

文章编号: 1005-0523(2008)04-0038-05

基于 IDL 语言的虚拟场景的重建和漫游

尹燕

(华东交通大学 信息工程学院, 江西 南昌 330013)

摘要: 基于图像的建模和绘制方法与基于传统的几何绘制方法相比有很多优点,但是在场景几何参数未知的情况下,要实现单幅图片和图像序列的漫游,难度很大. 论文对虚拟场景重建的目的和重要性以及 TIP 技术的原理作了介绍. 在 TIP 技术的基础上,利用 IDL 语言根据透视投影关系和运动路径,为场景加入了一个近似的三维信息,并将前后图像的模型根据它们的位置关系串连起来实现了一种未知运动路径情况下的虚拟场景重建和漫游.

关键词: TIP; IDL; 虚拟场景; 重建; 漫游

中图分类号: TP391.9

文献标识码: A

虚拟场景重建的主要目的就是为实现在一个场景中的穿梭和漫游,这种场景漫游包括改变视点和视线方向,以便获取与 3 维场景所对应的 2 维平面图像. 近年来,虚拟场景重建技术受到越来越多的关注,它已成为计算机图形学、计算机视觉和图像处理等领域的热门研究课题. 随着经济和科学技术的发展,这种技术必将得到更加广泛的应用. 近几年发展起来的基于图像的建模和绘制技术越来越受图像界的关注. 基于图像的绘制方法直接从已知的图像中插值或重投影获得中间视点的图像. 和传统的基于几何的绘制方法相比,基于图像的建模和绘制方法建模更快、更方便,在保证高绘制速度的同时,可以获得高度真实感的图像.^[1-2]

TIP 技术实现了从一幅图片或一幅画到场景的重建和漫游,论文在 TIP 技术的基础上利用 IDL 语言实现了虚拟场景的重建和漫游.

1 TIP 技术

TIP (tour into a picture, 场景漫游) 技术^[3]最早由 Horry 在 1997 年的 SIGGRAPH 会议上提出来的. 它的主要思想是先从图像中恢复场景的伪 3 维模型,然后在其中漫游.

TIP 方法的整个流程如图 1 所示. TIP 将整个场

景模型分为前景物体模型和背景模型两个部分. 为了从图像中交互获得场景的模型, Horry 提出了一种类似于蛛网的交互建模方法. 用户只需交互指定模型的内矩形和消失点,就可以计算出模型的其它部分. 这样,背景模型就被分为 5 个部分,分别对应 3 维模型中的 5 个平面,它们是上平面(又称天花板)、下平面(又称地面)、左平面和右平面以及背面. 当背景模型确定后,前景物体的模型也可以交互获得,并且每一个前景物体可以看作是背景中的一个多边形,而该多边形相对各个背景平面的位置则可以通过计算得出.

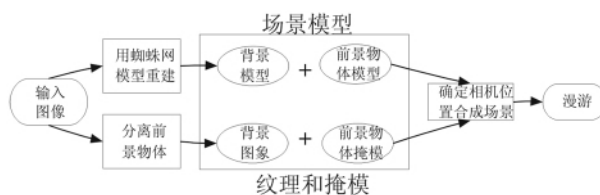


图1 原 TIP 技术流程图

利用 IDL (Interactive Data Language) 语言在图像处理上的强大功能并基于 TIP 技术实现了虚拟场景的重建和漫游.

2 基于 IDL 语言的虚拟场景重建

2.1 虚拟场景的重建原理

原理框图如图 2 所示

收稿日期: 2008-01-11

作者简介: 尹燕(1978-),女,湖北黄梅人,讲师,硕士,研究方向为图象处理,数据挖掘.

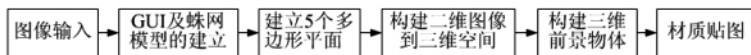


图2 虚拟场景的重建原理框图

采用的图像如图3示,是一幅 JPEG 格式、经过 PHOTOSHOP 处理的名为“Palace”的图像。



图3 图像输入

为了建立三维场景需要建立 GUI(Graphical User Interface), 需要用户在图像中提供两个点指定模型的内矩形和消失点, 用于建立蛛网模型。根据建立蛛网模型的原理, 利用 IDL 语言实现上述各个模块, 需要用到三个函数来得到内矩形、消失点及外矩形, 函数结构如下:

```
FUNCTION function - name, input - 1, input - 2,
...
...
return, output
END
```

最后利用“plot”及“/device”可以得到蛛网模型。

2.2 建立5个多边形平面及构建二维图像到三维空间

建立5个多边形平面及构建二维图像到三维空间是虚拟场景中的关键步骤。

(1) 建立5个多边形平面

建立蛛网模型是为了建立有六个平面三维立体面, 在本设计中只需5个面: 上平面(又称天花板)、下平面(又称地面)、左平面和右平面以及背面。

从实际图中可以看出, 外矩形的拐角并不是二维图像的拐角; 为了建立完整的蛛网模型需要通过加上环境边缘(左、右、上和下边缘), 对图像进行伸展。因此, 需要重新定义图像的边缘及更新新图像的数据, 再重新定义五个平面。利用 IDL 语言中的 xline 和 if 条件语句可以重新选择数据, 得到新的图像模型。

(2) 构建二维图像到三维空间

在二维图像中产生了一些点, 基于这些点, 可以计算出在三维世界中的相应点。三维位置计算的算法如下描述。如图4示, “C”相机的光源中心, “h”是光源中心到地面的距离, 二维图像可以看作是其中一个平面图, 所以, P_1 、 P_2 和 P_3 是图像中的三个

点; 从光源出发的一束光线经二维图像投影到三维世界中(如 P_1 到 P_1'), 但是 P_1 的确切位置不知道。图4中的直线方程如下式:

$$L(\lambda) = C + \lambda(P - C) \quad (1)$$

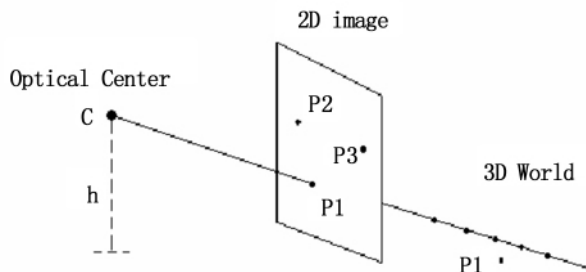


图4 三维位置计算

式(1)中, 如果 λ 值确定, 就可以得到 P' 。假设 P 是二维图像中的点坐标值为 (x, y) , $C = [0 \ h \ 0]$, 式(1)可以写成:

$$L(\lambda) = \begin{bmatrix} 0 \\ h \\ 0 \end{bmatrix} + \lambda \begin{bmatrix} x - vx \\ y - vy - h \\ f \end{bmatrix} = \begin{bmatrix} x(\lambda) \\ y(\lambda) \\ z(\lambda) \end{bmatrix} \quad (2)$$

式(2)中, (vx, vy) 是消失点的坐标, f 是焦距。对于下平面, $y = 0$, 所以 $y(\lambda) = 0$, 由式(2)得到二维图像中的点 P 投影到三维世界的位置为 $(x(\lambda), y(\lambda), z(\lambda))$ 。但是, 其他平面并没有 $x(\lambda), y(\lambda), z(\lambda)$ 中的任何一个值, 在三维世界中的位置计算稍微复杂一些, 解决方法是:

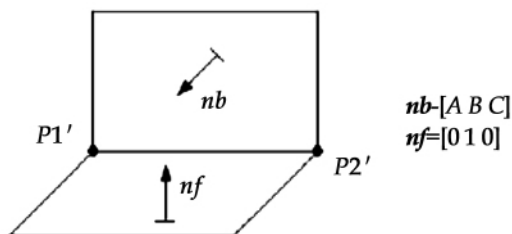


图5 平面法线

- (1) 首先计算出平面的方程
- (2) 应用式(1)计算三维位置

如图5示,对于背面,与下平面正交,所以它们的法线存在 $nf^*nb=0$ 的关系, nf 是下平面的法线, nb 是背面的法线. 图中 $P1'$ 和 $P2'$ 是已经计算出来的三维空间中的点,背面上的点的有下面的等式:

$$nb^T \cdot P1' + d = 0 \quad (3)$$

在式(3)中,假设 $d=1$,至少需要两个已知点 $P1'$ 和 $P2'$ 计算 $nb = [A \ B \ C]$,所以联立下式:

$$\begin{cases} nb^T \cdot p1' + 1 = 0 \\ nb^T \cdot p2' + 1 = 0 \\ nf \cdot nb^T = 0 \end{cases} \quad (4)$$

可以得到:

$$nb^T \cdot \begin{bmatrix} p1' \\ p2' \\ nf \end{bmatrix} = - \begin{bmatrix} 1 \\ 1 \\ 0 \end{bmatrix}, \text{所以 } nb^T = \begin{bmatrix} p1' \\ p2' \\ nf \end{bmatrix}^{-1} \cdot \begin{bmatrix} -1 \\ -1 \\ 0 \end{bmatrix} \quad (5)$$

得到 $nb = [A \ B \ C]$,利用平面表达式 $Ax + By + Cz + d = 0$,由平面中的点 (x, y, z) 及 $d=1$,可以到背面的方程,再利用式(2)得到 λ 值后可以得到相应点的三维坐标.

和背面相似,其他平面的方程也可以得到,就能计算出二维图像中的点的三维坐标.

在 IDL 语言中,三维坐标计算实现如下:

```
FUNCTION I2W,dx,dy,vx,vy,h,f,w
  px = w##( dx - vx)
  py = h + w##( dy - vy - h)
  pz = w##f
  p = [px,py,pz]
  return p
END
```

上述语句中,定义函数名为 I2W,用于计算二维图像中的点在三维空间中的坐标,语句中: (dx, dy) 是输入图像中点的坐标, (vx, vy) 是消失点的坐标, h 是光源中心到地面的距离, f 是焦距, w 是 λ .

利用上述函数,定义各平面中的点的三维坐标,可以得到三维背景模型.

- (3) 构建三维前景物体

提取三维前景物体关键步骤同样是三维位置的计算及 IDL 实现.

因为前景物体 leo 是处在三维空间中,需要知道其位置,因此,需要计算出其在三维空间中的坐标,利用两个函数“I2W”和“normal”来实现,得到三维前景物体的构建模型图. $o0$ 和 $o1$ 是前景物体在下平面中的点位置, $o2$ 和 $o3$ 是前景物体在背面中的点位置. 利用 IDL 语言得到前景物体.

2.3 材质贴图

上述步骤中,已经完成了构建三维空间的所有数据,可以进行三维空间的构建和材质贴图. 根据图像对象树步骤,在 IDL 语言中利用“WIDGET_BASE”、“WIDGET_DRAW”、“WIDGET_CONTROL”和“WIDGET_BASE”, $oPoly \rightarrow SetProperty$, obj_new 和 $oMODEL$ 命令可以得到初始的三维场景.

3 纹理的纠正与映射技术及漫游

3.1 纹理的纠正与映射技术

漫游的过程实际就是对二维图像进行角度的转化以及对图中局部图像的放大或缩小,就会导致近景纹理表示的非常清晰,而远景纹理越靠近,就越模糊. 这种模糊的纹理是导致漫游图片模糊的主要原因,而且当视点靠近场景背面的时候,图像会模糊到无法接受的程度. 这也是 Horry 算法的主要局限.

纹理映射技术是用来增强场景真实感,同时提高图形绘制速率最有效的方法. 其基本原理是把体现场景模型的表面属性和局部细节信息的图像映射到所建立的几何模型的表面,来表示模型的表面的属性和细节. 从而避免了对物体的表面属性和局部细节进行繁琐的建模,以及在绘制时的繁琐计算.

为了把射影重建提升到度量重建或欧氏重建,可以把三维视觉分为四个几何空间层次射影空间、仿射空间、度量空间和欧氏空间. 把三维视觉分层有利于执行重建. Faugera 给出了如何利用把射影几何升级至仿射几何,以及把仿射几何升级至度量几何与欧氏几何来完成重建^[4].

二维图像平面上的纠正单应性矩阵 H ,是一个 3×3 阶的可逆矩阵,把一图像平面上的点 x 变换为 x' 可表示为:

$$x' = Hx \quad (6)$$

其中

$$H = \begin{bmatrix} p_{11} & p_{12} & p_{13} \\ p_{21} & p_{22} & p_{23} \\ p_{31} & p_{32} & p_{33} \end{bmatrix} \quad (7)$$

其中 p_{33} 可以取值为 1, H 包含 8 个自由度,且可以分解为以下形式:

$$H = MN \quad (8)$$

其中, M 是单应性矩阵的度量部分,它是一个相似变换. 可以写成以下形式:

$$M = \begin{bmatrix} sR & t \\ 0^T & 1 \end{bmatrix} \quad (9)$$

其中 R 是旋转矩阵, t 是平移向量, s 是各向同性的比例因子, $0^T = (0, 0)$. 此时 M 含有个 4 自由度. 剩下的 4 个自由度包含在 H 矩阵的非度量部分

N 中. 这里可以把 N 认为是把 l_∞ , I 和 J (灭线和两个圆环点) 映射到规范值. 即把 l_∞ 映射到 $(0, 0, 1)^T$, 把 I 和 J 映射到 $(l, \pm I, 0)^T$. 这里的圆环点是指灭线 l_∞ 与绝对二次曲线在图像平面上的影像的交点. 而 N 又可以分为仿射和射影两部分如图 6 示:

$$N = H_a H_p \quad (10)$$



图6 射影变换和仿射变换示意图

根据 H_p 把 l_∞ 映射到 $(0, 0, 1)^T$, 如果把 H_p 写作 $(p_1, p_2, p_3)^T$, 把 l_∞ 写作 $(l_{\infty 1}, l_{\infty 2}, l_{\infty 3})^T$ 则可以得到:

$$l_\infty = H_p^T (0, 0, 1)^T = [p_1 p_2 p_3] \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} = p_3 \quad (11)$$

所以, 可以通过灭线的表达式来求得:

$$H_p = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ l_{\infty 1} & l_{\infty 2} & l_{\infty 3} \end{bmatrix} \quad (12)$$

因为 l_∞ 是一个齐次向量, 所以 H_p 包含了两个

自由度. 余下的两个自由度包含在 H_a 中. 而 H_a 是将 $H_p I$ 映射到 $(1, i, 0)^T$, 其中 $H_p I = (\alpha - i\beta, 1, 0)^T$. 将实部和虚部分别计算可以得到以下结果:

$$H_a = \begin{bmatrix} \frac{1}{\beta} & -\frac{\alpha}{\beta} & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (13)$$

其中, α 和 β 是圆环点的实部与虚部, 也是最后两个自由度.

纹理的纠正与映射过程总结如下:

- (1) 利用该平面的两个灭点使用 $l_\infty = u \times v$ 计算出灭线
- (2) 根据公式(11) 计算出矩阵 H_p ;
- (3) 由计算出来的 H_p , 把该平面上的纹理图案进行射影变换;
- (4) 将仿射纹理图案变换成为边长为 2^n 正方形纹理图案;
- (5) 利用纹理映射函数, 将纹理映射到已重建好的简单的三维场景模型上去, 使它具有相片般真实感.

分别对其他可见平面进行上述操作, 就可以恢复整个模型的视觉外观.

3.2 漫游

应用上面的理论给出了漫游的几个示例, 如图 7 所示.

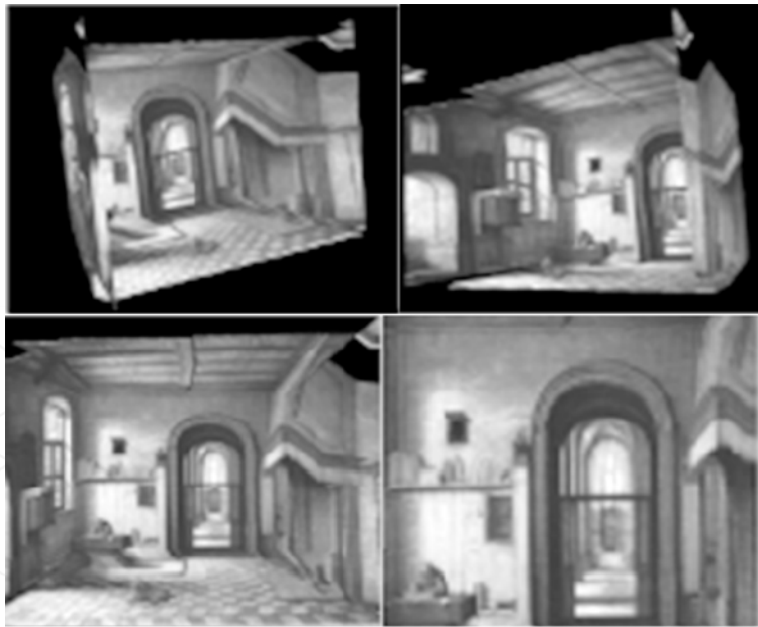


图7 漫游示例

4 结束语

在 TIP 技术的基础上,提出了一种未知运动路径情况下利用 IDL 语言实现基于图像序列的场景重建和漫游. 由于受图像序列视角范围的限制,不可能从图片中获得场景的三维信息,论文根据透视投影关系和运动路径,为场景加入了一个近似的三维信息,并将前后图像的模型根据它们的位置关系串连起来,就实现了在图像序列中的漫游.

通过对几个图片的测试,基本能完成漫游. 但是需要用户在图像中提供两个点指定模型的内矩形和消失点,用于建立蛛网模型. 所以,场景的建模还需要人工交互,尚不能全自动完成,这些都有待于进一步的研究.

参考文献:

- [1] 吴琼玉等. 基于图像序列的虚拟场景重建和漫游 [J]. 中国图象图形学报, 2006, 11(1): 113 - 118.
- [2] 吴琼玉等. 一种基于 TIP 技术的场景重建和漫游方法 [J]. 计算机工程与科学, 2006, 28(4): 45 - 47.
- [3] Shenchang Eric Chen. QuickTime VR—An Image - based Approach to Virtual Environment Navigation [C]. Los Angeles, CA: ACM Press. 1995. 29 - 38.
- [4] Olivier Faugeras. Stratification of 3 - D vision: projective, affine, and metric representations [J]. Journal of the Optical Society of America. 1995. 12(3): 465 - 484.
- [5] 韩培友. IDL 可视化分析与应用 [M]. 西安: 西北工业大学出版社, 2006.
- [6] 闫殿武. IDL 可视化工具入门与提高 [M]. 北京: 机械工业出版社, 2003.

Construction and Wandering of Virtual Environment Based IDL Language

YIN Yan

(School of Information Engineering, East China Jiaotong University, Nanchang 330013, China)

Abstract: Compared with the traditional geometric approach, the approach of image - based modeling and rendering has many advantages. However, it is hard to implement modeling from one single picture if geometric parameter is not known. The purpose and importance of construction of virtual environment and theorem of TIP technique is introduced in this paper. Based on TIP technique, by means of IDL, perspective projection and moving path, the paper realizes the construction and its wandering of virtual environment with the unknown path.

Key words: TIP; IDL; virtual environment; construction; wandering

(责任编辑: 李 萍)