

第1季

第4章： 机器人传感器



主讲人：张虎

(小虎哥哥爱学习)

- 先导课
- 第1季：快速梳理知识要点与学习方法 ✓
- 第2季：详细推导数学公式与代码解析
- 第3季：代码实操以及真实机器人调试
- 答疑课

----- (永久免费 • 系列课程 • 长期更新) -----

本书内容安排

一、编程基础篇

第1章：ROS入门必备知识

第2章：C++编程范式

第3章：OpenCV图像处理

二、硬件基础篇

第4章：机器人传感器

第5章：机器人主机

第6章：机器人底盘

三、SLAM篇

第7章：SLAM中的数学基础

第8章：激光SLAM系统

第9章：视觉SLAM系统

第10章：其他SLAM系统

四、自主导航篇

第11章：自主导航中的数学基础

第12章：典型自主导航系统

第13章：机器人SLAM导航综合实战

搞机器人，有必要学习硬件吗？



我不关心程序在硬件中如何运行，
我只关心代码会不会出BUG!!!

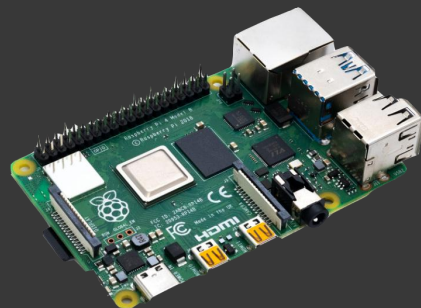
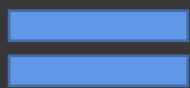
搞机器人，有必要学习硬件吗？



搞机器人，有必要学习硬件吗？



机器人



硬件电路



小纸人



机器人就是一台 具有丰富输入输出接口的计算机

鸡哥，
你有啥
才艺？



唱、跳、rap、篮球

Music !

```
def __init__(self, datadir, ndims):
    idfile = os.path.join(datadir, "id.txt")
    self.names = [x.strip() for x in str.split(open(idfile).read())]
    self.name2index = dict(zip(self.names, range(len(self.names))))
    self.ndims = ndims
    self.featurefile = os.path.join(datadir, "features.txt")
    print "[BigFile] %d features, %d dimensions" % (len(self.names), self.ndims)
    print "binary: %s" % self.featurefile
    print "txt: %s" % idfile

def read(self, requested, isname=True):
    if isname:
        index_name_array = [(self.name2index[x], x) for x in requested]
    else:
        assert(min(requested) >= 0)
        assert(max(requested) < len(self.names))
        index_name_array = [(self.name2index[x], x) for x in requested]
```

机器人就是一台 具有丰富输入输出接口的计算机



- 传感器：提供数据指令的输入输出
- 主机：运行数据处理的程序

机器人传感器 对编程的意义？

- 正确使用数据
- 软硬件深度优化编程

内容概要

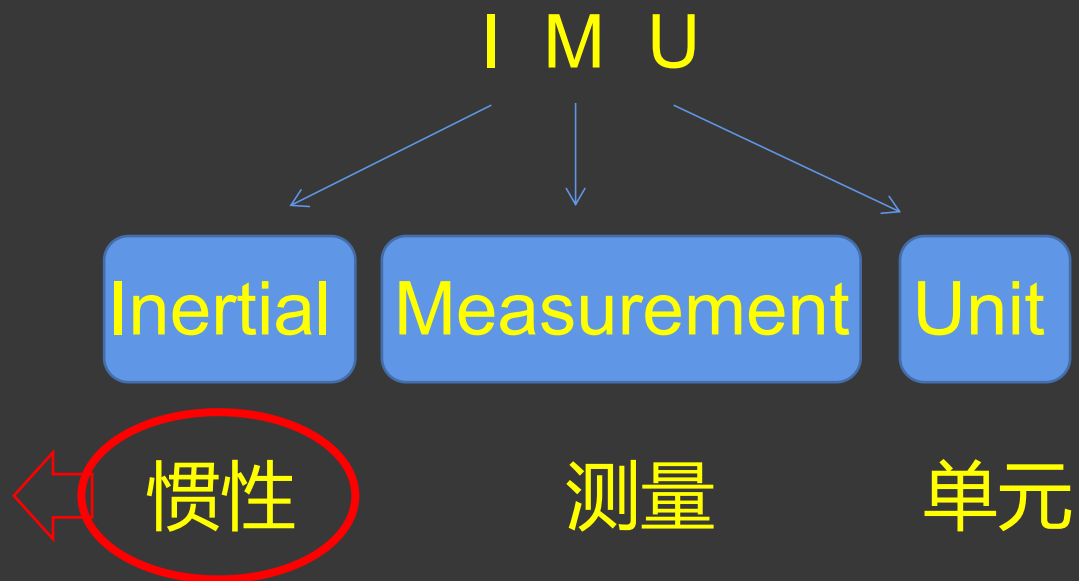
4.1 惯性测量单元

4.2 激光雷达

4.3 相机

4.4 带编码器的减速电机

4.1 惯性测量单元



4.1 惯性测量单元

机器人中的IMU有什么用？

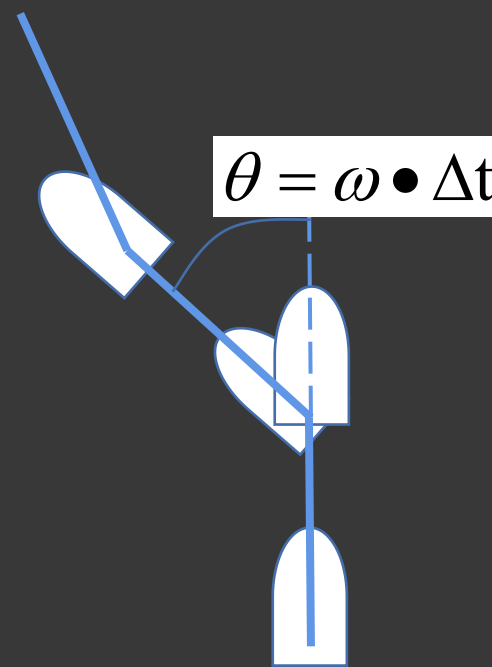
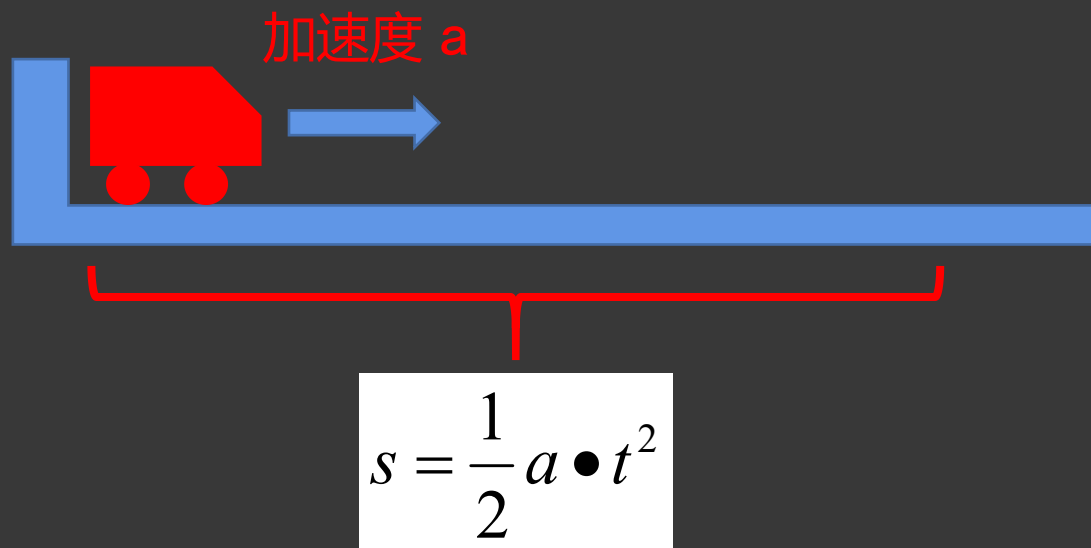
- ① 应用惯性测量
- ② 多传感器融合

4.1 惯性测量单元

机器人中的IMU有什么用？

① 应用惯性测量

② 多传感器融合

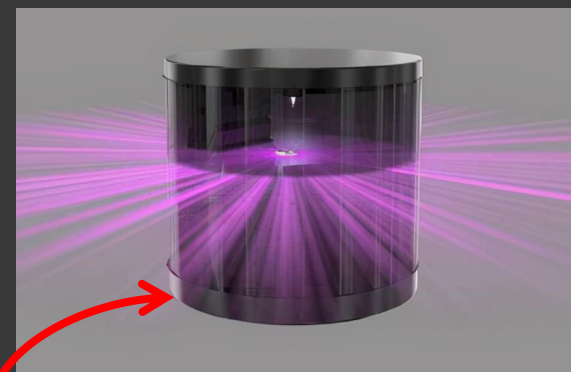
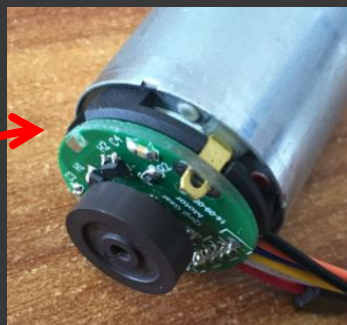
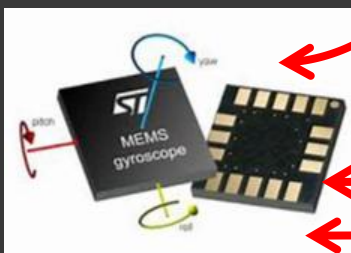


4.1 惯性测量单元

机器人中的IMU有什么用？

① 应用惯性测量

② 多传感器融合



为什么能融合？

- * IMU数据与其他传感器数据互补
- * IMU数据频率很高

4.1 惯性测量单元

- 工作原理

■ 原始数据采集

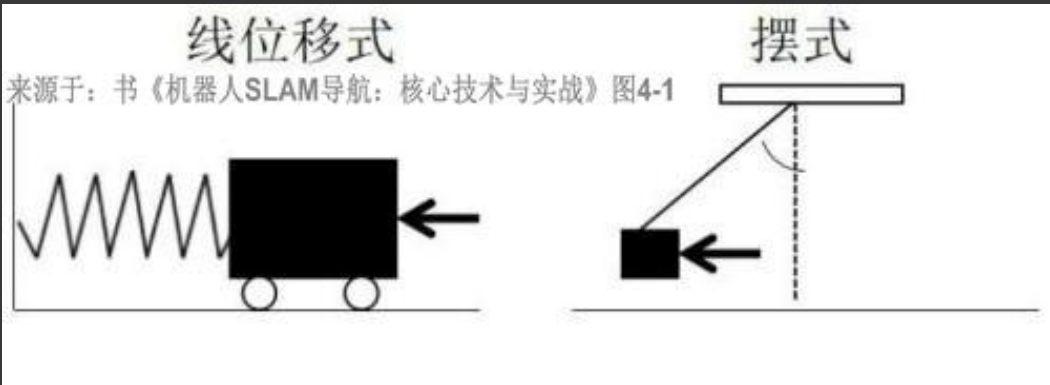
■ 参数标定

■ 数据滤波

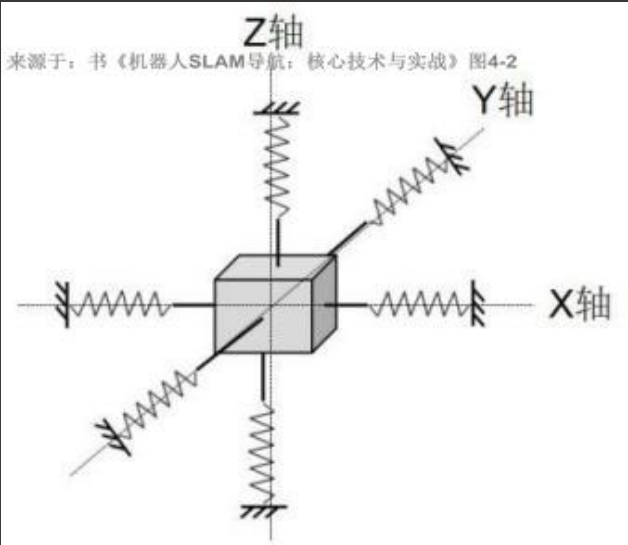
■ 姿态融合
- 加速度测量

■ 角速度测量

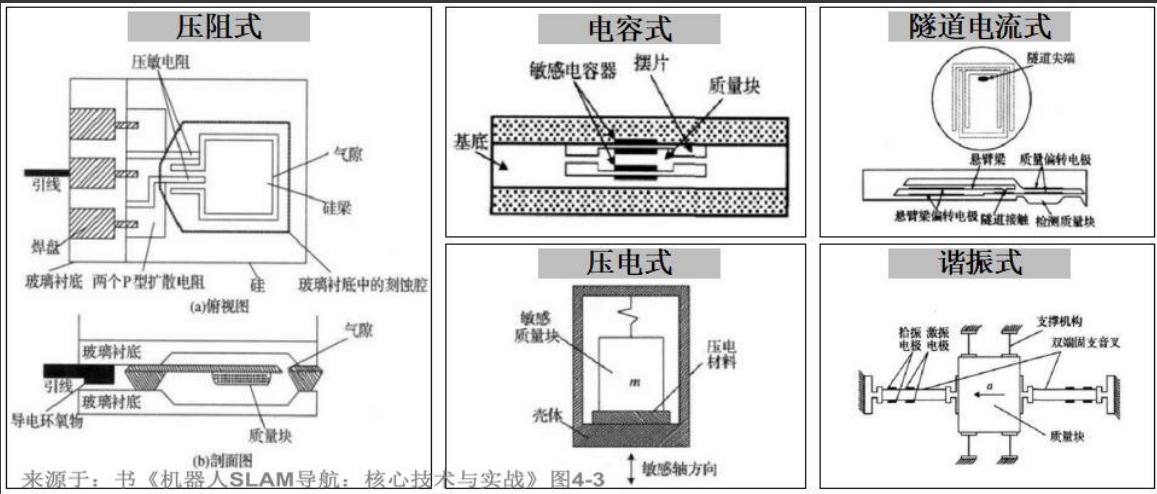
■ 磁力测量



力学原理



三轴模型



MEMS微机电模型

4.1 惯性测量单元

- 工作原理
 - 原始数据采集
 - 参数标定
 - 数据滤波
 - 姿态融合
-
- 加速度测量
 - 角速度测量
 - 磁力测量



陀螺的定轴性



4.1 惯性测量单元

- 工作原理
 - 原始数据采集
 - 参数标定
 - 数据滤波
 - 姿态融合
- ➔
- 加速度测量
 - 角速度测量
 - 磁力测量



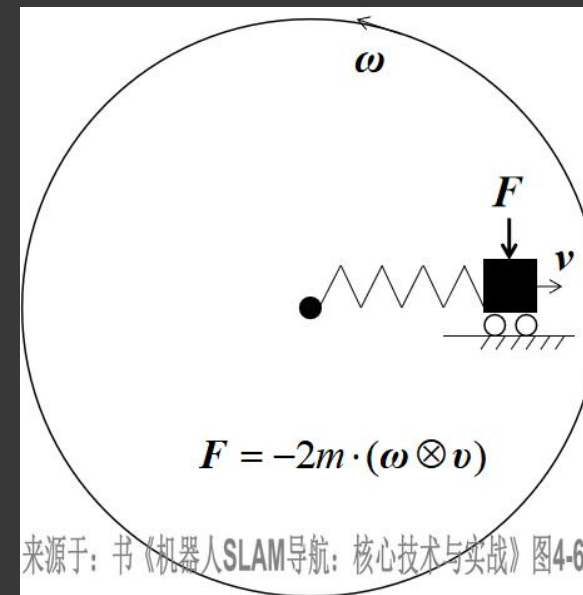
来源于：书《机器人SLAM导航：核心技术与实战》图4-4

机械陀螺



来源于：书《机器人SLAM导航：核心技术与实战》图4-5

激光陀螺仪

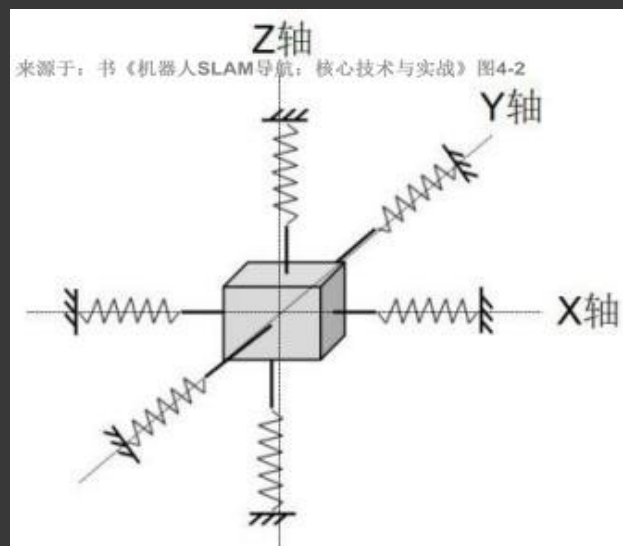


来源于：书《机器人SLAM导航：核心技术与实战》图4-6

MEMS微机电陀螺仪

4.1 惯性测量单元

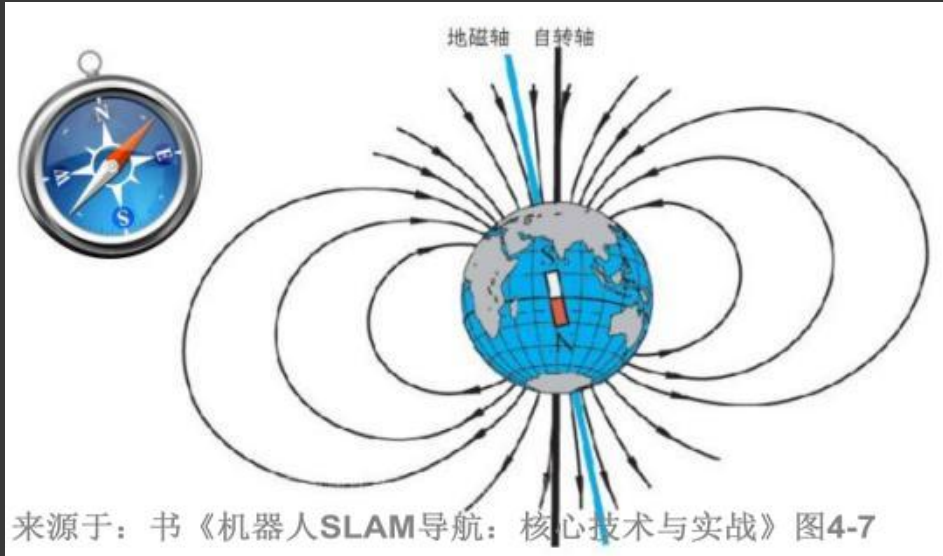
- 工作原理
 - 原始数据采集
 - 参数标定
 - 数据滤波
 - 姿态融合
- 加速度测量
 - 角速度测量
 - 磁力测量



三轴模型

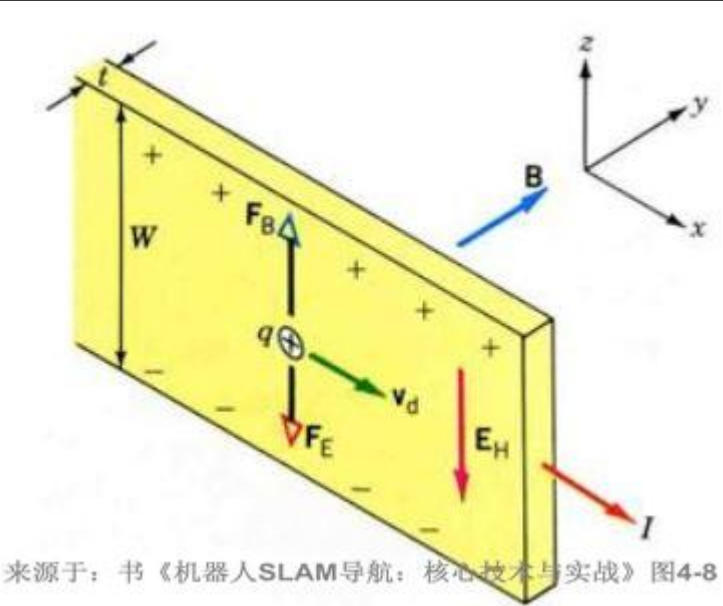
4.1 惯性测量单元

- 工作原理
- 原始数据采集
- 参数标定
- 数据滤波
- 姿态融合
- 加速度测量
- 角速度测量
- 磁力测量



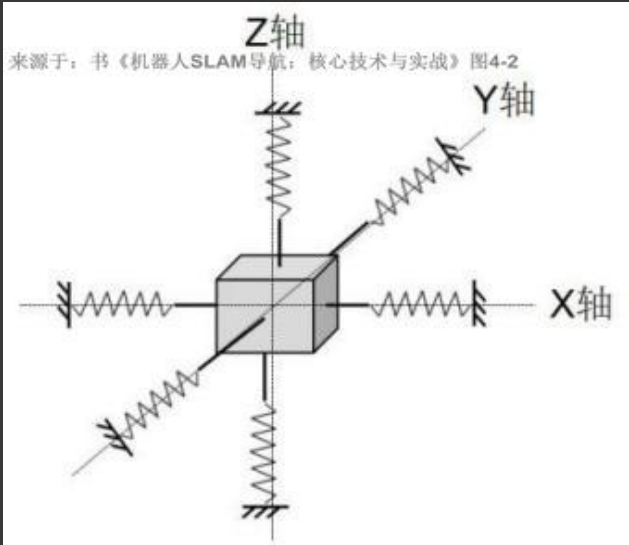
指南针

来源于：书《机器人SLAM导航：核心技术与实战》图4-7



来源于：书《机器人SLAM导航：核心技术与实战》图4-8

霍尔效应电子磁力计



来源于：书《机器人SLAM导航：核心技术与实战》图4-2

三轴模型

4.1 惯性测量单元

- 工作原理

- 原始数据采集



- 参数标定

- 数据滤波

- 姿态融合

① 什么是九轴IMU?

三轴加速度、三轴角速度、三轴磁力、（气压）

② IMU有哪些性能参数?

量程、非线性度、零偏、轴间灵敏度、噪声密度、温偏

③ 有哪些常见的IMU芯片?

MPU9250、ADIS16405

4.1 惯性测量单元

- 工作原理

■ 原始数据采集

■ 参数标定

■ 数据滤波

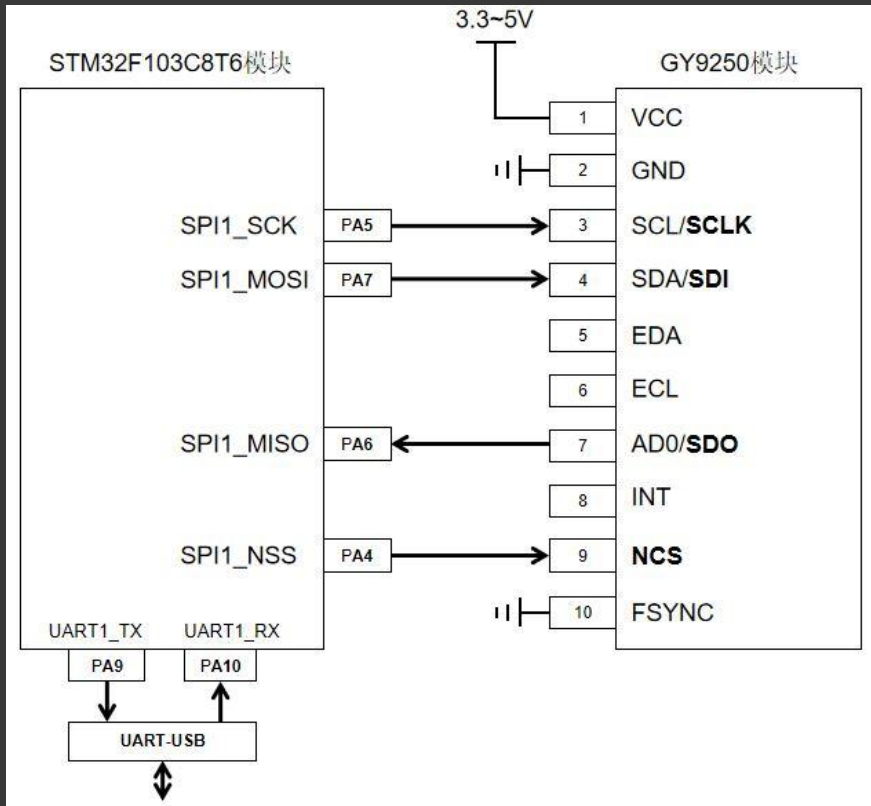
■ 姿态融合

➡

■ 硬件电路搭建

■ 固件驱动开发

■ 上位机ROS驱动程序



4.1 惯性测量单元

- 工作原理

■ 原始数据采集

■ 参数标定

■ 数据滤波

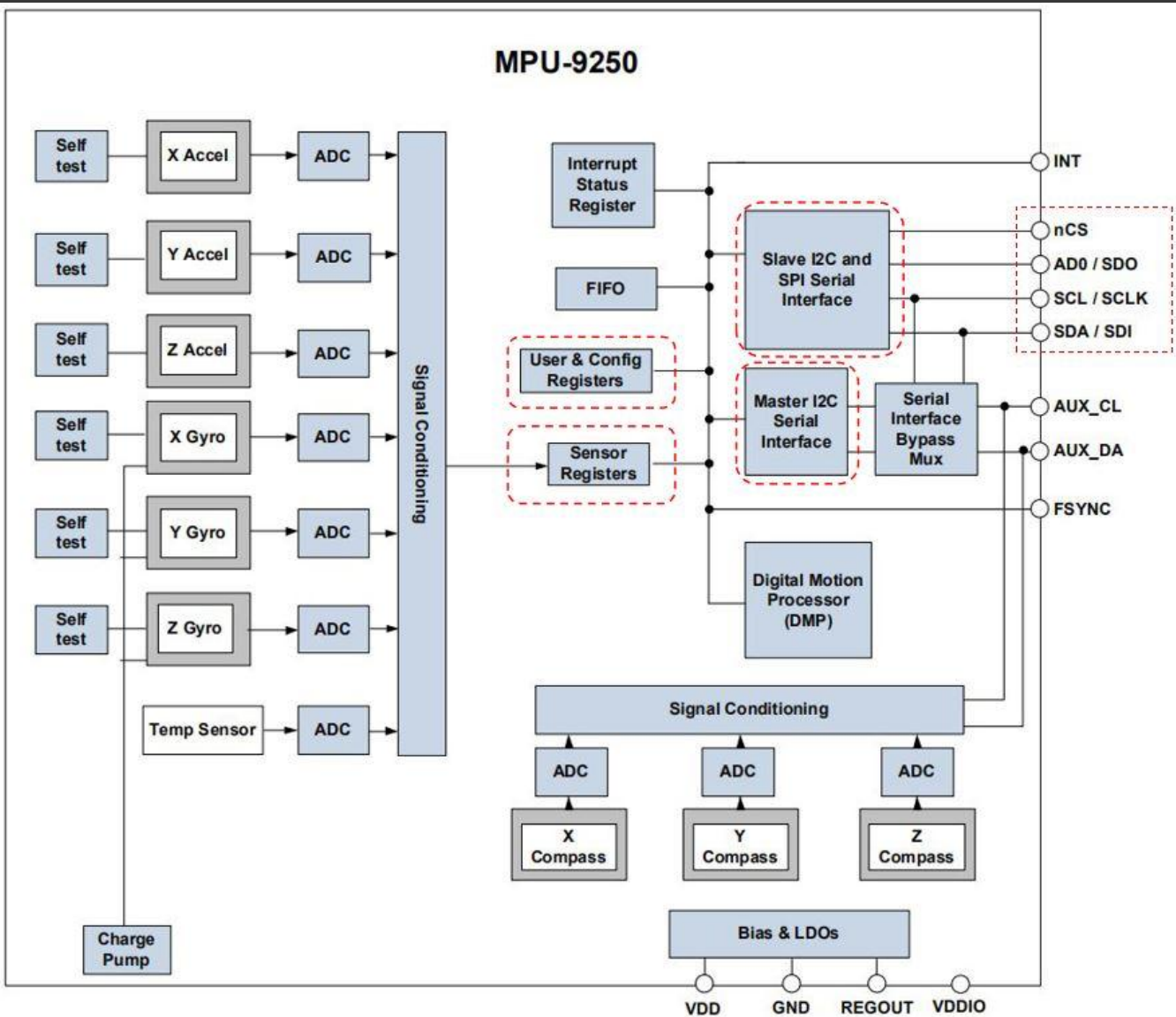
■ 姿态融合
- 硬件电路搭建

■ 固件驱动开发

■ 上位机ROS驱动程序

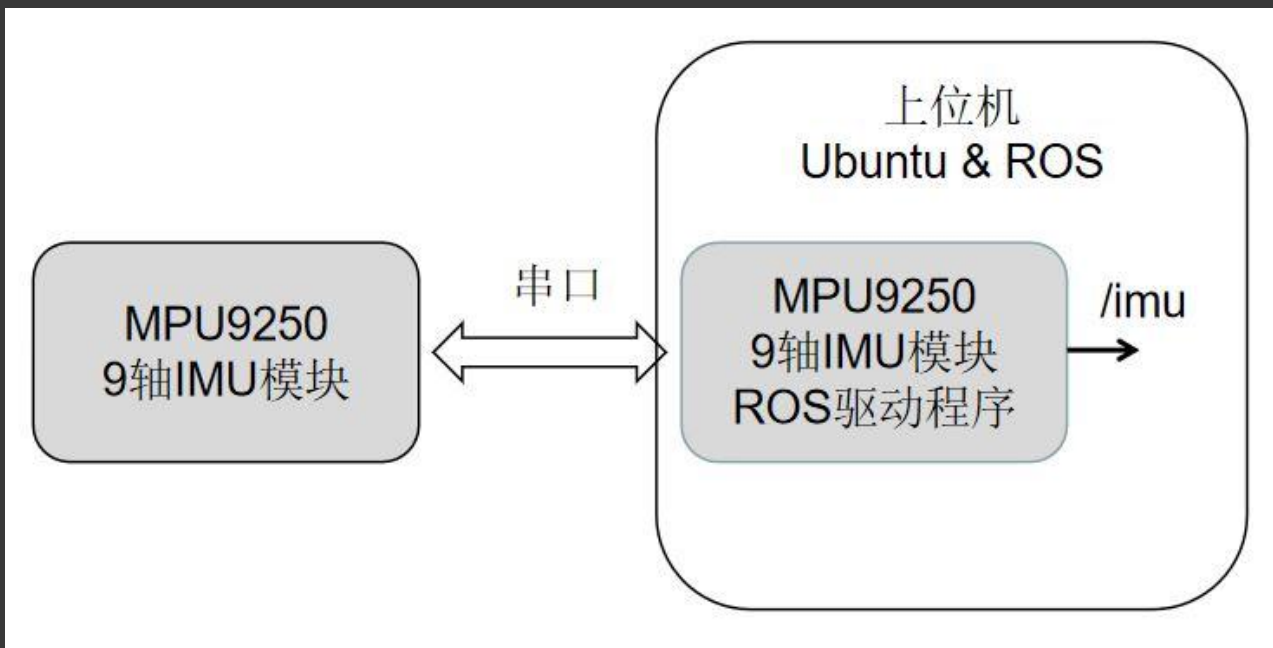
```
mpu9250 reg interface
{
    void write_mpu9250_i2c_mag_reg(u8 reg,u8 value)
    u8 read_mpu9250_i2c_mag_reg(u8 reg)
    u8 write_mpu9250_reg(u8 reg,u8 value)
    u8 read_mpu9250_reg(u8 reg)
}

u8 SPI1_ReadWriteByte(u8 TxData)
```



4.1 惯性测量单元

- 工作原理
 - 原始数据采集
 - 参数标定
 - 数据滤波
 - 姿态融合
-
- 硬件电路搭建
 - 固件驱动开发
 - 上位机ROS驱动程序



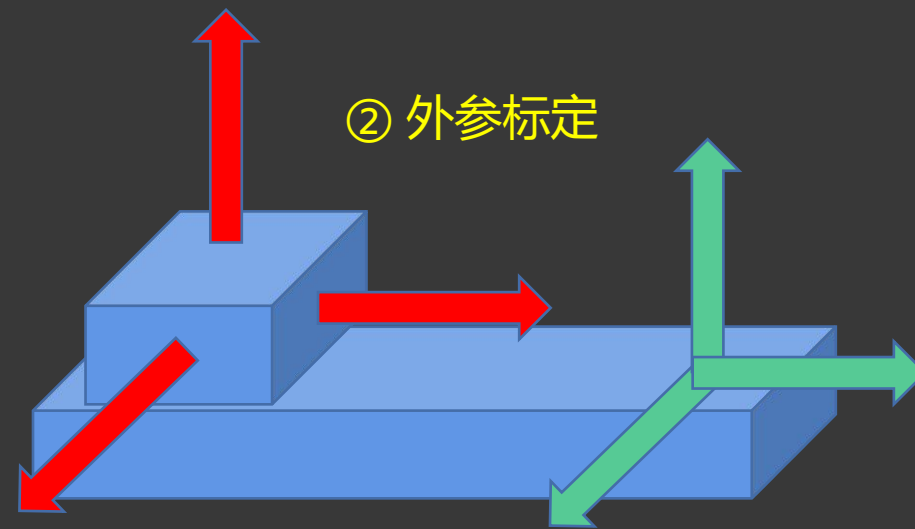
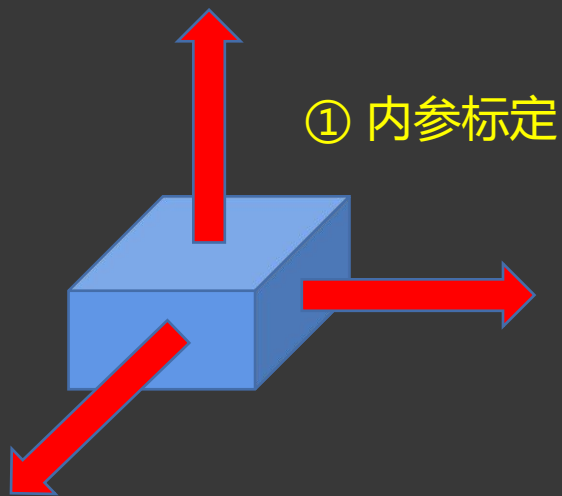
- IMU硬件模块
- 串口通信
- 上位机ROS驱动程序
- 数据发布话题/imu

4.1 惯性测量单元

- 工作原理
- 原始数据采集
- 参数标定
- 数据滤波
- 姿态融合

为什么要标定？

如果IMU测量数据本身就存在很大的误差，即输入到系统的就是错误信息，上层应用系统的算法做得再好也会输出错误结果。



4.1 惯性测量单元

- 工作原理

- 原始数据采集

- 参数标定

- 数据滤波

- 姿态融合

- 良率检测

- 内参标定过程

- 标定模型改进

- 重复上电对零偏的影响

- 温度对零偏的影响

- 振动对零偏的影响

- 高冲击容忍度

- 非线性度

将一些建模困难的重要误差项作为指标，对指标存在较大误差的IMU模块进行筛选

4.1 惯性测量单元

- 工作原理

■ 原始数据采集

■ 参数标定

■ 数据滤波

■ 姿态融合
- 良率检测

■ 内参标定过程

■ 标定模型改进

$I^{cal} = T \bullet S \bullet (I - B)$, 写成矩阵的形式:

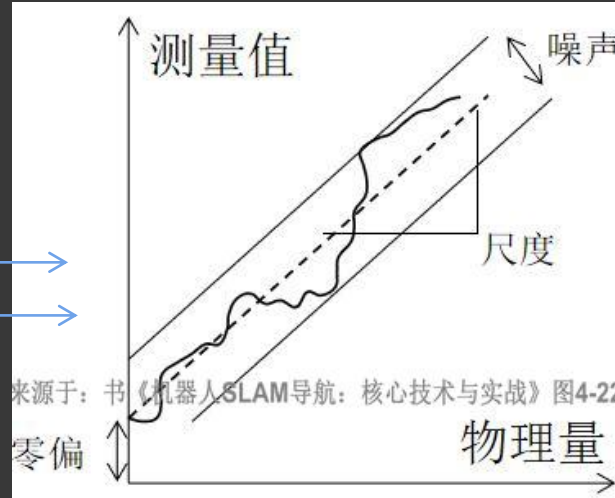
$$\begin{bmatrix} I_x^{cal} \\ I_y^{cal} \\ I_z^{cal} \end{bmatrix} = \begin{bmatrix} 1 & T_{xy} & T_{xz} \\ T_{yx} & 1 & T_{yz} \\ T_{zx} & T_{zy} & 1 \end{bmatrix} \bullet \begin{bmatrix} S_x & 0 & 0 \\ 0 & S_y & 0 \\ 0 & 0 & S_z \end{bmatrix} \bullet \left(\begin{bmatrix} I_x \\ I_y \\ I_z \end{bmatrix} - \begin{bmatrix} B_x \\ B_y \\ B_z \end{bmatrix} \right)$$

线性误差模型

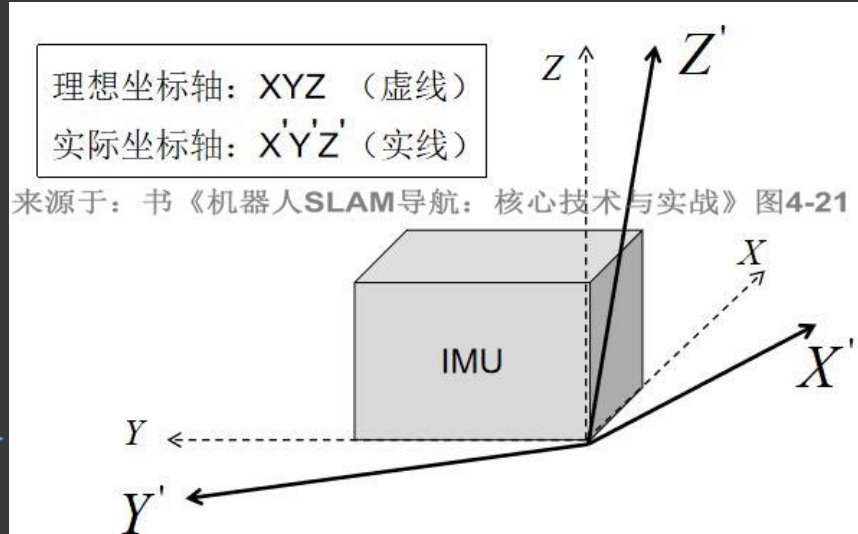
零偏

尺度偏差

轴偏差



来源于：书《机器人SLAM导航：核心技术与实战》图4-22



来源于：书《机器人SLAM导航：核心技术与实战》图4-21

4.1 惯性测量单元

- 工作原理

■ 原始数据采集

■ 参数标定

■ 数据滤波

■ 姿态融合

■ 良率检测

■ 内参标定过程

■ 标定模型改进

$I^{cal} = T \bullet S \bullet (I - B)$, 写成矩阵的形式:

$$\begin{bmatrix} I_x^{cal} \\ I_y^{cal} \\ I_z^{cal} \end{bmatrix} = \begin{bmatrix} 1 & T_{xy} & T_{xz} \\ T_{yx} & 1 & T_{yz} \\ T_{zx} & T_{zy} & 1 \end{bmatrix} \bullet \begin{bmatrix} S_x & 0 & 0 \\ 0 & S_y & 0 \\ 0 & 0 & S_z \end{bmatrix} \bullet \left(\begin{bmatrix} I_x \\ I_y \\ I_z \end{bmatrix} - \begin{bmatrix} B_x \\ B_y \\ B_z \end{bmatrix} \right)$$

$A^{cal} = T^{acc} \bullet S^{acc} \bullet (A - B^{acc})$, 写成矩阵的形式:

$$\begin{bmatrix} A_x^{cal} \\ A_y^{cal} \\ A_z^{cal} \end{bmatrix} = \begin{bmatrix} 1 & T_{xy}^{acc} & T_{xz}^{acc} \\ T_{yx}^{acc} & 1 & T_{yz}^{acc} \\ T_{zx}^{acc} & T_{zy}^{acc} & 1 \end{bmatrix} \bullet \begin{bmatrix} S_x^{acc} & 0 & 0 \\ 0 & S_y^{acc} & 0 \\ 0 & 0 & S_z^{acc} \end{bmatrix} \bullet \left(\begin{bmatrix} A_x \\ A_y \\ A_z \end{bmatrix} - \begin{bmatrix} B_x^{acc} \\ B_y^{acc} \\ B_z^{acc} \end{bmatrix} \right)$$

$$\sqrt{[\text{var}(A_x)]^2 + [\text{var}(A_y)]^2 + [\text{var}(A_z)]^2} < P$$

$$\min \sum_{n=1}^N \left(\|g\|^2 - \|T^{acc} \bullet S^{acc} \bullet (A - B^{acc})\|^2 \right)^2$$

加速度计标定

$W^{cal} = T^{gyro} \bullet S^{gyro} \bullet (W - B^{gyro})$, 写成矩阵的形式:

$$\begin{bmatrix} W_x^{cal} \\ W_y^{cal} \\ W_z^{cal} \end{bmatrix} = \begin{bmatrix} 1 & T_{xy}^{gyro} & T_{xz}^{gyro} \\ T_{yx}^{gyro} & 1 & T_{yz}^{gyro} \\ T_{zx}^{gyro} & T_{zy}^{gyro} & 1 \end{bmatrix} \bullet \begin{bmatrix} S_x^{gyro} & 0 & 0 \\ 0 & S_y^{gyro} & 0 \\ 0 & 0 & S_z^{gyro} \end{bmatrix} \bullet \left(\begin{bmatrix} W_x \\ W_y \\ W_z \end{bmatrix} - \begin{bmatrix} B_x^{gyro} \\ B_y^{gyro} \\ B_z^{gyro} \end{bmatrix} \right)$$

$$u_{g,k} = \text{RK4n}(T^{gyro} \bullet S^{gyro} \bullet (W_i - B^{gyro}), u_{a,k-1}), \text{其中} i = 1, 2, 3, \dots$$

$$\min \sum_{n=2}^N \|u_{a,k} - u_{g,k}\|^2$$

陀螺仪标定

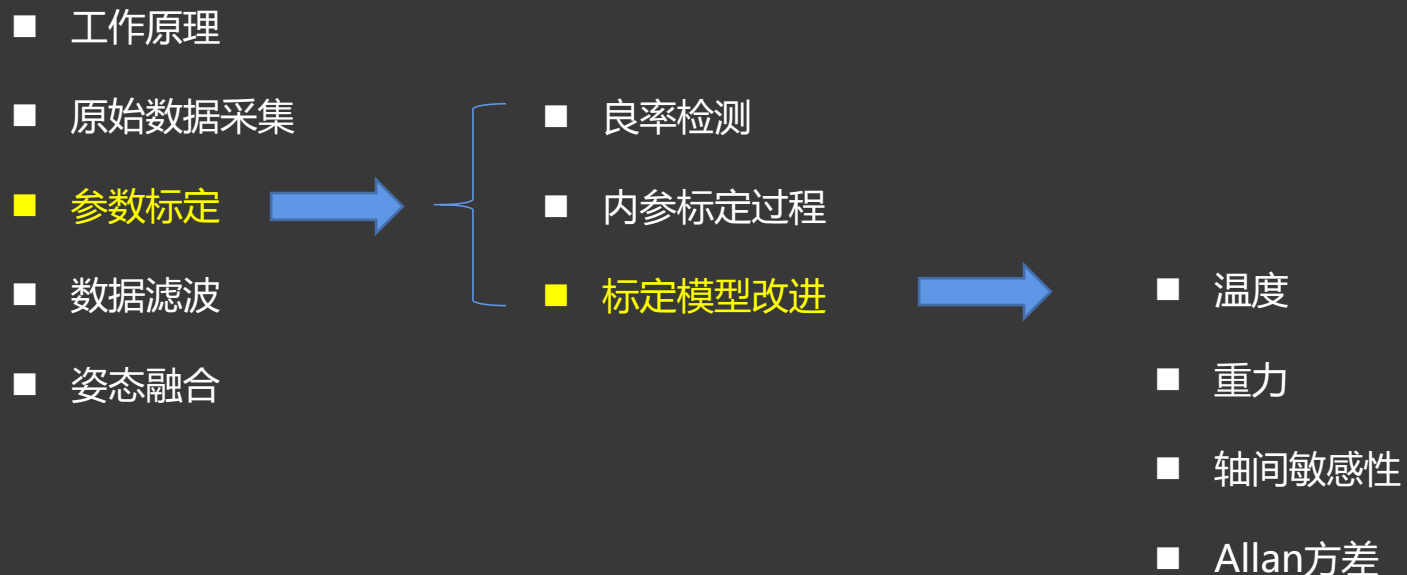
$M^{cal} = T^{m2a} \bullet (M - B^{mag})$, 写成矩阵的形式:

$$\begin{bmatrix} M_x^{cal} \\ M_y^{cal} \\ M_z^{cal} \end{bmatrix} = \begin{bmatrix} T_{11}^{m2a} & T_{12}^{m2a} & T_{13}^{m2a} \\ T_{21}^{m2a} & T_{22}^{m2a} & T_{23}^{m2a} \\ T_{31}^{m2a} & T_{32}^{m2a} & T_{33}^{m2a} \end{bmatrix} \bullet \left(\begin{bmatrix} M_x \\ M_y \\ M_z \end{bmatrix} - \begin{bmatrix} B_x^{mag} \\ B_y^{mag} \\ B_z^{mag} \end{bmatrix} \right)$$

$$\min \sum_{n=1}^N \left(A^{cal} \bullet (T^{m2a} \bullet (M - B^{mag})) - Q \right)^2$$

磁力计标定

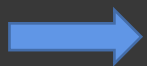
4.1 惯性测量单元



* 关于标定模型改进的细节，请参考课本相关内容

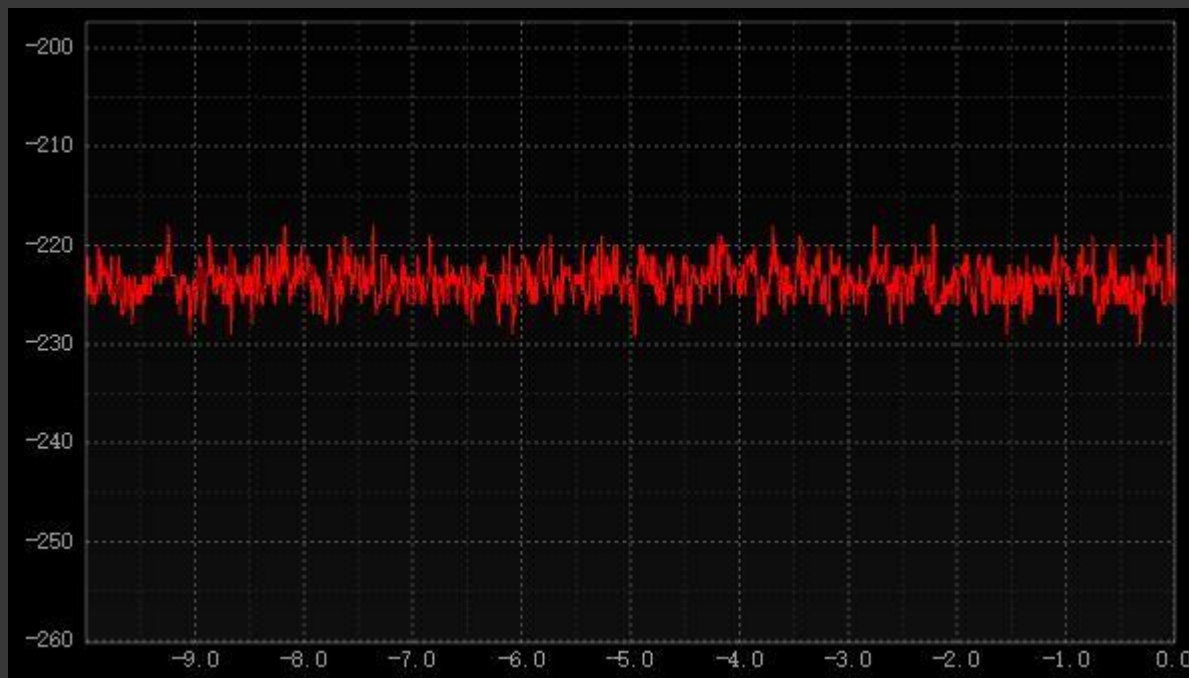
4.1 惯性测量单元

- 工作原理
- 原始数据采集
- 参数标定
- 数据滤波
- 姿态融合



经过标定后的IMU数据，依然存在抖动、毛刺之类的噪声，因此需要滤波降噪

为什么要滤波？



4.1 惯性测量单元

- 工作原理
 - 原始数据采集
 - 参数标定
 - 数据滤波
 - 姿态融合
- 
- 均值滤波
 - 滑动滤波
 - 滑动中值滤波
 - RC低通数字滤波
 - IIR数字滤波

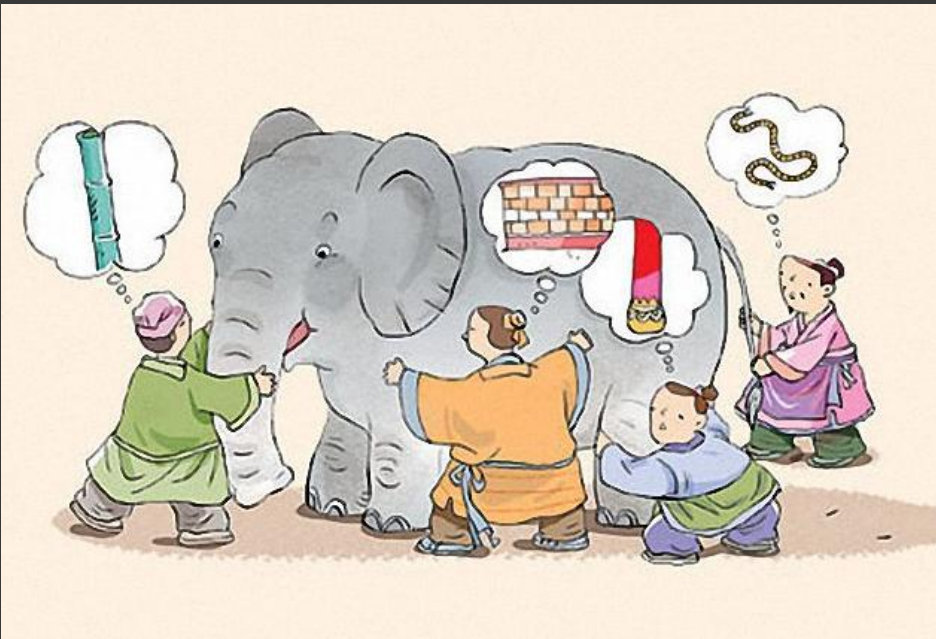
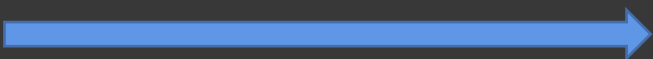
加速度 ---> 低通滤波

角速度 ---> 高通滤波

磁力 ---> 低通滤波

4.1 惯性测量单元

- 工作原理
- 原始数据采集
- 参数标定
- 数据滤波
- 姿态融合



名词	全称	解释
IMU	惯性测量单元	三轴加速度 三轴角速度
MARG	磁力、角速度、重力	三轴加速度 三轴角速度 三轴磁力
AHRS	航姿参考系统	三轴加速度 三轴角速度 三轴磁力 航姿
INS	组合导航系统	全球卫星导航 惯性导航 视觉导航

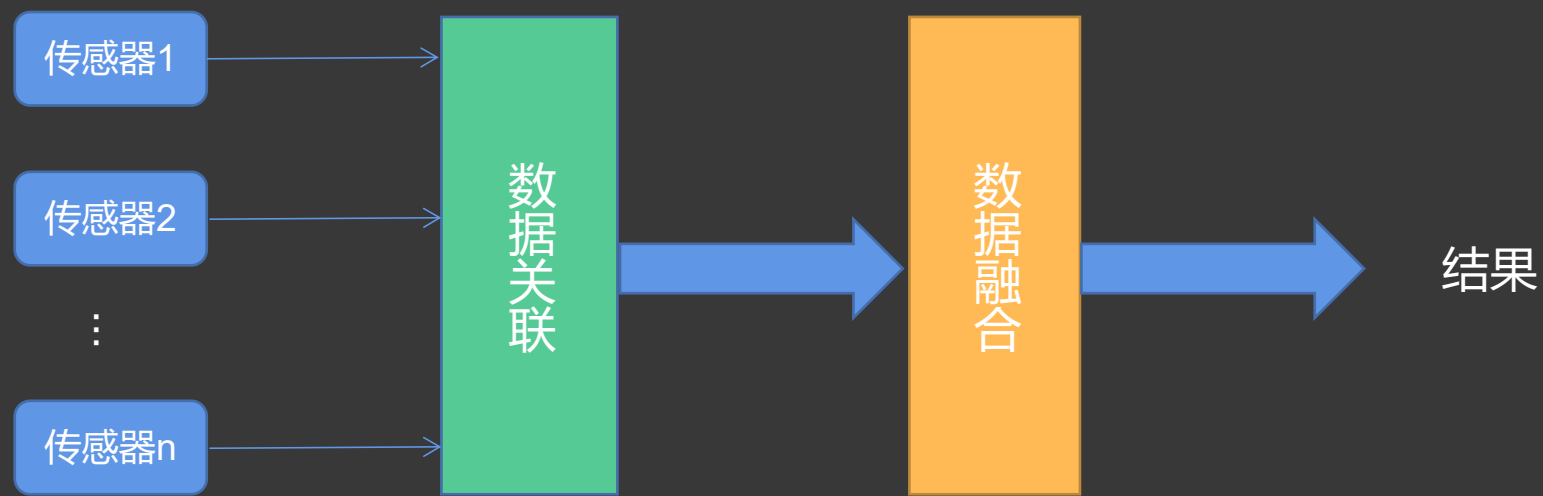
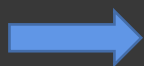
来源于：书《机器人SLAM导航：核心技术与实战》表4-7

* 多传感器融合的本质：

- 盲人摸象
- ①互补
- ②改进

4.1 惯性测量单元

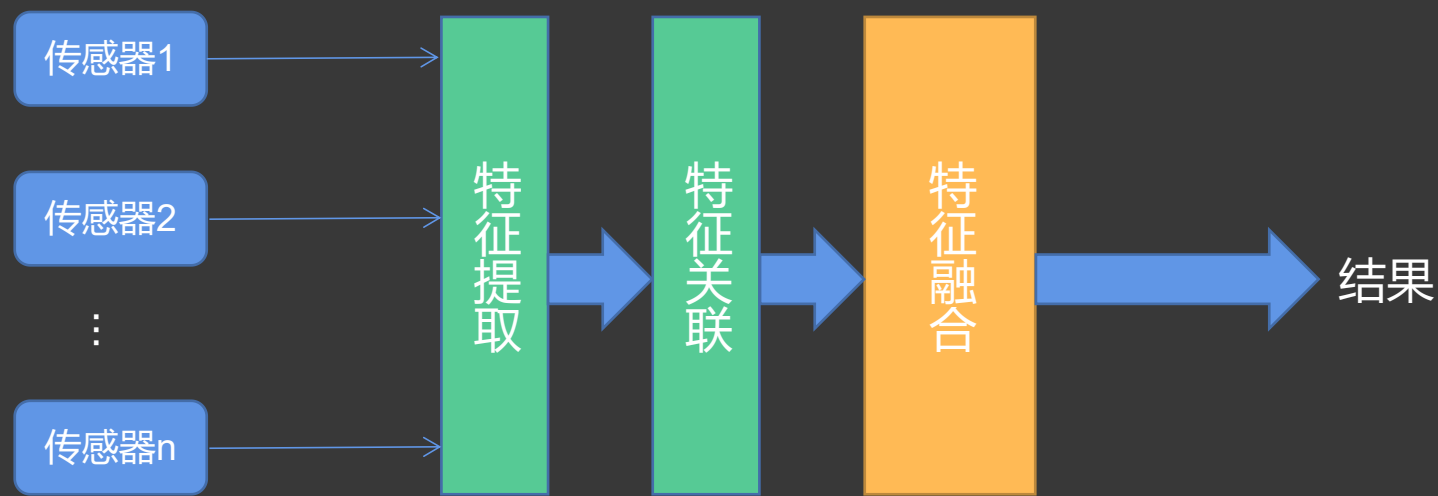
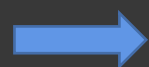
- 工作原理
- 原始数据采集
- 参数标定
- 数据滤波
- 姿态融合



初级融合

4.1 惯性测量单元

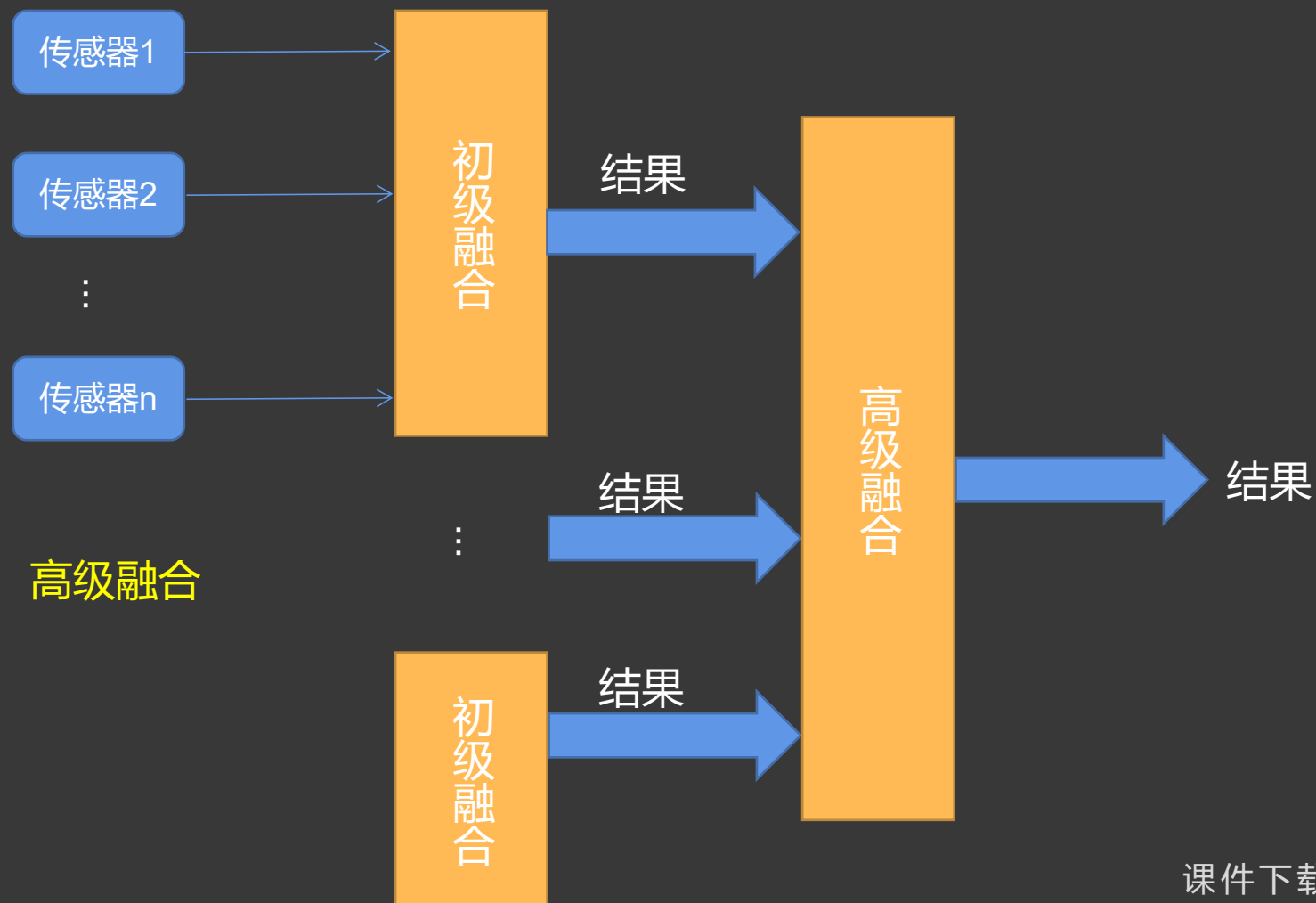
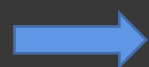
- 工作原理
- 原始数据采集
- 参数标定
- 数据滤波
- 姿态融合



次级融合

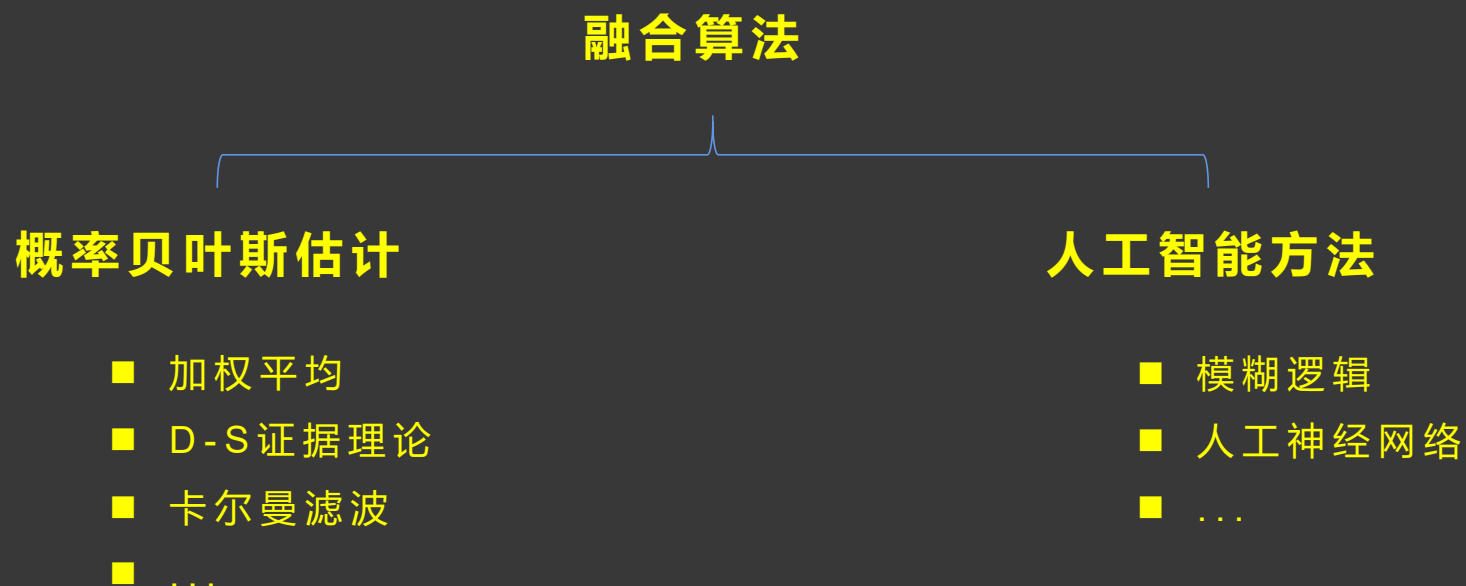
4.1 惯性测量单元

- 工作原理
- 原始数据采集
- 参数标定
- 数据滤波
- 姿态融合



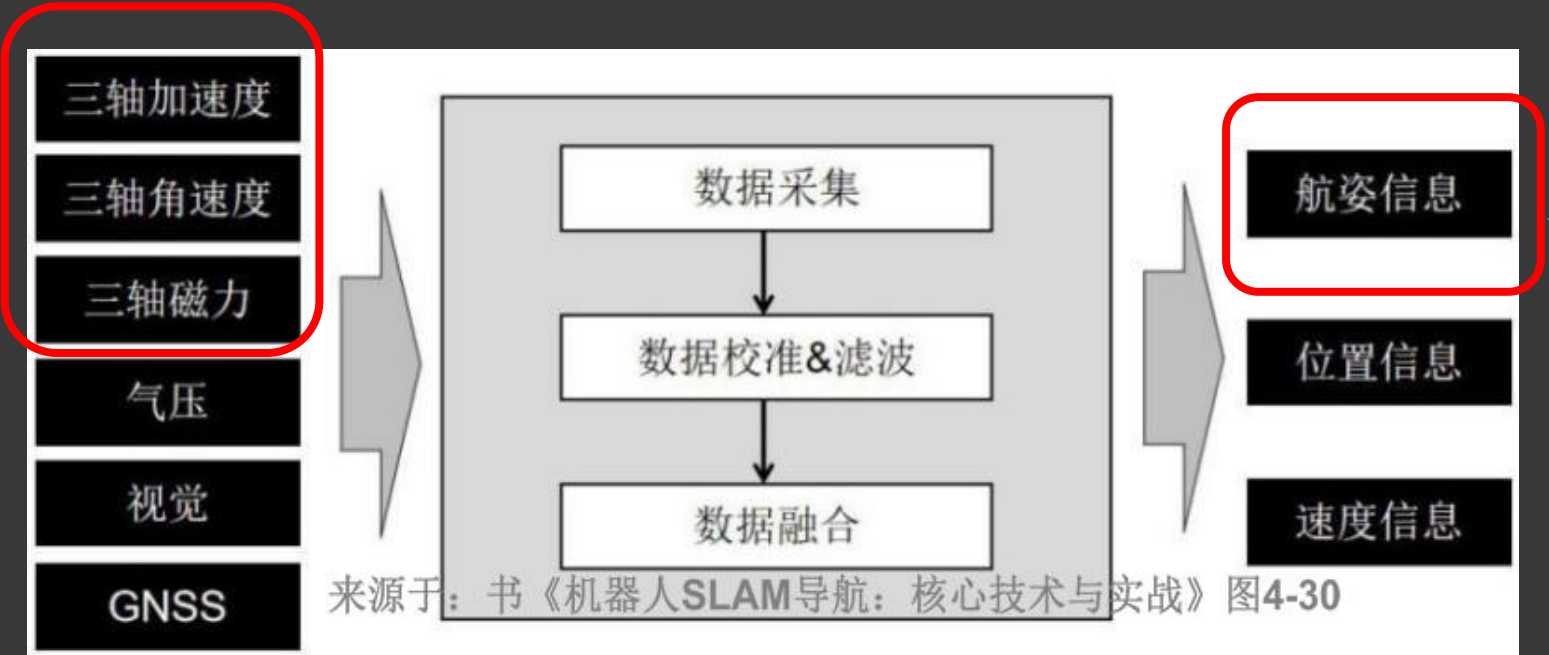
4.1 惯性测量单元

- 工作原理
- 原始数据采集
- 参数标定
- 数据滤波
- 姿态融合



4.1 惯性测量单元

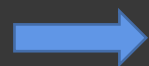
- 工作原理
- 原始数据采集
- 参数标定
- 数据滤波
- 姿态融合



典型的导航系统框图

4.1 惯性测量单元

- 工作原理
- 原始数据采集
- 参数标定
- 数据滤波
- 姿态融合



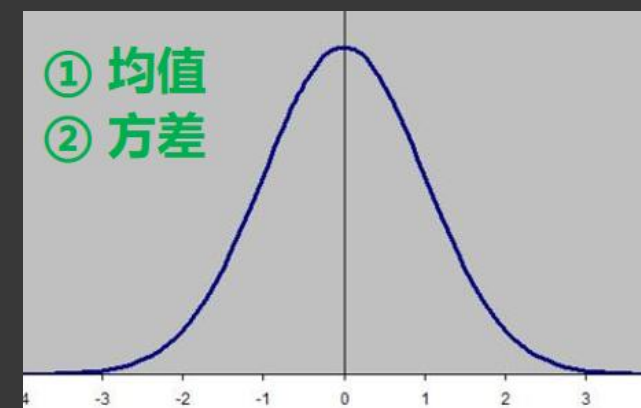
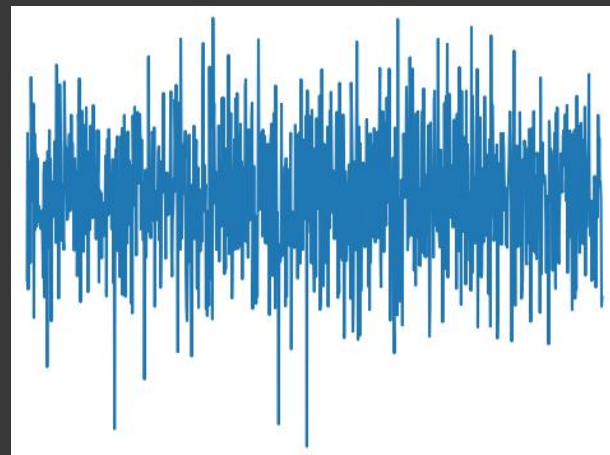
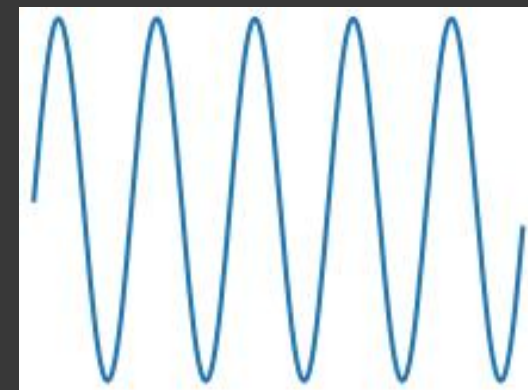
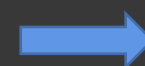
■ 经典滤波器：高通、低通、带通

(过滤特定频段的噪声)

■ 现代滤波器：统计学方法

(过滤白噪声)

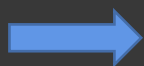
卡尔曼滤波



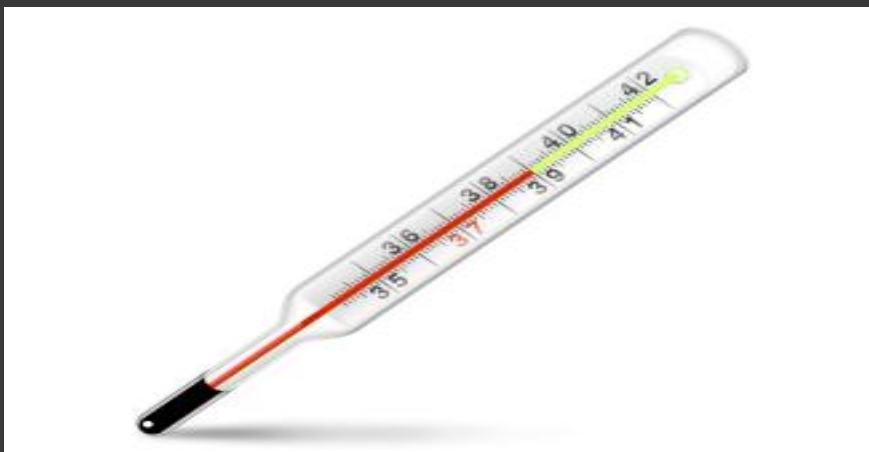
4.1 惯性测量单元

卡尔曼滤波

- 工作原理
- 原始数据采集
- 参数标定
- 数据滤波
- 姿态融合



举个用统计学方法对温度测量值估计的例子，用两个精度不同的温度计测量体温，并且假设两个测量误差都是服从零均值的正态分布。



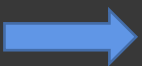
温度测量值： T_1



温度测量值： T_2

4.1 惯性测量单元

卡尔曼滤波

- 工作原理
- 原始数据采集
- 参数标定
- 数据滤波
- 姿态融合 

$$T_1 = T + \varepsilon_1$$

$$T_2 = T + \varepsilon_2$$

其中：

$$\varepsilon_1 \sim N(0, \sigma_1^2), \varepsilon_2 \sim N(0, \sigma_2^2), E[\varepsilon_1 \varepsilon_2] = 0$$

平均

VS

加权平均

$$\hat{T}_{\text{est1}} = \frac{T_1 + T_2}{2}$$

$$D[\hat{T}_{\text{est1}}] = D\left[\frac{T_1 + T_2}{2}\right] = \frac{D[T_1] + D[T_2]}{4} = \frac{\sigma_1^2 + \sigma_2^2}{4}$$

$$\hat{T}_{\text{est2}} = \alpha \cdot T_1 + (1 - \alpha) \cdot T_2$$

$$\begin{aligned} D[\hat{T}_{\text{est2}}] &= D[\alpha \cdot T_1 + (1 - \alpha) \cdot T_2] \\ &= \alpha^2 D[T_1] + (1 - \alpha)^2 D[T_2] \\ &= \alpha^2 \sigma_1^2 + (1 - \alpha)^2 \sigma_2^2 \end{aligned}$$

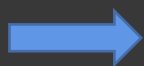
$$\begin{aligned} \min \{D[\hat{T}_{\text{est2}}]\} &= \alpha^2 \sigma_1^2 + (1 - \alpha)^2 \sigma_2^2 \Big|_{\alpha = \frac{\sigma_2^2}{\sigma_1^2 + \sigma_2^2}} \\ &= \frac{\sigma_1^2 \sigma_2^2}{\sigma_1^2 + \sigma_2^2} \end{aligned}$$

基本不等式

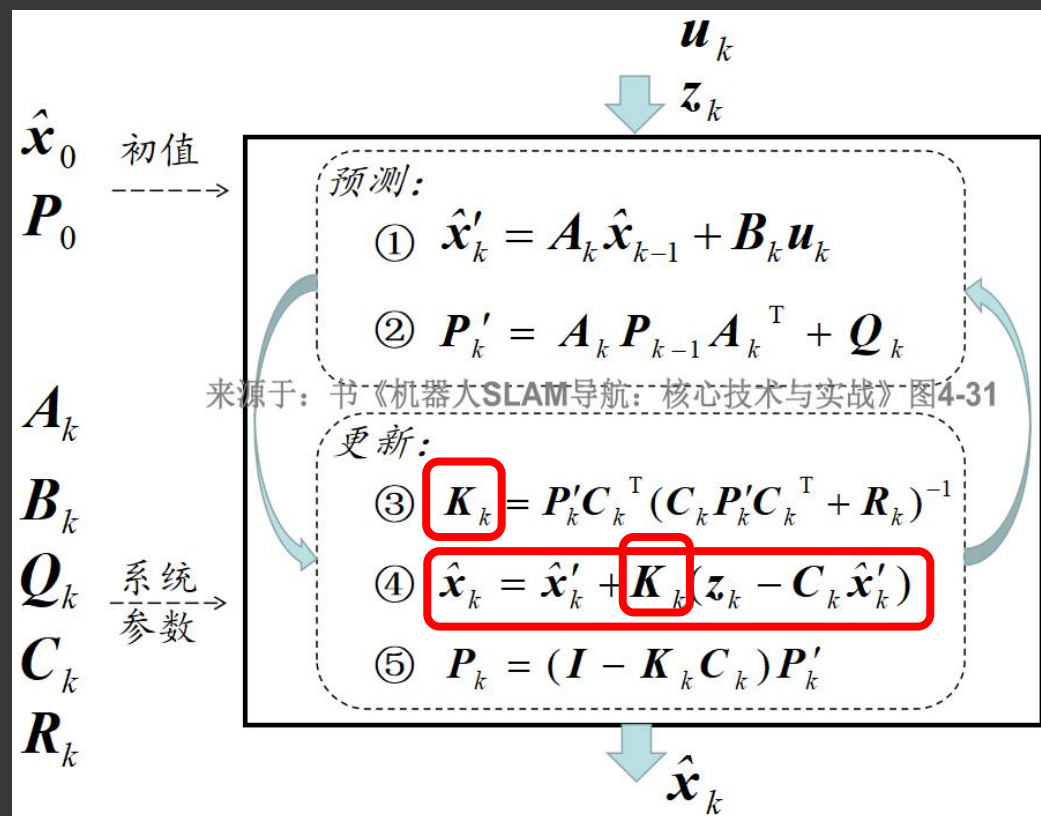
$$\frac{\sigma_1^2 \sigma_2^2}{\sigma_1^2 + \sigma_2^2} \leq \frac{\sigma_1^2 + \sigma_2^2}{4}$$

4.1 惯性测量单元

- 工作原理
- 原始数据采集
- 参数标定
- 数据滤波
- 姿态融合



卡尔曼滤波



卡尔曼滤波系统框图

4.1 惯性测量单元

- 工作原理
- 原始数据采集
- 参数标定
- 数据滤波
- 姿态融合 

卡尔曼滤波具体怎么应用？

① 卡尔曼滤波是一种很通用的思想，最优估计

信号处理、数据融合、EKF-SLAM、运动预测、自动控制

② 卡尔曼滤波，有很多变种

EKF/ESKF、UKF/CKF、SRCKF、FP_EKF
(解析法、点估计法、Square-Root Cubature KF、Fixed Point EKF)

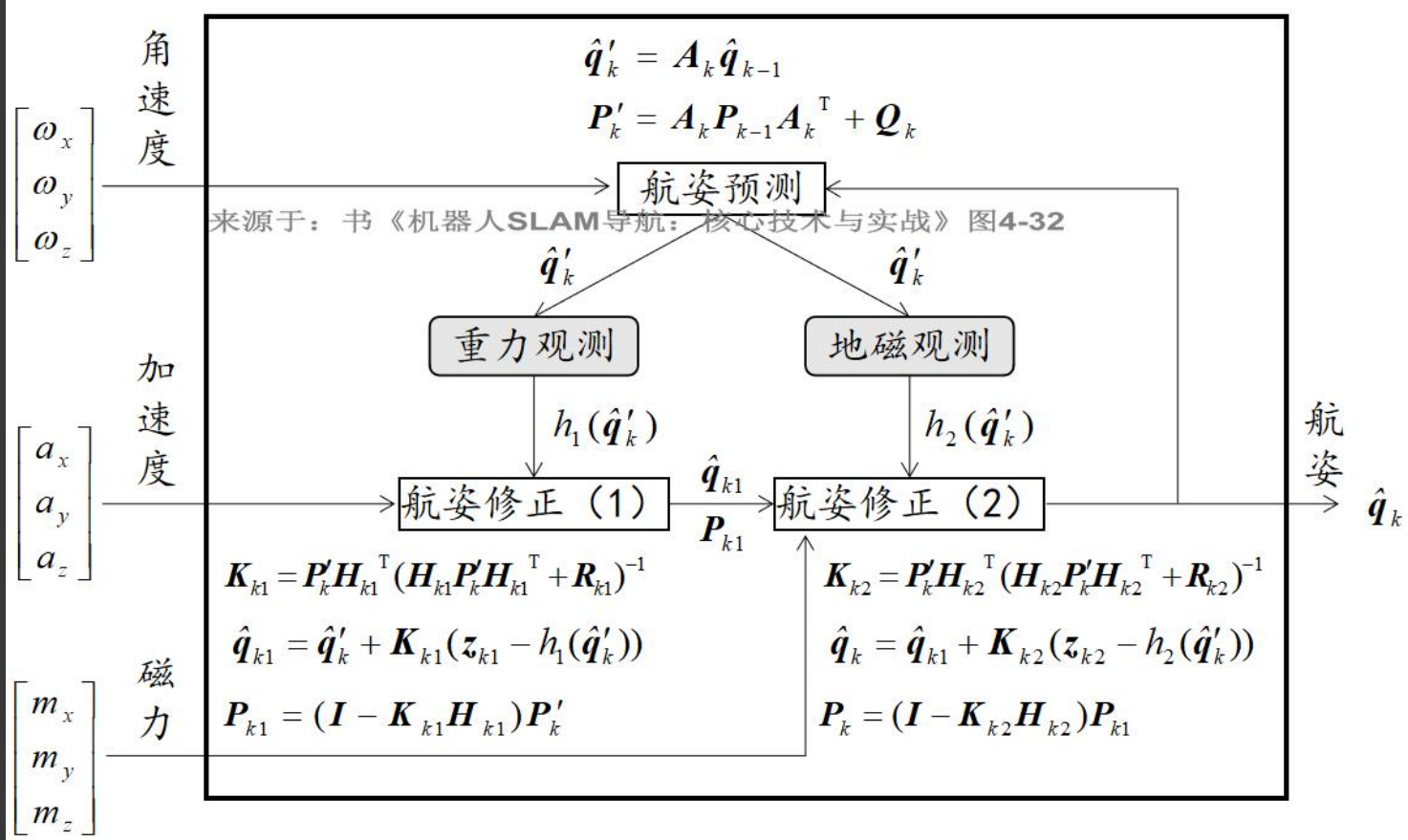
③ 卡尔曼滤波的经典框架，要替换成具体的数学模型和数据参量

状态量 (x)，状态量协方差矩阵 (P)
状态转移方程系数 (A, B, Q)
观测方程系数 (C, R)

4.1 惯性测量单元

- 工作原理
- 原始数据采集
- 参数标定
- 数据滤波
- 姿态融合 ➡

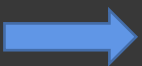
卡尔曼滤波具体怎么应用？



9轴IMU卡尔曼滤波具体实现

4.1 惯性测量单元

- 工作原理
- 原始数据采集
- 参数标定
- 数据滤波
- 姿态融合

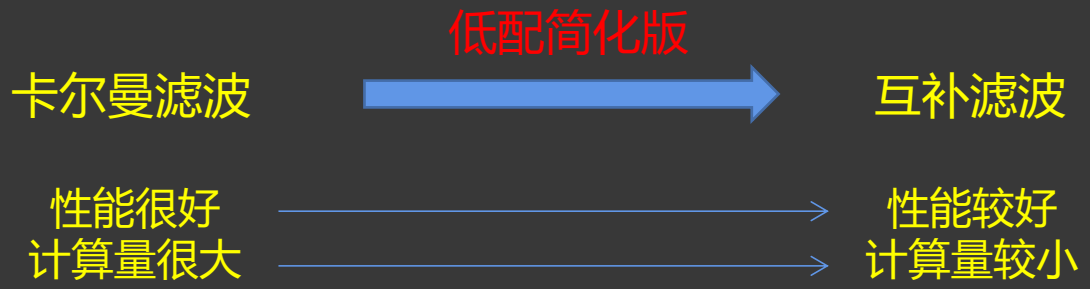


■ 卡尔曼滤波
利用卡尔曼滤波进行融合，虽然可以得到具有优良动态性能和精度的航姿，但是由于卡尔曼滤波每一步的计算都需要计算噪声模型参数和误差修正增益参数，迭代过程计算量较大，在一些中低端处理系统中不易实现。这里就来介绍一种替代的方法，即互补滤波，保证融合精度的同时计算量上会大大减小。

■ 互补滤波
互补滤波也是采用了加权平均的思想，跟卡尔曼滤波相比，互补滤波中的加权系数的求解会简单很多。

$$\hat{q} = \alpha \cdot q_1 + (1 - \alpha) \cdot q_2$$

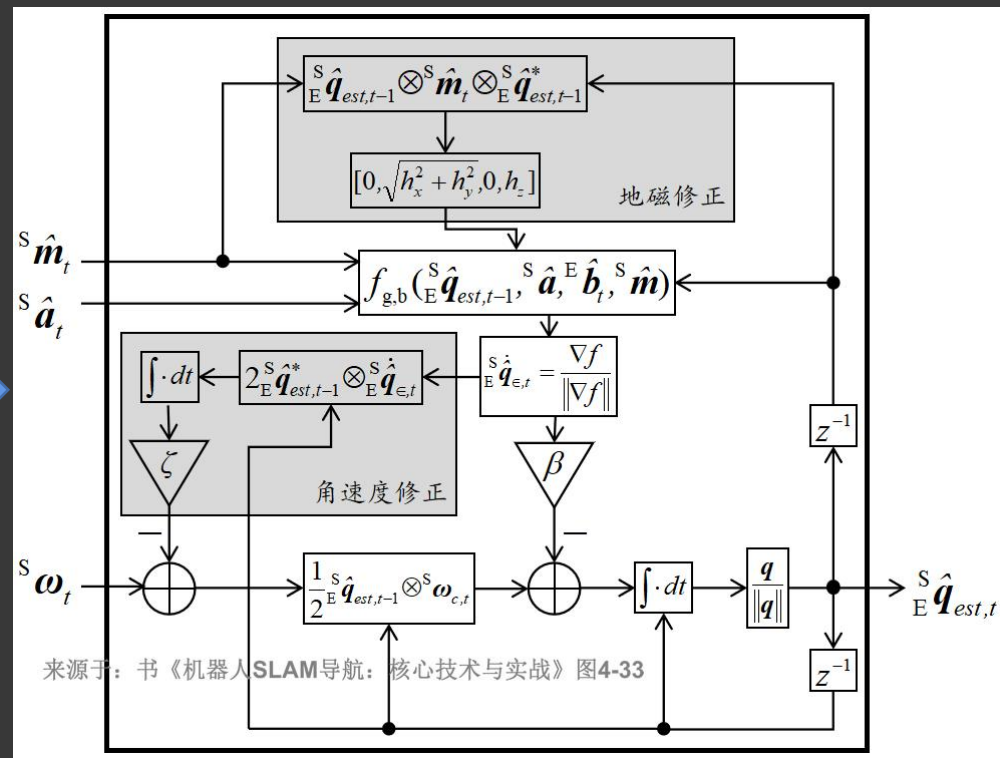
- Madgwick
- Mahony



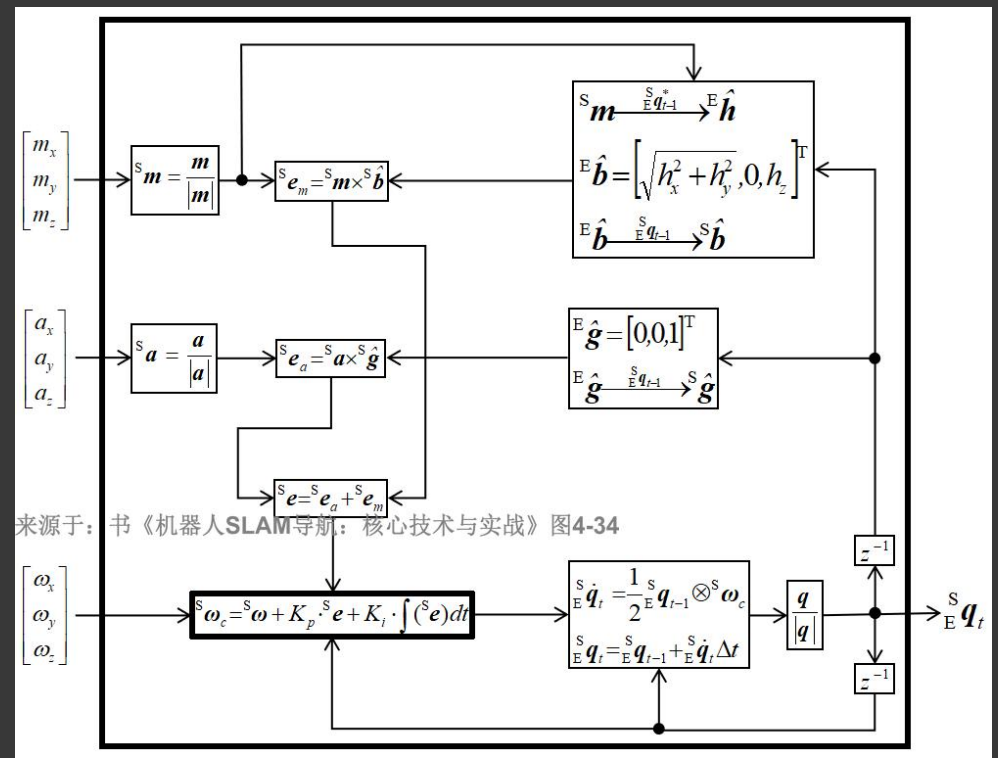
4.1 惯性测量单元

- 工作原理
- 原始数据采集
- 参数标定
- 数据滤波
- 姿态融合

互补滤波



Madgwick滤波具体实现

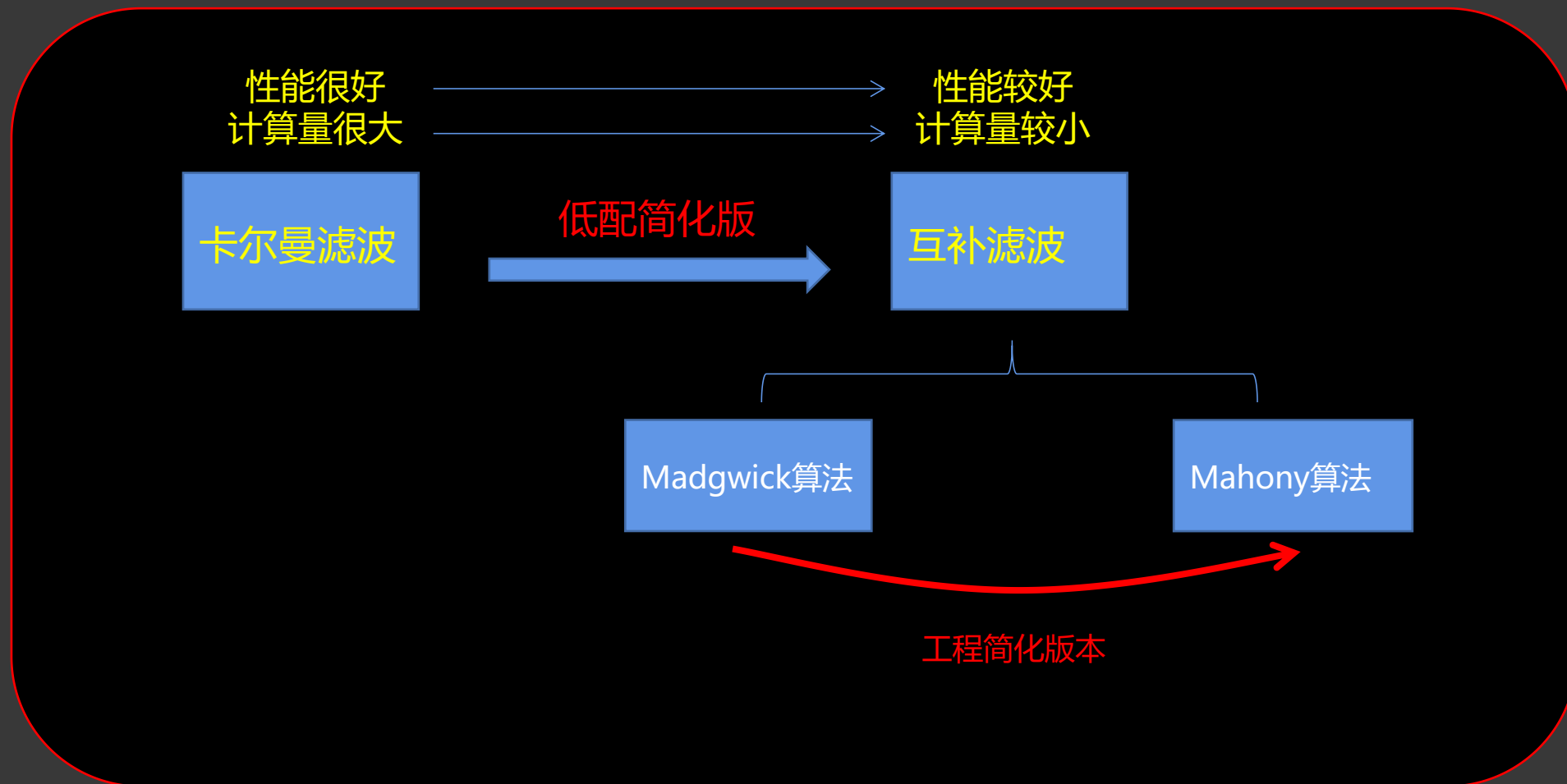
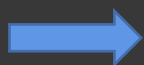


Mahony滤波具体实现

工程简化版本

4.1 惯性测量单元

- 工作原理
- 原始数据采集
- 参数标定
- 数据滤波
- 姿态融合



内容概要

4.1 惯性测量单元

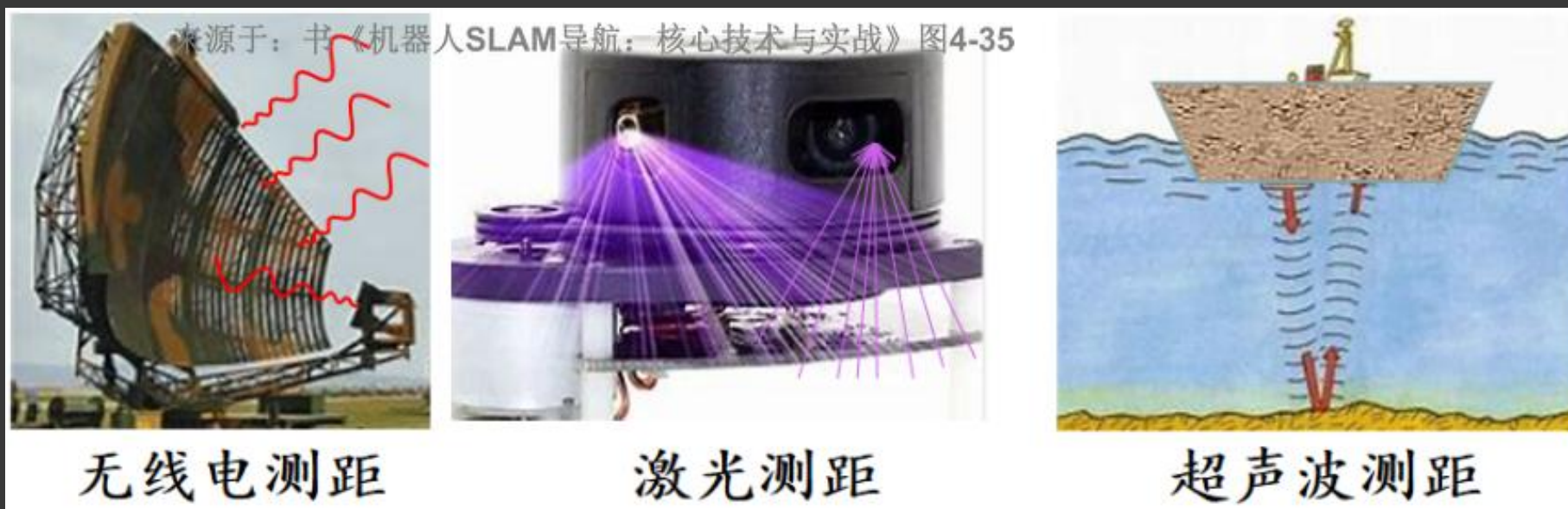
4.2 激光雷达

4.3 相机

4.4 带编码器的减速电机

4.2 激光雷达

什么是雷达？



①探测介质

②距离测量

4.2 激光雷达

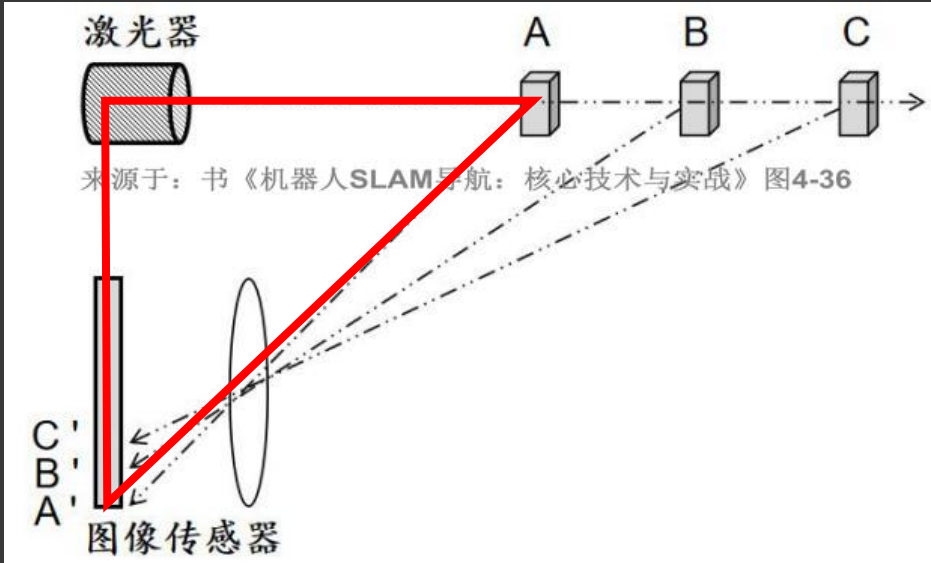
- 工作原理

■ 性能参数

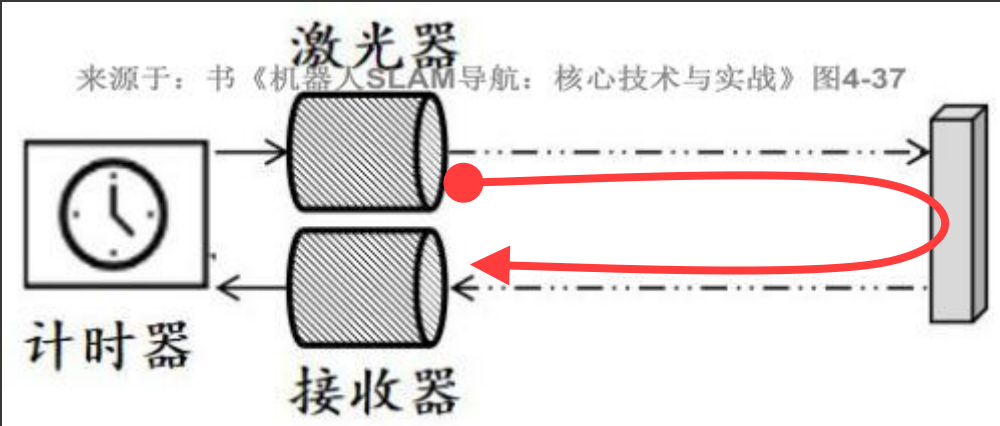
■ 数据处理
- ➡

■ 测距原理

■ 扫描原理



三角测距
(成本低，精度低)



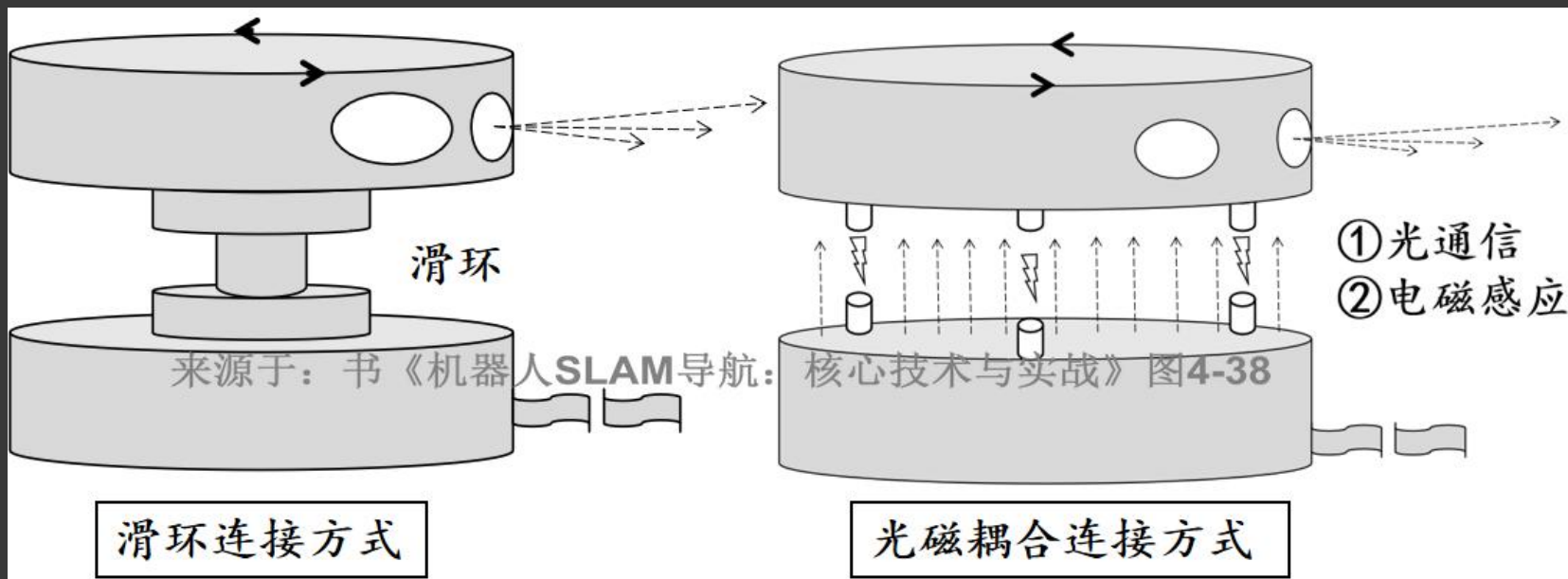
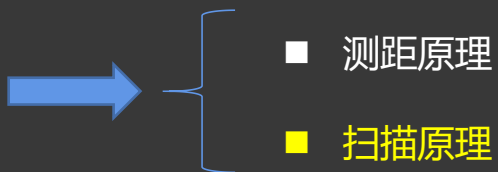
TOF (Time of Fly) 测距
(成本高，精度高)

4.2 激光雷达

■ 工作原理

■ 性能参数

■ 数据处理



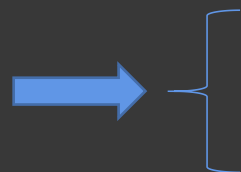
测距模组旋转机构

4.2 激光雷达

■ 工作原理

■ 性能参数

■ 数据处理



■ 测距原理

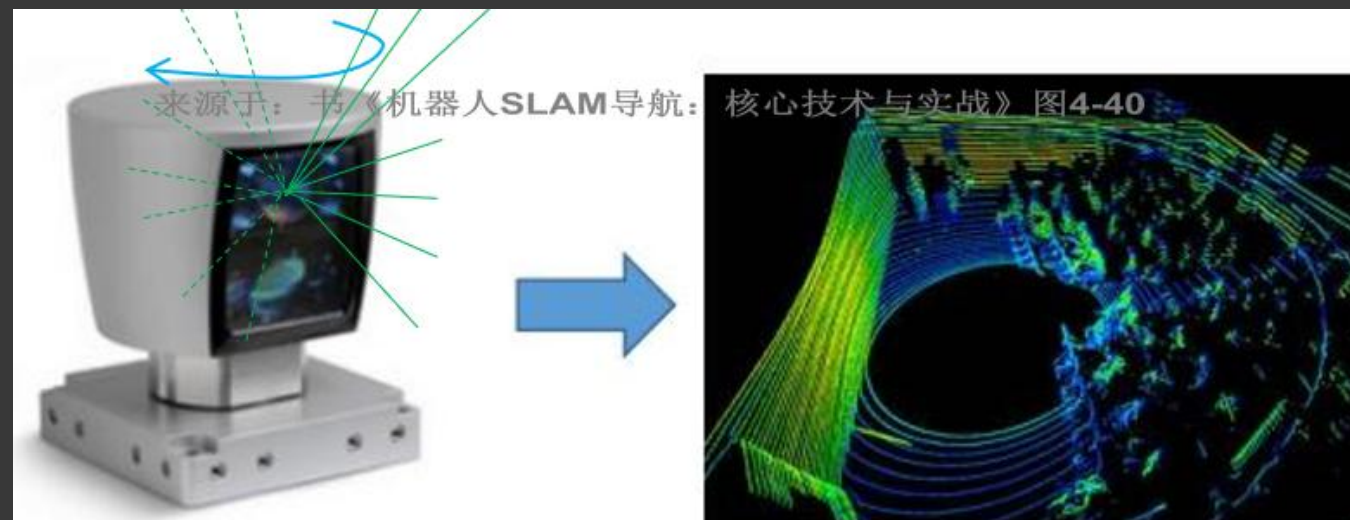
■ 扫描原理



单线激光雷达（点动成面）



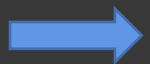
其他激光雷达



多线激光雷达（面动成体）

4.2 激光雷达

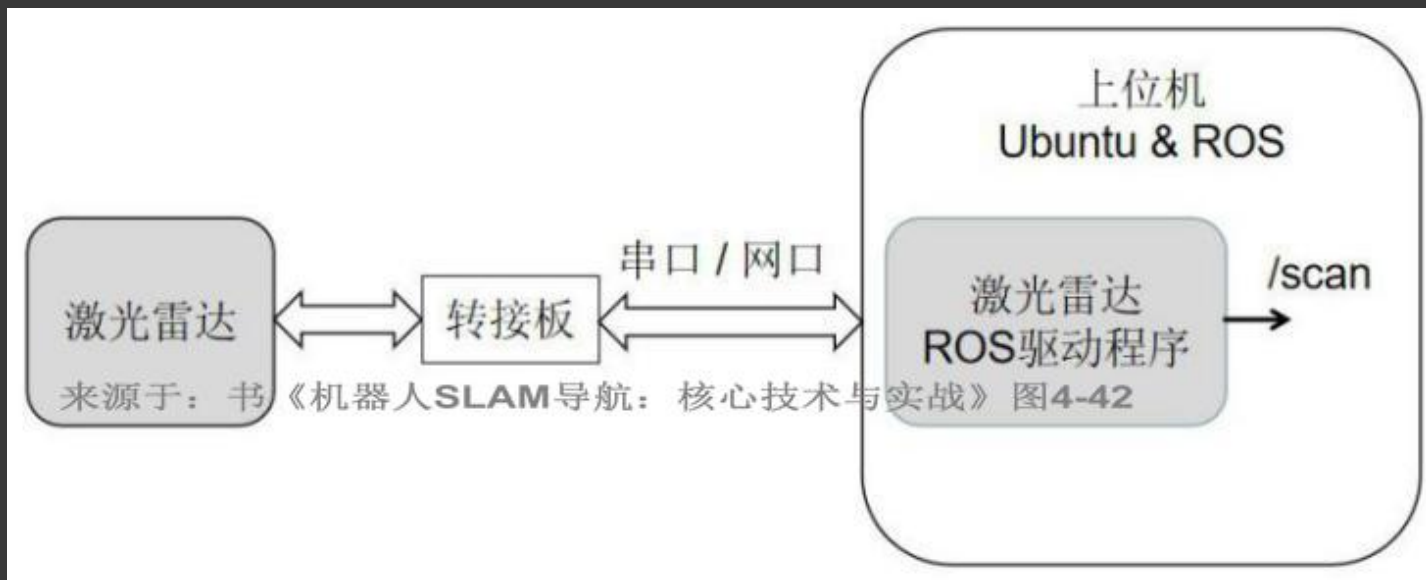
- 工作原理
- 性能参数
- 数据处理



- 激光线数
- 测距频率
- 扫描频率
- 测距量程
- 扫描角度
- 距离分辨率
- 角度分辨率
- 使用寿命

4.2 激光雷达

- 工作原理
 - 性能参数
 - 数据处理
- ➔
- 上位机ROS驱动程序
 - 扫描点处理
 - 扫描点的时间同步



激光雷达原始数据采集

话题/scan的数据类型：

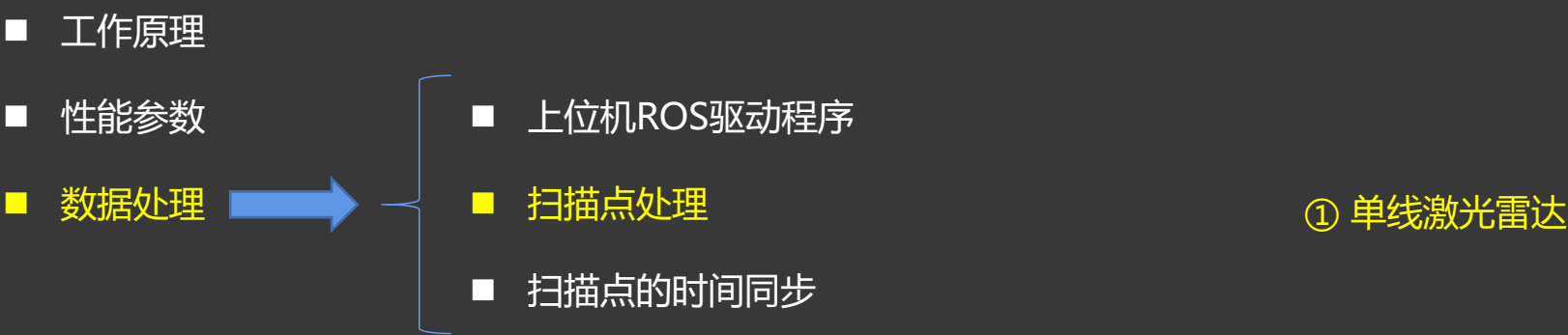
sensor_msgs::LaserScan

sensor_msgs::MultiEchoLaserScan

sensor_msgs::PointCloud2

http://docs.ros.org/api/sensor_msgs/html/msg/LaserScan.html
http://docs.ros.org/api/sensor_msgs/html/msg/MultiEchoLaserScan.html
http://docs.ros.org/api/sensor_msgs/html/msg/PointCloud2.html

4.2 激光雷达



laser_filters功能包	LaserArrayFilter	将雷达数据存入数组便于后续处理
	ScanShadowsFilter	滤除因自身遮挡而产生的干扰数据
	InterpolationFilter	在可信任的扫描点之间插值
	LaserScanIntensityFilter	滤除在设定强度阈值之外的数据
	LaserScanRangeFilter	滤除在设定距离范围之外的数据
	LaserScanAngularBoundsFilter	滤除在设定扫描角度范围之外的数据
	LaserScanAngularBoundsFilterInPlace	滤除在设定扫描角度范围之内的数据
	LaserScanBoxFilter	滤除在设定区域范围之内的数据

4.2 激光雷达

- 工作原理
- 性能参数
- 数据处理



- 上位机ROS驱动程序
- 扫描点处理
- 扫描点的时间同步

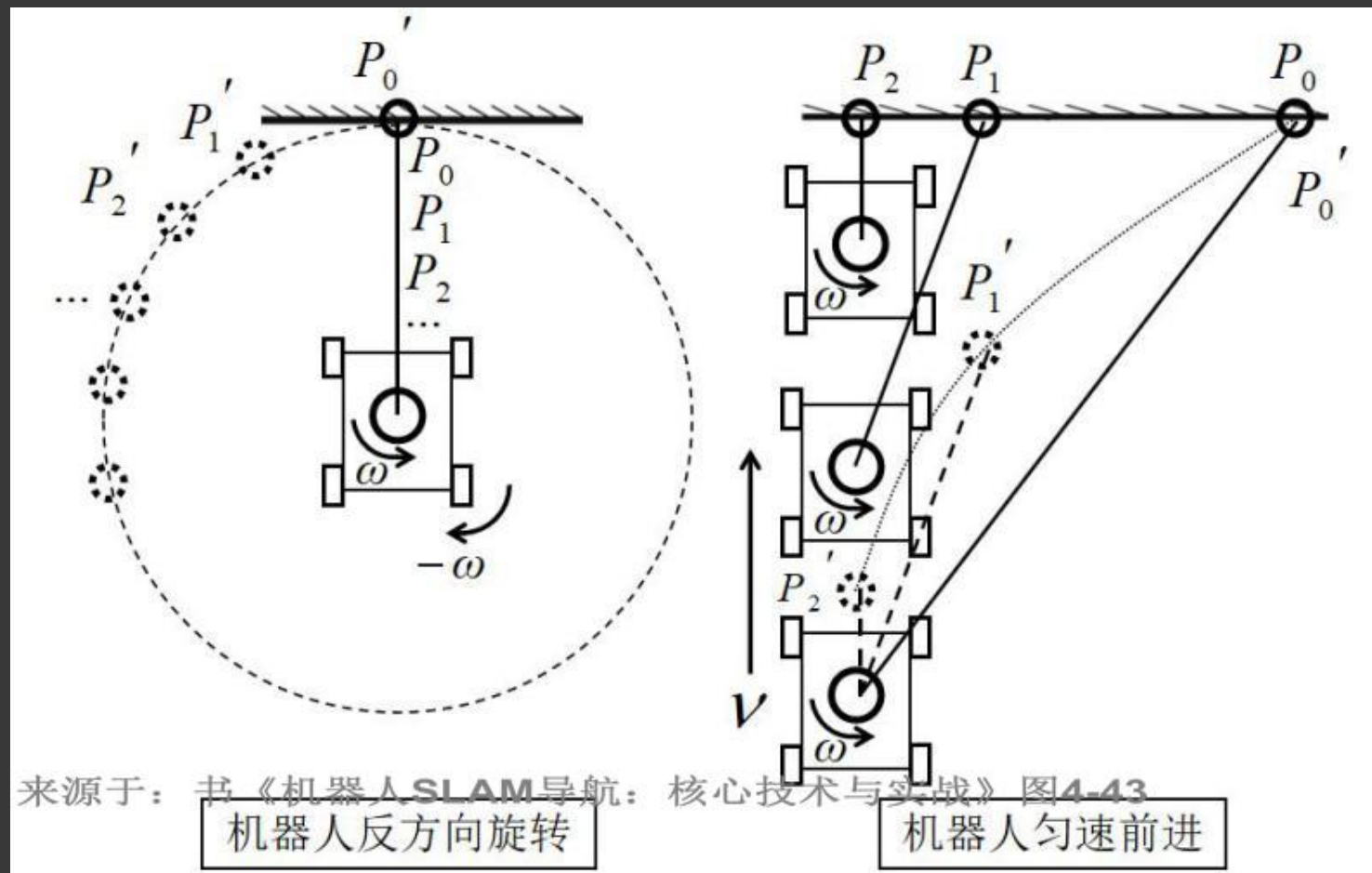
PCL点云库

② 多线激光雷达

segmentation	分割
Features	特征描述与提取
Visualization	可视化
Surface	曲面重建
Recognition	识别
Search	检索
Registration	配准
Keypoints	关键点
Filters	滤波
Sample consensus	采样一致性
octree	八叉树
common	通用模块
IO	输入输出
kdtree	k维树

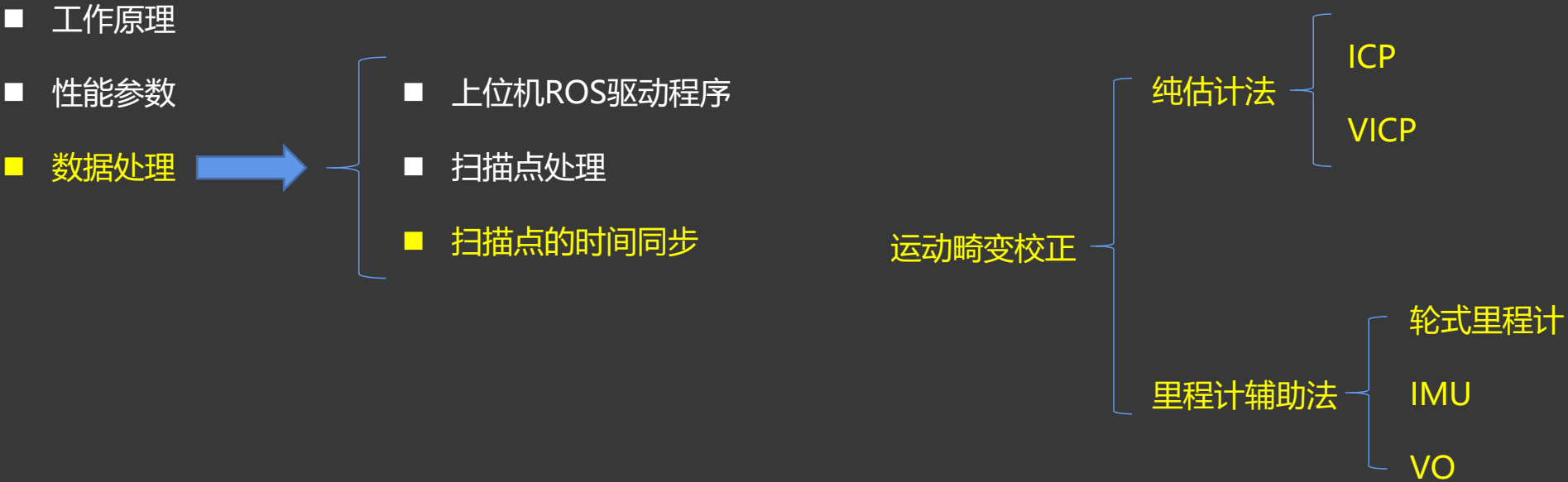
4.2 激光雷达

- 工作原理
 - 性能参数
 - 数据处理
-
- 上位机ROS驱动程序
 - 扫描点处理
 - 扫描点的时间同步



运动畸变

4.2 激光雷达



内容概要

4.1 惯性测量单元

4.2 激光雷达

4.3 相机

4.4 带编码器的减速电机

4.3 相机

相机



摄像头

录影机



摄影机



4.3 相机

机器人中用什么相机？



4.3 相机

- 单目相机
 - 双目相机
 - RGB-D相机
-
- 单目相机原理
 - 上位机ROS驱动程序
 - 单目相机标定
 - 单目图像传输



4.3 相机

■ 单目相机

■ 双目相机

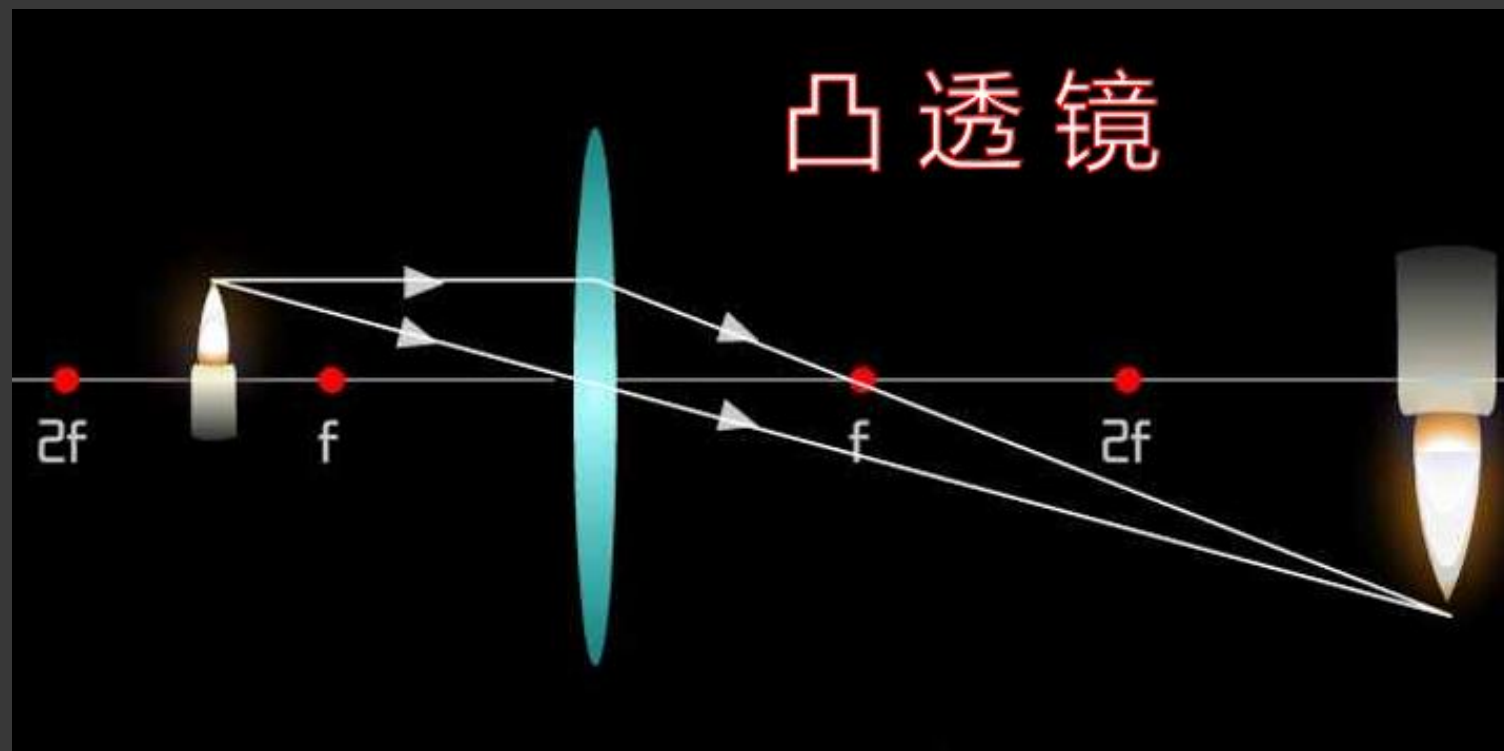
■ RGB-D相机

■ 单目相机原理

■ 上位机ROS驱动程序

■ 单目相机标定

■ 单目图像传输



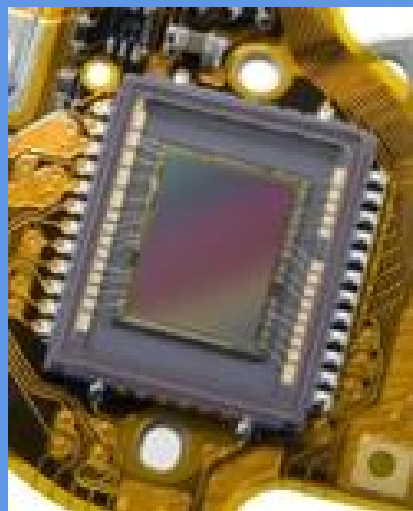
4.3 相机

- 单目相机
 - 单目相机原理
 - 上位机ROS驱动程序
 - 单目相机标定
 - 单目图像传输
- 双目相机
- RGB-D相机

将光线聚焦到成像芯片

小孔成像 + 凸透镜 = 单目相机

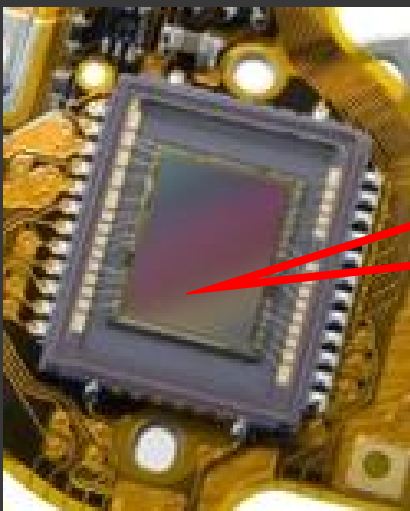
成像芯片 (CMOS/CCD)



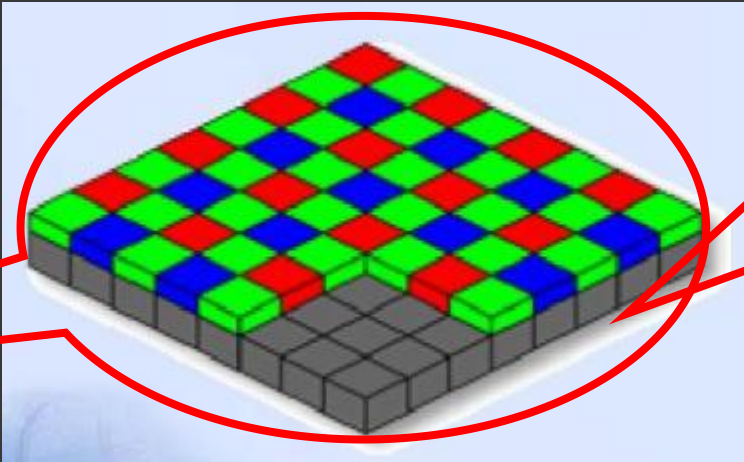
镜头

4.3 相机

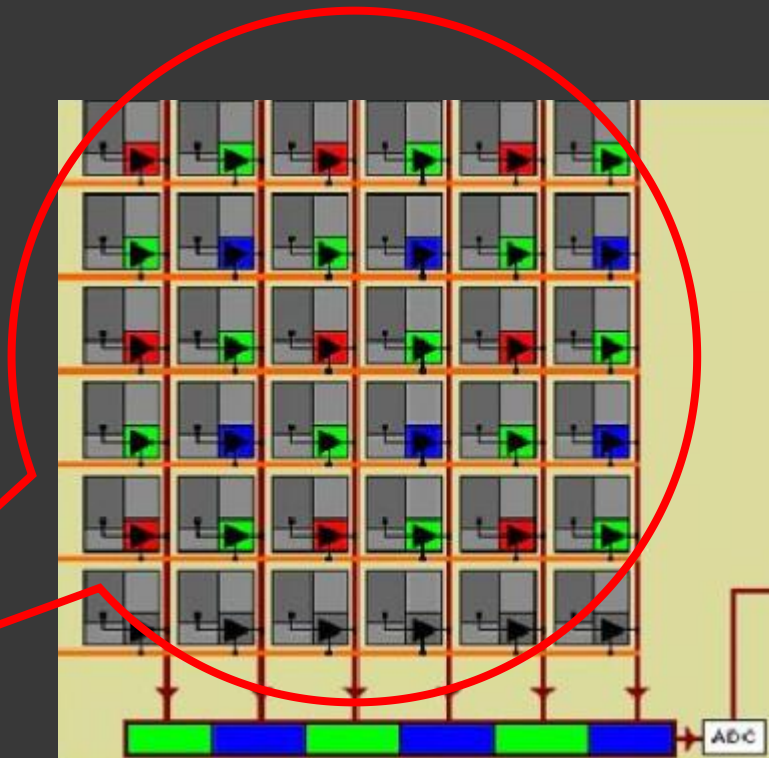
- 单目相机
 - 单目相机原理
 - 上位机ROS驱动程序
 - 单目相机标定
 - 单目图像传输
- 双目相机
- RGB-D相机



成像芯片



像素阵列



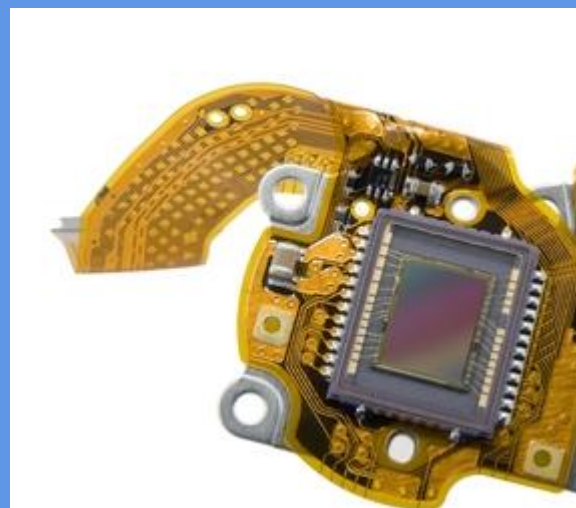
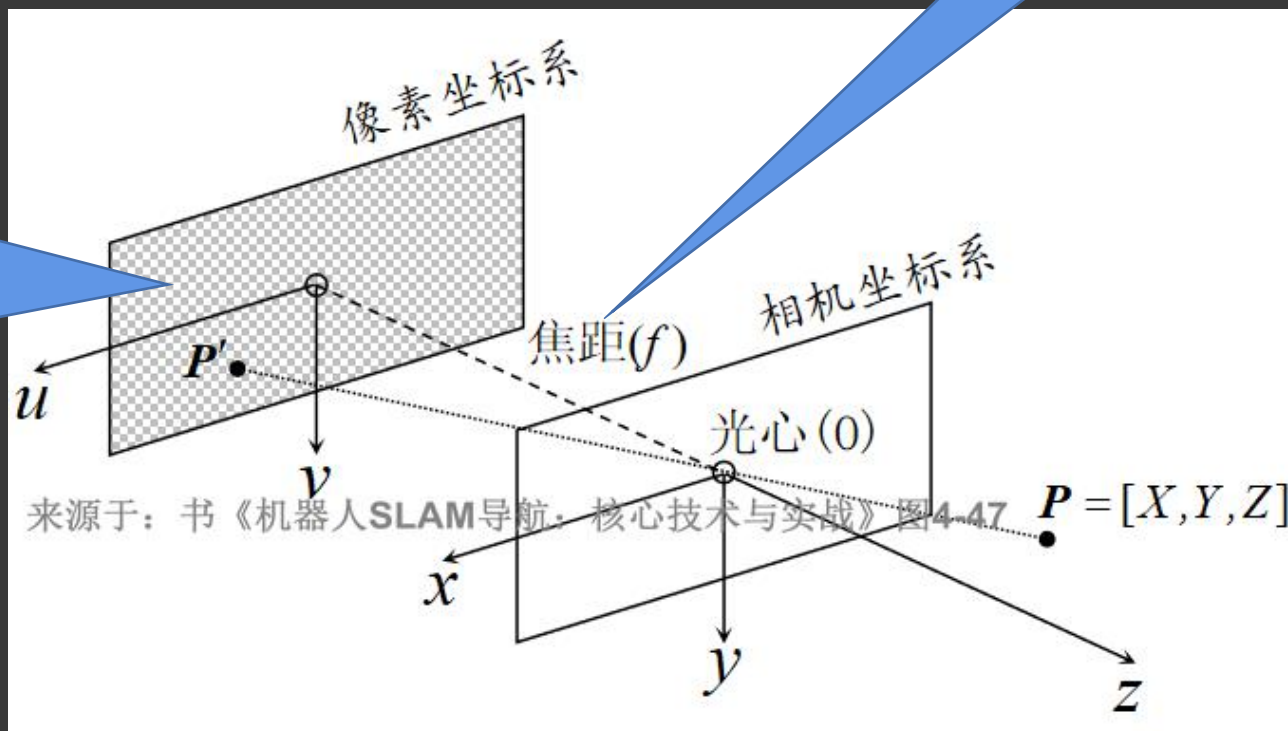
微电路

4.3 相机

- 单目相机
- 双目相机
- RGB-D相机

■ 单目相机原理

- 上位机ROS驱动程序
- 单目相机标定
- 单目图像传输



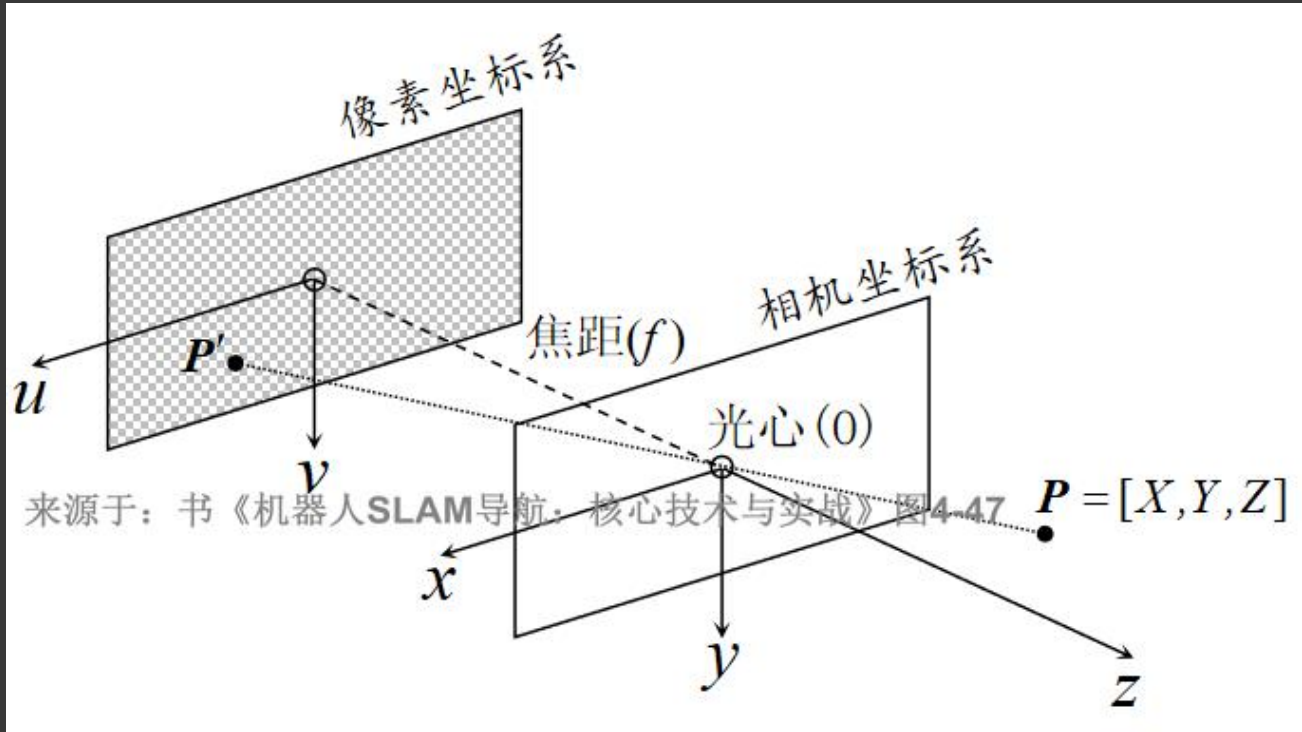
4.3 相机

- 单目相机
- 双目相机
- RGB-D相机
- 单目相机原理

■ 上位机ROS驱动程序

■ 单目相机标定

■ 单目图像传输



无畸变内参模型

$$\begin{cases} U = \alpha(\frac{f}{Z} \cdot X) + c_x \\ V = \beta(\frac{f}{Z} \cdot Y) + c_y \end{cases}$$



$$\begin{bmatrix} U \\ V \\ 1 \end{bmatrix} = \frac{1}{Z} \begin{bmatrix} f_x & 0 & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = \frac{1}{Z} \mathbf{K} \mathbf{P}$$

理想情况的模型

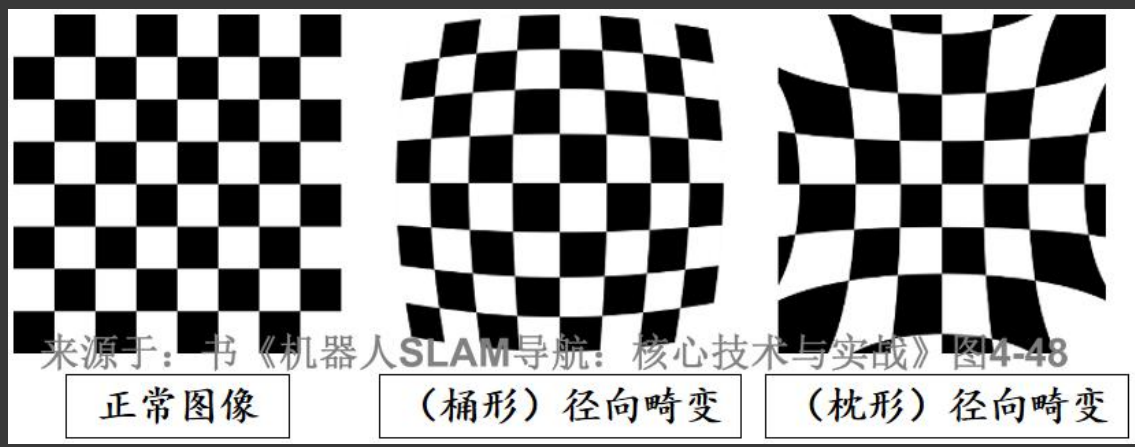
4.3 相机

- 单目相机
- 双目相机
- RGB-D相机
- 单目相机原理
- 上位机ROS驱动程序
- 单目相机标定
- 单目图像传输

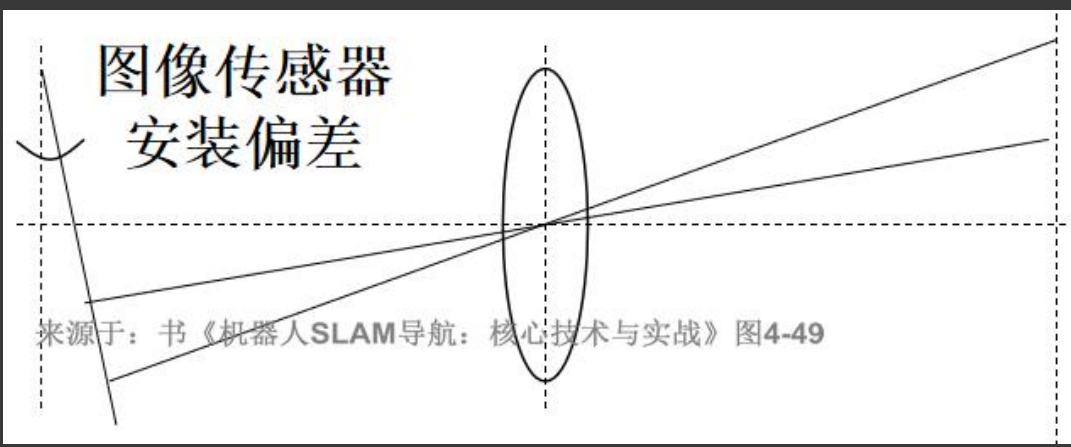
畸变内参模型

$$\begin{cases} r^2 = X^2 + Y^2 \\ X_{\text{distort}} = X \cdot (1 + k_1 r^2 + k_2 r^4 + k_3 r^6) + 2p_1 XY + p_2 \cdot (r^2 + 2X^2) \\ Y_{\text{distort}} = Y \cdot (1 + k_1 r^2 + k_2 r^4 + k_3 r^6) + 2p_2 XY + p_1 \cdot (r^2 + 2Y^2) \\ U = \frac{f_x}{Z} \cdot X_{\text{distort}} + c_x \\ V = \frac{f_y}{Z} \cdot Y_{\text{distort}} + c_y \end{cases}$$

实际情况的模型

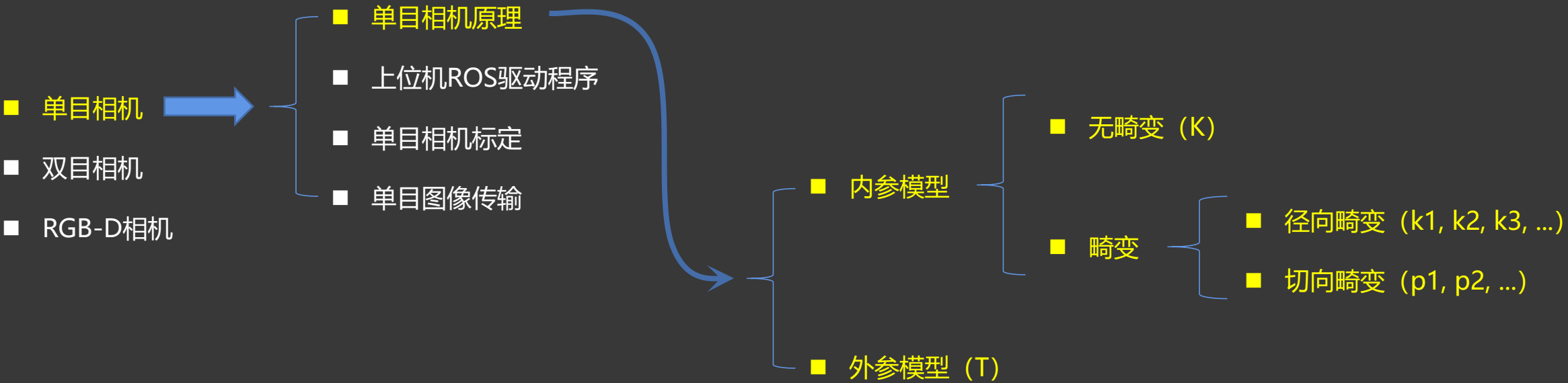


径向畸变



切向畸变

4.3 相机

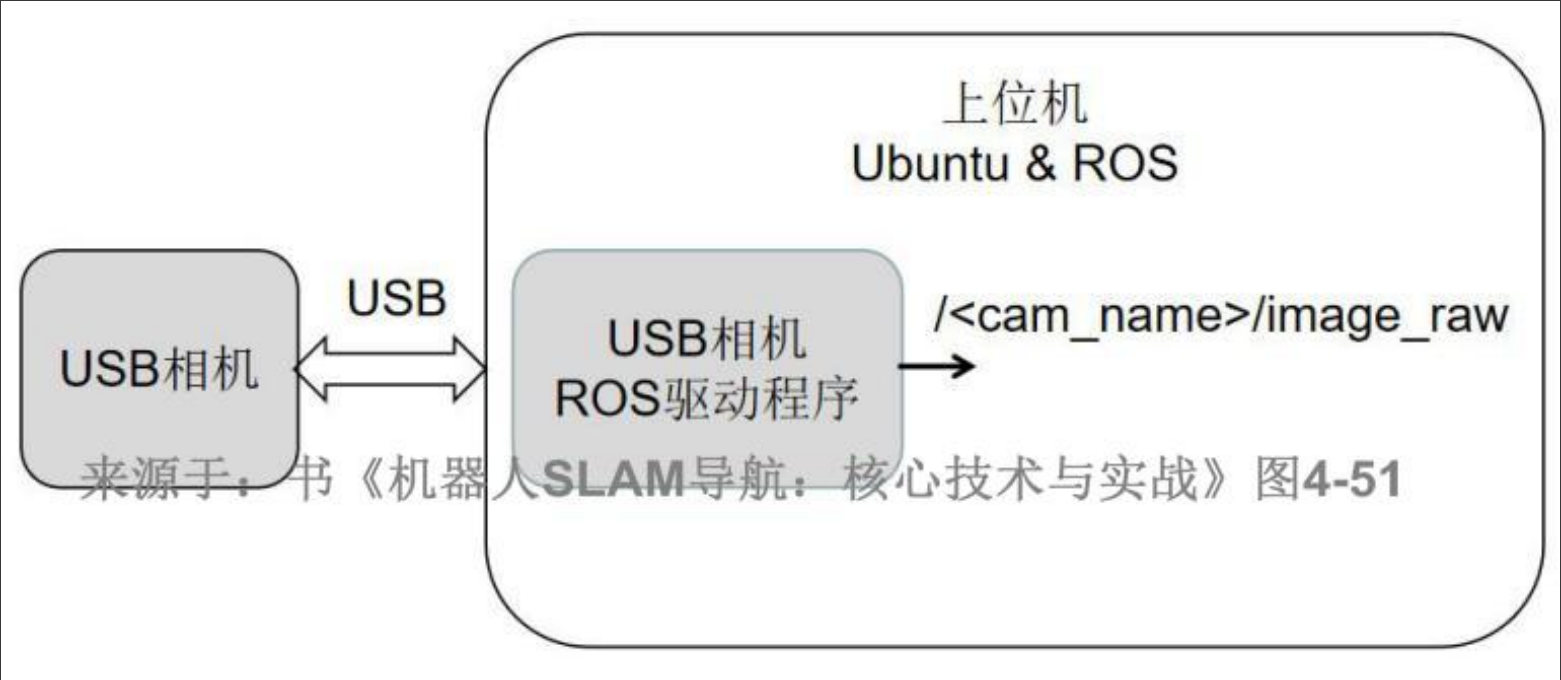
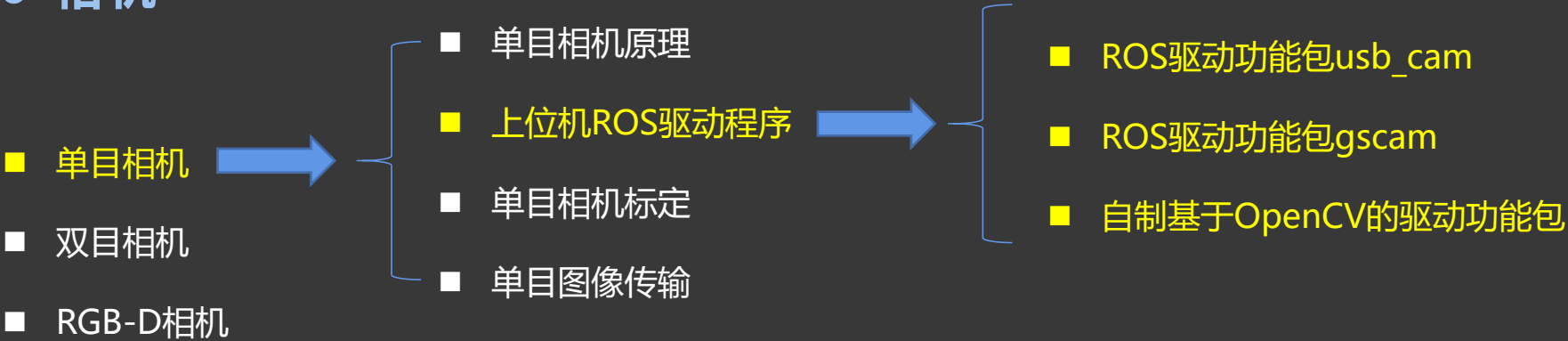


$$\begin{bmatrix} U \\ V \\ 1 \end{bmatrix} = \frac{1}{Z} \boldsymbol{K} \boldsymbol{P}^c = \frac{1}{Z} \boldsymbol{K} (\boldsymbol{R} \boldsymbol{P}^w + \boldsymbol{t}) = \frac{1}{Z} \boldsymbol{K} \boldsymbol{T} \boldsymbol{P}^w$$

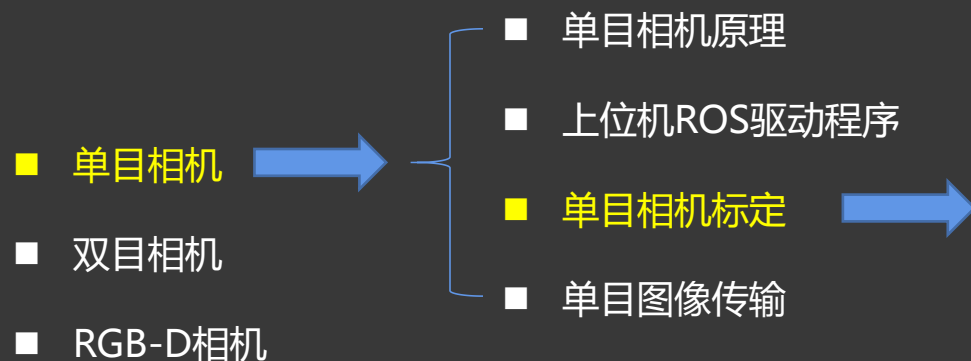
■ 内参模型

■ 外参模型

4.3 相机



4.3 相机



什么是内参标定?

$$\begin{bmatrix} U \\ V \\ 1 \end{bmatrix} = \frac{1}{Z} \mathbf{K} \mathbf{P}^c = \frac{1}{Z} \mathbf{K} (\mathbf{R} \mathbf{P}^w + \mathbf{t}) = \frac{1}{Z} \mathbf{K} \mathbf{T} \mathbf{P}^w$$

解方程，求内参矩阵 \mathbf{K} 以及畸变校正参数 k, p

主流方法：自标定

比如：棋盘格法

camera_calibration功能包

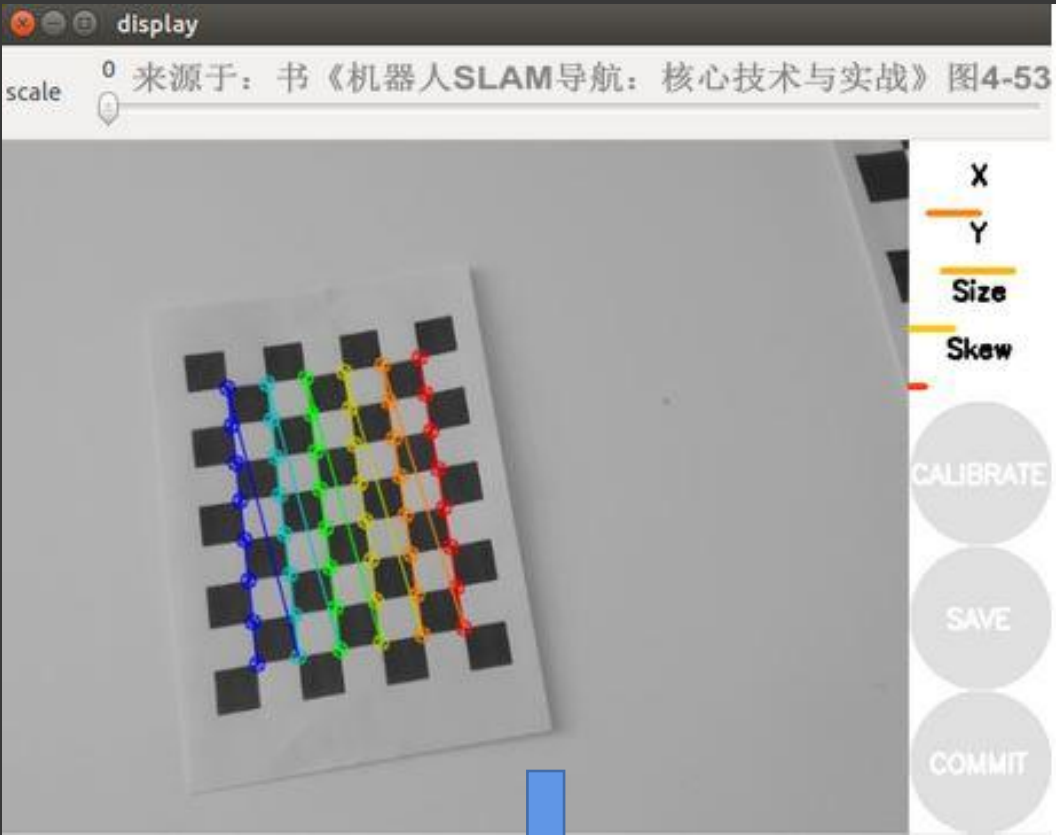
4.3 相机

■ 单目相机

■ 双目相机

■ RGB-D相机

- 单目相机原理
- 上位机ROS驱动程序
- 单目相机标定
- 单目图像传输



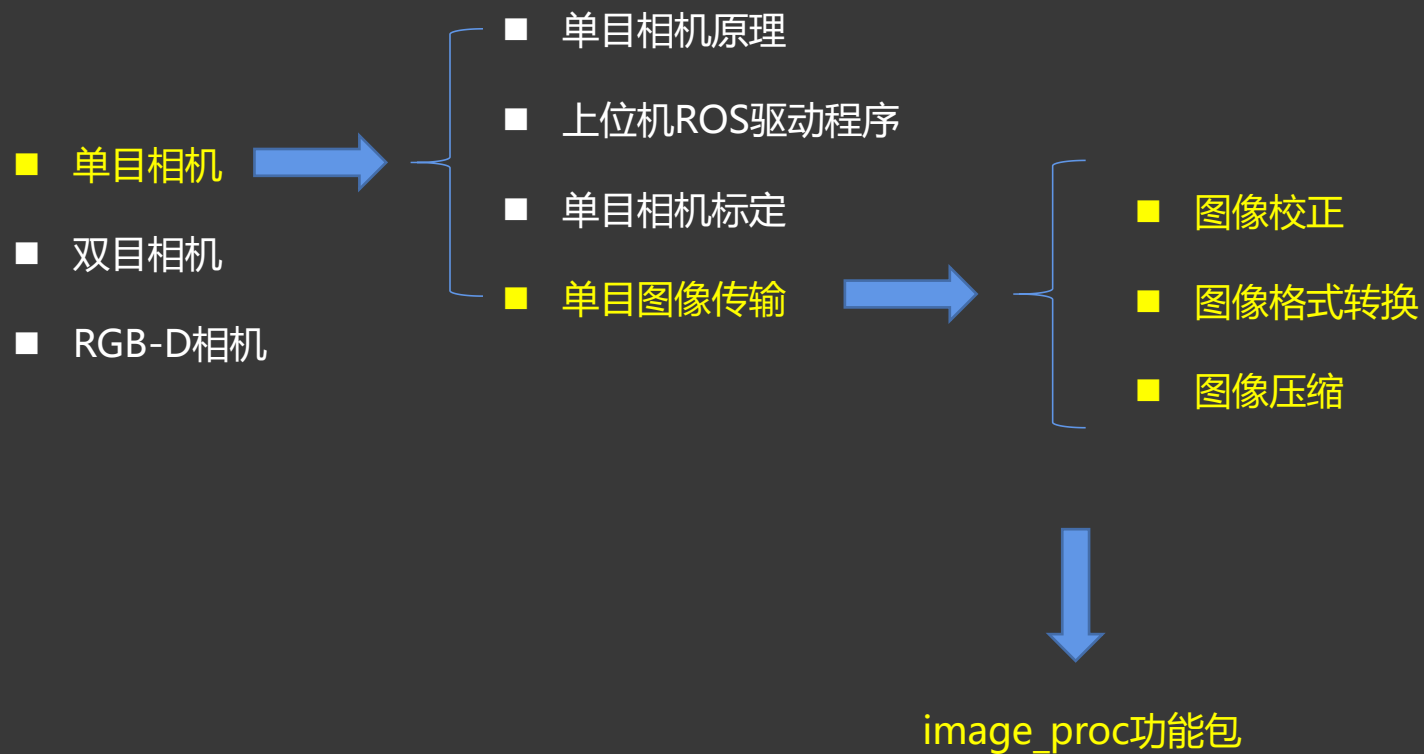
估计

内参 K 和畸变系数 k_1, k_2, k_3, p_1, p_2

保存

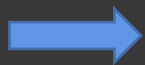
~/.ros/camera_info/head_camera.yaml

4.3 相机



4.3 相机

- 单目相机
- 双目相机
- RGB-D相机



- 双目相机原理
- 上位机ROS驱动程序
- 双目相机标定
- 双目图像传输

单目 + 单目 = 双目



4.3 相机

- 单目相机

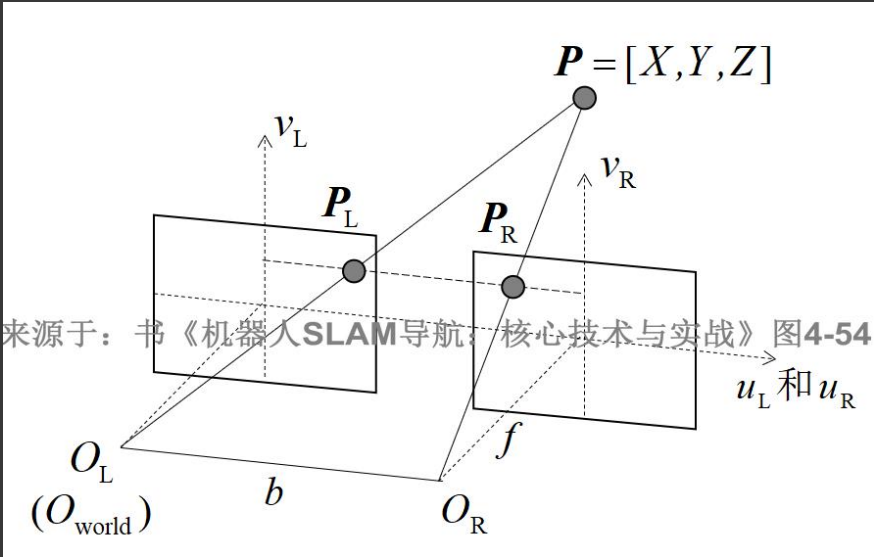
■ 双目相机

■ RGB-D相机
- 双目相机原理

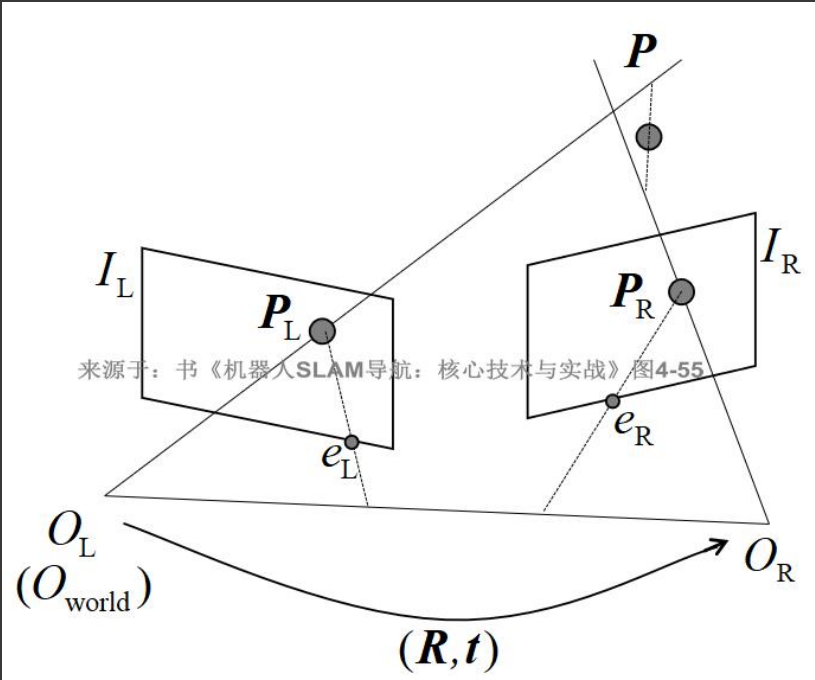
■ 上位机ROS驱动程序

■ 双目相机标定

■ 双目图像传输

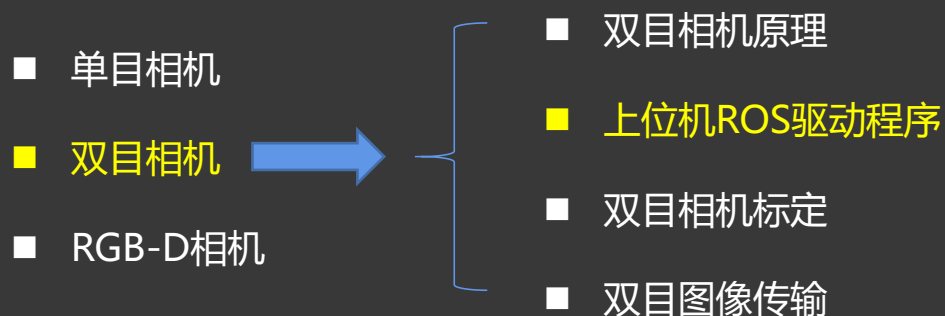


$$\frac{Z-f}{Z} = \frac{b-U_L+U_R}{b}, \text{ 化简后 } Z = \frac{f \cdot b}{U_L-U_R}$$



$$\begin{aligned} s_L P_L &= K_L P \\ s_R P_R &= K_R (R P + t) \end{aligned}$$

4.3 相机



双目相机的ROS驱动厂家一般都会提供，直接下载下来安装就行了：

- ① 一种是双目相机直接被上位机系统识别成两个独立的单目相机设备，按单目相机的方法分别读取左右相机的图像并分别发布到对应的话题；
- ② 另一种是双目相机上传的左图、右图拼接在一起的一张大图，上位机可以直接发布这张拼接的大图，也可以将大图拆分后分别发布。

4.3 相机

- 单目相机
 - 双目相机
 - RGB-D相机

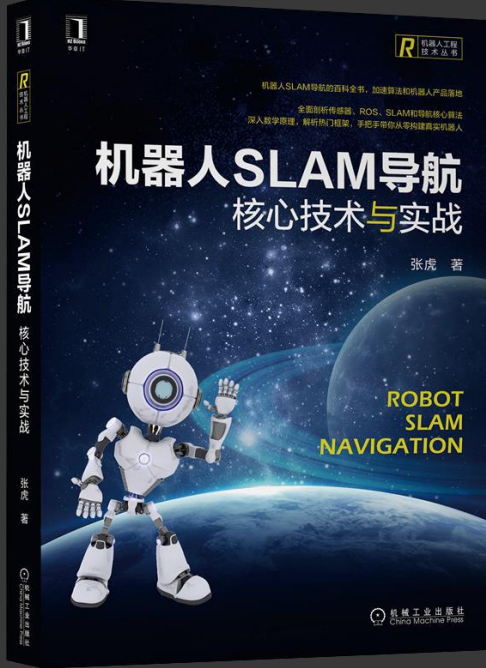
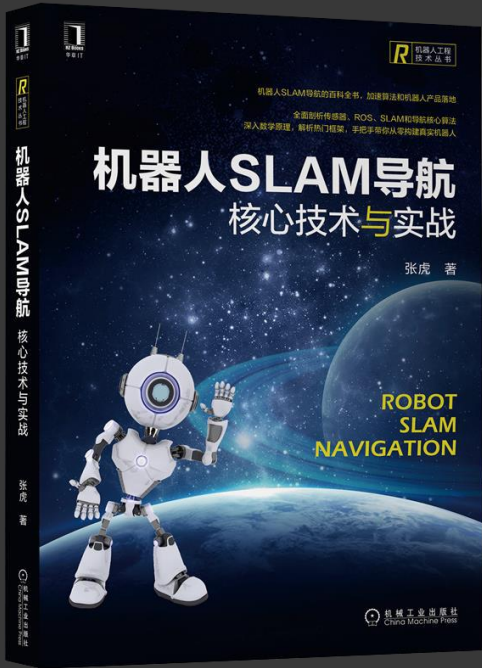
➔
- 双目相机原理
 - 上位机ROS驱动程序
 - 双目相机标定
 - 双目图像传输



独立输出
左图

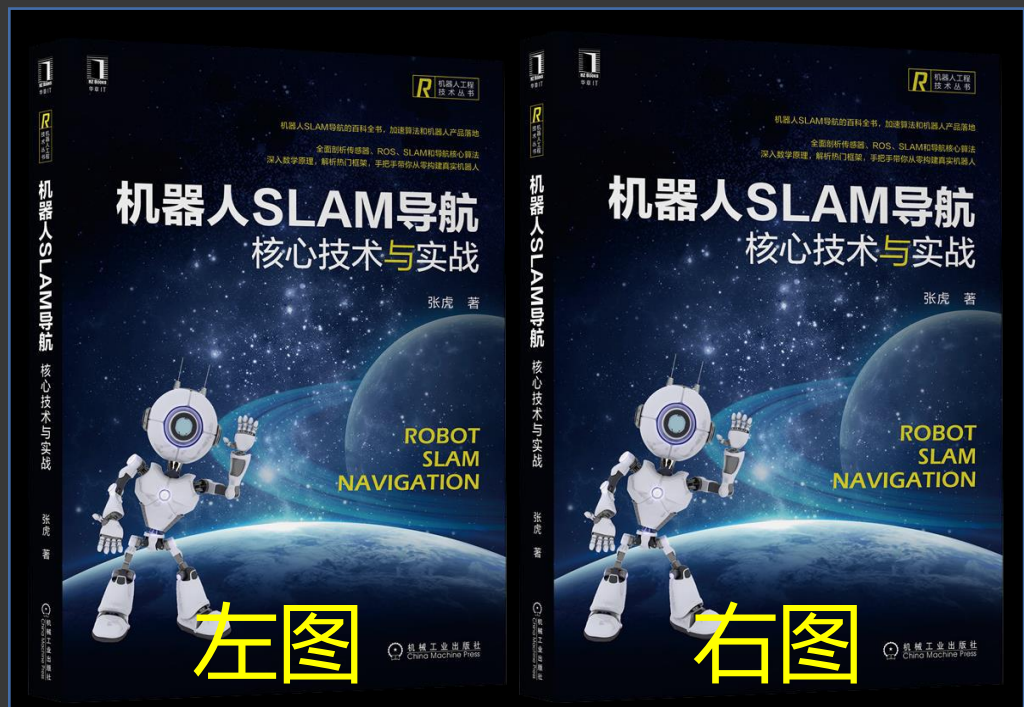


独立输出
右图



4.3 相机

- 单目相机
 - 双目相机
 - RGB-D相机
- - 双目相机原理
 - 上位机ROS驱动程序
 - 双目相机标定
 - 双目图像传输



4.3 相机

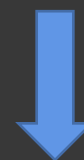
- 单目相机
- 双目相机
- RGB-D相机



- 双目相机原理
- 上位机ROS驱动程序
- 双目相机标定
- 双目图像传输

双目的内参标定

有什么不同?



$$\begin{bmatrix} U \\ V \\ 1 \end{bmatrix} = \frac{1}{Z} \mathbf{K} \mathbf{P}^c = \frac{1}{Z} \mathbf{K} (\mathbf{R} \mathbf{P}^w + \mathbf{t}) = \frac{1}{Z} \mathbf{K} \mathbf{T} \mathbf{P}^w$$

解方程，求内参矩阵 \mathbf{K} 以及畸变校正参数 k, p

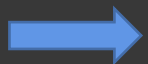
求左右相机之间的参数 \mathbf{T}

主流方法：自标定

比如：棋盘格法

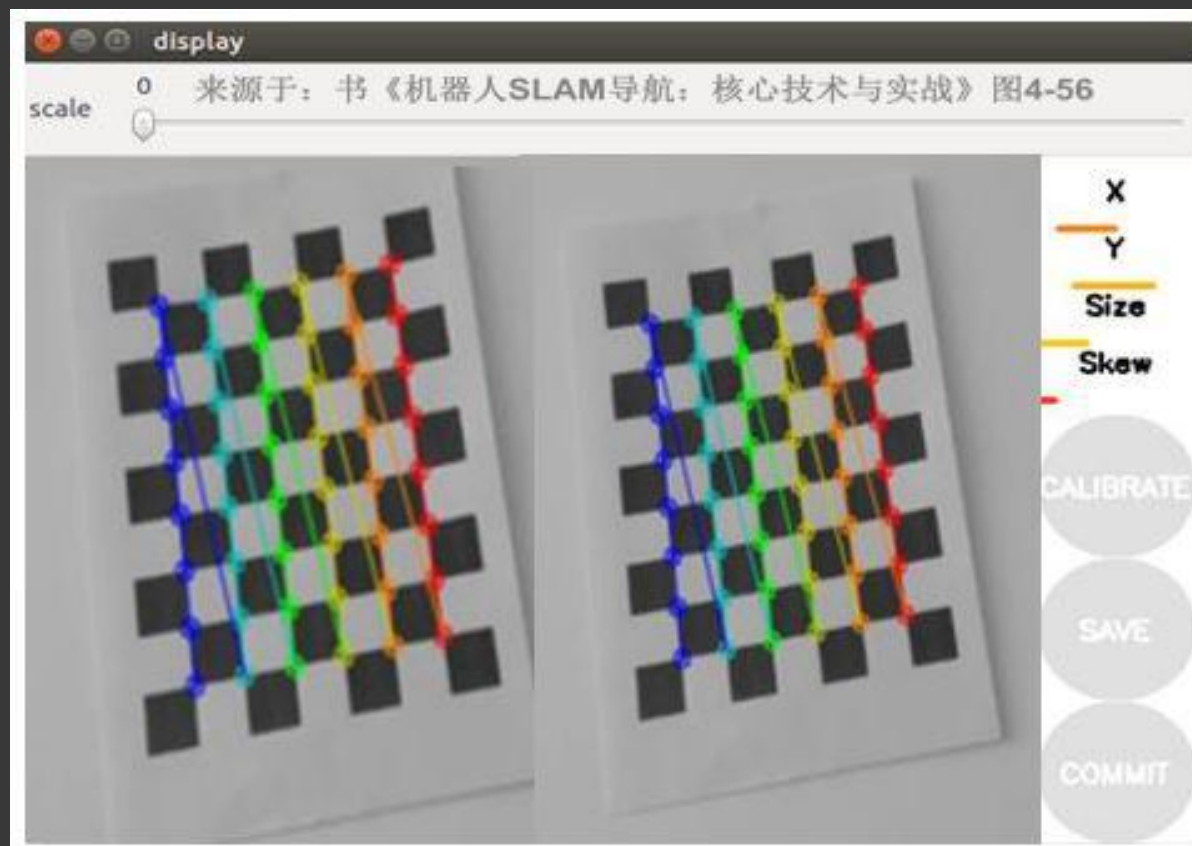
4.3 相机

- 单目相机
- 双目相机
- RGB-D相机

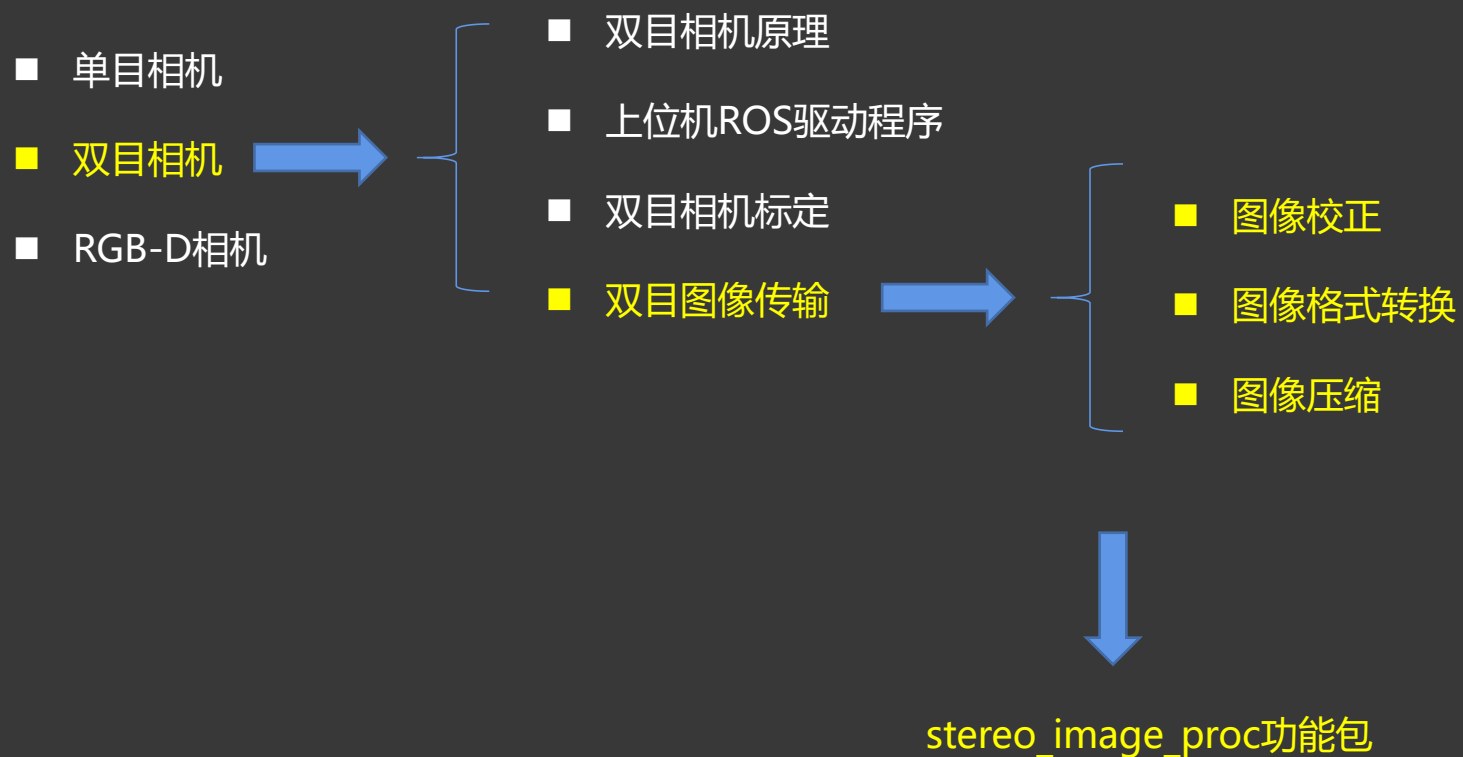


- 双目相机原理
- 上位机ROS驱动程序
- 双目相机标定
- 双目图像传输

camera_calibration功能包



4.3 相机

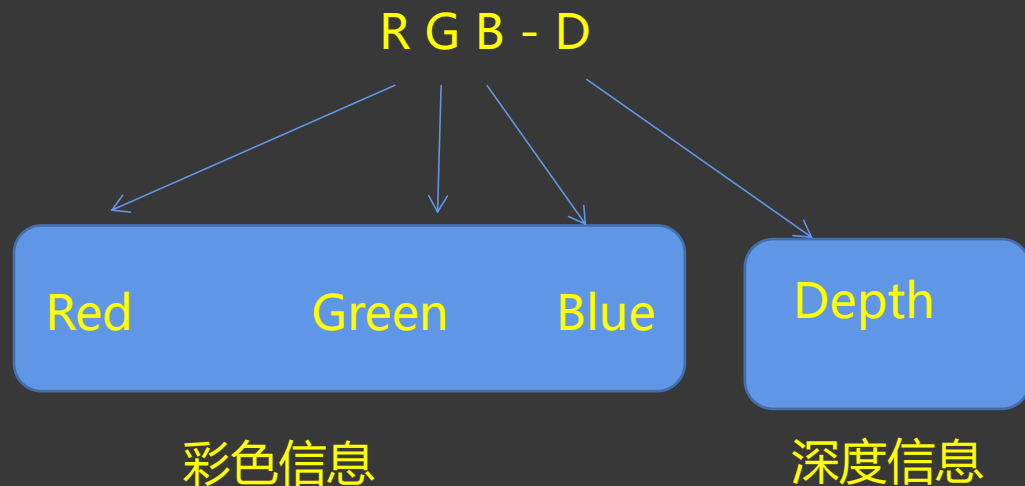


4.3 相机

- 单目相机
- 双目相机
- RGB-D相机 →

为什么需要RGB-D相机？

- 像激光雷达一样主动探测深度信息
- 同时还测量图像彩色信息

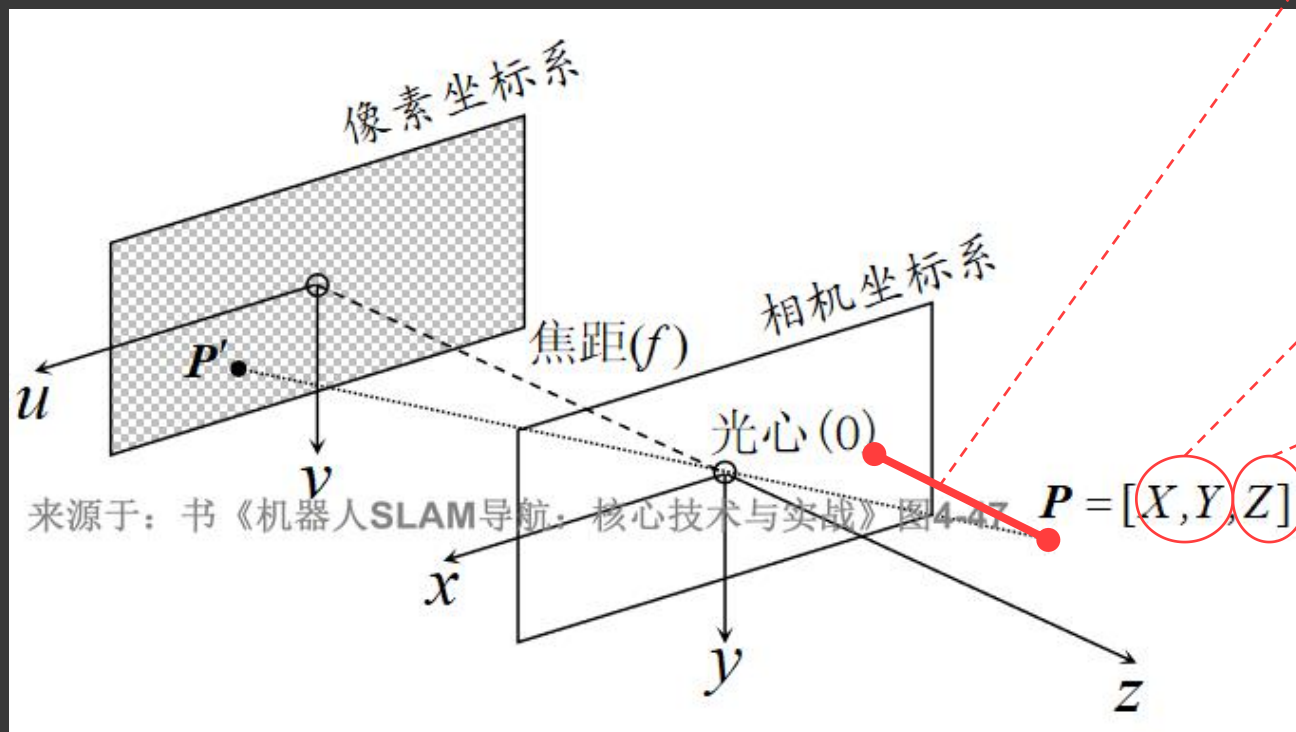


4.3 相机

- 单目相机
- 双目相机
- RGB-D相机 →

深度信息
是什么？

深度：Z



决定物体的高矮胖瘦

决定物体距离相机的远近
(深度)

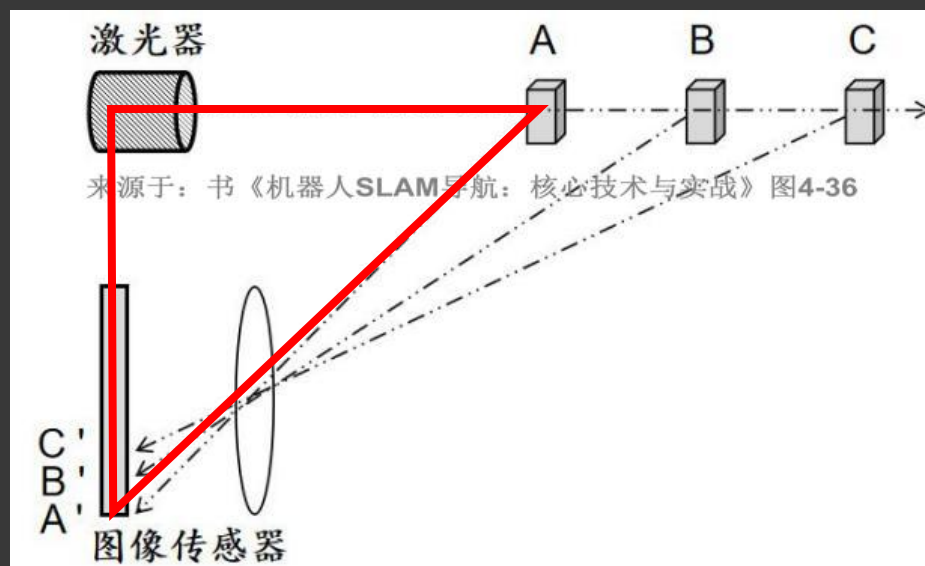
4.3 相机

■ 单目相机

■ 双目相机

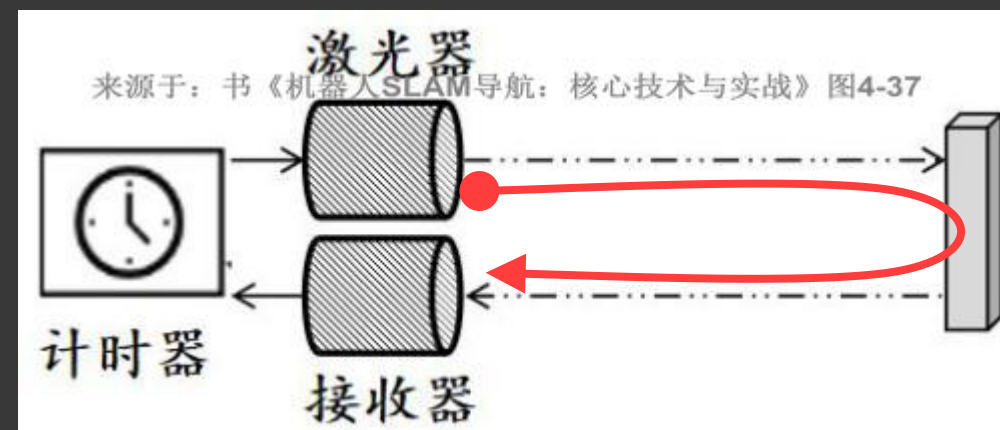
■ RGB-D相机 →

测距原理 (回顾)



三角测距

(成本低, 精度低)

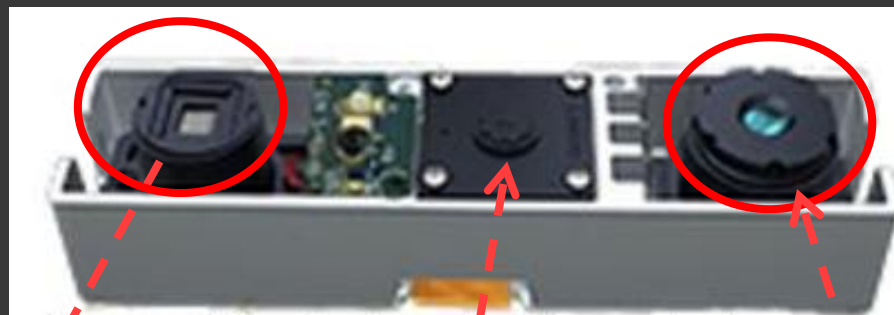
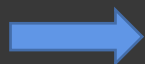


TOF (Time of Fly) 测距

(成本高, 精度高)

4.3 相机

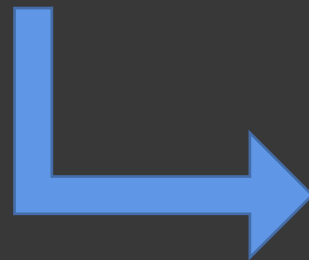
- 单目相机
- 双目相机
- RGB-D相机



普通RGB摄像头

红外光
接收头

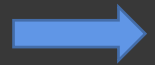
红外光
发射头



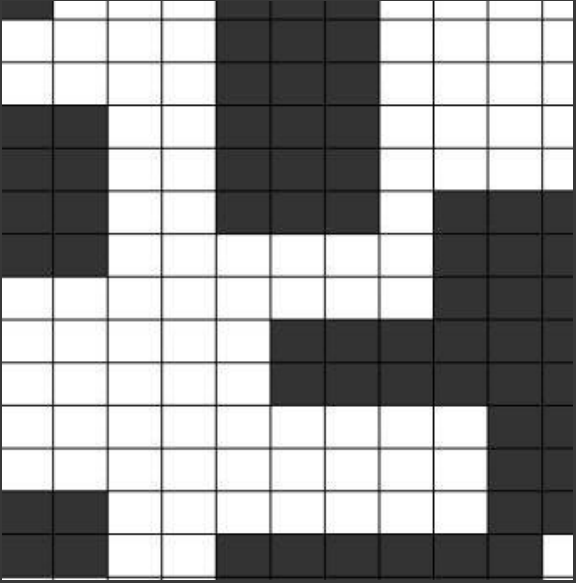
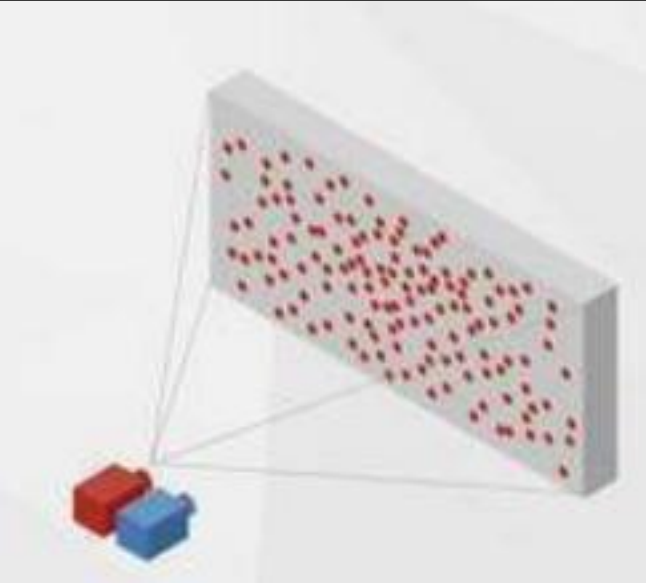
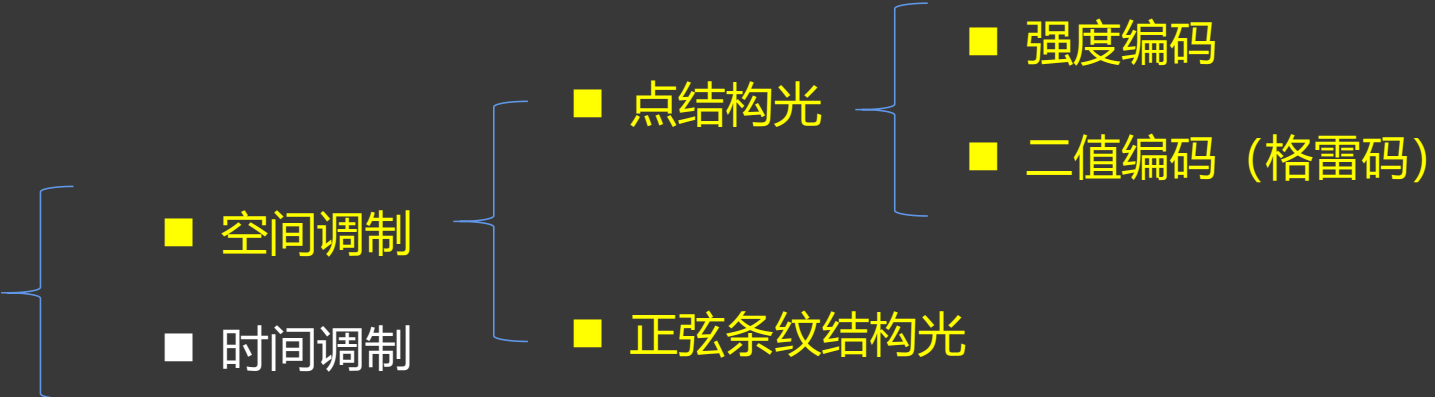
- 空间调制
- 时间调制

4.3 相机

- 单目相机
- 双目相机
- RGB-D相机



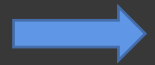
红外光
发射头



- 强度编码
- 二值编码 (格雷码)

4.3 相机

- 单目相机
- 双目相机
- RGB-D相机



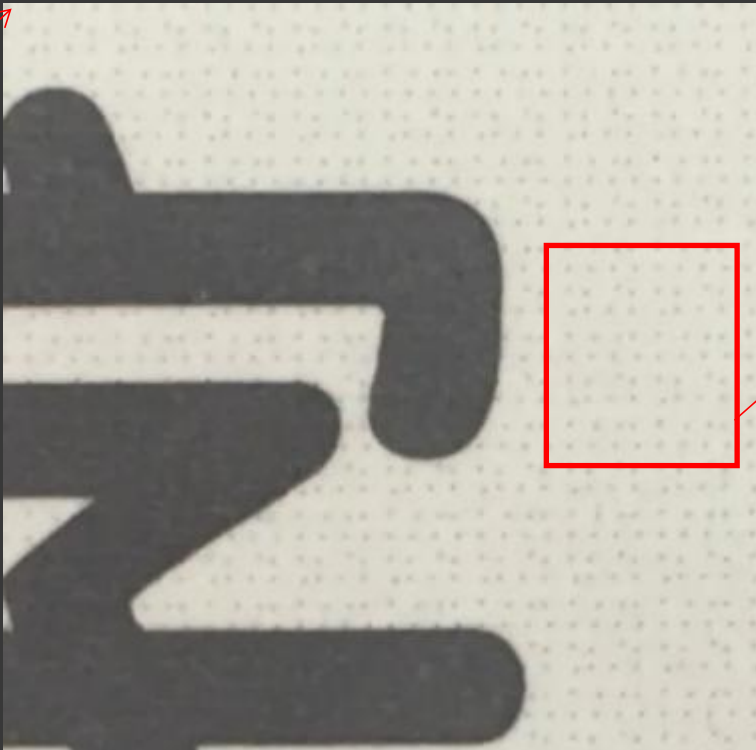
红外光
发射头

- 空间调制
- 时间调制

■ 点结构光

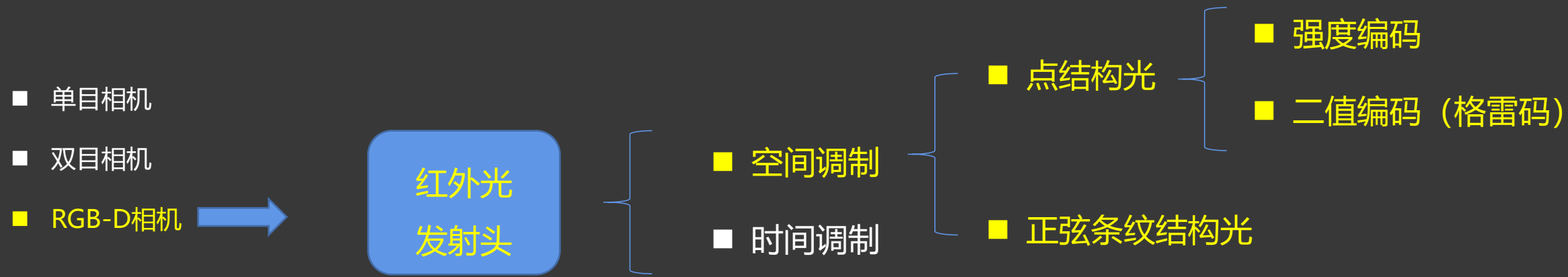
- 强度编码
- 二值编码 (格雷码)

■ 正弦条纹结构光

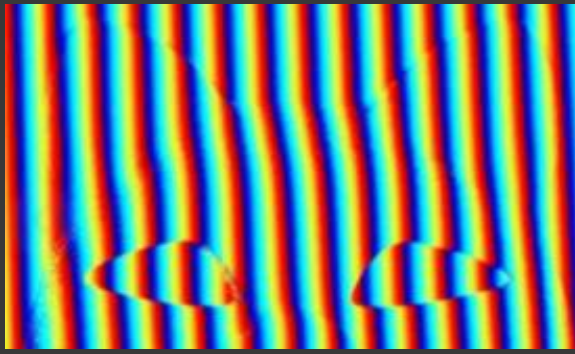


二值编码

4.3 相机



正弦条纹 (发射)



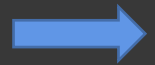
正弦条纹 (接收)



深度信息

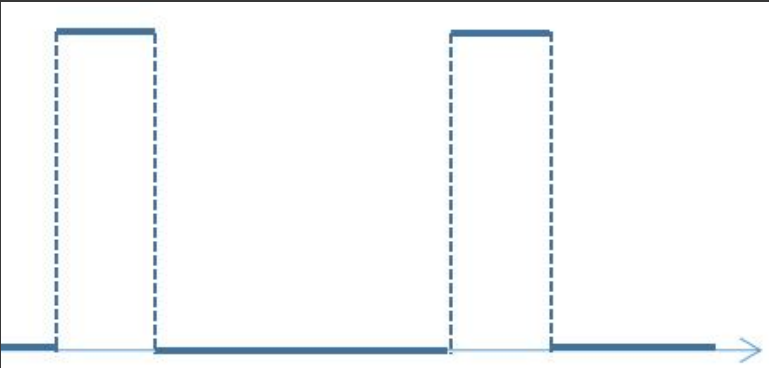
4.3 相机

- 单目相机
- 双目相机
- RGB-D相机

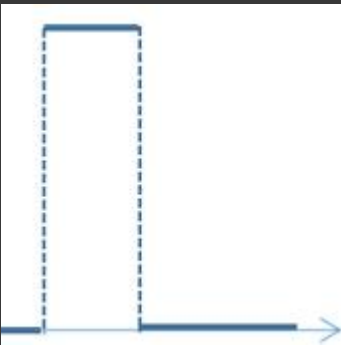


红外光
发射头

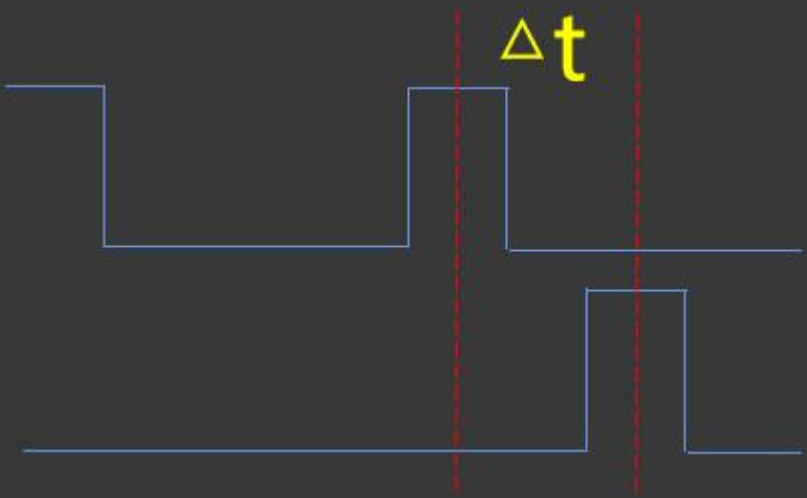
- 空间调制
- 时间调制
 - 调幅
 - 调频
 - 调相/相关码



激光脉冲（发射）

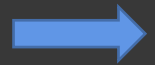


激光脉冲（接收）



4.3 相机

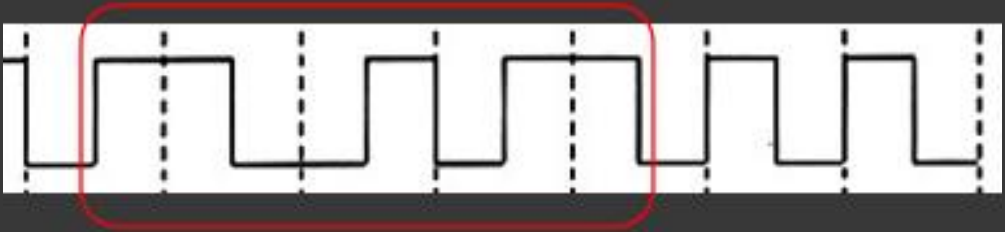
- 单目相机
- 双目相机
- RGB-D相机



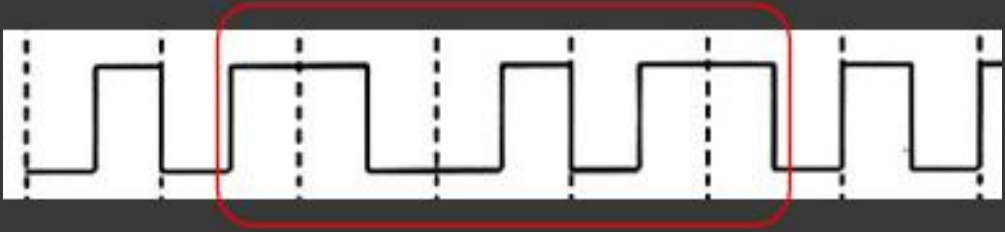
红外光
发射头



- 空间调制
- 时间调制
 - 调幅
 - 调频
 - 调相/相关码



激光脉冲（发射）

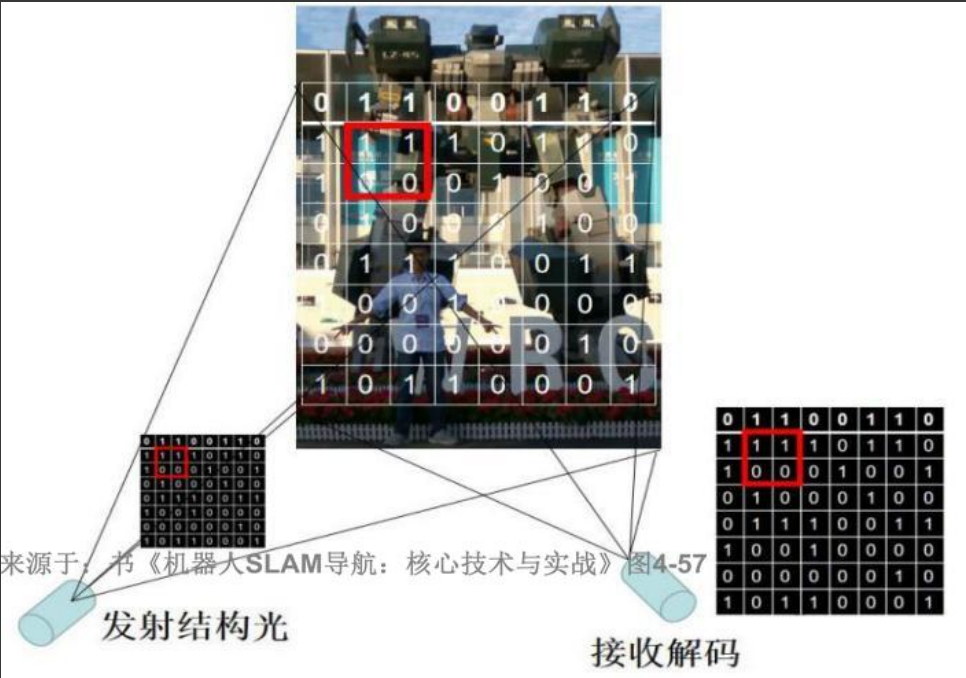


激光脉冲（接收）

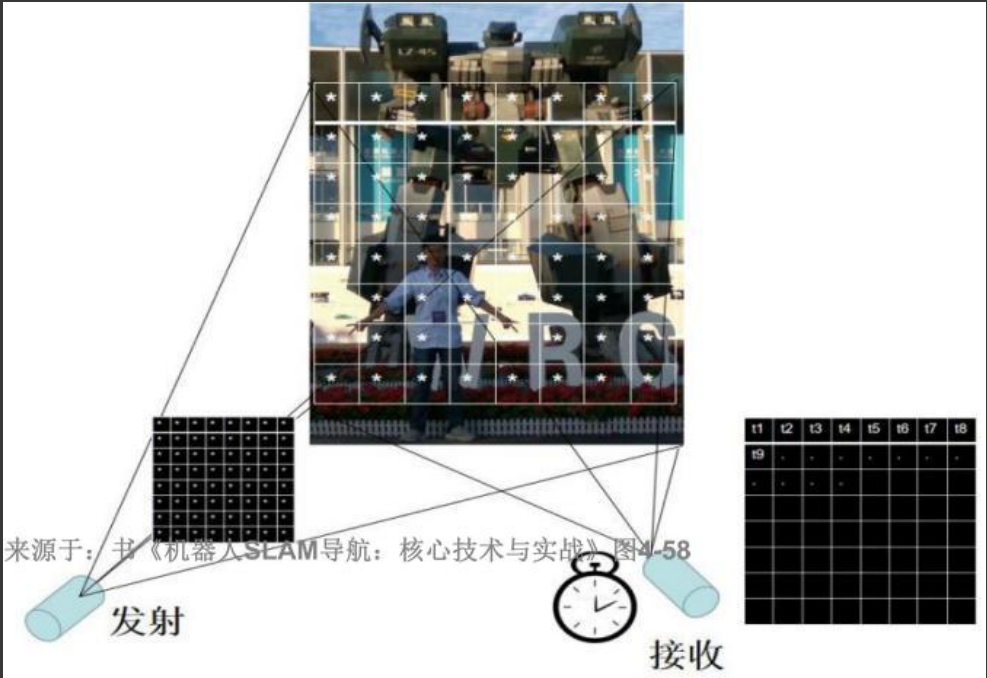
4.3 相机

- 单目相机
- 双目相机
- RGB-D相机 ➡

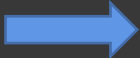
三角测距



TOF测距



4.3 相机

- 单目相机
- 双目相机
- RGB-D相机 

* 驱动RGB-D相机比较简单，厂家一般都提供配套的驱动程序，一般都是OpenNI接口

* RGB-D相机一般出厂都经过了标定，或者厂家会提供标定程序给用户，所以关于RGB-D相机的使用比较简单。

内容概要

4.1 惯性测量单元

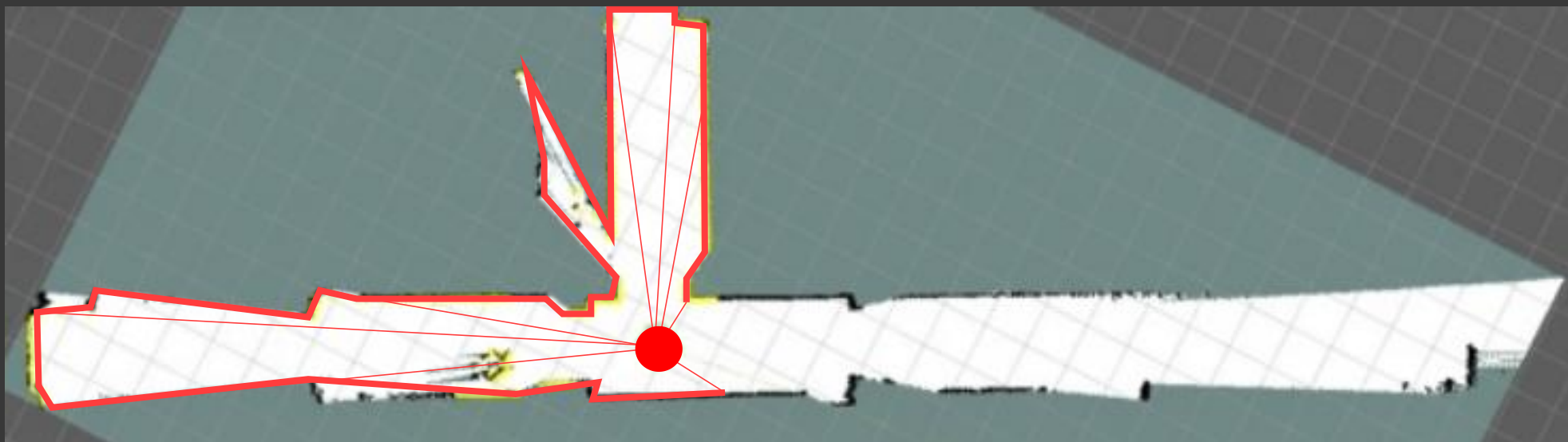
4.2 激光雷达

4.3 相机

4.4 带编码器的减速电机

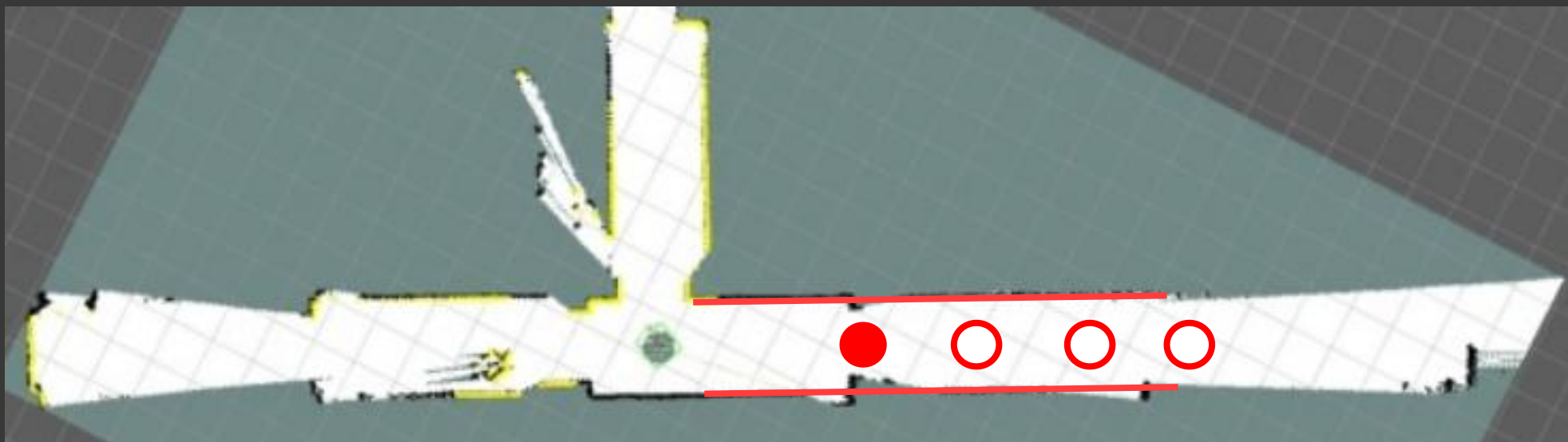
4.4 带编码器的减速电机

为什么要用编码器？



4.4 带编码器的减速电机

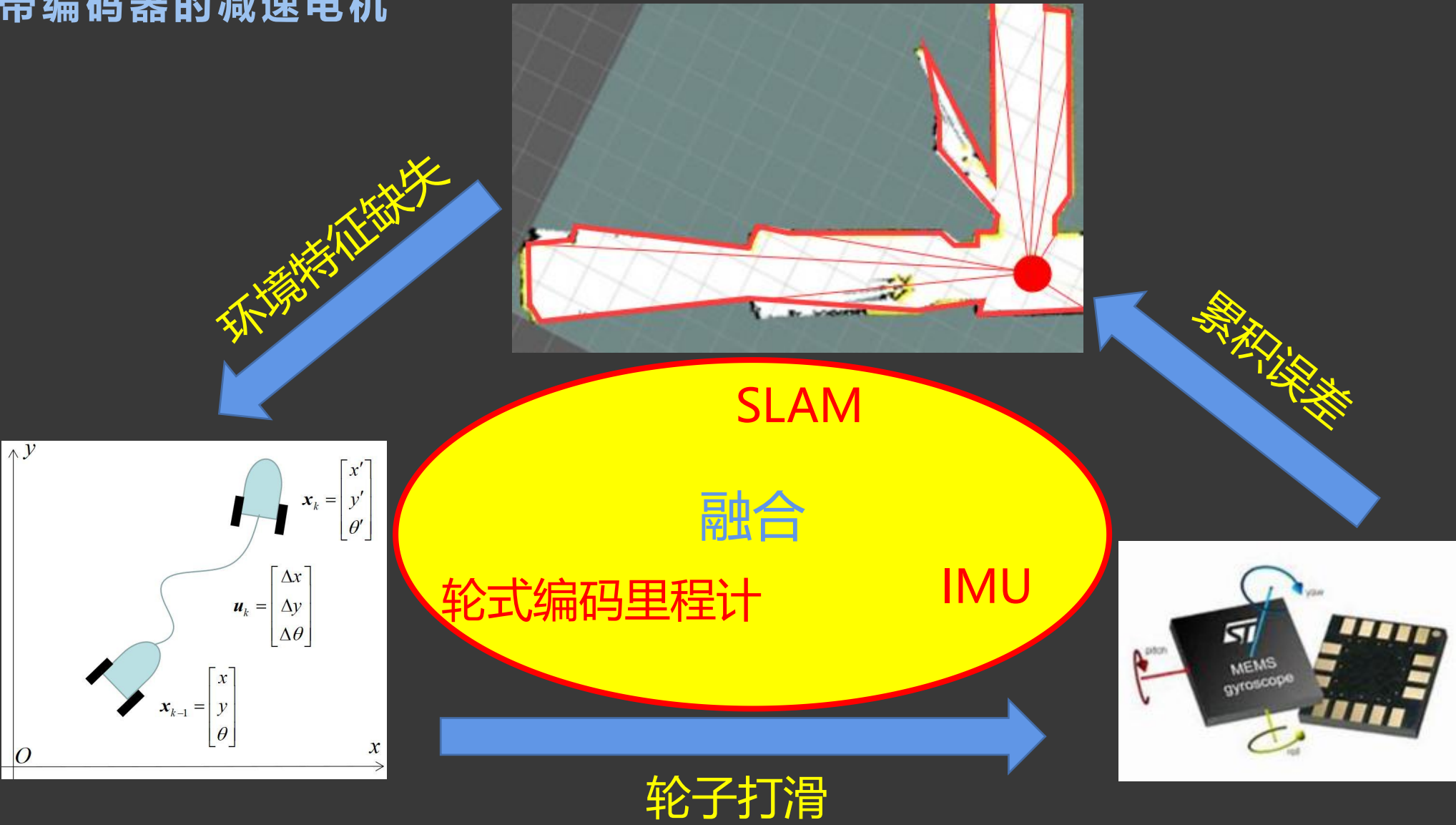
为什么要用编码器？



环境特征不明显时（比如走廊、空旷房间）

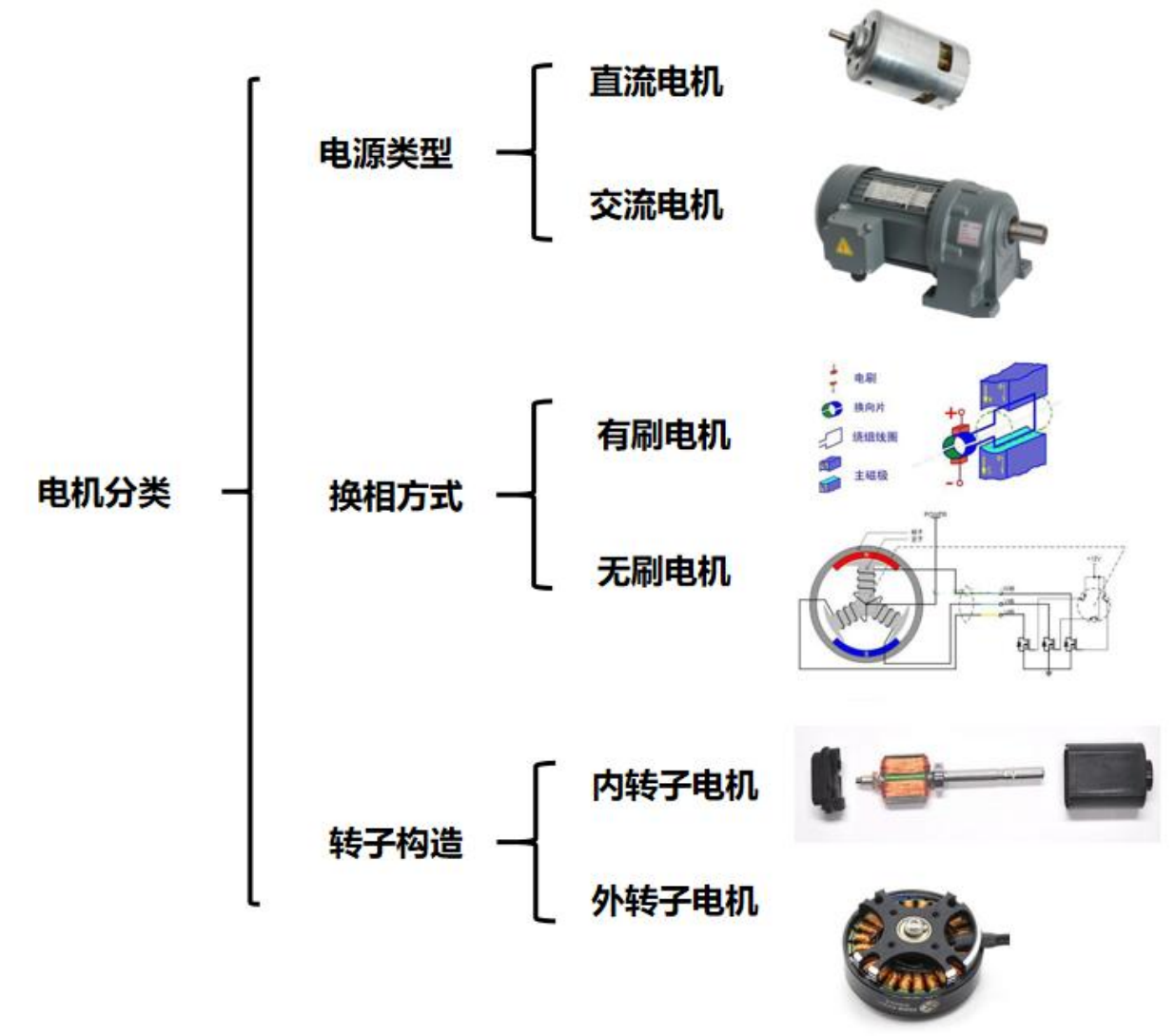
SLAM纯定位会失效

4.4 带编码器的减速电机



4.4 带编码器的减速电机

- 电机
- 电机驱动电路
- 电机控制主板
- 轮式里程计



4.4 带编码器的减速电机

电机？

减速电机？

编码减速电机？

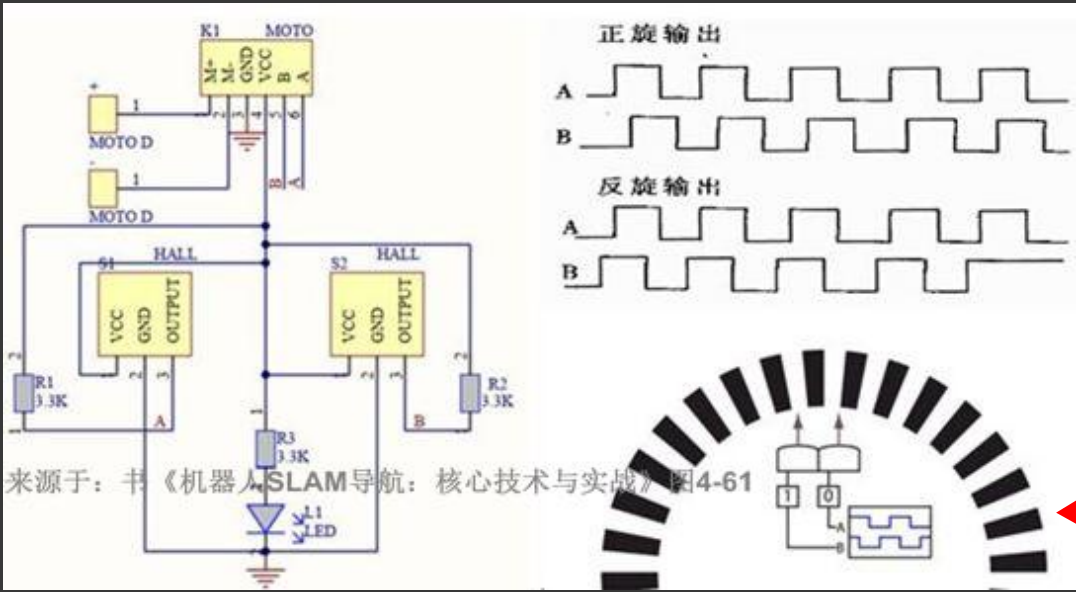
■ 电机



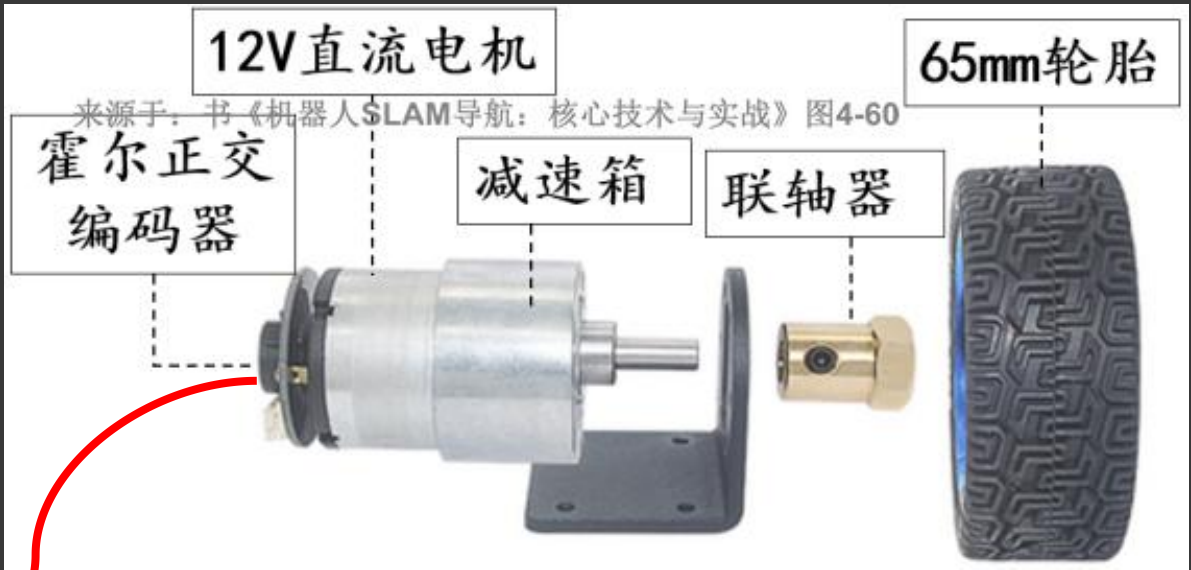
■ 电机驱动电路

■ 电机控制主板

■ 轮式里程计



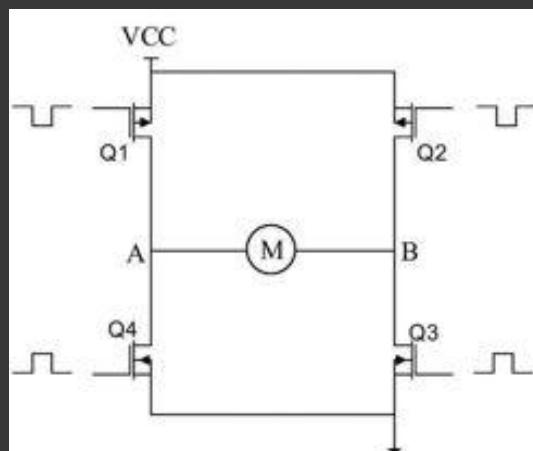
来源于：书《机器人SLAM导航：核心技术与实战》图4-61



来源于：书《机器人SLAM导航：核心技术与实战》图4-60

4.4 带编码器的减速电机

- 电机
 - 电机驱动电路
 - 电机控制主板
 - 轮式里程计
- ➔
- 有刷驱动电路
 - 无刷驱动电路



H桥电路



L298N



TB6612FNG

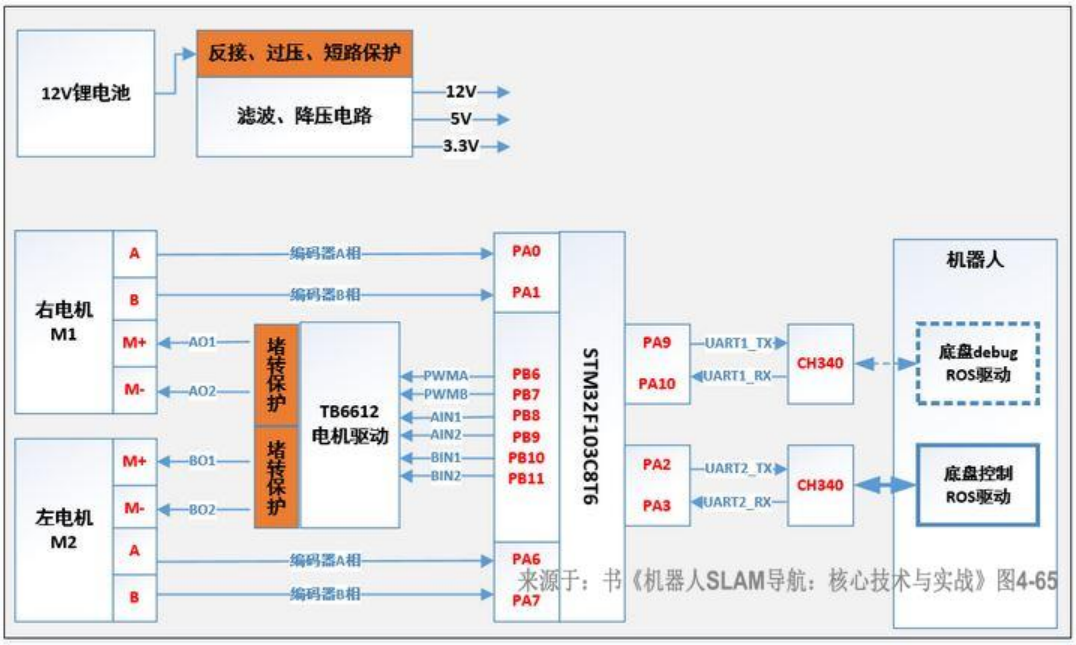


功率MOS管

来源于：书《机器人SLAM导航：核心技术与实战》图4-62

4.4 带编码器的减速电机

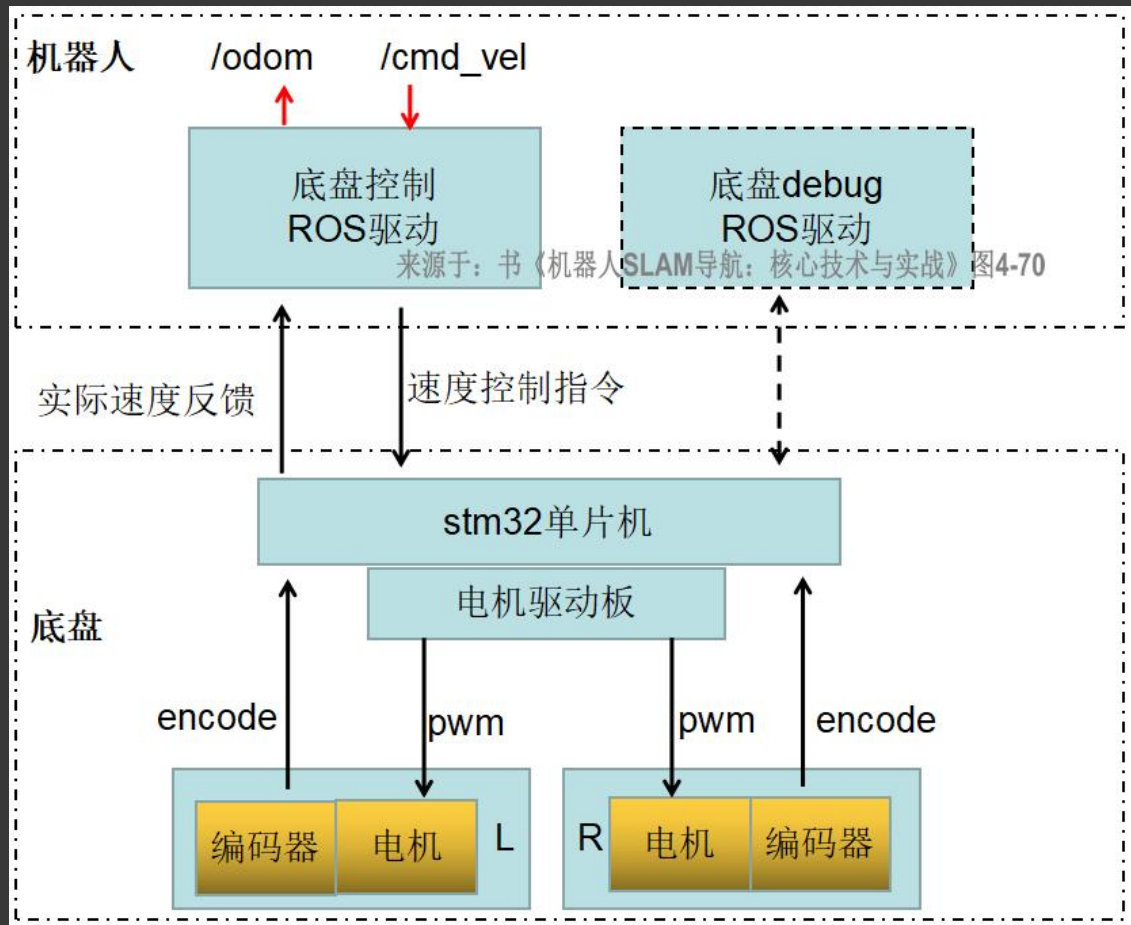
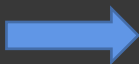
- 电机
 - 电机驱动电路
 - 电机控制主板
 - 轮式里程计
- ➔
- 上、下位机主板方案对比
 - PID控制
 - 通信协议



电机控制集成电路框图

4.4 带编码器的减速电机

- 电机
- 电机驱动电路
- 电机控制主板
- 轮式里程计



- 上位机ROS驱动中，一方面按照通信协议与下位机交换数据；另一方面向ROS中发布里程计话题/odom和订阅速度控制话题/cmd_vel。通信协议数据的收发解析相对容易，根据下位机传回来的编码值求解里程计是核心与难点。
- 里程计的解算，跟机器人底盘的结构和具体驱动方式有关，所以里程计解算将放在第6章来进行系统讨论。

- 例程源码下载： https://github.com/xiihoo/Books_Robot_SLAM_Navigation
- 课件PPT下载： www.xiihoo.com

敬请关注,长期更新...

下集预告