

一种支持多种运动方式的三维虚拟角色路径 规划和姿态调整方法

黄宁宁^{1,2} 曹存根²

(首都师范大学计算机科学联合研究院 北京 100048)¹

(中国科学院计算技术研究所智能信息处理重点实验室 北京 100190)²

摘要 针对三维动画场景,提出了一种支持多种运动方式的虚拟角色路径规划方法。根据具体角色的动作特征(飞、爬、走)建立不同的模型,然后利用 A* 算法搜索出一条无碰撞路径,最后对路径进行优化。实验表明,应用该方法规划出的路径适应于不同尺度的虚拟角色,并支持飞行、爬行、行走等多个运动动作,体现了三维路径规划的基本特征。

关键词 虚拟现实,三维路径规划,飞行,爬行,行走

中图分类号 TP391.9 **文献标识码** A

Motion Path Planning Method for 3D Virtual Characters with Different Motion Modes

HUANG Ning-ning^{1,2} CAO Cun-gen²

(Joint Institute of Computer Science, Capital Normal University, Beijing 100048, China)¹

(Key Laboratory of Intelligent Information Processing, Institute of Computing Technology, Chinese Academy of Sciences, Beijing 100190, China)²

Abstract A path planning approach to virtual 3D character motion was proposed. According to the specific characteristics of a character, we designed a relevant model. Then we used the A* algorithm to search for a collision-free path, and finally, the motion path was optimized. Experimental results show that path planned by our method can satisfy characteristics of different scales and action types, including flight, crawling, walking.

Keywords Virtual reality, 3D path planning, Flight, Crawling, Walking

1 引言

智能动画生成^[1]是一个多学科交叉的领域,其思想是通过利用人工智能的方法来提高动画制作的效率^[2]。在智能动画生成中,为了尽量减少人工干预,三维虚拟角色的运动路径规划是一个值得关注的基础课题。其目标是在三维动画场景下,给出虚拟角色的起始状态、目标状态及动作描述,根据角色自身特点及场景环境因素,为角色自动规划出一条合理的无碰撞路径。

路径规划是机器人学中的关键技术,已广泛应用于游戏制作、军事仿真、手机导航及探月工程等众多领域。根据三维场景建模的方式,现有的相关研究分为如下几类。

可视图法。该方法由 Nilsson^[3]于 1969 年提出,是以多边形近似描述起始状态、目标状态及场景中所有的障碍物,连接这些多边形的顶点,并保证连线不与障碍物相交,形成可视图,在图中搜索得到路径。Brian Salomon 等^[4]提出了基于可视图法的 PRM 算法,实现了三维虚拟漫游系统。该算法通过随机采样构造搜索图,效果受到采样粒度(采样概率)的影响,而采样粒度很难合理控制。

Voronoi 图法。其属于计算几何的经典问题,目前广泛

应用于路径规划问题^[5],其思想是将场景空间描述为 Voronoi 归纳图,图中的点与周围障碍物距离相等。赵文婷、彭俊毅^[6]讨论了如何根据无人机在各种情况下的威胁分布构建 Voronoi 图,计算航路代价、航迹搜索和航迹优化,性能较好,但是该方法存在一些局限性。Avneesh Sud 等^[7]利用 Voronoi 图建模,实现了动态环境中多 agent 的路径规划。该算法存在采样稀疏等一些问題,其准确性有待提高。基于 Voronoi 图的路径规划,实时性较好,但不能保证路径最优。

栅格分解法。作为目前研究最广泛的路径规划方法之一^[8,9],其主要思想是将场景空间分解为不相互重叠的区域(即为栅格),形成栅格集,然后根据栅格的隐式邻接关系搜索栅格集,得到最短路径。栅格划分方法可分为精确栅格法和非精确栅格法两种。精确栅格法得到的是尺寸相同的栅格,非精确栅格法则根据场景空间的疏密程度进行自适应分解。Srikanth Bandi 等^[10]提出了一种基于精确栅格法的 Simple bushfire 路径规划算法,其具有很好的拓展性,能扩展应用到多目标以及动态场景的复杂 3D 环境,但是算法效率依赖于栅格的精度,在复杂场景下效果难以保证。Jozef Vörös^[11]通过构造一种基于非精确栅格的树型结构(四叉树或八叉树)的 distance map 来搜索获取合理路径,以便在二维或三维环境下较

到稿日期:2011-02-21 返修日期:2011-05-12

黄宁宁(1986—),男,硕士生,主要研究方向为智能动画生成、路径规划, E-mail: huangnn888@gmail.com; 曹存根(1964—),男,研究员,博士生导师,主要研究方向为人工智能。

快、较好地完成简单场景的路径规划。但是该算法在搜索结点邻域时采用了基于指针的方式沿树型结构上升和下降搜索,不太适用于复杂场景的规划问题。史红兵等^[12]采用八叉树的场景建模方式,实现了虚拟场景摄像机规划,并对路径进行了优化,效果良好。

其他路径规划算法还有切线图法^[13]、神经网络法^[14]、人工势场法^[15]、遗传算法^[16]、蚁群算法^[17]、免疫算法^[18]和模糊逻辑法^[19]等。

归纳起来,上述方法从不同角度对路径规划进行了研究,但由于较少考虑三维场景的复杂性和角色动作的多样性,因此规划出的路径很难符合角色自身特征的要求。具体存在以下问题:

- 1)不能支持不同尺度的角色;
- 2)角色动作类型单一;
- 3)较少涉及角色姿态的调整。

为解决这些问题,本文设计出一套针对三维动画场景(Maya)的路径规划方法。本方法的主要贡献在于两个方面:一是适应于不同尺度的虚拟角色,如麻雀、鸽子、虚拟人、大象、蚂蚁、壁虎等;二是支持多个动作,包括飞行、爬行及行走3种运动类型,以及运动过程中的姿态调整动作。

本文第2节介绍三维路径规划方法的设计;第3节介绍系统实现和测试,最后总结全文。

2 三维路径规划方法设计

我们的工作基于三维动画制作软件 Maya,在该三维动画场景中,路径规划的基本流程如图1所示。

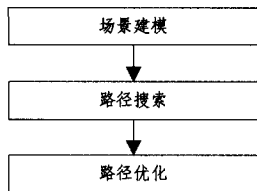


图1 规划流程图

场景建模就是采取一定的方法来准确地表达场景中的模型信息,是对场景中地形、障碍物等场景信息的预处理。该部分是三维场景路径规划的核心关键问题。充分获取场景中的模型信息,并设计恰当的数据结构组织这些信息,十分有助于搜索出逼真的路径,以提高路径搜索的效率。

路径搜索是在场景建模的基础上,根据角色本身和场景的约束限制条件,采用某种搜索算法搜索出一条无碰撞路径。这里主要采用启发式搜索算法 A* 算法,需要设计启发式函数、计算扩展邻域及进行碰撞检测。

路径优化可以得到更好的生成效果。我们的方法在设计中主要包括两部分:一是对搜索出的路径进行平滑处理;二是对场景中的特殊位置加入语义标注。第二部分可以根据具体场景的特殊情况,由动画建模师按照一定的规则灵活添加。我们主要处理第一部分。

根据场景中角色的运动特征,将三维路径规划分为飞行、爬行及行走3种类型。以下先介绍数据结构——八叉树,然后按照路径规划的流程详细说明每种类型。

2.1 八叉树数据结构

作为四叉树在三维空间的扩展,八叉树^[9]是一种重要的

场景组织结构。

借鉴并应用施斌^[9]三维场景分割方法,本文采用线性八叉树的编码基准分割体系,如图2所示。

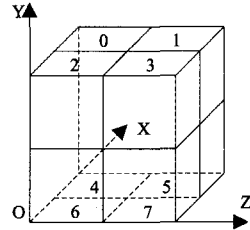


图2 线性八叉树分割体系

可以看到,线性八叉树的编码基准体系采用八进制,即0~7分别代表8个子结点空间,引入方向的概念,可以分为:

北部={0,1,4,5},南部={2,3,6,7}

西部={0,2,4,6},东部={1,3,5,7}

上部={0,1,2,3},下部={4,5,6,7}

结点每分解一次就增加一位编码表示,若设根结点层次为0,编码为空,则第一层结点编码分别为0~7,第二层结点编码相应为00,01,02,03,……,75,76,77。依此类推,逐层分解,逐个编码。

八叉树结点有两种:空结点和灰结点。其中灰结点包含采样信息。八叉树分解终止条件为当前结点为空结点或其边长达到角色最大边长。建树过程描述如下:

- 1)获取整个场景的范围和采样信息,将其放入八叉树根结点,建立第一个结点立方体;
- 2)对于当前结点,若其满足分解终止条件,则不再分解;否则,继续将其分解编码,得到8个子结点;
- 3)对于每个子结点,重复步骤2),直至建树完毕。

至此,建立了基于八叉树层次结构的场景描述,如图3所示,白色结点表示空结点,不包含采样点;灰色结点包含采样点。

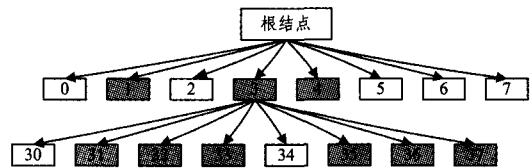


图3 八叉树分解过程

2.2 飞行路径规划

飞行路径规划对场景的精确性要求较低,运动主体为鸽子、麻雀等飞禽,故可采用八叉树进行简化场景建模。

场景建模 以角色绑定盒的最小边长为采样半径,对场景中各物体模型的组成面片进行采样,获取相应的采样点集。根据采样点集建立八叉树结构的场景描述。

路径搜索 获取八叉树叶子结点中的空结点,组成搜索集,用于 A* 算法搜索空间最短路径。启发式函数 $f(n) = g(n) + h(n)$,其中 $g(n)$ 是从起始结点到结点 n 的路径耗散, $h(n)$ 是从结点 n 到目标结点的距离。采用最常见的欧式距离公式来度量两个结点之间的距离。

定义1(八叉树结点间的距离) 结点 i 的中心点为 (x_i, y_i, z_i) ,结点 j 的中心点为 (x_j, y_j, z_j) ,则结点 i 和结点 j 之间的距离定义为:

$$\text{dist}(i, j) = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2 + (z_i - z_j)^2} \quad (1)$$

搜索过程中,扩展当前结点要考虑的方向为6个,即东、西、南、北、上、下。这里采用肖乐斌^[20]等人提出的线性八叉树邻域寻找算法,在搜索集中查询邻域。

路径优化 本文选择3次样条曲线分段拟合搜索出的路径点。3次样条曲线具有连续的二阶导数,可以很好地满足拟合要求。为保证拟合出的曲线不超出路径结点区域边界,需增选一定量的关键点。通过改进王金敏^[21]等人提出的关键点选择策略,得到八叉树建模条件下关键点的选择策略:如果下一结点是同级结点,则选择该结点的中心点作为关键点;否则,选择小结点中心点在大结点边界的对称点作为增选关键点。举例说明,图4为空间路径结点俯视图, O, P 为两个小结点的中心, A, B, C 为增选的关键点,其中 A, B 为 O 的对称点, C 为 P 的对称点;虚线箭头为初始路径,实线箭头为加入关键点后的路径。

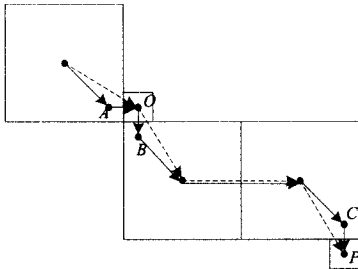


图4 关键点选择

最后,应用3次参数样条曲线拟合实线路径,最终生成的曲线通过所有的关键点。针对三维曲线特征,选用其参数方程如下:

$$\begin{cases} x=x(t) \\ y=y(t) \\ z=z(t) \end{cases}$$

式中, x, y, z 为关键点的三维坐标值, t 为关键点在路径中的序号($1 \leq t \leq n$)。以 t 为自变量分别拟合 x, y, z 。

姿态控制 飞行角色的姿态主要体现在翅膀摆动和角色朝向两个方面。

定义2(飞行动作元) 飞行角色完成一次翅膀来回扇动称为一个飞行动作元。

飞行动作由一系列飞行动作元组成。由动画建模师设定一个飞行动作元中翅膀的摆动幅度,并根据角色飞行的速度,设定翅膀摆动频率。有了这些数据,就可以实现飞行的程序化控制。

在 Maya 场景中,鉴于欧拉角控制角色朝向时会出现万向锁问题^[22],我们使用“Aim(目标)”工具约束角色,使角色总是瞄准 Aim,通过移动 Aim 控制约束角色的方向。如图5所示,当角色位于A点时,使角色的 Aim 位于B点,这样角色就朝向B点。依此类推,始终使 Aim 位于当前路径点的下一路径点,实现对角色朝向的控制。

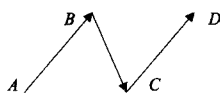


图5 飞行路径

2.3 爬行路径规划

爬行类路径规划的运动主体是蚂蚁或壁虎这类爬行动物角色。此类运动的动作自由度最大,且运动角色可以位于场景中相应物体的任意表面。可见,三维坐标位置已不足以描

述角色的状态,需重新定义爬行角色的状态。

定义3(角色的法向量) 角色的法向量定义为从角色脚部指向角色身体,方向与角色所处物体表面垂直。

定义4(爬行角色的状态) 已知爬行类角色的位置为 (x_i, y_i, z_i) ,法向量为 (u_i, v_i, w_i) ,则角色的状态由其位置和法向量共同决定,即 $[x_i, y_i, z_i, u_i, v_i, w_i]$ 。

场景建模 角色状态的复杂性决定了场景建模的精确性。此时,八叉树的简化建模方式已不能满足要求,因为其规划单元是八叉树结点,即一个立方体。本文采取精确化建模方式,由于 Maya 动画场景中的物体是由一系列四边形面片(也有极少的三角形)组成的,因此将规划单元具体到这些四边形面片,称为单元面片。单元面片的中心点作为角色的位置,其法向量作为角色的法向量。

此外,还需测试每个单元面片与其他面片的邻接关系,以利于下一步的路径搜索。利用 Maya 提供的 C++ API,可以获得同一物体上单元面片间的邻接关系。而对于不同物体,则需要相交测试。为了高效地进行相交测试,这里仍借助于八叉树,其步骤如下:

- 1) 获得场景中所有单元面片的采样信息,组成单元面片集;
- 2) 根据单元面片集,利用八叉树划分场景,每个叶子结点存储位于该结点的所有单元面片,该过程类似于飞行路径规划中的建模方式;
- 3) 对于八叉树的每个叶子结点,若其包含的单元面片属于同一物体,则不进行相交测试;否则,对该结点内的单元面片,两两进行相交测试,并存储结果。

四边形可以分解为两个三角形,所以两单元面片间的相交测试可以转化成空间两三角形间的相交测试^[22]。至此,得到了蕴含位置、邻接关系等信息的单元面片集,该集合即为路径搜索的搜索集。

路径搜索 该阶段同样需要定义两单元面片间的距离,以利于 A* 算法的搜索扩展。

定义5(边界面) 已知单元面片为 n 边形,若与其具有公共边的相连单元面片的数目小于 n ,则定义该单元面片为边界面。

定义6(单元面片间距离) 单元面片 i 的中心点为 (x_i, y_i, z_i) ,法向量为 (u_i, v_i, w_i) ,单元面片 j 的中心点为 (x_j, y_j, z_j) ,法向量为 (u_j, v_j, w_j) ,则单元面片 i 和单元面片 j 之间的距离定义如式(1)所示。特殊地,当 $dist(i, j)=0$ 且两面片法向量相反时,角色处于同一面片的两个状态。若该面片为边界面,则设置 $dist(i, j)$ 为该面片的边长,角色可以调整姿态,到达该边界面的另一状态;否则 $dist(i, j)$ 为无穷大,以示不可通过。

在搜索扩展时,对于当前单元面片,其扩展方向为8个:前、后、左、右、左前、左后、右前和右后,并加上与其相交的面片。而对于能否扩展,则要进行碰撞检测。

角色的绑定盒有6个面,选择角色面部朝向的面作为测试面,如图6所示。

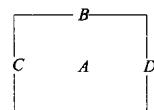


图6 角色正面图

图6中测试面上标有4个点:A,B,C,D。若角色所处单元面片的中心点为 $P=[x,y,z]$,单位法向量为 \vec{f} ,角色宽度为 w ,高度为 h ,朝向为单位向量 \vec{o} ,则角色宽度方向为 $\vec{w}=\vec{f}\wedge\vec{o}$ 。A,B,C,D 4点坐标分别为:

$$P_A=P+0.5 * h * \vec{f} \quad (2)$$

$$P_B=P+h * \vec{f} \quad (3)$$

$$P_C=P_A-0.5 * w * \vec{w} \quad (4)$$

$$P_D=P_A+0.5 * w * \vec{w} \quad (5)$$

由此可得,若角色从当前单元面片爬行到下一扩展面片 j ,则A,B,C,D 4点相对应地移动到 A_j, B_j, C_j, D_j ,这样就形成了4条线段 AA_j, BB_j, CC_j, DD_j 。利用这4条线段做碰撞检测,步骤如下:

- 1)对于每条测试线段,计算其经过的所有八叉树结点单元;
- 2)测试该线段与每个结点内的单元面片是否相交,转化成线段与三角形的相交测试问题;
- 3)在以上两步的迭代测试中,若没出现相交,则角色爬行没发生碰撞,面片 j 可扩展;否则,发生碰撞,面片 j 不可扩展。

可以使角色先爬行到当前面片与可扩展面片的交点处,经此缓冲,然后到达扩展面片中心点处,这样爬行效果会更加逼真。基于碰撞检测,由A*算法搜索出路径点集,每个路径点表示形式为 $[x_i, y_i, z_i, u_i, v_i, w_i]$,其中前3个值为该路径点的三维坐标,后3个值为其法向量。

路径优化 同样选择3次样条曲线分段拟合路径点,其关键点选择策略是:对于连续的路径点,若其法向量无变化,则可对这些点充分拟合,面片中心点作为关键点;若法向量发生变化,则面片中心点及面片之间的交点均作为关键点,且分段要多些。

姿态控制 爬行角色的状态和路径描述比较复杂,需要对姿态控制做相应的调整。爬行角色大都为多足动物,我们主要研究常见的四足动物,如图7所示。



图7 四足动物

图7中,A点为右后足,B点为左后足,C点为右前足,D点为左前足。根据四足动物的运动特征可知:当它慢速爬行时,A和D一起运动,B和C一起运动;当它快速爬行时,A和B一起运动,C和D一起运动。由此,定义爬行动作元。

定义7(爬行动作元) 慢速爬行时,A和D先前移一步,接着B和C也前移一步,将角色完成的这两步爬行动作称为一个爬行动作元。同理,快速爬行时,爬行动作元定义为C和D先前移一步,接着A和B前移一步。

与飞行动作相似,爬行动作同样由一系列爬行动作元组成。由动画建模师根据角色爬行的速度,设定角色肢体的移动幅度和频率,然后编程,实现角色的自动爬行。

在控制角色状态时,同样使用“Aim(目标)”工具,并对Aim位置的选择策略做以下修改:在图4中,若角色位于A

点,B点坐标为 p_B ,法向量为 \vec{f} ,则Aim点的位置为 $(p_B-\vec{f} * h)$,即让Aim点在B点沿B点法向量反方向移动 h (设定参数)长度的距离。依此类推。此方法可有效控制角色的方向和法向量。

2.4 行走路径规划

路径规划主要针对大型运动角色(如人或大象等)。在行走过程中,角色在地形表面移动,其法向量保持不变(与重心方向相反)。如果遇到陡坡,将无法经过。可见,此类运动受重力方向的约束,同时需要精确的地形信息。

经过对角色行走特点的分析,发现此类规划与爬行路径规划大同小异,在其基础上加入角色法向量约束,就可达到要求。即在A*算法搜索过程中,加入扩展条件:若扩展面片的法向量与角色法向量夹角大于一定阈值(如 45°),则扩展面片不可经过。

在姿态控制方面,与爬行角色相比,行走角色的不同主要表现在动作特征和朝向控制两方面。

行走角色大致分为两足动物和多足动物,其动作元定义如下。

定义8(行走动作元) 对于两足动物,将左脚先迈一步,接着右脚迈一步,这两步称为一个行走动作元;对于多足动物,其定义同爬行动作元。

朝向控制方案是:对于两足动物,通过计算其转弯时绕Y轴旋转的角度来控制朝向;对于多足动物,如果遇到斜坡,其法向量会略微有些变动,此时采用爬行角色的朝向控制方案。

3 系统实现和测试

在Visual Studio 2005(C++)平台下,通过编写Maya插件的形式来完成路径规划功能。在采样场景数据的过程中,使用到了Maya提供的一套C++API。另外,利用Maya提供的一种脚本语言MEL,实现对角色肢体运动的控制。图8为系统结构简图。

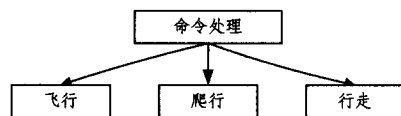


图8 系统结构简图

在命令处理模块,通过解析输入的命令文本,从场景中获取角色的大小范围、位置、起始状态、目标状态及动作类型等信息。例如,命令文本“鸽子飞行 $x_0, y_0, z_0, x_i, y_i, z_i$ ”,其中鸽子为角色名,飞行为动作类型,剩余信息为角色起始位置和目標位置。

经过命令处理,根据得到的角色动作类型,分别转至飞行、爬行和行走3个规划模块,最终生成路径。最后,应用MEL语言,编程控制角色姿态,使其沿着规划路径运动,生成关键帧动画。

所选择的典型测试用例如表1所列。

表1 测试用例

| 动作类型 | 测试用例 |
|------|--------------|
| 飞行 | 简单场景、复杂场景 |
| 爬行 | 简单场景、复杂场景、球体 |
| 行走 | 简单场景、复杂场景、球体 |

表 1 中简单场景的模型包括地表面片 1 个、与地表相交的桥 1 个、悬空障碍 5 个;复杂场景则在简单场景的基础上增加 10 个悬空障碍;对于球体,则在其表面进行爬行和行走测试。

以复杂场景和球体为例分析测试结果。图 9 为复杂场景中规划出的飞行路径, S 为开始点, T 为目标点。可见,飞行路径能够绕过障碍,并且在转弯处有拟合后的平滑效果。图 10 为爬行路径图,开始点 S 为地表面上的一点,而目标点 T 位于桥面,角色沿着规划出的路径爬行,就可成功绕开悬空障碍,由地面到达桥面,且路径完全依附地形表面,在地面和桥面的交换处,有明显的交点转折。图 11 进一步地说明该路径可以穿过孔洞;图 12 表示在球体表面的规划情况,对于爬行角色,规划均可成功,而行走角色的规划,则大都失败。此外,每种角色的姿态调整也都成功实现。

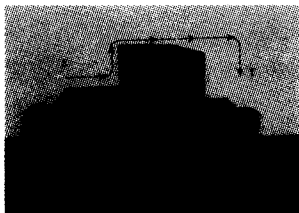


图 9 复杂场景飞行路径

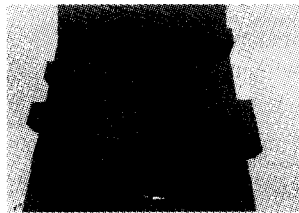


图 10 复杂场景爬行路径全局图



图 11 复杂场景爬行路径侧视图



图 12 球体表面爬行路径

整个规划过程在 Pentium(R) 4, CPU 3.00GHz, 2GB 内存, Windows XP 环境下进行。Maya 版本为 2008。对于简单场景和复杂场景,飞行规划时间均在 0.5s 以下,爬行规划时间分别为 2.23s 和 2.65s。行走规划的耗时与爬行规划大致相同;对于球体,各类规划时间均小于 0.5s。

从实验结果可以看出,所提方法可以为角色规划出合理的无碰撞路径,且该路径在满足角色本身动作约束的条件下达到最优。在时间方面,飞行规划时间较少,说明八叉树简化建模的高效性;对于爬行和行走规划,由于需要精确建模,故耗时与飞行相比比较长;由于对球体建模时,无需进行相交测试和碰撞检测,故规划时间较短。此外,每种规划类型对角色的尺度大小没有限制,并支持姿态调整,满足研究要求。

结束语 路径规划是智能动画生成中的一个重要组成部分。本文以运动角色的动作特征为切入点,设计出一整套路径规划方案,包括自动生成适合于 3 类具有不同运动方式的虚拟角色的运动路径,以及在运动过程进行必要的姿态调整,获得了较满意的动画效果。

进一步的工作重点是提高爬行和行走规划的效率,降低场景建模及碰撞检测的耗时。此外,群体的路径规划也是下一步要研究的问题,涉及相同动作类型的群体和不同动作类型的群体。

- [1] 曹存根,李良君,李志豪,等. 智能动画创作系统 PNAI 的研究进展[J]. 系统科学与数学,2008,28(11):1407-1431
- [2] 李良君. 动画自动生成中摄影方案的设计与实现[D]. 北京:首都师范大学,2007
- [3] Nilsson N J. A Mobile Automaton; An Application of Artificial Intelligence Techniques [C] // Proceedings International Joint Conference on Artificial Intelligence. 1969:509-520
- [4] Salomon B, Garber M, Lin M C, et al. Interactive Navigation in Complex Environments Using Path Planning [C] // Proceedings of Symposium on Interactive 3D Graphics. Monterey, California, 2003:41-50
- [5] Canny J F. A Voronoi Method for the Piano Movers Problem [C] // IEEE International Conference on Robotics and Automation. 1985:530-535
- [6] 赵文婷,彭俊毅. 基于 VORONOI 图的无人机航迹规划[J]. 系统仿真学报,2006,18(增刊 2):159-162
- [7] Sud A, Andersen E, Curtis S, et al. Real-time Path Planning in Dynamic Virtual Environments Using Multi-agent Navigation Graphs[J]. IEEE Transactions on Visualization and Computer Graphics, 2008:526-538
- [8] Kambhampati S K, Davis L S. Multiresolution Path Planning for Mobile Robots[J]. IEEE Journal of Robotics and Automation, 1986(RA-2,3):135-145
- [9] 施斌. 虚拟场景下自动路径选择的研究[D]. 南京: 河海大学, 2008
- [10] Bandi S, Thalmann D. Path finding for human motion in virtual environments[J]. Computational Geometry, 2000,15(1-3):103-127
- [11] Vörös J. Low-cost Implementation of Distance Maps for Path Planning Using Matrix Quadrees and Octrees[J]. Robotics and Computer Integrated Manufacturing, 2001,17(6):447-459
- [12] 史红兵,张毅彬,董若锋,等. 虚拟场景自动漫游的路径规划算法[J]. 计算机辅助设计与图形学学报,2006,18(4):592-597
- [13] Liu Y H, Arimoto S. Computation of the Tangent Graph of Polygonal Obstacles by Moving-line Processing[J]. IEEE Transaction on Robotics and Automation, 1994,10(6):823-830
- [14] 禹建丽,孙增圻,成久洋之. 一种快速神经网络路径规划算法[J]. 机器人,2001,23(3):201-205
- [15] 刘涛,李海滨,段志信. 基于人工立场的机器人路径规划研究[J]. 计算机仿真,2007,24(11):144-146
- [16] 陈华华,郭晔,杜敬,等. 基于改进型遗传算法的动态避障路径规划[J]. 传感技术学报,2006,19(2):520-524
- [17] 刘视菊,杨青川,辜吟吟. 蚁群算法在机器人路径规划中的应用研究[J]. 计算机科学,2008,35(5):263-265
- [18] 胡小明. 基于免疫算法的机器人路径规划研究[D]. 上海:上海大学,2005
- [19] 毕盛,朱金辉,闵华清,等. 基于模糊逻辑的机器人路径规划[J]. 机电产品开发与创新,2006,19(1):21-22
- [20] 肖乐斌,龚建华,谢传节. 线性四叉树和线性八叉树邻域寻找的一种新算法[J]. 测绘学报,1998,27(3):195-203
- [21] 王金敏,崔奇,初楠. 运用三次样条曲线拟合机器人运动路径[J]. 机械设计,2005,22(7):44-46
- [22] Tomas A M, Eric H. 实时计算机图形学(第 2 版)[M]. 普建涛,译. 北京:北京大学出版社,2004