

基于 OpenGL 的水波动画模拟

Based on OpenGL for water surface cartoon simulation

(新疆大学) 马义俊 巴力登

MA Yi-jun BA Li-deng

摘要: 本文详细叙述了在 VS2005 平台下,利用 OpenGL 图形技术,对水表面进行模拟,提出一种改进的 \sin 函数方法。并给出了快速实现水表面模拟的测试结果,得到了水表面模拟更加逼近的效果。

关键词: 海水表面仿真; OpenGL; \sin 函数

中图分类号: TP311 **文献标识码:** A

Abstract: The paper details the simulation of water surface using OpenGL Graphic technique on VS2005 platform, also we put forward an improving method of \sin function. And the testing result is speediness, and the simulation of water surface is also more approach.

Key words: simulation of Ocean surface; OpenGL; \sin Function

前言

近年来,计算机图形学的发展突飞猛进并有着极为广泛的用途。计算机图形学的发展使得三维表现技术得以形成,而真实感图形的生成算法是计算机图形学的重要内容,其中真实感海水表面的生成是近年来研究的热点和难点。对于海水的仿真有很多难点,首先是海水的波动受力很复杂,是各种力的综合,因此研究它的运动就非常复杂;其次海水的形状是随机的,动态的,还有海水的模型方程很难用普通方法求解,而且过程复杂,最后海水表面的颜色取决于光的折射和反射以及水底的一些杂物的折射等等,由于位置不同而产生不同的颜色。

OpenGL 是由 Silicon Graphics 公司推出的一种高性能图形开发软件工具包,该工具包括大约 120 条不同的指令,它独立于硬件,独立于窗口系统,由于具有这个优越的特性,因此可以运行于各种操作系统,而且可以在各个平台之间进行移植,并且能在网络环境下以客户/服务器模式工作,已经成为专业图形处理、科学计算等高端应用领域的标准图形库,因此 Microsoft, SGI, IBM 等在计算机市场中占主导地位的大公司都用它作为自己的标准图形库。

1 模拟水表面动画的方法

大量的文献对海洋表面仿真与动画,无论是在计算机图形学和海洋学等领域都有讨论。而且也已经有很多种方法对其进行仿真。

1 比如用 \sin 函数来进行仿真,也就是运用了水波的特性;

2 用傅立叶和里维克斯变换描写水波在计算机图形学里的实现,这种方法伸缩性很好,不仅可用于以高品质的水动画录像资料而且也可以用于实时性的仿真;

3 当然也用运用物理模型的方法,这种方法运用了一些理想状态,然后用图形学的方法把它表示出来,典型的就运用著名的流体方程 Navier-Stokes 方程,这个复杂的数学程序的有

益之处是转换高程和速度势成为一个比较简单动力学;

4 就是运用粒子系统,这种方法的思路是将流体看作粒子集;

以上四种方法是前人用的最多的,对于第一种,这种方法比较简单而且也给出了实际的数学模型,该模型对于硬件要求比较低,模拟出来的结果也可以达到一般的要求;第二种方法可以模拟出很好的海面效果,但是这种方法对硬件的配置要求比较高,尤其是 GPU;第三种方法运用了著名的流体方程,但是转化的情况其实是只是简单的海洋高度场,包括上升时候的色散关系;第四种方法可以模拟浪花飞溅但是不能模拟水波。

本文所采用的方法是第一种方法,也就是 \sin 函数,因为 \sin 函数的数学模型比较简单,连续性和可微性比较好,但是 \sin 函数非常光滑,而海水表面并不是光滑的,尤其是在浪尖和浪底的时候出现尖锐处,我们应用一些方法解决了这个问题,使得效果更佳。

2 模拟方案

1. \sin 函数的应用:

我们应用一组 \sin 函数来模拟海水表面的高度场和海水表面每一点的法线向量和切线向量。在处理网格顶点的时候,应用 \sin 函数的水平位移来设置网格的每一个顶点,用 \sin 函数的法向量产生海水表面的法向量图,这样有利于我们进行纹理映射。

首先我们需要选择一组 \sin 曲线,如下所示:

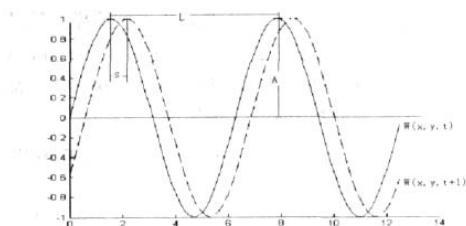


图 1

其中: L : 波长;

A : \sin 曲线的幅值;

S : 波速,它可以用相位来计算;

马义俊: 硕士研究生

基金项目: 新疆自治区高校重点科研项目资助(XJE-DU2007108)

接下来引出第*i*条曲线的水平位置(*x*,*y*)与时间*t*之间的关系,以及高度场的方程:

$$W_i(x, y, t) = A_i \times \sin(D_i \cdot (x, y) \times \omega_i + t \times \varphi_i) \quad 1.1$$

$$H(x, y, t) = \sum (A_i \times \sin(D_i \cdot (x, y) \times \omega_i + t \times \varphi_i)) \quad 1.2$$

为了实现场景中的动态效果,可以随机产生一些 sin 曲线,这样可以形成逼真的效果。当曲线向前运动一直到退出场景的时候,隐藏该曲线,然后随机的产生一个不同参数的曲线。

2.法向量和切向量的产生:

因为已经有具体的表达式,所以可以很方便的知道每一点的方向,而且可以看出法向量和切向量在*x*轴上和*y*轴上是相对独立的。并且在产生的时候我们进行归一化,如下所示,其中*B*(*x*,*y*)表示法向量,*T*(*x*,*y*)表示切向量

$$B(x, y) = \left[1, 0, \frac{\partial}{\partial x} (H(x, y, t)) \right] \quad 2.1$$

$$T(x, y) = \left[0, 1, \frac{\partial}{\partial y} (H(x, y, t)) \right] \quad 2.2$$

因为(1.2)式是很容易求偏微分,这对于这两个向量的计算很方便,而且在编程的时候很容易实现。

3.sin 曲线所遇到的问题:

从图 1-1 中可以看出, sin 曲线非常的光滑,具有很好的连续性和可微性,而现实的海面都有很尖的波峰和很宽的波谷,这就是 sin 曲线在模拟水波动画所遇到的问题。我们可以将海水除了波峰和波谷之外的地方看作是光滑的,因此我们处理的只剩下波峰和波谷这两个点。

对于这两个点的解决办法,我们可以对 sin 函数加幂次方来实现,在后边的程序中可以看到这种方法对模拟海面的波峰和波谷都有很好的效果,并且在下边给出的式子中我们可以很明显的看出该函数是可微的,这对于以后的步骤有很大的方便之处:

$$W_i(x, y, t) = A_i \times \left[\frac{\sin(D_i \cdot (x, y) \times \omega_i + t \times \varphi_i) + 1}{2} \right]^k \quad 3.1$$

在此处,我们给出了取不同*k*值的时候所对应的曲线的波峰和波谷,从图中可以看出,加入*k*次幂以后明显的产生浪尖和比较宽的浪底。

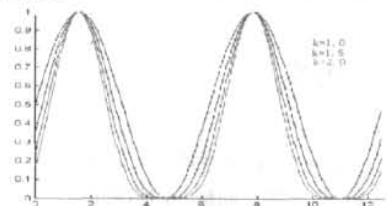


图 2

3 具体程序实现

Visual Studio 2005 是 Microsoft 最新推出的编程工具,包含了一系列高效的、智能的开发工具。Visual Studio 2005 和 .NET Framework 2.0 将在应用程序开发的所有方面取得大幅进展。

1. OpenGL 在 win32 下的运行机制

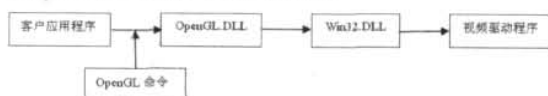


图 3

2.主要程序:

```
MyWaterView::CMyWaterView()
{
```

```
mesh = 0;
ttrans[0]=0.0f;
ttrans[1]=0.0f;
amplitude = 0.03;
freq = 5.0f;
phase = .003;
}
```

这段程序主要是定义频率和幅度、相位等。当我们适当改变它们大小的时候就会产生不同的波浪,比如改变幅度 amplitude 的大小就会引起很大的变化,适当调整它可以实现稍微平静的海面或者波浪起伏的海面。

```
.....
glTexCoord2f(s, t); glVertex3f(x, a, z);
s += d; t += d; x = -1.0 + 2.f*s;
z = -1.0 + 2.f*t;
y = amplitude * sinf(freq * 2 * M_PI * t + off) + 0.015;
y1 = amplitude * sinf(freq * 2 * M_PI * t + off + 2) + 0.015;

y2 = amplitude * sinf(freq * 2 * M_PI * t + off + 4) + 0.015;
y3 = amplitude * sinf(freq * 2 * M_PI * t + off + 6) + 0.015;
y = pow(y, 2); y1 = pow(y1, 3);
y1 = pow(y2, 4); y1 = pow(y3, 5);
a = y + y1 + y2 + y3;
glTexCoord2f(s, t); glVertex3f(x, a, z);
off += phase;
.....
```

这段是程序的核心部分,运用上面的初始化参数和 sin 函数以及在上文中讨论的方法进行海水表面的仿真。程序 $y = \text{amplitude} * \sin(\text{freq} * 2 * \text{M_PI} * t + \text{off}) + 0.015$; 这是我们所选择的一组 sin 函数中的一个函数,然后让它们的相位相继滞后,产生波动的表面, $y = \text{pow}(y, 2)$ 这段程序是让我们去除 sin 函数的波峰与波谷平滑的部分,因为在 VS2005 里没有办法实现带有小数的指数计算,因此我们选择了邻近的整数作为指数,其中参数 *a* 表示了 sin 函数曲线的叠加。而网格的生成我们用三角网格产生。并且我们在程序中加入了雾化效果,在此不详述。

2.运行结果:

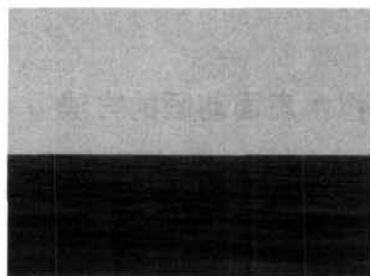


图 4

4 结论与展望

本文是在 vs2005 的平台下,利用改进的 sin 函数以及 OpenGL 强大的图形开发库模拟了海水的表面,而且 sin 曲线的实现对于硬件的要求比较低,程序的编写也比较简单,在一般的 PC 上均可以实现海面的模拟效果,当然还有下一步工作就是如何实现波浪飞溅等等效果。(下转第 259 页)

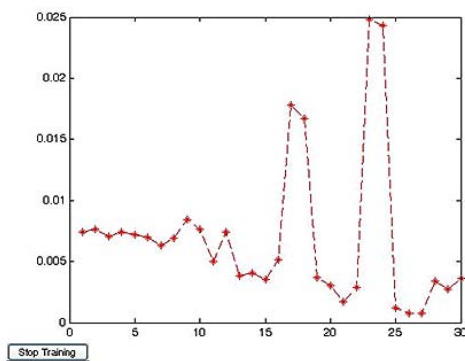


图3 样本实际分类值与标准值的误差

4 结论

本文利用基于窗口技术的图像分割方法及 BP 神经网络分类器,对针织面料缝绉加工引起的疵点部分类型进行了识别。可以从正常的针织面料表面区分缝线位置,并对缝绉加工中常见的因拉力不均造成的面料起伏疵点及褶皱疵点进行检测,成功区分了三类图像。本文对纺织品面料加工疵点的自动检验进行了初步研究,今后的研究目标将集中在自动检测织物加工疵点类型的扩展以及三维化检验等方面。

本文作者创新点:

- 1 研究纺织面料的缝绉加工工艺疵点检测。
- 2 以缝绉加工时容易出现加工疵点的针织面料为研究对象。
- 3 以针织罗纹线圈为结构单元构成分割窗口,实现图像处理。

参考文献

- [1]Takato, M., Takagi, Y., and Mori, T.. Automated Fabric Inspection Using Image Processing Techniques [J]. SPIE Auto. Inspect. High Speed Archit, 1988, 104: 151-158.
- [2]Cohen, Fernands S., and Fan, Zhigang etc.. Automated Inspection of Textile Fabrics Using Textural Models[J]. IEEE Trans. Pattern Anal. Machine Intell, 1991, 13(8): 803-808.
- [3]I-Shou Tsai and Ming-Chuan Hu. Automatic Inspection of Fabric Defects Using an Artificial Neural Network Technique[J]. Textile Research Journal. 1996, 66(7): 474-482.
- [4]李立轻,黄秀宝.基于织物自适应正交小波的疵点检测[J].东华大学学报,2001,27(4):82-87.
- [5]李立轻,黄秀宝.用于疵点检测的织物自适应正交小波的实现[J]. 东华大学学报,2002,28(2):77-81.
- [6]权蔚蔚,吴斌,张计华.基于神经网络的脂肪肝 B 超图像识别研究[J]. 微计算机信息,2007,(9-3):312-314.

作者简介:钱素琴(1971-),女(汉族),上海市,东华大学副教授,硕士,主要研究方向:图像处理及模式识别;董爱华(1970-),女(汉族),上海市,东华大学副教授,博士,主要研究方向:模式识别。

Biography: QIAN Su-qin (1971-),female (China),Shanghai, Donghua University, associate professor, master, Research area: image processing and pattern recognition.

(200051 上海 东华大学 信息科学与技术学院) 钱素琴 董爱华 (Dept of Institute of Information Science and Technology, Donghua University, Shanghai, 200051, China) QIAN Su-qin DONG Ai-hua

通讯地址:(200072 上海市灵石路 999 弄 13 号 403 室) 钱素琴

(收稿日期:2009.01.23)(修稿日期:2009.02.25)

(上接第 216 页)

本文作者的创新点:本文充分利用了 OpenGL 强大的图形、图像及动画处理功能,应用改进的 sin 曲线模拟了水波表面的动画,改进了前人用 sin 曲线直接模拟水波表面动画的不足之处。在图 4 中看出效果很好。

参考文献

- [1]赵启升.基于 OpenGL 三维物体建模方法的研究与实现[J]. 微计算机信息,2005,26:149-151.
- [2]郭伟雷,李少洪.基于 VC++/OpenGL 的引信仿真测试可视化系统软件设计[J]. 微计算机信息,2005,51-8:109-111
- [3]St. Petersburg State Polytechnical University, St. Petersburg, Russia, Real-time simulation of water surface
- [4]Damien Hinsinger; Fabrice Neyret; Marie-Paule Cani Interactive Animation of Ocean Waves
- [5]中国水利水电;《OpenGL 高级编程与可视化系统开发》
- [6]Randima Fernando; Chapter 1,《GPU Gems》,57-88
- [7]Jens Schneider and Rüdiger Westermann Scientific Visualization & Imaging Group Aachen University of Technology Seffenter-Weg 23 52056 Aachen,Germany Towards Real-Time Visual Simulation of Water Surfaces

作者简介:马义俊(1983-),男,籍贯:甘肃,硕士研究生,研究方向:计算机信息技术与控制;巴力登(1952-),男,籍贯:新疆维吾尔自治区,教授,研究方向:计算机信息技术与控制,计算机图形学。

Biography: MA Yi-jun (1983-), male, Gan Su province, graduate student, focus on the information technology and control of computer. (830008 新疆乌鲁木齐新疆大学电气工程学院) 马义俊 巴力登 (College of Electric and Engineering, Xinjiang University, Urumqi 830000, China) MA Yi-jun BA Li-deng

通讯地址:(830008 新疆大学电气工程学院 06 级研究生) 马义俊

(收稿日期:2009.01.23)(修稿日期:2009.02.25)

《PLC 技术应用 200 例》

PLC(可编程控制器)广泛地应用在冶金、机械、机器人、石油化工、电力传动、纺织机械、注塑机、包装机械、印刷机械、造纸机械、机床、自来水厂、污水处理、煤矿机械、焊接机械、榨糖机械、制烟机械、工程机械、水泥机械、玻璃机械、食品机械、灌装机械、橡胶机械、船舶、铁路、窑炉、车辆、智能建筑、电梯控制、中央空调控制、大型医疗机械、起重卷扬机械、大坝闸门、大型泵站……。

各行各业机械工程师,电气设备工程师,高级技工都需要具备 PLC 的知识,才能做好本职工作。本书汇集 200 多个硬 PLC 和软 PLC 在各行业的应用实例,PLC 故障诊断实例,PLC 抗干扰措施,PLC 使用经验、PLC 技术发展,均在本书之中论述。

本书适合大专院校机械类、电气类、电力类、自动控制和自动化类专业的本科、研究生做毕业设计参考,同时适合老师进行教学、搞科研项目参考。本书是上述各行业的工程技术人员、技术工人的必备参考书,同时也是工厂和科研单位的技术领导、设备采购负责人的参考书。凡具备高中以上文化水平的人均可成为读者。

200 多个西门子、三菱、美国通用电气、施耐德、欧姆龙、罗克韦尔、松下电器、和泉……等 PLC 应用实例,任您选读。一技之长,改变人生。

大 16 开,每册定价 110 元(含邮费)。预购者请将书款及邮费通过邮局汇款至

地址:北京海淀区皂君庙 14 号院鑫雅苑 6 号楼 601 室

微计算机信息

邮编:100081

电话:010-62132436

010-62192616 (T/F)

http://www.autocontrol.com.cn

http://www.autocontrol.cn

E-mail:editor@autocontrol.com.cn;

E-mail:control-2@163.com