

# 图像漫游：双灭点的松弛蜘蛛网格算法

刘喜作<sup>1)</sup> 徐晓刚<sup>1),3)</sup> 周明<sup>2),1)</sup> 鲍虎军<sup>3)</sup> 马利庄<sup>4)</sup>

<sup>1)</sup>(大连舰艇学院自动化系,水中武器系,大连 116018) <sup>2)</sup>(西北工业大学航海学院,西安 710072)

<sup>3)</sup>(浙江大学 CAD&CG 国家重点实验室,杭州 310027) <sup>4)</sup>(上海交通大学计算机系,上海 200030)

**摘要** 为了解决蜘蛛网格算法在图像漫游中局限性,提出了双灭点的松弛蜘蛛网格算法。该算法在用于图像实时漫游时,用双灭点进行计算,其与人提出的方法相比,更具通用性,不仅可以解决单灭点、双灭点、多灭点和灭线等不同类型的照片漫游问题,而且减轻了用户交互的工作量。此外,为了进一步提高图像的绘制质量,以获得高质量的漫游效果,还采用了一种分片绘制的策略。多种类型的图片实验证明了该算法的有效性。

**关键词** 双灭点 图像漫游 绘制

中图法分类号: TP391.9 文献标识码: A 文章编号: 1006-8961(2005)03-0344-05

## Image Based Walkthrough: An Approach Using a Relaxing Spidery Mesh with Two-vanishing Points

LIU Xi-zuo<sup>1)</sup>, XU Xiao-gang<sup>1),3)</sup>, ZHOU Ming<sup>2),1)</sup>, BAO Hu-jun<sup>3)</sup>, MA Li-zhuang<sup>4)</sup>

<sup>1)</sup>(Department of Automation, Department of Underwater Weapons, Dalian Naval Academy, Dalian 116018)

<sup>2)</sup>(College of Marine Engineering, Northwestern Polytechnical University, Xi'an 710072)

<sup>3)</sup>(State Key Laboratory of CAD&CG, Zhejiang University, Hangzhou 310027)

<sup>4)</sup>(Department of Computer Science & Engineering, Shanghai Jiaotong University, Shanghai 200030)

**Abstract** A new approach for solving the spidery mesh limits in image-based virtual touring, which uses a relaxing spidery mesh with two-vanishing points, is proposed in this paper. Comparing with other researches work, this approach can be applied to a wider range of images for walkthrough. We have found that it can tour into the picture which has only one vanishing point, double vanishing point, and multiple vanishing points. The test result also shows that our approach can be effectively applied to the picture which contains vanishing line. In our approach, the interactive work is very simple as the user need only to interactively choose two vanishing points, the background and front object. To avoid the texture distortion, a small block texture mapping technique is used to improve the rendering quality of scene. In the test, different kinds of images are used to show the effectiveness of our approach.

**Keywords** two-vanishing points, image-based walkthrough, rendering

## 1 引言

建立真实感较强的场景,以实现大范围漫游,一直都是计算机图形学研究的目标之一。由于基于图像的绘制方法(image-based rendering,简称IBR)避

开了复杂场景的几何建模与绘制,且可直接利用照相机拍摄得到的实景图像通过适当组合来产生场景的新视图,其交互显示的开销只与分辨率有关,而与场景的复杂度无关,因此,它具有快速、简单、逼真的优点,已成为近年来一个新的研究热点。目前,按技术实现的方式,IBR方法可以分为以下两大类:一类

基金项目:国家自然科学基金项目(60173035,60473141);国家重大基础研究发展规划项目(2002CB312102)

收稿日期:2004-02-12;改回日期:2004-08-30

第一作者简介:刘喜作(1975~),男,1998年毕业于大连舰艇学院航海专业,获作战指挥学士学位,2004年获大连舰艇学院兵种战术硕士学位,助教。目前研究方向为图像建模与绘制、图像处理、系统仿真等。E-mail:xxgang@cad.zju.edu.cn

是通过图像来获得新图像;另一类是利用图像加简单几何模型来获得新图像。

其中,第1类方法的主要思想是首先拍摄多张图像,然后对图像进行各种变换、采样,以获得新视点的图像,进而实现漫游。这种方法的特点虽不需要建立几何模型,但由于受采样图像的限制,故漫游范围较小<sup>[1-3]</sup>。就目前的状况来说,单纯利用采样图像重建技术还很难绘制一般场景,许多问题仍待解决。

第2类是以 Horry 等的 TIP (tour into the picture) 算法思想为代表<sup>[4]</sup>,即首先利用一幅或多幅图像构建简单几何模型,然后用图像加简单几何模型来表达场景,最后通过生成新视点图像来实现基于照片的漫游。这类方法由于引进了几何模型,因此不仅可以实现较大范围的漫游,同时还能获得较好的视觉效果,已成为近年来国际上的一个研究热点,并先后提出了多种改进算法<sup>[5-9]</sup>,但由于这些算法都只能处理一种类型的照片,因此不具一般性,同时存在场景清晰度不高的问题。

针对已有算法存在的不足,本文提出了一种基于松弛蜘蛛网格的算法,用该算法计算时,左墙和右

墙各自单独形成灭点,同时利用分片绘制的策略来进一步提高绘制质量。实验证明,本文方法不仅可以处理单灭点、多灭点和灭线等不同类型的照片,同时可以保持漫游清晰、自然。

## 2 双灭点松弛蜘蛛网格模型

本文的主要技术路线是:首先通过交互指定图片的前景和背景来进一步建立前景、背景的3维网格模型,然后把纹理映射到网格上,通过建立新的场景来实现照片的实时漫游。

### 2.1 背景建模

Horry 等提出的蜘蛛网格模型<sup>[4]</sup>如图1(a)所示,它对场景有以下严格的限制:(1)用户在交互中可指定灭点和内部矩形,灭点虽可以任意指定,但必须在内部矩形之内;(2)局部的3维模型必须满足下列条件:①相邻的矩形相互垂直,②后墙与投影平面平行。图1(a)这种网格只适用于较广阔的景物或类似于平行的街道这类规则场景,这样恢复出的3维场景才不至于变形严重。

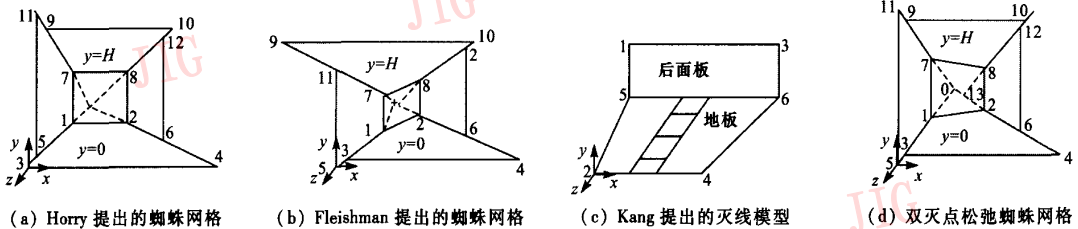


图1 各种算法的背景划分与坐标定位

Fig. 1 Different mesh and the background model

Fleishman 等对原来的蜘蛛网格放宽了约束条件<sup>[5]</sup>,放宽了约束条件的蜘蛛网格如图1(b)所示,它的特点如下:(1)左墙和右墙不一定必须相互平行,这就意味着这两个墙4条水平线不必汇聚到单一的灭点上。也就是左墙和右墙将分别有各自的汇聚点;(2)后墙虽不必平行视平面,但是这些墙必须与地板垂直。用这种网格计算时,还是按单灭点方法来进行,灭点取为两汇聚点的中间点。

Kang 等对背景模型进行了简化<sup>[8]</sup>,即只需要地板和后墙,其他面都用前景代替,简化后的模型如图1(c)所示。

文献[4,5]在建立模型时,虽都做了场景投影汇聚到一点的假设,但在实际的照片中,往往存在多个甚至无穷多个灭点,这样在3维构建和投影时,就

会产生误差而导致变形。文献[8]虽然背景模型简单,但处理单灭点的照片时,则使前景建模变得十分复杂;另外,地板是一个由视点位置向无穷远处延伸的平面,虽然在纹理映射过程中,可取相应的远处近似为无穷远点,但地板拉伸变形比较明显。

为解决一般照片的漫游问题,本文建立了以下模型:(1)左墙和右墙各自单独形成灭点,两灭点之间的连线为灭线,且左右墙各点分别按照各自的灭点进行计算以构建3维场景;(2)后墙不必平行视平面,但是这些墙必须与地板垂直;(3)场景的高度以左墙计算得到的高度为基准,右墙根据其进行相应的调整来保证一致的场景高度,以便产生较好的透视效果(如图1(d)所示)。

本文建立了如下坐标系:以照片左下角点为原

点,  $x$  轴水平向右,  $y$  轴垂直向上,  $z$  轴垂直  $x-y$  平面向外(如图 1(d)所示), 视点的位置位于两个灭点中间,  $z=0$ , 视点方向垂直  $x-y$  平面向内。设  $V_i$  ( $i=0, \dots, 13$ ) 表示图中对应的两个灭点(0 和 13)和 12 个顶点, 且以左墙计算为例:  $V_1 = (0, 0, z_1)$ , 地板在  $y=0$  的平面上, 视点到投影平面的距离为  $f$ , 视点到后墙的实际距离为  $F$ , 左墙的高度为  $H$ , 则从 2 维图像上就可以得到左墙各点的 2 维坐标和内部矩形左侧的高度  $h$ 。根据投影原理

$$f/F = h/H \quad (1)$$

由于左灭点  $V_0$  在 3 维空间被认为是无穷远的点, 所以其中投影参数  $f$  与  $F$  的比值可由图像上左灭点  $V_0$  到  $V_1$  在  $y$  方向上的距离  $h_1$  与左灭点  $V_0$  在  $y$  方向上坐标分量  $h_0$  的比值来确定,  $h$  为图像上顶点  $V_7$  到  $V_1$  在  $y$  方向上的距离。根据式(1)得到的 3 维空间左墙的高度  $H$ , 以及  $V_1$  的深度坐标  $z_1 = -h_0/h_1$ , 则可以确定  $V_7 = (0, H, z_1)$ 。同样根据投影原理依次可以计算剩余左右墙各点的坐标。

## 2.2 前景建模

对于前景建模, Horry 用多边形来构建前景物体。如果前景某物体复杂, 则用多个多边形来处理解决<sup>[4]</sup>, 这适用于从多幅图片中构建前景或从别的图片提取构建前景, 对于单张图片来说, 前景某物体无论多复杂, 其最终都将映射到平面上, 当然也可以采用文献[10]的建模方法来进行前景建模, 但由于单张照片的信息终究有限, 用四边形来构建图片中的前景物体最方便和实用, 因此, 本文中也采用四边形来构建图片中前景物体的网格模型, 对于多个前景物体可用多个四边形来处理<sup>[4]</sup>。

## 2.3 绘制

本文用含有透明通道  $\alpha$  值的 RGBA 图片作为前景图片进行绘制, 若  $\alpha$  值为 1, 说明此点属于前景物体, 则绘出其 RGB 值; 若  $\alpha$  值为 0, 将显示背景颜色, 则其绘制过程如图 2 所示。如果改变视点位置

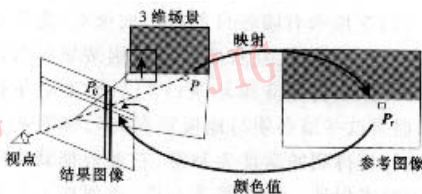


图 2 场景绘制过程示意图

Fig.2 Rendering mechanism

和方向后产生新画面中的  $P_0$  点, 且它在构建的 3 维场景中对应是  $P$  点, 而  $P$  点在未改变视点位置和方向的参考图像中对应的是  $P_r$  (下角  $r$  代表 reference) 点, 那么新画面中的  $P_0$  点将用参考图像中与对应  $P_r$  点的颜色来绘制。如果  $P$  点处在背景中, 那么就将背景图像作为参考图像进行绘制; 如果  $P$  点处在前景中, 那么这时要检查  $P_r$  点的  $\alpha$  值是否为 1, 如果为 1, 说明  $P$  点位于前景物体之内, 则前景图像中  $P_r$  点的颜色值将赋给  $P_0$ , 否则, 将移到下一点进行判断, 直到检查完为止。

由于这样逐点进行绘制是非常耗时的, 因此本文用纹理映射进行绘制, 使绘制速度有非常大的改善, 但绘制的速度需取决于前景物体的多少和图片的大小。在进行纹理映射时, 如果简单地把整块纹理贴到构建的空间去, 则纹理将产生很大的变形, 其解决方法将在第 3 节中详述。

## 3 用小三角面片绘制

在进行纹理映射时, OpenGL 将自动对纹理坐标进行投影校正, 而本文从图片上继承的纹理则已经经过了投影变换, 如果再简单地贴到构建的空间去, 即实际上就经过了两次投影校正, 则纹理将产生很大的变形(如图 3(a)所示)。为了避免这种情况, 可按投影方向, 把每个面, 即各个蜘蛛网格形成的墙面分成许多小三角形面片来进行纹理映射, 每个墙面的小三角面片虽可数视情况而定, 但在保证变形可以接受的情况下, 应尽量减少小三角面片数, 不然会影响绘制速度。另外, 当纹理图片较大时, 用适当数量的小三角形面片映射, 不但可以避免变形, 而且所映射的结果图像更为清晰(如图 3(c)、图 3(e)所示)。

在分成若干小三角面片进行纹理映射时, 各个墙面作为单独纹理进行映射, 虽然各墙面的小三角面片是连续地进行纹理映射绘制的, 但墙与墙之间的纹理连续性会受影响, 即直接进行纹理映射会出现切痕(如图 3(d)所示)。

本文采用 Texture-Border 的思想<sup>[11]</sup>来按比例调整墙的纹理片坐标值, 即调整纹理片的长度和宽度(原纹理长度和宽度为  $w$  和  $h$ , 调整后为  $\frac{w-2}{w} \times w$  和

$\frac{h-2}{h} \times h$ ), 以保证墙的纹理片的边界区域只参与

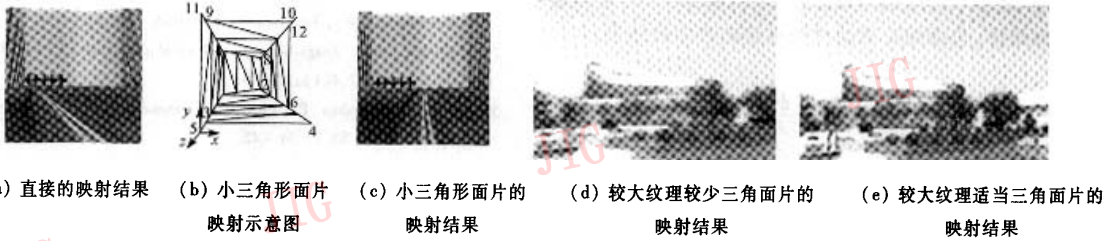


图3 用小三角形面片进行纹理映射示意图

Fig.3 Small block texture mapping

OpenGL 双线性插值运算而不显示。图3(e)就是用此方法进行绘制的,图片中间的切痕明显消失,效果很好。

### 4 实验与讨论

实验采用的机器为PIII850,内存为256M,显卡为GeForce256,32M显存,绘制508×424pixels的图片绘制速度可以达到70帧/s。通过对单灭点、双灭点、多灭点和灭线等各种类型的图片试验证明,本文算法具有鲁棒性好、适应广的特点,可应用于各种实拍图片、绘画等类型图片的实时漫游。

图4是本文工作与前人工作的比较。图4(a)是包含双灭点的图片,文献[4]的方法只能处理单灭点的情况,无法处理图4(a)中的情形。图4(b)是采用文献[5]的方法对图4(a)进行3维构建,显然场景中的马路变形严重。图4(a)显示了本文算法的背景处理方法,图4(c)是本文方法构建的3维场景,由该图可以看出,变形很小,并具有较好的视觉效果。



(a) 用网格勾勒出各个面 (b) Fleishman 方法的绘制结果 (c) 本文方法的绘制结果

图4 本文方法与Fleishman等人方法的比较

Fig.4 Comparison between the model of Fleishman et al. and our model

图5(a)是包含灭线的典型图片,文献[4]的方法无法进行漫游处理,而本文算法却可以很方便地处理此类图片。处理时,在图片上先勾勒出背景的后墙和地板,使双灭点形成的灭线处在后墙的适当



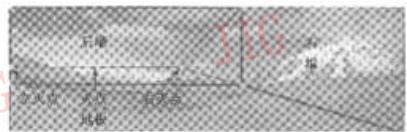
(a) 原始照片,建立网格 (b) 漫游效果

图5 本文方法处理有灭线图片的效果

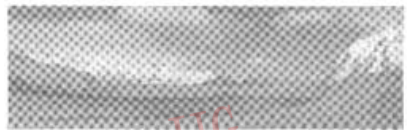
Fig.5 Image with multiple vanishing points

位置,以便把后墙放到场景中适当位置,然后进行纹理映射。白线框是对前景舰艇的勾勒。图5(b)是各种视点的漫游效果图,由该图可见,质量很高。

图6是本文算法对只包含后墙、地板和右墙的照片处理效果。该照片无法用原始蜘蛛网格和Kang等的灭线方法处理,但用本文方法却可交互构建出后



(a) 原始照片,建立网格



(b) 漫游效果1



(c) 漫游效果2

图6 本文方法处理含有灭线和右墙图片的效果

Fig.6 Image with multiple vanishing points and right wall

墙、地板和右墙的网格。图 6(b)和图 6(c)是漫游效果图,包括视点移动和视点旋转产生的结果。

## 5 结论与进一步工作

基于图像的漫游是计算机视觉领域、计算机图形学领域的研究热点之一,本文提出的双灭点松弛蜘蛛网格算法可以有效地解决一般照片的漫游问题,其既可以处理单灭点图片,又可以解决双灭点或多灭点的情况,并且能够在网格中形成灭线来代替灭点,用于处理包含灭线的图片,此外,本文算法更便于用户根据图片的特点建立各种简化模型,以实现有效的图片漫游,并表现出很强的灵活性。

本文尽管采用了分片绘制的策略来提高场景的绘制质量,但由于采用单张照片进行场景构建,信息有限,当移近视点时,场景仍会变得模糊不清,因此应采用高分辨率图片和对漫游范围进行限制等方法来解决。

进一步的工作包括以下几个方面:

(1) 在图片漫游时,引入动态背景纹理,以获得更佳的漫游效果;

(2) 采用多分辨率图像,扩大漫游范围,以提高绘制质量;

(3) 提高前景建模逼真度,以便从多幅照片中提出前景信息;

(4) 对全景图进行建模和漫游;

(5) 对视频进行处理,以便获得更大范围的漫游效果。

### 参考文献 (References)

1 Chen S. E. QuickTime VR-An image-based approach to virtual

environment navigation [A]. In: Proceedings of ACM SIGGRAPH '95 Conference [C], Los Angeles, CA, USA, 1995: 29 ~ 38.

2 Michitaka Hirose. Image-based virtual world generation [J]. IEEE Multimedia. 1997, 4(1): 27 ~ 33.

3 Levoy M, Hanrahan P. Light field rendering [J]. Computer Graphics, 1996, 30(3): 31 ~ 42.

4 Horry Youichi, Anjyo Ken-ichi, Arai Kiyoshi. Tour into picture: using a spidery mesh interface to make animation from a single image [A]. In: Proceeding of ACM SIGGRAPH '97 [C], Los Angeles, CA, USA, 1997: 225 ~ 232.

5 Fleishman S, Chen B, Kaufman A, et al. Navigating through sparse views [A]. In: Proceedings of Symposium on Virtual Reality Software and Technology '99 [C], London England, 1999: 82 ~ 87.

6 Li N, Huang Z. Tour into the picture revisited [A]. In: WSCG '2001 Conference [C], Plzen, Czech Republic, 2001: 41 ~ 48.

7 Chu Siu-Hang. Animating chinese landscape paintings and panorama [博士学位论文] [D]. Hong Kong: The Hong Kong University of Science and Technology, 2001.

8 Kang H W, Pyo S H, Anjyo K, et al. Tour into the picture using a vanishing line and its extension to panoramic images [A]. In: Eurographics 2001 [C], Manchester, England, 2001: 3 ~ 12.

9 Zhang Y C, Liu Xue-hui, Wu Eng-hua. Image-based real time walk through [J]. Journal of Software, 2002, 13(9): 1796 ~ 1803. [张严辞, 刘学慧, 吴恩华. 基于图像的实时漫游 [J]. 软件学报, 2002, 13(9): 1796 ~ 1803.]

10 Zhang L, Dugas-Phocion G, Samson J-S, et al. Single view modeling of free-form scenes [J]. Journal of Visualization and Computer Animation, 2002, 13(4): 225 ~ 235.

11 Segal M, Akeley K. The OpenGL graphics system: A specification (version 1.2.1) [M]. Los Angeles, CA, USA: Mountain View, April, 1999.