

# 新虚拟场景模型下智能漫游算法设计与实现<sup>①</sup>

于丽莉 戴青 (解放军信息工程大学 电子技术学院 河南 郑州 450004)

**摘要:** 基于对相关智能预测算法的分析,发现现有算法并未考虑到虚拟场景节点间存在的潜在关联关系,降低了算法的预测准确率。针对此问题,在一种新的立方体虚拟场景模型的基础上,提出了一种智能漫游算法,通过为每个节点建立一个以当前节点为根结点的完全二叉树结构,利用节点间潜在的关联关系判断下一步可能被操作的对象,对预测对象提前进行绘制并调入内存,有效弥补了相关算法存在的缺陷,从而提高了虚拟场景漫游的效率和实时性,算法具有较高的准确性。

**关键词:** 虚拟场景; 立方体模型; 虚拟对象; 智能漫游算法; 完全二叉树

## Design and Implementation of Intelligent Roaming Algorithms Based on New Virtual Scene Model

YU Li-Li, DAI Qing

(Institute of Electronic Technology, PLA Information Engineering University, Zhengzhou 450004, China)

**Abstract:** Based on the analysis of the relevant intelligent prediction algorithm, it is found that existing algorithms don't consider the potential relevance of the relationship among the virtual scene node, and they reduce the prediction accuracy of the algorithm. To tackle this problem, based on a new cube virtual scene model, an intelligent roaming algorithm is proposed. The algorithm establishes complete BiTree structure for each node, uses the potential association between the nodes to predict the object which may be operated next, protracts the predicted object in advance and then transfers it into the memory. This algorithm effectively compensates for the shortcomings of the related algorithm, thereby improves the efficiency and real time of roaming virtual scene. The algorithm is highly accurate.

**Keywords:** virtual scene; cube model; virtual object; intelligent roaming algorithms; complete BiTree

基于图像的虚拟漫游系统实现的关键是保证场景显示实时性和显示画面视觉一致性。全景图是目前最简单的、使用最广泛的一种虚拟实景空间构造方法<sup>[1]</sup>。在用户漫游场景时,要根据用户输入,实时地从球面全景图生成并显示相应的场景视图,这是球面全景图系统最重要的性能,即视图生成的质量和速度。因为这直接影响场景漫游的视觉效果。如果视图生成速度较慢,则场景显示会有跳动感;如果视图的分辨率很低,则用户的视角受限制或降低了景物的清晰程度。由于视觉感知的质量在用户对环境的主观感知中占有最重要的地位。为了使用户能够真正的沉浸于由计算机所生成的虚拟环境中,生成的环境必须足够逼真和自然,也就是说虚拟场景模型的构造能够决定着一个

虚拟环境的好坏。这些问题的存在会降低用户漫游场景的真实感程度,所以需要作相应的研究以提高场景漫游系统在这两方面的指标。

### 1 一种新的虚拟场景模型

虚拟场景模型是将所有的视点按照空间的位置关系并通过链的形式联系在一起,这里的视点是指观察者某一时刻在虚拟场景中的观察点,在生成的虚拟场景中起到了管理全景图的作用,并且还能够确定视点空间的位置<sup>[2]</sup>。在以往的虚拟场景模型中,节点通常以节点网络形式进行链接,在这种节点的组织形式中,视点间仍处于二维状态,这并不符合真实的三维世界,作者在此基础上提出了一种新的虚拟场景模型<sup>[3]</sup>,该

<sup>①</sup> 收稿时间:2009-11-07;收到修改稿时间:2009-12-14

模型能更好的真实反映三维世界的情况。

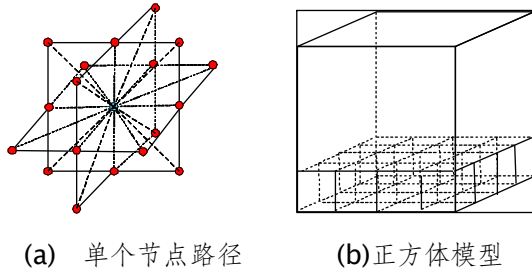


图 1 新的场景模型

新的虚拟场景模型依据人们浏览习惯建造，采用了位于球面全景图球心处的视点作为虚拟场景构造模型中的单个节点，模型中的每个节点都有 18 个方向的浏览路径，如图 1(a)所示，不仅扩充了原虚拟场景构造模型的浏览路径，而且增强了场景的真实感，更好的反映三维世界的情况。根据视点间相应的空间位置关系，每个节点在其浏览路径方向上通过空间链与其他节点链接，多个节点间关系在图 1(b)中正方体模型中体现，用户可以根据该模型构建一个视点间模型为正方体的基于球面全景图的虚拟场景。在这样的虚拟场景中，用户不仅可以实现单个视点内的环视和俯仰视，也能满足对整个场景空间的全方位观察。因此，新提出的虚拟场景构造模型可以更好地实现用户的全方位浏览。

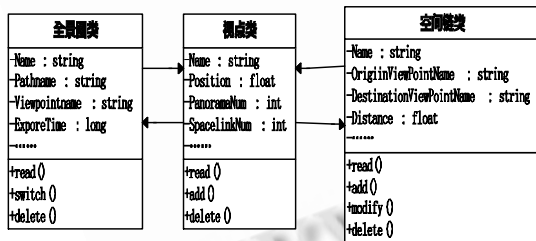


图 2 虚拟对象类关系

全景图类封闭了全景图对象，其成员包括了全景图像文件名称、路径名称、视点名称和浏览时间等信息。视点类则定义了视点对象信息，其中保存了该视点的名称、位置信息、属于该视点的全景图的数量，以及以该视点为源视点的空间链的数量及名称等相关信息。空间链类定义了空间链对象，离散的视点通过空间链的链接构造出一个完整的虚拟场景。空间链类中保存了空间链的名称、源视点和目的视点的名称，空间链的长度及方向。

在该模型中空间链的定义如下：每个点具体的描述可以参照文献[4]中二维平面模型，用方向和距离描述，两者结合，描述复杂，实现简洁，达到了效果，形式化描述如下所示，三维的视点空间模型是一个五元集合：

$3D-SpaceLink = \langle ID \rangle \langle Source \rangle \langle Destination \rangle \langle Orientation \rangle \langle Distance \rangle$

其中 **Source** 为源节点，**Destination** 为目的节点，**Orientation** 为方向，**Distance** 为距离。源节点和目的节点可以使用三维坐标(X, Y, Z)来表示。针对视点空间中某个节点存在的 18 个路径方向，定义一个数组 **A[18]**，其中 **A[i]**表示固定的一个方向，如 **A[0]**表示该节点的 X 平面上的正北方向。因为在模型中，中心视点的浏览路径是一次到边的方向，一次到顶点的 45 度方向，以 45 度为间隔出现不同的方向。因此对于数组 **A[18]**中 **A[i]**对应的节点，可以规定，**Distance** 值 **D** 需要先根据 **Orientation** 中 **A[i]**来判断，当 **i** 为偶数时，**Distance** 为常量 **d**，当 **i** 为奇数时，为  $\sqrt{2}d$ ，如下公式所示

$$Distance = \begin{cases} d & i \text{ 为偶数} \\ \sqrt{2}d & i \text{ 为奇数} \end{cases}$$

由本模型构造的虚拟场景将有以下几个特点：

(1) 首先将视点间二维的转移扩展为三维，也就是说不但视点内的浏览是三维的，而且视点间的漫游也是三维的，这更为符合我们的真实世界，也更符合人类的浏览习惯。

(2) 单个节点的路径得到扩展，由原来的 8 条，16 条[5]扩展为本文所设计模型中的 18 条，如模型中中心区域的节点下一步可到达的节点有 18 个节点，而且这 18 条路径分布在立体三维空间中，有效的拓展了虚拟场景的空间感，增强场景表现的真实性。

(3) 每个节点在其浏览路径方向上通过空间链与其他节点链接，观察者可以沿空间链在视点之间进行转移，完成虚拟环境中的游历，这种模型在用户的实时漫游时体现出其优越性。

## 2 智能漫游算法

虚拟场景漫游包括视点内漫游和视点间漫游。浏览全景图是根据视角和可见的范围取部分全景图进行重投影，实现场景的浏览。在通常的漫游系统中，常

采用的漫游算法是直接利用重投影算法进行实时绘制,该算法中包含许多平方开方、三角函数,需要较长的处理时间和较大的计算量,容易产生视觉延迟,对此许多文献中提出了重投影算法的改进算法<sup>[6]</sup>,一定程度上缓解了漫游的视觉连续性问题,但对于复杂场景或大视野的计算,仍显不足,基于图像的虚拟漫游系统实现的关键是漫游的无间断性,即用户在实时浏览时,无明显的视觉延迟。在计算机处理能力有限的情况下,最简单的解决方法<sup>[7]</sup>是将全部虚拟对象调入内存进行缓存并预处理,以备浏览工具使用,但是由于图像大多数据量较大而现有的计算机资源有限,因此上述方法是行不通的,要实现漫游的实时性必须从优化资料的处理算法和资源的调度策略着手,最大限度地降低计算机的计算量,提高现有资源的利用效率,针对此情况本文提出了智能漫游算法。

智能漫游是指先预测用户下一步最有可能浏览的虚拟对象,依照计算结果将预测的虚拟对象调入内存,从而减少实时绘制时所需要的时间消耗,真正达到漫游的连续性,实现用户浏览的实时性。在本文中采用的这种新场景模型中,用户能浏览的下一个对象可以有 18 个选择,如何从中确定一个操作对象是智能漫游的关键,同时智能漫游算法应在提高预测准确性的同时尽量简单,如果因为算法复杂导致预测的时间接近甚至超出了处理所有可能浏览对象的时间,那么预测就失去了意义。

## 2.1 问题描述

在虚拟场景漫游中,文献<sup>[7]</sup>中采用的方案是计算当前视点一步可到达的范围内的每个结点的浏览概率,寻找概率值最大的一个进行预测绘制。算法虽然简单易实现,但精准度不够,算法只考虑了结点被访问的次数,而并没有考虑结点前后间存在的关联关系。如由节点 A→B 有 3 次, A→D 有 10 次,由节点 C→B 的有 30 次,于是对象 B 被访问的总次数为 33 次,而实际上同节点 A→B 的次数只有 3 次,关联度并不高,若由该文献所采用的算法来计算,则节点 A 的下一个预测对象为 B,而实际漫游数据表明 A 与 D 的关联度更高,即由节点 A 访问 D 节点的概率更大,这种算法容易出现错误,因此不能仅仅从被访问次数来作为依据进行预测。

## 2.2 智能漫游算法分析与设计

在虚拟场景中所有虚拟对象集合为  $A=\{a_1, a_2, \dots, a_m\}$ , 通过一个操作可直接浏览对象集合为

$B=\{b_1, b_2, \dots, b_n\}$ , A 与 B 的关系满足: B 包含于 A 且  $b_i$  为当前节点仅通过一个操作指令即可浏览的对象,由于这些对象均为浏览者下一浏览的候选目标,所以它们也是智能漫游算法的处理对象。

在智能漫游算法中,考虑到上述问题,根据本文所采用的新型虚拟场景模型,用户在这种虚拟场景按照该场景拓扑结构进行漫游,算法是根据各节点的浏览次数变化,设计一个完全二叉树,用户的浏览次数可以作为各节点一个属性值且在不断更新,算法每次执行时需要尽快找出浏览次数最大的节点  $b_i$ , 进行用户浏览路径可能性的预测。

该完全二叉树建立如下:以当前的视点 P 作为根结点,而当前视点 P 的可能下一个动作浏览的对象集  $\{B_i\}$  中浏览次数最少的为根结点的最左孩子结点,从小到大依次排列,并设定完全二叉树中每个到根结点路径的权值之和为该结点从当前结点浏览的次数,在高度等于 1 的结点中,孩子结点与父结点浏览次数差为结点间路径的权值。因此,根据该完全二叉树模型,计算当前视点 P 的可能下一个动作浏览的对象集  $\{B_i\}$  中在预测结果中对应节点的全景图像,并且对到根结点的路径权值最大的结点优先进行绘制。

在本文提出的算法中,为每个结点建立个数组,用来记录该节点到下一个节点的次数,某个结节点的浏览次数  $R(B_i)=\sum m_i$ ,其中  $m_i$  为第  $i$  个结点  $B_i$  到根结点的路径上相邻两个结点间的权值。对于视点集合  $\{B_i\}$  中每个节点都有对应的浏览次数的描述,系统可以根据这些次数生成一棵带权值的完全二叉树模型,取当前的视点 P 为完全二叉树的根结点,模型结构如下图所示。

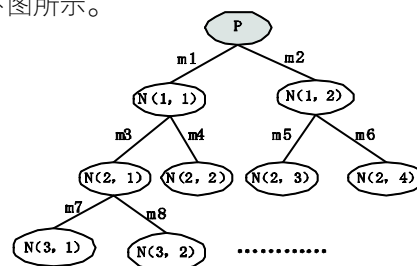


图3 完全二叉树模型结构图

本文设计的方案中,总是以当前视点为根结点,在一步可到达范围内的所有节点中,每次用户的浏览路径上作为目前点后继结点的次数。通过分析,可知文献<sup>[7]</sup>中方案实现简单但准确率稍低,本文方案存储

数据量相比第一种而言要大一些，同时计算开销也大些，由于每个节点的绘制需要占用大量的计算与存储开销，不可能所有节点均提前进行绘制，在本文采用的以正方体为模型构造的虚拟场景中，用户有 18 个方向的可选择漫游的节点，构成完全二叉树后最高层(高度为 4)只有四个叶子结点，因此在该算法中指定只优先计算二叉树中四个叶子结点中的最右边的叶子所对应的浏览节点，由算法的构造原理可知，显然这是概率最大的节点。其中每个节点的存储格式如下，如当前结点为 A，则节点 A 处的存储格式为如下所示。

表 1 节点存储格式

节点	浏览次数
B	20
C	5
D	4
E	28
...	...

采用完全二叉树结构对节点的这一属性值进行存储比较便于实现，在用户浏览过后也方便对树结构进行更新，而且每个树的结点还可以设计专门的结构存储节点的其它属性，比如节点的位置信息等；另外完全二叉树在数据结构上也有成熟的生成树算法以及查找和遍历算法可以使用，因此对算法的实现非常有利。

### 3 智能漫游算法的实现

本文采用的新的虚拟场景模型，视点间距离可以进行选择，也即当空间链的 distance 属性值在一定范围内时，视点间的切换并不会影响浏览的效果或在可接受的范围内，中间图像的作用可忽略不计，于是可以略去中间图像类以节省存储空间。智能算法的流程图如图 4 所示。

智能漫游算法具体步骤如下：

**Step1:**首先判断下一个结点是否存在，若存在则调用智能漫游算法，否则等用户浏览完提示没有下一个可操作结点，以避免陷入无法返回的局面。

**Step2:**调用智能漫游算法，预测出下一个浏览节点，判断其类型是全景图类，空间链类，视点类。

**Step3:**多线程同时进行，在不影响当前浏览动作的同时，调用不同的算法绘制预测对象类，全景图类调用重投影算法，空间链类在空间链的存储链表中读取其属性，指向等，视点类则读取其位置信息和相应

的全景图信息，绘制完后调入内存。

**Step4:**判断用户是否继续漫游。不继续则 break，释放预测对象的内存，返回漫游主界面。若继续漫游，当用户触发热区操作时，判断用户下一步进行操作的对象是否是预测的对象，若是则将提前绘制完的对象根据计算被显示于显示区中，到 step5；若下一步进行操作的对象不是预测的对象，判断对象类型，再调用针对该对象类型的绘制方法，如判断为全景图类，则直接调用反投影算法实时绘制，显示于显示视窗中，同时释放预测对象所占用的空间，到 step6。

**Step5:**释放当前节点的内存，智能漫游算法在每次更改当前视点时启动，到 setp6。

**Step6:**记录该节点访问的节点，在表格中将浏览次数加 1，重新建立完全二叉树。返回 step1。

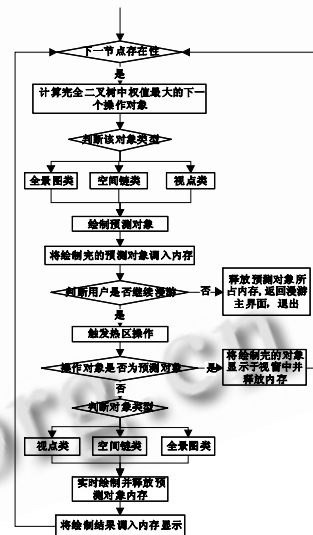


图 4 智能漫游算法流程图

在 QuickTime VR 中，为保持运动的持续性，在跳到下一个相邻的位置时，观察方向必须保持不变，在创建场景的时候，全景图可通过匹配它们的方向被链接在一起<sup>[8]</sup>，本文也采取这种方式，通过匹配 3D-SpaceLink 的方向值 Orientation 将全景图链接在一起，使得全景图切换在漫游时产生自然而准确的效果。

### 4 实验结果分析

针对上述提出的智能漫游算法，实验平台如下 Pentium M 1.70GHz, 内存 1GB, 编程环境为 Visual

C++6.0, 运行在 Windows XP 环境下, 采用相同的重投影算法, 在新的场景模型下, 分别实现了虚拟场景的智能漫游与常规漫游, 该漫游系统可以使用户进行上升、下降、向左、向右四个方向的浏览及前进与后退操作, 可以通过鼠标或键盘进行操作控制。

帧速率是指计算机每秒钟画面更新次数, 单位是“帧/秒”(frame per second, fps)。其数值越大, 显示的动画越流畅。为了得到比较精确的 fps 数值, 采用 fps 误差小于 1ms 的 Windows 常用计时函数 QueryPerformanceFrequency() 和 QueryPerformanceCounter() 的组合<sup>[9]</sup>进行计时。本文所提出的智能漫游算法是在当前节点浏览开始时并行运算, 在不同的分辨率下, 多次对其进行运算所得帧速率平均值与常规漫游算法帧速率的平均值相比较绘制成如下表格。

表 2 实验结果对比

帧格式	360*240	515*512	800*600	1024*1024
智能漫游 帧速率	37.0267	35.0712	32.8046	30.2758
常规漫游 帧速率	30.1573	28.8968	23.3250	19.9240

实验发现常规漫游算法在场景图像较小, 分辨率较低的情况下效果令人满意, 这是由于在这种情况下常规漫游算法的实时绘制时间与内存调度时间都较小, 与智能漫游算法的时间差较小; 而对于场景图像较大, 分辨率较高的情况下, 由于常规漫游算法的实时绘制与内存调度的时间开销比较大, 已不能满足平滑的仿真至少以 24fps 到 30fps 的速度显示的要求, 用户在漫游时产生明显的闪烁, 延迟等现象, 在实时性上无法达到视觉上的连续的效果, 使得用户漫游质量急剧下降; 而在高分辨率图像的复杂场景中智能漫游算法表现更为出色, 不仅满足了平滑仿真的帧速率要求, 达到视觉上连续无延迟的效果, 同时新的场景模型下, 用户的漫游范围得到极大扩充, 有效的弥补了二维场景模型下用户漫游范围局限性的缺陷。

## 5 结论

随着虚拟现实的发展, 人们对于虚拟场景漫游实时性与视觉一致性的要求越来越高, 本文在新的立方体场景模型的基础上, 针对以往智能预测算法精确度不高的问题, 采用二叉树机制, 根据作为当前结点后序浏览结点的浏览次数, 提出了一种准确度更高的智能预测算法, 充分考虑了结点间潜在的关联关系, 同时实验论证了在不影响当前系统效率的情况下, 显著提高系统漫游的速度与更高的预测准确性, 具有一定的实用价值。

### 参考文献

- 1 Shum HY Szeliski R. Omnivergent Stereo. IEEE International Conference on Computer Vision, September 1999.
- 2 齐越. 自由虚拟实景空间的研究与实现[博士学位论文]. 长沙: 国防科学技术大学, 2001.
- 3 于丽莉, 戴青. 一种新的基于球面全景图的虚拟场景模型设计与分析. 计算机研究新进展, 2009.
- 4 孙立峰, 钟立. 虚拟实景空间的实时漫游. 中国图像图形学报, 1999.
- 5 陈源. 基于图像的虚拟场景生成技术研究[硕士学位论文]. 郑州: 解放军信息工程大学, 2008.
- 6 冯海波. 虚拟全景空间生成技术与实现[硕士学位论文]. 成都: 电子科技大学, 2007.
- 7 喻昱. 基于图像的虚拟场景构造和漫游系统研究. 西安: 西安理工大学, 2002.
- 8 Shenchang Eric Chen. Quick Time VR--An Image Based Approach to Virtual Environment Navigation. Computer Graphics Proceeding, Annual Conference Series, 1995.
- 9 吴益明, 卢京潮, 魏莉莉, 闫建国. 基于 Windows 系统环境下的精确定时过程实现. 计测技术, 2005, 25(6): 67-69.