

基于改进分形算法和位移纹理映射的仿真“竹”的实现

罗 燕¹ 吴中福¹ 郭选昌²

(重庆大学计算机学院 重庆 400044)¹ (重庆大学艺术学院 重庆 400044)²

摘 要 植物仿真是计算机图形学中的一个重要研究课题。针对研究者少有涉及的仿真“竹”建模问题作了一些探索,根据“竹”是单轴分枝植物的特点,对分形算法的 L-系统建模方法作了相应的改进,并阐述了在对“竹”建模中所运用的各种相关技术和真实感表现方法等。特别是文中所引用的位移映射技术解决了“竹节”模拟的难点。所提方法可以逼真地模拟出“竹”在自然界中的形态及其特质,这在植物的仿真方面给出了一种新的解决思路。

关键词 L-系统,单轴分枝,位移映射,纹理映射,仿真“竹”

“Bamboo” Simulation Based on Improvement Fractal Algorithm and Displacement and Texture Mapping

LUO Yan¹ WU Zhong-fu¹ GUO Xuan-chang²

(Department of Computer, Chongqing University, Chongqing 400044, China)¹

(Department of Art, Chongqing University, Chongqing 400044, China)²

Abstract Plants simulation is a important studies project in the computer graphics. We discussed the “Bamboo” simulation modeling which is involved in very few researchers. We improved the L-system algorithm based on the “Bamboo” single-axis plant characteristics, and explained the relevant different technicals and methods used in “Bamboo” modeling and the realistic performanc. We quoted the displacement texture mapping technique to solve the difficult problem simulating the section header of “bamboo”. Through this method, you can simulate realistic “Bamboo” form and characteristics in the nature. This paper gave a new solution in plant simulation.

Keywords L-system, Single-axis, Displacement mapping, Texture mapping, Simulation bamboo

1 引言

20 世纪 70 年代,法国数学家 Mandelbrot 创立了分形几何学^[1],分形(Fractal)一词用以描述那些不规则而欧氏几何又无法描述的几何现象和物体,且分形现象广泛地存在于我们身边。分形几何学以其独特的手段来解决整体与部分的关系问题,利用空间结构的对称性和自相似性,并采用各种模拟真实图形的模型,使生成的整个景物细节呈现出无穷回归的性质,且丰富多彩,具有奇妙的艺术魅力。在所生成的景物中,可以有结构性较强的树、山峰,也可以是结构性较弱的火、云及烟等。

分形几何学的核心思想是物体形状整体和局部的自相似性,这个概念与自然界中大量不规则事物的形状特征是吻合的,因此分形学适合于描述自然界中的不规则物体。借助于分形算法的计算机生成,即从少量的数据生成复杂的自然景物图形,我们在仿真模拟方面前进了一大步。植物形态作为自然界中最常见的景观之一,其复杂性也适合用分形理论进行模拟。分形理论和技术提出后,在世界上引起了广泛重视,在数学、物理、化学、生物、经济学、计算机科学、艺术等领域广泛地展开了对它及其应用的研究,逐渐发展和完善成为一个理论体系。

生成分形图形的关键是要有一个合适的模型来描述对

象,根据所选分形模型的不同,产生分形图形的方法可大致分为如下 4 类:基于 L-系统的分形图形;迭代函数系统 IFS 方法;随机插值法;粒子系统模型法。

植物的仿真是计算机图形学中的一个重要研究课题,到目前为止有很多介绍构造和绘制植物的方法,本文针对研究者少有涉及的仿真“竹”建模问题作了一些探索,根据“竹”是单轴分枝植物的特点,对分形算法的 L-系统建模方法作了相应的改进,并详细阐述了在对“竹”建模中运用的各种技术及真实感表现方法等,再运用 Bezier 曲线实现“竹”主干因受重力影响而弯曲的模拟,并结合扫描曲面造型方法实现“竹”主干圆柱体的弯曲,特别是应用位移映射技术解决了“竹节”模拟难点问题。最后运用纹理映射实现了“竹”枝干和叶的渲染。采用本文方法,可以逼真地模拟出“竹”在自然界中的形态,这在植物的仿真方面给出了一种新的解决思路。

2 分形的 L-系统及其改进

2.1 分形的 L-系统基本原理

1968 年,Arvid Lindenmayer 发明了一个用来对多细胞生命体进行模拟和建模的规则,随后被命名为 L 系统^[2]。按 L-系统理论,可以将植物分裂的过程近似地视为一个递归、迭代的过程,并进而将一种植物对应一个迭代函数,通过改变其中的某些参数来模拟植物的生长过程。1984 年 A. R.

Smith^[3,4]等人率先将L-系统引入计算机图形学中,立即显示了L-系统在计算机模拟方面的巨大潜力,为计算机模拟植物形态的真实感图形提供了有力的工具。

经过发展,L-系统现在已经成为植物形态模拟的主要方法之一,它通过形式化语言来描述植物的拓扑结构,并将语言中特定的字符与特定的植物结构相对应,通过文法生成单词字符串来表达植物的结构。而单词的生成过程则可以用来表达植物的生长发育过程。它的多种表达式,能够模拟出不同情况下的植物形态。L-系统在描述植物的拓扑结构时,具有定义简单、结构化程度高、易于实现等优点。在计算机图形学中,可视化模型通常都是用场景图来描述的,这些图形则是由各种基本的几何形状例如线、三角形、圆柱面组合通过移位、旋转而成的。而L-系统的字符串经过解析可以产生场景图,然后再用这些场景图来展示二维三维的模型。

本文对仿真“竹”的研究就是建立在该方法基础之上的。

2.2 分形的L-系统算法生成的三维植物流程图

图1给出了三维L-系统模拟算法的流程图。根据L-系统自身的特点,提出了用3个参数控制植物的生成模型,这是一个适用于各种植物建模的原理性L-系统建模流程图,其具体内容如下:

(1)植株根部半径。该参数主要控制植株生长范围,即控制整个植株的大小。需要说明的是该参数指植株茎的根部半径。

(2)半径衰减因子。该参数主要控制植株粗细变化的趋势,它和植株根部半径一起控制整个植株分枝的数量。需要说明的是该参数控制植物分枝茎底部的半径。

(3)高度衰减因子。该参数主要控制植株生长的高度,即控制生成单元长度依次递减的速度。由于局限于微机的性能,因此本文在植物三维模型满足自身特性的前提下对结构作了适当简化,例如对“竹”等具有多级分枝结构的植物仅采用了三级分枝结构,而叶为第四级分枝结构。

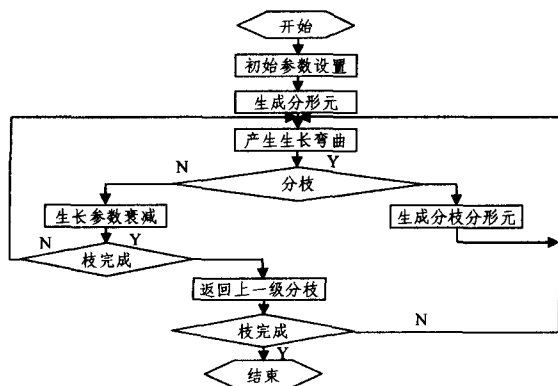


图1 三维L-系统模拟算法流程图

2.3 用分形的L-系统对单轴分枝植物建模的方法

任意一棵树木都是由若干树枝和树叶组成的,每一条树枝又由若干分枝组成。因此分枝和树叶是树木最基本的组成单元。虽然植物在外部形态上呈现出千姿百态,但是植物的分枝模式总是有规律可寻的,它们的外部形态是其内部某种形态控制机制的反映,而外部形态是可以由植物的分枝模式来描述的。植物的分枝模式一般分为图2所示的两种,于是只要研究出这两种分枝的建模方法便可以模拟出不同植物的形态^[5]。

(1)单轴分枝模式。顶芽不断向上生长,形成主干。同时

侧芽也发展成为侧枝,侧枝又以同样方式形成次级侧枝,但侧枝不及主干粗、长。这种分枝模式有明显的主轴,称为单轴分枝模式,其整个植物形成挺拔向上的形态,因此也称为总状分枝。单轴分枝的主干上能产生各级分枝,主干的伸长和粗细比侧枝强得多。因此,这种分枝模式,主干显著,如竹子、松树、云杉、杨树等都属此类。

(2)合轴分枝模式。顶芽发育一定时期后死亡或生长缓慢,而位于顶芽下面的侧枝就取而代之,继续发育,形成强壮的侧枝,连接在原来的主轴上。以后,这种侧枝上的顶芽又停止发育,再由它下面的侧芽来代替,便形成了弯曲的主轴。以这种方式分枝植物的地上部分呈开放状态,典型的有柳树、榆树等。



图2 植物分枝形态模式

分析完植物的单轴分枝特点后,从单轴分枝植物的整体外形特征出发,对原有的分形L-系统建模算法进行改进,便得到单轴分枝植物的建模算法描述如下:

输入:单轴分枝植物分形L-系统初始参数;

输出:单轴分枝植物三维形态;

Step1 设置初始值,将第一个分形元P存入链表; //设置单轴分枝植物L-系统的初始参数;

Step2 If L-系统的初始参数为空 Then break; Else

Step3 End If; //判断是否满足L-系统的循环条件;

Step4 将第一个分形元P绕X轴旋转 α 度,Create(P); //生成一个新的节点P1;

Step5 For($i=1; i \leq n; i++$) {If P1点绕Y轴旋转 β 度 Then Create(P1)}; //生成n个新节点 $\{P_1, P_2, \dots, P_n\}$;

Step6 $\{P_1, P_2, \dots, P_n\}$ 绕Z轴旋转 γ 度; //生成的n个新节点绕Z轴旋转 γ 度;

Step7 If n个新节点绕Z轴旋转结束 Then MoveUp(Δ); Step3; Else Step5 End If //沿Z轴向上生长 Δ 长度,生成一段植物枝干。

3 分形L-系统对“竹”的建模及其弯曲形态的实现

3.1 “竹”的外形特点及L-系统建模

“竹”最主要的特点是有有一根竖直向上的主干,称为第一级枝干。“竹”的第二级枝干是从主干分出的。从“竹”的形态分析它的第二级枝干可知,从主干分出的侧枝上的分枝都较细小,主干与分枝直径的比值很大。然而,二级枝干和主干有一点是相同的,即二级枝干从形态上看仍然具有单轴分枝的特点。图3给出了“竹”的分枝形态示意图,这里排除了重力环境等对它产生影响的因素,仅单纯分析“竹”的分枝形态。

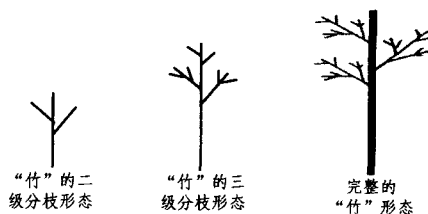


图3 分析“竹”的分枝形态示意图

以上“竹”的建模算法用伪代码表示如下:

```
CreateConifer(int level,float height,float  
radius)
```

```
{  
    vector<CNode> LstFractal; //存储分形元  
    vector<CNode> LstConifer; //存储节点  
    初始化分形元;  
    for(int i=0;i<level;i++){  
        for(判断是否遍历完 LstFractal 中的所有点){绕 x 轴旋转  
            角度  $\alpha$  ( $\alpha$  服从均值为给定值, 方差为  $\pi/45$  的正态分  
            布);  
            在上次旋转基础上绕 y 轴旋转角度  $\beta$  ( $\beta$  服从均值为  $\beta +$   
             $\pi/3$ , 方差为  $\pi/45$  的正态分布); 在上次旋转基础上绕 z  
            轴旋转角度  $\gamma$  ( $\gamma$  服从均值为  $\gamma + 2\pi/3$ , 方差为  $\pi/45$  的  
            正态分布);  
            if(随机数判断该层是否有第 n 个分枝){  
                旋转任意角度;  
            } // end if  
        } // end for  
        沿 z 轴向上生长, 形成一段“竹”的枝干;  
        更新分形元;  
    } // end for  
} // end
```

图 4 是改进并加入随机因子后所得的“竹”形态。

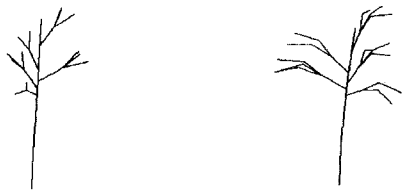


图 4 改进并加入随机因子后所得的“竹”形态

3.2 用 Bezier“致曲”技术实现“竹”主干弯曲的模拟

现实世界中的植物枝干形态各异,而笔直的枝干是很少见的。竹笋刚出土时,可视为笔直向上生长,但长成为成竹后,因受到重力等因素的影响,主干会产生弯曲。因此,在对“竹”模拟时,竹的主干需要引入某种“致曲”技术。而竹枝的弯曲则是在节头处生长的角度不同而产生的,是一种折线弯曲,于是可在建模过程中运用随机角度生长法解决分枝的折线弯曲问题,在这里就不赘述了。

“竹”的主干需要引入“致曲”技术,而几何图形中的曲线种类很多,但不是所有的曲线都能适合植物生长的形态。在植物弯曲形态建模中引入的“致曲”技术需要满足一定的条件。例如,必须便于计算机实现,形态易于控制等等。因此,综合考虑,本文引入了 Bezier“致曲”技术。

Bezier“致曲”技术是 1962 年法国雷诺汽车公司工程师构造的一种以逼近为基础的参数曲线和曲面生成的设计方法^[6,7]。该方法将函数逼近与几何表示结合起来,使得设计师在计算机上就像使用作图工具一样得心应手。用 Bezier“致曲”技术来模拟植物的枝干时,只需给出枝干上的几个点就可以了。图 5 为 Bezier“致曲”技术实例及“竹”主干在引入该方法后实现的弯曲效果模拟。

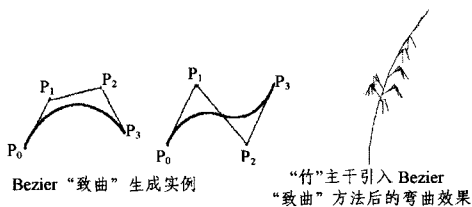


图 5 “竹”主干引入 Bezier“致曲”方法后实现的弯曲效果模拟

4 “竹”形态的模拟

4.1 用圆柱体模拟“竹”的枝干

前面简单介绍了 Beizer“致曲”方法及其在植物模拟中的应用。但是,现实世界的植物是具有三维形态的。仅简单地用线条来表现枝干显然还远远不够。通常的做法是选择一种和植物枝干很相似的几何体来代替线条。于是很自然会想到使用圆柱体或者圆台来模拟植物枝干。但是,使用标准的几何体模拟植物枝干比较生硬,没有真实感,如图 6 所示。

现实世界中的植物枝干因受到外界诸多因素的影响,极少情况是笔直的。因此,需要找到一种方法使模拟出的几何形体可以具有更多变化。本文采用扫描曲面造型方法来实现“竹”主干的弯曲。

目前,扫描曲面造型方法^[8]由于其灵活性而成为曲面造型的基本工具。且其基本思想十分简单:一个在空间移动的几何集可扫描出一个实体,再根据扫描剖面线和基准线的不同可形成许多复杂的曲面。扫描曲面造型技术应用于树枝弯曲形态的模拟,一方面使树枝几何形状更逼近真实,另一方面还解决了主干与枝条之间的光滑连接问题,从整体效果上增强了图形的真实感。

因此,我们用前面介绍的 Bezier“致曲”法作为导引线,用一条封闭的几何曲线做截面线,再利用这些曲线生成的圆柱体来模拟“竹”的枝干,从而生成三维立体的植物,如图 7 所示。图中的(a)是控制点,(b)为(a)的 3 个控制点生成的 Bezier 曲线,(c)为选取曲线上间隔均匀的 4 个点,(d)为在(c)中选取的点处构造截面并连成曲柱的结果。

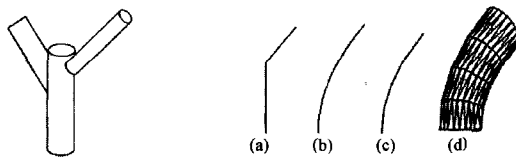


图 6 用圆柱体模拟植物枝干 图 7 曲柱体的生成过程示意图

用圆柱体代替曲线之后基本上就可以生成一棵稍有立体感的“竹”形态了。但是,还需要在“竹”的生成过程中加入衰减因子,这是因为不仅每个枝干从根部到末梢有衰减,而且每级枝干之间也存在衰减。为了绘制出“竹”分枝处的连续性,将每个枝干根部的半径取为上一级枝干的末梢半径。且因“竹”末梢分枝很细小,还要在这层分枝上生成竹叶,所以,用曲线或直线来绘制这一层分枝。

4.2 用位移映射解决“竹节”形态模拟问题

在三维“竹”的真实感建模中,需要解决的另一个难点问题是在“竹节”的生成问题。“竹”区别于其他植物的一个很重要的元素,就是“竹节”。

利用传统的造型方法,比如用多边形和曲面来生成“竹节”的形状非常不方便。存储一棵“竹”或一丛“竹”所有“竹节”的凹凸细节需要很大的内存开销,且绘制效率低,因此,我们采用位移映射技术来生成“竹节”,因为它与纹理映射类似,只要记录一个或者几个“竹节”就可以映射到大量不同的几何面上,从而产生复杂的几何细节,同时位移映射可以方便地采用包围盒进行宏观的形状控制。结果显示:这种方法能够在较少内存和计算开销的情况下绘制出较高真实度的“竹节”。

由 Cook 提出的位移映射(displacement mapping)技术,能够产生非常复杂的几何细节,而不需要对这些细节逐个进行生成^[9]。它支持用户用位移映射或移位绘制器来生成几何

曲面的细节,并添加移位细节^[10,11]。这也是位移映射和凹凸纹理的根本区别。两者很相似,区别在于位移映射是能添加几何细节,而凹凸纹理则只是对映射曲面不同部位的法向量做扰动,即只对平面产生凹凸效果,不能影响几何结构。位移映射技术的特征是只在绘制场景时,几何细节才被加上去,因此,可以自适应地根据具体位置选择可见几何细节的精细程度,以减小场景绘制的几何复杂度。

位移映射通常支持两种绘制技术^[12],一是用微面重构场景,再用扫描线填充的绘制技术;二是光线跟踪绘制技术。我们采用光线跟踪技术,场景中的其它竹节可以同时进行统一绘制。光线跟踪绘制位移映射也有两种技术:(1)利用几何微面重构场景由位移映射进行绘制,因为它的内存开销太大,而且庞大的数据量会使绘制效率大大降低,我们不采用这种方法来绘制;(2)亦即是我们采用的方法,其不重构场景,只是在绘制时把移位信息加到几何面上去。

显然,位移映射技术很适合用来构造“竹节”。其原因在于:首先,在最后实现“竹”的三维模型时,需要对“竹”模型的细节进行有效的生成,因“竹节”比较突出,采用凹凸纹理不能满足要求;其次是,一棵“竹”或一丛“竹”中的竹干与竹枝包含许多竹节,其复杂度使得很难逐一地生成每个“竹节”的细节,需要用位移映射工具来减少数据量,还能保证有足够精致的几何细节。

竹节的生成过程:先构造一个圆柱体,通过实体数据采样,确定“竹节”的位置,并由此计算出位移映射的包围盒,最后生成所需要的竹节,图8是利用位移映射生成竹节的过程示意图及实现的效果。

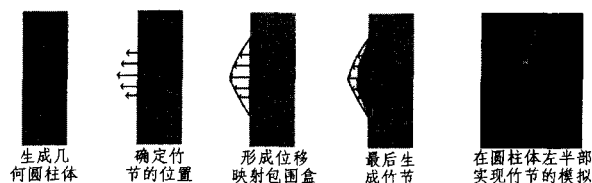


图8 利用位移映射生成竹节过程的示意图

5 用纹理映射实现“竹”枝干和叶的渲染

5.1 “竹”枝干的渲染

纹理映射是增强真实感的手段。而且用 OpenGL 实现纹理贴图也十分方便。对“竹”的枝干采用这种渲染方式,就是把一幅二维竹皮纹理图片包裹到表现枝干的曲面上以达到增强真实感的效果。它实质上就是把一幅图片包裹在指定的表面上。特别需要提及的是,我们在已经生成的“竹”三维模型中,运用纹理寻址的原理,使三维竹节的数据采集与二维竹皮纹理中的数据采集保持位置一致,就可以实现竹节的三维实体与竹节纹理的准确结合,达到仿真的最佳效果。

用 OpenGL 实现纹理贴图要完成以下 4 个步骤^[13]:(1)加载纹理图片,通常采用的函数为 `auxDIBImageLoad`,但是这个函数只能加载 `bmp` 格式的图片,如果在比较高的 VC 版本上可以使用 `Cimage` 中的 `load` 函数来加载,它可以加载 5 种常见的图片格式;(2)控制滤波,由于纹理图片的大小不可能和需要贴图的位置完全吻合,因此需要滤波。在 OpenGL 中提供了 `GL_NEAREST`, `GL_LINEAR` 两种滤波方式;(3)说明映射方式;(4)给出顶点的纹理坐标和几何坐标,绘制出场景。以上是最简单的纹理贴图,实质上就是把一幅竹皮的图

片包裹在指定的表面上。

5.2 “竹”叶的渲染

绘制植物叶子是增强植物真实感非常重要的一环。区别植物的方法都是对植物叶进行比较。因此,如果能为绘制好的植物枝干添加不同类的叶子,就可以绘制出不同种类的植物。而且,在同一形态结构的枝干上绘制不同形状的植物叶子也会使植物显现出不同的姿态。

常见的添加植物叶的方法有两种:第一种是将植物叶的外形用多边形表示,将叶子简化成一个填充多边形,即绘制出近似于叶片形状的几何图形;第二种是利用纹理贴图,也就是在植物叶的节点处贴一张植物叶的图片来表达。

我们采用 OpenGL 标准的纹理贴图方法绘制竹叶。首先,将竹叶纹理图片读入缓冲区,然后将竹叶纹理图片绑定,在需要贴纹理的地方将纹理坐标和位置坐标一一对应起来,便可把一幅图片覆盖到给定的任何一个位置上了。为了产生不同大小和各种形态的竹叶,应使之具有多样的变化。随机选取竹叶的大小、叶片表面的法向量,即可将组成每一片竹叶的多边形绕 x 轴、 z 轴转一个随机角度、缩放随机倍数加以实现。

通过以上渲染就可以相当逼真地表现竹叶的自然形态,表达出竹叶特有潇洒洒脱的韵致。

6 实验结果

最后,在 MAYA 平台上进行“竹”最后的三维模型生成,MAYA 平台提供了最基本的技术支持构架,如 L -系统等,同时它也是一个开放式的构架,我们可根据需要对其进行扩展,添加自己所需的内容,如改进的 L -系统、竹节、竹皮、竹叶等的生成和渲染方法等等,生成的仿真“竹”效果如图9,图10所示。仿真“竹”可应用于植物学研究、影视特技、游戏开发、虚拟现实系统等多个领域。



图9 生成的仿真“竹”丛外观效果图



图10 仿真“竹”的近距离细部效果图

参考文献

- [1] Whittaker E T, Watson G N. A course of modern analysis[M]. Cambridge University Press, 1952
- [2] Lindenmayer A. Mathematical models for cellular interactions in development i&ii[J]. Journal of Theoretical Biology, 1968
- [3] Bell A D, Roberts D, Smith A. Branching patterns in the simulation of plant architecture[J]. Journal of Theoretical Biology, 1979, 81: 351-375
- [4] Smith A R. Plants fractals and formal languages[J]. Computer

- [5] Prusinkiewicz P, Lindenmeyer A, et al. The Algorithmic Beauty of Plants [M]. Springer-Verlag, 1990
- [6] 王辉. 基于 L 系统的虚拟园林观赏树木生长建模研究[D]. 长沙: 中南林业科技大学, 2006: 7-9
- [7] Hearn D, Baker M P. 计算机图形学[M]. 蔡士杰, 吴春熔, 孙正兴, 译. 电子工业出版社, 2002
- [8] Farm G. Curves and Surfaces for Computer Aided Geometric Design: A Practical Guide, 4th Edition [M]. Boston: Academic Press, 1996
- [9] 李钢, 刘华明. 基于 NURBS 的扫描曲面造型方法的研究[J]. 机械研究与应用, 2000, 13(3): 5-6
- [10] Cook R L. Shade trees[J]. Computer Graphics, 1984, 18(3):

- [11] Hanrahan P, Lawson J. A language for shading and lighting calculations[J]. Computer Graphics, 1990, 24(4): 289-298
- [12] Pharr M, Hanrahan P. Geometry caching for ray-tracing displacement maps[C]//Proc. of the 7th Eurographics Workshop on Rendering. Porto, Portugal, 1996: 31-40
- [13] Schaufler G, Priglinger M. Efficient displacement mapping by image warping[C]//Proc of the 10th Eurographics Workshop on Rendering. Granada, Spain, Eurographics, 1999: 175-186
- [14] 付恺, 李春霞, 杨克俭, 等. Bump Mapping 原理及在 OpenGL 下的实现[J]. 交通与计算机, 2004, 22(2): 54-57
- [15] 杨刚. 全景图拼接算法的设计与实现[J]. 重庆工学院学报: 自然科学版, 2007, 21(19): 107-110

(上接第 247 页)

结束语 F 矩阵是许多计算机视觉应用中的重要参数, 其计算的准确性决定了后续的处理步骤能否成功。本文将常见的 F 矩阵估计方法划分为线性法、非线性法和鲁棒法 3 大类, 共计 11 种, 通过仿真数据和真实图像实验对各自的性能进行了评估。实际应用中, 由于视觉系统所采用的图像特征定位及匹配方法各有差异, 可根据具体情况合理选择本文介绍的 F 矩阵计算方法, 以实现高精度的 3D 信息获取。

参考文献

- [1] 马颂德, 张正友. 计算机视觉-计算理论与算法基础[M]. 北京: 科学出版社, 1998
- [2] Luong Q-T, Faugeras O D. The fundamental matrix: Theory, algorithms and stability analysis[J]. International Journal of Computer Vision, 1996, 1(17): 43-76
- [3] Longuet-Higgins H C. A computer algorithm for reconstructing a scene from two projections[J]. Nature, 1981, 293: 133-135
- [4] Mester R. On the mathematical structure of direction and motion estimation[C]//3rd International Symposium on Physics in Signal and Image Processing (PSIP). Grenoble, France, January 2003
- [5] Pollefeys M, et al. Self-calibration and metric reconstruction in spite of varying and unknown internal camera parameters[J]. International Journal of Computer Vision, 1999, 32(1): 7-20
- [6] Zhang Z, Deriche R, Faugeras O, et al. A robust technique for matching two uncalibrated images[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 1995, 82(2): 1129-1139
- [7] Hartley R, Gupta R, Chang T. Stereo from uncalibrated cameras [C]//Proceedings of the International Conference on Computer Vision and Pattern Recognition. IEEE, Urbana Champaign, IL
- [8] Mohr R, Boufama B, Brand P. Accurate projective reconstruction [C]//Mundy J, Zisserman A, eds. Applications of Invariance in Computer Vision. Vol. 825 of Lecture Notes in Computer Science, Berlin: Springer-Verlag, 1993: 257-276
- [9] Zhang Z. A new multistage approach to motion and structure estimation: From essential parameters to Euclidean motion via fundamental matrix[R]. 2910. INRIA Sophia-Antipolis, France
- [10] Maybank S J, Faugeras O. A theory of self-calibration of a moving camera[J]. International Journal of Computer Vision, 1992, 8(2): 123-151

- [11] Faugeras D, Luong Q T, Maybank S J. Camera self-calibration: theory and experiment[C]//Proceedings of the European Conference on Computer Vision. Santa Margherita L., 1992: 321-334
- [12] Lourakis M I A, Deriche R. Camera self-calibration using the singular value decomposition of the fundamental matrix: From point correspondences to 3D measurements[R]. 3748. INRIA Sophia-Antipolis, August 1999
- [13] Lourakis M I A, Deriche R. Camera self-calibration using the Kruppa equations and the SVD of the fundamental matrix: The case of varying intrinsic parameters [R]. 3911. INRIA Sophia-Antipolis, Mars, 2000
- [14] Hartley R I, Zisserman A. Multiple view geometry in computer vision[M]. Cambridge University Press, 2000
- [15] Hartley R I. In defense of the 8-point algorithm[C]//Proceedings of the 5th Proc. International Conference on Computer Vision. Boston, MA: IEEE Computer Society Press, 1995: 1064-1070
- [16] Faugeras Q, Mourrain B. On the geometry and algebra of the point and line correspondences between n images[C]//Proceedings of the 5th Proc. International Conference on Computer Vision. Boston, MA: IEEE Computer Society Press, 1995: 951-956
- [17] Zhang Z, Deriche R, Faugeras O, et al. A robust technique for matching two uncalibrated images through the recovery of the unknown epipolar geometry[J]. Artificial Intelligence Journal, 1995, 78(10): 87-119
- [18] Zhang Z. Determining the epipolar geometry and its uncertainty: A review[J]. International Journal of Computer Vision, 1998, 27(2): 161-195
- [19] Zhang Z. Parameter Estimation Techniques: A tutorial with application to conic fitting[J]. Image and Vision Computing Journal, 1997, 15(1): 59-76
- [20] Fischler M A, Bolles R C. Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography[J]. Comm. Assoc. Comp. Mach., 1981, 24(6): 381-395
- [21] Torr P H S. Motion segmentation and outlier detection [D]. Dept. of Engineering Science, University of Oxford, 1995
- [22] Zuliani M, Kenney C S, Manjunath B S. The MultiRANSAC algorithm and its application to detect planar homographies[C]//IEEE International Conference on Image Processing. September 2005