

激光SLAM的发展和 应用

主讲人 曾书格

超凡创新技术负责人
电子科技大学硕士





激光SLAM简介



1. SLAM的定义与分类



2. SLAM的框架



3. 激光SLAM



4. 相关数学基础



激光SLAM简介



1. SLAM的定义与分类



2. SLAM的框架



3. 激光SLAM



4. 相关数学基础



SLAM的定义

定义

- Localization: 在给定地图的情况下, 估计机器人的位姿。 移动机器人导航中最重要, 也最难
- Mapping: 在给定机器人位姿的情况下, 估计环境地图。 可以不实时
- SLAM: 同时估计机器人的位姿和环境地图。 移动机器人中最重要的内容

SLAM解决的问题

- 机器人在环境中的位姿;
- 导航过程中需要的环境地图。



SLAM的分类

基于传感器的分类

类型	适用场景	特点
二维激光SLAM	室内（可解决95%的问题）	成本低
三维激光SLAM	室外、测绘	成本高 信息量较大
视觉SLAM	室内室外通用	信息量丰富
声纳SLAM	水下，绘制海图	



二维激光
SLAM



三维激光
SLAM



视觉SLAM



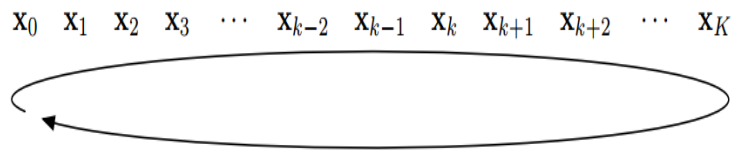
声纳SLAM



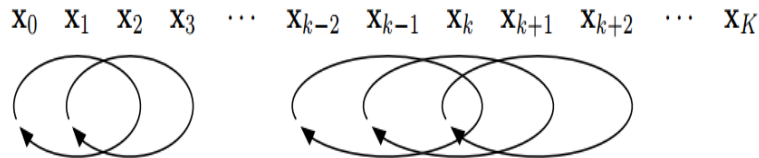
SLAM的分类

基于后端的分类

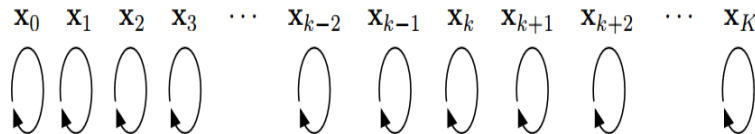
类型	特点
图优化方法 Graph-based	估计全部位姿 (最通用)
滑动窗口方法 sliding-window	估计固定数量的位姿 (VIO, VO)
滤波器方法 filter-based	估计当前位姿 (Pose-Fusion)



图优化方法



滑动窗口方法



滤波器方法



激光SLAM简介



1、SLAM的定义与分类



2、SLAM的框架



3、激光SLAM



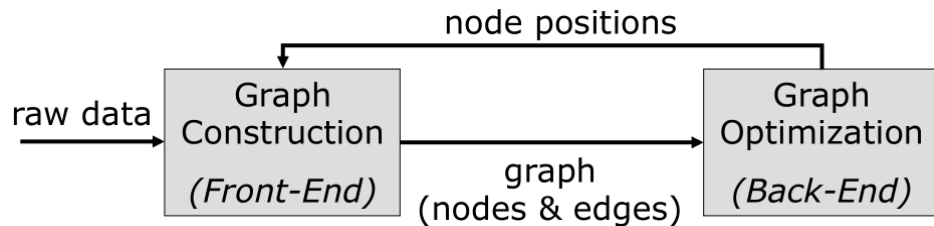
4、基本数学基础



Graph-based SLAM的框架

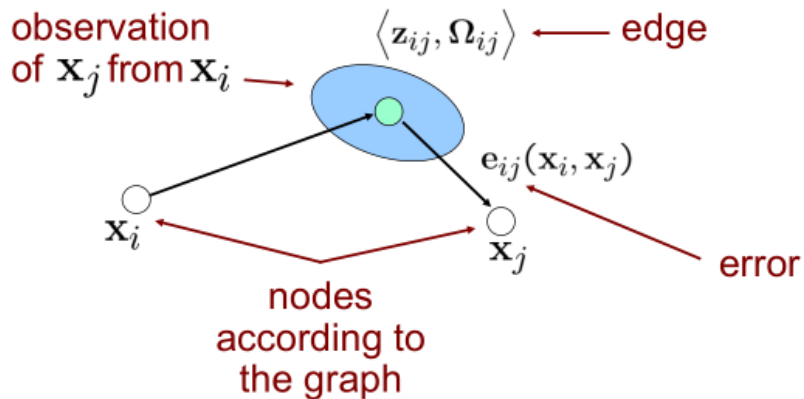
Graph-based SLAM

- 前端匹配(局部匹配、帧间匹配)
- 回环检测
- 后端优化



图的结构

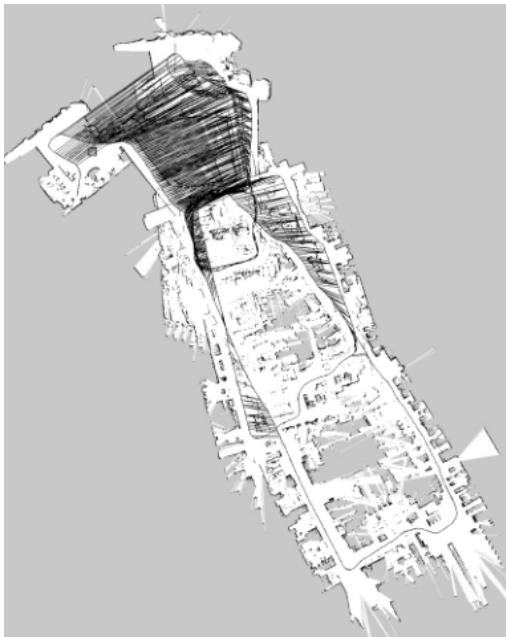
- Graph: 表示SLAM的过程
- Node: 机器人的位姿
- Edge: 节点之间的空间约束关系



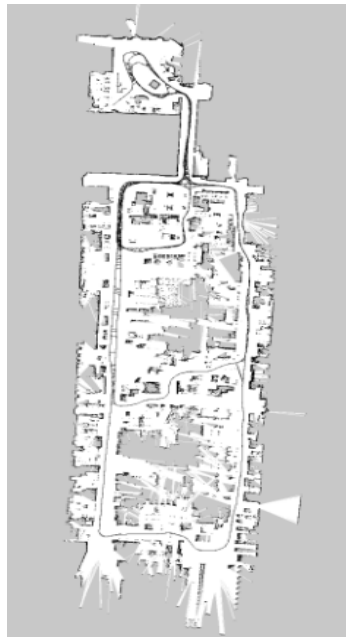


Graph-based效果

图优化前



图优化后

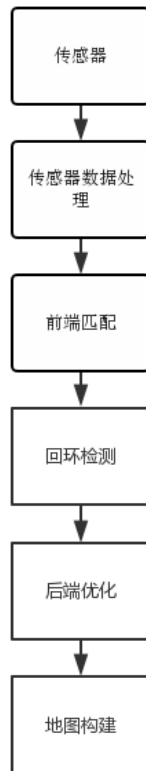




激光SLAM的pipeline

pipeline

- 1. 数据处理(激光雷达去畸变、里程计标定)
- 2. 激光帧间匹配(计算两帧激光之间的相对位姿)
- 3. 激光回环检测(识别回到访问过的环境)
- 4. 后端优化(Pose-Graph, 非线性最小二乘优化)





激光SLAM

数据处理--非常重要

1. 里程计标定

- 机械标称值不代表实际值，实际误差可能较大
- 里程计的精度对于机器人定位和建图至关重要

2. 激光雷达运动畸变去除

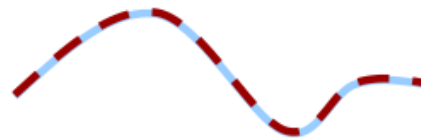
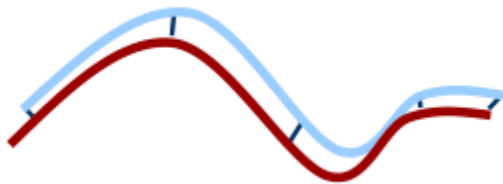
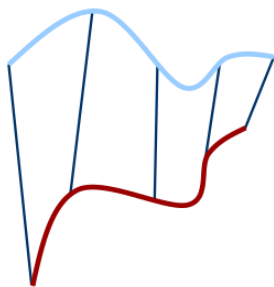
- 一帧激光数据的采集需要时间
- 运动畸变会让数据严重失真，影响匹配精度



激光SLAM

帧间匹配算法--核心部分

- ICP(Iterative Closest Point)
- PI-ICP(Point-to-Line Iterative Closest Point)
- NICP(Normal Iterative Closest Point)
- CSM(Correlation Scan Match)
- NDT(Normal Distance Transform)
- Optimization-Based
- Feature-Based





回环检测

方法	特点
Scan-to-Scan	计算量最小，容易产生歧义
Scan-to-Map	计算量中等，中等歧义
Map-to-Map	计算量最大，基本不产生歧义

后端优化

- 高斯牛顿方法
- LM方法



激光SLAM简介



1、SLAM的定义与分类



2、SLAM的框架



3、激光SLAM



4、基本数学基础



2D激光SLAM



1、2D激光SLAM的介绍



2、2D激光SLAM的发展



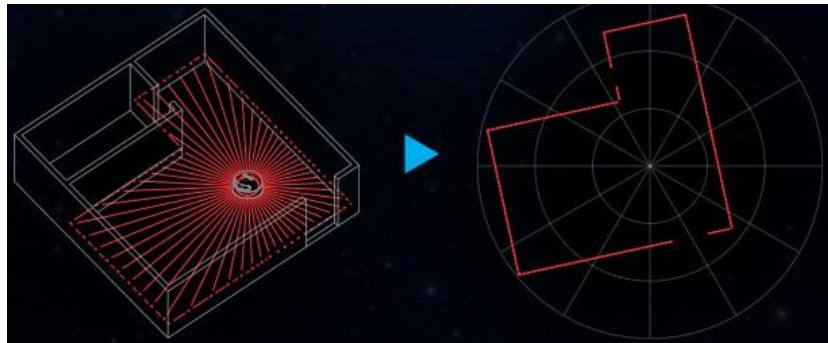
3、2D激光SLAM的应用



2D激光SLAM的介绍

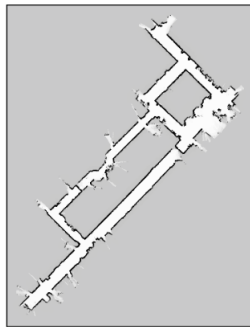
2D激光SLAM的输入

- IMU数据
- 里程计数据
- 2D激光雷达数据



2D激光SLAM的输出

- 覆盖栅格地图
- 机器人的轨迹 or PoseGraph





2D激光SLAM的介绍

2D激光SLAM的帧间匹配方法

方法	特点
PL-ICP(Point-to-Line ICP)	点到线的距离，更符合室内结构化场景， 依赖初始解
CSM(Correlation Scan Match)	暴力搜索，能找到最优解
Optimization-Based	依赖于地图的梯度，可以轻易的引入额外约束

2D激光SLAM的回环检测方法

- Scan-to-Map
- Map-to-Map
- Branch and Bound & Lazy Decision
- State of Art: CSM+梯度优化



2D激光SLAM的发展

Filter-based

方法	特点
EKF-SLAM(90年代)	随着环境的增大，状态量急剧增大
FastSLAM(2002~2003)	粒子滤波，每一个粒子表示一个地图
Gmapping(2007)	FastSLAM的升级版本
Optimal RBPF(2010)	Gmapping的基础上进一步优化

Graph-based

方法	特点
Karto-SLAM(2010)	首个基于优化的2D激光SLAM的开源算法，很经典
Cartographer(2016)	与Karto-SLAM原理类似，更完整



2D激光SLAM的应用

数据的预处理---非常重要!!!

- 轮式里程计的标定
- 激光雷达运动畸变去除
- 不同系统之间的时间同步

实际环境中的问题

- 1. 环境变化
- 2. 几何结构相似环境
- 3. 建图的操作复杂
- 4. 全局定位
- 5. 地面材质的变化
- 6. 机器人载重的改变



2D激光SLAM的趋势—与视觉融合

视觉提供的信息

- 高精度的里程信息
- 信息量丰富的视觉地图

融合解决的问题

- 环境变化
- 几何结构相似环境
- 建图的操作复杂
- 全局定位
- 地面材质的变化
- 机器人载重的改变



3D激光SLAM



1、3D激光SLAM的介绍



2、3D激光SLAM的发展



3、3D激光SLAM的应用



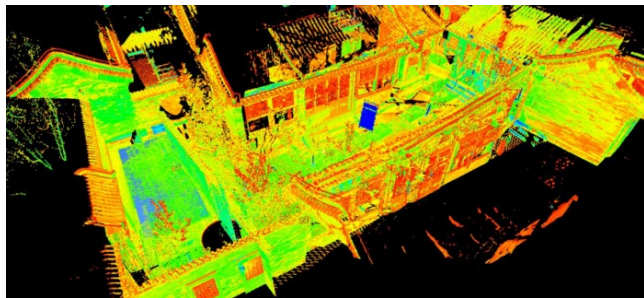
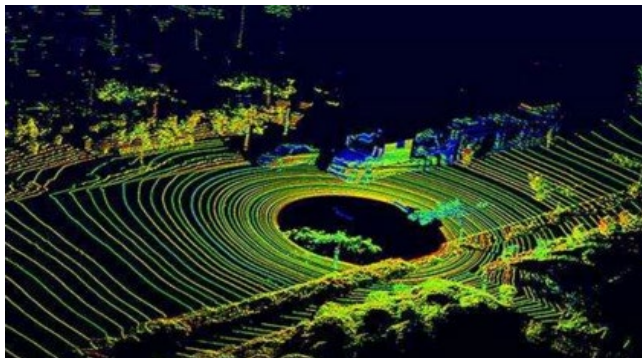
3D激光SLAM的介绍

3D激光SLAM的输入

- IMU数据
- 里程计数据
- 3D激光雷达数据

3D激光SLAM的输出

- 3D点云地图
- 机器人的轨迹 or PoseGraph





3D激光SLAM的介绍

3D激光SLAM的帧间匹配方法

方法	特点
Point-to-Plane ICP	点到面的距离，类似于2D的点线
Plane-to-Plane ICP(GICP)	面到面的距离
NDT	划分网格，拟合高斯分布，速度快
NICP	在ICP的基础上，引入法向量信息，精度高
IMLS-ICP	对点云进行局部曲面拟合
Feature-based Method	提取特征点和描述符进行匹配



3D激光SLAM的应用

数据的预处理

- 轮式里程计的标定
- 激光雷达运动畸变去除
- 不同系统之间的时间同步

与视觉融合

- 3D激光雷达为视觉特征提供深度信息
- 视觉辅助激光雷达进行运动畸变去除
- 视觉辅助回环检测
- 视觉提供精确里程信息



激光SLAM中的问题

1、退化环境(Degeneration Environment)

2、地图的动态更新(Map Update)

3、全局定位(Global Localization)

4、动态环境定位(Dynamic Localization)



激光SLAM简介



1、SLAM的定义与分类



2、SLAM的框架



3、激光SLAM



4、基本位姿转换



数学基础：位姿表示和转换矩阵

位姿表示

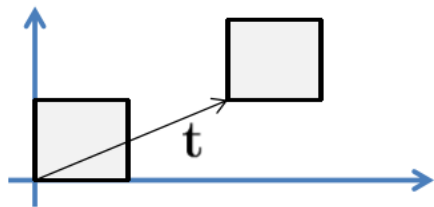
机器人在坐标系W中的坐标： (x, y, θ)

机器人坐标系B到坐标系W的转换矩阵：

$$T_B^W = \begin{bmatrix} \cos \theta & -\sin \theta & x \\ \sin \theta & \cos \theta & y \\ 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} R & t \\ 0 & 1 \end{bmatrix}$$

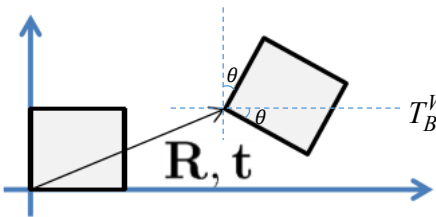
思考：旋转矩阵 $R = \begin{bmatrix} \cos \theta & -\sin \theta \\ \sin \theta & \cos \theta \end{bmatrix}$ 是如何得到的？

仅平移，无旋转：



$$T_B^W = \begin{bmatrix} 1 & 0 & x \\ 0 & 1 & y \\ 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} I & t \\ 0 & 1 \end{bmatrix}$$

平移+旋转：



$$T_B^W = \begin{bmatrix} \cos \theta & -\sin \theta & x \\ \sin \theta & \cos \theta & y \\ 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} R & t \\ 0 & 1 \end{bmatrix}$$



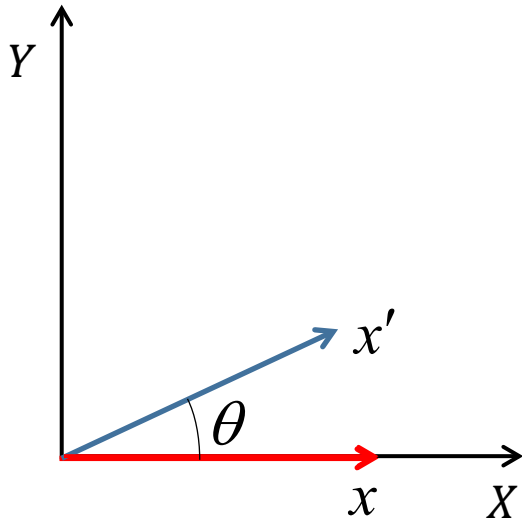
数学基础：位姿表示和转换矩阵

主动旋转(向量旋转)

- 向量旋转
- 生成在同坐标系下的新向量
- 多用于计算机图形学(CG)领域

$$x' = R(\theta) \cdot x$$

$$R(\theta) = \begin{bmatrix} \cos \theta & -\sin \theta \\ \sin \theta & \cos \theta \end{bmatrix} \text{ 旋转矩阵}$$





数学基础：位姿表示和转换矩阵

主动旋转(向量旋转)

- 坐标系由一组基底定义，坐标系中的任意一个向量都表示为基底的线性组合
- 向量 (a, b) 实际表达式:

$$\begin{bmatrix} a \\ b \end{bmatrix} = a \begin{bmatrix} 1 \\ 0 \end{bmatrix} + b \begin{bmatrix} 0 \\ 1 \end{bmatrix}$$

- 对于向量的旋转，可以等价于对基底的旋转:

$$R(\theta) \begin{bmatrix} a \\ b \end{bmatrix} = a R(\theta) \begin{bmatrix} 1 \\ 0 \end{bmatrix} + b R(\theta) \begin{bmatrix} 0 \\ 1 \end{bmatrix}$$



数学基础之位姿表示和转换矩阵

主动旋转(向量旋转)

- X 轴基向量 $(1,0)$ 的旋转

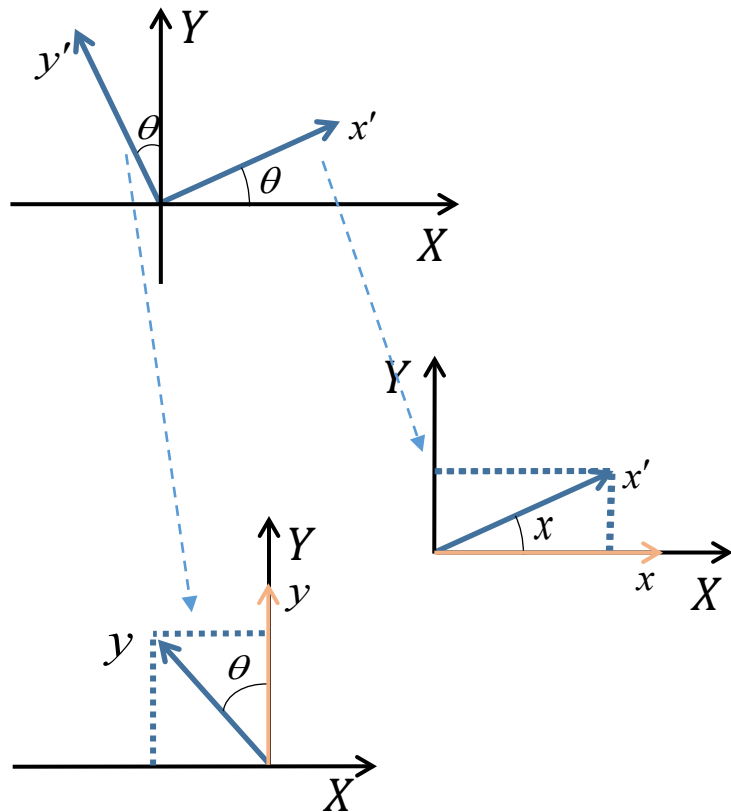
$$x = [1, 0]^T \quad x' = [\cos \theta, \sin \theta]^T$$

$$x' = R(\theta) \cdot x = \begin{bmatrix} R_{11} & R_{12} \\ R_{21} & R_{22} \end{bmatrix} \cdot \begin{bmatrix} 1 \\ 0 \end{bmatrix} \Rightarrow \begin{cases} R_{11} = \cos \theta \\ R_{21} = \sin \theta \end{cases}$$

- Y 轴基向量 $(0,1)$ 的旋转

$$y = [0, 1]^T \quad y' = [-\sin \theta, \cos \theta]^T$$

$$y' = R(\theta) \cdot y = \begin{bmatrix} R_{11} & R_{12} \\ R_{21} & R_{22} \end{bmatrix} \cdot \begin{bmatrix} 0 \\ 1 \end{bmatrix} \Rightarrow \begin{cases} R_{12} = -\sin \theta \\ R_{22} = \cos \theta \end{cases}$$



思考：如果顺时针旋转，旋转矩阵 R 的表达式是否与逆时针一致？需要提示的是按照约定，逆时针旋转角度为正。



数学基础之位姿表示和转换矩阵

被动旋转(坐标系旋转)

- 坐标系旋转
- 某一个物理量在不同坐标系下的表示
- 多用于状态估计领域

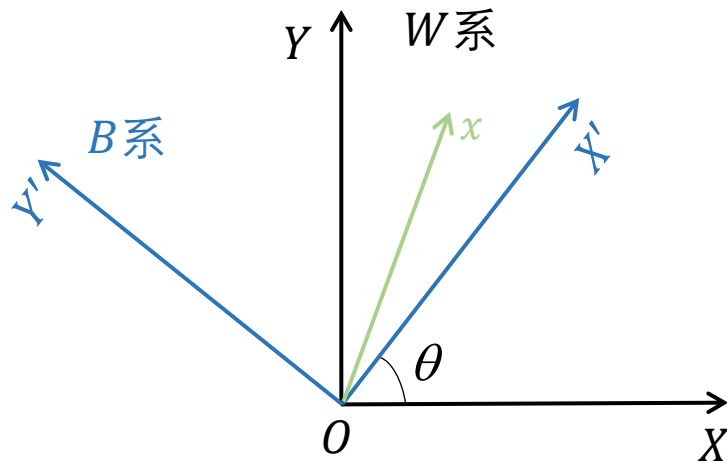
$$x_W = R_B^W x_B$$

$$R_B^W = \begin{bmatrix} \cos \theta & -\sin \theta \\ \sin \theta & \cos \theta \end{bmatrix}$$

其中,

x_B 表示物理量 x 在 B 坐标系中的坐标,

x_W 表示物理量 x 在 W 坐标系中的坐标。



B 系由 W 系逆时针旋转得到



数学基础之位姿表示和转换矩阵

被动旋转(坐标系旋转)

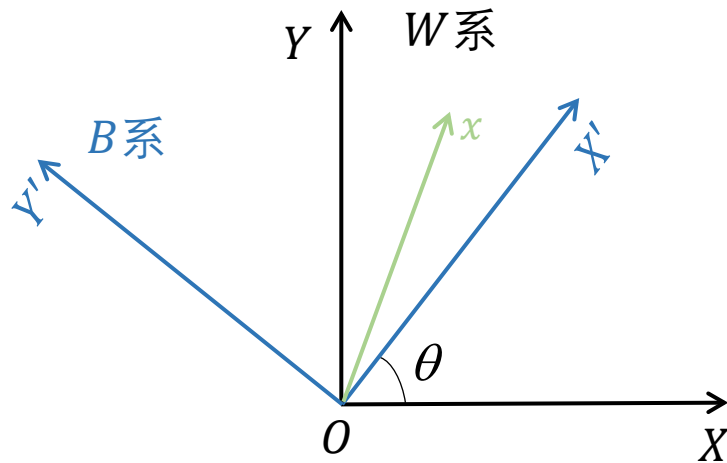
- 设向量 x 在 B 系中的坐标 x_B 为 (a, b)

$$x_B = \begin{bmatrix} a \\ b \end{bmatrix} = a \cdot OX' + b \cdot OY'$$

- 如果知道 B 系基向量在 W 系中的表达式，即可知道向量 x 在 W 系中的坐标 x_W 。
- 根据旋转可得， OX' 和 OY' 在 W 系中的坐标：

$$OX' = \begin{bmatrix} \cos \theta \\ \sin \theta \end{bmatrix}$$

$$OY' = \begin{bmatrix} -\sin \theta \\ \cos \theta \end{bmatrix}$$



B 系由 W 系逆时针旋转得到



数学基础之位姿表示和转换矩阵

被动旋转(坐标系旋转)

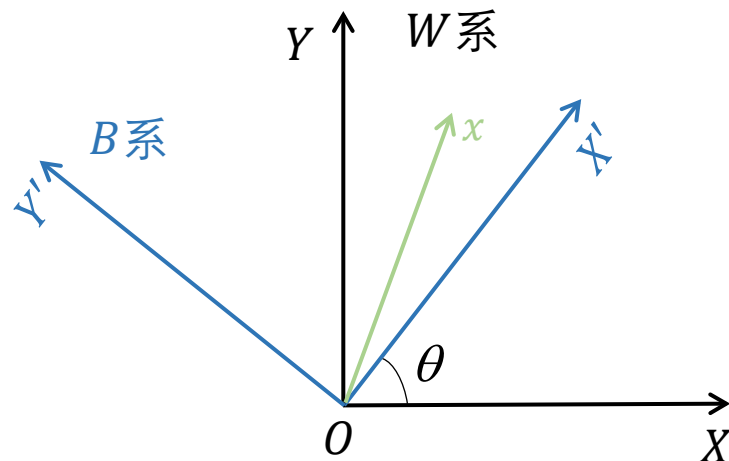
- 向量 x 在 W 系中的表达式:

$$x_W = a \begin{bmatrix} \cos \theta \\ \sin \theta \end{bmatrix} + b \begin{bmatrix} -\sin \theta \\ \cos \theta \end{bmatrix}$$

写成矩阵形式:

$$x_W = \begin{bmatrix} \cos \theta & -\sin \theta \\ \sin \theta & \cos \theta \end{bmatrix} \begin{bmatrix} a \\ b \end{bmatrix} = \begin{bmatrix} \cos \theta & -\sin \theta \\ \sin \theta & \cos \theta \end{bmatrix} x_B$$

$$x_W = R_B^W x_B \implies R_B^W = \begin{bmatrix} \cos \theta & -\sin \theta \\ \sin \theta & \cos \theta \end{bmatrix}$$



B系由W系逆时针旋转得到



数学基础之位姿表示和转换矩阵

转换矩阵

- 假设 B 系在 W 系中的位姿为：

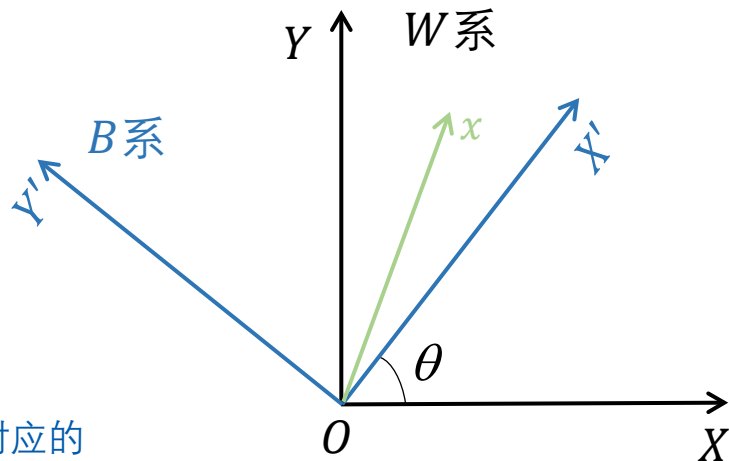
$$p_B = (x, y, \theta) = (t_B, \theta)$$

- 已知 B 系中的某个量，在 W 系中的表达：

$$x_W = R_B^W x_B + t_B = \begin{bmatrix} R_B^W & t_B \\ 0 & 1 \end{bmatrix} \begin{bmatrix} x_B \\ 1 \end{bmatrix}$$

$$T_B^W = \begin{bmatrix} R_B^W & t_B \\ 0 & 1 \end{bmatrix}$$

可以认为位姿跟转换矩阵是对应的



B 系由 W 系逆时针旋转得到



数学基础之位姿表示和转换矩阵

位姿表示

- 坐标系 B 在坐标系 W 中的坐标:

$$(x, y, \theta)$$

- 坐标系 B 到坐标 W 的转换矩阵:

$$T_B^W = \begin{bmatrix} \cos \theta & -\sin \theta & x \\ \sin \theta & \cos \theta & y \\ 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} R & t \\ 0 & 1 \end{bmatrix}$$

- 坐标系 W 到坐标系 B 的转换矩阵:

$$T_W^B = \begin{bmatrix} R^T & -R^T t \\ 0 & 1 \end{bmatrix} = \begin{bmatrix} R^{-1} & -R^{-1} t \\ 0 & 1 \end{bmatrix}$$

$$(x, y, \theta) = T2V(T_A^B) = (T_A^B(0,2), T_A^B(1,2), a \tan 2(T_A^B(1,0), T_A^B(0,0)))$$

$T2V$: 将矩阵转换成向量

- 机器人 A 在坐标系 W 的坐标:

$$(x_A, y_A, \theta_A), \begin{bmatrix} \cos \theta_A & -\sin \theta_A & x_A \\ \sin \theta_A & \cos \theta_A & y_A \\ 0 & 0 & 1 \end{bmatrix}$$

- 机器人 B 在坐标系 W 的坐标:

$$(x_B, y_B, \theta_B), \begin{bmatrix} \cos \theta_B & -\sin \theta_B & x_B \\ \sin \theta_B & \cos \theta_B & y_B \\ 0 & 0 & 1 \end{bmatrix}$$

- 机器人 A 在机器人 B 中的坐标:

$$T_A^B = T_W^B T_A^W = T_B^{W^{-1}} T_A^W$$



课程大纲

1. 激光SLAM

- 1.1 激光SLAM的发展历史
- 1.2 激光SLAM的流程

2. 传感器数据处理I：里程计运动模型及标定

- 2.1 里程计模型--简化的线性模型
- 2.2 里程计模型--实际模型
- 2.3 里程计标定原理

3. 传感器数据处理II：激光雷达数学模型和运动畸变去除

- 3.1 激光雷达数学模型
- 3.2 运动畸变概念及影响
- 3.3 基于纯激光雷达的运动畸变去除
- 3.4 基于里程计辅助的运动畸变去除

4. 激光SLAM的前端配准方法一

- 4.1 ICP匹配方法
- 4.2 PL-ICP匹配方法
- 4.3 NICP匹配方法
- 4.4 IMLS-ICP匹配方法

5. 激光SLAM的前端配准方法二

- 5.1 爬山法(拟梯度法)
- 5.2 基于高斯牛顿的优化方法
- 5.3 NDT方法
- 5.4 相关方法和分支定界方法



课程大纲

6. 基于图优化的激光SLAM方法

- 6.1 Pose Graph的概念
- 6.2 回环检测方法
- 6.3 非线性最小二乘原理
- 6.4 非线性最小二乘求解SLAM
- 6.5 经典开源算法(cartographer)源码讲解

7. 基于已知定位的建图

- 7.1 地图分类
- 7.2 基于占用概率的栅格地图构建
- 7.3 基于TSDF的栅格地图构建

8. life-long Mapping

- 8.1 life-long Mapping的概念以及应用
- 8.2 最小信息量激光帧的选取
- 8.3 Pose-Graph的精确边缘化
- 8.4 Pose-Graph的近似边缘化

9. 3D激光SLAM的介绍

- 9.1 3D激光SLAM的介绍
- 9.2 LOAM算法原理
- 9.3 ALOAM代码讲解
- 9.4 主流开源激光SLAM代码介绍



课程设置

作业

- 基于ROS系统，C++编程
- 实现关键函数
- 每一小节有对应的课后作业，课程完结有一个把所有的模块连接起来的大作业



作业

详细见第一节作业说明文档



结语

感谢各位聆听!

Thanks for Listening

