|  |  |
| --- | --- |
| 评阅人  老师 |  |

|  |  |
| --- | --- |
| 实验成绩  成绩 |  |



自动化学院本科生实验报告

**机器人学 课 程 实 验 报 告**

|  |
| --- |
| 专业 人工智能 班级 2101班 姓名 钱兴宇 学号 8207211912 |
| 预定时间 星期 五 节次 3,4 实际实验时间 星期 五 节次 3,4 |
| 地点 信息楼307 台号 授课教师 谢斌 指导教师 谢斌 |

|  |
| --- |
| 实验名称 实验一、机器人系统认知与示教编程  一、实验原理、目的与要求  **1.实验原理**  **（1）紧急停止按钮**  **默认情况下，机器人系统配有两个急停按钮，按下后将触发紧急停止。一个安装在控制柜上，另一个安装在示教器上。**  **（2）机器人系统组成**  **一个完整的机器人系统包括机器人本体，控制器，示教器，连接电缆，软件，末端执行器及其他附件。**   1. **机器人本体**   **机器人本体主要结构件材质为铸造铝合金，共6个自由度，每个关节电机都配有制动器。**  **本体由以下几个部分构成：**  **腕部——腕部有三个自由度，位于机器人输出端，包括轴4、轴5、轴6。**  **小臂——小臂位于大臂和腕关节之间，由轴3电机驱动。**  **大臂——大臂位于小臂和腰关节之间，由轴2电机驱动。**  **腰部——腰部位于大臂和基座之间，由轴1电机驱动。**  **基座——基座是整个机器人的基础，其后部布置有机器人的电气接口。**   1. **远程访问控制器Windows系统内打开机器人桌面控制程序HBRobotConfigure.exe** 2. **启动EtherCAT服务程序** 3. **启动机器人运动控制程序** 4. **桌面控制程序的使用**   **关于示教：**   1. **单击“连接”与控制器建立连接（机器人运动控制程序已启动）。** 2. **单击“添加机器人模型”，并单击“运行”，显示实际或仿真机器人状态。** 3. **左侧“使能”和“下电”，控制机器人的使能状态，使能后才能控制机器人运动。** 4. **“坐标系”控制机器人运动的坐标系，选择“轴”时，右侧数字对应的是机器人关节的运动控制按键。选择“世界”时，为笛卡尔空间运动，参考系为基坐标系，右侧数据对应的是x,y,z,Rx,Ry,Rz。选择“工具”时参考坐标系为工具坐标系，即tool的选项，默认为法兰坐标系。选择“工件”时参考系为工件坐标系，即wobj的选项，默认为基坐标系。e.左侧“Simulation”为ture时为仿真模式，false为真机模式。**   **关于脚本执行：**  **程序内输入脚本名称：如/hanbing/app/Lin0.MOV，单击左侧“启动”即可运行脚本。单击“停止”结束脚本运行。存点框内输入关节和位置名称，如：j1和p1，单击“关节”和“位置”记录示教点，在脚本编写时使用。运行脚本可自行创建和编写，后缀名称.MOV。放入至/hanbing/app下即可在界面中启动。**   1. **目的与要求** 2. **了解机器人系统的基本组成；** 3. **增强学生机器人动手操作能力；** 4. **熟悉机器人的基本操作。**   二、实验仪器设备及环境  **1、机器人**  **2、桌面控制应用程序**  三、实验内容步骤  **1、设备启动，启动EtherCAT服务程序，启动机器人运动控制程序。**  **2、尝试控制机器人的各个关节轴，并且尝试理解每个轴的运动会对末端位姿带来怎样的效果。**  **3、示教机器人，并保存机器人抓取工作空间中若干个点。**  **4、编写机器人脚本并运行机器人脚本，实现机器人的抓取作业。**  **机器人抓取脚本：**  **robot.CreateGrip()**  **robot.sleep(1000)**  **robot.MoveA(j2011,v1)--机器人初始化运动，运动到j2011点**  **robot.MoveL(p1012,v1)--运动到抓取点上方**  **robot.MoveL(p1013,v1)--运动到抓取点**  **robot.sleep(1000)robot.ControlGrip(1)—夹爪闭合抓取物体**  **robot.sleep(1000)--开始放置**  **robot.MoveL(p1012,v1)--运动到抓取点上方**  **robot.MoveL(p1014,v1)--运动到放置点上方**  **robot.MoveL(p1015,v1)--运动到放置点**  **robot.sleep(500)robot.ControlGrip(0)—夹爪松开放置物体**  **robot.sleep(500)--恢复到初始位置**  **robot.MoveL(p1014,v1)--运动到放置点上方**  **robot.MoveA(j2011,v1)—回到初始位置**  **robot.DestroyGrip()**  四、实验数据记录及数据处理    robot.CreateGrip()  robot.sleep(1000)  robot.MoveA(j2011,v1)--机器人初始化运动，运动到 j2011 点  robot.MoveL(p1012,v1)--运动到抓取点上方  robot.MoveL(p1013,v1)--运动到抓取点  robot.sleep(1000)  robot.ControlGrip(1)—夹爪闭合抓取物体  robot.sleep(1000)  --开始放置  robot.MoveL(p1012,v1)--运动到抓取点上方 robot.MoveL(p1014,v1)--运动到放置点上方 robot  .MoveL(p1015,v1)--运动到放置点  robot.sleep(500)  robot.ControlGrip(0)—夹爪松开放置物体  robot.sleep(500)  --恢复到初始位置  robot.MoveL(p1014,v1)--运动到放置点上方  5  robot.MoveA(j2011,v1)—回到初始位置  robot.DestroyGrip()  五、实验结论探讨及分析  **通过实验，了解了机器人系统的基本组成，包括机器人本体、控制器、示教器、连接电缆、软件、末端执行器及其他附件。掌握了机器人的基本操作，包括启动 EtherCAT 服务程序、启动机器人运动控制程序、示教机器人、编写机器人脚本并运行机器人脚本等。熟悉了机器人的各个关节轴的运动对末端位姿的影响，能够根据需要控制机器人的各个关节轴，实现机器人的抓取作业。**  **通过本次实验，了解了机器人系统的基本组成和基本操作，掌握了机器人的各个关节轴的运动对末端位姿的影响，学会了使用机器人桌面控制程序，增强了对机器人的动手操作能力，为今后的学习和工作打下了坚实的基础。** |



自动化学院本科生实验报告

**机器人学 课 程 实 验 报 告**

|  |
| --- |
| 专业 人工智能 班级 2101班 姓名 钱兴宇 学号 8207211912 |
| 预定时间 星期 二 节次 7,8 实际实验时间 星期 二 节次 7,8 |
| 地点 信息楼306 台号 授课教师 谢斌 指导教师 谢斌 |

|  |
| --- |
| 实验名称 实验二、机器人正运动学与工具坐标系标定  一、实验原理、目的与要求  **1.实验原理**  **正运动学部分：**  **对于一个具有n自由度的操作臂，它的所有连杆位置可由一组n个关节变量来确定。这样的一组变量常称为n\*1的关节向量。所有关节矢量组成的空间称为关节空间。操作臂在空间中位置与姿态是在空间相正交的轴上进行描述的，一般称这个笛卡尔空间，或任务空间和操作空间。操作臂的位置与姿态可以在关节空间、笛卡尔空间进行描述。**  **正运动学是利用机器人各个关节变量的信息求取机器人末端的位置与姿态。即实现关节空间到笛卡尔空间的变换。机器人的各个关节同末端之间的关系是通过连杆坐标系构件起来的。机器人相邻两个连杆之间有一个公共的关节轴。沿两个相邻连杆公共轴线方向的距离可以用一个参数描述，该参数称为连杆偏距,记为d.用另一个参数描述两相邻连杆绕公共轴线旋转的夹角，该参数称为关节角，记为θ。两关节的公垂线长度记为a（即连杆长度），连杆转角记为α。为了描述每个连杆于相邻连杆之间的相对位置关系，需要在每个连杆上定义一个坐标系。**  **根据坐标系所在连杆的编号对其命名，因此固连在连杆i上的坐标系称为坐标系{i}。通常按照如下方法确定连杆上的坐标系：坐标系{i}的Z轴于关节轴i重合，坐标系{i}的原点位于公垂线于关节轴i的交点处。坐标系{i}的X轴沿着公垂线指向关节i+1.按照右手定则，确定坐标系{i}的Y轴。**  **屏幕截图 2024-06-20 183515**  **修正 D-H 参数法中连杆坐标系的建立步骤：**   1. **找出关节轴，并标出这些轴线的延长线。** 2. **找出关节i − 1 和关节i之间的公垂线或关节轴i − 1 和i的交点，以关节轴i − 1 和关节 i的交点或公垂线与关节轴i − 1 的交点作为连杆坐标系{ i − 1}的原点。** 3. **规定Zi−1轴沿关节轴i − 1 的指向。** 4. **规定Xi−1轴沿公垂线的指向，如果关节轴i − 1 和 i 相交，则规定Xi−1轴垂直于关节轴 i − 1 和i所在的平面。** 5. **按照右手定则确定Yi−1 。** 6. **当第一个关节变量为 0 时，规定坐标系{0}和{1}重合。对于坐标系{n},其原点和Xn 的方向可以任意选择。但是在选择时，通常尽量使连杆参数为 0.**   **根据连杆坐标系的建立步骤（修正 D-H 参数法），可知连杆坐标系{i}在坐标系{ i − 1}**  **中的描述为：**     1. **目的与要求** 2. **学习机器人连杆坐标系建立的理论** 3. **学习 D-H 参数的建立方法** 4. **学习建立机器人的正运动学方程，并且用 MATLAB 实现** 5. **学习并且掌握机器人工具坐标系的标定方法**   二、实验仪器设备及环境  **1.珞石XB7机器人**  **2.2、MATLAB以及MATLAB机器人工具箱**  三、实验内容步骤  **正运动学部分：**   1. **观察XB7机器人构型，找出关节及连杆位置，了解连杆间的位置关系。标出关节位置及连杆长度。**   **屏幕截图 2024-06-20 184404**   1. **根据修正 D-H 参数法推导连杆坐标系{i}相对于坐标系{ i − 1}的变换矩阵。** 2. **根据修正 D-H 参数推导 XB7 机器人各连杆变换矩阵的表达式。**   **屏幕截图 2024-06-20 184709**  **屏幕截图 2024-06-20 184739**   1. **根据各个连杆的变换矩阵表达式推导正运动学表达式。** 2. **编写正运动学代码。**   四、实验数据记录及数据处理     1. 实验结论探讨及分析   **本次实验主要围绕机器人正运动学与工具坐标系标定展开，通过理论学习和实际操作，深入了解了机器人运动学的基本原理和应用方法。 在实验过程中，学习了机器人连杆坐标系建立的理论，掌握了 D-H 参数的建立方法，并通过 MATLAB 实现了机器人的正运动学方程。 通过本次实验，学习了机器人工具坐标系的标定方法，并通过实际操作，成功地对机器人的工具坐标系进行了标定。**  **由输出结果可知，本次实验结果基本符合预期，通过桌面示教器显示位置和 X-Y-Z 固定角描述姿态与X-Y-Z 固定角描述姿态变化到齐次矩阵描述姿态对比，在误差允许范围内两组数据基本相同，证明了所建立的正运动学模型的正确性。**  **通过本次实验，不仅掌握了机器人正运动学与工具坐标系标定的基本方法和技能，还培养了我们的团队合作精神和解决实际问题的能力。** |



自动化学院本科生实验报告

**机器人学 课 程 实 验 报 告**

|  |
| --- |
| 专业 人工智能 班级 2101班 姓名 钱兴宇 学号 8207211912 |
| 预定时间 星期 五 节次 5,6 实际实验时间 星期 五 节次 5,6 |
| 地点 信息楼307 台号 授课教师 谢斌 指导教师 谢斌 |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 实验名称 实验三、机器人逆运动学  一、实验原理、目的与要求  **1.实验原理**  **可解性：**  **求解操作臂的逆运动学方程是一个非线性问题。已知坐标系{n}相对于坐标系{0}的变换矩阵求解关节角度。对于6自由度的操作臂，由12个方程，其中6个是未知的。旋转矩阵分量生成的9个方程中，有3个是独立的。再加上变换矩阵的3个位置矢量共可写出6个方程，其中含有6个未知量。这些方程一般为非线性超越方程，很难求解。解是否存在的问题完全取决于操作臂的工作空间。工作空间是操作臂末端执行器所能到达的范围。若解存在，则被指定的目标点必须在工作空间内。操作臂末端执行器可以从任意方向到达的空间一般称为灵巧空间。机器人末端执行器可以到达的空间称为可达空间。**  **多解性：**  **在求解逆运动学方程时可能遇到的多解问题。例如：一个具有3个旋转关节的平面操作臂，有于从任何方位均可到达工作空间内的任何位置，因此在平面中有较大的灵巧工作空间。解的个数取决于操作臂的关节数量，它是连杆参数和关节运动范围的函数。非零参数越多，解的最大数目就越大。对于一个全部为旋转关节的六自由度操作臂来说，可能多达16中解。常见六自由度操作臂，一般存在8个不同的解。**  **解法：**  **与线性方程组不同，非线性方程组没有通用的求解算法。如果关节变量能够通过一定算法确定，这种算法可以求出与已知位置和姿态相关的全部关节变量，操作臂便是可解的。操作臂的求解方法可内成两大类：封闭解和数值解法。由于数值解法的迭代性质，因此要比相应的封闭解法的求解速度慢。但数值解法更通用。封闭解是基于解析形式的解法。所有包含转动关节和移动关节的串联型6自由度机构均是可解的。但是这种解一般是数值解，对于6自由度机器人来说，只有在特殊情况下才有解析解。这种存在解析解的机器人具有如下特性：存在几个正交关节轴或者有多个绕x轴的旋转角α为0或±90.常见的6自由度机器人一般都存在解析解。**   1. **目的与要求** 2. **了解逆运动学相关的基本概念及相关特性：多解性，奇异问题等。** 3. **学习建立机器人的逆运动学模型并编写代码求解。** 4. **理解工具坐标系的意义并且将工具坐标系加入到逆解程序中。**   二、实验仪器设备及环境  **（1）珞石ROKAEXB7机器人**  **（2）MATLAB**  三、实验内容步骤   1. **定义机器人的 DH 参数和工具坐标系变换矩阵。** 2. **输入末端执行器的位姿矩阵 KPS44。该矩阵描述了末端执行器在空间中的位置和姿态。** 3. **调用 IKHide 函数计算关节角度。将 KPS44 作为输入参数传递给 IKHide 函数，函数会返回一个 1x6 的关节角度向量 Radian。** 4. **输出关节角度的最优解 Radian。**   **补充的代码如下：**  **%%求取角度1的两个解 Q1(1),Q(2)分别为两个解**  **Q1(1)= atan2(y,x);%atan2括号内的需要补全**  **Q1(2)= atan2(-y,-x);%atan2括号内的需要补全**  K = (x\*x + y\*y + z\*z - a2\*a2 - a3\*a3 - d4\*d4 + a1\*a1 - 2 \* a1\*x\*cos(Q1(1)) - 2 \* a1\*y\*sin(Q1(1))) / (2 \* a2);  if (a3\*a3 + d4\*d4 - K\*K>=0)  ss = sqrt(a3\*a3 + d4\*d4 - K\*K);  Q3(1)= atan2(a3, d4) - atan2(K, ss);  **Q3(2) = atan2(a3, d4) - atan2(K, -ss); %需补充此处**  else  Q3(1)= 10000;  Q3(2) = 10000;  End  s1 = sin(Q1(i)); s3 = sin(Q3(j));  c1 = cos(Q1(i)); c3 = cos(Q3(j));  s23 = ((-a3-a2\*c3)\*z-(c1\*x+s1\*y-a1)\*(d4-a2\*s3))/(z^2+(c1\*x+s1\*y)^2);  c23 = ((-d4+a2\*s3)\*z+(a3+a2\*c3)\*(c1\*x+s1\*y-a1))/(z^2+(c1\*x+s1\*y)^2);  **Q23(i,j) = atan2(s23,c23);%atan2括号内的需要补全**  **Q2(i,j)=Q23(i,j)-Q3(j);%需补充此处**  s23 = sin(Q23(i,j)); c23 = cos(Q23(i,j));  sss1 = -ax\*s1 + ay\*c1;  sss2 = -ax\*c1\*c23 - ay\*s1\*c23 + az\*s23;  sssss1 = -nx\*(c1\*c23\*s4 - s1\*c4) - ny\*(s1\*c23\*s4 + c1\*c4) + nz\*(s23\*s4);  sssss2 = nx\*((c1\*c23\*c4 + s1\*s4)\*c5 - c1\*s23\*s5) + ny\*((s1\*c23\*c4 - c1\*s4)\*c5 - s1\*s23\*s5) - nz\*(s23\*c4\*c5 + c23\*s5);  **Q6(i,j) = atan2(sssss1,sssss2);%atan2括号内的需要补全**  四、实验数据记录及数据处理   |  |  |  |  | | --- | --- | --- | --- | | **起 始 关 节 位 置** | **终止关节位置** | **计算位置及姿态** | **计算关节角度** | | **0 0 0 0 1.57 0** | **0.2 0.2 0.2 0.2 0.2 0.2** |  | **0.2000 0.2000 0.2000 0.2002 0.2000 0.1998** | | **0 0 0 0 1.57 0** | **0.2 -0.2 0.2 -0.2 0.5 0.2** |  | **0.2000 -0.2000 0.2000 -0.2000 0.5000 0.2000** | | **0 0 0 0 1.57 0** | **0.2 0.4 0.2 -2.94 0.4 -2.84** |  | **0.2000 0.4000 0.2000 -0.2940 0.4000 -0.2840** |   五、实验结论探讨及分析  **在本次实验中，成功实现了机器人逆运动学的计算过程，得到了一组满足条件的关节角度作为最优解。这些结果不仅验证了我们的计算方法的正确性，也为实际机器人的运动控制提供了理论依据。通过本实验，我深刻理解了逆运动学在机器人技术中的应用价值和重要性，对逆运动学相关的基本概念及相关特性有了详细深刻的认识，理解了工具坐标系的意义。** |



自动化学院本科生实验报告

**机器人学 课 程 实 验 报 告**

|  |
| --- |
| 专业 人工智能 班级 2101班 姓名 钱兴宇 学号 8207211912 |
| 预定时间 星期 五 节次 7,8 实际实验时间 星期 五 节次 7,8 |
| 地点 信息楼307 台号 授课教师 谢斌 指导教师 谢斌 |

|  |
| --- |
| 实验名称 实验四、五 机器人手眼标定  一、实验原理、目的与要求  **1.实验原理**  **机器人视觉，是指不仅要把视觉信息作为输入，而且还要对这些信息进行处理，进而提取出有用的信息提供给机器人，从而使机器人能够感知环境及目标，提取有效信息，引导及辅助机器人工作的视觉技术。**  **我们将机器人类比成人来看：**  **人的眼睛来定位物体及周边环境，引导人走到桌子前，使手臂带着手去拾取物体。整个过程和我们即将学到的机器人视觉过程相当，本课程主要涉及用于机械臂抓取的机器人视觉系统。**  **机器人视觉系统主要作用是识别定位，引导机器人作业，按照传感器类型主要分为以下两种：2D机器人视觉及3D机器人视觉。其中，2D机器人视觉主要用于平面物体的识别定位，输入图像主要是彩色RGB图像或者灰度图像；3D机器人视觉主要用于立体物体的识别定位，输入主要为深度图或点云，二者可根据相机内参进行相互转换。**  **手眼标定就是要理清“眼睛”和“手”的关系，解决“眼睛中看到的东西，手要怎么才能去碰到”的问题。**  **通常情况下，对于机器人的手眼系统，有两种基本方式：Eye-In-Hand 和****Eye-To-Hand。在第一种方式下，视觉系统与机器人的末端执行器的相对位姿是固定不变的。在第二种方式下，视觉系统与机器人的基座相对位姿固定不变，不随机器人的运动而改变。**  **本次实验内容是针对Eye-To-Hand这种配置展开的，且我们所采用的手眼标定方法为便于理解的AX=B求解方法。**  **XA=B标定法：**    **代求量X为相机坐标系到机器人基坐标系的关系，先将上节课内容的工具坐标系进行输入。**  **2.目的与要求**  **（1）了解手眼标定相关的基本概念及其意义。**  **（2）学习并掌握眼在手外的手眼标定法，根据所述原理编写代码进行手眼标定求解。**  **（3）能够使用手眼标定结果，移动机器人去到相机坐标系中的选定点。**  二、实验仪器设备及环境  **1、珞石ROKAE XB7机器人**  **2、标定板**  **3、MATLAB**  三、实验内容步骤   1. **先预备一张空白纸张，并在其上标注六个鲜明的圆点。** 2. **通过matlab 获取图像。** 3. **相机视野下所拍摄物体的点云获取，运行convertPointcloud.m（convertPointcloud是将彩色图和深度图合成点云图的程序）获取test.ply（test.ply是点云文件，点云文件有很多种格式，ply这种格式包含xyzRGB六维信息）。** 4. **利用cloudcompare软件，分别选取点云中6个点的坐标将其记录在camerapoints中，其中这六个点为步骤一中对应的标识点。** 5. **然后将机器人工具坐标系调整到tool11，然后用工具末端触碰点云中选中的点。（移动机械臂，使用机械臂末端接触相机坐标系下的六个点，记录六个点在基座标系下的坐标，并保存在robotpoints文件中）。即可获取手眼标定矩阵。** 6. **将被抓物体放置在云台上，利用matlab中image acquisition应用程序采集彩色图像并保存为image File文件；然后保持纸质标定板（物体）不动，转换到深度相机进行拍照，同样保存为image File文件。** 7. **更改pixcel\_get.py文件中彩色图片路径，运行.py文件即可获取被抓物体中心点的像素坐标。** 8. **打开xyzAquisition.m文件，更改像素坐标和深度照片的路径，将手眼标定得到的R\_T矩阵补充至xyzAquisition.m文件中，并运行文件即可得到机器人基座标系下的坐标;** 9. **将所得到坐标代入机器人运行程序中，即可移动机械臂至指定位置并抓取物体。**   四、实验数据记录及数据处理        五、实验结论探讨及分析  **本次实验主要围绕机器人手眼标定展开，通过理论学习和实际操作，深入了解了机器人视觉系统的基本原理和应用方法。**  **在实验过程中，首先学习了机器人视觉系统的基本概念和作用，了解了 2D 机器人视觉和 3D 机器人视觉的区别和应用场景。接着，学习了手眼标定的基本原理和方法，掌握了 AX=B 求解方法和 XA=B 标定法，并通过实际操作，成功地对机器人进行了手眼标定。实验结束，成功实现了机器人的手眼标定，机器人能够实现抓取目标物体。实验结果符合预期。**  **通过本次实验，不仅掌握了机器人手眼标定的基本方法和技能，还培养了的团队合作精神和解决实际问题的能力。** |



自动化学院本科生实验报告

**机器人学 课 程 实 验 报 告**

|  |
| --- |
| 专业 人工智能 班级 2101班 姓名 钱兴宇 学号 8207211912 |
| 预定时间 星期 节次 实际实验时间 星期 节次 |
| 地点 台号 授课教师 谢斌 指导教师 谢斌 |

|  |
| --- |
| 实验名称 实验六、栅格地图构建  一、实验原理、目的与要求  **1.实验原理**  **利用给定的移动机器人位姿和采集的激光数据，构建 2D 的占用栅格地图。红点代表当前时刻移动机器人的位姿，绿点代表当前时刻激光采集的 数据，灰度图代表栅格地图。**  **移动机器人坐标（Body Frame）和栅格地图坐标（Map Frame）的定义如下：**    **设激光和移动机器人坐标系完全重合。**  **编程说明：**  **（1）practice.mat 数据文件**  **practice.mat 中包括激光的测距数据 d（ranges），激光测距的角度（scanAngles），移**  **动机器人的位姿，采集时间。**  **（2）bresenham.p 计算栅格中由机器人所在的栅格单元和激光测量障碍物的栅格单位这两点来决定的直线，返回属于这条直线上栅格单元索引。用法示例请见 example\_bresenham.m**  **（3）example\_test.m 主程序，用于测试代码**  **（4）occGridMapping.m 需要补充完整的代码。**  **2.目的与要求**  **1、 学习机器人栅格地图构建的理论**  **2、 掌握世界坐标系、机器人坐标系、栅格地图之间的变换**  **3、 学习并且掌握栅格地图构建的方法，并且用 MATLAB 实现**  二、实验仪器设备及环境  **1、MATLAB 以及 MATLAB 机器人工具箱**  三、实验内容步骤  **（1）practice.mat 数据文件**  **practice.mat 中包括激光的测距数据 d（ranges），激光测距的角度（scanAngles），移动机器人的位姿，采集时间。**  **（2）bresenham.p**  **计算栅格中由机器人所在的栅格单元和激光测量障碍物的栅格单位这两点来决定的直线，返回属于这条直线上栅格单元索引。**  **（3）example\_test.m 主程序，用于测试代码**  **（4）补充完整occGridMapping.m 的代码。**  **补充代码如下：**  **N = size(pose,2);**  **numScans = size(scanAngles);**  **for j = 1:N % for each time,**  **x = pose(1, j);**  **y = pose(2, j);**  **theta = pose(3, j);**    **ix\_robot = ceil(x \* myResol) + myOrigin(1);**  **iy\_robot = ceil(y \* myResol) + myOrigin(2);**  **%disp([ix\_robot, iy\_robot]);**    **% Find grids hit by the rays (in the gird myMap coordinate)**  **rays = ranges(:, j);**  **x\_occ = rays .\* cos(scanAngles + theta) + x;**  **y\_occ = -rays .\* sin(scanAngles + theta) + y;**  **ix\_occ = ceil(x\_occ \* myResol) + myOrigin(1);**  **iy\_occ = ceil(y\_occ \* myResol) + myOrigin(2);**    **% Find occupied-measurement cells and free-measurement**  **% cells(查找占用的测量单元和空闲的测量单元)**  **occ = sub2ind(size(myMap), iy\_occ, ix\_occ); % Convert to 1d**    **free = [];**  **for k = 1:numScans**  **[ix\_free, iy\_free] = bresenham(ix\_robot, iy\_robot, ix\_occ(k), iy\_occ(k));**  **free = [free; iy\_free, ix\_free];**  **end**  **free = sub2ind(size(myMap), free(:, 1), free(:, 2)); % Convert to 1d 转换为1维**  **% Update the log-odds更新**  **myMap(occ) = myMap(occ) + lo\_occ;**  **myMap(free) = myMap(free) - lo\_free;**  **% Saturate the log-odd values**  **myMap(myMap > lo\_max) = lo\_max;**  **myMap(myMap < lo\_min) = lo\_min;**    **%% Visualize the myMap as needed**  **imagesc(myMap); hold on;**  **plot(ix\_robot, iy\_robot, 'rx', 'LineWidth', 3); % indicate robot location with red 'x'**  **plot(ix\_occ, iy\_occ, 'g.', 'MarkerSize', 6); % indicate laser hits with green dots**  **hold off;**  **colormap('gray'); % set colormap to gray**  **pause(0.001);**  **end**  四、实验数据记录及数据处理  **1.实验结果**  **栅格地图构建结果如下：**  **6-16-2**  五、实验结论探讨及分析  **通过完成occGridMapping.m中函数的编写与测试，实现了基于占据栅格地图算法的机器人环境地图构建。该函数成功地将机器人的激光雷达扫描数据、当前位姿，以及预设参数，整合用于构建一个二维占据栅格地图。地图通过累计每次扫描得到的占用与自由空间信息，逐步反映了环境结构。本实验成功实现了栅格地图构建，对数似然比的使用提高了地图构建的鲁棒性和准确性。构建栅格地图为后续路径规划、避障策略的制定提供了可靠的地图基础。** |



自动化学院本科生实验报告

**机器人学 课 程 实 验 报 告**

|  |
| --- |
| 专业 人工智能 班级 2101班 姓名 钱兴宇 学号 8207211912 |
| 预定时间 星期 节次 实际实验时间 星期 节次 |
| 地点 台号 授课教师 谢斌 指导教师 谢斌 |

|  |
| --- |
| 实验名称 实验七、扩展卡尔曼滤波定位  一、实验原理、目的与要求  **1.实验原理**  **1. 状态估计与预测**  **状态方程：机器人运动模型（f(x, u)）描述了系统状态随时间的变化，基于当前状态x和控制输入u。EKF首先基于上一时刻的状态和控制量，通过状态方程预测下一时刻的状态x\_pred。**  **2. 预测阶段的雅可比矩阵**  **雅可比矩阵J\_F：jacobF(x, u)计算了状态方程关于状态x的偏导数，即状态转移矩阵，用于预测过程中状态协方差的传播。**  **3. 观测模型**  **观测方程：h(x)表示如何从真实状态得到观测值，通过GPS传感器观测到的位置信息来建模。观测值z反映了系统的外部观测。**  **4. 观测阶段的雅可比矩阵**  **雅可比矩阵J\_H：jacobH(x)计算了观测模型关于状态x的偏导数，用于将状态误差协方差转换到观测空间，评估观测噪声对状态估计的影响。**  **5. 更新步骤**  **卡尔曼增益计算：结合预测的协方差矩阵和观测模型的雅可比矩阵，计算卡尔曼增益K，这是衡量预测误差和观测误差之间信任度的指标。**  **状态更新：将观测值与预测值的差异乘以卡尔曼增益，以此来修正预测状态，得到更准确的后验状态估计。**  **协方差更新：更新状态协方差矩阵，反映新获得信息后的不确定性减少。**  二、实验仪器设备及环境  **1、MATLAB 以及 MATLAB 机器人工具箱**  三、实验内容步骤  **（1）function u = doControl(time)**  **运动控制量仿真函数，该函数用来仿真机器人运动控制量 u=[线速度 角速度]T。u 是与**  **时间 time 相关的函数。**  **（2）function x = f(x, u)**  **机器人运动模型。通过当前机器人位姿 x 和运动控制量 u,计算出下一个时刻机器人的位**  **姿。**  **（3）function jF = jacobF(x, u)**  **机器人运动模型的雅可比矩阵 jF 的计算函数。**  **（4）function z = h(x)**  **机器人观测模型。设定移动机器人仅携带 gps 传感器进行观测，通过当前机器人位姿获**  **得当前 gps 观测值 z。**  **（5）function jH = jacobH(x)机器人观测模型的雅可比矩阵 jH 的计算函数。**  **（6）function [z, x, xd, u] = Observation(x, xd, u)**  **机器人导航仿真函数。通过给定机器人前一时刻真实位姿 x、航迹推算值 xd 和运动控**  **制量 u，计算当前时刻机器人真实位姿 x、带噪声的航迹推算值 xd、带噪声的运动控制量 u**  **和带噪声的观测值 z。**  **（7）function [] = ExtendedKalmanFilterLocalization()**  **主程序，用于仿真机器人的导航运动，并通过 EKF 实现机器人的实时定位，最终绘制**  **出机器人的真实位姿、gps 观测值、航迹推算定位值和 EKF 定位值。**  **补充的EKF 的核心部分代码如下：**  **% step1：预测**  **% （1）计算出当前时刻机器人位姿的预测值，xPred**  **xPred = f(xEst, u);**  **% （2）计算上述预测值的协方差，PPred**  **jF = jacobF(xEst, u);**  **PPred = jF \* PEst \* jF' + Q;**  **% step2：观测**  **% （1）计算真实观测和预测观测间的误差，即新息，y**  **zPred = h(xPred);**  **y = z - zPred;**  **% （2）计算新息的协方差，S**  **jH = jacobH(xPred);**  **S = jH \* PPred \* jH' + R;**  **% step3：更新**  **% （1）更新机器人位姿预测为估计值，xEst**  **K = PPred \* jH' / S;**  **xEst = xPred + K \* y;**  **% （2）更新估计值的协方差，PEst**  **PEst = (eye(length(xEst)) - K \* jH) \* PPred;**  四、实验数据记录及数据处理  **1.实验结果**  **7-1**  五、实验结论探讨及分析  **实验结果展示了Ground Truth（真实轨迹）、GPS观测点、Dead Reckoning（纯预测轨迹）以及EKF估计轨迹，直观地体现了EKF滤波器在减少定位误差方面的作用。**  **Error Ellipse表明估计位置的协方差，随着滤波过程的进行，该椭圆逐渐收缩，反映了定位不确定性的降低。**  **本次实验运用了扩展卡尔曼滤波器进行移动机器人的实时定位，仿真结果显示EKF能够有效减少定位误差。通过对比真实轨迹、航迹推算以及EKF估计轨迹，验证了EKF算法在机器人导航中的实用性和鲁棒性。** |



自动化学院本科生实验报告

**机器人学 课 程 实 验 报 告**

|  |
| --- |
| 专业 人工智能 班级 2101班 姓名 钱兴宇 学号 8207211912 |
| 预定时间 星期 节次 实际实验时间 星期 节次 |
| 地点 台号 授课教师 谢斌 指导教师 谢斌 |

|  |
| --- |
| 实验名称 实验八、移动机器人导航  一、实验原理、目的与要求  **1.实验原理**  **（1）建图算法学习：Gmapping**  **实验中 turtlebot3 机器人利用自身的单线激光雷达在仿真环境中探索，接受雷达数据， 实时地构建出环境地图并确定自己的位置，即为 SLAM（Simultaneous Localization And Mapping）技术，其中本实验采用的建图算法为经典的激光 SLAM 算法——Gmaping。 Gmaping 在移动机器人建图中被广泛使用，它使用粒子滤波器来估计移动机器人在环境中的位置，并同时构建环境的地图。**  **（2）导航算法学习：A\*、Dijkstra、DWA**  **Turtlebot3\_navigation 中默认可使用 A\*与 Dijkstra 算法作为全局轨迹规划算法，而局部轨迹规划算法默认采用 DWA 算法。**  **A\*算法是一种启发式搜索算法，常用于在图形或网络中找到最短路径。它结合了广度 优先搜索和启发式评估函数的特点，以有效地探索潜在的最优解。A 算法使用一个估计函数来评估当前节点到目标节点的代价，并根据代价选择下一个要探索的节点。这个估计函数由两部分组成：从起始节点到当前节点的实际代价（通常表示为 g 值）和从当前节点到目标节点的预估代价（通常表示为 h 值）。A\*算法具有较好的搜索效率和路径质量。**  **Dijkstra 算法是一种用于在带权重的图中找到最短路径的算法。它通过不断选择最短路径的节点来逐步扩展搜索范围，直到找到目标节点或搜索完所有节点。Dijkstra 算法使用一个距离值来记录从起始节点到当前节点的最短路径长度，并根据距离值选择下一个要探索的节点。算法的核心思想是贪心策略，即每次选择距离最短的节点进行扩展。Dijkstra 算法适用于没有负权边的图，并能找到起始节点到其他所有节点的最短路径。**  **DWA 算法是一种用于移动机器人导航的动态路径规划算法。它通过结合机器人的运动模型和环境感知数据，实现实时路径规划和避障。DWA 算法基于采样搜索的思想，通过在机器人的动态窗口中生成一组候选速度样本，并评估每个样本的轨迹的优劣来选择最佳的速度指令。DWA 算法可以在动态环境中进行快速的路径规划和避障，适用于移动机器人的实时导航。**   1. **实验目的**   **（1）学习机器人操作系统 ROS 的安装、环境配置与测试，并掌握工作空间的创建。**  **（2）学习机器人全局和局部路径规划理论。**  **（3）学习机器人导航算法在机器人操作系统 ROS 中的实现。**   1. **实验要求**   **在 ROS 环境下，构建有障碍的环境场景，设定起点 A 和目标点 B，通过路径规划和避障控制，实现移动机器人自主从 A 到 B 的导航。**  **（1）ROS 的安装和环境配置**  **-ROS 的版本不限。若课程项目选择实物的，可选择实物支持的 ROS 版本进行安装。**  **（2）创建 ROS 工作空间**  **创建一个 catkin 工作空间，并编译。**  **（3）通过操控 Turtlebot3 机器人采用 gmapping 方法实现 SLAM 建图**  **（4）设定目标点，实现 Turtlebot3 的自主导航**  二、实验仪器设备及环境  **1、VMware、ROS 及 Turtlebot3 功能包**  三、实验内容步骤  **ROS 下 Turtlebot3 仿真及建图与导航实现**  **（1）安装 Turtlebot3 依赖包**  **打开终端，输入以下命令安装 Turtlebot3 依赖的包文件**  **（2）创建项目并安装 Turtlebot3**  **（3）使用 Turtlebot3 进行 SLAM 建图**   1. **打开新终端，进入项目文件“catkin\_turtlebot3”。** 2. **输入命令启动 Gazabo，是通过 roslaunch 命令启动 src 文件夹中特定节点。** 3. **打开新终端，启动 SLAM 进行建图，该命令是在可视化工具 rviz 中打开并进行 SLAM。** 4. **打开新终端，输入键盘控制的命令。** 5. **打开新终端，输入命令保存地图，地图被保存至主目录中，包含 2 个文件： map.pgn：地图图片， map.yaml：地图信息。**   **（4）使用扫描地图进行自动导航仿真**   1. **运行 Gazabo。** 2. **读取地图并运行导航程序。** 3. **初始激光数据可能无法对其地图，此时利用 Rviz 上方工具栏 2D Pose Estimate 箭头标出，机器人在地图中的实际位置，随后即可成功匹配。** 4. **此时点击上方红色箭头按钮：2D Nav Goal，随后在地图上任意地点点击设定导航目标位置，小车便开始自主规划移动。**   四、实验数据记录及数据处理  **运行 Gazabo：**  **屏幕截图 2024-06-11 093717**  **启动 SLAM 进行建图：**  **屏幕截图 2024-06-11 115729**  **建图完毕：**  **屏幕截图 2024-06-11 120422**  **保存地图：**  **屏幕截图 2024-06-11 120708**  **读取地图并运行导航程序：**  **屏幕截图 2024-06-19 161428**  五、实验结论探讨及分析  **本次实验中，使用 Turtlebot3 机器人平台，结合 ROS ，成功实现了 SLAM功能，完成了目标点设定与自主导航的任务。通过 gmapping 生成的地图精确度高，能较好地反映实际环境布局。在SLAM过程中，Turtlebot3的定位精度较高，能够在复杂的环境中准确识别自身位置。机器人能够根据设定的目标点，自主规划路径并避障到达指定地点，展现了良好的自主导航能力。即使在遇到意外障碍时，也能重新规划路径，体现出较好的适应性和鲁棒性。** |