

3 维卡通水与物体交互作用的动画建模

方建文^{1),2)} 于金辉¹⁾ 陈海英³⁾

¹⁾(浙江大学 CAD&CG 国家重点实验室, 杭州 310058) ²⁾(衢州学院信息与电子工程系, 衢州 324000)

³⁾(浙江大学 CADAL 研究中心, 杭州 310027)

摘要 为了生成水与物体交互作用的卡通动画, 提出一个混合式的 3 维卡通水模型。先构造出复杂的各种卡通水波模型, 然后在 2 维工作区上对它们进行结构上的组合并映射到 3 维水面上进行绘制; 障碍物在 3 维水面上的覆盖区域被检测出来之后, 将其逆映射到 2 维工作区并在工作区内实现流水波形和障碍物所在区域的碰撞检测, 根据碰撞检测结果触发障碍物附近的激起浪花波形。该方法的特点在于既表现了卡通流水与障碍物的碰撞, 又保持了手工绘画的艺术感。此外, 用户还可以方便地对波形的形态、位置、速度等进行设置与调控。系统利用 GPU (图形处理器) 硬件处理能力实现了对 3 维卡通水与物体交互的实时绘制。

关键词 卡通动画 流水 模型 图形处理器 计算机动画

中图分类号: TP274 **文献标志码:** A **文章编号:** 1006-8961(2010)03-507-06

Computer Generation of 3D Cartoon Water Colliding with Objects

FANG Jianwen^{1),2)}, YU Jinhui¹⁾, CHEN Haiying³⁾

¹⁾(State Key Laboratory of CAD&CG, Zhejiang University, Hangzhou 310058)

²⁾(Department of Information and Electronic Engineering, Quzhou College, Quzhou 324000)

³⁾(CADAL Research Center, Zhejiang University, Hangzhou 310027)

Abstract We present a hybrid model for 3D cartoon water modeling and rendering. First, we constructs some cartoon water forms and arranges them on 2D working region and map them onto 3D surfaces. The region covered with an obstacle object is detected on the 3D surface and then mapped reversely onto the 2D working region. The spray form is added if the flowing water forms collide with the mapped region. Our method is able to preserve the cartoon style of water forms during the interactions between the obstacle object and the cartoon water. Moreover, user can adjust the water forms, positions and the movement rate with ease. The real-time rendering is achieved by taking advantage of the graphics processing unit.

Keywords cartoon animation, water, modeling, GPU, computer animation

0 引言

目前已经有多部 3 维卡通动画电影出现, 但只是对动画角色造型以及对其动态控制进行夸张(即卡通化)处理, 而影片中的特效却仍采用真实感方

式处理, 结果使得影片中特效和角色风格不一致。造成这种现象的原因是 3 维卡通特效方面的研究基本上还处于起步阶段, 而真实感特效的研究已经有 20 多年历史, 其中流水动画的研究已经有 100 多篇论文发表^[1-2]。目前计算机图形学对水动画的生成主要利用计算流体动力学(CFD)对水进行物理的模

基金项目: 国家自然科学基金项目(60673007); 国家高技术研究发展计划(863)项目(2006AA01Z312); 浙江省科技计划项目(2007C21043)

收稿日期: 2008-09-11; 改回日期: 2008-09-24

第一作者简介: 方建文(1972—), 男, 副教授。现为浙江大学 CAD&CG 国家重点实验室计算机应用技术专业博士研究生。主要研究方向为计算机动画、非真实感绘制。E-mail: fangjianwen@cad.zju.edu.cn

拟,或是用粒子系统^[3],它们能生成非常逼真的流水效果,并在影视特技中得到广泛的使用。而传统卡通流水在形态上以及动态上都是风格化的,上述生成真实感流水的方法很难进行流水形态和动态的风格化处理。

近几年来有少数研究人员开始关注卡通流水效果的计算机建模研究。于金辉等人用随机正弦波合成卡通流水^[4];用样条曲线构造卡通流水浪花的形状并在 CAVE (cave automatic virtual environment) 系统中显示出立体卡通流水效果^[5]。陈天洲等人提出一个卡通流水框架生成与电视动画质量相当的流水动画。Fiore^[7]和 Thornton^[8]等人分别用粒子生成卡通喷水和大水冲击礁石的动画,其缺点是生成的喷水动画具有强烈的粒子感觉,缺乏手工绘制卡通流水效果的美感与力量感。Eden 等人把流体模拟的水面作为输入,然后对流体表面用轮廓线以及平坦颜色绘制得到卡通化的流水效果^[9]。这种运用流体模拟的方法其动态是真实的,但形态上缺乏手工绘画感觉。于金辉等提出基于模板的方法来生成卡通水效果,包括流水,喷水以及涟漪^[10]。

在上述相关工作中,只有文献[5,8-9]中的模型可以生成 3 维流水效果,但文献[5,10]只能生成不太激烈的平稳流水波形,且不能处理卡通流水和物体之间的碰撞,基于粒子和流体模拟^[8-9]虽然可以处理水和物体之间的碰撞,但缺乏手工绘画的艺术感。

生成 3 维卡通流水的主要难点在于卡通流水造型以及动态都是风格化的,更为激烈的卡通流水波形在形态上比较复杂,直接构造它们的 3 维卡通模型非常困难。这里提出一个混合式的 3 维卡通水模型,先在 2 维空间上构造出复杂的各种卡通水波,然后映射到 3 维水面上,同时还能处理水流和物体之间的碰撞,如流水撞击水中石头引起的浪花以及石头落入水中引起的溅水等。本文方法的特点在于既表现了卡通流水与障碍物的碰撞,又保持了手工绘画的艺术感。此外,用户还可以方便地对波形的形态、位置、速度等进行设置与调控。本系统用 Direct3D 实现,利用 GPU 硬件处理能力实现了对 3 维卡通水与物体交互的实时绘制。该系统框架如图 1 所示。

其中,粗线方框对应预处理阶段,细线方框对应动画生成阶段。卡通水波模型包括流水、流水碰撞

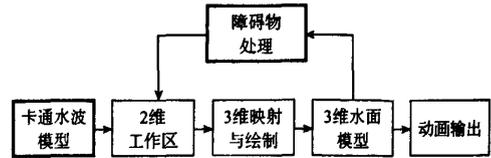


图 1 3 维卡通水绘制过程

Fig. 1 Rendering process of 3D cartoon water

障碍物以及溅水中所需的各种卡通水波模型;2 维工作区中按照一定的空间结构安排、绘制所需的各种卡通水波,并随时间动态更新绘制内容;3 维映射把相应的 2 维水波映射到 3 维水面上,绘制处理包括水波内部涂色、水波轮廓线绘制以及在波形内部随机添加少量的空洞等;3 维水面是表示水面的曲面模型,在该曲面上绘制各种运动的卡通水波;障碍物处理先在 3 维空间检测障碍物与流水曲面相交区域,然后将该区域映射到 2 维工作区,并在工作区进行运动水波与障碍物区域之间的碰撞检测以及添加激起的浪花波形。

1 卡通水波模型

图 2 为几幅手工绘制的卡通水画面,图 2(a)为山涧激流,图 2(b)为流水撞击石头,图 2(c)为溅水效果。在本文将针对这些卡通波形建模,具体将分类介绍不同类别卡通波形的构造方法。

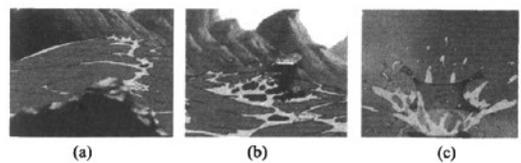


图 2 手工绘制的卡通水

Fig. 2 Samples of hand drawn cartoon water

1.1 流水主波模型

图 2(a)所示的卡通流水波中有大致等间隔的主波以及零星分布在它们之间的散波。每个主波在整体上呈低频震荡状,其前方由一系列小的弧线构成,其后方由一些较大的抛物线状曲线构成。在主波上还随机出现一些小洞。

主波的形状可以用一个较为简单的方法模拟。先用一个随机过程模型生成一系列的节点

$$M_i = w + \Delta_i \quad (i=1, \dots, n) \quad (1)$$

式中, M_i 为第 i 个节点, w 为一常数,它控制着波形

的平均高度, Δ 为随机扰动, n 为生成节点数。接下来在生成节点中每次取两个相邻点 M_i, M_{i+1} , 然后在它们中间添加一个新节点 M_{in} , 令其幅度在 $(0.3 \sim 0.5) \times \min(M_i, M_{i+1})$ 之间变化, 最后对 M_i, M_{in} 和 M_{i+1} 用样条曲线插值得到一个弧状曲线。以同样方式生成下一个弧状线直至到最后一个节点为止, 这些弧线就是主波的前端边缘曲线。主波后端的边缘线用类似的方法生成, 只不过节点幅度以及采样区间均比前端边缘线的要大。图3给出用该方法生成的主波形状。

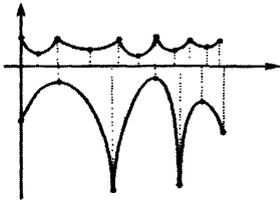


图3 主波波形
Fig. 3 Main wave form

1.2 激起浪花模型

对于流水碰撞障碍物后激起的浪花波形, 其模型主要由两部分构成: 前端波头和后端波尾。其前端边缘线构造方法如下: 首先在一个椭圆上利用随机变化跨度采样, 用样条曲线对这些采样点插值得到一骨架曲线(图4虚线), 再其上添加一系列开口朝外的短弧, 每个短弧的跨度和高度随机变化。在其内部加入一些小椭圆, 其大小和方向也随机变化, 表示破碎的洞。其后端用少量大弧线模拟, 用于表现激起浪花的尾部, 一个完整的激起浪花波形如图4实线所示。

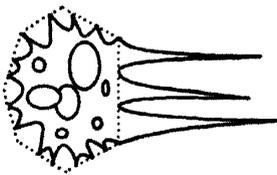


图4 由碰撞激起的浪花波形
Fig. 4 Spray over the obstacle object

应该指出, 激起浪花的尾部并不是由于碰撞造成的。添加它们是因为在系统中卡通水波遇到障碍物后要分别在侧面通过障碍物, 如果仅添加激起浪花的前半部分波形会在障碍物后面露出没有波形的区域, 在视觉效果上非常不合理。如果把尾部波形加上会在视觉效果上使得障碍物后面的区域同其他

邻近区域波形看起来一致。

1.3 溅水模型

当一个物体掉入水中后, 物体挤压使得水在其周围向上溅起, 溅起来的水面在边缘上有不规则的凹凸。在整体上, 溅水随时间从小到大, 并伴有一些零散的波形, 然后边缘撕裂, 溅水面下降, 最后消失。

溅水在上升阶段和下降阶段其边缘形态是不一样的。对于上升阶段, 先确定 m 个控制点 C_i (系统中 m 在 4 ~ 8 范围内随机选定, 它对应溅水边缘上的凸点数量), 其横向坐标 u_i 和纵向坐标 v_i 分别确定如下:

$$\begin{aligned} u_i &= id + \Delta \\ v_i &= v'_i t - gt^2/2 \end{aligned} \quad (i = 1, \dots, m) \quad (2)$$

式中, d 为 $1/m$, $\Delta \ll d$ 为随机扰动, v'_i 为随机分配的一个初速度, g 为重力加速度, t 是从溅水开始起到目前的持续时间。

在溅水边缘上, 上述控制点之间还需添加若干点来控制边缘线形状。这些点高度较 C_i 低, 采用递归二分插值算法分段确定它们的位置, 该过程可以用伪码表示如下:

```

For (i = 1 to m - 1) {
    Set CPi = NULL;
    Set level stochastically between 1 and 3;
    GSAMP(Ci, Ci+1, CPi, level); }

GSAMP(Ci, Ci+1, CPi, level) {
    fr ← Rndf(0.4, 0.6);
    Cin. u ← uifr + ui+1(1.0 - fr);
    Cin. v ← min(vi, vi+1) * Rndf(0.8f, 0.9f);
    if (level = 1) then CPi ← CPi ∪ Ci ∪ Cin;
    else {
        GSAMP(Ci, Cin, CPi, level - 1);
        GSAMP(Cin, Ci+1, CPi, level - 1); }
}
    
```

这里 $Rndf(a, b)$ 表示生成从 a 到 b 之间的一个随机数。 C_{in} 是 C_i 与 C_{i+1} 之间生成的一个新控制点, fr 是控制点 C_{in} 位置的一个随机参数, CP 为各段经过二分插值之后所得到的所有点的集合, 最后用样条曲线对 CP 中的点进行插值得到溅水边缘线, 图5(a)就是 $level$ 取 1 生成的波形。

在溅水下降阶段, 其边缘线生成方法与上升阶段边缘线类似, 只是控制点分布更密一些, 边线上高凸点和低凹点之间高度差要小些(图5(b))。

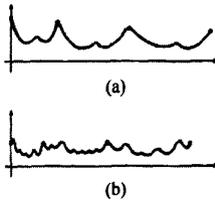


图 5 溅水边缘线

Fig. 5 Boundary lines of water splash

1.4 零散水波

在流水主波之间以及溅水周围还伴有一些零散波形,它们可以用一些简单的几何模型确定出控制点,然后用样条曲线对控制点插值得到各种散波形状,如图 6 所示。

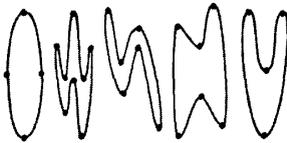


图 6 各种散波波形

Fig. 6 Scattered wave forms

2 2 维工作区与 3 维水面模型

2 维工作区是一个用归一化坐标 (u, v) ($0 \leq u < 1, 0 \leq v < 1$) 表示的 2 维区域(图 7),在其上面对各种卡通水波进行排列以及绘制。以卡通流水为例,工作区的上下两个边线作为流水路径参考线,每隔一定距离安排一个主波,在主波之间随机安排一些散波。为了把 2 维工作区上的水波映射到 3 维水面上,所有卡通波形都通过 Constrained Delaney Triangulation 进行三角化处理。

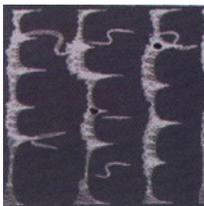


图 7 各种波形在 2 维工作区上的排列

Fig. 7 Wave arranged on 2D working region

3 维水面在系统中表示水波运动所在曲面,它可以是静态的,如图 8(a)表示的河流水面曲面以及图 8(b)表示的碰撞障碍物激起的浪花所在曲面;也

可以是随时间动态变化的,如图 8(c)表示的溅水曲面。这些曲面有的则需要用现有的 3 维图形软件如 3DMAX 根据出场景设计出来(如图 8(a)(b)),有的可以用一些曲面公式生成,如呈喇叭形状的溅水曲面,它可以用如下参数方程确定:

$$R_{up}(t) = 1 + 7t + \Delta, H_{up}(t) = 2t$$

$$R_{dn}(t) = 1 + \Delta, H_{dn}(t) = 2t$$

这里 R_{up}, H_{up} 表示溅水曲面上升阶段的半径和高度, R_{dn}, H_{dn} 表示溅水曲面下降阶段的半径和高度, $0 \leq t < 1, \Delta$ 表示随机扰动,它的引入是为了避免生成的曲面过分光滑而造成机械感。

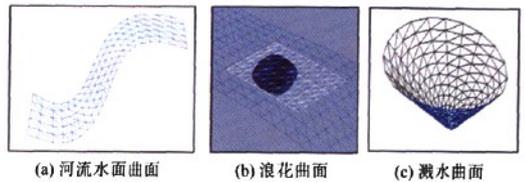


图 8 3 维水面

Fig. 8 3D water surfaces

系统中无论是用 3 维软件设计的还是用公式生成的水面都是由网格构成,每个网格节点 P 在曲面上都对应一个行、列坐标 $[i, j]$ ($i = 0, \dots, M - 1; j = 0, \dots, N - 1$),这里 M 和 N 分别表示网格上的行数 and 列数。对于不同的水面模型 M 和 N 取值可以不同。

3 3 维映射与绘制

为了把 2 维工作区上水波形内部的各个点映射到 3 维水面上,需要把它们表示成水波三角网格节点的某种关系。如对于 2 维工作区中水波中的某点 $P(u, v)$,先计算它在 3 维水面上的坐标: $u' = u(M - 1), v' = v(N - 1)$,然后对它们向下取整数得到 i 和 j ,通过它们可以在 3 维水面上得到一个网格点 $P'_1[i, j]$,再取另外 3 个网格点 $P'_2[\min(i + 1, M - 1), j], P'_3[\min(i + 1, M - 1), \min(j + 1, N - 1)], P'_4[i, \min(j + 1, N - 1)]$,最后根据 $f_1 = u' - i$ 和 $f_2 = v' - j$ 两个量对这 4 个点进行双线性插值得点 $P(u, v)$ 在 3 维水面上的对应点 $P'(x, y, z)$,通过这种映射方法可以将 2 维工作区中的点一一映射到 3 维水面上。

反之,对于 3 维水面中的网格点 $P'[i, j]$,也可以通过 $u = i / (M - 1), v = j / (N - 1)$ 求其在 2 维工作区中的对应点 $P(u, v)$ 。图 9 为上述映射的示意图。

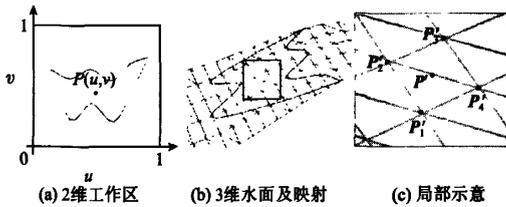


图 9 映射与绘制过程

Fig. 9 Process of mapping and rendering

应该指出,虽然这种映射方式与纹理图像映射非常相似,但本文方法是把水波形上的点从 2 维工作区映射到 3 维水曲面上,这样就保证了各种水波形状以矢量化方式进行映射,避免了普通纹理图像映射在离视点很近处发生的模糊现象。

在绘制阶段,每个 3 维水曲面用较深蓝色绘制,水波用浅蓝色绘制,水波边缘用深蓝色绘制。

4 障碍物处理

卡通流水波形在流动区域中如果遇到障碍物时要在障碍物两侧流过,同时在障碍物面朝水流方向激起浪花,此外在障碍物与水面交界处还有动态的纹波。为了生成这些效果,首先对障碍区域进行检测。

在系统中用户先把障碍物模型(如石头)拖拽到水面曲面中。系统先在水面曲面上的每条横向线按顺序与障碍物进行求交运算,如图 10(a)所示,设 P 为起点,箭头方向为求交的射线方向。如果射线与障碍物相交,在所有的交点中取与 P 最近的一个点 P_1 ,最远的一个点 P_2 ,如只有一个交点就取 $P_1 = P_2$,然后在水面网格点中分别求出与 P_1, P_2 最近的网格点 P_{B_1}, P_{B_2} 。图 10(b)就是由所求网格点组成的障碍物边界线。利用这些网格点的行列坐标按照前述的方法就可以求出障碍物在 2 维工作区中对应的障碍区域。

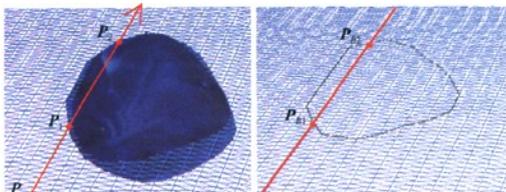


图 10 碰撞区域检测

Fig. 10 Collision detection

接下来把障碍区域映射到 2 维工作区,然后在该工作区生成一个包围盒,在流水波形运动期间检测各波形和包围盒的碰撞,如果发生碰撞则流水波形分成两部分分别在障碍物两侧通过,同时触发溅起浪花绘制模块。

5 结果

系统是在 Pentium 2.8G, 1G 内存, NVIDIA 7300 GT 显卡, Window XP 操作系统下实现,开发软件是 DirectX 9.0 SDK。

图 11 是系统生成的 3 维卡通流水在运动路径上碰撞石头的 6 幅动画序列。图 12 为改变视点的 3 幅动画序列。图 13 为表现溅水的 9 幅动画序列。

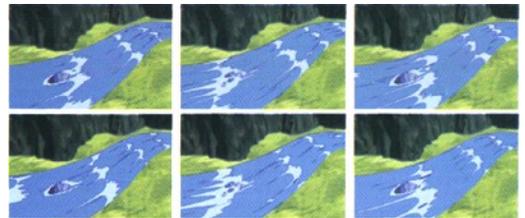


图 11 3 维卡通流水与障碍物碰撞

Fig. 11 3D cartoon water interacts with obstacles



图 12 改变视点的动画序列

Fig. 12 3D cartoon water in different views

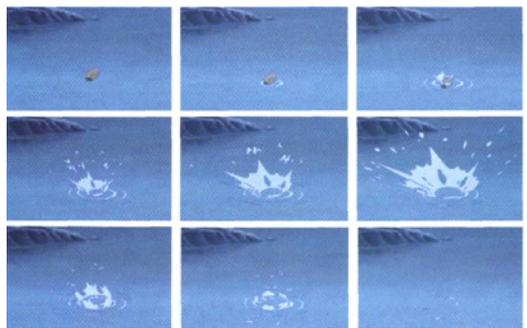


图 13 卡通溅水过程

Fig. 13 Splashing process

6 结 论

提出一个 3 维卡通水绘制的方法。用比较简单的模型构造出较为复杂的各种卡通水波形状,实现了在 3 维空间表现卡通流水和障碍物之间的碰撞效果。通过该系统用户可以方便地对波形的形态、位置、速度等进行设置与调控。利用 GPU 硬件处理能力实现了对 3 维卡通水与物体交互的实时绘制。下一步工作打算向更复杂的卡通水动画生成扩展,包括海水面在不同风力条件下的水波造型以及动态控制等。

参考文献 (References)

- [1] Iglesias A. Computer graphics for water modeling and rendering: A survey [J]. Computer Graphics Forum, 2004, 20 (8): 1355-1374.
- [2] Adabala L, Manohar S. Techniques for realistic visualization of fluids: A survey [J]. Computer Graphics Forum, 2002, 21: 65-81.
- [3] Reeves W T. Particle systems—a technique for modeling a class of fuzzy objects [J]. ACM Transactions on Graphics, 1983, 2(2): 91-108.
- [4] Yu Jinhui, Xu Xiaogang, Peng Qunsheng. Cartoon water forms synthesis using random sinusoidal functions [J]. Journal of Computer Research and Development, 2001, 38 (5): 519-523. [于金辉,徐晓刚,彭群生. 用随机正弦波拟合卡通流水[J]. 计算机研究与发展, 2001, 38(5): 519-523.]
- [5] Yu Jinhui, Xu Xiaogang, Wan Huagen, et al. A cartoon water model for stereo animation [J]. Journal of software, 2002, 13(4): 475-481.
- [6] Chen Tianzhou, Yan Hongjie, Yu Jinhui. A cartoon water system [J]. Journal of System Simulation, 2004, 16(11): 2479-2492. [陈天洲, 颜宏杰, 于金辉. 卡通流水生成系统[J]. 系统仿真学报, 2004, 16(11): 2479-2492.]
- [7] Fabian Di Fiore, Johan Claes, Frank Van Reeth. A framework for user control on stylised animation of gaseous phenomena [C]// Proceedings of Computer Animation and Social Agents (CASA2004), Geneva, Switzerland: John Wiley & Sons, 2004: 171-178.
- [8] Thornton John David. Directable simulation of stylised water splash effects in 3D space [C] // Proceedings of SIGGRAPH 2006 Conference. New York: ACM Press, 2006: 94.
- [9] Eden A M, Bargteil A W, Goktekin T G, et al. A method for cartoon-style rendering of liquid animations [C]// Proceedings of Graphics Interface. New York: ACM Press, 2007: 51-55.
- [10] Yu Jinhui, Jiang Xin'an, Chen Haiying, et al. Real-time cartoon water animation [J]. The Journal of Computer Animation and Social Agent, 2007, 18 (4-5): 405-414.