

# 基于图像的实时漫游\*

张严辞<sup>1</sup>, 刘学慧<sup>1</sup>, 吴恩华<sup>1,2</sup>

<sup>1</sup>(中国科学院 软件研究所 计算机科学重点实验室,北京 100080);

<sup>2</sup>(澳门大学 科技学院,澳门)

E-mail: yanci\_zhang@hotmail.com

http://www.ios.ac.cn

**摘要:** 提出了一个点和多边形模型混合的场景表达方式,从而实现了复杂场景的视点不受限制的实时漫游.它从多幅带深度的参考图像出发,在预处理阶段区别对待场景中平面物体和曲面物体在参考图像中的对应像素.对于在参考图像所占区域较大的空间平面,用传统的多边形模型方式对其进行表达,试图恢复出其平面方程,然后通过采样密度比较和重采样过程,将该空间平面在所有参考图像中的出现合并成一个均匀采样的纹理图像;而对于空间曲面或在参考图像中所占面积较小的平面,使用点的形式对其进行表达,通过采样密度的比较去掉冗余的点,将保留下来的点按其空间位置进行聚类.同时,对于场景中那些不被所有参考图像所拍摄到而在漫游过程中可能形成空洞的部分,提出一个空洞预填补技术,在预处理阶段即对这类空洞进行填补,从而大大减少了漫游阶段出现空洞的几率.在漫游阶段则使用纹理映射和点 Warping 的方式进行绘制,以充分利用图形硬件的加速功能.

**关键词:** 基于图像的绘制;实时漫游;采样密度;纹理映射

中图法分类号: TP391 文献标识码: A

在高度真实感的虚拟环境中进行实时漫游一直是图形界的研究热点,其问题的关键之一在于如何有效地表达一个场景,以便在漫游阶段可以实时并且高质量地进行绘制.对于以多边形模型来表达场景的传统图形学来说,虽然现在的图形显示卡的能力得到了极大的发展,但仍然不能满足对一个多边形数量巨大的场景进行实时绘制的要求.近年来提出的基于图像的建模和绘制技术,试图通过场景的全光采样<sup>[1-6]</sup>或者参考图像的重投影<sup>[7-10]</sup>实现具有高度真实感的虚拟场景的绘制.但这些方法还存在如下缺点:

- (1) 表达一个场景所需数据量过于巨大(如 Light Field 等<sup>[2-4]</sup>,需要巨大的存储量);
- (2) 在漫游过程中视点位置受限(如 LDI<sup>[8]</sup>限制新视点必须在 LDI 视点附近,同心平图<sup>[6]</sup>限制新视点必须在一个圆形区域内);
- (3) 对场景有一定的限制(如全景拼图<sup>[5]</sup>限制场景中不能有障碍物);
- (4) 生成新视点下的图像存在不同程度的走样(如存在垂直方向上的走样,甚至错误<sup>[5,6]</sup>).

对基于图像的建模和绘制方法来说,用图像和几何模型对场景进行混合表达应该是一种合理、有效的表达方式<sup>[11]</sup>,而混合表达方式的关键在于如何在图像和几何模型之间达到恰当的折衷.本文提出了一种点和多边形模型混合的虚拟场景表达方法,试图以图像和几何模型混合的方式实现对复杂虚拟场景的视点不受任何限制的实时漫游.我们从多幅带深度信息的参考图像出发,在预处理阶段为场景中平面物体和曲面物体在参考图像

\* 收稿日期: 2002-02-26; 修改日期: 2002-07-08

基金项目: 国家自然科学基金资助项目(69873044);澳门大学基金资助项目(RG009/99-00S/WEH/FST)

作者简介: 张严辞(1975 - ),男,四川成都人,博士生,主要研究领域为基于图像的绘制,虚拟现实;刘学慧(1968 - ),女,湖南长沙人,博士,副研究员,主要研究领域为计算机图形学,虚拟现实;吴恩华(1947 - ),男,江苏南通人,博士,研究员,博士生导师,主要研究领域为计算机图形学,可视化,虚拟现实.

中的对应像素建立不同模型,在绘制阶段对不同模型实施不同的绘制机制.

本文提出的算法的主要贡献是:

- (1) 将场景用点和面的方式进行混合表达;
- (2) 提出一个采样密度比较的算法,剔除了参考图像中对场景采样密度低的像素,从而去掉了多幅图像间的冗余信息,减小了表达一个场景所需的数据量;
- (3) 提出了一个空洞预填补算法,在预处理阶段即对那些不出现在所有参考图像中,但在漫游过程可能形成的空洞进行填补.

本文第 1 节介绍一些与本文相关的工作.第 2 节~第 6 节介绍我们提出的算法的预处理过程,分别讨论参考图像中对应空间平面和曲面的像素的区分、像素间的采样密度的比较、空间平面对应的纹理图像的生成、纹理图像中空洞的预填补以及空间曲面对应像素点的聚类处理.第 7 节介绍绘制过程.第 8 节给出我们的漫游系统中的一些图像和试验数据.第 9 节对算法进行总结.

## 1 相关工作

### 1.1 全光采样

描述虚拟场景的所有光流信息的全景函数是一个七维函数<sup>[12]</sup>.但对于静态的场景来说,五维的全景函数<sup>[11]</sup>就足以描述了.即便如此,想完全表达这个五维函数也是极其困难的,因此,往往利用一些约束条件,将局部虚拟场景的光流场简化为更低维的函数.比如,Light Field 和 Lumigraph<sup>[2~4]</sup>通过描述一个无遮挡立方体包围盒内的光场,将五维的全景函数简化为四维;同心拼图<sup>[6]</sup>通过以同心圆切线方向的数据组织将无遮挡同心圆区域的光流场函数简化为 3D;全景拼图<sup>[5]</sup>可以看做是同心拼图的推广,它在一个没有障碍物的空间用全方位摄像机沿着构成非规则网格的路径进行图像捕获,然后通过网络上前、后两个摄像机捕获到的参考图像来生成网格内新视点下的图像.

### 1.2 3D Warping方程和LDI

McMillan 提出了一个基于像素深度的 3D Warping 算法<sup>[7]</sup>.它通过将参考图像中的像素逐个 Warping 到目标视点下来生成新视点图像.LDI<sup>[8]</sup>通过在同一个像素位置上保存位于该视线方向上的多个空间点的深度值,将虚拟场景统一在同一个视点下,从而保持了只 Warping 一幅参考图像的简单性.LDI Tree<sup>[10]</sup>建立在 LDI 的基础上,它把多幅参考图像组织成层次结构,并保留了各幅参考图像不同采样密度的采样,在绘制时则根据当前视点和视线在 LDI Tree 中选择合适的 LDI 来进行绘制.而文献<sup>[9]</sup>则尝试在一个场景中建立多个视点下的 LDI 从而达到对场景的虚拟漫游.

### 1.3 VDTM

VDTM<sup>[13]</sup>在已知场景多边形模型的基础上,通过一个可见性判定的算法来决定生成新视点图像时所需的参考图像,然后使用 Projective 纹理映射来进行绘制.与其他 IBR 方法相比,其最大的不同是,VDTM 使用纹理映射而不是点 Warping 或寻找对应光线的方式进行绘制,充分利用了图形卡的加速功能,从而提高了绘制效率.

## 2 参考图像像素的分类

为了区别对待参考图像中对应空间平面和曲面的像素,我们首先需要分别标记出空间中平面物体和曲面物体在参考图像中对应的像素点.

我们使用边缘检测算子在参考图像的深度空间进行边缘检测.该过程实际上是一个图像分割过程.对每个分割出来的像素区域  $A$ ,我们用最小二乘法来构造一个平面来拟合区域  $A$  中像素点对应的 3D 点.为了提高平面最小二乘拟合的可信度,我们规定,只有当区域  $A$  中像素点的数目大于预定阈值时,我们才拟合其区域内的点.对于拟合出的空间平面,我们通过如下两个公式来衡量拟合误差:

$$D_{\max} = \max_{(x,y) \in A} (D_{(x,y)}(P)), \quad D_{\text{avg}} = \frac{\sum_{(x,y) \in A} D_{(x,y)}(P)}{\text{numPnts}}$$

其中  $D_{(x,y)}(P)$  表示像素  $(x,y)$  对应的 3D 点到重建平面  $P$  的距离,  $\text{numPnts}$  表示像素区域  $A$  中的像素数目。

若  $D_{\max}$  和  $D_{\text{avg}}$  足够小, 则认为区域  $A$  中的像素确实对应于某个 3D 平面, 否则认为区域  $A$  中的像素不对应于 3D 空间平面。

在本阶段结束以后, 参考图像中所有的像素被分为两类:

(1) 成功重建的像素, 即那些 3D 空间对应点位于某个空间平面上的像素;



Fig.1 Reference image and its pixels classification result  
图 1 参考图像及其像素分类结果

(2) 未能成功的像素, 其中包括其 3D 空间对应点位于某个空间曲面的像素以及虽然其 3D 空间对应点位于某空间平面上, 但该平面在参考图像上所占的像素区域过小的像素。

图 1 是一幅参考图像及其平面重建的结果, 其中重建结果图像中白色的部分代表没有成功重建的像素, 而其他不同的灰度则分别代表不同的重建平面。

### 3 采样密度的比较

一般来说, 场景中的任意物体都会出现在多幅参考图像中, 而且不同的参考图像对其有不同的采样密度. 相对于空间中的同一部分, 那些采样密度低的像素就是参考图像间的冗余信息. 而采样密度比较的目的就在于去掉那些对空间同一部分采样密度较低的像素, 只保留采样密度最好的像素。

我们使用 Jacobi 矩阵的对应行列式来衡量每个像素对空间的采样密度<sup>[14]</sup>. 该行列式被广泛应用于衡量某个变换所引起的局部面积变化。

#### 3.1 比较重建平面间的采样密度

在进行采样密度比较之前, 我们必须首先知道来自于不同参考图像的哪些重建平面对应于空间中同一个平面. 通过比较重建平面的方程、法向量以及它们在空间中的位置, 我们可以容易地获得这个对应关系。

假设像素  $p$  和像素  $q$  来自于不同参考图像, 但它们对应于同一个 3D 点  $(X,Y,Z)$ . 设它们的像素坐标为  $(s_p, t_p), (s_q, t_q)$ .  $p, q$  的窗口坐标  $(x_p, y_p, z_p, 1), (x_q, y_q, z_q, 1)$  存在如下关系:

$$\begin{bmatrix} wx_q \\ wy_q \\ wz_q \\ w \end{bmatrix} = \begin{bmatrix} m_0x_p + m_4y_p + m_8z_p + m_{12} \\ m_1x_p + m_5y_p + m_9z_p + m_{13} \\ m_2x_p + m_6y_p + m_{10}z_p + m_{14} \\ m_3x_p + m_7y_p + m_{11}z_p + m_{15} \end{bmatrix} \quad (1)$$

其中,  $M = M_q M_p^{-1} = \begin{bmatrix} m_0 & m_4 & m_8 & m_{12} \\ m_1 & m_5 & m_9 & m_{13} \\ m_2 & m_6 & m_{10} & m_{14} \\ m_3 & m_7 & m_{11} & m_{15} \end{bmatrix}$ ,  $M_p$  和  $M_q$  分别是参考图像从空间坐标到窗口坐标的变换。

设像素  $p$  的空间对应点位于某个重建平面上, 该重建平面的方程系数设为  $(A,B,C,D)$ , 则  $z_p$  可以写为

$$z_p = -\frac{ax_p + by_p + d}{c} \quad (2)$$

其中

$$\begin{aligned} a &= m'_0A + m'_1B + m'_2C + m'_3D, & b &= m'_4A + m'_5B + m'_6C + m'_7D, \\ c &= m'_8A + m'_9B + m'_{10}C + m'_{11}D, & d &= m'_{12}A + m'_{13}B + m'_{14}C + m'_{15}D, \end{aligned}$$

$$M_p^{-1} = \begin{bmatrix} m'_0 & m'_4 & m'_8 & m'_{12} \\ m'_1 & m'_5 & m'_9 & m'_{13} \\ m'_2 & m'_6 & m'_{10} & m'_{14} \\ m'_3 & m'_7 & m'_{11} & m'_{15} \end{bmatrix}.$$

假设所有参考图像的长、宽相同,则它们具有相同的从窗口坐标到像素坐标的变换,因此, Jacobi 矩阵可以写成

$$J = \begin{pmatrix} \frac{\partial s_q}{\partial s_p} & \frac{\partial s_q}{\partial t_p} \\ \frac{\partial t_q}{\partial s_p} & \frac{\partial t_q}{\partial t_p} \end{pmatrix} = \begin{pmatrix} \frac{\partial x_q}{\partial x_p} & \frac{\partial x_q}{\partial y_p} \\ \frac{\partial y_q}{\partial x_p} & \frac{\partial y_q}{\partial y_p} \end{pmatrix} = \begin{pmatrix} \frac{\partial}{\partial x_p} \left( \frac{wx_q}{w} \right) & \frac{\partial}{\partial y_p} \left( \frac{wx_q}{w} \right) \\ \frac{\partial}{\partial x_p} \left( \frac{wy_q}{w} \right) & \frac{\partial}{\partial y_p} \left( \frac{wy_q}{w} \right) \end{pmatrix}. \quad (3)$$

根据式(3),我们就可以比较  $p, q$  两点对空间平面同一区域的采样密度.若  $|J| > 1$ ,则认为  $q$  点比  $p$  点有较好的采样密度,因此将  $p$  点标记为无用点;否则,认为  $p$  点比  $q$  点有较好的采样密度.

### 3.2 未能成功重建像素的采样密度比较

对应空间曲面物体的像素的采样密度比较与重建平面采样密度比较的惟一不同是,由于这些像素并不位于某个重建平面上,因此我们无法获得式(2).解决的方法是利用该像素空间对应点的切平面来代替重建平面,为此,我们需要估计每个没有成功重建像素点的法向量.由于已经存在许多成熟的法向量估计算法,因此,我们直接采用算法<sup>[15]</sup>来进行没有成功重建像素的法向量估计.

## 4 纹理图像的生成

对重建平面来说,使用纹理映射而不是点 Warping 的方式来进行绘制显然效率更高,但纹理图像上的像素应该是均匀采样的,而参考图像上的像素区域都是透视投影产生的结果,即非均匀采样,因此,我们需要对这些保留下来的像素区域进行均匀的重采样,以生成可以作为纹理使用的图像.

重采样过程中需要解决的问题是决定合适的重采样密度.我们仍借助于 Jacobi 矩阵的对应行列式来计算重采样密度.对于重建平面中经采样密度比较而保留下来的每个像素,计算从该像素所在的参考图像到纹理图像的变化所对应的  $|J|$ ,这个  $|J|$  即是衡量这个变换所引起的像素对应的局部区域的面积变化率.而我们计算纹理图像的重采样密度的原则就是使所有像素对应的平均面积的变化率为 1,即

$$\frac{\sum_{i=1}^N |J_i|}{N} = 1, \quad (4)$$

其中  $N$  代表属于当前重建平面的经过采样密度比较而保留下来的所有像素的数目.

根据式(4)可以得到重采样密度  $\rho$ :

$$\rho = \sqrt{\frac{\sum_{i=1}^N \left( \frac{\partial}{\partial x_i} \left( \frac{w_i \rho u_i}{w_i} \right) \frac{\partial}{\partial y_i} \left( \frac{w_i \rho v_i}{w_i} \right) - \frac{\partial}{\partial x_i} \left( \frac{w_i \rho v_i}{w_i} \right) \frac{\partial}{\partial y_i} \left( \frac{w_i \rho u_i}{w_i} \right) \right)}{N}}.$$

## 5 空洞预填补

### 5.1 关于空洞的分析

对于复杂的场景来说,几乎不可避免地存在一些没有被所有参考图像拍摄到的部分,因此,在漫游过程中,空洞的出现几乎是不可避免的.如果我们只是选用所有参考图像中的某一些图像来生成新视点下的图像,那么导致空洞出现的原因有两种:

- (i) 空洞部分未出现在作为源图像的参考图像中,但可以通过搜索其他参考图像而被填补;

(ii) 空洞部分没有出现在所有的参考图像中.

问题的关键是,当空洞出现时,我们并不知道它的类型,而只能试图搜索其他一些可能填补空洞的参考图像.如果搜索完毕仍然不能填补空洞,则使用对空洞附近的像素进行插值等方法来填补空洞.但这种策略还存在一些缺点:

(1) 在漫游阶段进行搜索其他参考图像、插值等操作,会降低绘制的速度.

(2) 由于空洞的相邻像素并不一定与空洞像素属于同一个物体,因此,插值的结果往往不好.而且,在漫游过程中随着视点不停地变化,空洞的相邻像素也在发生变化,因此,插值的结果也不稳定.

(3) 更主要的是,如果在这些空洞后存在其他物体,那么在绘制时,这些物体会透过空洞显示出来,这时我们很难检测到空洞的出现.

实际上,通过以上的预处理过程,我们可以很自然地避开(i)类空洞.这是因为在进行了重建、采样密度的比较等一系列操作之后,我们使用的信息已不再局限于某几幅参考图像,而是来自于全部的参考图像,因此不可能产生(i)类空洞.所以,在本文提出的算法下出现的空洞必然是(ii)类空洞.对于这类空洞,我们提出了一个在预处理阶段就对重建平面上的空洞进行填补的算法.由于此时的填补工作是在单独的重建平面上进行的,不会受到场景中其他物体的影响,因此,其填补质量肯定会好于绘制阶段的插值填补.

## 5.2 空洞的区分和填补

重建平面中的空洞并不一定是由于没有被所有参考图像拍摄到而产生的.例如,图 2(b)就是从若干幅参考图像中恢复出来的对应于图 2(a)中的桌面的纹理,其中黑色部分表示空洞,但是这些空洞并不都属于重建平面.从图 2(c)中可以看到,纹理左上角区域的空洞区域并不属于重建平面,因此不能被填补,否则会错误地遮挡其后面的物体.

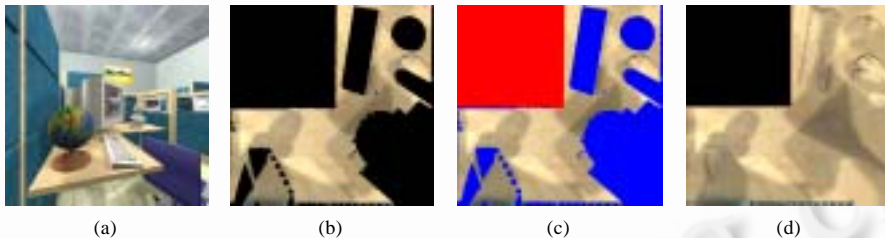


Fig.2 Different holes in the texture reconstructed  
图 2 重建纹理中不同的空洞

空洞区分的基本思想是,如果重建平面上的洞是属于重建平面本身所具有的洞,那么至少在某一幅参考图像中,我们可以通过这个洞看到洞后面的物体;如果所有参考图像都无法看到这个洞之后的物体,则认为这个洞是由于没有被所有参考图像所拍摄到而产生的.根据这个原则,对于重建平面上的每个空洞像素  $p$ ,我们只需检查该像素在所有参考图像中对应位置的像素,如果这些对应位置上的像素中存在一个像素被像素  $p$  遮挡,则认为  $p$  是属于重建平面本身的空洞,否则认为它是由于拍摄所产生的空洞.

我们把空洞填补问题看做是一个受限的纹理生成过程,采用一个与文献[16]类似的算法来进行填补,图 2(d)就是空洞填补后的结果.

对空洞进行预填补的一个额外的好处是可以降低表达一个空间平面的复杂度.如果我们不进行空洞预填补,则所需表达的空间平面就可能是一个包含许多洞的形状复杂的多边形(如图 2(b)所示),而在空洞预填补之后,空间平面对应的多边形的形状大为简化(如图 2(d)所示).

## 6 未能成功重建的像素点的聚类

对于没有成功重建的像素,即使我们使用与文献[17]类似的算法恢复出它们的几何模型,也会由于这些几何模型过于复杂而不符合我们对实时漫游的要求.因此,对于这些没有成功重建的点,在绘制阶段,我们仍然按点 Warping 的方式<sup>[7,8]</sup>进行处理.为了提高绘制效率,我们需要在未知这些点的 3D 模型的情况下,尽量剔除那些

落在当前视域以外的点.

我们的方法是对这些点按其空间坐标进行聚类,然后用包围盒包裹这些聚类.而在绘制阶段,剔除那些落在视域外包围盒内的点,从而提高绘制效率.

对于聚类的原则,我们希望最后聚类的结果是每个聚类都对应于空间中一个真实的物体.这样的聚类,我们称为基于点的物体.

定义. 一个基于点的物体是指满足如下两个条件的 3D 空间点集  $S$ :

- (1) 对  $\forall p \in S$ , 必然在  $S$  中存在另外一点  $q$  满足  $D(p, q) \leq \varepsilon$ , 其中  $D(p, q)$  表示  $p, q$  间的距离,  $\varepsilon$  是某个常数;
- (2) 对  $\forall p \in S$  和  $\forall t \notin S, D(p, t) > \varepsilon$ .

显然,解决这个聚类问题最直接的方法是计算  $S$  内任意两点间的距离,然后根据这个距离直接进行聚类,但这种方法的计算量过大.我们提出了一个避免直接计算两点间的距离的方法.首先,我们构造一个包围整个点集  $S$  的包围盒,然后将这个包围盒均匀地划分为若干个大小相同的小立方体,每个立方体的边长都是  $\frac{\sqrt{2}}{4}\varepsilon$ . 我们的聚类规则是:假设某点  $p$  落在某个小立方体  $c$  内,如果另一点  $q$  也落在  $c$  内或者  $c$  的相邻小立方体内,那么我们认为  $p$  和  $q$  属于同一个基于点的物体.

根据上述规则,我们构造了一个自底向上的聚类算法.对于  $S$  中任意一点  $p$ ,假设  $p$  落在某个小立方体  $c$  内,那么:

- (1) 如果当前没有其他点落在  $c$  或者  $c$  的相邻小立方体中,那么构造一个新的基于点的物体,并把  $p$  加入到该物体中;
- (2) 如果当前已经存在一些点  $q_i$  落在  $c$  或者  $c$  的相邻小立方体中,并且所有这些都属于同一个基于点的物体  $O$ ,那么将  $p$  加入到  $O$  中;
- (3) 如果当前已经存在一些点  $q_i$  落在  $c$  或者  $c$  的相邻小立方体中,但这些点都分属于几个不同的基于点的物体,则将所有这些物体合并成一个新的物体  $O_{\text{new}}$ ,并将  $p$  加入到  $O_{\text{new}}$  中.

当聚类算法结束以后,我们对每一个基于点的物体都构造一个 OBB-Tree<sup>[18]</sup>,这些 OBB-Tree 将应用到以后的裁减算法中.

## 7 绘制

在漫游过程中,我们分别使用纹理映射和点 Warping 的方式对重建平面和基于点的物体进行绘制,其中对基于点的物体的绘制是影响绘制效率的关键因素,因此,提高对基于点的物体的绘制效率将极大地改善整个绘制程序的效率.

我们提高基于点的物体的绘制效率的主要手段在于尽量裁减掉那些在当前视点下不可见的点,其中包括两方面裁减,即对那些落在当前视域外的点和被遮挡点的裁减.

对于视域外的点的裁减,我们可以根据在预处理阶段生成的 OBB-Tree 进行.在漫游过程中,我们遍历每一棵对应基于点的物体的 OBB-Tree,如果某个 OBB 节点的 8 个顶点都落在当前视域以外,我们就认为当前 OBB 节点和所有属于当前 OBB 节点的子节点都不可见.

对遮挡点的裁减要复杂一些.由于我们并不知道这些点对应的 3D 模型,因此我们很难获得点与点之间的遮挡信息.但是对于重建平面,我们是知道其空间方程的,因此,对于那些被重建平面遮挡的点,我们还是有办法将其裁减掉的.我们使用了一个与文献[19]类似的算法来进行裁减.

## 8 实验结果

我们在一台拥有 256M 内存,32M 显存,GeForce2 Pro 显卡的 AMD 1GHz PC 上执行本文的算法.参考图像和目标图像的分辨率都是 400×400,每个像素的颜色用 24bit 表示.我们共使用了 88 幅参考图像来描述一个场景.在漫游过程中,我们对漫游者的视点和实现方向都没有任何限制.如图 3 所示为一些在漫游过程中生成的



图像.

由于在漫游过程中的不同位置,我们的裁减算法将裁减掉不同数目的未能成功重建的点,因此,我们很难获得算法的绘制时间,但是在最坏情况下,我们都可以保证 12fps 的绘制速度.同时,在我们的算法下,我们可以更有效地对一个场景进行表达,对 88 幅参考图像来说,需要大约 93.9M 的空间来保存颜色和深度信息,经过我们的预处理算法之后,我们只需 20.6M 的空间来保存所提取出的 220 个重建平面的纹理信息和 16.8M 的空间来保存未能成功重建的点的信息.最后,我们总共只需要 37.4M 的空间来表达相同的场景,这只是原始保存空间的 39% 左右.



Fig.3 Output images in the walkthrough

图 3 漫游过程中生成的图像

## 9 结 论

基于图像的绘制中一个很重要的问题就是如何有效地表达一个场景.本文提出的算法区别对待空间中平面和曲面所对应的像素.对于空间平面,我们对其进行重建,并从所有参考图像中提取出对应于该平面的像素,通过采样密度的比较,去掉参考图像间的冗余信息,然后通过重采样将采样密度比较过程中保留下来的像素转化为纹理图像,这样,在绘制阶段就可以利用图形卡的加速功能.对于空间曲面对应的像素,我们并不试图去重建其几何模型,而仍然用点的方式对其进行表达,同样地,采样密度的比较去掉了冗余的点,然后通过按空间位置进行聚类来加速对其绘制.同时,对于空间重建平面,我们通过预处理阶段的空洞预填补,大大降低了漫游过程中出现空洞的几率.正是由于我们合理、有效地表达了场景,使得我们在漫游过程中对视点位置和视线方向没有任何限制,同时在绘制速度上也能达到实时要求.

## References:

- [1] McMillan, L., Bishop, G. Plenoptic modeling: an image-based rendering system. In: Mair, S.G., Cook, R., ed. SIGGRAPH'95 Proceedings. New York: ACM Press, 1995. 39~46.
- [2] Levoy, M., Hanrahan, P. Light field rendering. In: Fujii, J., ed. SIGGRAPH'96 Proceedings. New York: ACM Press, 1996. 31~42.
- [3] Gortler, S.J., Grzeszczuk, R., Szeliski, R., *et al.* The lumigraph. In: Fujii, J., ed. SIGGRAPH'96 Proceedings. New York: ACM Press, 1996. 43~54.
- [4] Buehler, C., Bosse, M., McMillan, L., *et al.* In: Pockock, L., ed. SIGGRAPH 2001 Proceedings. New York: ACM Press, 2001. 425~432.
- [5] Aliaga, D.G., Garlborn, I. Plenoptic stitching: a scalable method for reconstructing 3D interactive walkthroughs. In: Pockock, L., ed. SIGGRAPH 2001 Proceedings. New York: ACM Press, 2001. 443~450.
- [6] Shum, H.Y., He, L.W. Rendering with concentric mosaics. In: Waggenspack, W., ed. Proceedings of the SIGGRAPH'99. New York: ACM Press, 1999. 299~306.
- [7] McMillan, L. An image-based approach to three-dimensional computer graphics [Ph.D. Thesis]. University of North Carolina, 1997.
- [8] Shade, J., Gortler, S.J., He, L., *et al.* Layered depth images. In: Cunningham, S., Bransford, W., Cohen, M.F., ed. SIGGRAPH'98 Proceedings. New York: ACM Press, 1998. 231~242.
- [9] Popescu, V., Lastra, A., Aliaga, D., *et al.* Efficient warping for architectural walkthroughs using layered depth images. In: Proceedings of the IEEE Visualization'98. 1998. 211~215.

- [10] Chang, C.F., Bishop, G., Lastra, A. LDI tree: a hierarchical representation for image-based rendering. In: Waggenspack, W., ed. Proceedings of SIGGRAPH'99. New York: ACM Press, 1999. 291~298.
- [11] Shum, H.Y., Kang, S.B. A review of image-based rendering techniques. In: IEEE/SPIE Visual Communications and Image Processing (VCIP) 2000. 2~13.
- [12] Adelson, E.H., Bergen, J.R. The plenoptic function and the elements of early vision. In: Landy, M., Movshon, J.A., eds. Computational Models of Visual Processing. Cambridge, MA: MIT Press, 1991. 3~20.
- [13] Debevec, P.E., Yu, Y.Z., Borshukov, G.D. Efficient view-dependent image-based rendering with projective texture-mapping. In: Eurographics Rendering Workshop 1998.
- [14] Coorg, S., Teller, S. Real-Time occlusion culling for models with large occluders. In: Proceedings of the 1997 Symposium on Interactive 3D Graphics. 1997. 83~90.
- [15] Yagel, R., Cohen, D., Kaufman, A. Normal estimation in 3D discrete space. The Visual Computer, 1992,8(5-6):278~291.
- [16] Wei, L.Y., Levoy, M. Fast texture synthesis using tree-structured vector quantization. In: Brown, J.R., Akeley, K., ed. SIGGRAPH 2000 Proceedings. New York: ACM Press, 2000. 479~488.
- [17] Hoppe, H., DeRose, T., Duchamp, T., *et al.* Surface Reconstruction from Unorganized Points. In: Thomas, J.J., ed. SIGGRAPH'92 Proceedings. New York: ACM Press, 1992. 71~78.
- [18] Gottschalk, S., Lin, M.C., Manocha, D. OBB tree: a hierarchical structure for rapid interference detection. In: Fujii, J., ed. SIGGRAPH'96 Proceedings. New Orleans, LS: ACM Press, 1996. 171~180.
- [19] Herkbert, P.S. Fundamentals of texture mapping and image warping [MS. Thesis]. Berkeley: University of California, 1989.

## Image-Based Real Time Walkthrough\*

ZHANG Yan-ci<sup>1</sup>, LIU Xue-hui<sup>1</sup>, WU En-hua<sup>1,2</sup>

<sup>1</sup>(Key Laboratory of Computer Science, Institute of Software, The Chinese Academy of Sciences, Beijing 100080, China);

<sup>2</sup>(Faculty of Science and Technology, University of Macau, Macau, China)

E-mail: yanci\_zhang@hotmail.com

<http://www.ios.ac.cn>

**Abstract:** A hybrid representation of environment models is proposed by combining points and polygons through which a real time walkthrough of a complex environment can be achieved. Starting from multiple depth reference images, the pixels of images are classified into two categories, corresponding to the planar and non-planar surfaces in 3D space. For the pixels corresponding to the planar surfaces, the plane coefficients are reconstructed and all their appearances in the reference images are merged to form uniform-sampled texture images by a comparison of sampling rate and resampling. For the pixels corresponding to the non-planar surfaces, their point representation is kept and no 3D models are reconstructed, the redundant pixels are removed by sampling rate comparison. The remained pixels are organized by OBB-tree according to their space coordinates. At the same time, the holes that are unable to be captured by all the reference images are pre-filled in the preprocessing phase so that the probability of hole appearance in walkthrough is greatly reduced. Under this hybrid representation, texture mapping and point warping are employed to render the novel views, to take full advantages of the acceleration utility of graphics hardware.

**Key words:** image-based rendering; real-time walkthrough; sampling rate; texture mapping

\* Received February 26, 2002; accepted July 8, 2002

Supported by the National Natural Science Foundation of China under Grant No.69873044; the Research Grant of the University of Macau of China under Grant No.RG009/99-00S/WEH/FST