公开

分类号：\_\_\_\_\_\_\_\_\_\_\_\_ 密　　级：\_\_\_\_\_\_\_\_\_\_\_\_\_\_

10142

U D C： \_\_\_\_\_\_\_\_\_\_\_\_ 单位代码：



**硕士学位论文**

基于单目视觉的室内移动机器人导航技术研究与实现



论 文 编 号：\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

学 科 名 称：\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**2021年7月14日**

# 摘要

导航是实现机器人自主移动不可缺少的技术，一直以来是机器人研究领域的一个重要方向。尽管基于全球定位系统（Global Positioning System，GPS）的导航已取得较好的结果，由于信号遮挡及定位误差大等原因，并不适用于室内、巷道等狭小环境中的移动机器人导航。同时定位与制图（Simultaneous localization and mapping, SLAM）是一种可以解决狭小环境移动机器人导航的方法。然而，传统的基于卡尔曼滤波的SLAM算法难以满足实际需求。动物却具有惊人的导航能力，本文搭建了一个模仿动物导航机制的类脑导航系统。该系统基于Michael等提出的一种模仿大鼠(Rat) 海马计算模型SLAM（RatSLAM）算法。RatSLAM采用大脑导航机制中的头朝向细胞、位置细胞以及栅格细胞编码机制，模仿了动物的导航能力。但是，该方法采用简单的模板匹配的方法实现场景识别，以完成闭环检测，对于环境信息单一的场景容易出现误匹配，进而导致制图失败。

本文采用ORB算法对环境图像进行特征提取，生成描述子，采用描述子间的汉明距离来度量图像间的相似性，以提高图像匹配的准确率。然后将描述子聚类成单词，建立树状字典，将原算法中顺序查询图像检索转换为树状字典查询，提高了图像检索的速度。在公开数据集和真实场景中实验结果表明，本文提出的回环检测方案提高了RatSLAM算法的图像匹配的准确率与召回率，与此同时所提方法不会随着机器人在环境中探索时间的增长，而增加检索时间，确保始终能达到实时处理。同时本文在改进的RatSLAM基础上搭建了一套基于单目视觉的移动机器人自主导航系统，实现了基于类脑认知地图的移动机器人导航。从所构建的认知地图中，使用迪杰斯特拉（Dijkstra）算法，实现全局路径规划，寻找当前位置到目标位置的最优全局路径，采用动态窗口法（dynamic window approach，DWA）完成局部路径规划，实现了移动机器人沿着全局最优路径避开动态障碍物到达目标位置的目的。

关键词：移动机器人，RatSLAM，认知地图，ORB，导航

# Abstract

Navigation is an indispensable technology for achieving autonomous robot movement and has been an important direction in the field of robotics research. Navigation based on GPS system achieves better results, it is not suitable for mobile robot navigation in narrow environments such as indoor and alleyways due to occlusion and large positioning errors. Simultaneous localization and mapping (SLAM) is a method that can solve the problem of mobile robot navigation in indoor environments. However, the current conventional Kalman filter-based SLAM algorithms still cannot meet the practical needs. This study builds a brain-like navigation system, which mimics animals’ navigation mechanism because animals have amazing navigation ability. The core of the study is an improved RatSLAM algorithm and a cognitive map-based navigation method. RatSLAM uses the head-facing cells, location cells, and raster cells encoding mechanisms in the brain navigation mechanism to mimic the navigation ability of animals. However, the method uses a simple template matching approach to achieve scene recognition and complete closed-loop detection. For scenes with single environmental information, it is prone to generate scenes mismatching and mapping failure.

Based on the cognitive map, this study further implements the navigation of mobile robots. Present study also proposed an image matching algorithm, which was based on ORB algorithm and bag-of-words model, to achieve fast and accurate scene recognition. The ORB algorithm was employed to generate the eigenvectors of the scene images. Therefore, image matching was converted into the calculation of the distance between the eigenvectors, which improves the accuracy of image matching. Additionally based on the bag-of-words model, the eigenvectors cluster into words with a search tree, which not only improves the speed of image matching but also reduces the amount of storage space. The proposed image matching algorithm was applied to improve the success rate of RatSLAM for constructing cognitive maps. Meanwhile, this thesis realizes autonomous navigation of mobile robots based on cognitive maps. According to the constructed cognitive map, Dijkstra algorithm was used to search the optimal global path from the current position to the target position, and the dynamic window method was used to avoid dynamic obstacles. Ultimately the mobile robot could reach the destination with the successful avoidance of dynamic obstacles along the global optimal path.

**Key Words:** Mobile robot, RatSLAM, Cognitive Map, ORB, Navigation

# 目录

[摘要 I](#_Toc80537617)

[Abstract II](#_Toc80537618)

[第1章 绪论 1](#_Toc80537620)

[1.1 研究背景及意义 1](#_Toc80537621)

[1.2 国内外研究现状 3](#_Toc80537622)

[1.2.1 SLAM概述 3](#_Toc80537623)

[1.2.2 视觉SLAM研究现状 4](#_Toc80537624)

[1.2.3 移动机器人类脑导航研究现状 4](#_Toc80537625)

[1.3 论文主要内容 5](#_Toc80537626)

[1.4 论文的组织结构 6](#_Toc80537627)

[第2章 相关技术概述 8](#_Toc80537628)

[2.1 空间认知模型 8](#_Toc80537629)

[2.2 RatSLAM算法模型 9](#_Toc80537630)

[2.2.1 局部细胞模型 10](#_Toc80537631)

[2.2.2 位姿细胞模型 11](#_Toc80537632)

[2.2.3 经验制图 12](#_Toc80537633)

[2.3 ORB算法 14](#_Toc80537634)

[2.4 本章小结 17](#_Toc80537635)

[第3章 RatSLAM模型优化 18](#_Toc80537636)

[3.1 回环检测方案 18](#_Toc80537637)

[3.1.1 特征提取与匹配算法 20](#_Toc80537638)

[3.1.2 图像检索模型 21](#_Toc80537639)

[3.1.3 闭环决策模型 24](#_Toc80537640)

[3.2 RatSLAM优化方案 29](#_Toc80537641)

[3.3 实验与分析 30](#_Toc80537642)

[3.3.1 实验数据 30](#_Toc80537643)

[3.3.2 评价标准 31](#_Toc80537644)

[3.3.3 分析与对比 33](#_Toc80537645)

[3.4 本章小结 36](#_Toc80537646)

[第4章 基于改进的RatSLAM导航 37](#_Toc80537647)

[4.1 全局路径规划 38](#_Toc80537648)

[4.2 局部路径规划 39](#_Toc80537649)

[4.3 实验与分析 41](#_Toc80537650)

[4.4 本章小结 43](#_Toc80537651)

[第5章 现实场景中应用 44](#_Toc80537652)

[5.1 系统总体结构 44](#_Toc80537653)

[5.2 系统实现 45](#_Toc80537654)

[5.3 实验结果 46](#_Toc80537655)

[5.4 本章小结 48](#_Toc80537656)

[第6章 总结 49](#_Toc80537657)

[参考文献 51](#_Toc80537658)

[致谢 54](#_Toc80537659)

# 第1章 绪论

## 1.1 研究背景及意义

从简单、枯燥、重复性的劳动中解放出来是人们一直追求的梦想，也是人们创造发明机器人的主要目的。机器人具有可移动的能力后，可以进一步扩大其使用范围并能更好地提高其使用效率。但是如何使机器人在复杂的环境中像人一样自由移动，实现机器人的自主导航功能一直是难以解决的问题。20世纪中后期，人们进行了对人工智能与移动机器人关联的探索，从而使人工智能推动了移动机器人导航技术的快速发展。目前，移动机器人的导航技术已取得了一系列突破性进展，极大地推动了移动机器人在工业、商业、日常生活等场景中的应用，为先进技术的发展与社会的进步做出了卓越的贡献。如图1.1（A）所示仓储物流中搬运货物的移动机器人，通过相机检测地面上的二维码实现机器人的导航功能；如图1.1（B）所示扫地机器人，使用相机采集环境中的信息，通过对图像的处理实现了移动机器人的自主导航功能；如图1.1（C）所示，使用激光传感器采集周围环境信息构建栅格地图，实现移动机器人的自主运动功能；如图1.1（D）所示小米扫地机器人，使用激光和惯性测量单元融合的信息，实现移动机器人定位及导航功能。

|  |  |
| --- | --- |
|  | d009b3de9c82d158edc87bda5d018addbc3e4227 |
| （A）智能仓储机器人 | （B）扫地机器人 |
|  | u=965323876,2729505039&fm=26&gp=0 |
| （C）激光导航机器人 | （D）小米扫地机器人 |

图1.1 室内移动机器人

Fig. 1.1 Different kinds of robots

移动机器人导航需要解决三个关键问题：地图构建、定位及路径规划。全球定位系统（Global Positioning System，GPS）依赖卫星技术为移动机器人实现导航功能提供了精准的位置信息。但是在室内、地下等有遮挡的环境中，通信传播就会影响，无法为机器人提供精确的位置信息。在这种情况下移动机器人因缺失外源性定位信息变成“盲人”，无法准确执行导航任务。虽然此时可以使用航迹推算来估计机器人的位置信息，但是在真实环境中，机器人的运动存在漂移，基于航迹推算的位置估计累计误差会不断的增大，使得估计位置与真实位置具有较大偏差。因此在GPS等外源性信息不能为移动机器人长期、稳定的提供精确位置信息时，如何使移动机器人在未知的环境中实现导航任务是问题的关键。

同步定位与地图构建（Simultaneous localization and mapping，SLAM）是指移动机器人在未知环境中[11]，通过自身携带的传感器感知环境中的信息，在环境探索的同时完成环境地图的构建，并根据构建的地图信息定位自身在地图中的位置，在不断的探索中完成环境地图精准的构建。SLAM的核心问题是，如何解决在没有任何先验知识的情况下构建出环境的地图信息，减少导航系统对外源性定位信息的依赖程度，是实现室内移动机器人导航的关键技术之一。

在过去几十年里，人们探索了基于各种传感器的SLAM方法，其中大部分都采用激光雷达、深度相机等昂贵精密的传感器构建栅格地图、点云地图等高分辨率的环境地图，同时使机器人能够在环境中精确地定位自身的位置。高精度的传感器成本高，且由于需要处理的信息量大，对计算能力和存储空间都有较高的要求，阻碍了其在日常应用中的推广应用。

为了改进传统的移动机器人导航技术，学者们注意到了自然界中动物超高的导航本领。在自然界中动物的感知能力有限，不能像激光雷达、声纳及惯性测量单元等高精度传感器一样精确的测量环境信息，却拥有惊人的导航能力，因为生物对地图的构建依赖于鲁棒性处理策略而不是对环境的精确描述。例如，大鼠的视觉感知系统无法像RGB-D传感器那样精确地测量环境中自身到达物体距离信息，也没有丰富的颜色信息，却可以精确的在方圆几米范围内觅食并顺利地返回巢穴。经过研究发现，在动物的大脑内存在一个高效、稳定的“GPS”导航系统[12]。如果在移动机器人的控制中枢里也融入类脑导航处理机制，这就能够使移动机器人摆脱高成本的束缚。模拟哺乳类动物在非结构化环境中的导航能力，类脑导航的出现初步实现了这一设想。类脑导航是模拟哺乳类动物脑内导航机制启发而产生的一种新型导航方案，将导航技术、人工智能与生物学研究相互融合，提高移动机器人的自适应能力。

综上所述，在众多传感器中，单目视觉传感器具有价格低、结构简单、体积小等优点，基于单目视觉的导航技术在室内移动机器人上仍是一个重要的研究方向。在此背景下，本文借鉴哺乳类动物大脑导航机制，将导航技术、人工智能和生物学研究相互融合，提高移动机器人的自适应能力，实现基于单目视觉的室内移动机器人的导航功能。

## 1.2 国内外研究现状

### 1.2.1 SLAM概述

移动机器人在陌生的环境中，首先通过自身携带的传感器探索环境信息进行地图构建，同时使用地图信息定位机器人在该环境中的位置信息，在不断的探索中完成环境地图的精确构建。随着对SLAM研究的不断完善，将已有的算法归纳总结为两大类。

第一类，基于滤波的SLAM算法，其中最成功的三种概率算法为卡尔曼滤波算法[13]、最大期望算法和粒子滤波算法[14]。卡尔曼滤波使用递归方法解决离散数据线性滤波问题，它是由一组数学方程提供的一种计算（递归）方法，以最小化误差的方式估计当前状态的过程。粒子滤波也称为蒙特卡洛定位（Monte Carlo Localisation，MCL），这种算法利用带权重粒子的分布来估算位置分布，故无需考虑后验分布所来带的形状或约束。FastSLAM是一种高效的SLAM算法，主要有由扩展卡尔曼滤波和粒子滤波组成，一个用于地标估计，另一个用于机器人位置估计。

第二类，基于图优化的SLAM算法，如GraphSLAM、ORB-SLAM、VINS算法。GraphSLAM算法是第一个采用全局图优化的SLAM算法，图中的顶点代表机器人的位姿信息，边代表位姿之间的关系。通过编码器或者闭环检测提供位姿关系优化全局图，从而消除移动机器人产生的累计误差。ORB-SLAM算法是目前基于图像特征点匹配的最流行SLAM算法，该算法包含两个图，小图用于局部BA的优化，大图负责全局位姿图的回环检测与优化[15]。

这两类算法都是为了消除计算偏差而提出的，区别在于基于滤波的SLAM算法是通过串行计算递推估计误差，所以很难消除累计误差；而基于图优化的算法是从不同时间维度出发并行估计测量值，寻找全局最优值，累计的误差增长缓慢。这些方法在移动机器人导航应用上都取得了较好的研究成果，因为在计算之前他们都需要对传感器的测量以及机器人运动学做出较强的先验假设，所以这些方法存在鲁棒性较差的缺陷。当移动机器人的传感器出现偏差或者运动学特性发生变化时，使用的SLAM算法都会受到较大的影响，甚至出现不可逆的性能发散问题。现有的SLAM算法通常是把导航、感知、控制模块分开设计。当在动态环境中时，传统的导航算法无法通过感知信息与导航信息自主生成决策指令，从而无法解决导航过程中突发情况，因此传统的SLAM算法通常只适用于特定的某一类场景，导致可扩展性较差。

### 1.2.2 视觉SLAM研究现状

在室内环境特征信息丰富的情况下，使用相机比激光能够提取到更多有用的信息。目前基于视觉SLAM算法主要分为三类，单目视觉SLAM、双目视觉SLAM、RGB-D视觉SLAM，其中单目相机价格低、体积小、操作简单等优点深受人们的喜爱，因此对单目视觉SLAM研究是一个重要的方向。

Davision等人在2007年第一次提出了基于实时单目视觉的MonoSLAM[16]算法，该算法对实时性要求特别高，因此在计算机性能不足的情况下，需要对采集的图像进行有选择性的稀疏处理，同时对路标的存储数量有限，容易导致特征点丢失不利于构建环境的地图。但该算法具有实时性，有利于机器人的自主导航功能，后来的很多视觉SLAM算法都在此基础上提出。PTAM是基于关键帧的实时单目视觉SLAM算法，通过对关键帧的处理估计机器人的位姿[17-18]。该算法首次将Track和Map分开作为视觉SLAM算法的前端特征跟踪和后端地图优化的两个线程同时进行的。Newcombe等人提出了基于单像素，建立稠密3D点云地图的DTAM算法[19]。Mahon等人提出了一种基于扩展信息滤波器，适用于大规模环境中导航的SLAM算法[20]。Engl等人提出了基于直接法的LSD-SLAM算法[21]，增加了单目相机的应用范围。Mur-Artal等人提出了基于PTAM改进的ORB-SLAM算法[15]，该算法是目前最为流行的单目视觉SLAM算法，之后又扩充了双目相机和RGB-D相机，该算法主要提取ORB特征点，实时建立三维点云地图。

### 1.2.3 移动机器人类脑导航研究现状

传统的单目视觉SLAM算法无法更好的实现移动机器人导航功能，然而自然界动物具有惊人的导航本领。例如只用302个神经元的线虫依然能够凭借气味避开危险物寻找到食物。大鼠的视觉感知系统无法像RGB-D传感器那样精确地测量环境中自身到达物体距离信息，也没有丰富的颜色信息，却可以精确在方圆几米范围内觅食并顺利地返回巢穴。因此模拟动物导航的本领使移动机器人在没有高精度传感器下，依然能够实现自主导航功能，为移动机器人导航研究提供了新方向、新思路。

Burgees等人根据海马体中位置细胞的工作原理提出了一种仿生空间表示模型，并在微型机器人上验证了位置细胞放电对于地图构建与导航的有效性。Yuan等人在移动机器人上实现了基于位置细胞和栅格细胞工作机制的实时定位系统[22]。DeepMind公司利用人工智能复现了哺乳类动物的大脑导航功能，提出了栅格表示的矢量导航模型并将其应用在虚拟迷宫平台中，实现了虚拟移动机器人在迷宫中的自主探索、环境感知、空间多尺度表达以及目标点导航任务。Fleischer等人通过模拟栅格细胞的工作方式构建前馈连接结构，利用位置细胞的表征模式来反向监督栅格细胞网络的学习，提高移动机器人实时定位技术的稳健性。Zeng等人利用栅格细胞的神经编码机制，构建了位置和速度连个编码的栅格细胞模型，并在St Lucia Suburb数据集上完成了移动机器人的SLAM任务[23]。

2014年诺贝尔生理学或医学奖获得者发现了基于动物导航机制的大脑定位系统细胞。John O’Keefe等人开创了认知地图的构建，在机器人上采用细胞模型进行认知地图的构建[24]。Michael J. Milfor等人根据生物学发现的头朝向细胞、位置细胞和网格细胞，采用连续吸引子网络模型大脑中的细胞发放原理提出了RatSLAM算法，为以后的人提供了经验，该方法采用半米制拓扑图来表示建立的环境地图[25-26]。田博等人在RatSLAM算法的基础上首次采用RGB-D摄像头完成认知的构建，使用扫描强度和深度信息融合的方式改进了回环检测的准确性[27]。Guangzhi Tang等人采用脉冲神经网络代替连续吸引子网络，实现了移动机器人的认知地图的导航构建[28]。张潇等人在RatSLAM算法的基础上引入光学双轴传感器和MIMU信息，建立了航位推算模型[29]。陈孟元等人使用HSV图像代替RGB图像完成图像特征的提取基于匹配提高了RatSLAM算法的回环检测的准确率[30]。唐华锦等人采用神经网络的方式对栅格细胞的连接权重进行更新[31]。于乃功等人在RatSLAM算法中加入了条纹细胞对速度进行积分估计机器人的位置，同时采用FAST特征提取和SURF描述子组成的回环检测模型对原算法进行了改进[32]。

RatSLAM算法在感知图像处理的过程中使用的是绝对差总和模型（SAD）进行计算。计算方式比较简单，不需要对图像进行复杂的处理和计算，只是简单地把新的图像转为灰度图像，然后计算对应的像素差值，归一化求的平均误差是否大于设置的阈值，进行判断图像是否相似。这种计算方式容易受噪声的干扰存在较多的误匹配。因此本文针对RatSLAM算法中视觉感知能力差的问题进行了深入的研究，设计了一种融合特征匹配的方法，提高了RatSLAM回环检测的准确率和召回率。

## 1.3 论文主要内容

本文针对RatSLAM算法中视觉感知能力差的问题，设计了一种新的融合特征提取与树检索策略的图像匹配方法，提高了原RatSLAM算法闭环检测的准确率和召回率，改善了制图质量，并利用改进的RatSLAM算法搭建了一个移动机器人导航系统，实现了基于类脑认知地图的导航，在公开数据集和真实的室内环境中验证了所搭建系统的有效性。本文的主要贡献总结如下：

（1）结合快速视觉特征提取方法Oriented Fast and Rotated Brief（ORB）和视觉词袋模型（Bag of Word，BOW），提出了一种新的高效准确的图像匹配方法改进了RatSLAM算法的视觉感知模块，提高了算法回环检测的准确率和召回率。采用ORB算法对环境图像进行特征提取并生成描述子，以提高图像匹配的准确率；然后将描述子聚类成单词，建立树状字典，将图像检索转换为查询字典得到的向量间距离的计算，提高图像检索的速度，以实现移动机器人实时建图。

（2）基于改进的RatSLAM算法，搭建了一套基于单目视觉的移动机器人自主导航系统，实现了基于类脑认知地图的移动机器人导航。基于改进的RatSLAM方法构建的认知地图，使用迪杰斯特拉（Dijkstra）算法，实现全局路径规划，寻找当前位置到目标位置的最优全局路径，采用动态窗口法（dynamic window approach，DWA）完成局部路径规划，实现了移动机器人沿着全局最优路径避开动态障碍物到达目标位置的目的。

（3）搭建了仿真实验环境和真实实验平台，在公开数据集上和真实室内环境中验证了本文改进RatSLAM算法制图的优越性；在仿真环境和真实室内环境中验证了基于类脑认知地图的导航方法的可靠性。

## 1.4 论文的组织结构

论文总体分为六个章节，分别为绪论、相关技术概述、RatSLAM模型优化、基于改进的RatSLAM算法导航、现实场景中应用以及总结，本文的章节安排如下：

第1章：绪论。本章主要介绍了本文的研究背景及意义，同时本章节中对SLAM建图及导航的研究现状加以阐述，同时阐明了提高回环检测的准确率和召回率对RatSLAM的重要性。最后又介绍了本文研究内容及解决的问题。

第2章：相关技术介绍。结合论文的研究内容介绍了哺乳类动物大脑中的空间认知机理及空间计算模型，包括栅格细胞、位置细胞和头朝向细胞指引哺乳类动物导航的原理。然后介绍了RatSLAM模型，通过融合头朝向细胞和栅格细胞编码环境的位置信息，使用机器人车轮编码器提供的速度信息驱动模型中的能量包完成路径积分，估计机器人的位置信息。当遇到熟悉的场景时，通过对模型注入能量来模拟细胞发放实行重定位功能。最后又介绍了ORB特征提取算法，对图像信息检测描述子、提取特征并生成描述子的方法。

第3章：RatSLAM模型优化。本文提出采用ORB算法和BoW算法构建的回环检测模型提高RatSLAM算法中回环检测的准确率和召回率。使用ORB算法提取图片特征，并采用汉明距离算法计算图像之间的相似性，提高图像匹配的准确率及召回率，改进回环检测功能。使用BoW算法把图像特征点聚类成单词，然后又采用树的方式存储单词，每个单词都有一个逆索引表维护图像之间的关系，加快了图像的检索速度，达到了实时处理图像的要求。同时为了改进RatSLAM稳定的构建环境地图，本文又加入了关键帧的筛选、动态阈值的选定、岛的匹配方式、序列匹配检测是否是正确的回环判断以及时间一致性检测，确保回环检测的准确率。

第4章：基于改进的RatSLAM导航。基于改进的RatSLAM算法构建的认知地图，使用Dijkstra算法，实现全局路径规划，寻找当前位置到目标位置的最优全局路径，采用DWA算法完成局部路径规划，实现了移动机器人沿着全局最优路径避开动态障碍物到达目标位置。

第5章：现实场景中应用。首先详细介绍了类脑导航系统实现环境、系统的总体设计结构和软件模块设计，其中软件模块设计又分为局部视图细胞模块设计、位姿细胞模块设计、经验制图模块设计和导航模块设计。然后详细介绍了实验系统平台的软硬件设备及实验环境。最后在室内环境中验证本文改进算法的可行性，实验结果表明在室内环境中本文搭建的类脑导航系统，可以自主移动到达目标位置，实现了移动机器人导航功能。

第6章：总结。对论文的研究内容做了详细的概括与总结，描述了在研究过程中所遇到的问题，并分析了论文进一步改进的方向。

# 第2章 相关技术概述

近年来，来自人类和动物大量的神经生理学实验结果帮助人类揭开了空间记忆机制和大脑结构的神秘面纱。尽管可以从一种认知科学的角度来研究空间认知，但是行为层面的跨学科分析以及潜在的神经机制为人类提供了更坚实的基础和证据。科学家对啮齿类动物的大脑进行了大量的实验，发现了头朝向细胞、位置细胞、栅格细胞、加速度细胞、边界细胞、条纹细胞等这些神经元细胞。这些细胞在啮齿动物的大脑内构建了一套高效、稳定的“GPS”导航系统。

## 2.1 空间认知模型

（1）位置细胞

1971年伦敦大学学院的美国科学家John O’Keefe等研究团队经过大量的研究实验表明海马体中存在一种特殊的神经细胞，这种细胞在大鼠处于空间某个特定位置时，海马体中的一些神经细胞会产生强烈的放电反应。但当大鼠处于其他位置时就几乎没有放电反应，这两者相差往往高于10-100倍，后来就把这种在海马体中的神经细胞命名为“位置细胞”[33]。位置细胞的放电活动状态表征了大鼠在环境中的某个地点的记忆，如图2.1所示。位置放电反映了动物的位置，而不是在这个位置发出的行为，不同的细胞在不同的位置有位置野，整合起来它们覆盖了整个测试环境。

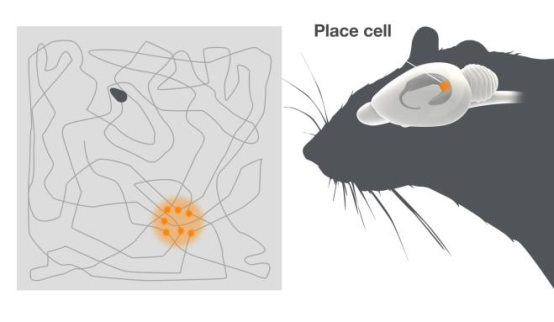


图2.1 位置细胞

Fig. 2.1 Place Cells

（2）头朝向细胞

1984年纽约大学James Ranck等人在大鼠前下托中发现了一类与动物头部朝向直接相关的神经元细胞，并被命名为头朝向细胞（head direction cell）[34]。经过实验研究发现大鼠脑内的每个头朝向细胞都会编码一个方向，当大鼠的头朝向一个方向时，相应的神经元细胞开始放电，当大鼠的头偏向另一个方向时，另一片对应的神经元细胞开始放电，而之前的神经元放电将会减弱，大鼠头的朝向改变90°时，头朝向细胞的放电频率将将为0，如图2.2所示。头朝向细胞在很多动物的脑内都被检测到，如蜜蜂，蚂蚁等生物。

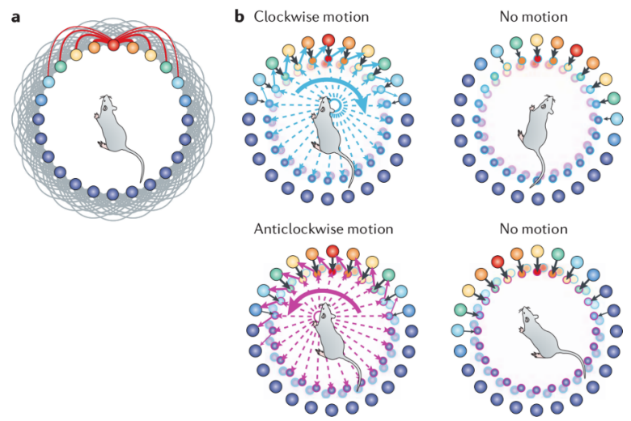


图2.2 一维吸引子网络头朝向细胞在圆上的排布

Fig. 2.2 One-dimensional attractor map model for head direction encoding

（3）栅格细胞

May-Britt Moser和EdvardMoser从内嗅皮层中的记录中识别出栅格细胞[35]，在多个位置发射的神经元，它们以周期性的三角形图案细分整个表面，每个三角形由多个参数来描述：间距，方向和相位，如图2.3所示。研究表明在足够大的实验环境中，许多内嗅皮层细胞在整个环境中均以规则的间隔显示出呈栅格状的结构，这在海马体的任何地方都看不到，该栅格具有六边形形状，内角为60°和120°。栅格单元网络定义的空间坐标系似乎已锚定到单个环境的特定地标。

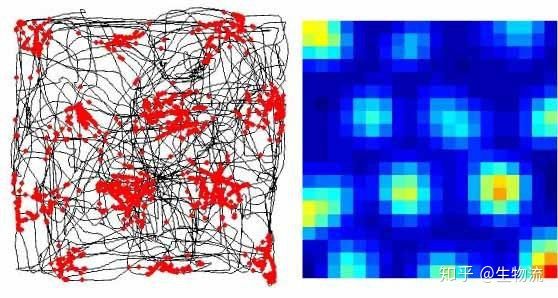


图2.3 栅格细胞

Fig. 2.3 grid cells

## 2.2 RatSLAM算法模型

Gordon F.Wyeth和Michael J.Milford通过对大鼠的一系列研究观察，提出了一种基于海马计算模型的RatSLAM系统，该系统模仿大鼠的神经行为学特征，能够在真实的环境中构建半米制拓扑地图。RatSLAM算法主要分为三个部分组成，如图2.4所示，分别为局部视图细胞（Local view cells）、位姿细胞（Pose cells）、经验制图（Experience Map）。局部视图细胞处理机器人的视觉信息，位姿细胞由竞争神经网络构成，利用细胞状态的变化，完成路径积分的功能来估计机器人的位置。经验制图采用二维空间平面的形式存储环境的信息，通过视觉信息定位机器人的位置，当检测到机器人经过先前位置时，进行位置校正优化构建的环境地图。

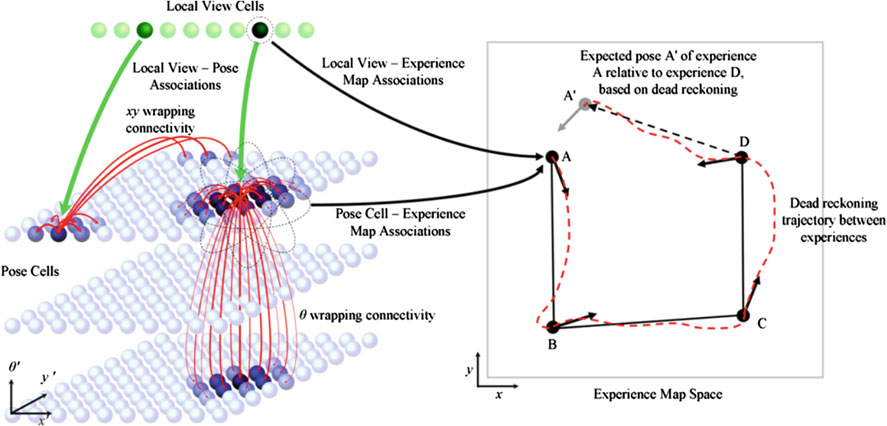


图2.4 RatSLAM系统主要模块

Fig. 2.4 The major modules of the RatSLAM system

### 2.2.1 局部细胞模型

局部视图细胞是一个可扩展的单元数组，每个单元代表环境中不同的视觉场景。RatSLAM通过机器人在移动过程中不断地感知环境，以场景模板的形式形成机器人的经验记忆，这些有序的模板逐渐储存成机器人的视觉场景模板库，在机器人到达熟悉的场景时，能够迅速地从模板库中与之匹配。而到达一个新的场景时，则会将当前场景以模板的形式储存起来并添加入模板库之中，形成新的‘记忆’。

在RatSLAM中，局部视觉模板是一组神经单元，它们以映射和定位系统可用的形式表示对机器人外部传感器的输入。局部视图细胞和位姿细胞之间同时发生两种相互作用：通过联想学习进行映射，以及通过向位姿细胞注入能量来定位。当看到一个新的视觉场景时，将创建一个新的本地视图单元，并与该场景中的原始像素数据相关联。如图2.5显示了局部视图模块和姿态单元结构以及它们之间形成的链接。

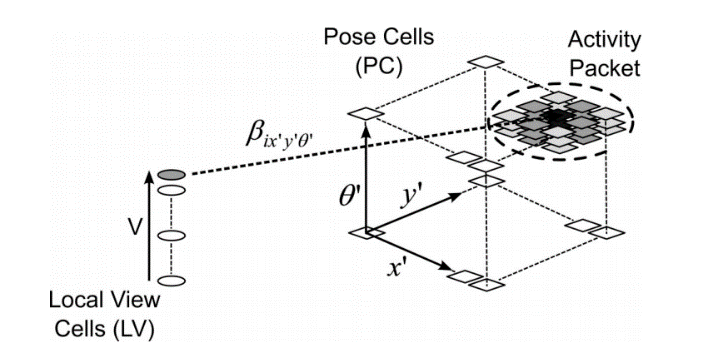


图2.5 局部视觉模块与位姿细胞间联系示意图

Fig. 2.5 The positional diagram of the local visual modules and the pose cells

视觉场景通过学习函数与机器人相信的位置之间学习关联链接，这增加了共同激活的局部视图和姿态单元之间的连接强度，主动的局部视图单元通过这些链接向姿态单元注入能量。连接强度的更新如下公式所示：

 （1.1）

其中是局部视觉单元的活动强度，是当前时刻位姿单元的活动能量强度。

局部视图和位姿单元之间的关联链接会将活动能量注入位姿细胞单元，这是RatSLAM用来实现视觉场景重新定位的功能。机器人通过模板匹配的方式检测当前局部场景与场景序列之间的相似性，在匹配出一个熟悉的局部视觉单元时，会激活与之匹配的局部视图单元，该局部视图单元将活动注入到与其关联的位姿单元中，能量注入量与关联强度成正比。位姿单元的活动能量变化公式如下公式所示：

 （1.2）

其中，是局部视觉校正常量。

### 2.2.2 位姿细胞模型

RatSLAM算法中最核心的部分，Milford等人将用于模拟头朝向细胞和位置细胞的竞争吸引子网络结构融合起来构建了一个三维的吸引子网络模型，并将其命名为位姿细胞，位姿单元的三个维度分别由位置和转向角度表征，如图2.6所示，并由该吸引子模型编码机器人的位姿估计。

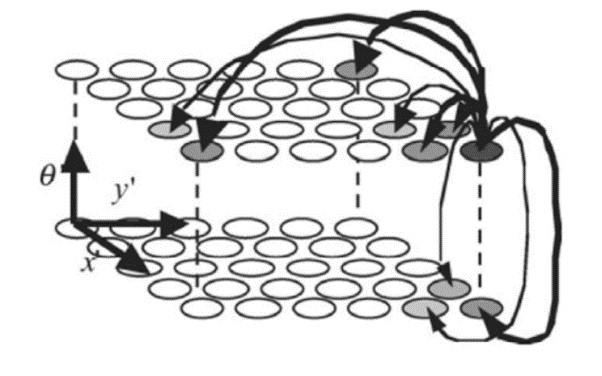


图2.6 位姿细胞三维吸引子模型

Fig. 2.6 The attractor model of pose cells in three dimensions

RatSLAM用CAN模型（Continuous Attractive Net-Work）来构建三维位姿吸引子网络，位姿细胞中的每个位姿单元的活动范围都是在0到1之间。每个单元中的活动编码机器人具有与该单元相关联的特定姿势的概率。在单个位姿单元的基础上，这没有太多的意义，但当被看作一个整体时，集成活动可以被解码，而聚集的单元簇称为活动包或能量包，而能量强度最大的能量包的质心提供了机器人最有可能的位姿的估计。活动最强烈的单元簇将编码机器人最可能的姿态。其他不太占主导地位的单元簇编码关于机器人姿态的其他可能假设。

在该吸引子网络中，位姿单元之间具有相互的兴奋性连接以及抑制性连接，并且这种连接是无视位姿网络的边界的。这种动态连接的行为是通过局部兴奋性、全局抑制性连接来实现的，与头部方向和位置细胞网络一样，竞争吸引子网络动态控制姿态网络中的活动。其内部动力学可以分为三个阶段，局部的兴奋度更新、全局抑制性连接、能量归一化实现。

RatSLAM利用三维离散高斯分布建立兴奋性权重矩阵，每个位姿单元活动都会对其它位姿单元其刺激作用，位姿单元离该单元越近，刺激作用越强，反之越弱，在位置处的刺激加权矩阵表示为：

 （1.3）

其中和是代表位置网络和头朝向网络的常数。*a*、*b*和*c*分别表示距离*x*、*y*和*θ*位姿坐标之间的距离。通过局部兴奋性连接，每个位姿单元的活动强度能够得到更新，其更新量为每个位姿单元对其的刺激的求和，可以表示为：

 （1.4）

其中代表（*i*，*j*，*k*）位于*i*，*j*，*k*处的位姿单元的当前激活程度，、和分别代表三维位姿吸引子网络的规模，遍历所有位姿单元之后，每个位姿单元的能量将得到局部兴奋度的更新。

由于多个姿态假设（由多个活动包表示）需要时间来竞争，并通过进一步的视觉输入来增强，因此抑制相对温和这意味着竞争对手的数据包可以共存很长一段时间。与此同时，对这些位姿单元进行局部抑制性连接和全局抑制作用，抑制作用表示为：

 （1.5）

其中是与前式刺激加权矩阵类似的三维离散高斯分布的抑制加权矩阵，是全局抑制常量。

能量归一化实现是指在视觉和路径整合对网络注入能量后，保持位姿网络中激活能量的总和为1，归一化后的位姿单元的活动水平表示为：

 (1.6)

### 2.2.3 经验制图

在构建环境地图的初期，位姿单元提供的空间表示与移动中的机器人的环境的度量布局很好地对应。然而，随着里程误差的积累和视觉闭合事件的发生，位姿单元所代表的空间变得不连续-位姿网络中的相邻单元可以表示机器人真实环境中相差很远的位置。此外，由于位姿网络代表一个有限的区域，但网络边缘的连通性意味着理论上一个位姿网络可以映射一个无限的区域，这就导致一个位姿单元代表多个物理位置的情况。在这种情况下，单纯地依赖位姿网络确定机器人物理空间下的位置是不可靠的，因此RatSLAM采用经验图来表示机器人的经历地图，它将位姿单元和局部视图单元的信息结合起来表示机器人的位置，如下图2.7所示。

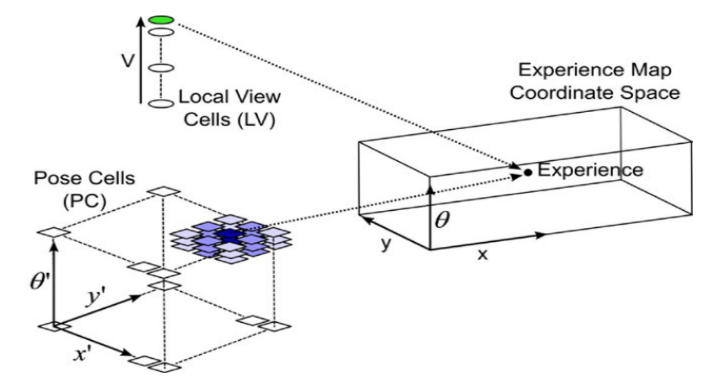


图2.7 RatSLAM经历制图示意图

Fig. 2.7 The diagram of RatSLAM experience mapping

为了得到机器人的唯一位姿估计，地图中的每一个节点都是由一个三元组构成，这个三元组由被称为经验节点（experiences）组成，环境认知地图由一系列的经验节点构成，在机器人移动过程中，相邻的两个节点之间会创建一个联系，并通过转移矩阵表示第*i*个节点和第*j*个节点间的拓扑关系。

每个经历节点由三个部分组成，分别是当前位姿网络的位姿估计，该位置处位姿单元所联系的局部视觉模板以及当前位姿单元的活动强度组成，经验节点的三元组表示为：

 （1.7）

当位姿细胞和局部视图细胞中的当前活动状态与任何现有经验相关的状态不紧密匹配时，即当目前储存的经验不足以准确表达当前机器人的位姿时，RatSLAM会创建新的节点。具体为在已有的节点集的信息中进行逐一对比判断，一个分数度量*S*用于比较当前姿态和局部视图状态与每一次体验相关的状态之间的匹配程度，根据匹配函数S的值判断是否需要创建新的节点，其函数计算为：

 （1.8）

其中代表位姿单元活动强度的系数，表示局部视觉模板的权重系数，其权重主要体现了机器人对位置的信任程度以及对场景变换的敏感程度，根据环境的不同，权重系数不同。当大于最大阈值时，机器人认为自身认知应得到添加，此时建立新的经历节点，随着机器人在经验之间的转换，在先前活跃的经验点与新的经验点之间会建立一个新的link用来建立新节点和旧节点之间的联系，形成的链接为：

 （1.9）

其中是两种经验之间的相对里程姿势，而是在两种经验之间移动的时间。机器人可以使用这些时间信息来规划从当前位置到期望目标位置的路径。可以使用里程转移信息找出到达目标位置最近的路径。通过转移矩阵，可以得到新的经历节点表示为：

 （1.10）

机器人的经历节点的不断地构建过程中，会不断产生累计误差。而RatSLAM出现位置闭环时，节点匹配函数小于最小阈值，此时不建立新的节点，认为回到之前的经验节点。RatSLAM利用图像松弛（Graph Relaxtion）算法来进行地图修正，图松弛算法在整个图中分布里程误差，提供机器人环境的地图，可以很容易地被人类解释。并能够很大程度修正机器人移动中产生的累计误差，其修正公式为：

 （1.11）

其中，是修正常量，是存储的经历节点到当前经历节点需要纠正的节点数量。图松弛算法的纠正机制是连续的，主要表现在闭环点位置的纠正。这里面的取值很重要，过大容易导致环境地图不稳定，太小容易失去纠正功能，地图变得不直观，一般情况下取0.5，此时地图能够以较快的速度收敛到稳定状态，得到的环境地图也比较直观、简洁，为后续导航提供较为优越的先验条件。

## 2.3 ORB算法

ORB（Oriented FAST and Rotated Brief）[36]算法是由Rublee在2011年提出的一种特征检测算法，该算法融合了FAST（Features from Accelerated Segment Test）[37]特征检测算法和BRIEF（binary robust independent elementary feature）[38]描述子算法，在此基础上又加入多尺度性和旋转不变性。ORB算法最著名的优点就是计算速度快，计算效率高，比SIFT（Scale Invariant Feature Transform）特征检测算法和SURF（Speeded Up Robust Features）特征检测算法快很多倍。

（1）特征提取

FAST算法提取关键点的过程是通过计算角点坐标P周围的16个像素围成的圆圈，当在圆圈内存在连续n个像素，并且他们比P处的灰度大于或小于阈值t，就把P作为一个角点。基于这条检测原则，在剩余的所有坐标中检测圆圈内的所有像素，如图2.8所示。

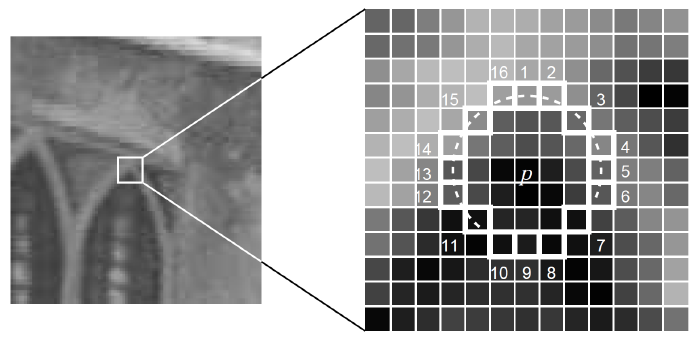


图2.8 FAST角点检测原理

Fig. 2.8 FAST corner point detection principle

针对FAST特征检测算法不具备多尺度的特性做了改进，同时增加了旋转不变性。FAST算法通过创建一个8层的金字塔图像，来完成多尺度不变性。首先对创建的金字塔中的每层图像做基于FAST算法的角点检测，根据特征点最后的得分值作为响应值，但是仅仅通过响应值的大小无法判断出特征点的好坏，因为在图像特征提取过程中有些处于边缘处的特征点响应值很大，非常不利用后期角度的提取。因此使用Harris算法对FAST算法提取出的特征点进行再一次的计算响应值，然后根据计算出的响应值对提取出的特征点进行排序，删除差的特征点只保留较好的特征点。金字塔每层图像中都要有一定数量的特征点被保留下来，因此根据FAST算法检测到的特征点数量一般为该层金字塔图像所需特征点数量的2倍。

Harris角点检测算法计算特征点响应值的方法是：把需要检测的特征点作为中心，然后把特征点周围*k**k*大小的范围内，利用Sobel算子计算图像中所有像素的*x*方向和*y*方向的梯度，把得到的*Nx*和*Ny*的值进行求和，计算公式如下所示：

 （1.12）

Harris算法计算特征点的响应值公式定义如下所示：

 （1.13）

公式中*k*表示系数，而*s*为尺度系数它的值与*k*大小相关，然后在根据强度质心的概念原理计算出特征点的方向角度。FAST算法是根据灰度质心检测出角点位置，角点强度计算是从角点中心位置出发的偏移量，这个偏移量使用矢量表示。该补丁区域一阶矩定义公式如一下所示：

 （1.14）

式中，表示（*x*，*y*）位置处的像素灰度值。通过一阶矩公式计算得到补丁区域的质心C，坐标表示为：。把特征位置作为圆心，特征点周围的补丁区域定义为邻域，质心C的角度使用补丁区域的角度表示，也就是该特征点的角度，公式定义如下：

 （1.15）

为了更近一步提高ORB算法的旋转不变性，在补丁区域内以特征点作为中心，以*r*作为半径建立一个圆形区域，然后建立弦心距*n*和弦长*m*的映射函数，当*n*的大小从0到逐渐增大时，可以得到不同长度的弦和弦上的像素，实现便利圆形内所有像素的目的。通过这种方法使用FAST算法检测的特征点就具有了角度信息。

描述符的创建，为了实现具有旋转不变性的BRIEF描述符，则需要对特征点为中心的补丁区域旋转。补丁区域内的坐标变化如下：

 （1.16）

式中，为变换前的坐标，变换之后的坐标。这样在进行补丁区域内点的比较时，就需要比较变换之后坐标位置上的像素灰度值，这种方法称为导向BRIEF。在经过坐标变换之后，补丁区域内的随机点对之间的相关性会增大，表现出更强的一致性。ORB采用贪婪算法并按照规则寻找随机点对，这样随机点对的结果呈现出更好的多样性，这种方法定义为rBRIEF。ORB算法除了完全保留FAST和BRIEF快速的特征外，增加了旋转不变性。

（2）特征描述

Brief描述子是由Calonder等基于随机树的思想提出的一种描述子，具有计算量小、便于存储和比较等优点。Brief描述子选择*S**S*的方形支撑区域，从中提取特征进行描述，当描述子以向量形式呈现时，称为描述向量。生成描述子得步骤分为两步：第一步在*S**S*得方形支撑区域内随机选取两点，并记录第一点和第二点相对被描述点的空间位置。第二步在比较第一点和第二点的灰度值大小，并将结果（0或1）记录在描述向量的第一位。重复步骤一，在*S**S*的方形支撑区域内随机选取不同的两点，并记录第一个点和第二点相对被描述点的空间位置。在重复步骤二，将结果记录在描述向量的第二位。如此反复记录每次随机点对的选取位置，从而得到描述向量的每一位的产生方法。为便于存储一般描述向量取2n位（如28位），意味着要进行2n次比较。需要注意的是，在每个描述子的生成过程中，在步骤1和步骤2中记录的空间对的位置和次序不能改变。

随机点对的第一点和第二点都服从独立同分布规律，以*S**S*方形支撑区域的中心为坐标原点，第一点和第二点都满足正态分布，也就是随机点主要集中在支撑区域的中心区域，越远离中心的位置出现的概率越小。设支撑区域p内随机选取的第一点为*x*，第二点为*y*，两点的灰度值分别为*I*(*x*)、*I*(*y*)，则两点的比较式如以下所示：

 （1.17）

设共选取了*nd*对随机点对，进行了*nd*次比较，其中第*i*次比较的结果为,则Brief描述向量可以用二进制数的形式表示为：

 （1.18）

由上式可见，Brief描述向量便于按位存取，在匹配过程中只需要计算汉明距离（Hamming distance），也就是比较有多少位不同。需要说明的是，由于点对比较对噪声敏感，为提高描述子的稳定性，在生成描述子之前先要对支撑区域进行高斯平滑滤波。

由于*S**S*支撑区域内的不重复随机点对最多有对，如果Brief描述向量的长度为*nd*，则必须满足，即

 （1.19）

其中，*S*为整数。与SURF描述子比较，Brief描述子在匹配任务中的性能相当，但速度提升了许多倍。

## 2.4 本章小结

本章首先介绍了空间认知计算模型，包括头朝向细胞、栅格细胞、位置细胞。头朝向细胞编码角速度信息，确定机器人的方向；位置细胞编码移动机器人的位置信息，通过移动机器人提供的速度信息驱动位置细胞中的活动状态，来估计机器人的位置。然后又详细介绍了RatSLAM算法，包括视觉信息处理、二维吸引子网络、重定位、拓扑地图构建、图优化算法，为后面的研究工作奠定了基础。最后又介绍了ORB特征提取算法。

# 第3章 RatSLAM模型优化

RatSLAM在环境感知中采用基于灰度值的模板匹配方法，来判断一个场景是否曾经去过的地方。该方法采用两图像灰度值的平均绝对误差来度量其差异性。如果两图像的平均绝对误差小于预先设定的阈值，那么这两个图像就判断为同一场景。对新加入的每一帧图像都会与历史存储的所有模板图像进行匹配，以此来检测是否存在回环。该方法采用的图像相似性计算方法过于简单，使得图像匹配的准确率与召回率较低。与此同时，该方法图像检索采用顺序查询方法，计算时间复杂度为。随着存储的模板图像数量增加，计算量会越来越大，最终达不到实时处理的要求。本文针对RatSLAM算法在视觉感知中存在的不足，进行了改进。本文采用ORB算法对环境图像进行特征提取，生成描述子，采用描述子间的汉明距离来度量图像间的相似性，以提高图像匹配的准确率。然后将描述子聚类成单词，建立树状字典，将原算法中顺序查询图像检索转换为树状字典查询，计算时间复杂度为变为，提高了图像检索的速度。在公开数据集和真实场景中实验结果表明，本文提出的回环检测方案提高了RatSLAM算法的图像匹配的准确率与召回率，与此同时所提方法不会随着机器人在环境中探索时间的增长，而增加检索时间，确保始终能达到实时处理。

## 3.1 回环检测方案

回环检测又称闭环检测。在视觉SLAM技术中，回环检测主要用来纠正机器人的位姿误差。机器人在真实环境中，使用携带的传感器采集周围环境的信息，估计自身的位姿信息。因传感器噪声及周围环境的影响位姿估计会产生偏差。随着移动机器人运行时间的增长，机器人姿势估计的累积误差将无限制地持续增长，最终导致建图失败。机器人通过比较某一位置点采集的图像与过去访问过的保存在模板库里的场景图像的相似程度来判断是否曾去过。如果相似程度较高，则发生闭环，纠正机器人位姿估计的累计误差。闭环检测是数据关联的一种类型，对SLAM至关重要。因为它们允许纠正位姿估计累积错误，降低机器人位姿估计的不确定性。候选的环路闭合通常是通过重复观测来检测和验证，然后融合到当前的SLAM估计中。闭环检测事件可能会改变SLAM地图的形状，一些SLAM算法比其他算法能更有效地处理这些变化。

为了使机器人具有实现闭环检测的功能，需要让机器人具有环境感知的能力，通过判断图像之间的相似程度来判断是否回环，完成闭环检测。这个方法类似于人类的决策方式。当人们看到两张图片时，很容易识别两者之间的相似度，以及是否是同一位置。基于视觉闭环检测是一种图像匹配的方法。当检测到存在闭环时，后端根据这一信息，对地图进行优化，纠正移动机器人的位姿估计累计误差，就会得到全局一致的拓扑地图。基于外观的图像匹配不涉及里程计的信息，只考虑图像之间的关系，因此本文采用图像与图像之间的匹配来改进闭环检测方法。

图像匹配是将不同时间、不同位置下获取的两幅或多幅图像进行匹配、计算相互之间的相似度的过程。图像匹配一般分为四步：特征提取、图像匹配，计算图像空间坐标变换的参数和坐标变换。特征提取是图像匹配的关键步骤，是图像精确匹配的根本保证。全局特征是从整幅图像中提取得到的信息，反应全体像素的特性。全体像素的特征可以是灰度、纹理和颜色。全局特征仅生成单个特征向量，因此可以通过比较特征向量来对比两幅图像的内容。例如，一张图为楼内走廊的场景，另一张图为屋内桌子和椅子的场景，采用全局颜色的图像特征就能轻松区别这两张图。局部特征是从图像的某个特定区域中提取的信息，通常与图像的某种或某几种特性（如灰度、颜色或纹理）的变化相关，但不一定能准确定位发生变化的位置。局部特征描述的对象可以是图像中的像素点，也可以是边缘线或小的图像块。

本文提出基于视觉词袋模型的回环检测算法主要分为两个部分，离线部分和在线部分，如图3.1框架所示。其中离线部分的任务是：

（1）使用ORB算法提取图像特征并生成描述子；

（2）使用K-means聚类算法对描述子进行聚类生成单词，然后用树的形式储存单词，最终生成词汇树。

在线部分的任务是：

（1）采用ORB算法提取新图像中的特征点并计算生成描述子；

（2）使用K-means聚类算法生成相应的单词，将单词输入词汇树，并为每个叶子节点存储的单词赋予TF-IDF（Term Frequency-Inverse Document Frequency）[39]权重，计算得到词袋向量；

（3）重复（1）和（2）的操作，在执行过程中持续更新逆索引列表，构造数据库；

（4）在步骤（3）中，当新的图像到来时，利用新图像生成的词袋向量在数据库中进行检索，当发现历史数据库中，存在图像与新图像的相似度大于设置的阈值时，则进行序列匹配验证，判断是否出现回环。



图3.1 基于词袋模型的回环检测算法框架图

Fig. 3.1 Framework diagram of loopback detection algorithm based on bag-of-words model

### 3.1.1 特征提取与匹配算法

本文采用ORB算法对图像进行特征提取。首先利用FAST算法检测图像中的关键点，然后利用BRIEF描述子对其进行描述，最后进行特征描述的相似度判断完成特征点的匹配，其算法流程图如图3.2所示。



图3.2 ORB算法流程图

Fig. 3.2 ORB algorithm flow chart

在对图像经过特征提取及描述后，采用描述子间的汉明距离来判断两图像是否匹配成功。由前面介绍可知，ORB算法是通过二进制来描述特征点的描述子。描述子的距离表示了两个图像特征之间的相似程度。对于二进制的描述子通常采用汉明距离来计算相似性。距离越小说明相似度越高。

但是，直接采用ORB算法进行图像匹配，将汉明距离最小的特征点判断为匹配点，只能采用顺序查询方法。该匹配方法简单粗暴、匹配的正确率较高但是运算量很大。在图像较少时，使用该方法较好，但是，在视觉SLAM回环检测中，常常存储了大量的历史图像信息，如果所有的图像都使用暴力匹配计算相互间的相似性，会导致计算延迟，达不到SLAM中的实时性要求。基于这种考虑本文在此基础上加入了视觉词袋模型算法，用来提高图像匹配的效率，以达到实时的要求。

### 3.1.2 图像检索模型

为了提高图像匹配效率，本文采用基于词袋模型的图像检索方法。词袋，也就是Bag-of-Words（BoW）[40]，最初应用于神经语言程序学和信息检索领域。起初，该模型不考虑文本语言的结构和语法，而是用一系列无序的单词表示待检索的文字或者文档，基于这个机制，词袋模型被广泛应用于计算机视觉领域。视觉词袋模型的目的是用图像中比较有代表性的特征来描述一副图像。例如，一张图像中有一张桌子、一把椅子，而另外一张图像中有一个人和一把椅子，根据这样的描述，就可以度量出两张图片的相似度。

（1）词汇树构建方法

本文采用图像构建树状字典。字典是由很多单词组成，每一个单词代表一个概念。单词和图像中的特征点的概念不同，它不是单纯的由一幅图像上提取出来的，而是多张图像的某一类特征的集合。本文采用K-means聚类方法生成单词，主要步骤包括以下几步：

第一步，在图像中随机选取*n*个中心点：*a1,…,ak*。

第二步，对每个样本，计算它与中心之间的距离，选取最小的作为它的归类。

第三步，重新计算每个类的中心点。

第四步，如果每个中心点的变化都很小，则算法收敛，结束；否则返回执行第二步。

通常情况下字典的规模较大，如果用最朴素的方法，和每一个单词进行比较，查找时间复杂度为，效率太低。本文采用树状结构来存储单词，形成字典，具体步骤如下：

第一步，在根节点，用K-means算法把所有的样本聚类*k*类。这样就得到了第一层。

第二步，对第一层中的每个节点，在把属于该节点的样本聚类成*k*类，得到下一层。

第三步，以此类推执行第二步，最后得到叶子节点层。叶子节点层即为所谓的单词。

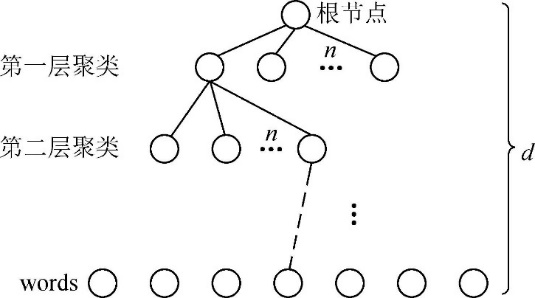


图3.3 K叉树字典示意图

Fug. 3.3 The diagram of K-tree dictionary

如图3.3所示，最后一层叶子节点代表单词，并储存其信息，中间层节点的目的是为了找到叶子节点。构造分支为*k*，深度为*d*的字典树，叶子节点层最多能存储*kd*个单词数量。通过这颗树查找某个单词所在的叶子节点，从根结点出发对树进行深度优先遍历，即可找到叶子节点存储的单词，实现了对数级别的查找效率。

生成词汇树（即字典）的伪代码如表3.1所示。词汇树的存储结构如图3.4所示，树最低端的叶子节点存储的就是字典中的单词。

表3.1 词汇树生成伪代码

Tab. 3.1 Vocabulary tree generation pseudocode

|  |
| --- |
| 算法1 词汇树生成 |
| 输入：图像特征描述子集合*P*={*p1*,*p2…pn*}，分支数*k*，迭代次数*d* |
| 输出：词汇树*S*={*s1*:*xn,s2*:*xj…sk*:*xq*} |
| 1：使用ORB算法提取图像特征并生成描述子，随机选取*n*个描述子作为聚类中心点，记为*a1，a2…an*。 |
| 2：计算描述子集合*P*中每一个描述子到聚类中心的距离，然后根据距离的远近把描述子划分到最近的聚类中心*xq*所表示的聚簇q中：    t表示迭代次数，表示描述子到聚类中心的汉明距离，表示聚类后所包含的描述子集合 |
| 3：聚类完成后计算聚簇内的平均值，作为下层聚类的中心 |
| 4：重复步骤2和步骤3，直到迭代次数达到*d* |
| 5：返回词汇树*S*={*s1*:*xn,s2*:*xj…sk*:*xq*} |



图3.4 词汇树结果图

Fig. 3.4 Vocabulary tree result map

（2）数据库建立与查询方法

在环境探索中，出现频率较高的物体，比如地板，不能作为区分场景的标志物；而出现频率较低的物体，比如门牌号，具有代表性和唯一性，能用来识别场景。因此，为了提高视觉词袋模型表征环境信息的能力，本文为不同图片中每个单词赋予不同的权重，采用TF-IDF算法为图像中聚类形成的单词进行加权。

TF表示某个单词在一张图像中出现的频率越高，表示图像间的区分度越高，而IDF表示某个单词在字典中出现的频率越高，图像间的区分度越低。本论文在建立字典时计算IDF采用的方式是：统计某个叶子节点中的特征数量相对于所有特征数量的比例，作为IDF部分，定义公式为：

 （3.1）

其中，*n*为叶子节点中所有的特征数量，统计的数量为。

TF值的是某个特征在图像中出现的频率。假设一张图像A中某个单词出现的次数为，而这张图像中所有出现的单词次数为*n*，那么TF的计算公式为：

 （3.2）

于是，的权重计算公式为TF乘IDF之积：

 （3.3）

对于图像*A*考虑权重之后，图像中所有的特征点对应到每个单词，组成它的BoW，计算公式如下：

 （3.4）

由于相似的特征落到同一个类中，因此实际的中存在大量的零。尽管如此，通过词袋模型把单个向量描述了一副图像A。这个向量是一个稀疏的向量，它的非零部分表示了图像A中含有那些单词，这些部分的值就是TF-IDF的值。

为了加快计算回环检测时根据单词快速的找到相似的图像信息，在数据库建立过程中本文为每个单词构建一个逆索引表，如表3.2所示。该列表包含了所有历史图像中所提取出的单词，同时该列表还包含了每个单词在每张图片中对应的权重

表3.2 逆索引列表结构示意表

Tab. 3.2 Schematic table of the structure of the inverse indexed list

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 单词序号 | Word 1 | | | |
| 图像序号 | Image 3 | Image 35 | Image 89 | … |
| 权重 | 0.24 | 0.85 | 0.67 | … |

每当添加新的关键帧图像到数据库时，首先通过ORB算法提取图像特征并生成描述子，然后通过BoW计算生成词袋向量并存入数据库中，在添加的过程中更新逆索引列表，其算法过程如表3.3所示。

表3.3 加入新图像算法伪代码

Tab. 3.3 Add new image algorithm pseudocode

|  |
| --- |
| 算法2 数据库加入新图像 |
| 输入：新图像*F*，词汇树*S*={*s1*:*xn,s2*:*xj…sk*:*xq*} |
| 输出：新图像通过BoW生成词袋向量*Vn*，更新逆索引列表 |
| 1：提取图像F的特征点并生成描述子集合*P*={*p1*,*p2…pn*} |
| 2：将图像特征描述子集合*P*={*p1*,*p2…pn*}，输入词汇树*S*={*s1*:*xn,s2*:*xj…sk*:*xq*}，计算生成词袋向量*vn* |
| 3：利用公式 3.14计算图像所包含的各单词的TF-IDF权重 |
| 4：更新逆索引列表 |

当视觉SLAM算法运行时，新图像的关键帧会不断的加入到数据库，同时新图像还会在数据库中进行图像匹配，查找是否有相同的历史场景，用以判断是否发生回环。回环检测运行，会通过ORB算法对新图像提取特征点并计算相应的描述子，通过K-means聚类算法生成若干个单词，这些单词在词汇树中都具有单独的索引列表，根据这些索引列表可以快速的查找到与新图像具有相同单词的历史图像，然后计算新图像与历史图像之间的相似性，找出历史图像中是否有相同的场景信息，从而判断是否发生回环。

（3）词汇树生成

本文提出的基于ORB算法和BoW算法改进的基于RatSLAM算法的闭环检测，事前需要训练一个词汇树。本文训练的词汇树采用的是与实验场无关的数据集图像Bicocca\_2009-02-25b（RAWSEEDS，2007–2009），该公开数据集是在意大利米兰比可卡大学（University of Milano -Bicocca）的建筑内采集的，总共包含了26335张图像。本文使用这个数据集训练生成词汇树，之后的所有实验均采用此词汇树。首先把数据集中的图像信息全都提取出来，然后使用K-means聚类算法进行训练，该词汇树的分指数设为8，单词所包含的最大特征描述子数量为6，最后训练出一个包含294810个单词的词汇树。

### 3.1.3 闭环决策模型

本文提出符合改进RatSLAM算法的闭环决策模型，加入关键帧的选取，动态阈值的设置，以及候选图像的查找与验证。

（1）关键帧选取

为了提高图像处理的效率，本文做了关键帧的选取。基于视觉SLAM算法构建环境地图时，环境中的路标信息对导航与建图是十分关键的，由于相机的频率较高会采集大量的图像信息，如果每一帧图像都进行匹配计算，产生不必要的浪费。RatSLAM算法中会对每一帧图像都进行计算。本文对这一缺点进行了改进，做了关键帧的选取。如果选取的关键帧太近，将会导致关键帧之间的相似度过高；如果关键帧选取的过远，则不容易检测出之前到过的位置。闭环检测时经常会遇到第*k*帧和第*k*-1帧、*k*-2帧最为相似，而这种相邻的闭环检测没有任何意义。所以选取关键帧时，要选取具有代表性，彼此之间需要存在一定的差异，又能涵盖整个环境的关键帧。

本文关键帧的选取策略简述如下：首先将单目相机采集到的图像信息进行灰度化处理，检测图像中的特征点数量；如果提取的特征点数量小于阈值A，则认为当前图像没有价值的参考，则舍去本张图像，例如白墙、地板等表面信息较少的场景。计算当前帧与上一关键帧的相似度，如小于阈值B说明当前帧图像与上一关键帧图像的相似度较大，选为关键帧图像进行处理。其功能实现如图3.5所示。



图3.5 关键帧选取和闭环点检测流程图

Fig. 3.5 The flow chart of Key frame selection and closed-loop detection

（2）相似性阈值的选择

原算法中匹配的阈值为C，当在模板库中找到与当前图像误差小于C的图像则判断为回环点。对任意两幅图像，都可以计算出他们之间的相似性得分，但是只是利用这个分值的绝对大小并不是太好的选择。有些环境的外观本来就很相似，另一些环境中的场景信息存在较大的差异。考虑到这种情况，本文选取一个先验相似度s，它表示当时关键帧与前十个关键帧相似性的均值，然后其他分值都参照这个值进行选取。这是为了避免引入绝对的相似性阈值，使得算法能够使用更多的环境。

（3）查找候选图像

构建的数据库中存储着所有历史关键帧图像以及它们之间的关系（逆索引列表）。当来了新的关键帧图像时，通过BoW算法将图像生成对应的词袋向量，在数据库中利用已构建的逆索引列表，找出与新图像包含相同单词的历史图像，将找出的所有历史图像集合记为F={,,…}；然后利用构建的数据库找出所有历史图像对应的词袋向量，将所有的词袋向量集合记为V={,,…}；最后可以得到新图像与所有历史图像的匹配对，记作<,>，<,>，<,>…<,>。

利用公式计算词袋向量和的相似性分数如下：

 （3.5）

对于任意两张图像都可以通过这种方式计算出他们之间的相似度。当计算出所有候选关键帧的相似度分数后，舍去其中分数*s̃*小于*τ*（*τ*是预先设置的阈值）的匹配对。这里设置的阈值*τ*筛选掉相似度较低的匹配对。因为这个阈值是人为设置的，所以可以通过调整该阈值来筛选出需要的候选匹配对，最后保留的图像集合为，其对应词袋向量集合为。

一般情况下，相邻图像一般很相似，为了避免某一个特殊的图像造成错误的闭环，这里引入岛的概念。岛指的是连续一组关键帧图像的集合，记作，其中*a*和*b*分别代表关键帧图像的序列号。每一个岛都有一张代表图像，即岛内相似度值*s*最大的图像，然后计算当前帧图像与岛内所有图像相似度的平均值。引入岛之后，在回环检测算法中就可以避免某一特殊相似图像产生误匹配的错误。

利用保留的图像集合生成岛，对于任意图像*Fc*∈，如果其关键帧图像序列号已经位于一个存在的岛图像序列内，则将图像*Fc*归纳进岛内，并且调整岛的图像序列的大小，使得图像*Fc*的前*n*帧图像和后*n*帧图像的图像都包含在岛内，即=∪；如果图像*Fc*的图像序列号不在任何已存在的岛的图像序列内，则创建一个岛，即以图像*Fc*为中心，将前*n*帧图像和后*n*帧图像作为岛内的图像，构成初始大小为2*n*+1个连续图像序列的岛。对于每一个岛，需要计算当前帧图像与内部所有图像的相似度的平均值，计算公式如下所示：

 （3.6）

计算新图像与岛内每张图片的相似性求出平均值，在相似度均值最高的岛内，找出与新图像最相似的图像，将有可能成为闭环点，岛的构建过程如表3.4所示。

表3.4 构建岛的伪代码

Tab. 3.4 Pseudocode for building islands

|  |
| --- |
| 算法3 创建岛 |
| 输入：图像集合*F*，关键帧图像序列号范围参数*n* |
| 输出：岛的集合*I*，岛内部图像平均分数的集合*G* |
| 1：对于图像*Fc*，如果存在一个岛使*a*<*c*<*b*，则执行步骤2，否则执行步骤3 |
| 2：将图像*Fc*将图像*Ic*包含进岛，并使用式=∪调整岛的图像序列大小 |
| 3：以图像*Fc*为中心创建新岛 |
| 4：重复步骤1，直到图像集合内所有图像都被使用，集合I创建完成 |
| 5：根据公式（3.6），计算得到集合*G* |
| 6：返回岛的集合I和岛内部图像平均分数的集合*G* |

综上所述候选图像查找流程表，如表3.5所示。

表3.5 查找候选图像算法伪代码

Tab. 3.5 Find Candidate Image Algorithm Pseudocode

|  |
| --- |
| 算法4 查找候选图像 |
| 输入：新图像*Ft*，词汇树*S*={*s1*:*xn*,*s2*:*xj…sk*:*xq*}，数据库，阈值 |
| 输出：分值最大的岛，候选图像 |
| 1：使用ORB算法提取图像Ft特征并生成描述子集合*P*={*p1*,*p2*…*pn*}，通过词汇树*S*={*s1*:*xn*,*s2*:*xj…sk*:*xq*}计算得到对应的词袋向量*vt* |
| 2：利用构建的数据库中每个单词维护的逆索引列表检测与图像Ft含有相同单词的历史关键帧图像列表F={,,…}，并得到对应的词袋向量v={,,…} |
| 3：采用公式（3.17）计算*Ft*与图像列表*F*中所有图像的相似性，筛选出相似值大于的历史关键帧图像，并得到图像集合和对应的词袋向量 |
| 4：通过算法3为图像集合中的关键帧图像创建岛，计算得到岛的集合*I*和分数集合*G* |
| 5：根据分数集合*G*，选出图像相似度平均值最高的岛，而岛中图像与Ft图像最相似的图像作为候选图像 |
| 6：返回分值最大的岛，候选图像 |

利用算法3将把图像集合中的图像构建成小大不一且相互不重叠的岛。图3.6所示，构建岛的实例，圆圈代表图像，图中包含7张图像序列。图中构建的岛红色圆表示岛中的代表图像。



图3.6 岛的构造示意图

Fig. 3.6 Schematic diagram of the structure of the island

（4）序列匹配检验

为了防止产生错误的回环，本文添加了序列匹配来检测步骤（3）中选出的候选图像是否是正确的回环。序列匹配是将当前图像及前几个关键帧与模板库中候选图像的连续几帧进行匹配。具体步骤如下：设定序列匹配的长度值为*d*，表示模板数据库中关键帧的局部序列包含*d*张图片；选取当前图像信息*d*张连续关键帧组成的序列；然后将当前帧图像的序列与模板库中相同长度的局部序列进行匹配；最后计算每一对对应的图像进行相似度计算，若每一对图像的相似性都大于设置的阈值*s*，则判断为回环。

（5）闭环后验

基于外观的闭环检测完全依赖于环境中的图像信息没有利用任何的几何信息，由于观测数据的误差以及外界环境的变化等因素，导致环境场景信息相似的图像很容易被判断为闭环，造成误匹配。因此需要对闭环信息进行检测确定是否是正确的。闭环后验是在检测出发生闭环之后的最后一个检验步骤，以应对感知偏差的问题。通常比较常用的两种检测算法是：空间一致性检测和时间一致性检测。空间一致性检测是利用了地图的几何信息，当检测到两帧之间存在闭环时，对两个关键帧对应的位置坐标计算距离，如果距离过远的话，则此闭环是错误的闭环，反之则是正确的闭环。时间一致性检测是指在一段时间内多次检测到闭环，则证明此闭环是正确的。由于机器人运动的场景很大，会存在较大的累计误差，不能很好的估计距离阈值是多少比较合适。而时间一致性检测，只考虑是否有连续熟悉的场景，而且比较符合动物的行为，因此本文选择了时间一致性检测算法来检测闭环是否是真实的，这大大降低了错误的闭环检测。

闭环检测成功后，又将会遇到另一个问题。例如，当检测到第1个关键帧与第*n*个关键帧存在回环时，对构建的认知地图进行优化是有帮助的。而后面的第*n*+1关键帧、第*n*+2关键帧与第1个关键帧构成回环对地图的构建和优化没有任何意义。因为之前的闭环校正已经对图进行了优化，更多的回环并不能带来更多的信息。因此，当发生闭环检测之后，检测相近的回环将会聚成一类，从而使闭环检测不会检测同一类的闭环。

## 3.2 RatSLAM优化方案

本文在RatSLAM模型的基础上融合ORB算法和词袋模型对原算法进行优化，如图3.7所示。其中，ORB算法负责提取图中的特征信息，提高图像匹配的准确率和召回率。在ORB算法的基础上又添加了词袋模型，用来加快在视觉模板库中查找图像的速度的，达到实时计算的要求。由于单目相机采集图像的频率较高，需要处理的图像信息过多，单纯降低采样频率，有可能会丢失重要的信息。因此，本文加入了关键帧的选取。并且为了提高闭环检测的准确度，本文进一步将原RatSLAM算法中寻找一张最相似的历史图像的匹配策略优化为寻找多张相似图像，然后引入岛的概念，将单张图像匹配变为前后连续几帧图像的匹配，取平均结果最好的那个岛中的代表图像作为候选匹配图像，成为候选的闭环点。然后，将候选图像前n帧与当前图像的前n帧一一匹配。如果都以较高的相似度匹配成功，则判断该候选图像为匹配图像。最后向该图像对应的位姿细胞注入能量，共同驱动位姿细胞网络的运行，完成重定位。

|  |
| --- |
|  |
| （A）原算法模型 （B）改进算法模型 |

图3.7 基于词袋模型的RatSLAM系统结构

Fig. 3.7 Structure of RatSLAM system based on bag-of-words model

本文改进的RatSLAM算法将新添加的关键帧图像进行特征提取并计算生成描述子，然后对描述子的计算生成词袋向量，将当前帧与模板库中关键帧进行匹配，计算图像之间的相似性，通过闭环决策模型判断是否发生回环，并将检测到的闭环结果发送到后端，执行图优化算法，对构建的环境地图进行位置校正，消除或减少系统的累计误差。如图3.8所示为闭环检测流程图。



图3.8 闭环检测流程图

Fig. 3.8 Closed-loop testing flow chart

## 3.3 实验与分析

为了验证本文改进的RatSLAM方法的有效性，本章节采用了二个公开数据集和一个我们自己采集的真实室内场景数据集来对所提方法进行测试，以证明改进方法比原算法的建图效果更好。

### 3.3.1 实验数据

（1）新加坡Infocomm研究所办公室数据集

该数据集由新加坡Infocomm研究所田博等人采集制作[27]。该数据集是机器人在半径为35m的办公楼里探索1050s所记录的里程计信息和RGB-D图像信息。是目前基于RatSLAM算法研究中使用最多的验证数据集。因此，本文也采用该数据集来验证我们提出的方法。

（2）KITTI\_00数据集

KITTI\_00数据集（A project of Karlsruhe Institute of Technolohy and Toyota Technological Institute at Chicago）是目前国际上在自动驾驶环境下进行计算机视觉算法性能测评中所能使用的最大公开数据集。该数据集由卡尔斯鲁厄理工学院（Karlsruhe Institute of Technology）和丰田美国技术研究院（Toyota Technological Institute at Chicago）联合发布。本文使用第00（KITTI00）组数据集，该数据集包含了4541帧由单目相机拍摄到的城市图像，其行驶长度约为15公里。

（3）采集的真实室内场景数据集

为了进一步验证本文方法的有效性，我们在自己实验楼室内的走廊环境中，操作先锋机器人绕着走廊移动两圈，采集的了里程计信息和RGB图像信息，作为第三个验证算法的数据集。

### 3.3.2 评价标准

两张图片如果从同一位置拍摄，那么回环算法检测应该给出“这是回环”的结果。反之，如果两张图片不是从同一位置拍摄的，算法应该给出“这不是回环”的判断。然而，算法有可能将不是同一位置拍摄的图片检测为“这是回环”，也可能将同一位置拍摄的两张图片检测为“这不是回环”。回环检测所有可能出现的结果分4种情况，见表3.6。

表3.6 回环检测的结果分类

Tab. 3.6 Classification of the results of loopback testing

|  |  |  |
| --- | --- | --- |
| 算法/事实 | 是回环 | 不是回环 |
| 是回环 | 真阳性（True Positive） | 假阳性（False Positive） |
| 不是回环 | 假阴性（False Negative） | 真阴性（True Negative） |

TP（True Positive，真阳性）：判断结果为正例，实际结果也为正例，正确的判断出了两张图像在同一位置。

TN（True Negative，真阴性）：预测结果为负例，实际结果也为负例，正确的判断出了两张图片不是同一位置拍摄的。

FP（False Positive，假阳性）：又称为感知偏差，预测结果为正例，而实际结果为负例，错误的判断了两张来自不同场景的图片为一场景采集得到的，如图3.9所示。

FN（False Negative，假阴性）：又称为感知变异，预测结果为负例，而实际判断结果为正例，错误的把两张来自同一场景采集到的图片判断为不同场景采集得到，如图3.10所示。



图3.9 感知偏差

Fig. 3.9 Perceptual bias



图3.10 感知变异

Fig. 3.10 Perceptual variation

回环检测算法的评价指标通常使用准确率和召回率来表示。其中准确率代表的是真阳性在算法检测出所有回环中的比例，如公式3.7所示。召回率代表的是真阳性在事实上所有回环中的比例，如3.8所示。

 （3.7）

 （3.8）

在SLAM中，对准确率的要求更高，而对召回率则相对宽松。由于假阳性的回环将在后端的位姿图中添加根本错误的边，会导致优化算法给出完全错误的结果，最后整个构建的地图都失效。相比之下，召回率低一些，顶多有部分的回环没有被检测到，地图可能受一些累计误差的影响，然而仅需一两次回环就可以完全消除它们。所以在选择回环检测算法时，本文更倾向于把参数设置得更严格，在检测之后再加上回环验证的步骤。

### 3.3.3 分析与对比

（1）图像匹配结果对比

本小节将改进的图像匹配方法与原RatSLAM算法的图像匹配方法在不同数据集伤进行结果对比，验证本文提出的图像匹配方法的有效性。图3.11是新加坡Infocomm研究所办公室数据集上的实验结果。其中，横轴表示新采集的图像的数量，纵轴代表视觉模板库图像的数量。当未发生闭环时，机器人不断将新的场景加入视觉模板库中，而当机器人发生闭环时，则回到之前所记录的图像的Id，该图可以判断机器人在SLAM中的闭环性能。在新加坡Infocomm研究所办公室数据集中，如图3.11（A）所示，在500帧处连续的检测到错误的闭环点。而本文改进算法结果如图3.11（B）图所示，分布的散点表示错误的检测到闭环点，没有出现大范围的检测错误的闭环。在KITTI\_00数据集中，如图3.12（A）所示，原RatSLAM算法没有发生错误的闭环，但是只检测到了两处闭环，没有检测出所有的闭环位置。而本文改进的算法结果，如图3.12（B）所示，完全正确的检测出了5处闭环且没有发生错误的闭环。在采集的真实室内场景数据集实验验证结果中，如图3.13（A）所示，原算法检测出了很多与之前场景位置的错误闭环。本文改进的算法如图3.13（B）所示，从1000帧处开始连续的检测到闭环点，正是机器人运行第2圈的时候。

|  |  |
| --- | --- |
|  |  |
| （A）RatSLAM模板匹配图 | （B）本文改进的模板匹配图 |

图3.11 新加坡Infocomm研究所办公室数据集

Fig. 3.11 Infocomm Institute Singapore office dataset

|  |  |
| --- | --- |
|  |  |
| （A）RatSLAM模板匹配图 | （B）本文改进的模板匹配图 |

图3.12 KITTI\_00数据集

Fig. 3.12 KITTI\_00dataset

|  |  |
| --- | --- |
|  |  |
| （A）RatSLAM模板匹配图 | （B）本文改进的模板匹配图 |

图3.13 采集的真实室内场景数据集

Fig. 3.13 Real indoor scene dataset collected

表3.7 闭环检测结果比较

Tab. 3.7 Comparison of closed-loop test results

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 数据集 | 算法 | Number of true positives （TP） | Number of false positives(FP) | Number of false negatives(FN) | Precision | Recell |
| 新加坡Infocomm研究所办公室 | RatSLAM | 367 | 192 | 298 | 65.6% | 55.1% |
| 本文算法 | 413 | 37 | 331 | 91.8% | 55.5% |
| KITTI\_00 | RatSLAM | 235 | 86 | 403 | 73.2% | 36.8% |
| 本文算法 | 474 | 0 | 260 | 100% | 64.6% |
| 真实实验 | RatSLAM | 337 | 389 | 415 | 46.4% | 44.8% |
| 本文算法 | 445 | 0 | 374 | 100% | 54.4% |

与原图像匹配算法相比，本文改进图像匹配算法准确率和召回率在三个数据集上都有所提升（见表3.7）。在新加坡Infocomm研究所办公室数据集上原算法RatSLAM的准确率为65.6%，召回率为55.1%，本文改进后的回环检测算法在RatSLAM算法上的准确率为91.8%，召回率为55.5%，准确率提升了26.2%，召回率提升了0.4%。在KITTI\_00数据集上，原算法RatSLAM准确率为73.2%，召回率为36.8%，本文改进RatSLAM模型的准确率为100%，召回率为64.6%，准确率提升了26.8%，召回率提升了27.8%。在采集的真实室内场景数据集，原算法RatSLAM的准确率为46.4%，召回率为44.8%，改进后的RatSLAM模型的准确率为100%，召回率为54.4%，准确率提升了53.6%，召回率提升了9.6%。

（2）认知地图对比

|  |  |
| --- | --- |
|  |  |
| （A）实验楼结构图 | （B）里程计 |
|  |  |
| （C）基于RatSLAM算法构认知地图 | （D）基于本文改进的RatSLAM算法构建的认知地图 |
| 图3.14 新加坡Infocomm研究所办公室数据集  Fig. 3.14 Infocomm Institute Singapore office dataset | |

图3.14展示了新加坡Infocomm研究所办公室数据集数据构建环境地图结果。图3.14（B）为采集数据集时机器人移动的里程计信息，可以看到非常的杂乱。图3.14（C）为RatSLAM原算法构建的环境地图，可以看到有些杂乱，那是发生了错误的闭环，错误的进行了图优化，导致构建的地图，存在很大的误差。图3.14（D）为本文改进RatSLAM算法构建的环境地图，中间两条轨迹是由于机器人在同一个走廊正反两个方向采集的数据，导致不会拟合为一条线。本文构建的环境地图与真实的环境地图非常吻合，验证了本文所提方法的有效性。

|  |  |  |
| --- | --- | --- |
|  |  |  |
| （A）里程计 | （B）基于RatSLAM算法构认知地图 | （C）基于本文改进的RatSLAM算法构建的认知地图 |
| 图3.15 KITTI\_00数据集  Fig. 3.15 KITTI\_00dataset | | |

图3.15展示的是原算法和本文改进算法在KITTI\_00数据集实验结果对比。图3.15（A）中可以看到原始里程计构建的轨迹地图，同一条道路运行经过两遍，地图显示距离很远，从地图中观察会错误的把一条线路的两次轨迹当成两条线路。图3.15（B）为原RatSLAM构建的地图，可以看到有些地方检测到了回环并进行了图优化步骤，但是有两小段没有检测到闭环。图3.15（C）为本文改进后算法构建的地图，可以看到所有需要闭环的地方都准确的检测到了闭环，最后构建出了完美的环境地图。

本文改进的RatSALM算法在三个数据集上都表现出来良好的制图效果，比原算法制图结果明显更优，验证了本文所提出的回环检测模型的有效性

|  |  |  |
| --- | --- | --- |
|  |  |  |
| （A）里程计 | （B）基于RatSLAM算法构认知地图 | （C）基于本文改进的RatSLAM算法构建的认知地图 |
| 图3.16 采集的真实室内场景数据集  Fig. 3.16 Real indoor scene dataset collected | | |

图3.16展示了本文采集的真实室内场景数据集构建的环境地图结果。图3.16（A）为原始里程计信息，可以看出第二圈与第一圈已经发生了很大的误差，如果机器人继续运行，此误差会越来越大。图3.16（B）为RatSLAM原算法构建的地图，由于走廊存在很多白墙、及相似的场景较多，导致发生错误的闭环，引发错误的图优化算法，最后导致地图混乱。图3.16（C）为本文改进的RatSLAM算法构建的环境地图，可以看出第二圈检测出了正确的回环，成功构建出了环境地图。

## 3.4 本章小结

通过对RatSLAM算法分析，发现算法中回环检测模型存在计算复杂度高、正确率和召回率低的缺陷，本文基于这个缺陷提出采用ORB算法提取图片特征，并用汉明距离计算图像之间的相似性，提高图像匹配的准确率及召回率，改进回环检测功能。然而这种计算方式复杂度高，采用顺序查找图像时间复杂为，随着机器人对环境地图的不断探索，视觉模板库随之增大，最终达不到实时计算的要求。因此，本文又提出了BoW算法，把图像特征点聚类成单词，然后又采用树的方式存储单词，每个单词都有一个逆索引表维护图像之间的关系，加快了图像的检索速度，达到了实时处理图像的要求。同时为了改进的RatSLAM稳定的构建环境地图，本文加入了关键帧筛选、动态阈值的选定、岛的匹配方式、序列匹配检测是否是正确的回环判断以及时间一致性检测，确保回环检测的准确率。本文最后在三个不同的室内环境数据集上做了实验验证，证明了本文所提方法的有效性。

# 第4章 基于改进的RatSLAM导航

路径规划是移动机器人实现自主移动的关键技术之一。SLAM算法为室内移动机器人的路径规划提供了环境地图信息，指引移动机器人的自主导航。本文在改进的RatSLAM算法基础上，实现移动机器人自主导航的功能。本文采用的导航方法包括全局路径规划，利用改进的RatSLAM 构建的认知地图起始位置和目标位置之间获取一条最优路径，然后采用局部路径规划，避开最优路径上的动态障碍物，到达目标位置。

本文在认知地图的基础上实现了移动机器人的自主导航，首先，加载构建的认知地图并进行初始化定位。机器人通过单目摄像头传入的图像信息与图像模板库中的数据信息进行匹配，判断数据库中是否有与当前图像相似的数据，如果有则进行重定位，否则调整机器人的位置进行寻找，直到找到为止。当机器人确定所在地图中的位置后，给定机器人要移动的位置，使用Dijkstra算法进行全局路径规划，找到一条距离最短的路经。使得机器人沿着全局路径规划线进行移动，当遇到动态障碍物时，使用DWA算法进行局部路径规划调整全局路径规划的线路，找到一条避开障碍物的路径，使机器人顺利绕过障碍物到达目标点。其导航功能流程图如图4.1所示。



图4.1 导航流程图

Fig. 4.1 Navigation flow chart

## 4.1 全局路径规划

本文在改进的RatSLAM 构建的认知地图上采用全局路径规划，寻找起始位置到目标位置的最短路径。全局路径规划是实现移动机器人导航功能的关键技术之一，在全局路径规划过程中机器人应具备对环境信息的先验知识。这些信息包括环境中障碍物的位置、大小、形状、方向。还需要知道自身和目标相对环境中的位置。为了使全局路径规划算法的搜索效率和性能最大化，通常会使用最优化的算法来进行优化。然而，这种技术具有局限性，此方法在动态环境中应用是不可靠的，因为多个移动物体在环境中移动时，会影响移动机器人对环境的估计，无法沿着预先规划好的全局路线到达目标点。所以需要融合局部路径规划算法避开动态的障碍物。

RatSLAM所生成地图是认知地图，具有拓扑特性。Dijkstra算法能够快速规划出最优路径，且不需要复杂的计算[41]。因此，本文选用Dijkstra算法来全局路径规划，将Dijkstra算法应用于RatSLAM所构建的经历地图，利用经历地图的节点信息以及节点从下建立的联系中包含的时间和距离信息，根据距离最近原则或者时间最短原则，都能够找出一条符合要求的最佳全局路径。其具体做法如下：

（1）确定目标节点，人为设置一物理坐标作为目标位置，然后对地图节点进行检索，根据与目标位置距离为判断依据，比较每个节点距目标位置的距离，并从中找出距离最短的节点作为目标节点。

（2）初始化，与Dijkstra算法类似，设置两个栈结构，S为已找出的路径节点集合，U为待搜索集合。开始时将当前位置作为源节点加入S中，剩余的所有节点放入U中。

（3）通过经历地图的其他经历节点间，根据节点之间的联系（link），找出与S中最新节点具有联系的其它节点，并以到该节点的距离和到目标节点位置的距离加权求和，得到最近的节点，将其加入S中，并将该节点从U中去除。

（4）重复（3）的过程，直到检索到目标节点的位置为止。此时从起始点到目标位置的所需经历点所有节点已经找出来了。

（5）将S栈中的元素依次弹出，可以得到从目标节点到起始节点的节点序列，而这个序列形成的路线即系统所需要的全局路径。

如图4.2所示，在基于本文改进的RatSLAM算法构建的认知地图中规划出最优路径规划。浅蓝色细线为认知地图，蓝色粗线为全局路径规划的路线，从图可以看出机器人在得到导航目标位置时，会在当前地图中计算出一条最优的路经作为全局路径。

|  |  |
| --- | --- |
| A | B |
|  |  |
| C | D |
|  |  |

图4.2 全局路径规划实验图

Fig. 4.2 Global path planning experimental diagram

## 4.2 局部路径规划

本文采用局部路径规划来避开动态障碍物。局部路径规划主要解决机器人移动期间避障的问题，当机器人检测到有障碍物时，改变机器人的运动轨迹。在实际的环境中障碍物不都是静态的，对于环境中存在的动态障碍物，机器人需要利用声纳传感器，实时探测周围的有没有障碍物以及障碍物的距离。当障碍物突然出现时，立刻检测到障碍物的信息，并迅速的规划出新的路径避开障碍物，到达目标点。因此机器人导航的过程中需要局部路径规划算法避开动态的障碍物。全局路径规划完之后在结合局部路径规划来控制移动机器人运动到目标点，同时移动机器人上的声纳传感器会实时检测障碍物信息，使机器人避开障碍物到达目标点。

本文选用效果较好的最常用的动态窗口法DWA实现局部路径规划。DWA算法利用速度采样的方式，给出多条可选局部路径，在从中找到一条最优的路径，完成机器人的移动。DWA算法将搜索空间减小到在动态约束下可到达的某些位置，然后利用这些约束，达到快速筛选的目的。在DWA算法中，通过设置一定的模拟时间，并在机器人的速度空间内，通过采样多组速度，用这些速度在一定时间下的位移来模拟机器人的可能轨迹，对这些采样出来的轨迹进行评价，以某种代价函数进行评估，将代价最低的路径作为机器人的最佳路径，并以这个速度驱动移动机器人[42-44]。

本节基于DWA算法的思想原理，设计了一种基于RatSLAM经历地图的局部路径规划算法。通过之前的全局路径规划本文已经得到的全局最优路径。然后，采用DWA算法的机制驱动机器人从当前位置按照全局规划出来的路径逐渐到达目标位置。具体算法如下：

（1）分段导航设计。将全局路径分成从当前位置到目标位置所需要经历的节点序列，并将这些节点依次作为局部路径规划的目标位置，发布给局部路径规划器，并为了保证移动的连续性，在到达目标结点前，不需要机器人精确地到达中间的导航点，可以提前结束该段路径的导航，提前将目标调至下一导航点。由于构建环境地图时，已存储了每个节点的位置信息和角度信息，因此可以直接计算节点之间的相对角度，计算公式如下：

 （4.1）

其中，表示连续*n*个节点的夹角之和，代表当前节点的角度，代表当前节点的后*n*个节点的角度。当值大于设置的阈值则表明在此处机器人处于拐点的位置，则把当前的节点坐标设置为局部导航目标。

（2）坐标转换计算。由于在导航的过程中需要实时检测周围的障碍物，因此需要一个统一的坐标系，即需要将RatSLAM经历地图中的世界坐标系中转换成机器人自身的机器坐标系，此时得到坐标系变换后的目标导航点。如图4.3所示，机器人的位置中心点坐标系与世界坐标系之间的关系，通过一下公式计算得到机器人在世界坐标系中的位置：

 （4.2）

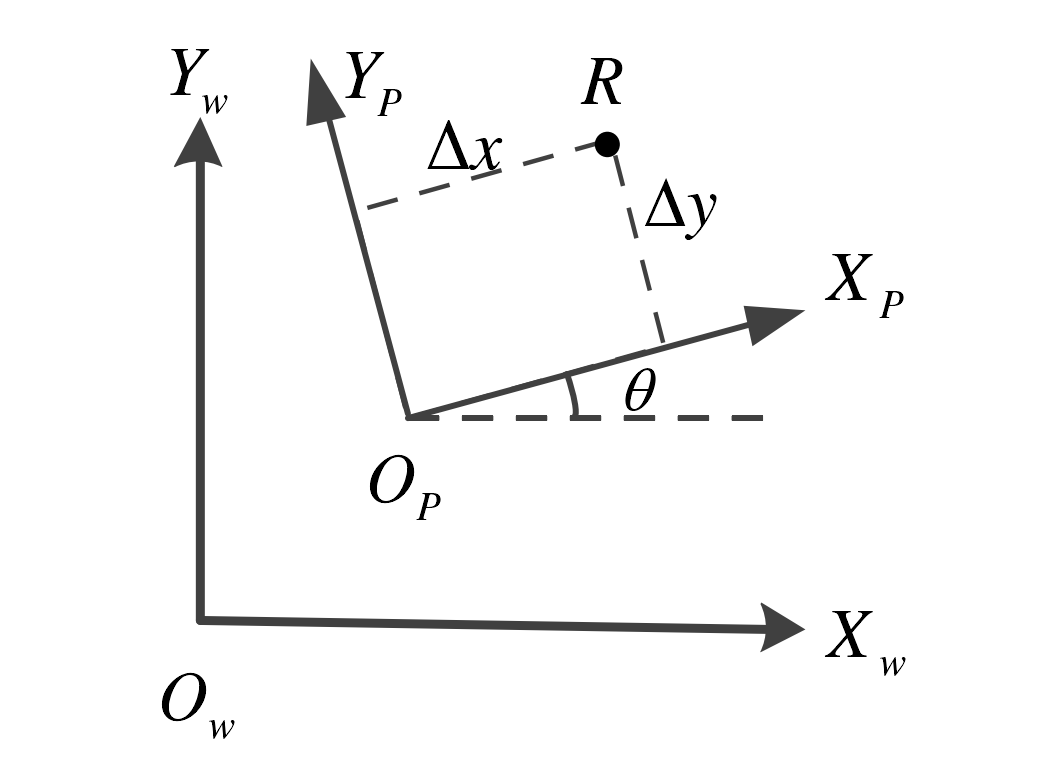


图4.3 机器人坐标转换

Fig. 4.3 Robot coordinate conversion

（3）速度限制约束。通过分析机器人的自身条件限制等因素，需要将速度限制在一定可行范围内，可以减小计算量以及得到更加符合真实情况的模拟轨迹。

（4）代价函数设计。DWA算法使用需要根据机器人导航的要求设计合理的代价函数，用以快速合理的规划局部路径。机器人局部路径规划重要是为了避开障碍物、快速、准确的避开障碍物到达目标。基于这三个方面的考虑，本文设计三种代价评价函数，现实移动机器人动态避障的功能，检测障碍物代价函数为：

 （4.3）

其中，和分别代表障碍物相对机器人的位置。通过判断障碍物距机器人的距离来判断路径的代价，障碍物越近，代价越高。机器人移动速度代价函数为：

 （4.4）

其中为期望线速度，v为速度窗口的模拟速度，该式表明了速度越接近期望速度，所需的代价就越小。目标位置代价函数为：

 （4.5）

这个函数主要是为了机器人能够精准地导航至目标位置，通过这个代价函数，离目标位置越远，代价越高，因此，机器人将会逐渐接近目标位置，最后到达目标位置。整体代价函数为：

 （4.6）

（5）速度分配策略。将得到的最佳局部路径中应用的速度和角速度发布给机器人，驱动移动机器人，逐渐移动至最终目标位置。

（6）最终的角度控制策略。在中间的导航过程中不考虑机器人导航点的角度，而当到达最终的位置附近时，开始角度控制，驱动机器人的转向最终在目标节点中所储存的角度信息。

## 4.3 实验与分析

本文使用MobileSim仿真软件验证导航方法的可行性。如4.4（A）所示，仿真机器人在环境中的地图轨迹由浅蓝色细线标出。在起始点S到目标点T间，采用本文的全局路径规划算法，成功找到最短路径，如图4.4（B）所示。最优路径由深蓝色粗线标，红色点标记出为局部导航的目标位置。然后，通过本文采用局部路径规划算法，移动机器人沿着最短路径向目标点T逐步移动如图4.4（C）、（D）、（E）、（F）所示。红色线为仿真机器人实际的运动轨迹。通过该实验表明器人可以快速的沿着全局路径规划路线到达目标位置。

|  |  |
| --- | --- |
| A | B |
|  |  |
|  |  |
| C | D |
|  |  |
| E | F |
|  |  |

图4.4 局部导航实验结果图

Fig. 4.4 Local navigation experiment results graph

为了测试算法的动态避障功能，在全局最优路径上加入了障碍物O，如图4.5（A）所示。实验中，仿真机器人能够巧妙的避开障碍物到达设置的目标位置，如图4.5（B）(C)、(D)所示。从图4.5中机器人实际的运动轨迹来看，仿真机器人遇到障碍物时可以重新规划轨，迹避开障碍物朝着目标位置到移动，验证了本文算法的局部动态避障功能。

|  |  |
| --- | --- |
| A | B |
|  |  |
| C | D |
|  |  |
| E | F |
|  |  |

图4.5 加入障碍物后局部导航实验图

Fig. 4.5 Experimental map of local navigation after adding obstacles

## 4.4 本章小结

本章主要实现基于改进的RatSLAM模型的导航算法，实现移动机器人自主移动功能，主要设计了基于认知的全局路径规划和局部路径规划方法。根据认知地图的拓扑特征，选用了基于认知地图的Dijkstra算法，快速准确的找出一条从当前位置到达目标位置的最优的全局路径，为局部路径规划提供了可靠的规划数据。随后又设计了基于经历地图的DWA算法，根据系统的特性和要求，设计出相应的代价函数，进而得到适合本文构建认知地图的局部路径规划算法，实现动态避障功能。本章在仿真环境中验证全局路径规划与局部路径规划的可行性，并在仿真实验中添加了障碍物，验证了局部路径规划动态避障的有效性。

# 第5章 现实场景中应用

## 5.1 系统总体结构

移动机器人类脑导航系统主要有四个功能模块组成分别是：局部视图模块、位姿细胞模块、经验制图模块、导航模块，如图5.1所示。首先，本系统通过局部视图模块实现对大鼠的视觉。单目摄像头周围信息的采集，并传回ROS后端进行处理。若当前位置是首次抵达，则将当前位置的环境信息保存到经验制图模块，用来实现闭环见则。其次，本系统通过构建位姿细胞模块实现对大鼠脑内位置细胞的发放模式进行模拟，并将来自局部视图模块的能量注入到位置细胞中，同时基于机器人的速度信息对能量包的驱动进而完成路径积分，实现对机器人位置信息精确估计。当机器人来到熟悉场景时，本系统通过对位置细胞注入能量实现对大鼠细胞放电过程的模拟，最终实现机器人的重定位。经验制图模块则用于环境认知地图的构建。该模块保存了机器人的位姿信息，当发生闭环时通过启用图松弛算法对已有的认知地图校正。导航模块根据构建的环境认知地图进行导航，实现机器人的自主移动功能。首先根据认知地图找到一条最优的路径，然后控制机器人沿着这条路径移动，当移动的过程中遇到了可移动的障碍物，则通过局部路径规划的算法避开障碍物到达目标点。



图5.1 总体结构图

Fig. 5.1 The diagram of General structure

本文基于类脑移动机器人导航系统各个模块之间的数据传输如图5.2所，整个系统有6个节点，每个节点是一个进程处理，摄像头和机器人为整个系统提供原始信息输入，即图像信息和速度信息。局部视图模块主要选取关键帧，储存起来作为视觉模板。对图像库进行匹配找到相似的图像信息，反馈给位姿细胞模块。位姿模块根据局部视图模块传递的信息和机器人的速度信息，进行路径积分估计机器人的位置，然后编码对应的栅格细胞，当发生闭环时，细胞激活完成重定位。经验制图模块根据位姿细胞发送的信息和机器人发送的速度信息，创建拓扑地图。路径规划模块根据经验制图模块创建的地图，进行路径规划，发送运动指令，控制机器人到达目标点。



图5.2 节点关系图

Fig. 5.2 The diagram of Node Relations

## 5.2 系统实现

类脑移动机器人导航系统是在ubuntu18.04系统下，使用ROS系统和RoboWare软件开发的。原始图片的采集使用单目摄像头传感器，速度信息由机器人提供。建图结果通过RVIZ软件实现可视化。

ROS机器人操作系统（Robot Operating System，ROS）[45]是一个软件开发框架，在2010年被发布出来，并随着近年来对ROS系统的更新和发展，ROS系统已经成为了目前机器人研究领域最热门的机器人软件开发平台，ROS最大的特点是能够通过代码的开源交流和重复利用，使开发人员能够快速整合不同的研究成果，这样不仅可以加快机器人产品的设计和算法开发，还为机器人开发人员提供了大量的工具、库和协议，已经成为广泛流行的机器人软件编写平台，主要是因为它的灵活性、健壮性和模块化。

Rviz是一款功能强大的ROS三维可视化工具。它允许用户查看模拟的机器人模型，记录机器人的传感器信息，并重放记录的传感器信息。通过可视化机器人的所见、所思、所做，用户可以调试机器人应用，从传感器输入到计划内（或计划外）的动作。

表5.1所示为实验系统所需的软硬件设备，图5.3为搭建的硬件实验设备。本论文为了验证能够在真实环境中应用，搭建了两套实验设备。（1）usb单目相机和P3-AT与笔记本电脑通过有线连接。（2）手机和Amigbot机器人通过wift与笔记本无线连接。实验结果证明均能正常运行。

表5.1 实验环境配置表

Tab. 5.1 Experimental environment configuration table

|  |  |
| --- | --- |
| 软硬件名 | 型号 |
| 移动机器人 | Pioneer3-AT/Amigobot |
| 单目相机 | Aoni A10 |
| 笔记本 | GT76 Titan DT 9SF-050CN |
| 手机 | 华为P20 |
| 操作系统 | Ubuntu18.04 |
| 第三方库 | Ppencv3.4 |



图5.3 硬件实验设备

Fig. 5.3 The pictures of experimental hardware

本实验场景是\*\*\*\*\*办公楼的四层走廊进行的。如图5.4所示，实验楼走廊的区域为30m40m的环境，黄色代表机器人实际运行的轨迹，通过键盘控制P3-AT移动机器人沿着黄色的线逆时针运行探索办公楼环境，平均运动速度设置为0.2米/秒。从单目相机获取的视频流输入到改进的RatSLAM算法中，构建环境的认知地图。在完成地图构建后，利用生成的认知地图进行路径规划和指定目标位置的自主导航，目标位置可以是地图中的任何位置，当P3-AT距离目标目的地0.2m时，就认为P3-AT已经达到目标。



图5.4 办公楼走廊平面图

Fig. 5.4 The floor plan of office building corridors

## 5.3 实验结果

本实验运行了1380s，P3-AT移动机器人里程计传感器生成了13776个坐标位置，单目相机采集了31490张图片，如图5.5所示为真实场景中的环境信息。基于本文改进的RatSLAM算法运行结果，选取了876个关键帧，生成1248个坐标位置。从运行情况来看，基于改进的RatSLAM系统可以正常的运行，能够在当前实验环境种实时的构建环境地图，如图5.6（A）（B）。在运行过程中，系统能够实时提取具有环境标志物信息的特征点，且较为准确的估计出机器人的移动轨迹和构建的环境地图。

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

图5.5 环境信息

Fig. 5.5 Environmental Information

|  |  |
| --- | --- |
|  |  |
| （A）670s时结果 | （B）1380s时结果 |

图5.6 办公楼走廊实验结果示意图

Fig. 5.6 The experimental results of office building corridors

本文继而在以上的环境地图基础上做了导航实验验证。机器人的起始位置坐标为（0，0），目标位置坐标为（14.6，31.3），使用全局路径规划算法为机器人导航规划路径，结果如图5.7（A）所示，其中绿色的线为规划的导航线路，五角星代表机器人的初始位置，四角图案代表导航的终点。把全局路径上的分段点赋给局部路径规划的终点，使机器人可以具有局部避障的功能而到达目标位置，如图5.7（B）（C）（D）所示。

|  |  |
| --- | --- |
| A | B |
|  |  |
| C | D |
|  |  |

图5.7 由认知图生成的全局路径

Fig. 5.7 Global paths generated from cognitive maps

本文实验是基于真实环境下进行的实时建图和导航实验，先利用视觉传感器感知并采集周围信息来构建认知地图，再根据认知地图进行导航实验。由于单目摄像头不能测量距离信息，无法实现精确的定位，特别是对机器人方向的定位，导致目前整个系统的鲁棒性较差。多次实验之后，才能使机器人按照路径规划的路线进行导航，然而很多时候在到达目标位置前迷失位置。但就总体而言，本文设计的导航系统基本实现了实时建图和有效导航的功能，实验也验证了系统的有效性。

## 5.4 本章小结

本章利用前文所改进的RatSALM以及在此基础上建立的路径规划算法搭建一套移动机器人类脑导航系统。该系统包括局部视图模块、位姿细胞模块、经验制图模块和导航模块。然后将该系统加载到P3-AT机器人中，实现移动机器人在真实室内环境的自主移动功能。P3-AT机器人在室内办公楼走廊环境内进行探索，构建出精确的环境认知地图，并在此基础上，成功实现了从起始点到目标点的导航，说明了本文所搭建的类脑导航系统的实用性。

# 第6章 总结

本文基于视觉感知和空间认知机制对移动机器人导航技术展开了深入探索。以室内环境下的导航为研究对象，基于图像处理和机器学习技术，采用ORB算法和词袋模型对RatSLAM算法中回环检测进行改进。同时本文又基于RatSLAM算法改进的基础上，实现了移动机器人类脑导航功能，使机器人可以自主移动。对于室内移动机器导航有着重要的指导意义。

本文的主要工作总结分为以下几个方面：

（1）本文采用ORB算法对环境图像进行特征提取，生成描述子，采用描述子间的汉明距离来度量图像间的相似性，以提高图像匹配的准确率。然后将描述子聚类成单词，建立树状字典，将原算法中顺序查询图像检索转换为树状字典查询，提高了图像检索的速度。同时本文又加入了关键帧筛选、动态阈值的选定、岛的匹配方式、序列匹配检测是否是正确的回环判断以及时间一致性检测，确保回环检测的准确率。本文在公开数据集和仿真环境中实验结果表明，本文提出的回环检测方案提高了RatSLAM算法的图像匹配的准确率与召回率，与此同时所提方法不会随着机器人在环境中探索时间的增长，而增加检索时间，确保始终能达到实时处理。

（2）基于改进的RatSLAM算法，搭建了一套基于单目视觉的移动机器人自主导航系统，实现了基于类脑认知地图的移动机器人导航。基于改进的RatSLAM方法构建的认知地图，使用迪杰斯特拉算法，实现全局路径规划，寻找当前位置到目标位置的最优全局路径，采用动态窗口法完成局部路径规划，实现了移动机器人沿着全局最优路径避开动态障碍物到达目标位置。本文在仿真实验和真实实验做了实验验证，证明了本文所提方法的可行性。

（3）搭建了仿真实验环境和真实实验平台，在公开数据集上和真实室内环境中验证了本文改进RatSLAM算法制图的优越性；在仿真环境和真实室内环境中验证了基于类脑认知地图的导航方法的可靠性。本文采用VC++实现整个类脑移动机器人导航算法，编写了局部视图细胞和导航模块细胞模块代码，整个系统实现了高度模块化，有利于今后对算法的扩展。

尽管本文对移动机器人类脑导航方法进行了一些研究，做了一些改进并取得了成功。但是本课题涉及的领域非常广泛，本论文所做的工作十分有限。并不能满足对这项工作的要求，尤其是导航的鲁棒性和准确性，还需要大量的工作来完善。进一步研究工作可以从以下几个方面进行：

（1）融合更多空间功能细胞。本文算法只是用了头朝向细胞、栅格细胞、位置细胞。近几年又发现了速度细胞、边界细胞、加速细胞等导航功能细胞。之后研究使用速度细胞编码机器人的速度信息；边界细胞编码障碍物；加速度细胞编码速度的变化信息。结合速度细胞和加速度细胞，编码速度，实现机器人精确的路径积分功能。

（2）提高闭环点检测精度。本文基于ORB改进的闭环检测算法与RatSLAM算法中基于绝对差总和的闭环检测相比，准确率有了明显的提高，然而周围的环境信息，如白墙、光线等场景，对其匹配的结果影响较大。因此，本文所改进的算法在环境场景认知方面有待提高。目前，通过神经网络识别环境中的物体信息来进行闭环检测取得了比较好的效果，即使同一物体在不同角度观察仍然能够识别。

（3）导航定位。本文构建了精准的认知地图，并根据认知地图实现了实时的导航功能，本文的位置识别是通过观察某一位置点采集到的图像并从过去的访问记忆中检索出具有相似图像的位置点，这种方式定位精度较差。目前基于激光或者深度摄像头的SLAM算法都能稳定的完成导航的功能。

# 参考文献

1. 西格沃特, 诺巴克什, 斯卡拉穆扎. 自主移动机器人导论[M]. 西安：西安交通大学出版社，2013.
2. 陈孟元. 移动机器人仿生SLAM算法研究[D].合肥：中国科学技术大学.
3. 张珍珍. 轮式移动机器人研究综述[J]. 电子技术与软件工程，2016(23)：120-120.
4. Jan S, Herbert P, Ouzounis CA. BatSLAM: Simultaneous Localization and Mapping Using Biomimetic Sonar[J]. Plos One 2013, 8:54076.
5. 斯白露, 罗壹凡. 空间记忆与类脑导航研究进展[J]. 人工智能，2020，14：17-32.
6. 蒲慕明, 徐波, 谭铁牛. 脑科学与类脑研究概述[J]. 中国科学院院刊，2016，031：723-736.
7. Saurabh Gupta et al. Cognitive Mapping and Planning for Visual Navigation[J]. International Journal of Computer Vision, 2019, : 1-20.
8. Burak Y, Fiete IR, Sporns O. Accurate Path Integration in Continuous Attractor Network Models of Grid Cells[J]. Plos Computational Biology, 2008, 5(2): e1000291.
9. 曾太平. 哺乳动物空间导航的神经计算模型研究[D]. 北京：中国科学院大学.
10. 桂旻, 斯白露. 基于海马体位置细胞的空间定位神经网络模型[J]. 计算机仿真，2019，036(007)：269-272.
11. Dissanayake M , Newman P , Clark S , et al. A solution to the simultaneous localization and map building (SLAM) problem[J]. IEEE Trans Ra, 2013, 17(3):229-241.
12. Moser E I, Kropff E Moser M B . Place cells, grid cells, and the brain's spatial representation system[J]. Annual Review of Neuroscience, 2008, 31(1): 69-89.
13. Basar T. A New Approach to Linear Filtering and Prediction Problems[M]. Wiley-IEEE Press, 2009.
14. Grisetti G , Stachniss C, Burgard W. Improved Techniques for Grid Mapping With Rao-Blackwellized Particle Filters[J]. IEEE Transactions on Robotics, 2007, 23(1): 34-46.
15. Mur-Artal R, Montiel JMM, Tardos JD. ORB-SLAM: A Versatile and Accurate Monocular SLAM System[J]. IEEE Transactions on Robotics, 2015, 31:1147-1163.
16. Davison A J, Reid I D, Molton N D, et al. MonoSLAM: Real-time Single Camera SLAM[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2007, 29(6): 1052-1067.
17. Klein G, Murray D. Parallel Tracking and Mapping for Small AR Workspaces[C]. IEEE and ACM International Symposium on Mixed and Augmented Reality. ACM, 2007:1-10.
18. Klein G, Murray D. Improving the Agility of Keyframe-Based SLAM[C]. European Conference on Computer Vision. Springer-Verlag, 2008:802-815.
19. Newcombe R A, Lovegrove S J, Davison A J. DTAM: Dense Tracking and Mapping In real-time[C]. IEEE International Conference on Computer Vision. IEEE, 2011:2320-2327.
20. Mahon I, Williams S B, Pizarro O, et al. Efficient View-Based SLAM Using Visual Loop Closures[J]. IEEE Transactions on Robotics, 2008, 24(5):1002-1014.
21. Engel J , Schps T , Cremers D . LSD-SLAM: Large-scale direct monocular SLAM[J]. European Conference on Computer Vision, 2014.
22. Hu J, Yuan M, Tang H. Hebbian learning analysis of a grid cell based cognitive mapping system[C]. IEEE Congress on Evolutionary Computation. 2016.
23. Zeng T, Si B. Mobile Robot Exploration Based on Rapidly-exploring Random Trees and Dynamic Window Approach[C]. 2019 5th International Conference on Control, Automation and Robotics (ICCAR). 2019.
24. O'Keefe J, Dostrovsky J. The hippocampus as a spatial map. Preliminary evidence from unit activity in the freely-moving rat[J]. Brain Research, 1971, 34:171-175.
25. Milford M, Wyeth G, Prasser D. RatSLAM[C]. IEEE International Conference on Robotics & Automation. 2004.
26. Ball D, Heath S, Wiles J. OpenRatSLAM: an open source brain-based SLAM system[J]. Autonomous Robots, 2013, 34.
27. Bo T, Shim VA, Yuan M. RGB-D based cognitive map building and navigation[C]. Intelligent Robots and Systems (IROS), 2013 IEEE/RSJ International Conference on. IEEE 2014.
28. Tang H, Huang W, Narayanamoorthy A. Cognitive memory and mapping in a brain-like system for robotic navigation[J]. Neural Networks, 2017, 87:27-37.
29. 张潇, 胡小平, 张礼廉. 一种改进的RatSLAM仿生导航算法[J]. 导航与控制，2015，14：73-79,47.
30. 秦国威, 孙新柱, 陈孟元. 基于HSV图像匹配的改进型RatSLAM算法研究[J]. 四川理工学院学报(自然科学版)，2018，31(05)：54-60.
31. Huajin Tang and Rui Yan and Kay Chen Tan. Cognitive Navigation by Neuro-Inspired Localization, Mapping, and Episodic Memory[J]. IEEE Trans. Cognitive and Developmental Systems, 2018, 10(3): 751-761.
32. 于乃功, 苑云鹤, 李倜. 一种基于海马认知机理的仿生机器人认知地图构建方法[J]. 自动化学报，2018，044：52-73.
33. Moser EI, Kropff E, Moser MB. Place cells, grid cells, and the brain's spatial representation system[J]. Annual Review of Neuroscience, 2008, 31, 69-89.
34. Hu J, Yuan M, Tang H. Hebbian learning analysis of a grid cell based cognitive mapping system[C]. IEEE Congress on Evolutionary Computation. 2016.
35. Taube J, Muller R, Ranck J. Head-direction cells recorded from the postsubiculum in freely moving rats. II. Effects of environmental manipulations[J]. Journal of Neuroscience, 1990, 10: 436-447.
36. Rublee E, Rabaud V, Konolige K. ORB: an efficient alternative to SIFT or SURF[C]. IEEE International Conference on Computer Vision, ICCV 2011, Barcelona, Spain, November 6-13, 2011.
37. Nichol MB, Knight TK, Dow T. Fast Algorithms for Mining Association Rules[J]. 1994.
38. Calonder M , Lepetit V , Strecha C. BRIEF: Binary Robust Independent Elementary Features[C]. European Conference on Computer Vision. Springer, Berlin, Heidelberg, 2010.
39. 董蕊芳, 柳长安, 杨国田. 一种基于改进TF-IDF的SLAM回环检测算法[J]. 东南大学学报:自然科学版，2019，49：251-258.
40. Giveki Davar. Scale-space multi-view bag of words for scene categorization[J]. Multimedia Tools and Applications, 2020, 80(1) : 1223-1245.
41. 李萍. 迪杰斯特拉算法的改进与实现[J]. 信息化建设, 2016(2).
42. 陈光明, 曾碧, 吕凌. 一种动态窗口下改进的机器人路径规划[J]. 计算机工程与应用, 2012, 48(22)：244-248．
43. M Seder, I Petrovic. Dynamic window based approach to mobile robot motion control in the presence of moving obstacles[C]. IEEE International Conference on Robotics&Automation, 2007: 1986-1991.
44. Desheng Feng,Lixia Deng,Tao Sun,Haiying Liu,Hui Zhang,Yang Zhao. Local Path Planning Based on an Improved Dynamic Window Approach in ROS[A]. 西安交通大学2020, 8.
45. 马丁内斯. ROS机器人程序设计[M]. 机械工业出版社, 2014.

# 致谢

时光飞逝，转眼间这三年紧张又繁忙的大学生活就此画上一个圆满的句号。回想起这三年，有苦有乐，学习过程是苦的，但是结果是甜的，达到今天这个高度，付出了自己很多心血和时间，当收获到了出乎意料的成果时，所付出的这一切都是值得的。

首先，我要由衷的感谢我的导师\*\*\*老师、\*\*\*老师。是\*老师给了我这个跳板，让我在学习生涯中上了一层楼，\*老师任教多年沉淀下来的学者气质像一盏明灯一样为我的科研道路指明方向。\*老师无微不至的关心和指导使我顺利完成了硕士论文的修改工作，在我修改毕业论文过程中遇到实验方面的难点，第一时间就会请教\*老师，\*老师会不厌其烦的给我进行详细讲解，这让我的问题迎刃而解。从课题选择，实验论证到论文书写，老师们都倾尽了大量的心血。读研期间，无论是代码编程还是论文写作，我的指导老师们都从零开始教起，他们的耐心指导，引导我独立思考，使我硕士课题的顺利开展和完成打下了坚实的基础。衷心的感谢两位老师对我的教诲。

同时，也要向我的父母发出最诚挚的感恩，没有他们在背后默默的支持与付出，就没有我今天的成果，他们无微不至的爱无以为报。

此外，还要感谢研究室的师兄师弟以及我的室友朋友们，在我遇到困难、没有思路的时候，他们会腾出自己的宝贵时间帮助我分析问题提出解决问题的思路。在跑实验走程序崩溃的时候，在实验室熬夜通宵的时候，也是他们的陪伴给予了我莫大的支持和鼓励。正因为他们的陪伴，让我这三年时光充满了欢声笑语，让枯燥无味的学习生活增添了几分乐趣。

毕业在即，在以后的工作生活中我会铭记各位老师的教诲，继续努力学习，成为国家栋梁之材，同时，我要向审阅论文的各位老师和参评论文答辩的各位老师致以感谢，感谢各位老师在百忙之中抽出宝贵的时间来查看论文并提出意见。