# **同时定位与地图构建（SLAM）：一份全面的理论阐述**

## **第一部分：SLAM的理论基础**

### **第一章：同时定位与地图构建（SLAM）导论**

#### **1.1 根本问题：“我在哪里？”与“世界是什么样子？”**

在自主机器人的广阔领域中，一个核心且根本性的问题驱动了数十年的研究：一个智能体（如机器人、无人机或自动驾驶汽车）如何在未知环境中，仅依靠自身传感器，既能确定自己的位置，又能构建出周围环境的地图？这个问题可以分解为两个紧密耦合的子问题：定位（Localization）和地图构建（Mapping）1。同时解决这两个问题，便是“同时定位与地图构建”（Simultaneous Localization and Mapping, SLAM）的核心任务。

SLAM问题的起源可以追溯到20世纪80年代中期，当时的机器人社区正努力让机器人在没有全球定位系统（GPS）的室内或水下等环境中实现自主导航 3。这一需求催生了一个被广泛认可的挑战，通常被形象地描述为“鸡生蛋还是蛋生鸡”的困境：为了精确地定位，机器人需要一张准确的地图；而为了构建一张准确的地图，机器人又需要知道自己的精确位置.1 由于机器人初始时既没有地图，也不知道自己的位置，这两个问题相互依赖，形成了一个看似无法破解的循环。

这种固有的循环依赖性揭示了SLAM问题的本质。它并非简单地将定位和建图两个独立问题叠加，而是一个统一的、高维的联合估计问题。当机器人使用其不确定的位姿估计来更新地图中某个地标（Landmark）的位置时，该地标的位置不确定性就与机器人的位姿不确定性产生了关联。反之，当机器人后续利用这个位置不确定的地标来重新定位自己时，其新的位姿估计的不确定性又会受到地标不确定性的影响。这个过程不断迭代，导致系统中所有状态变量——包括机器人整个运动轨迹上的所有位姿和地图中所有地标的位置——都变得高度相关 5。这种无处不在的概率相关性是SLAM问题的核心特征，它决定了必须采用联合概率推断的框架来寻求最优解，任何试图将定位与建图完全解耦的简单方法都难以获得理想的结果。

#### **1.2 SLAM的重要性与影响**

在过去的几十年里，SLAM技术取得了飞速发展，并被公认为实现真正自主性的基石技术之一 6。它使得机器能够在没有先验知识的环境中，通过增量式的方式构建环境模型，并利用该模型进行自我定位 4。SLAM的应用领域极其广泛，涵盖了室内机器人、自动驾驶汽车、无人机、水下航行器、增强现实（AR）和虚拟现实（VR）等多个前沿领域 3。无论是扫地机器人在家庭环境中规划清扫路径，还是火星车在遥远的星球上探索未知地形，SLAM都扮演着不可或令的关键角色。

#### **1.3 SLAM流程概念概览**

一个典型的现代SLAM系统通常被划分为两个主要部分：前端（Front-end）和后端（Back-end）7。

* **前端**：也称为视觉里程计（Visual Odometry, VO）或激光雷达里程计（LiDAR Odometry），主要负责处理原始传感器数据。它的任务是从连续的传感器读数（如图像序列或激光雷达扫描）中估计出机器人局部的、短期的运动，并进行数据关联，即判断当前的观测数据对应于地图中的哪个部分。前端为后端提供一个初始的位姿估计。
* **后端**：负责对前端输出的带有噪声和累积误差的位姿估计进行优化。后端维护一个全局的地图和机器人轨迹，通常以图（Graph）的形式表示。当机器人重新识别出一个曾经访问过的地方时，就会产生一个所谓的“回环闭合”（Loop Closure）5。回环闭合为后端优化提供了一个强大的约束，能够显著消除长时间累积的漂移，从而获得全局一致的地图和轨迹。

整个SLAM过程是一个增量式的循环：机器人移动，传感器感知环境，前端估计局部运动，后端优化全局地图和轨迹。正是通过前端和后端的协同工作，以及关键的回环检测机制，SLAM系统才能够在未知环境中实现鲁棒而精确的定位与建图。

### **第二章：SLAM的概率论表述**

为了在充满不确定性的现实世界中解决SLAM问题，概率论提供了一个强大而严谨的数学框架。几乎所有现代SLAM算法都建立在概率贝叶斯推断的基础之上。

#### **2.1 数学预备知识与符号定义**

在深入探讨概率模型之前，我们首先统一定义SLAM问题中的核心变量和符号 4：

* **状态向量 xk​**：表示机器人在离散时间点 k 的状态，通常指其位姿（Pose），即位置和姿态。在二维（2D）平面中，位姿可以表示为 xk​=[x,y,θ]T，其中 (x,y) 是坐标，θ 是航向角。在三维（3D）空间中，位姿通常用特殊欧几里得群 SE(3) 中的一个元素来表示，包含一个三维平移向量和一个三维旋转矩阵。
* **控制向量 uk​**：表示从时间点 k−1 到 k 之间施加给机器人的控制输入，或者由里程计（Odometry）测得的运动量。例如，轮式机器人的轮速编码器读数，或者无人机的电机指令。
* **地图 m**：表示环境的模型。在许多SLAM系统中，地图被参数化为一组静态地标（Landmarks）的集合，即 m={m1​,m2​,…,mn​}，其中 mj​ 是第 j 个地标的位置向量。
* **观测向量 zk​**：表示在时间点 k 机器人传感器获取的测量数据。例如，相机图像中观测到的地标像素坐标，或激光雷达测得的到地标的距离和角度。

此外，我们通常用大写字母表示变量的历史序列：

* X0:k​={x0​,x1​,…,xk​}：从初始时刻到 k 时刻的机器人位姿序列。
* U1:k​={u1​,u2​,…,uk​}：从时刻1到 k 的控制输入序列。
* Z1:k​={z1​,z2​,…,zk​}：从时刻1到 k 的观测数据序列。

#### **2.2 SLAM的贝叶斯框架**

在概率框架下，SLAM问题被表述为一个状态估计问题：根据所有可用的测量数据（控制输入和环境观测），计算机器人位姿和环境地图的后验概率分布（Posterior Probability Distribution）3。这个过程遵循贝叶斯滤波的递归思想，其核心由两个基本模型构成 5：

1. 运动模型（Motion Model）或状态转移模型（State Transition Model）: p(xk​∣xk−1​,uk​)  
   该模型描述了机器人状态随时间演化的规律。它给出了在已知上一时刻位姿 xk−1​ 和当前控制输入 uk​ 的条件下，当前位姿 xk​ 的概率分布。由于电机打滑、地面不平等因素，机器人的实际运动总存在不确定性，运动模型正是对这种不确定性的数学刻画。
2. 观测模型（Observation Model）或测量模型（Measurement Model）: p(zk​∣xk​,m)  
   该模型描述了传感器感知过程。它给出了在已知机器人当前位姿 xk​ 和环境地图 m 的条件下，获得特定观测数据 zk​ 的概率。由于传感器噪声、环境干扰等因素，测量过程也充满不确定性，观测模型就是对这种不确定性的量化。

这两个模型是所有概率SLAM算法的基石，它们共同定义了SLAM问题的动态贝叶斯网络（Dynamic Bayesian Network, DBN）结构 3。

#### **2.3 完整SLAM与在线SLAM**

基于上述概率框架，SLAM问题可以进一步细分为两种主要形式：完整SLAM（Full SLAM）和在线SLAM（Online SLAM）3。

* 完整SLAM（Full SLAM）：  
  完整SLAM的目标是估计整个机器人运动轨迹以及地图，即求解整个历史状态的联合后验概率分布。其数学表达式为：  
  p(x1:T​,m∣z1:T​,u1:T​,x0​)  
    
  这种形式也被称为平滑（Smoothing）问题。它利用所有时间段内的信息（包括未来的信息）来优化过去的每一个状态。例如，在时刻 T 发生的回环闭合，可以用来修正时刻 1 的位姿估计。因此，完整SLAM能够获得全局最优和最一致的解。图优化（Graph Optimization）等基于批处理的方法是解决完整SLAM问题的典型代表。
* 在线SLAM（Online SLAM）：  
  在线SLAM的目标是只估计当前时刻的机器人位姿以及地图，而将所有过去时刻的位姿都通过积分（或求和）的方式边缘化掉（Marginalized out）。其数学表达式为：  
  p(xT​,m∣z1:T​,u1:T​,x0​)=∫⋯∫p(x1:T​,m∣z1:T​,u1:T​,x0​)dx1​…dxT−1​  
    
  这种形式也被称为滤波（Filtering）问题。它是一个递归的过程，每当有新的数据到来，就在上一时刻的估计基础上进行更新。这种方法计算效率高，适合实时应用。卡尔曼滤波器（Kalman Filter）和粒子滤波器（Particle Filter）是解决在线SLAM问题的经典方法。

完整SLAM和在线SLAM之间的区别，实际上是**全局一致性**与**计算效率**之间的根本权衡，这一权衡深刻地影响了SLAM算法的演进历程。在线SLAM的边缘化操作是“单向”的：一旦过去的某个位姿 xk−1​ 被积分掉，它的估计值就无法再根据未来的信息（如回环闭合）进行修正。这个过去位姿的估计误差被“固化”到了当前状态的不确定性中，虽然计算高效，但牺牲了全局修正的能力。相比之下，完整SLAM保留了所有历史位姿 x1:T​，一个在 T 时刻的回环约束可以连接到早期的位姿 xj​，使得优化器能够调整从 j 到 T 的整段轨迹，从而实现全局一致性。在早期，由于计算能力的限制，完整SLAM的巨大计算量令人望而却步 3。直到高效的稀疏优化技术出现后，完整SLAM才变得实用，并因其卓越的全局一致性而成为现代SLAM的主流范式。

## **第二部分：经典方法：滤波思想**

在SLAM发展的早期，滤波方法因其递归更新的特性和相对较低的计算需求而占据主导地位。其中，扩展卡尔曼滤波器（Extended Kalman Filter, EKF）是第一个被成功应用于SLAM的算法，为整个领域奠定了理论基础。

### **第三章：扩展卡尔曼滤波器SLAM（EKF-SLAM）**

#### **3.1 卡尔曼滤波器简述**

卡尔曼滤波器是一种用于线性动态系统状态估计的优化递归算法。它通过一个“预测-校正”循环来工作 14：

* **预测（Prediction）**：根据系统的运动模型，预测下一时刻的状态及其不确定性。
* **校正（Correction）**：利用新的观测数据，修正预测的状态，从而获得更精确的估计。

卡尔曼滤波器的核心假设是系统模型（运动和观测）是线性的，并且所有噪声（过程噪声和测量噪声）都服从高斯分布。然而，SLAM中的运动和观测模型几乎总是非线性的（例如，涉及角度和三角函数）。为了将卡尔曼滤波器应用于非线性系统，\*\*扩展卡尔曼滤波器（EKF）\*\*应运而生。EKF的核心思想是在当前状态估计点对非线性函数进行一阶泰勒级数展开，用一个线性函数来局部近似非线性函数，然后应用标准卡尔曼滤波器的框架 14。

#### **3.2 EKF-SLAM：将EKF应用于SLAM问题**

EKF-SLAM将整个SLAM问题（机器人位姿和所有地标位置）打包进一个巨大的状态向量，并用一个协方差矩阵来描述所有这些变量的不确定性。

* 状态向量与协方差矩阵：  
  EKF-SLAM的状态向量 xt​ 包含了机器人的位姿 xR​ 和所有 N 个地标的位置 {m1​,…,mN​}。对于一个二维环境，状态向量的维度是 (3+2N) 14。xt​=T  
    
  与之对应，协方差矩阵 Pt​ 是一个巨大的 (3+2N)×(3+2N) 矩阵，它不仅包含了机器人自身位姿的不确定性（PRR​）和各地标位置的不确定性（PMM​），更重要的是，它还包含了机器人位姿与各地标位置之间的互协方差（PRM​）14。这个协方差矩阵是EKF-SLAM的核心，它显式地编码了SLAM问题中所有状态变量之间的相关性。
* **EKF-SLAM的循环过程**：
  1. **预测**：当机器人移动时，系统使用运动模型来预测机器人新的位姿。由于只有机器人移动而地标是静止的，所以在这个阶段，只有状态向量中与机器人相关的部分会更新，协方差矩阵中与机器人位姿不确定性相关的部分（PRR​ 和 PRM​）会增加，表示不确定性的增长 14。
  2. **观测与校正**：当机器人观测到一个已知的地标时，系统进入校正阶段。首先，计算预测的观测值与实际观测值之间的差异（称为新息，Innovation）。然后，计算卡尔曼增益（Kalman Gain），它决定了我们应该在多大程度上相信新的观测。最后，利用卡尔曼增益来更新整个状态向量和整个协方差矩阵。这个更新过程会减小系统的不确定性 14。
  3. **地标初始化**：当观测到一个新的、未知的地标时，需要将其加入到状态向量中，并相应地扩展协方差矩阵，为其赋予一个初始的不确定性 16。

#### **3.3 EKF-SLAM的“阿喀琉斯之踵”：复杂性与一致性**

尽管EKF-SLAM在理论上非常优雅，但它在实际应用中面临着两大致命缺陷：

* **计算复杂性**：EKF-SLAM最严重的问题是其计算复杂性。在每次校正步骤中，都需要更新整个 (3+2N)×(3+2N) 的协方差矩阵。这个操作的计算复杂度为 O(N2)，其中 N 是地标的数量 14。这意味着随着地图规模的增大，算法的计算量会呈二次方增长，使其对于包含数百个以上地标的大规模环境变得不切实际 14。
* **数据关联与一致性**：EKF-SLAM对数据关联（Data Association）的正确性极为敏感。数据关联是指正确地判断当前观测到的特征是地图中的哪个地标。一旦发生错误的数据关联，错误的约束会被引入系统，可能导致滤波器迅速发散，使得地图和定位结果完全崩溃 18。此外，EKF依赖于对非线性模型的一阶线性化。当系统非线性程度很高时（例如，机器人进行快速旋转），线性化近似的误差会很大，这同样可能导致滤波器发散，最终得到不一致的结果 14。

EKF-SLAM中稠密的协方差矩阵既是其力量的源泉，也是其致命的弱点。当机器人观测一个地标（如 L1​）时，它不仅更新了自身位姿和 L1​ 的估计，还会通过协方差矩阵中的相关项，间接更新地图中所有其他地标（L2​,L3​,…）的估计，即使这些地标在当前时刻并未被直接观测到 5。这种“一处更新，处处受益”的机制，使得地图的相对结构能够随着观测的增加而单调地变得更加精确 5，这是EKF-SLAM理论上的一大优势。然而，正是为了维持和更新这个包含了所有变量间相关性的稠密协方差矩阵，才导致了

O(N2) 的计算灾难。这种理论上的完美主义与现实中的计算限制之间的矛盾，直接催生了对更高效算法的探索，例如试图稀疏化信息矩阵的SEIF（Sparse Extended Information Filter），以及更具革命性的FastSLAM。

### **第四章：粒子滤波器SLAM与FastSLAM**

为了克服EKF-SLAM的局限性，特别是其对高斯假设的依赖和二次方的计算复杂度，研究者们转向了另一种强大的贝叶斯滤波技术——粒子滤波器，并由此发展出了著名的FastSLAM算法。

#### **4.1 用于定位的粒子滤波器**

粒子滤波器，也称为序贯重要性采样（Sequential Importance Sampling, SIS）滤波器，是一种非参数化的贝叶斯滤波方法 11。与EKF用单一的高斯分布来近似后验概率不同，粒子滤波器使用一组带权重的随机样本（称为“粒子”）来表示后验概率分布。每个粒子代表对系统状态的一个具体假设。这种表示方式的强大之处在于，它能够逼近任意形状的概率分布，包括多峰分布（multi-modal），这在EKF中是无法做到的。在机器人定位问题中，这种方法被称为蒙特卡洛定位（Monte Carlo Localization, MCL），其中每个粒子代表一个机器人可能位姿的假设。

#### **4.2 Rao-Blackwellization分解原理**

直接将粒子滤波器应用于完整的SLAM状态（机器人位姿和所有地标）是不可行的，因为状态空间的维度极高。随着地标数量的增加，为了有效覆盖整个状态空间，所需的粒子数量会呈指数级增长，这就是所谓的“维度灾难”。

**Rao-Blackwellization**定理为此提供了一个优雅的解决方案。该定理指出，如果一个高维联合概率分布中的变量可以被划分，使得一部分变量在给定另一部分变量的条件下变得相互独立，那么这个高维估计问题就可以被分解为一系列低维的、更易于处理的子问题 20。

对于SLAM问题，完整的后验概率可以被分解为：

p(x1:T​,m∣z1:T​,u1:T​)=p(m∣x1:T​,z1:T​)⋅p(x1:T​∣z1:T​,u1:T​)

这个公式的核心思想是：如果我们知道了机器人的确切轨迹 x1:T​，那么对每个地标 mj​ 的位置估计就变成了相互独立的子问题。因为在给定机器人轨迹后，对一个地标的观测只与该地标和机器人的位姿有关，而与其他地标无关 11。

#### **4.3 FastSLAM：一种分解式解决方案**

FastSLAM正是基于Rao-Blackwellization原理设计的算法，它巧妙地将SLAM问题分解 6：

* **机器人轨迹估计**：使用粒子滤波器来估计机器人的轨迹 x1:T​。系统维护 M 个粒子，每个粒子代表一条完整的路径假设。
* **地标位置估计**：对于每个粒子（即每条假设的轨迹），系统都维护一套独立的地标估计器。由于在给定轨迹的条件下各地标的估计是独立的，因此每个粒子都附带有 N 个小型的、独立的滤波器，通常是低维的EKF（例如，每个二维地标对应一个2x2的EKF）21。

**FastSLAM算法流程**如下：

1. **采样（Sampling）**：对于每个粒子，根据运动模型 p(xt​∣xt−1[i]​,ut​) 预测并采样一个新的位姿 xt[i]​。这是FastSLAM 1.0版本的提议分布（Proposal Distribution）11。
2. **测量更新（Measurement Update）**：对于每个粒子 i 和当前观测到的每个地标 j：
   * 使用观测数据 zt​ 来更新该粒子所携带的关于地标 j 的EKF。
   * 根据这次观测的似然度（likelihood），计算并更新该粒子 i 的重要性权重（Importance Weight）。权重反映了这条路径假设与真实观测的匹配程度。
3. **重采样（Resampling）**：根据所有粒子的重要性权重，对粒子集合进行重采样。权重高的粒子更有可能被复制，权重低的粒子则被淘汰。这个步骤使得粒子向高似然区域集中 11。

**算法复杂度**：FastSLAM的每个更新步骤的复杂度大约为 O(M⋅logN)（当使用树状结构存储地标时）或 O(M⋅N)（当使用简单列表时），其中 M 是粒子数量。这相比于EKF-SLAM的 O(N2) 是一个巨大的进步，使得处理大规模地图成为可能 20。

#### **4.4 FastSLAM 2.0及其局限性**

* **改进的提议分布**：FastSLAM 1.0的提议分布只使用了运动模型，忽略了当前的观测信息。如果传感器非常精确，这种提议可能会产生很多低权重的粒子。FastSLAM 2.0对此进行了改进，它将当前观测 zt​ 也纳入了提议分布的计算中，即从 p(xt​∣xt−1​,ut​,zt​) 中采样。这使得粒子分布更接近真实的后验分布，从而可以用更少的粒子达到更好的效果 11。
* **核心挑战**：
  + **粒子退化（Particle Depletion）**：重采样步骤虽然必要，但可能导致粒子多样性的丧失。如果所有粒子都坍缩到一个错误的假设上，滤波器将无法恢复，尤其是在大范围环境中 6。
  + **路径退化（Path Degeneracy）**：随着时间的推移，早期决策的错误会不断累积。由于每个粒子都代表一条完整的历史路径，如果早期的某个位姿选择不佳，这条路径的权重可能会持续降低，最终导致只有极少数路径拥有不可忽略的权重。算法“无法忘记过去”是其根本问题之一 11。

FastSLAM的出现代表了SLAM领域的一次重要思想转变，即从依赖解析高斯近似的EKF，转向了能够处理任意分布的非参数采样方法。它通过Rao-Blackwellization分解，成功地规避了在高维空间中使用粒子滤波器的“维度灾难”，为处理非线性、非高斯问题提供了一个比EKF更鲁棒的方案。然而，FastSLAM自身面临的粒子退化问题，以及为每个粒子维护一张独立地图所带来的巨大内存开销，促使研究者们继续探索新的方向。这最终引向了下一个伟大的范式——图优化，它能够在不进行采样的情况下，直接处理完整的轨迹。

## **第三部分：现代方法：图优化**

随着对SLAM问题结构理解的深入以及计算能力的提升，基于优化的方法，特别是图优化（Graph-based SLAM），逐渐取代滤波方法，成为现代SLAM系统的主流框架。它将SLAM问题转化为一个大规模的非线性最小二乘问题，并利用稀疏性高效求解。

### **第五章：基于图的SLAM表述**

#### **5.1 图的抽象**

图优化方法的核心思想是将SLAM问题抽象成一个图（Graph）结构。这个概念最早由Lu和Milios在1997年提出 3。

* **节点（Nodes / Vertices）**：图的节点代表了需要优化的状态变量。在最常见的形式中，节点代表机器人在不同时刻的位姿 xi​。在某些系统中，地图中的地标 lj​ 也可以作为节点存在于图中 1。
* **边（Edges / Constraints）**：图的边代表了状态变量之间的空间约束，这些约束来源于传感器的测量。每一条边连接两个节点，并编码了它们之间的相对关系 1。主要有以下几类边：
  + **里程计边（Odometry Edges）**：连接连续的两个位姿节点（例如 xi​ 和 xi+1​），由运动模型（如轮式里程计或IMU）的测量 ui​ 产生。
  + **观测边（Observation Edges）**：连接一个位姿节点 xi​ 和一个地标节点 lj​，由传感器对地标的观测 zij​ 产生。
  + **回环边（Loop Closure Edges）**：连接两个时间上不连续的位姿节点（例如 xi​ 和 xj​，其中 j≫i）。当机器人识别出它回到了一个之前访问过的地方时，就会添加这样一条边。这是修正累积漂移、保证全局地图一致性的最关键约束。

#### **5.2 从概率到误差函数**

图优化的目标是，找到所有节点（位姿和地标）的一个最优配置，使得这个配置与图中所有边所代表的测量约束的**一致性最大化** 3。

在假设测量噪声服从高斯分布的前提下，最大化所有观测的联合似然概率（Maximum Likelihood Estimation, MLE），等价于最小化所有测量误差的马氏距离（Mahalanobis Distance）的平方和 3。这个过程将概率推断问题转化为了一个非线性最小二乘优化问题。

* **误差函数 eij​(xi​,xj​)**：为图中的每一条边定义一个误差函数。它计算了真实的测量值 zij​ 与一个根据当前节点状态估计值 (xi​,xj​) 计算出的预测测量值 z^ij​(xi​,xj​) 之间的差异 1。eij​(xi​,xj​)=zij​−z^ij​(xi​,xj​)  
    
  例如，对于一条里程计边，z^ij​ 是从位姿 xi​ 变换到 xj​ 的相对位姿，而 zij​ 是里程计的实际读数。
* **信息矩阵 Ωij​**：每一个误差项都由一个信息矩阵进行加权。信息矩阵是测量协方差矩阵 Σij​ 的逆，即 Ωij​=Σij−1​。它的作用是给更精确（不确定性更小）的测量赋予更高的权重 1。
* **全局目标函数**：整个SLAM问题最终被表述为求解一个状态向量 x∗，该向量能够最小化所有加权误差的平方和，即最小化全局目标函数 F(x) 1：x∗=argxmin​F(x)=argxmin​(i,j)∈C∑​eij​(xi​,xj​)TΩij​eij​(xi​,xj​)  
    
  其中 C 是图中所有约束（边）的集合 27。

#### **5.3 位姿图与因子图**

* **位姿图（Pose Graph）**：是图优化SLAM的一种特定且常见的形式，其中图的节点只包含机器人的位姿。地标观测被用来在观测到同一地标的不同位姿之间直接创建约束边。这种方式将地标变量边缘化，降低了优化问题的规模。
* **因子图（Factor Graph）**：是一种更通用的图表示方法，它是一个二分图。一类节点是变量节点（Variable Nodes），代表位姿、地标等待优化的状态。另一类节点是因子节点（Factor Nodes），代表概率约束（如里程计约束、观测约束）。每个因子节点连接到它所约束的变量节点上。这种表示方法更清晰地揭示了问题的概率结构，是GTSAM等现代优化库所采用的核心表示 28。

图优化SLAM的一个深远影响在于它将整个系统清晰地解耦为**前端**和**后端**两个部分，这种模块化设计极大地促进了SLAM技术的发展和复用。前端负责处理与特定传感器相关的任务，如从图像中提取特征、匹配激光雷达扫描等，其核心任务是构建图——即生成节点和边 3。而后端则是一个通用的优化引擎，它接收前端构建的抽象图（节点、边、误差函数和信息矩阵的集合），并求解由此定义的非线性最小二乘问题 24。后端并不关心这些约束是来自相机、激光雷达还是GPS；它只负责找到最优的节点配置。这种解耦带来了巨大的软件工程优势：研究人员可以独立地改进前端（例如，开发更好的视觉特征）或后端（例如，设计更快的优化算法）。同时，它也使得多传感器融合变得异常简单——只需为不同类型的传感器定义新的边，然后将它们添加到同一个图中即可。这种模块化的思想是现代高性能SLAM系统（如ORB-SLAM3 30）取得巨大成功的关键原因之一。

### **第六章：求解非线性最小二乘问题**

将SLAM问题转化为图优化后，核心任务就变成了如何高效地求解这个大规模的非线性最小二乘问题。由于误差函数 eij​，特别是那些涉及三维旋转（即 SE(3) 群上的运算）的函数，是高度非线性的，因此不存在直接的闭式解 26。必须采用迭代优化的方法，从一个初始猜测（通常由里程计提供）开始，逐步逼近最优解 25。

#### **6.1 迭代优化：高斯-牛顿法与列文伯格-马夸尔特法**

迭代优化的核心思想是在当前估计值附近对非线性问题进行局部线性化。

* **线性化**：利用一阶泰勒级数展开，将非线性误差函数 f(xk​+Δx) 在当前估计点 xk​ 附近近似为一个线性函数：f(xk​+Δx)≈f(xk​)+J(xk​)Δx。其中，J 是误差函数关于状态变量的雅可比矩阵（Jacobian Matrix）25。
* 高斯-牛顿法（Gauss-Newton, GN）：  
  GN法通过求解一个线性方程组来计算每一步的增量 Δx。这个线性方程组被称为正规方程（Normal Equations） 26：(JTΩJ)Δx=−JTΩe  
    
  这里的 H=JTΩJ 是对真实海森矩阵（Hessian Matrix）的近似，它规避了计算复杂的二阶导数 32。  
  b=−JTΩe 是梯度相关的向量。求解这个线性方程组得到增量 Δx，然后更新状态：xk+1​=xk​+Δx，并重复此过程直至收敛。GN法在接近最优解时收敛速度很快，但如果初始估计较差，它可能不稳定甚至发散 25。
* 列文伯格-马夸尔特法（Levenberg-Marquardt, LM）：  
  LM法是一种更为鲁棒的“信赖域（Trust Region）”方法，它巧妙地结合了高斯-牛顿法和梯度下降法（Gradient Descent）31。LM法在正规方程中引入了一个阻尼因子  
  λ：  
  (JTΩJ+λI)Δx=−JTΩe
  + 当 λ 很小时，LM法的行为接近于高斯-牛顿法，收敛速度快。
  + 当 λ 很大时，LM法的行为接近于梯度下降法，步长较小，虽然收敛慢，但能保证朝着误差减小的方向前进，增加了稳定性。  
    LM算法会根据每一步迭代的效果（即总误差是否减小）来动态调整 λ 的值 25。这种自适应的特性使得LM法成为求解SLAM后端优化的标准和首选方法，因为它在鲁棒性和收敛速度之间取得了很好的平衡 31。

#### **6.2 效率的关键：稀疏性**

如果直接求解正规方程，需要计算并求逆一个稠密的 H 矩阵，其复杂度为 O(n3)（n 为状态向量总维度），这对于大规模SLAM是不可接受的。然而，SLAM问题具有一个至关重要的内在属性——**稀疏性**。

* **雅可比和海森矩阵的结构**：在SLAM中，任何一个测量（边）都只关联到少数几个状态变量（节点）。例如，一条连接位姿 xi​ 和 xi+1​ 的里程计边，其误差只与这两个位姿有关。因此，在总的雅可比矩阵 J 中，对应于这条边的行，只有在对应于 xi​ 和 xi+1​ 的列上才有非零元素，其余全为零。
* **稀疏性来源**：这种局部连接性导致雅可比矩阵 J 是一个高度稀疏的矩阵。因此，由 H≈JTJ 构成的海森矩阵（或称为信息矩阵）也同样是高度稀疏的 26。海森矩阵中的非零块  
  Hij​ 恰好对应于SLAM图中存在约束边的节点对 (i,j) 35。
* **利用稀疏性**：利用这种稀疏结构，我们可以使用高效的稀疏线性代数求解器（如稀疏乔列斯基分解，Sparse Cholesky Factorization）来求解线性系统 HΔx=b，而无需进行显式的矩阵求逆。这些求解器的计算复杂度远低于 O(n3)，更多地取决于图的拓扑结构而非节点总数 29。正是这一关键的技术突破，使得大规模图优化SLAM变得可行，并直接催生了g2o、Ceres Solver、GTSAM等一系列强大的优化库 31。

SLAM问题的物理结构（即传感器的局部感知特性）直接转化为了其数学结构（即海森矩阵的稀疏性）。一个传感器的测量范围是有限的，它只能建立局部联系。这种物理上的局部性，精确地映射到了优化问题中雅可比矩阵和海森矩阵的稀疏模式上。海森矩阵的非零元素布局，实际上就是SLAM图的邻接矩阵的结构。这种稀疏性并非近似，而是SLAM问题的一个内在的、精确的属性。对这一内在稀疏性的认识和利用，是图优化SLAM的“文艺复兴” 3，它将一个计算上看似无法处理的问题，转变为一个在现代计算平台上可以实时求解的问题，这也是图优化方法能够主宰现代SLAM领域的根本原因。

#### **6.3 增量式SLAM：iSAM与iSAM2**

对于在线应用，每次增加新的测量就从头重新优化整个图，效率仍然不高。为了解决这个问题，增量式平滑与建图（Incremental Smoothing and Mapping, iSAM）及其后续版本iSAM2被提了出来 24。这些方法的核心思想是，在每次增加新的节点和约束时，不重新计算整个线性系统，而是通过更新信息矩阵的因子分解（如QR分解或乔列斯基分解）来增量式地更新解。由于每次更新只涉及图的一小部分，这种方法可以实现高效的在线平滑。

#### **表1：主要SLAM范式对比**

为了清晰地总结前述内容，下表对三种主要的SLAM算法范式进行了比较，突出了它们在核心原理、性能和适用性上的关键差异。

| 特性 | EKF-SLAM | FastSLAM | 图优化SLAM (平滑) |
| --- | --- | --- | --- |
| **问题类型** | 在线SLAM (滤波) | 在线SLAM (滤波) | 完整SLAM (平滑) |
| **概率模型** | 递归贝叶斯滤波（高斯假设） | Rao-Blackwellized粒子滤波 | 最大后验概率(MAP)估计（通过非线性最小二乘） |
| **状态表示** | 单一状态向量和稠密协方差矩阵 | 带权重的粒子集，每个粒子有自己的地图 | 位姿和地标组成的图 |
| **处理非线性** | 一阶泰勒线性化（可能发散） | 通过采样本质上非线性 | 迭代非线性优化(GN/LM)，更鲁棒 |
| **处理多模态** | 否（单峰高斯分布） | 是（多个粒子代表多个假设） | 否（寻找单个局部最优解） |
| **计算复杂度** | 每步 O(N2) (N=地标数) | 每步 O(MlogN) (M=粒子数, N=地标数) | 因稀疏性接近轨迹长度的线性关系 |
| **核心优势** | 概念简单，能捕捉所有相关性 | 更好的可扩展性，能处理非高斯分布 | 最高的精度，全局一致性，模块化(前端/后端) |
| **核心劣势** | 对大地图不适用，对线性化/数据关联错误敏感 | 粒子退化，路径退化，需要大量粒子 | 单次更新计算成本高于滤波器，需要好的初值 |

## **第四部分：特定传感器的前端与挑战**

SLAM的前端负责将原始传感器数据转化为后端优化器可以处理的图结构（节点和边）。前端的设计与所使用的传感器类型密切相关，不同的传感器带来了不同的机遇和挑战。本部分将深入探讨视觉、激光雷达以及视觉-惯性组合这三种主流SLAM的前端技术。

### **第七章：视觉SLAM（Visual SLAM, VSLAM）**

视觉SLAM使用相机作为主要传感器。相机成本低、体积小、能提供丰富的环境信息，使其成为最受欢迎的SLAM传感器之一。

#### **7.1 相机作为传感器**

VSLAM的基础是相机模型，最常用的是针孔相机模型。该模型描述了三维世界中的一个点 P 如何通过相机光心投影到二维图像平面上的像素点 p。这个投影关系可以用一个相机矩阵来表示：p=KP，其中 K 是相机内参矩阵，

是相机外参，即相机在世界坐标系中的位姿[38]。VSLAM的核心挑战就是从一系列二维图像中，反向推断出相机自身的运动（即

的变化）和三维场景的结构（即 P 的位置）。

#### **7.2 单目VSLAM的尺度不确定性**

使用单个相机（Monocular Camera）进行SLAM时，会遇到一个固有的、无法解决的问题——**尺度不确定性**（Scale Ambiguity）39。直观地说，一个近处的小物体和一个远处的大物体在相机图像中可能看起来完全一样。从数学上讲，如果我们将真实的相机轨迹和地图中的所有三维点坐标同时乘以一个任意的缩放因子

s，它们在图像上的投影结果是完全相同的。这意味着单目VSLAM只能恢复出场景的相对结构，而无法确定其绝对物理尺寸。这种不确定性导致单目SLAM的漂移具有7个自由度（DoF）：3个平移、3个旋转，以及1个尺度 41。

解决尺度不确定性通常有以下几种方法：

* **利用已知尺寸的物体**：如果在场景中观测到一个尺寸已知的物体（如一张A4纸），就可以此为参照物来确定全局尺度 39。
* **多传感器融合**：与能够提供绝对尺度信息的传感器融合，例如惯性测量单元（IMU）或GPS。IMU可以测量重力加速度，从而提供一个绝对的尺度参考 40。这是目前最主流和最有效的方法。
* **利用特殊相机**：某些特殊相机，如双像素（Dual-pixel）传感器，可以利用其独特的成像原理来估计绝对尺度 43。

#### **7.3 立体与RGB-D SLAM：引入深度信息**

为了直接克服尺度不确定性，可以使用能够提供深度信息的相机。

* 立体视觉SLAM（Stereo VSLAM）：  
  立体相机由两个水平放置、基线（baseline）距离 b 已知的相机组成。通过三角测量原理，可以从左右两幅图像中同一个点的位置差异（称为视差，disparity, d）计算出该点的深度 Z。其基本关系为 Z=df⋅b​，其中 f 是相机焦距 44。立体SLAM的关键技术挑战在于  
  **对应点匹配问题**（Correspondence Problem），即准确地找到左右图像中的同名点。通过**图像校正**（Rectification）技术，可以将这个二维搜索问题简化为在同一水平线（极线）上的一维搜索 45。由于能够直接获取深度，立体SLAM从一开始就能恢复出具有真实物理尺度的地图。
* RGB-D SLAM：  
  RGB-D相机（如Kinect、RealSense等）能主动发射红外光，直接测量并输出每个像素点的深度图（Depth Map），同时提供彩色的RGB图像 48。这极大地简化了VSLAM问题，因为每一帧图像都可以直接被转换成一个三维点云 38。因此，RGB-D SLAM的前端任务通常转化为  
  **连续三维点云的配准**（Registration）问题 50。这可以通过迭代最近点（ICP）算法，或者通过匹配二维图像特征并利用其已知的深度信息来求解相机运动 50。

#### **7.4 VSLAM前端范式：间接法 vs. 直接法**

VSLAM的前端主要分为两大技术流派：间接法（或称特征点法）和直接法。

* 间接法（Indirect / Feature-Based Methods）：  
  这是传统且非常稳健的方法，其处理流程为：提取特征点 → 计算描述子 → 特征匹配 → 位姿估计 52。
  + **核心特征：ORB**：在现代实时VSLAM系统（如ORB-SLAM）中，ORB（Oriented FAST and Rotated BRIEF）特征是核心 30。
    - **Oriented FAST**：使用FAST算法来快速检测角点（Keypoints）54。为了赋予特征点旋转不变性，它采用\*\*灰度质心法（Intensity Centroid）\*\*来计算每个角点的主方向。该方法计算角点邻域图像块的质心，从角点指向质心的向量便定义了该特征点的方向 57。
    - **Rotated BRIEF**：BRIEF是一种二进制描述子（Descriptor），它通过比较特征点邻域内一系列随机点对的灰度大小来生成一个二进制串，计算速度极快但本身不具备旋转不变性 54。ORB通过将BRIEF的采样模式（即那些点对的位置）根据Oriented FAST计算出的主方向进行旋转，从而“校正”了描述子，使其具备了旋转不变性，这就是所谓的“Steered BRIEF” 57。
  + **位姿估计**：通过将当前帧的2D特征点与地图中已有的3D地标点进行匹配，建立2D-3D对应关系。然后，利用这些对应关系求解\*\*PnP（Perspective-n-Point）\*\*问题来计算相机位姿。这个过程通常在一个RANSAC（Random Sample Consensus）循环中进行，以剔除错误的匹配 51。
  + **优缺点**：由于使用了具有不变性的特征描述子，间接法对光照变化和视角变化具有很强的鲁棒性。同时，因为它只处理稀疏的特征点，计算效率较高。其主要缺点是在纹理稀疏或重复的场景中，由于无法提取到足够或可区分的特征点，算法容易失败 60。
* 直接法（Direct Methods）：  
  直接法跳过了特征提取和匹配的步骤，直接在原始的像素灰度值上进行操作。其核心思想是，基于灰度不变假设（Brightness Constancy Assumption），最小化前后两帧图像之间对应像素的光度误差（Photometric Error）62。
  + **光度误差**：其数学形式为 E(ξ)=∑i​(Iref​(pi​)−I(ω(pi​,Dref​(pi​),ξ)))2。其中，Iref​ 是参考帧的图像，I 是当前帧的图像，ω 是一个变换函数，它根据相机位姿 ξ 和参考帧中像素 pi​ 的深度 Dref​，将 pi​ 投影到当前帧，并计算其灰度值。优化的目标就是找到最优的位姿 ξ 来最小化这个误差平方和 60。
  + **密度变体**：
    - **稠密法（Dense）**：如DTAM，使用图像中的所有像素进行优化。能够构建非常稠密的地图，但计算量巨大，通常需要GPU加速 64。
    - **半稠密法（Semi-dense）**：如LSD-SLAM，只选择图像中梯度（即灰度变化）较明显的像素（通常是边缘）进行优化，因为这些像素对光度误差的贡献最大 63。
    - **稀疏法（Sparse）**：如DSO，选择稀疏的像素点，但仍然是直接优化光度误差，而不计算描述子 64。
  + **优缺点**：能够利用图像中的所有信息（而不仅仅是角点），因此在纹理丰富的场景中可以获得高精度的位姿估计和更稠密的地图。但其致命弱点是严重依赖灰度不变假设，对光照变化、相机自动曝光等非常敏感 63。
* 半直接法（Semi-Direct Methods）：  
  如SVO，是一种混合方法，它试图结合两者的优点。它像间接法一样提取特征点，但随后像直接法一样，通过优化特征点周围小图像块（Patch）的光度误差来跟踪这些点，而不计算描述子 64。

VSLAM前端的发展历程，实际上反映了在**信息抽象**与**数据保真度**之间的一场根本性的权衡。间接法选择了高度抽象，将复杂的图像信息提炼为一小组具有不变性的特征点。这样做的好处是获得了对光照等干扰因素的鲁棒性，代价是丢弃了图像中绝大部分的信息，在无特征区域无能为力。而直接法则选择了最高的数据保真度，试图利用每一个像素的信息。这样做的好处是能够获得更稠密的几何信息和更高的精度，代价是必须依赖一个在现实世界中常常被打破的理想假设（灰度不变）。半直接法等混合方法的出现，正是为了在这两者之间寻找一个最佳的平衡点。因此，不存在“最好”的VSLAM前端，方法的选择高度依赖于具体的应用场景和环境条件。对于需要在光照多变的户外环境中进行长期导航的任务，间接法通常更为鲁棒；而对于在受控环境中进行高精度三维重建的任务，直接法则可能更具优势。

#### **表2：VSLAM前端范式对比分析**

| 特性 | 间接法 (基于特征) | 直接法 | 半直接法 |
| --- | --- | --- | --- |
| **示例系统** | ORB-SLAM | LSD-SLAM, DSO | SVO |
| **核心原理** | 最小化匹配特征点的几何重投影误差 | 最小化像素灰度的光度误差 | 最小化特征点图像块的光度误差 |
| **使用数据** | 稀疏的关键点集 (如ORB特征) | 所有像素 (稠密), 或高梯度像素 (半稠密) | 稀疏的关键点集 |
| **核心优势** | 因描述子不变性，对光照变化和宽基线匹配鲁棒性强 | 在纹理丰富区域精度高、可稠密重建；无特征提取步骤 | 旨在结合直接法的速度和特征点的鲁棒性 |
| **核心劣势** | 在低纹理区域失效；丢弃非特征区域信息 | 对光度变化敏感 (光照、自动曝光)；需要良好初值 | 鲁棒性不及纯间接法；稠密性不及纯直接法 |
| **地图密度** | 稀疏 | 稠密或半稠密 | 稀疏 |
| **计算成本** | 主要消耗在特征提取与匹配 | 主要消耗在对大量像素的优化上 | 通常比前两者都低，因其直接跟踪少量点 |

### **第八章：激光雷达SLAM（LiDAR SLAM）**

激光雷达（LiDAR）是另一种在SLAM中广泛应用的传感器。它通过发射激光束并测量其返回时间来直接获取环境的三维几何信息，生成精确、稠密的点云数据，并且不受环境光照变化的影响 67。

#### **8.1 基于配准的里程计：迭代最近点（ICP）算法**

LiDAR SLAM前端的核心任务是**点云配准**（Point Cloud Registration），即对齐连续两帧扫描得到的点云，从而计算出传感器的运动。\*\*迭代最近点（Iterative Closest Point, ICP）\*\*算法是解决这一问题的经典方法 68。

* **核心问题**：给定一个源点云 Q 和一个目标点云 P，寻找一个最优的刚体变换 T（包含旋转 R 和平移 t），使得变换后的源点云 T(Q) 与目标点云 P 对齐得最好。
* **ICP算法流程** 68：
  1. **寻找对应点**：对于源点云中的每一个点，在目标点云中寻找其最近的点作为对应点。为了提高搜索效率，通常会对目标点云构建k-d树等空间索引结构。
  2. **估计变换**：基于找到的对应点对，最小化一个误差度量来求解最优的变换 T。
  3. **应用变换**：将计算出的变换 T 应用于源点云。
  4. **迭代**：重复以上步骤，直到满足收敛条件（例如，变换量足够小或达到最大迭代次数）。
* **ICP的变体及其目标函数**：
  + 点对点ICP（Point-to-Point ICP）：这是最基础的ICP变体，其目标函数是最小化对应点之间的欧氏距离平方和：  
    E(T)=i∑​∥pi​−Tqi​∥2  
      
    其中 (pi​,qi​) 是一对对应点。这个问题有一个基于奇异值分解（SVD）的闭式解 69。
  + 点对面ICP（Point-to-Plane ICP）：这种变体通常比点对点ICP收敛更快，精度也更高，尤其是在结构化的环境中。它的目标函数是最小化每个源点到其对应目标点所在平面的距离平方和：  
    E(T)=i∑​((pi​−Tqi​)⋅ni​)2  
      
    其中 ni​ 是目标点 pi​ 处的法向量。这个目标函数允许点沿着平面“滑动”，而不是强制它们精确地落在对应点上，这更符合物理实际。求解这个问题需要迭代线性化，但通常收敛性更好 68。

#### **8.2 基于特征的LiDAR SLAM（如LOAM）**

直接在原始的稠密点云上运行ICP算法计算量巨大，且容易陷入局部最优。因此，与VSLAM的发展类似，基于特征的方法也被引入到LiDAR SLAM中，其中最著名的就是LOAM（Lidar Odometry and Mapping）及其变体 67。

* **特征提取**：LOAM不使用所有点，而是从点云中提取具有代表性的几何特征。它根据局部点的曲率来区分两类特征：
  + **边缘点（Edge Points）**：曲率大的点，通常位于物体的尖锐边缘或角落。
  + **平面点（Planar Points）**：曲率小的点，通常位于平坦的表面上。
* **扫描-地图匹配（Scan-to-Map Matching）**：与ICP的扫描-扫描（scan-to-scan）匹配不同，LOAM将当前帧提取的特征点与一个由过去多帧点云构建的局部地图进行匹配。这种扫描-地图的匹配方式利用了更多历史信息，因此比单次扫描间的匹配更为鲁棒和精确 72。
* **优化**：优化的目标是最小化特征点到其对应几何图元（边缘点到线，平面点到面）的距离。通过这种方式，可以稳健地求解出传感器的六自由度运动。

LiDAR SLAM从ICP到基于特征的方法的演进，与VSLAM从直接法到特征点法的发展轨迹惊人地相似。这反映了一个更深层次的原则：从处理原始、无结构的数据，转向利用更高级别的、抽象的几何图元，是提升SLAM系统鲁棒性和效率的共同途径。点对点ICP是最直接的匹配方式，但它对对应点关系的假设过于严格。点对面ICP则进了一步，它理解了“平面”这一几何概念，使得匹配更加灵活。而像LOAM这样的特征法，则将这种抽象推向了极致，它主动地去寻找场景中最具信息量和最稳定的几何结构（边缘和平面），并基于这些结构来构建约束。一个点到一个平面的约束，远比一个点到一个点的约束要更强、更不易产生歧义。这种利用环境结构信息来构建更强约束的思想，是现代高性能SLAM算法的精髓所在。

### **第九章：视觉-惯性SLAM（Visual-Inertial SLAM, VINS）**

为了弥补单一传感器的不足，多传感器融合成为SLAM领域的重要发展方向。其中，相机与惯性测量单元（IMU）的结合，即视觉-惯性SLAM（VINS），被证明是一种极为强大和实用的组合。

#### **9.1 融合的力量：相机与IMU**

* **惯性测量单元（IMU）**：IMU通常包含一个三轴陀螺仪（Gyroscope）和一个三轴加速度计（Accelerometer），能够以非常高的频率（通常大于100Hz）测量设备的角速度和加速度 42。
* **互补优势** 42：
  + **相机**：在光照良好、纹理丰富的环境中，能够提供精确的定位信息。但其频率较低，在快速运动时容易产生运动模糊，在无纹理区域会跟踪失败，并且单目相机存在尺度模糊问题。
  + **IMU**：能够提供高频的运动信息，不受外界环境（如光照、纹理）影响，并且能直接测量重力方向和提供运动的绝对尺度。但其主要缺点是，由于需要对带有噪声的加速度进行二次积分来得到位置，其误差会随时间快速累积，导致严重漂移。
* **融合目标**：将两者结合，可以取长补短，实现高频率、高精度、具有真实尺度且在各种环境下都非常鲁棒的位姿估计 42。IMU可以在相机失效的短暂时间内（如快速旋转或进入黑暗区域）提供运动估计，而相机的观测则可以持续地校正IMU的漂移，并估计其零偏（bias）。

#### **9.2 耦合策略：松耦合与紧耦合**

根据视觉信息和惯性信息融合的深度，VINS可以分为两种主要策略：

* 松耦合（Loosely-coupled）：  
  在这种方案中，视觉系统和IMU被当作两个独立的模块。视觉SLAM（或VO）模块独立运行，输出一个位姿估计。这个位姿估计随后被用作一个“观测值”，来更新一个基于IMU的滤波器（如EKF）42。
  + **优点**：实现简单，模块化程度高，可以将任何现成的VO系统与IMU进行组合 75。
  + **缺点**：是次优的。信息是单向流动的，视觉系统的误差会直接传递给IMU滤波器，而滤波器无法反过来修正视觉系统。一旦视觉模块失效，整个系统就无法获得更新，只能依赖IMU进行短暂的推算 75。
* 紧耦合（Tightly-coupled）：  
  在这种方案中，原始的视觉测量（如特征点的重投影误差）和原始的IMU测量被同时放入一个统一的优化框架中进行联合优化 42。
  + **优点**：能够达到理论上的最优解。视觉信息和惯性信息可以相互校正。例如，长期的视觉观测可以精确地估计出IMU的零偏，而IMU的约束可以帮助视觉系统进行初始化、恢复尺度、以及在特征点稀少时提供运动先验 74。
  + **缺点**：系统实现更为复杂，需要处理好两种传感器数据的时间同步、外参标定以及联合优化问题 76。

#### **9.3 IMU预积分理论**

在基于图优化的紧耦合VINS中，一个核心的挑战是如何处理频率极不匹配的两种数据。相机帧率通常在20-30Hz，而IMU数据频率可达数百Hz。

* **问题**：如果在优化图中为每一个IMU测量都创建一个状态节点，图的规模将变得无法承受。一个看似可行的方法是，在两个相机关键帧之间，将所有IMU测量积分起来，得到一个相对运动的约束。但是，这个积分结果依赖于积分起点的状态（位姿、速度、零偏）。在后端优化过程中，这些起点状态是不断被调整的，这意味着每次调整后，都需要重新进行IMU积分，这个计算开销同样巨大 42。
* 解决方案：预积分（Pre-integration）：  
  IMU预积分理论巧妙地解决了这个问题。它的核心思想是，将两关键帧之间的IMU测量在一个局部的、相对的参考系（即起始关键帧的体坐标系）中进行积分，从而得到一个只与IMU测量本身相关的相对运动增量 42。
  + **数学表述**：在关键帧 k 和 k+1 之间，对IMU的运动学方程进行积分，可以得到位移、速度和姿态的变化量。预积分通过变量代换，将这些积分项改写为不依赖于全局位姿的形式。例如，相对位移增量可以表示为 Δpij​=∬Rti​(at​−ba​)dt2，其中积分是在起始帧 i 的坐标系下进行的 42。
  + **零偏修正**：预积分的结果仍然依赖于IMU的零偏（bias）ba​ 和 bg​，而零偏也是优化变量之一。为了避免在零偏估计更新时重新积分，预积分理论使用了**一阶泰勒展开**来近似修正。即，当零偏有一个小的更新量 Δb 时，新的预积分值可以通过 Δp(b)≈Δp(b0​)+Jb​Δb 来快速计算，其中 Jb​ 是预积分量对零偏的雅可比矩阵，这个雅可比矩阵也可以在预积分过程中一并计算出来。这样，就可以在优化迭代中实现对预积分值的快速修正，而无需重新积分 42。

#### **9.4 VINS-Mono优化框架**

以著名的VINS-Mono系统为例，它采用了一种基于滑动窗口的图优化方法 73。

* **状态向量 χ**：优化窗口内的状态向量包括：窗口内所有关键帧的位姿、速度、IMU零偏，以及这些关键帧观测到的所有特征点的逆深度，还可能包括相机与IMU之间的外参 73。
* **代价函数**：优化的目标是最小化一个由三部分组成的代价函数，每一部分都是马氏距离的形式 73：χmin​⎩⎨⎧​∥rp​−Hp​χ∥Σp​2​+k∈B∑​∥rB​(z^k+1k​,χk​,χk+1​)∥Σk​2​+(l,j)∈C∑​∥rC​(z^lj​,χj​,λl​)∥Σl​2​⎭⎬⎫​
  + **先验因子 ∥rp​∥2**：来自于滑动窗口边缘化掉的旧状态信息，它将历史信息以先验的形式约束在当前优化问题中。
  + **IMU因子 ∥rB​∥2**：表示由优化变量（位姿、速度、零偏）计算出的相对运动与IMU预积分测量值之间的残差。
  + **视觉因子 ∥rC​∥2**：表示3D地图点在图像上重投影的位置与实际观测到的特征点位置之间的残差，即重投影误差。

IMU预积分是专门为在优化框架中实现紧耦合而设计的一种强大的理论工具。它不仅仅是简单地将高频数据打包，更是一种巧妙的计算技巧。它通过将繁重的非线性积分操作预先计算一次，然后在迭代优化中用极快的线性近似来进行修正，从而使得紧耦合VINS的非线性优化在计算上变得可行。这一技术是所有现代高性能VINS系统（如VINS-Mono）的基石。

## **第五部分：高级主题与未来前沿**

随着SLAM基础理论和核心算法的成熟，研究的焦点逐渐转向更具挑战性的高级问题，如保证长期运行的鲁棒性、构建更智能的地图、以及应对复杂动态环境等。同时，深度学习的浪潮也为SLAM带来了革命性的新思路。

### **第十章：回环闭合与全局地图一致性**

#### **10.1 漂移问题**

所有基于里程计的SLAM前端，无论是视觉的还是激光雷达的，都会因为测量噪声和模型误差而不可避免地累积误差。这种误差随着机器人运行时间的增长而不断累积，导致估计的轨迹与真实轨迹之间的偏差越来越大，这就是**漂移**（Drift）。漂移会使得构建的地图在全球尺度上变得不一致，例如，当机器人回到起点时，地图上可能会出现两个“起点”78。

**回环闭合**是解决漂移问题的唯一根本性方法。它指的是机器人能够识别出当前所处的场景是过去曾经访问过的地方，并在后端优化图中添加一条连接当前位姿与历史位姿的约束边。这个强大的全局约束能够像“拉橡皮筋”一样，将累积的误差在整个回环中进行分配和消除，从而得到一个全局一致的地图和轨迹 5。

#### **10.2 基于词袋模型的位置识别（DBoW2）**

回环检测的核心是**位置识别**（Place Recognition），即如何从成千上万的历史图像中，高效地找出与当前视图匹配的图像。逐一进行图像比对的暴力搜索方法在计算上是不可行的 81。\*\*词袋模型（Bag-of-Words, BoW）\*\*为此提供了一个高效的解决方案。

* **词袋模型类比**：在自然语言处理中，一篇文章可以被表示为一个词频的直方图，忽略语法和词序。类似地，在计算机视觉中，一张图像也可以被表示为一个由“视觉单词”组成的“袋子”81。
* **DBoW2流程** 81：DBoW2是专为视觉SLAM设计的、基于ORB特征的词袋库。
  1. **离线构建视觉词典**：首先，需要一个离线训练过程。从一个非常庞大的、多样化的图像数据集中提取数百万个ORB特征描述子。然后，通过层次聚类（如k-means）的方法，将这些描述子聚类成一个巨大的树状结构的**视觉词典**（Vocabulary Tree）。树的叶子节点就代表了一个个“视觉单词”53。
  2. **图像的BoW表示**：当需要表示一张新的图像时，首先提取其ORB特征。然后，将每个特征描述子从词典树的根节点开始向下传递，每次都选择距离最近的子节点，直到抵达一个叶子节点（即一个视觉单词）。最终，这张图像就被表示成一个稀疏的向量，该向量记录了图像中出现了哪些视觉单词以及它们的频率 84。
  3. **数据库查询与评分**：系统维护一个包含了所有历史关键帧BoW向量的数据库。当新的关键帧到来时，将其BoW向量与数据库中所有向量进行快速比较（通常计算L1或L2范数）。为了提高区分度，每个视觉单词的权重通常会用**TF-IDF**（Term Frequency-Inverse Document Frequency）进行加权，即降低那些在所有图像中都频繁出现的“通用”视觉单词的权重，而提高那些稀有的、具有辨识度的单词的权重 84。
  4. **时序与几何验证**：为了防止错误的匹配（感知混淆，Perceptual Aliasing，即两个不同的地方看起来很像），仅仅BoW向量相似是不够的。
     + **时序一致性验证**：一个可靠的回环候选通常需要连续几帧都与同一历史区域匹配 84。
     + **几何验证**：这是最关键的一步。系统会尝试在当前帧和回环候选帧之间进行特征点的几何匹配，并求解一个刚体变换（如通过PnP或Sim(3)求解）。只有当存在足够多的内点（Inliers）支持这个几何变换时，回环才被最终确认 82。

#### **10.3 位姿图优化**

一旦回环被确认，例如在位姿 xi​ 和 xj​ 之间，一条新的约束边就会被添加到后端的位姿图中。随后，后端优化器会重新求解整个图的最小二乘问题。这个过程会将回环带来的误差修正量，沿着图中的路径传播并分配到整个回环的所有节点上，最终得到一个全局一致的地图和轨迹 59。

### **第十一章：服务于机器人的地图表示**

#### **11.1 地图的目的**

在SLAM中，地图不仅仅是为了定位。一个好的地图是机器人执行更高级任务的基础，例如路径规划、导航、障碍物躲避以及人机交互 85。因此，地图的表示方式需要根据具体的应用需求来选择。

#### **11.2 地图表示方法分类**

* **稀疏特征地图（Sparse Feature-Based Maps）**：
  + **表示**：地图由一系列三维空间点（地标）及其关联的特征描述子组成，例如ORB-SLAM生成的地图 78。
  + **优点**：非常轻量级，占用内存极小，对于实时定位任务来说计算效率极高。
  + **缺点**：几何信息是稀疏的，无法表示物体的表面或环境中的可通行区域，因此对于导航和避障等任务几乎无用 78。
* 稠密体素地图（Dense Volumetric Maps）：  
  这类地图将空间离散化为一个个小的单元（二维中是栅格，三维中是体素），并存储每个单元的属性。
  + **占据栅格地图（Occupancy Grids）**：这是机器人领域最经典的地图表示。每个栅格/体素存储一个概率值，表示该空间被占据的概率 88。
    - **更新规则**：通常采用贝叶斯更新。为了数值稳定性，实践中常使用对数概率（log-odds）表示。新的观测会独立地更新每个栅格的占据概率 89。
    - **优点**：以概率的方式显式地表示了自由空间、占据空间和未知空间，这对于路径规划至关重要 89。
    - **缺点**：地图精度受限于栅格分辨率，且对于大范围三维空间，内存消耗巨大 88。\*\*八叉树（Octree）\*\*是一种常用的数据结构，通过只对包含信息的空间进行细分来有效缓解内存问题 89。
  + **截断符号距离场（Truncated Signed Distance Fields, TSDF）**：每个体素存储的值是该点到最近物面的**有向距离**，这个距离值通常在一个设定的阈值 μ 内被“截断”85。物体的表面被隐式地定义为距离值为零的等值面。
    - **更新规则**：通常是对来自新传感器的距离测量值进行加权平均 89。
    - **优点**：能够以亚体素（sub-voxel）的精度生成非常平滑、高质量的物体表面模型，非常适合用于三维重建和可视化 88。
    - **缺点**：不能显式地表示远离表面的自由空间，其本身不具备概率意义，因此直接用于路径规划较为困难 89。

地图表示的选择体现了**定位效率**、**重建质量**和**机器人任务可用性**之间的深刻权衡。纯粹为了定位，稀疏特征图是最高效的选择。为了让机器人能够规划路径，必须知道哪里是可通过的自由空间，占据栅格地图便成为自然之选。而如果目标是获得一个精美的三维模型用于可视化或检测，TSDF则是最佳方案。这种内在的矛盾表明，不存在一种“万能”的地图。先进的机器人系统可能会同时维护多种地图表示：一个稀疏特征图用于快速实时的定位，同时利用定位结果，在后台构建一个稠密的占据栅格图或TSDF模型用于导航和交互。这充分体现了机器人感知任务的“任务驱动”特性 91。

#### **表3：地图表示方法对比**

| 特性 | 稀疏特征地图 | 占据栅格地图 | 截断符号距离场 (TSDF) |
| --- | --- | --- | --- |
| **基本单元** | 三维点 + 描述子 | 体素 + 占据概率 | 体素 + 截断的有向距离 |
| **表示内容** | 环境中一组稀疏的、显著的三维点 | 空间中每个离散体素被占据的概率 | 由距离函数零点隐式定义的连续表面 |
| **主要用途** | 定位、位姿跟踪 | 路径规划、障碍物躲避、导航 | 高质量三维表面重建、可视化 |
| **内存占用** | 非常低 (与特征数量成正比) | 高 (与体积成正比)，通常用八叉树缓解 | 高 (与表面附近体积成正比)，通常用体素哈希或八叉树缓解 |
| **优点** | 轻量级，定位计算效率高 | 显式表示自由空间，具有概率意义，是规划的理想选择 | 亚体素精度，生成平滑、高保真度的表面 |
| **缺点** | 几何稀疏，对规划和稠密重建无用 | 精度受限于体素分辨率，外观呈“块状” | 不显式表示远离表面的自由空间，不太适合路径规划 |
| **示例系统** | ORB-SLAM, PTAM | Cartographer, OctoMap | KinectFusion, InfiniTAM, ESLAM |

### **第十二章：真实世界中的SLAM：动态与语义环境**

#### **12.1 静态世界假设及其局限性**

绝大多数经典SLAM算法都建立在一个核心假设之上：**世界是刚性且静态的**。然而，真实世界充满了变化：行人、车辆在移动，门窗在开合 9。动态物体会给SLAM系统带来灾难性的影响。因为它们上面的特征点相对于世界坐标系是运动的，这与SLAM的基本假设相悖。如果将这些动态点作为静态地标处理，就会在优化图中引入错误的约束，导致位姿估计严重错误，并最终构建出一张“鬼影”重重、无法使用的地图 92。

#### **12.2 动态SLAM：处理移动物体**

为了在真实动态环境中鲁棒运行，SLAM系统必须具备处理动态物体的能力。目前主要有两种策略：

* 策略一：检测并剔除（Detect-and-Remove）：  
  这是最常见的方法，它将动态物体视为需要被剔除的异常值（Outliers） 94。
  + **实现方法**：通过几何一致性检验（如RANSAC）、光流法，或者利用深度学习的目标检测（如YOLO）和语义分割（如SegNet）网络，来识别出图像中可能属于动态物体的区域，然后将这些区域内的特征点从后端优化中排除 92。DynaSLAM和DS-SLAM就是这种思想的代表性系统 92。
* 策略二：检测并跟踪（Detect-and-Track）：  
  这是一种更高级的范式，被称为SLAMMOT（Simultaneous Localization and Mapping and Moving Object Tracking） 95。
  + **核心思想**：它不再将动态物体视为噪声，而是将其作为环境状态的一部分进行显式的建模和跟踪。系统在估计自身位姿的同时，也估计和预测环境中其他移动物体的轨迹。这通常通过将问题分解为两个独立的估计器来实现：一个用于静态背景的SLAM，另一个用于动态物体的多目标跟踪 95。这种方法能够提供对环境更丰富的理解，对于自动驾驶等应用至关重要。

#### **12.3 语义SLAM：迈向高层理解**

传统的SLAM地图，无论是点云还是栅格图，都只包含了纯粹的几何信息，缺乏“意义”。机器人可能构建了一张完美的椅子点云模型，但它并不知道“这是一把椅子”7。\*\*语义SLAM（Semantic SLAM）\*\*的目标就是构建不仅在几何上精确，而且在语义上可理解的地图 96。

* **语义信息融合**：
  + **语义提取**：利用先进的深度学习模型，如目标检测网络（YOLO）或实例分割网络（Mask R-CNN），为图像中的像素、物体赋予类别标签（如“椅子”、“桌子”、“人”）96。
  + **语义地图表示**：地图不再仅仅是三维点的集合，而是由带有语义标签的物体组成的。例如，地图中可能包含一个标签为“椅子”的物体，它具有自己的位姿、尺寸和形状（可以用一个三维包围盒或椭球体来表示）97。
* **语义SLAM的优势**：
  + **提升鲁棒性**：语义信息为SLAM提供了强大的先验和约束。例如，识别出一个“杯子”是一个刚体，那么它上面的所有特征点都应该一起运动。识别出一个“人”是潜在的动态物体，应该被特殊处理。这种基于物体的约束可以极大地改善数据关联和回环检测的准确性 96。
  + **实现高级人机交互**：语义地图使得机器人能够理解更高级的指令，如“去厨房把杯子拿过来”96。
  + **支持高效推理**：机器人可以利用语义进行更高层次的推理。例如，识别出“地板”是可通行的平面，而“墙壁”是障碍物 91。

动态SLAM和语义SLAM的发展，标志着SLAM领域从纯粹的几何估计问题，向着\*\*整体场景理解（Holistic Scene Understanding）\*\*的范式转变。在这一新范式中，“地图”的概念被极大地丰富了，它不再是一个静态的几何数据库，而是一个多层次的、动态的、充满意义的世界模型。经典SLAM的目标是求解 p(位姿,地标∣测量)。而动态SLAM的目标则演变为求解 p(位姿,静态地图,动态物体∣测量)。语义SLAM则更进一步，其目标是求解 p(位姿,语义地图∣测量)，其中语义地图包含了物体的类别、属性、功能甚至相互关系。SLAM的这一演进轨迹，正推动着它从一个单纯的定位工具，向着构建真正智能体感知能力的核心模块迈进，这也是当前机器人学研究最重要的前沿方向之一 91。

### **第十三章：学习的革命：深度学习与神经SLAM**

深度学习的兴起为计算机视觉和机器人领域带来了革命性的变化，SLAM也不例外。学习方法正以前所未有的深度和广度渗透到SLAM的各个环节。

#### **13.1 深度学习在SLAM管线中的应用**

在传统的SLAM框架中，深度学习首先被用作强大功能的“即插即用”模块，以增强或替代某些传统组件 98：

* **特征提取与描述**：使用卷积神经网络（CNN）学习比手工设计的ORB等特征更强大、更具鲁棒性的特征。
* **位姿估计/视觉里程计**：通过有监督学习，训练神经网络直接从连续的图像对中回归出相机的相对运动，例如DeepVO等方法。
* **语义理解**：如前一章所述，利用目标检测和语义分割网络为SLAM提供高层语义信息，是深度学习在SLAM中最成功和最广泛的应用。

#### **13.2 端到端的学习方法**

一些研究尝试用一个单一的、巨大的神经网络来替代整个SLAM系统，实现从传感器输入到最终位姿和地图输出的“端到端”（End-to-End）学习 98。然而，这类方法目前仍面临巨大挑战，包括泛化能力差（在一个数据集上训练好的模型很难在全新环境中良好工作）、缺乏可解释性，并且在精度上通常仍逊于经典的、基于几何原理的方法 98。

#### **13.3 新前沿：基于神经辐射场（NeRF）的SLAM**

近年来，一种名为\*\*神经辐射场（Neural Radiance Fields, NeRF）\*\*的技术为SLAM带来了全新的思路，催生了“神经SLAM”（Neural SLAM）这一激动人心的研究方向。

* **NeRF简介**：NeRF是一种用于合成新视角图像的强大技术。它使用一个简单的多层感知机（MLP）来学习一个场景的连续、隐式的表示。这个网络将一个五维输入（一个三维空间点坐标 (x,y,z) 和一个二维观测方向 (θ,ϕ)）映射到一个体密度（Volume Density）σ 和一个颜色值 c 100。通过沿着相机射线对这些值进行体渲染（Volume Rendering），就可以合成出任意新视角下的图像。
* **基于NeRF的SLAM（例如iMAP, NICE-SLAM）**：
  + **隐式地图表示**：在NeRF-based SLAM中，地图不再是点云、体素或特征点的显式集合。**地图本身就是神经网络的权重** 102。
  + **跟踪与建图的联合优化**：整个SLAM过程变成了一个联合优化问题。当一帧新的RGB-D图像到来时：
    1. **跟踪（Tracking）**：保持网络权重（即地图）不变，通过优化相机位姿，来最小化从该位姿**渲染**出的图像与真实观测图像之间的**渲染损失**（通常是光度误差）。这相当于通过“分析-合成”（Analysis-by-Synthesis）的方式来定位相机 102。
    2. **建图（Mapping）**：选择一些关键帧，利用这些关键帧的观测数据来更新（即训练）网络权重，使其能更好地表示场景。在这个过程中，关键帧的位姿通常也会与网络权重一起进行联合优化 103。
  + **代表性系统**：
    - **iMAP**：是这一领域的开创性工作，首次证明了可以用单个MLP作为唯一场景表示，实现实时的SLAM 103。
    - **NICE-SLAM**：在iMAP的基础上进行了改进，它使用分层的、基于特征网格的编码方式，而不是一个单一的巨大MLP。这使得训练速度更快，能够构建更大、更精细的场景地图 107。
* **优缺点**：
  + **优点**：能够生成照片般逼真的稠密地图，并能合成任意新视角的图像。隐式表示是连续的，且内存占用相对较小 103。
  + **缺点**：计算量极大，对算力要求很高。同时，它也面临深度学习中的经典问题，如“灾难性遗忘”（即网络在学习新场景时忘记旧场景），并且如何有效地进行回环闭合仍然是一个巨大的挑战 101。

基于NeRF的SLAM从根本上重新定义了“地图”的概念，它将地图从一个**描述性的几何数据库**（存储着点、面等信息）转变为一个**生成式的函数**（一个能够从任意位姿生成图像的函数）。相应地，核心的优化问题也从最小化**几何误差**（如重投影误差、点面距离）转变为最小化**光度渲染损失**（渲染图像与真实图像的像素差异）102。这是一个深刻的范式转变，它将三维重建与神经渲染领域直接融合进了SLAM问题中。尽管目前仍处于早期发展阶段，但它为创建极其逼真和可交互的环境“数字孪生”开辟了全新的可能性。当然，这也引入了来自深度学习领域的一系列新挑战，如训练稳定性、泛化能力和巨大的计算成本。

### **第十四章：SLAM的未来：开放性挑战与前行之路**

#### **14.1 “SLAM问题解决了吗？”之辩**

这是一个在机器人学会议上经久不衰的辩论话题。从某种意义上说，对于基本的、在受控环境下的几何建图问题，我们可以认为其理论和概念框架已经基本建立 4。然而，当我们把目光投向在真实世界中进行长期的、鲁棒的、自主的运行时，答案则是否定的 80。SLAM远未被“解决”。

#### **14.2 “鲁棒感知时代”：关键研究前沿**

一篇影响深远的综述性论文将SLAM的未来发展方向概括为进入“鲁棒感知时代”（The Robust-Perception Age）80。这标志着研究重点从单纯追求几何精度，转向追求系统的鲁棒性、智能性和适应性。关键的开放性挑战包括：

* **鲁棒性与长期自主性**：
  + **故障安全SLAM（Failsafe SLAM）**：开发能够自我诊断、感知即将发生的失败（如错误的数据关联、传感器故障），并具备恢复机制的系统 91。
  + **终身学习与建图（Life-long Mapping）**：处理环境随时间发生的外观变化（如昼夜、季节更替）、光照变化，以及如何进行地图维护（学习新知识、遗忘过时信息、在需要时记起），并应对非刚性的、可形变的环境 91。
  + **参数自动调优**：开发能够“开箱即用”的SLAM系统，使其能根据不同场景自动调整内部参数，摆脱对专家手动调参的依赖。
* **可扩展性（Scalability）**：
  + **分布式SLAM**：研究多机器人协同建图，特别是在通信带宽受限或不稳定的情况下的高效、鲁棒的协同策略。
  + **资源受限平台**：为计算能力和内存极为有限的平台（如微型无人机、智能手机、甚至机器昆虫）设计轻量级的SLAM算法。
* **表示与推理（Representation and Reasoning）**：
  + **高层场景理解**：超越纯粹的几何重建，构建包含语义、物理属性、功能可供性（Affordances）的丰富世界模型。
  + **任务驱动的感知**：让机器人能够根据当前任务的需求，自适应地构建最合适的地图，而不是千篇一律地进行几何重建。
* **主动SLAM（Active SLAM）**：机器人如何智能地规划自己的运动，以便最高效地构建地图或最快地消除自身位置的不确定性？这涉及到在不确定性下的决策与规划问题 91。

#### **14.3 结语**

SLAM技术从一个机器人学领域的理论难题，发展至今，已成为人工智能和自主系统的基石技术之一。它的演进历程，从基于滤波的在线估计，到基于优化的全局平滑，再到与深度学习的深度融合，每一步都伴随着对问题本质更深刻的理解和计算工具的革新。

未来，SLAM的发展将不再局限于几何层面。一个真正智能的机器人需要一个能够理解世界、适应变化、并服务于任务的感知系统。这要求我们将经典的几何推理方法与强大的现代学习技术进行更深层次的、有机的结合。这条融合之路，将引领我们走向一个机器人能够真正鲁棒、自主、智能地感知和交互的未来。

#### 引用的著作

1. Graph SLAM: From Theory to Implementation - Federico Sarrocco, 访问时间为 六月 28, 2025， <https://federicosarrocco.com/blog/graph-slam-tutorial>
2. What is SLAM? A Beginner to Expert Guide - Kodifly, 访问时间为 六月 28, 2025， <https://kodifly.com/what-is-slam-a-beginner-to-expert-guide>
3. A Tutorial on Graph-Based SLAM - Uni Freiburg, 访问时间为 六月 28, 2025， <http://www2.informatik.uni-freiburg.de/~stachnis/pdf/grisetti10titsmag.pdf>
4. Simultaneous Localisation and Mapping (SLAM): Part I The Essential Algorithms - People @EECS, 访问时间为 六月 28, 2025， <https://people.eecs.berkeley.edu/~pabbeel/cs287-fa09/readings/Durrant-Whyte_Bailey_SLAM-tutorial-I.pdf>
5. SLAM Tutorial - Department of Computer Science, Columbia University, 访问时间为 六月 28, 2025， <https://www.cs.columbia.edu/~allen/F19/NOTES/slam_pka.pdf>
6. Dual FastSLAM: Dual Factorization of the Particle Filter Based Solution of the Simultaneous Localization and Mapping Problem | Request PDF - ResearchGate, 访问时间为 六月 28, 2025， <https://www.researchgate.net/publication/220061662_Dual_FastSLAM_Dual_Factorization_of_the_Particle_Filter_Based_Solution_of_the_Simultaneous_Localization_and_Mapping_Problem>
7. SLAM-past-present-future.pdf - cs.wisc.edu, 访问时间为 六月 28, 2025， <https://pages.cs.wisc.edu/~jphanna/teaching/25spring_cs639/resources/SLAM-past-present-future.pdf>
8. SLAM - arXiv, 访问时间为 六月 28, 2025， <https://arxiv.org/pdf/2104.06800>
9. Is Semantic SLAM Ready for Embedded Systems ? A Comparative Survey - arXiv, 访问时间为 六月 28, 2025， <https://arxiv.org/html/2505.12384v1>
10. Towards Collaborative Simultaneous Localization and Mapping: a Survey of the Current Research Landscape - arXiv, 访问时间为 六月 28, 2025， <https://arxiv.org/pdf/2108.08325>
11. SLaM Tutorial (Part I), 访问时间为 六月 28, 2025， <https://cse.sc.edu/~yiannisr/774/2015/SLAM1.pptx>
12. SLAM: Simultaneous Localization and Mapping: Mathematical foundations, 访问时间为 六月 28, 2025， <https://aicodewizards.com/2022/07/11/slam-simultaneous-localization-and-mapping-theoretical-part/>
13. Handbook of Robotics, 2nd Edition Simultaneous Localization and Mapping, 访问时间为 六月 28, 2025， <https://www.ipb.uni-bonn.de/wp-content/papercite-data/pdf/springerbook-slamchapter.pdf>
14. EKF SLAM - Jihong Ju, 访问时间为 六月 28, 2025， <https://jihongju.github.io/2018/10/05/slam-05-ekf-slam/>
15. Linear Algebra for EKF SLAM: A Deep Dive - Number Analytics, 访问时间为 六月 28, 2025， <https://www.numberanalytics.com/blog/linear-algebra-ekf-slam-robotics>
16. Robot Mapping EKF SLAM, 访问时间为 六月 28, 2025， <http://ais.informatik.uni-freiburg.de/teaching/ws12/mapping/pdf/slam04-ekf-slam.pdf>
17. python - Issue calculating landmark covariances in EKF-SLAM - Stack Overflow, 访问时间为 六月 28, 2025， <https://stackoverflow.com/questions/29379777/issue-calculating-landmark-covariances-in-ekf-slam>
18. Intro to the EKF Step 4: SLAM With Unknown Correspondence - Andrew Kramer's, 访问时间为 六月 28, 2025， <https://andrewjkramer.net/intro-to-the-ekf-step-4/>
19. Simultaneous Localization and Mapping (SLAM): EKF SLAM - YouTube, 访问时间为 六月 28, 2025， <https://www.youtube.com/watch?v=nUNSDXTY_oU>
20. FastSLAM 2.0: An Improved Particle Filtering Algorithm for Simultaneous Localization and Mapping that Provably Converges | Request PDF - ResearchGate, 访问时间为 六月 28, 2025， <https://www.researchgate.net/publication/2480644_FastSLAM_20_An_Improved_Particle_Filtering_Algorithm_for_Simultaneous_Localization_and_Mapping_that_Provably_Converges>
21. FastSLAM: A Factored Solution to SLAM Particle Filters Localization vs. SLAM Dependencies - MIT, 访问时间为 六月 28, 2025， <https://web.mit.edu/2.166/www/handouts/2166_fastslam_oct20.pdf>
22. CSE-571 Robotics Particle Filters Dependencies Particle Filter Algorithm - Washington, 访问时间为 六月 28, 2025， <https://courses.cs.washington.edu/courses/cse571/16au/slides/12-fastslam.pdf>
23. LSO-FastSLAM: A New Algorithm to Improve the Accuracy of Localization and Mapping for Rescue Robots - PMC - PubMed Central, 访问时间为 六月 28, 2025， <https://pmc.ncbi.nlm.nih.gov/articles/PMC8839077/>
24. SLAM Tutorial@ICRA 2016, 访问时间为 六月 28, 2025， <http://www.dis.uniroma1.it/~labrococo/tutorial_icra_2016/>
25. Gauss-Newton / Levenberg-Marquardt Optimization, 访问时间为 六月 28, 2025， <https://mat.uab.cat/~alseda/MasterOpt/optimization.pdf>
26. SLAM: Simultaneous Localization and Mapping: Theoretical part - AI Code Wizards, 访问时间为 六月 28, 2025， <https://aicodewizards.com/2022/09/16/slam-simultaneous-localization-and-mapping-theoretical-part-2/>
27. G2o: A general framework for graph optimization - ResearchGate, 访问时间为 六月 28, 2025， <https://www.researchgate.net/publication/224252449_G2o_A_general_framework_for_graph_optimization>
28. Nonlinear Graph Sparsification for SLAM - Robotics, 访问时间为 六月 28, 2025， <https://www.roboticsproceedings.org/rss10/p40.pdf>
29. Graph SLAM - Washington, 访问时间为 六月 28, 2025， <https://courses.cs.washington.edu/courses/cse571/23sp/slides/L09/Lecture09_Modern%20SLAM.pdf>
30. UZ-SLAMLab/ORB\_SLAM3: ORB-SLAM3: An Accurate ... - GitHub, 访问时间为 六月 28, 2025， <https://github.com/UZ-SLAMLab/ORB_SLAM3>
31. Analysis for Graph-Based SLAM Algorithms under g2o Framework - Carnegie Mellon University, 访问时间为 六月 28, 2025， <https://www.cs.cmu.edu/~tianxian/files/Analysis_for_Graph_Based_SLAM_Algorithms_under_g2o_Framework.pdf>
32. Leveraging the Levenberg-Marquardt Method on Vanishing Point Estimation | by Oleg Boev, 访问时间为 六月 28, 2025， <https://medium.com/@olegboev/leveraging-the-levenberg-marquardt-method-on-vanishing-point-estimation-in-computer-vision-5b10668b1ad5>
33. Gauss–Newton algorithm - Wikipedia, 访问时间为 六月 28, 2025， <https://en.wikipedia.org/wiki/Gauss%E2%80%93Newton_algorithm>
34. Efficient Levenberg-Marquat for SLAM - OPT 2024, 访问时间为 六月 28, 2025， <http://opt-ml.org/papers/2024/paper27.pdf>
35. Sparsity patterns of the Hessian and Laplacian matrices in a toy... - ResearchGate, 访问时间为 六月 28, 2025， <https://www.researchgate.net/figure/Sparsity-patterns-of-the-Hessian-and-Laplacian-matrices-in-a-toy-example-with-6-poses-and_fig2_363402523>
36. What is g2o? Competitors, Complementary Techs & Usage | Sumble, 访问时间为 六月 28, 2025， <https://sumble.com/tech/g2o>
37. A Comparison of Graph Optimization Approaches for Pose Estimation in SLAM - Laboratory for Autonomous Systems and Mobile Robotics, 访问时间为 六月 28, 2025， <https://lamor.fer.hr/images/50036607/2021-ajuric-comparison-mipro.pdf>
38. State of the Art in Real-time Registration of RGB-D Images - Visual Computing, 访问时间为 六月 28, 2025， <https://cg.cs.uni-bonn.de/backend/v1/files/publications/StateoftheArtinReal-timeRegistrationofRGB-DImages.pdf>
39. Solving the scale ambiguity in monocular SLAM : r/computervision - Reddit, 访问时间为 六月 28, 2025， <https://www.reddit.com/r/computervision/comments/3zovt9/solving_the_scale_ambiguity_in_monocular_slam/>
40. Monocular VO Scale Ambiguity Resolution Using an Ultra Low-Cost Spike Rangefinder, 访问时间为 六月 28, 2025， <https://www.scirp.org/journal/paperinformation?paperid=106397>
41. Scale Drift-Aware Large Scale Monocular SLAM - Robotics, 访问时间为 六月 28, 2025， <https://www.roboticsproceedings.org/rss06/p10.pdf>
42. [1708.03852] VINS-Mono: A Robust and Versatile Monocular Visual ..., 访问时间为 六月 28, 2025， <https://ar5iv.labs.arxiv.org/html/1708.03852>
43. Resolving Scale Ambiguity in Multi-view 3D Reconstruction using Dual-Pixel Sensors - European Computer Vision Association, 访问时间为 六月 28, 2025， <https://www.ecva.net/papers/eccv_2024/papers_ECCV/papers/06658.pdf>
44. Stereo depth cameras for mobile phones - Intel® RealSense™ Documentation, 访问时间为 六月 28, 2025， <https://dev.intelrealsense.com/docs/stereo-depth-cameras-for-phones>
45. Disparity Map in Stereo Vision | Baeldung on Computer Science, 访问时间为 六月 28, 2025， <https://www.baeldung.com/cs/disparity-map-stereo-vision>
46. Stereo Camera: Triangulation Explained, 访问时间为 六月 28, 2025， <https://www.mirellamelo.com/post/stereo-camera-triangulation-explained>
47. 1: Triangulation in a Stereo Vision system | Download Scientific Diagram - ResearchGate, 访问时间为 六月 28, 2025， <https://www.researchgate.net/figure/Triangulation-in-a-Stereo-Vision-system_fig9_267214563>
48. RGB-D SLAM Combining Visual Odometry and Extended Information Filter - MDPI, 访问时间为 六月 28, 2025， <https://www.mdpi.com/1424-8220/15/8/18742>
49. A Comparative Study of Registration Methods for RGB-D Video of Static Scenes - PMC, 访问时间为 六月 28, 2025， <https://pmc.ncbi.nlm.nih.gov/articles/PMC4063070/>
50. An Evaluation of the RGB-D SLAM System, 访问时间为 六月 28, 2025， <http://ais.informatik.uni-freiburg.de/publications/papers/endres12icra.pdf>
51. Robust RGB-D SLAM Using Point and Line Features for Low Textured Scene - PMC, 访问时间为 六月 28, 2025， <https://pmc.ncbi.nlm.nih.gov/articles/PMC7506666/>
52. Feature-based visual SLAM tutorial (part 0) - Roving Robots, 访问时间为 六月 28, 2025， <https://rovingrobots.wordpress.com/2020/06/16/feature-based-visual-slam-tutorial-part-0/>
53. ORB-SLAM2原理分析原创 - CSDN博客, 访问时间为 六月 28, 2025， <https://blog.csdn.net/qq_44089890/article/details/130087896>
54. 视觉里程计：特征点法（ORB特征） 原创 - CSDN博客, 访问时间为 六月 28, 2025， <https://blog.csdn.net/wjinjie/article/details/104451332>
55. ORB\_SLAM v2 - Taobotics Robot, 访问时间为 六月 28, 2025， <https://docs.taobotics.com/docs/handsfree/handsfree-en/Tutorial/Advanced/ORB_SLAM2/doc.html>
56. 基于ORB特征的视觉里程计算法优化 - Researching, 访问时间为 六月 28, 2025， <https://www.researching.cn/ArticlePdf/m00002/2019/56/21/211507.pdf>
57. A High-Performance ORB Feature Extraction Accelerator for ... - IJERM, 访问时间为 六月 28, 2025， <https://www.ijerm.com/download_data/IJERM1203011.pdf>
58. ALGD-ORB: An improved image feature extraction algorithm with adaptive threshold and local gray difference - PMC - PubMed Central, 访问时间为 六月 28, 2025， <https://pmc.ncbi.nlm.nih.gov/articles/PMC10593235/>
59. ORB-SLAM源码解读 - 无处不在的小土, 访问时间为 六月 28, 2025， <https://gaoyichao.com/Xiaotu/?book=ORB_SLAM%E6%BA%90%E7%A0%81%E8%A7%A3%E8%AF%BB&title=index>
60. LSD-SLAM: Large-Scale Direct Monocular SLAM - Jakob Engel, 访问时间为 六月 28, 2025， <https://jakobengel.github.io/pdf/engel14eccv.pdf>
61. SVR-Net: A Sparse Voxelized Recurrent Network for Robust Monocular SLAM with Direct TSDF Mapping - PubMed Central, 访问时间为 六月 28, 2025， <https://pmc.ncbi.nlm.nih.gov/articles/PMC10141714/>
62. LSD-SLAM (Large-Scale Direct Monocular SLAM) is a direct SLAM algorithm that minimizes photometric error instead of relying on discrete feature extraction to estimate camera motion and scene depth… - Irisyang, 访问时间为 六月 28, 2025， <https://medium.com/@irisyang886/lsd-slam-large-scale-direct-monocular-slam-is-a-direct-slam-algorithm-that-minimizes-photometric-c4d2db60d9cb>
63. Chapter 11 Photometric Error and Direct SLAM - Computer Vision Group, 访问时间为 六月 28, 2025， <https://cvg.cit.tum.de/_media/teaching/ss2023/mvg2023/material/chapter11_photometric_error_direct_slam.pdf>
64. Understanding how Direct Visual SLAM works | Kudan Global, 访问时间为 六月 28, 2025， <https://www.kudan.io/blog/direct-visual-slam/>
65. 视觉SLAM技术综述 - 阿里云天池, 访问时间为 六月 28, 2025， <https://tianchi.aliyun.com/forum/post/77441>
66. Direct Real-Time SLAM - Jakob Engel, Daniel Cremers, 访问时间为 六月 28, 2025， <http://www.dis.uniroma1.it/~labrococo/tutorial_icra_2016/icra16_slam_tutorial_engel.pdf>
67. A Plane-based LiDAR Odometry Method for Man-made Scene - UCL Discovery, 访问时间为 六月 28, 2025， <https://discovery.ucl.ac.uk/id/eprint/10188522/1/Boli_%20Plane_based_LiDAR_cdc2023.pdf>
68. Understanding Iterative Closest Point (ICP) Algorithm with Code - LearnOpenCV, 访问时间为 六月 28, 2025， <https://learnopencv.com/iterative-closest-point-icp-explained/>
69. ICP registration — Open3D 0.12.0 documentation, 访问时间为 六月 28, 2025， <https://www.open3d.org/docs/0.12.0/tutorial/pipelines/icp_registration.html>
70. 12.2: The Iterative Closest Point (ICP) Algorithm - Engineering LibreTexts, 访问时间为 六月 28, 2025， <https://eng.libretexts.org/Bookshelves/Mechanical_Engineering/Introduction_to_Autonomous_Robots_(Correll)/12%3A__RGB-D_SLAM/12.02%3A_The_Iterative_Closest_Point_(ICP)_Algorithm>
71. An Enhanced LiDAR-Based SLAM Framework: Improving NDT Odometry with Efficient Feature Extraction and Loop Closure Detection - MDPI, 访问时间为 六月 28, 2025， <https://www.mdpi.com/2227-9717/13/1/272>
72. Feature Extraction of Horizontal Plane and Optimization of 3D LiDAR SLAM in Indoor Environments - ResearchGate, 访问时间为 六月 28, 2025， <https://www.researchgate.net/publication/388982356_Feature_Extraction_of_Horizontal_Plane_and_Optimization_of_3D_LiDAR_SLAM_in_Indoor_Environments>
73. Formula Derivation and Analysis of the VINS-Mono - arXiv, 访问时间为 六月 28, 2025， <https://arxiv.org/pdf/1912.11986>
74. A Plug-and-Play Learning-based IMU Bias Factor for Robust Visual-Inertial Odometry - arXiv, 访问时间为 六月 28, 2025， <https://arxiv.org/html/2503.12527v1>
75. Loosely vs Tightly Coupled INS & GNSS | Point One Nav, 访问时间为 六月 28, 2025， <https://pointonenav.com/news/loose-vs-tight-coupling-gnss/>
76. What's The Difference Between Tight And Loose Coupling? - Clean Commit, 访问时间为 六月 28, 2025， <https://cleancommit.io/blog/whats-the-difference-between-tight-and-loose-coupling/>
77. MAVIS: Multi-Camera Augmented Visual-Inertial SLAM using SE₂⁢(3) Based Exact IMU Pre-integration - arXiv, 访问时间为 六月 28, 2025， <https://arxiv.org/html/2309.08142v5>
78. ORB\_SLAM v2 - Taobotics Robot, 访问时间为 六月 28, 2025， <https://docs.taobotics.com/docs/handsfree/handsfree/Tutorial/Advanced/ORB_SLAM2/doc.html>
79. An Effective LiDAR-Inertial SLAM-Based Map Construction Method for Outdoor Environments - MDPI, 访问时间为 六月 28, 2025， <https://www.mdpi.com/2072-4292/16/16/3099>
80. Past, Present, and Future of Simultaneous Localization And ... - arXiv, 访问时间为 六月 28, 2025， <http://www.arxiv.org/pdf/1606.05830v2.pdf>
81. Simple bag-of-words loop closure for visual SLAM - Nicolò Valigi, 访问时间为 六月 28, 2025， <https://nicolovaligi.com/articles/bag-of-words-loop-closure-visual-slam/>
82. Introduction to Loop Closure Detection in SLAM - Think Autonomous, 访问时间为 六月 28, 2025， <https://www.thinkautonomous.ai/blog/loop-closure/>
83. bagOfFeaturesDBoW - Bag of visual words using DBoW2 library - MATLAB - MathWorks, 访问时间为 六月 28, 2025， <https://www.mathworks.com/help/vision/ref/bagoffeaturesdbow.html>
84. Bags of Binary Words for Fast Place Recognition in Image Sequences, 访问时间为 六月 28, 2025， <http://doriangalvez.com/php/dl.php?dlp=GalvezTRO12.pdf>
85. coVoxSLAM: GPU accelerated globally consistent dense SLAM - arXiv, 访问时间为 六月 28, 2025， <https://arxiv.org/html/2410.21149v1>
86. Visual SLAM for Autonomous Ground Vehicles - Andreas Geiger, 访问时间为 六月 28, 2025， <https://www.cvlibs.net/publications/Lategahn2011ICRA.pdf>
87. SP-SLAM: Neural Real-Time Dense SLAM With Scene Priors - arXiv, 访问时间为 六月 28, 2025， <https://arxiv.org/html/2501.06469v1>
88. Large Scale 2D Laser SLAM using Truncated Signed Distance Functions - TU Darmstadt, 访问时间为 六月 28, 2025， <https://web.sim.informatik.tu-darmstadt.de/publ/download/2019_daun_ssrr.pdf>
89. Efficient Octree-Based Volumetric SLAM Supporting Signed ..., 访问时间为 六月 28, 2025， <https://www.doc.ic.ac.uk/~sleutene/publications/EVespaRAL_final.pdf>
90. ESLAM: Efficient Dense SLAM System Based on Hybrid Representation of Signed Distance Fields - CVF Open Access, 访问时间为 六月 28, 2025， <https://openaccess.thecvf.com/content/CVPR2023/papers/Johari_ESLAM_Efficient_Dense_SLAM_System_Based_on_Hybrid_Representation_of_CVPR_2023_paper.pdf>
91. Simultaneous Localization And Mapping: Present, Future, and the Robust-Perception Age, 访问时间为 六月 28, 2025， <https://www.researchgate.net/publication/304163768_Simultaneous_Localization_And_Mapping_Present_Future_and_the_Robust-Perception_Age>
92. Visual SLAM algorithm based on lightweight object detection network in dynamic environments - SPIE Digital Library, 访问时间为 六月 28, 2025， <https://www.spiedigitallibrary.org/conference-proceedings-of-spie/13539/1353905/Visual-SLAM-algorithm-based-on-lightweight-object-detection-network-in/10.1117/12.3057784.full?webSyncID=b0d3a6b0-3c97-c009-b6bc-d746b80d8e8a&sessionGUID=beeaed6a-e05c-117a-ac5a-9199362b9eb3>
93. OC-SLAM: Steadily Tracking and Mapping in Dynamic Environments - Frontiers, 访问时间为 六月 28, 2025， <https://www.frontiersin.org/journals/energy-research/articles/10.3389/fenrg.2021.803631/full>
94. A Dynamic Visual SLAM System Incorporating Object Tracking for UAVs - MDPI, 访问时间为 六月 28, 2025， <https://www.mdpi.com/2504-446X/8/6/222>
95. Simultaneous Localization, Mapping and Moving Object Tracking, 访问时间为 六月 28, 2025， <https://www.ri.cmu.edu/pub_files/pub4/wang_chieh_chih_2007_1/wang_chieh_chih_2007_1.pdf>
96. Semantic Visual Simultaneous Localization and Mapping: A Survey - arXiv, 访问时间为 六月 28, 2025， <https://arxiv.org/html/2209.06428v2>
97. Semantic Visual Simultaneous Localization and Mapping: A ... - arXiv, 访问时间为 六月 28, 2025， <https://arxiv.org/pdf/2209.06428>
98. Monocular visual simultaneous localization and mapping: (r)evolution from geometry to deep learning-based pipelines - arXiv, 访问时间为 六月 28, 2025， <https://arxiv.org/pdf/2503.02955>
99. MIT Open Access Articles NeRF-SLAM: Real-Time Dense Monocular SLAM with Neural Radiance Fields, 访问时间为 六月 28, 2025， <https://dspace.mit.edu/bitstream/handle/1721.1/153646/2210.13641.pdf?sequence=2&isAllowed=y>
100. NeRF: Neural Radiance Field in 3D Vision: A Comprehensive Review - arXiv, 访问时间为 六月 28, 2025， <https://arxiv.org/html/2210.00379v6>
101. (PDF) Current study on NeRF-based techniques in SLAM frameworks - ResearchGate, 访问时间为 六月 28, 2025， <https://www.researchgate.net/publication/378435472_Current_study_on_NeRF-based_techniques_in_SLAM_frameworks>
102. SLAM Meets NeRF: A Survey of Implicit SLAM Methods - MDPI, 访问时间为 六月 28, 2025， <https://www.mdpi.com/2032-6653/15/3/85>
103. [2103.12352] iMAP: Implicit Mapping and Positioning in Real-Time - ar5iv - arXiv, 访问时间为 六月 28, 2025， <https://ar5iv.labs.arxiv.org/html/2103.12352>
104. iMAP: Implicit Mapping and Positioning in Real-Time - Edgar Sucar, 访问时间为 六月 28, 2025， <https://edgarsucar.github.io/iMAP/>
105. iMAP: Implicit Mapping and Positioning in Real-Time, 访问时间为 六月 28, 2025， <https://arxiv.org/pdf/2103.12352>
106. iMAP: Implicit Mapping and Positioning in Real-Time - CVF Open Access, 访问时间为 六月 28, 2025， <https://openaccess.thecvf.com/content/ICCV2021/papers/Sucar_iMAP_Implicit_Mapping_and_Positioning_in_Real-Time_ICCV_2021_paper.pdf>
107. [CVPR'22] NICE-SLAM: Neural Implicit Scalable Encoding for SLAM - GitHub, 访问时间为 六月 28, 2025， <https://github.com/cvg/nice-slam>
108. NICE-SLAM: Neural Implicit Scalable Encoding for SLAM | Request PDF - ResearchGate, 访问时间为 六月 28, 2025， <https://www.researchgate.net/publication/363906504_NICE-SLAM_Neural_Implicit_Scalable_Encoding_for_SLAM>
109. CVPR Poster SNI-SLAM: Semantic Neural Implicit SLAM - CVPR 2025, 访问时间为 六月 28, 2025， <https://cvpr.thecvf.com/virtual/2024/poster/31285>