Jindong_Shi   2022.03.25

激光 SLAM 第一次作业

1. 了解 Linux 系统:阅读《鸟哥的 Linux 私房菜》自学前三部分内容,或利用互联网进行学习,简答以下问题;

(1)与巩固列举三个你常用的 Linux 命令,并说明他们的功能。

1. cd 用来切换目录，比如 cd .. 返回上一级目录，cd ~ 返回家目录，cd / 返回根目录
2. ls 列出当前目录下的所有文件，后面可跟参数，比如 ls -l 可显示当前目录下所有文件及其权限
3. touch 用来创建文件，mkdir 用来创建文件夹

(2)与巩固一句话简要介绍 Vim 的功能,如何在 Vim 中进行插入和删除,如何保存并退出 Vim?

Vim 时一款 Linux 上功能强大，使用便捷的文本编辑器。

首先 VIM 常用的有四个模式，:

- 正常模式 (Normal-mode)
- 插入模式 (Insert-mode)
- 命令模式 (Command-mode)
- 可视模式 (Visual-mode)

正常模式一般用于浏览文件，也包括一些复制、粘贴、删除等操作。这时击键时，一些组合键就是 vim 的功能键，而不会在文本中键入对应的字符。启动 Vim 后默认位于正常模式。不论是什么模式，按一下 <Esc> 键，就会切换到正常模式。在正常模式下，通常按 i 或 I 可进入插入模式，该模式启动以后，就会进入编辑状态，通过键盘输入内容。[1]

Vim 可 删除多行文本，在按退出键进入正常模式后，输入:n1,n2d，其中 n1 和 n2 指的是起始行号和结束行号，d 是删除(delete)关键字。

在正常模式下，按下:wq 可保存并退出。

(3)与巩固列举两种常用的 Linux 压缩和解压缩命令。

1. tar 与 gzip 命令结合实现文件的打包和压缩，tar 负责打包，gzip 用来压缩，压缩文件格式为 xxx.tar.gz, 具体命令如下:

    压缩文件: tar -zcvf 打包文件名.tar.gz 需要压缩的文件路径/文件
    解压缩文件: tar -zxvf 打包文件名.tar.gz

2. tar 与 bzip2 命令结合实现文件的打包和压缩，tar 负责打包，bzip2 用来压缩，压缩文件格式为 xxx.tar.bz2, 具体命令如下:

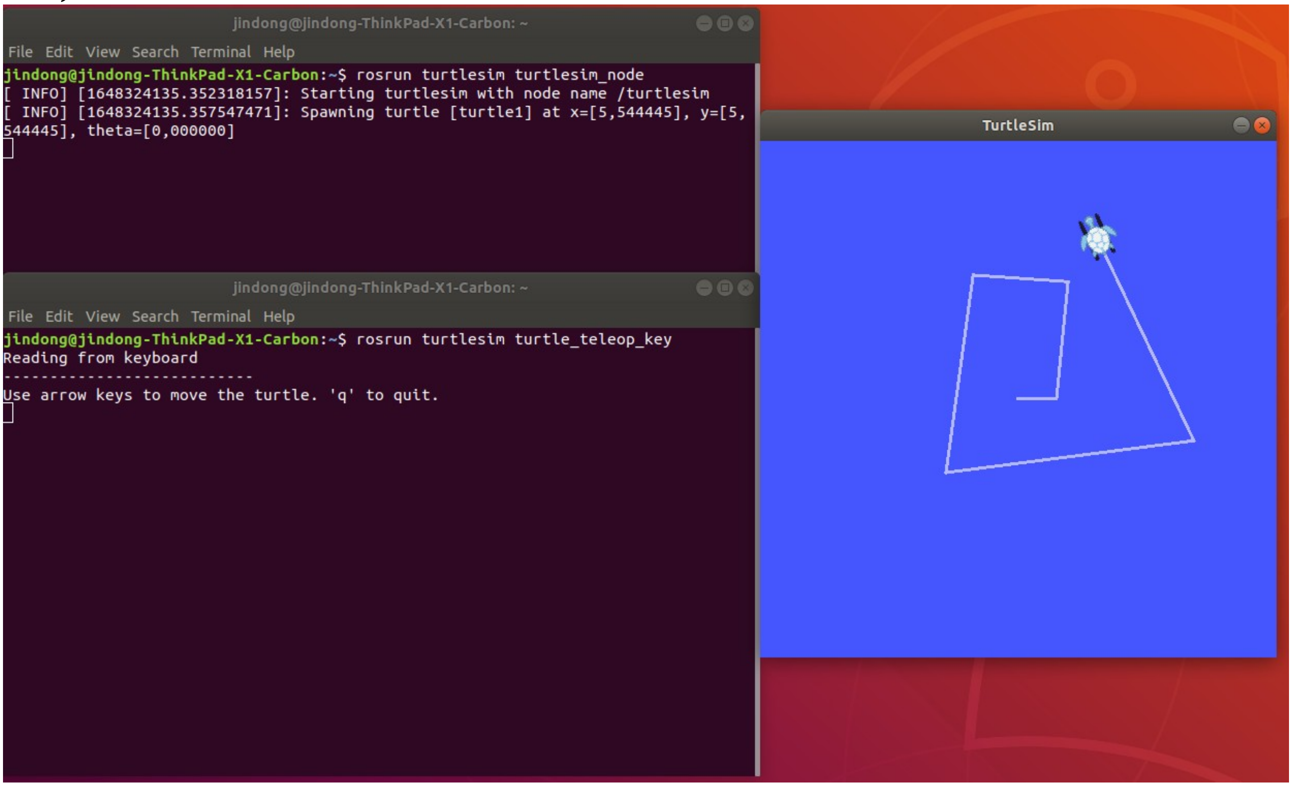
    压缩文件: tar -jcvf 打包文件名.tar.bz2 需要压缩的文件路径/文件
    解压缩文件: tar -jxvf 打包文件名.tar.bz2

3. 使用 zip

    压缩文件: zip   打包文件名.zip 需要压缩的文件
    压缩文件夹: zip -r 打包文件名.zip 需要压缩的文件夹
                 参数 -r 用来递归压缩目录内的所有文件和目录
    解压缩文件: unzip   打包文件名.zip

Jindong_Shi 2022.03.25

2. 了解 ROS:观看 ROS 免费公开课或前往 ROS 官网学习官方教程,安装好 ROS,提供运行小海龟跑的截图;



3. 学习机器人姿态描述入门材料,完成坐标转换推导;(3 分)

设机器人的世界坐标为 x_a , y_a ,其相对于世界坐标系的方向为 θ_a (右手坐标系)。假设机器人旁边有一物体在世界坐标系下的位姿为(x_b, y_b , θ_b),请问:

(1)该物体相对于机器人的位置和朝向是什么,即该物体在当前机器人坐标系下的位姿是多少?

(x_b, y_b , θ_b)

(2)机器人此时朝它的正前方(机器人坐标系 X 轴)行进了 d 距离,然后又转了 θ_d 角,请问物体此时在这一时刻机器人坐标系下的位姿是多少?

(1) 机器人坐标系 a 到世界坐标 w 的转换矩阵:

$$T_a^w = \begin{bmatrix} \cos\theta_a & -\sin\theta_a & x_a \\ \sin\theta_a & \cos\theta_a & y_a \\ 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} R_a^w & t_a^w \\ 0 & 1 \end{bmatrix}$$

那么世界坐标系 w 到机器人坐标系 a 的转换矩阵为:

$$T_w^a = \begin{bmatrix} {R_a^w}^{-1} & -{R_a^w}^{-1} t \\ 0 & 1 \end{bmatrix} = \begin{bmatrix} \cos\theta_a & \sin\theta_a & -\cos\theta_a \cdot x_a - \sin\theta_a \cdot y_a \\ -\sin\theta_a & \cos\theta_a & \sin\theta_a \cdot x_a - \cos\theta_a \cdot y_a \\ 0 & 0 & 1 \end{bmatrix}$$

物体坐标系 b 到世界坐标系 w 的转换矩阵:

$$T_b^w = \begin{bmatrix} \cos\theta_b & -\sin\theta_b & x_b \\ \sin\theta_b & \cos\theta_b & y_b \\ 0 & 0 & 1 \end{bmatrix}$$

那么物体坐标系 b 到机器人坐标系 a 的转换矩阵为:

$$T_b^a = T_w^a \cdot T_b^w = \begin{bmatrix} \cos\theta_a\cos\theta_b + \sin\theta_a\sin\theta_b & -\cos\theta_a\sin\theta_b + \sin\theta_a\cos\theta_b & \cos\theta_a\cdot(x_b - x_a) + \sin\theta_a(y_b - y_a) \\ -\sin\theta_a\cos\theta_b + \cos\theta_a\sin\theta_b & \sin\theta_a\sin\theta_b + \cos\theta_a\cos\theta_b & \sin\theta_a(x_a - x_b) + \cos\theta_a(y_b - y_a) \\ 0 & 0 & 1 \end{bmatrix}$$

$$= \begin{bmatrix} \cos(\theta_b - \theta_a) & \sin(\theta_a - \theta_b) & \cos\theta_a(x_b - x_a) + \sin\theta_a(y_b - y_a) \\ \sin(\theta_b - \theta_a) & \cos(\theta_b - \theta_a) & -\sin\theta_a(x_b - x_a) + \cos\theta_a(y_b - y_a) \\ 0 & 0 & 1 \end{bmatrix}$$

那么该物体相对于机器人位置和朝向为:

$$(x_b^a, y_b^a, \theta_b^a) = T2V(T_b^a) = (T_b^a(0,2),\ T_b^a(1,2),\ \mathrm{atan2}(T_b^a(1,0),\ T_b^a(0,0)))$$

$$= (\cos\theta_a(x_b - x_a) + \sin\theta_a(y_b - y_a),\ \sin\theta_a(x_a - x_b) + \cos\theta_a(y_b - y_a),\ \theta_b - \theta_a)$$

(2) 当机器人朝它正前方行进了 d 距离, 然后又转了 θ_d 角, 那么

$$T_{a'}^w = T_a^w \cdot T_d = \begin{bmatrix} \cos\theta_a & -\sin\theta_a & x_a \\ \sin\theta_a & \cos\theta_a & y_a \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} \cos\theta_d & -\sin\theta_d & d \\ \sin\theta_d & \cos\theta_d & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$= \begin{bmatrix} \cos\theta_a\cos\theta_d - \sin\theta_a\sin\theta_d & -\cos\theta_a\sin\theta_d - \sin\theta_a\cos\theta_d & \cos\theta_a\cdot d + x_a \\ \sin\theta_a\cos\theta_d + \cos\theta_a\sin\theta_d & -\sin\theta_a\sin\theta_d + \cos\theta_a\cos\theta_d & \sin\theta_a\cdot d + y_a \\ 0 & 0 & 1 \end{bmatrix}$$

$$= \begin{bmatrix} \cos(\theta_a + \theta_d) & -\sin(\theta_a + \theta_d) & \cos\theta_a\cdot d + x_a \\ \sin(\theta_a + \theta_d) & \cos(\theta_a + \theta_d) & \sin\theta_a\cdot d + y_a \\ 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} R_{a'}^w & t_{a'}^w \\ 0 & 1 \end{bmatrix}$$

$$\Rightarrow T_w^{a'} = \begin{bmatrix} {R_{a'}^w}^{-1} & -{R_{a'}^w}^{-1} t_{a'}^w \\ 0 & 1 \end{bmatrix} = \begin{bmatrix} \cos(\theta_a + \theta_d) & \sin(\theta_a + \theta_d) & -\cos(\theta_a + \theta_d)(\cos\theta_a\cdot d + x_a) - \sin(\theta_a + \theta_d)(\sin\theta_a\cdot d + y_a) \\ -\sin(\theta_a + \theta_d) & \cos(\theta_a + \theta_d) & \sin(\theta_a + \theta_d)(\cos\theta_a\cdot d + x_a) - \cos(\theta_a + \theta_d)(\sin\theta_a\cdot d + y_a) \\ 0 & 0 & 1 \end{bmatrix}$$

Jindong_Shi 2022.03.25

此时物体b到机器人坐标系a'的变换矩阵为:

$$T_b^{a'} = T_w^{a'} T_b^{w} = \begin{bmatrix} \cos(\theta_a+\theta_d)\cos\theta_b + \sin(\theta_a+\theta_d)\sin\theta_b & -\cos(\theta_a+\theta_d)\sin\theta_b + \sin(\theta_a+\theta_d)\cos\theta_b & x_b^{a'} \\ -\sin(\theta_a+\theta_d)\cos\theta_b + \cos(\theta_a+\theta_d)\sin\theta_b & \sin(\theta_a+\theta_d)\sin\theta_b + \cos(\theta_a+\theta_d)\cos\theta_b & y_b^{a'} \\ 0 & 0 & 1 \end{bmatrix}$$

$$= \begin{bmatrix} \cos(\theta_a+\theta_d-\theta_b) & \sin(\theta_a+\theta_d-\theta_b) & x_b^{a'} \\ -\sin(\theta_a+\theta_d-\theta_b) & \cos(\theta_a+\theta_d-\theta_b) & y_b^{a'} \\ 0 & 0 & 1 \end{bmatrix}$$

所以该物体相对于机器人的位置和朝向为:

$$x_b^{a'} = \cos(\theta_a+\theta_d)\cdot x_b + \sin(\theta_a+\theta_d)\cdot y_b - \cos(\theta_a+\theta_d)(\cos\theta_a\cdot d + x_a) - \sin(\theta_a+\theta_d)(\sin\theta_a\cdot d + y_a)$$

$$= \cos(\theta_a+\theta_d)(x_b - \cos\theta_a\cdot d - x_a) + \sin(\theta_a+\theta_d)(y_b - \sin\theta_a\cdot d - y_a)$$

$$y_b^{a'} = -\sin(\theta_a+\theta_d)\cdot x_b + \cos(\theta_a+\theta_d)\, y_b + \sin(\theta_a+\theta_d)(\cos\theta_a\cdot d + x_a) - \cos(\theta_a+\theta_d)(\sin\theta_a\cdot d + y_a)$$

$$= \sin(\theta_a+\theta_d)(\cos\theta_a\cdot d + x_a - x_b) - \cos(\theta_a+\theta_d)(\sin\theta_a\cdot d + y_a - y_b)$$

$$\theta_b^{a'} = \mathrm{atan2}\left(\frac{-\sin(\theta_a+\theta_d-\theta_b)}{\cos(\theta_a+\theta_d-\theta_b)}\right) = \mathrm{atan2}\left(\frac{\sin(\theta_b-\theta_a-\theta_d)}{\cos(\theta_b-\theta_a-\theta_d)}\right) = \theta_b - \theta_a - \theta_d$$

Jindong_Shi 2022.03.25

4. 完成基础数学坐标转换的代码作业。

编写代码如下:

```
// 机器人B在坐标系O中的坐标：
Eigen::Vector3d B( x: 3,  y: 4,  z: M_PI);
// 坐标系B到坐标O的转换矩阵：
Eigen::Matrix3d TOB;
TOB << cos( x: B( index: 2)), -sin( x: B( index: 2)), B( index: 0),
       sin( x: B( index: 2)),  cos( x: B( index: 2)), B( index: 1),
          0,           0,          1;

// 坐标系O到坐标B的转换矩阵:
Eigen::Matrix3d TBO = TOB.inverse();

// 机器人A在坐标系O中的坐标：
Eigen::Vector3d A( x: 1,  y: 3,  z: -M_PI / 2);

// 求机器人A在机器人B中的坐标：
Eigen::Vector3d BA;
// TODO 参照第一课PPT
// start your code here (5~10 lines)
Eigen::Matrix3d TOA;
TOA << cos( x: A( index: 2)), -sin( x: A( index: 2)), A( index: 0),
       sin( x: A( index: 2)),  cos( x: A( index: 2)), A( index: 1),
        0,           0,         1;
Eigen::Matrix3d TBA = TBO * TOA;
BA = Eigen::Vector3d( x: TBA( row: 0, col: 2),  y: TBA( row: 1, col: 2),  z: std::atan2( y: TBA( row: 1, col: 0),  x: TBA( row: 0, col: 0)));
// end your code here
```

计算结果与答案一致:

```
/home/jindong/Lidar_SLAM/Exercise/ch1/HW1/basicTransformStudy/cmake-build-debug/basicTransformStudy
The right answer is BA: 2 1 1.5708
Your answer is BA:      2      1 1.5708

Process finished with exit code 0
```

参考文献:

[1] 精通 VIM ， 此文就够了

https://zhuanlan.zhihu.com/p/68111471