

一种室内智能吸尘器充电返回算法

任盛怡,曹长修,马世文,张明瀚

(重庆大学 自动化学院,重庆 400030)

(rensyayao@163.com)

摘要:针对智能吸尘器自动返回充电所遇到的问题,提出在不增加传感器的条件下,使用 Dijkstra 算法实现返回充电座最短路径的方法。利用简化栅格法建立环境地图并分块,基于广度优先和深度优先算法对其进行生成树,最后使用 Dijkstra 算法得到最短路径。该算法通过软件实现了智能吸尘器自动返回充电的功能,在实际应用中效果良好。

关键词:广度优先算法;深度优先算法;Dijkstra 算法

中图分类号: TP242.6 **文献标志码:** A

Algorithm of intelligent cleaning-robot charging return

REN Sheng-yi, CAO Chang-xiu, MA Shi-wen, ZHANG Ming-han

(College of Automation, Chongqing University, Chongqing 400030, China)

Abstract: A Dijkstra algorithm to find the shortest path for cleaning-robot returns to charge was presented. After a subarea map was established based on simplified grid world maps, a spanning tree was fabricated through using the breadth-first and depth-first traversals, finally the shortest path was obtained by Dijkstra arithmetic. Without increasing other sensors, the cleaning-robot can achieve return to charge automatically. This algorithm has a good effect in its practical application.

Key words: breadth-first traversal; depth-first traversal; Dijkstra algorithm

0 引言

智能吸尘器可以帮助人们完成一些日常清洁工作,提高生活质量。其中的关键技术^[1]包括:传感技术,路径规划技术,吸尘技术和电源技术。尽管目前国内在这些技术上都有了很多相关的研究和分析^[2],但仍有许多关键技术问题需要解决与提高,路径规划是其中重要的关键技术之一。由于吸尘器使用蓄电池,工作一定的时间后肯定会出现电量不足的情况,此时只有对其进行充电才能保证继续工作。目前市场上的大部分产品要么没有自动返回充电功能,要么需增加其他传感器来实现该功能。因此,在不增加声呐、超声波传感器^[3-4]的基础上,从算法上寻找一种可行的自动返回充电的方法是一件有意义的事情。

1 结构介绍

本文所讨论的智能吸尘器充电座对本体发出的红外信号覆盖面范围为 1.5 m。当智能吸尘器处于覆盖范围之外且需要充电时,不能轻易找到充电座位置。针对此问题,本文提出了对室内进行简单栅格分块后最终使用 Dijkstra 算法来实现自动返回充电。室内智能吸尘器系统结构如图 1 所示。

各部分具体功能如下:

- 1) 红外传感器 1 用于探测右侧障碍物。
- 2) 红外传感器 2 为接收充电座红外信号。
- 3) 传感器 3 为内部传感器(包括陀螺仪、光电编码盘)用于检测方向、移动距离等。

- 4) 吸尘器本体的前方有两个防碰撞传感器,主要用于近距离探测障碍物。
- 5) 步进电机用于驱动左右两边的驱动轮。

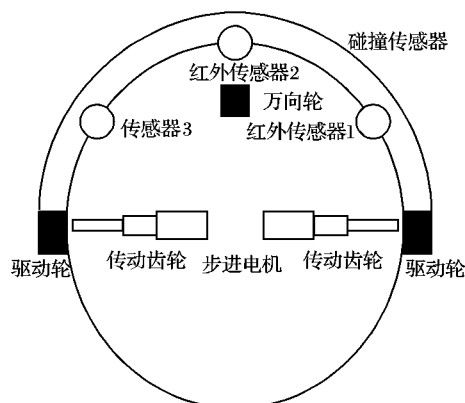


图1 智能吸尘器内部结构

2 环境空间的区域划分与探障

2.1 探测空间轮廓

智能吸尘器工作的环境可以分为静态结构化环境、动态已知环境和动态不确定环境。本文是在静态空间环境^[5]的基础上做划分。首先让智能吸尘器沿室内墙角行走一圈,右侧传感器始终保持为障碍物状态,设定右侧为障碍物区,左边为自由区。在行走过程中,根据内部传感器实时采集到的信息记录下所有左转和右转的转折点,作为勾勒室内轮廓的初始数据并对非自由区进行设障。探测结果如图 2 所示,白色为自由区,黑色为障碍物区。

收稿日期:2008-12-22;修回日期:2009-03-04。 基金项目:重庆市科技攻关重点项目(2004AA2001-8277-02)。

作者简介:任盛怡(1983-),女,四川绵竹人,硕士研究生,主要研究方向:智能感知与信号处理、数据分析与预测;曹长修(1937-),男,山东蓬莱人,教授,博士生导师,主要研究方向:计算机通信、神经网络、图像处理;马世文(1976-),男,宁夏吴忠人,硕士研究生,主要研究方向:专家系统;张明瀚(1981-),男,甘肃康乐人,硕士研究生,主要研究方向:数字信号处理。

2.2 二次遍历探测障碍物

2.2.1 合并关联

在探测室内轮廓地图的过程中,同时记录了所有转折点的上、下、左、右四个延长线方向上的交点,这些交点和所有转折点构成集合 Ω , Ω 即是对自由区域进行分块的初始数据。令 $Rarea(N) = [LB(N), RT(N)]$,其中 $Rarea(N)$ 表示第 N 块矩形块, $LB(N)$ 和 $RT(N)$ 分别表示第 N 块矩形块的左下顶点和右上顶点。对集合 Ω 中所有的点的坐标按横轴和纵轴排序得到相对应的数组 X 和数组 Y 。其分块算法步骤如下:

1) 对数组 X 中的元素做大循环,直到数组 X 溢出,则进入第3)步;如未溢出,记下当前的点 $LB(N) = (X(i), y_{start})$ (其中 y_{start} 表示探测时左下角纵坐标,此时 $y_{start} = Y(j)$)进入第2)步。

2) 对数组 Y 做元素 $X(i)$ 的循环,直到数组 Y 溢出,返回第1)步。未溢出则判断当前点 $(X(i+1) - 0.5, Y(j) + 0.5)$ 是否属于障碍区,不属于, j 加1后进入第2)步;属于,判断下面两种情况:

a) 当 $j \neq 1$ 时, $RT(N) = [X(i+1), Y(j)]$, $N = N + 1$, $y_{start} = Y(j+1)$,返回第2)步。

b) 当 $j = 1$ 时,令 $y_{start} = Y(j+1)$,进入第2)步。

3) 结束。

为了极大地减少数据的存储,对初步分块进行合并优化。对所有情况分析得出,只要相邻的矩形块满足下面条件之一即可合并。

1) $a(i+1) = b(i)$, $a(j) = b(j)$, $a(j+1) = b(j+1)$;

2) $a(i) = b(i+1)$, $a(j) = b(j)$, $a(j+1) = b(j+1)$;

3) $a(i) = b(i)$, $a(j) = b(j+1)$, $a(i+1) = b(i+1)$;

4) $a(i) = b(i)$, $a(j+1) = b(j)$, $a(i+1) = b(i+1)$ 。

式中: $a(i, j)$, $a(i+1, j+1)$ 为块 a 的左下和右上坐标点; $b(i, j)$, $b(i+1, j+1)$ 为块 b 的左下和右上坐标点。

对图的遍历有广度优先和深度优先算法^[6]。对于有 n 个节点的有向图来说,这两种算法的时间复杂度都为 $O(n^2)$,因此随着节点 n 的增加,系统的开销增长较快,之所以将该算法用于本文系统有两个原因。

1) 两种遍历简单方便,适合对树的遍历。

2) 该智能吸尘器用于摆放合理的家居环境,空间自由区域较大,对其分块所形成的树中,大树枝相对较少,所以利用广度优先遍历算法按层遍历的特点,可以减少该算法遍历时系统所需的堆栈深度。

优化分块之后需要将每个块关联起来,采用广度优先遍历对所有分块进行遍历,最终生成树如图3虚线所示。

2.2.2 二次遍历

通过上述方法得到环境的大致形态后,还需要确定自由区中障碍物的位置。广度优先遍历仅仅将所有分块连接起来,当智能吸尘器按顺序(1 2 3 4 5 6 7 8 9 10 11 12 13 14)遍历时,来回的摆动太大,所以为了找到一条合适的路径来搜索这棵树,还要采用深度优先算法,对生成树进行遍历。最终得该路径为:(1 3 6 4 2 4 6 5 6 7 8 9 10 11 12 13 12 14 12 11 10 9 8 7 6 3 1)。这样最大限度地减少智能吸尘器在遍历过程中的来回摆动,使其能在遍历完所有节点后返回原来的起始位置。使用上述路径来搜索自由区中的障碍物,开启智能吸尘器的探障传感器,利用纵向或横向搜索方法按照深度优先算法所遍历的顺序探测所有矩形块。探测完后再次对自由区域进行合并优化。最终分块图与生成树如图4所示。

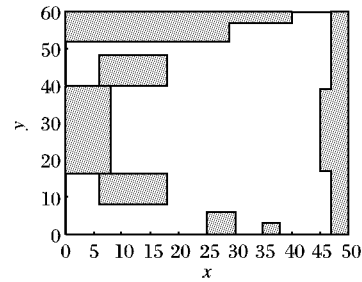


图2 室内轮廓

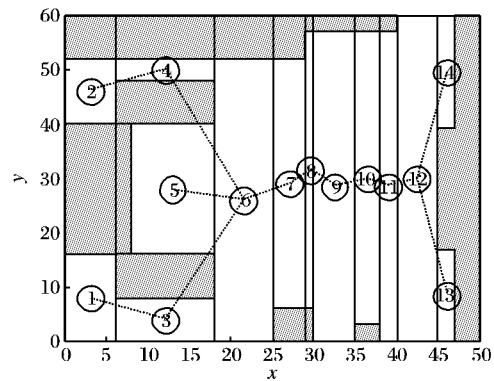


图3 合并优化分块,广度优先生成树

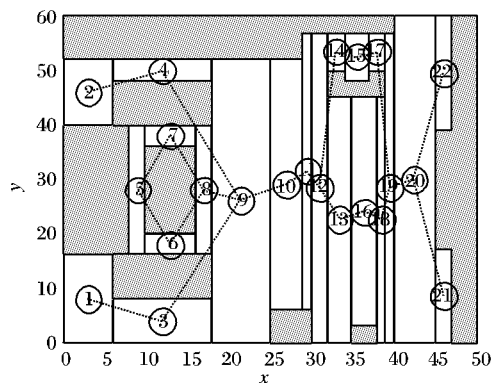


图4 二次遍历及生成树

3 寻找返回路径

前面已经提到,目前市场上同类产品要么通过增加传感器硬件,要么通过粗定位算法来实现自动返回充电功能,前者直接造成成本的大幅提高,后者运行效率低下,所以从算法上来提高返回的效率便是本文的重点。

3.1 粗定位算法

这里简单介绍粗定位算法思想:当智能吸尘器在返回充电途中,遇到的障碍物要么是墙壁,要么是孤立障碍物(桌椅等)。在相同的速率下,智能吸尘器绕行墙壁一圈的运行时间远远大于绕行孤立障碍物一圈的运行时间。根据智能吸尘器的行走速度和房间的大小,运行算法之前设定一个时间阈值 T ,当吸尘器绕行障碍物一圈的运行时间小于 T 时,认为是孤立障碍物(桌椅等);当大于阈值 T 时,认为是墙壁。当判断为绕行孤立障碍物时,返回原点后随机转动一个预先设定的转角后直走,直到碰到障碍物后,继续判断此障碍物是墙壁还是孤立障碍物,如此循环往复,直到找到墙壁后沿其边缘行走到达充电座为止。

3.2 Dijkstra 算法

以上算法在实际应用中效率低,为了达到消耗最少的电量返回充电座的目的,寻找一条最短的返回路径是解决问题的关键。目前解决最短路径问题^[7-8]的算法很多,由于一般充

电座处在相对固定的位置,所以问题转化为求任意一点到固定点间的最短路问题,Dijkstra 算法则是解决这一类问题比较合适的方法。

假设充电座位于矩形块 1。首先换算两两矩形块之间的权 $w(i,j)$,设相邻两矩形块中心点之间的距离为两矩形块之间的权,即 $w(i,j)$ 表示 i 矩形块与 j 矩形块之间的距离。 $v(i)$ 表示第 i 个矩形块, $i = \{1,2,\dots,22\}$ 。 $u(j)$ 表示第 j 矩形块与第 1 块之间的距离, $j = \{1,2,\dots,22\}$ 。该算法步骤如下:

1) 令 $u(1) = 0, u(j) = w(1,j) (2 \leq j \leq 22), S = \{v(1)\}, T = \{v(2), v(3), \dots, v(22)\}$, 其中 S 中的点给予永久标号, T 中的点给予临时标号, $r = 0$, 进入第 2) 步。

2) 在 T 中取一点 $v(i)$, 使得 $v(i) = \min\{u(j)\}$ 。如果 $u(i) = +\infty$, 停止, 表示从 $v(1)$ 到 T 中的各节点没有路; 否则转第 3) 步。

3) 令 $S = S \cup \{v(i)\}, T = T - \{v(i)\}, r = r + 1 (u(i)$ 改为永久标号)。如果 $r = 22$, 结束, 所有各点最短路已经求得; 否则转第 4) 步。

4) 对所有 $v(i) \in T$, 令 $u(j) = \min\{u(j), u(i) + w(i,j)\}$, 返回第 2) 步。

3.3 Dijkstra 算法路径修正

利用 Dijkstra 算法求得的最短路径是两两相邻矩形块之间的中点连线, 然而两相邻矩形块之间的中点连线不一定是直通, 所以直接走向下一块的中点时可能会发生碰撞现象。为了避免碰撞的发生, 所以在利用 Dijkstra 算法求得最短路后, 还要对这条路进行修正。修正规则如下: 取所求最短路中任意相邻两矩形块为 M 和 N (MN 为块中点), 假设本体此时从 M 向 N 行驶, 在 M 与 N 的连线 L 上取十个平均点, 依次判断这些点是否属于障碍区, 如无则认为直通, 可以直接行走; 如有则分以下两种情况:

1) 当 M 块与 N 块的交线及其延长线为 OPQ 时, 将坐标 P 作为插值, 使其路径为 MPN , 如图 6 所示。

2) 当 M 块与 N 块的交线及其延长线为 $OPP'Q$ 时, 将插值 P 或 P' 纵坐标和 M 纵坐标差值绝对值较小的一点作为转折点, 使其路径为 MPN 或 $MP'N$ 。

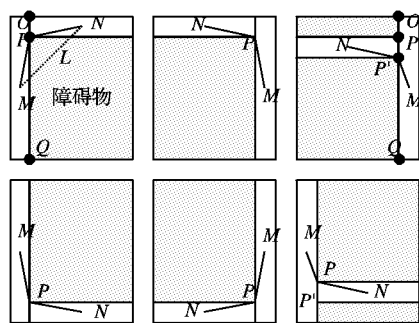


图 5 路径修正

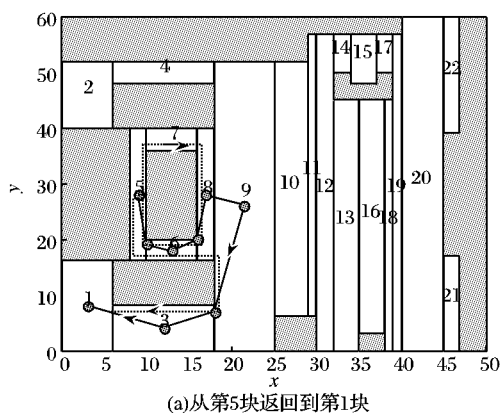
利用这种方法修正出的路径可以避免智能吸尘器在返回过程中发生碰撞现象。

4 仿真试验分析

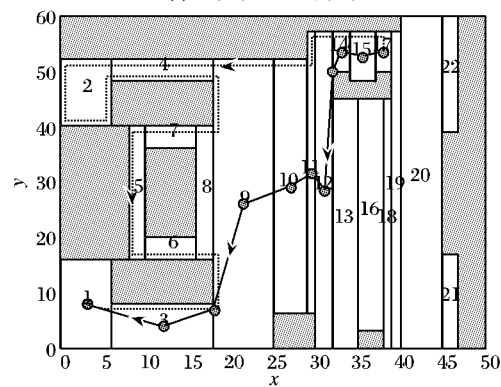
根据上文所述的环境划分和算法描述, 进行了仿真试验, 模拟出智能吸尘器分别采用 Dijkstra 算法与粗定位算法后的返回路径。在仿真试验中, 设定第 1 块为充电座所在位置, 分别让智能吸尘器处于其他任意位置, 且处于待充电状态; 假设粗定位算法中的随机转角是 90° 。仿真实验如图 6 所示。在图 6 中, 虚线表示粗定位返回路径, 实线表示 Dijkstra 算法返

回路。图 6(a) 中, 虚线表示智能吸尘器处于第 5 块时需要充电, 使用粗定位算法时先假设遇到孤立障碍物而不是墙壁, 则根据此算法, 需要将绕行障碍物一圈的运行时间与预先设定的时间阈值 T 进行比较, 如果小于时间阈值 T , 则认为不是墙壁。回到原点后旋转一个随机角度 (假定给定值为 90°) 后直走, 当碰到障碍物后继续用此方法进行时间阈值判断, 直到最终回到第一块为止。实线表示: 利用 Dijkstra 算法首先计算出最短路径为 (5, 6, 8, 9, 3, 1), 其次再对这条路径中相邻两块是否直通进行修正, 修正结果为 (5, 56 转折点, 6, 68 转折点, 8, 9, 93 转折点, 3, 1), 最后利用修正路径指导吸尘器返回充电座。

图 6(b) 中, 假定吸尘器处于第 17 块时需要充电。虚线表示的粗定位算法中, 它直接找到墙壁并沿其边缘回到第 1 块。实线表示使用 Dijkstra 算法得到的返回路径为 (17, 15, 14, 1412 转折点, 12, 11, 10, 9, 93 转折点, 3, 1)。



(a) 从第 5 块返回到第 1 块



(b) 从第 17 块返回到第 1 块

图 6 Dijkstra 算法与粗定位算法对比

比较图 6 中的两种路径, 可以清楚地看出在粗定位算法的返回路径中, 确实出现了对孤立障碍物的绕行, 增加了返回路径的长度和返回时间, 运行效率低下, 并不是理想的返回方法。而利用 Dijkstra 算法时返回距离大大减少, 无需绕行孤立障碍物, 运行效率有了很大的提高。

最后, 分别让智能吸尘器处于其他任意位置, 实验结果表明智能吸尘器在通过 Dijkstra 算法的操作之后都能够快速、准确地返回充电座。

综上所述, 通过 Dijkstra 算法为智能吸尘器寻找最短充电返回路径的方法是可行的。

5 结语

本文仿真实验验证了通过软件实现智能吸尘器自动返回充电功能的可行性。由于在不增加成本的条件下实现了自动

(下转第 1574 页)

的图像却基本一致。这说明利用信号的 MP 稀疏分解方法可有效地消除噪声的影响,提取出信号最主要的结构特征信息。

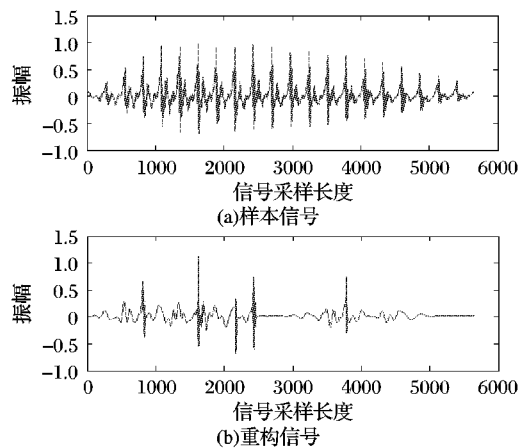


图2 样本信号重构

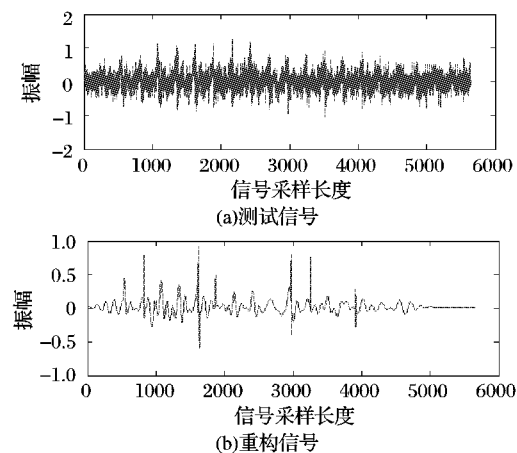


图3 测试信号重构

表1列举了重构原子选择为3,5,10的情况下,判别一个语音信号所花费的计算时间以及判别正确率。由于原子的时频参数集为 $\gamma = (s, u, v, w)$,所以一个原子只需4个参数就可以确定。

表1 不同原子个数下的匹配性能对比

原子个数	参数个数	计算时间/s	判别正确率/%
3	12	0.0045	71
5	20	0.0087	79
10	40	0.0161	94

由表1可以看出,对此类限定输入词汇量的问题,只需要

少数的几个原子便足以区分信号的结构特征,达到较高的判别正确率。

4 结语

本文针对目前语音匹配算法普遍存在的判别形式复杂,匹配参数多,计算复杂的问题,提出利用信号的 MP 稀疏分解方法来构造信号的匹配参数。由于 MP 稀疏分解方法可有效地滤除信号中的非结构误差,用较少的原子便可表现信号的结构特点,因此该方法在模式匹配中具有研究潜力和应用价值。文章对该方法进行了仿真试验,测试程序中 MP 稀疏分解的寻优算法采用的是基于 GA 和库划分的寻优策略,这增加了算法的运行效率。仿真试验表明以信号 MP 稀疏分解的最佳原子参数进行语音识别在基本的 LGB 判别算法下是有效的。如何从最佳原子参数中选取最优的判决向量以及选用一种什么样的判别算法以进一步提高判别准确率、减小运算量是需要继续研究的问题。

参考文献:

- [1] 林琳, 王树勋, 郭纲. 短语音说话人识别新方法的研究[J]. 系统仿真学报, 2007, 19(10): 2272-2275.
- [2] LI DENG, DONG YU. Use of differential cepstra as acoustic features in hidden trajectory modeling for phonetic recognition [C]// ICASSP 2007. Honolulu: IEEE Press, 2007: 445-448.
- [3] MALLAT S, ZHANG Z. Matching pursuit with time-frequency dictionaries [J]. IEEE Transactions on Signal Processing, 1993, 41(12): 3397-3415.
- [4] SHASHANKA M V, RAJ B, SMARAGDIS P. Sparse overcomplete decomposition for single channel speaker separation [C]// IEEE International Conference on Audio and Speech Signal Processing. Honolulu: IEEE Press, 2007: 15-20.
- [5] 王建英, 尹忠科, 张春梅. 信号与图像的稀疏分解及初步应用[M]. 成都: 西南交通大学出版社, 2006.
- [6] 王建英, 尹忠科. 基于稀疏分解的弱电信号检测方法[J]. 铁道学报, 2007, 29(2): 114-117.
- [7] ROSA M, VENTURA F, VANDERGHEYNSE P. Matching pursuit through genetic algorithms [EB/OL]. [2008-10-10]. <http://lts2www.epfl.ch/~figueras/GeneticAlgorithm.pdf>.
- [8] ARTHUR P L, PHILIPES C L. Voiced/unvoiced speech discrimination in noise using gabor atomic decomposition[C]// IEEE International Conference on Acoustics, Speech and Signal Processing. Hong Kong: IEEE Press, 2003: 820-823.
- [9] 高瑞, 徐华楠, 胡钢. 基于 GA 和过完备原子库划分的 MP 信号稀疏分解算法[J]. 科学技术与工程, 2008, 8(4): 914-917.

(上接第1553页)

返回充电功能,所以对产品的开发有一定的推广意义。值得提出的是,本文在分块时没有利用横向划分的优势,使得地图中宽阔的自由区域分块太多,增大了深度优先遍历的开销。如果利用横向划分的优势,对分块进一步优化组合,来减少自由区域划分的数量,便可以减少最终树枝的深度,优化深度优先算法。其次,房间中的物品摆放是建立在静态的基础上,如何应对动态的摆放,以及分区更少和返回路径更直等难题,这些将是下一步的研究工作。

参考文献:

- [1] 艾延廷, 杨明媛, 崔静, 等. 智能吸尘器控制系统软件设计[J]. 测控技术, 2007, 26(2): 73-75, 80.

- [2] 胡跃明, 丁维中, 吴析生. 吸尘机器人的研究现状与展望[J]. 计算机测量与控制, 2002, 10(10): 631-634.
- [3] 肖海荣, 富文军, 张晓军, 等. 基于多传感器数据融合的移动机器人导航[J]. 系统工程与电子技术, 2001, 23(7): 66-99.
- [4] 李群明, 能蓉, 褚健. 室内移动机器人定位方法研究综述[J]. 机器人, 2003, 25(6): 560-567, 573.
- [5] 阮晓刚, 徐绍敏. 一个室内清洁机器人的区域遍历与地图绘制[J]. 机器人技术与应用, 2006(4): 47-52.
- [6] 赵国玲, 任文娟. 数据结构[M]. 北京: 机械工业出版社, 2005.
- [7] 殷剑宏, 吴开亚. 图论及其算法[M]. 合肥: 中国科学技术大学出版社, 2003.
- [8] 周忠荣. 离散数学及其应用[M]. 北京: 清华大学出版社, 2007.