带阴影和无阴影效果的草地帧率, 分别为 68, 130 帧 · s⁻¹. 由此可见, 该方法能够很好地满足三维游戏等实时交互应用对于大面积草地实时绘制的要求.

当光源入射角分别为 60° 和 45° 时, 草地绘制效果的侧视图和顶视图, 如图 6 所示. 从图 6 可以看出, 通过逐像素光照的准确计算, 可以细致地绘制草相互遮挡生成的自阴影效果. 真实的草之间往往杂乱交错, 呈现出无规则分布的离散阴影效果. 在叠加了 Perlin 噪声阴影后, Perlin 噪声破坏了自阴影的均匀分布, 使得草看上去更加逼真.



(a) 光源入射角为 60° 的侧视图



(b) 光源入射角为 45° 的顶视图

图 6 草地绘制效果

Fig.6 Grassland rendering effect

5 结束语

本文提出了一种简化层状阴影算法模拟草的自阴影效果, 直接将整个草地当作一块巨大的均匀半透明体. 因此, 它不是一种精确的绘制算法, 而 Perlin 噪声阴影更不具备物理真实性. 所以, 方法适合用于表现草地的整体效果.

参考文献:

[1] LENGYEL J, PRAUN E, FINKELSTEIA A, et al. Real-time fur over arbitrary surfaces[M]. New York: ACM, 2001.

[2] BOULANGER K, PATTANAIK S, BOU ATOUCH K. PI 1809: Rendering grass in real-time with dynamic light sources and shadows[R]. Rennes: IRISA, 2006

[3] 杨 刚, 孙汉秋, 王文成, 等. 基于 GPU 真实感毛发绘制[J]. 软件学报, 2006, 17(3) : 577-586

[4] RANDIMA F. GPU Gems[M]. 姚勇, 等译. 北京: 人民邮电出版社, 2006: 73-83

[5] ALAN W, FABIO P. 3D 游戏(卷 1) : 实时渲染与软件技术[M]. 沈一帆, 译. 北京: 机械工业出版社, 2005: 171-172

Real-Time Virtual Game Oriented Real-Effect Grassland Rendering with GPU

CHEN Bai-sheng, ZHOU Xie, CHEN Duan-shen

(College of Computer Science and Technology, Huaqiao University, Quanzhou 362021, China)

Abstract: A grassland rendering method for augmenting the realistic effects of visual scenes in 3D games was presented based on the prerequisite of real-time requirement of visual games. It abstracted the grassland as a uniform and continuous transparent body, and a simplified layered rendering algorithm was proposed. Taking the powerful supporting function of the graphic processing units, the self-shadow effect of the grass was simulated efficiently by calculating the optical energy attenuation which passing through the grass. Moreover, Perlin noise were multiplied to represented the disperse shadow resulting from interlaced grass, which augmented the render effects greatly. Results show that the method renders the grass shadow at very low costs, and generates large-scale grassland with realistic effects at realtime.

Keywords: virtual game; rendering algorithm; grass shadow; Perlin noise

(责任编辑: 鲁斌 英文审校: 吴逢铁)