

2.3 各研究内容的具体技术路线

图 1 展示了本项目总体技术路线。各项研究内容的具体技术方案描述如下。

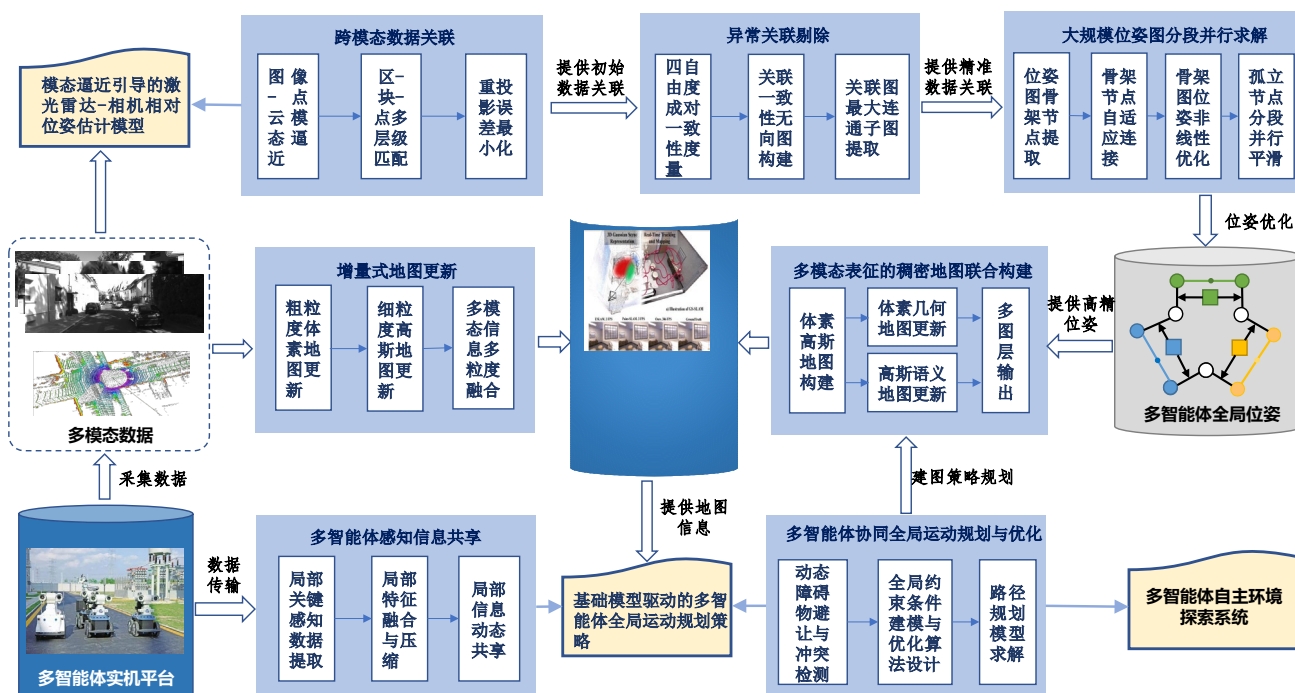


图 1：本项目总体技术路线。

(1) 跨模态数据关联

本项目提出了一种模态逼近引导的图像-点云相对位姿多层级估计模型（后文简称位姿估计模型）。图像-点云位姿估计任务主要面临两大难题：1）由于图像和点云的模态差异，导致跨模态特征提取难度大；2）传感器的感知范围不同，加上数据密度有差异，使得 2D-3D 匹配的质量欠佳。为了解决这些问题，位姿估计模型通过图像深度估计和点云体素化，令两种模态相互逼近，有效地消解了图像和点云在几何结构信息获取和特征空间上的差异，进而提取表征能力强的跨模态特征。此外，位姿估计模型还融入了“区-块-点”匹配策略。该策略按照逐步缩小匹配区域的思路来确定匹配的 2D 像素和 3D 点，大幅提升了 2D-3D 匹配的质量。课题组提出的图像-点云相对位姿多层级估计模型网络架构如图 2 所示。该网络将一对共视的相机 RGB 图像和激光雷达点云作为输入，采用多层匹配的方式确定匹配的像素-点云点对，最终输出激光雷达和相机之间的相对位姿。

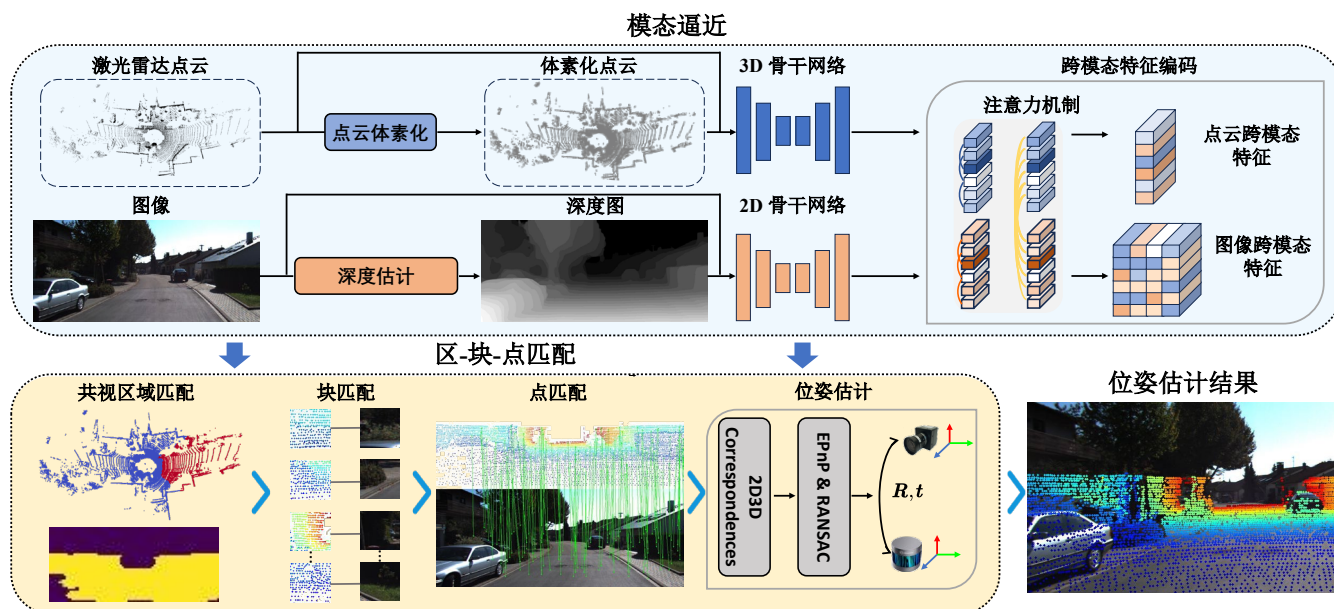


图 2：本项目提出的模态逼近引导的图像-点云相对位姿多层级估计模型的运行管线。

给定一对图像 $\mathbf{I} \in \mathbb{R}^{W \times H \times 3}$ 和点云 $\mathbf{P} \in \mathbb{R}^{N \times 3}$ (W 、 H 分别是图像的宽和高, N 是点云点的数量), 位姿估计模型的任务是估计二者之间的相对位姿 $\mathbf{T} \in \text{SE}(3)$ ($\{\mathbf{T}=[\mathbf{R}|\mathbf{t}], \mathbf{R} \in \text{SO}(3)|\mathbf{t} \in \mathbb{R}^3\}$)。位姿估计模型将该任务建模为一个 PnP (Perspective-n-Points) 问题, 即首先寻找匹配的 3D 点与 2D 像素, 再根据匹配点-像素对建立重投影方程, 进而求解相对位姿。该求解过程可被表述为:

$$\mathbf{T}^* = \underset{\mathbf{T}}{\text{argmin}} \sum_{\mathbf{u}_i, \mathbf{p}_i \in \mathcal{M}} \|\pi(\mathbf{K}, \mathbf{T}, \mathbf{p}_i) - \mathbf{u}_i\|^2 \quad (1)$$

其中, π 是从三维空间投影至图像平面的投影函数, $\mathbf{K} \in \mathbb{R}^{3 \times 3}$ 是相机内参数矩阵, $\mathcal{M} = \{(\mathbf{u}_i, \mathbf{p}_i) | \mathbf{u}_i \in \mathbb{R}^2, \mathbf{p}_i \in \mathbb{R}^3\}$ 是 2D-3D 匹配集合, \mathbf{u}_i 表示像素, \mathbf{p}_i 是与 \mathbf{u}_i 对应的点云中的 3D 点。

构建目标函数式 1 的关键在于要建立准确的 2D-3D 匹配关系集合, 即要获得准确的 \mathcal{M} 。下面介绍课题组设计的模态逼近引导的图像-点云相对位姿多层次估计模型的 2D-3D 匹配建立过程, 主要包括模型概述、模态逼近模块和“区-块-点”匹配策略三部分。

• 模型概述

如图 2 所示, 为了构建精准的 2D-3D 对应关系, 位姿估计模型采用了“逼近-融合-匹配”的架构, 主要由模态逼近模块和“区-块-点”匹配策略构成。模型运行时, 首先借助图像深度估计和点云体素化消除图像和点云之间的模态差异。通过图像深度估计, 图像也能获取与点云类似的场景几何结构信息; 点云体素化则让图像和点云可以采用相同的特征提取方式, 这为后续处理奠定了基础。随后, 模型使用注意力机制对齐特征空间, 进一步强化所提取的跨模态特征的表征能力。最后, 基于对齐后的跨模态特征, “区-块-点”匹配策略按照逐步缩小匹配范围的方式, 建立起 2D 像素与 3D 点之间的对应关系, 从而实现精准匹配。

• 模态逼近

课题组提出的模态逼近模块主要由图像深度估计模型、点云体素化网络和跨模态特征编码模块组成。通用图像深度估计模型 DepthAnything 被用来估计图像深度。将预测的深度图与原图像拼接构建伪 RGB-D 图像 $\mathbf{I}_d \in \mathbb{R}^{4 \times W \times H}$, 使用 2D 骨干网络 $\mathcal{F}_{2D}(\cdot)$ 提取相应特征 $\mathbf{F}_I = \mathcal{F}_{2D}(\mathbf{I}_d)$ 。

为了增强跨模态特征的表征能力, 我们对点云进行体素化处理, 使其具备与图像相似的数据拓扑结构。同时, 考虑到体素化所导致的信息丢失问题, 我们设计了一个具有两个分支的 3D 主干网络, 即点分支 $\mathcal{F}_p(\cdot)$ 和体素分支 $\mathcal{F}_v(\cdot)$ 。基于该网络, 点云的局部特征和全局特征 (分别用 \mathbf{F}_p^{local} 、 \mathbf{F}_p^{global} 表示) 可通过点分支获得: $\mathbf{F}_p^{local}, \mathbf{F}_p^{global} = \mathcal{F}_p(\mathbf{P})$; 体素化点云特征则由体素分支获取: $\mathbf{F}_v = \mathcal{F}_v(\mathbf{P}_v)$ 。

在获得了相应模态特征 ($\mathbf{F}_I, \mathbf{F}_p^{local}, \mathbf{F}_p^{global}, \mathbf{F}_v$) 后, 基于注意力机制进行跨模态特征编码:

$$\begin{aligned} \mathbf{w}_I &= \text{softmax}(\psi(\mathbf{F}_I, \mathbf{F}_p^{local})) \\ \tilde{\mathbf{F}}_P' &= \psi(\mathbf{w}_I \mathbf{F}_I, \mathbf{F}_p^{local}, \mathbf{F}_p^{global}) \\ \tilde{\mathbf{F}}_P &= \psi(\tilde{\mathbf{F}}_P', \mathbf{F}_v) \\ \mathbf{w}_P &= \text{softmax}(\xi(\mathbf{F}_I, \mathbf{F}_p^{local}, \mathbf{F}_p^{global})) \\ \tilde{\mathbf{F}}_I &= \xi(\mathbf{F}_I, \mathbf{w}_P \mathbf{F}_p^{local}, \mathbf{w}_P \mathbf{F}_p^{global}) \end{aligned} \quad (2)$$

其中, $\psi(\cdot)$ 和 $\xi(\cdot)$ 分别是点云和图像的特征编码函数, \mathbf{w}_I 和 \mathbf{w}_P 分别是图像和点云的特征权重。

• 区-块-点匹配策略

在获得了 (图像 \mathbf{I} 、图像跨模态特征 $\tilde{\mathbf{F}}_I$) 以及 (点云 \mathbf{P} 、点云跨模态特征 $\tilde{\mathbf{F}}_P'$, $\tilde{\mathbf{F}}_P$) 后, 区-块-点匹配的目标是建立精确的 2D 像素-3D 点匹配, 以便进行后续的位姿估算。图 3 展示了区-块-点匹配策略的工作管线。

区匹配。我们将共视区域识别建模为针对点和像素的二分类问题，并分别针对点和像素设计了不同的共视分类器 CVC_P 和 CVC_I 。考虑到该任务侧重于全局感知信息，聚合全局特征的跨模态特征 $\tilde{\mathbf{F}}_P'$ 和 $\tilde{\mathbf{F}}_I'$ 被用作 CVC_P 和 CVC_I 的输入。随后，共视分数 \mathbf{S}_P 和 \mathbf{S}_I 可获得： $\mathbf{S}_P = CVC_P(\tilde{\mathbf{F}}_P')$ ， $\mathbf{S}_I = CVC_I(\tilde{\mathbf{F}}_I')$ 。所获得的 $\mathbf{S}_P \in \mathbb{R}^{N \times 1}$ / $\mathbf{S}_I \in \mathbb{R}^{W \times H}$ 若大于阈值 γ_P/γ_I ，可判定相应的点/像素归属于共视区域，记作 $\mathbf{p}_c/\mathbf{u}_c$ 。与 \mathbf{p}_c 和 \mathbf{u}_c 相应的跨模态特征 $\tilde{\mathbf{F}}_P^c$ 和 $\tilde{\mathbf{F}}_I^c$ 也可以被获取。

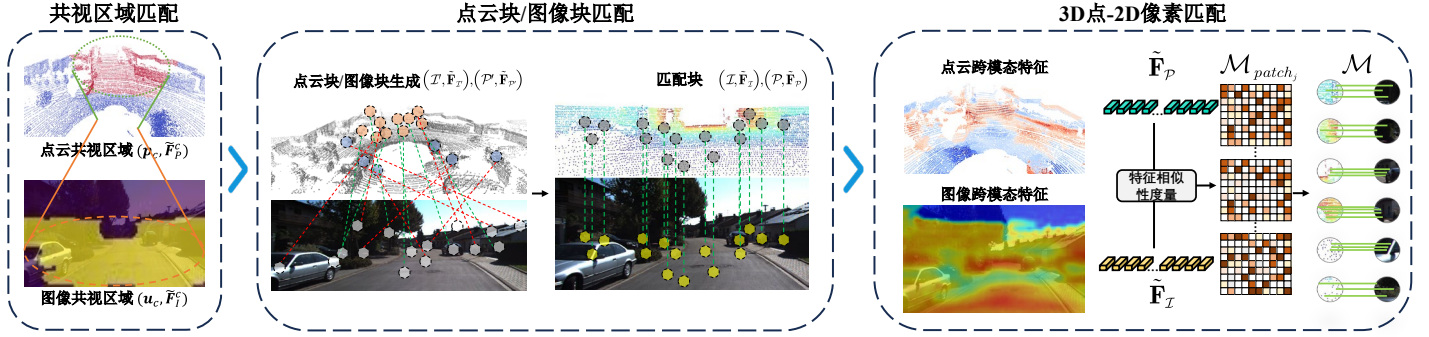


图 3：区-块-点匹配策略工作管线示意图。

块匹配。点云具有绝对尺度，与之不同的是，由于透视变换的存在，图像存在尺度模糊性。这就致使在一定距离范围内，当相机移动时，同一物体在图像中的大小看起来会发生改变，然而在点云中，该物体的大小却始终恒定。因此，像素与点的匹配并非是一一对应的关系。为了克服这种匹配模糊性，我们在区匹配和点匹配之间引入了一个额外的步骤：匹配点集与像素块。

通过网格划分可以获取图像块及其特征 $(\mathcal{I}, \tilde{\mathbf{F}}_I)$ ，其中 $\mathcal{I} = \{\mathcal{I}'_1, \dots, \mathcal{I}'_{K'} | \mathcal{I}'_{k'} \in \mathbb{R}^{3 \times W' \times H'}, k' \in [1, K']\}$ 。而点集 \mathcal{P} 的生成可被建模为一个分类问题。具体而言，以图像块的像素坐标作为标签，为点云构建一个块分类器 \mathcal{F}_{bc} 。借助 \mathcal{F}_{bc} ，能够得到分类结果 $\mathbf{S}_{P_i} = [S_{\mathcal{I}'_1}, \dots, S_{\mathcal{I}'_{K'}}]^T, S_{\mathcal{I}'_k} \in \mathbb{R}^{N \times 1}$ ： $\mathbf{S}_{P_i} = \mathcal{F}_{bc}(\tilde{\mathbf{F}}_P)$ 。这样一来，点集 \mathbf{P} 中每个点 \mathbf{p}_i 所匹配的属于 \mathcal{I} 的图像块 \mathcal{I}^* 可被估计：

$$\mathcal{I}^* = \underset{\mathcal{I}'_{k'}}{\operatorname{argmax}} (S_{\mathcal{I}'_1}, \dots, S_{\mathcal{I}'_{K'}}), k' \in [1, K'] \quad (3)$$

同时，根据 \mathbf{p}_i 的索引 i 可以提取对应的跨模态特征 $\tilde{\mathbf{F}}_{P_i}^c$ 。将与图像块匹配的点集及其相应特征记为 $(\mathcal{P}', \tilde{\mathbf{F}}_{P'})$ ， $\mathcal{P}' = \{\mathbf{p}_1, \dots, \mathbf{p}_{n'}\}$ ， $\tilde{\mathbf{F}}_{P'} = \{\tilde{\mathbf{F}}_{P_1}^c, \dots, \tilde{\mathbf{F}}_{P_{n'}}^c\}$ ，其中 n' 为与该图像块匹配的点的数量。

点匹配。通过区匹配和块匹配，我们获得了共视的点云区域 $(\mathbf{p}_c, \tilde{\mathbf{F}}_P^c)$ 、图像区域 $(\mathbf{u}_c, \tilde{\mathbf{F}}_I^c)$ 和匹配的点集 $(\mathcal{P}', \tilde{\mathbf{F}}_{P'})$ -图像块 $(\mathcal{I}, \tilde{\mathbf{F}}_I)$ 。基于以上信息，可以进一步建立像素-点的匹配。在区块 h 内，挑选处于二者交集的点和像素，即可获得小范围的匹配候选集 $\{\mathcal{P}_v, \mathcal{I}_k\}$ ($\mathcal{P}_v = \{\mathbf{p}_1, \dots, \mathbf{p}_n\}$ ， $\mathcal{I}_k = \{\mathbf{u}_1, \dots, \mathbf{u}_m\}$)， n 和 m 是点和像素的数量。该区块内的 2D-3D 匹配 \mathcal{M}_{block^h} 可通过计算跨模态特征间的余弦距离来确定，

$$\mathcal{M}_{block^h} = \begin{bmatrix} (\mathbf{u}_1, \underset{\mathbf{p} \in \{\mathbf{p}_1, \dots, \mathbf{p}_n\}}{\operatorname{argmin}} \delta(\mathbf{u}_1, \mathbf{p})) \\ \vdots \\ (\mathbf{u}_m, \underset{\mathbf{p} \in \{\mathbf{p}_1, \dots, \mathbf{p}_n\}}{\operatorname{argmin}} \delta(\mathbf{u}_m, \mathbf{p})) \end{bmatrix} \quad (4)$$

其中， $\delta(\cdot)$ 表示计算跨模态特征之间的余弦距离。对每个区块进行相同操作，最终的 2D-3D 匹配 \mathcal{M} 可被确定：

$$\mathcal{M} = \begin{bmatrix} \mathcal{M}_{block^1} \\ \vdots \\ \mathcal{M}_{block^h} \\ \vdots \\ \mathcal{M}_{block^H} \end{bmatrix} \quad (5)$$

(2) 大规模多源位姿图的数据关联筛选与分段并行求解管线

本项目拟基于跨模态数据关联构建全局位姿图，随后进行四自由度位姿图优化以完成关键帧位姿调整。受限于数据关联算法的局限性，原始数据关联组中往往会存在错误或不精确的异常关联。此外，在多智能体系统中，全局位姿图的规模一般都比较比较大。如果直接采用非线性优化的方式进行位姿解算，往往无法满足实时性需求。针对上述两方面的问题，在协同定位的过程中，我们首先借助四自由度成对一致性检测来筛选数据关联，将其中的异常数据关联剔除出去，以此保障协同定位的鲁棒性。随后，运用分治法思想，把全局位姿图求解任务进行拆解，通过分段并行求解的方式，实现快速且精准的位姿图优化求解。

给定关联算法输出的初始数据关联组，我们希望剔除其中的异常关联，并找到一组高质量数据关联用于构建全局位姿图。定义两帧 \mathcal{F}_i 和 \mathcal{F}_j 之间的关联为 l_{ij} 。给定一组原始关联观测值集合 \mathcal{L} ，需找到其内部最大的成对一致子集 \mathcal{L}_C ，然后清除 \mathcal{L} 内不属于 \mathcal{L}_C 的异常数据关联观测值。对于 \mathcal{L} 内任何两个关联 l_{ij} 和 l_{lk} 而言，如果它们满足以下条件：

$$C(l_{ij}, l_{lk}) < \gamma \quad (6)$$

则它们基于一致性度量指标 C 和阈值 γ 被定义为是一致的。基于这一“一致性”的定义，关联集合 \mathcal{L} 可以被表示为一个无向图 \mathcal{G} 。图 \mathcal{G} 内每个节点代表一个数据关联观测值。如果两个节点对应的关联观测值相互一致，则两节点被一条边连通。这样，找到一组两两一致的关联集 \mathcal{L}_C 便等价于寻找图 \mathcal{G} 的最大连通子图。该任务可以被转化为求解最大团的问题，目前已有诸多成熟的求解方案。

在视-惯里程计中，因重力加速度可被惯导感知，其在俯仰角与翻滚角两旋转自由度上可观，里程计在这两个维度上通常便不容易产生累积误差。因此，本项目所提出的成对一致性度量指标 C 是在四个自由度下完成计算的，这四个自由度包括平移的三自由度以及偏航角旋转自由度。具体来说，给定两个测量值 l_{ij} 和 l_{lk} ，它们的一致性评分可以被表示为：

$$C(l_{ij}, l_{lk}) = \|\mathbf{e}(l_{ij}, l_{lk})\|_2^{-1} \quad (7)$$

其中，误差 $\mathbf{e}(l_{ij}, l_{lk})$ 被定义为：

$$\begin{aligned} \mathbf{e}(l_{ij}, l_{lk}) &= [e_{ijlk}^{yaw}, (\mathbf{e}_{ijlk}^t)^T]^T \\ e_{ijlk}^{yaw} &= \hat{\theta}_{ij}^{yaw} + \theta_{jl}^{yaw} + \hat{\theta}_{lk}^{yaw} + \theta_{ki}^{yaw} \\ \mathbf{e}_{ijlk}^t &= [\hat{\mathbf{T}}_{ij} \mathbf{T}_{jl} \hat{\mathbf{T}}_{lk} \mathbf{T}_{ki}]_t \end{aligned} \quad (8)$$

其中， $\hat{\theta}_{ij}^{yaw}$ 和 $\hat{\theta}_{lk}^{yaw}$ 分别是关联 l_{ij} 和 l_{lk} 的相对偏航角， $\hat{\mathbf{T}}_{ij}$ 和 $\hat{\mathbf{T}}_{lk}$ 分别是 l_{ij} 和 l_{lk} 的相对位姿矩阵， θ_{jl}^{yaw} 和 θ_{ki}^{yaw} 分别是局部参考系下从 \mathcal{F}_j 到 \mathcal{F}_i 和从 \mathcal{F}_k 到 \mathcal{F}_i 的相对偏航角估计值， \mathbf{T}_{jl} 和 \mathbf{T}_{ki} 则为对应的相对位姿矩阵。符号 $(\cdot)_t$ 表示位姿矩阵的平移向量。

基于已通过成对一致性筛选的高质量数据关联，即可构建全局位姿图。随后，可使用所提出的两阶段位姿图分段求解算法完成全局位姿图优化。具体来说，我们将首先进行位姿图骨架节点的提取。随后

可自适应连接骨架节点以构建骨架图，并通过非线性优化完成骨架图位姿优化。最终，我们将通过并行平滑对剩余节点位姿进行求解以完成全局位姿图优化。图 4 为算法示意图。

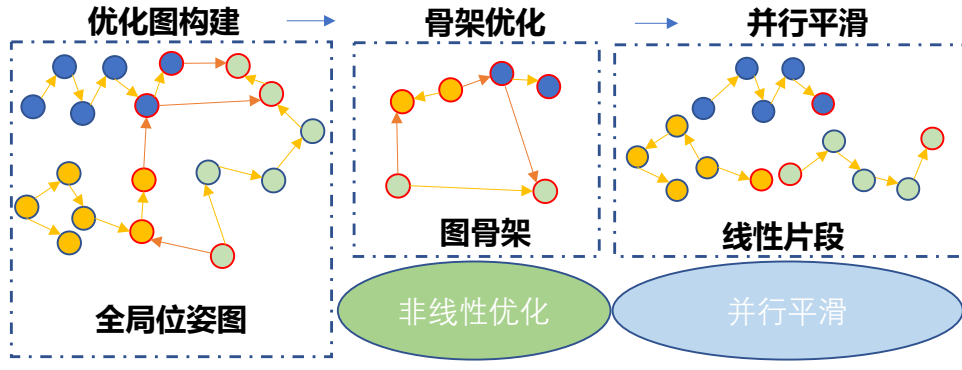


图 4：两阶段位姿图分段求解算法示意图。

在骨架提取之前，需要选择构成骨架子图基本结构的特殊关键帧。这些关键帧需满足下列要求之一：

- a) 关键帧被长期数据关联连接到其他帧；
- b) 关键帧与相邻帧间短期数据关联不准确。

其中，数据关联的准确性可基于相邻帧间的重投影误差进行判定。随后，通过自适应连接骨架节点来构建骨架图。在确定骨架子图内的节点与连接关系后，即可基于非线性优化技术实现相应关键帧的位姿求解。

在位姿图骨架被提取并优化后，其余关键帧将自然地分成多个不相关的段。随后，即可以并行方式分段实现剩余帧的位姿更新。为此，本项目提出了基于 EM (Expectation Maximization, 期望最大化) 算法框架的位姿平滑方案。以一段位姿节点 \mathcal{S}^j 为例， \mathcal{S}^j 可被定义为：

$$\mathcal{S}^j = \{\mathcal{F}^i | I^j < i \leq I^j + N^j\} \quad (9)$$

其中， I^j 是 \mathcal{S}^j 中首帧的前一帧的索引，而 N^j 是片段中关键帧的数量。在 \mathcal{S}^j 的平滑过程中，节点将以稠密连接模式连接。此外，除了 \mathcal{S}^j 中的关键帧，关键帧 \mathcal{F}^{I^j} 和 $\mathcal{F}^{I^j + N^j + 1}$ ，即段前一帧与段后一帧的位姿，也均参与位姿平滑。只是它们的位姿被设置为固定值，不会在平滑中被更新。至此，位姿平滑问题可归结为：

$$\min_{\mathcal{T}^j} \sum_{i=I^j}^{I^j + N^j + 1} \sum_{(i,k) \in \mathcal{E}^i} \|\mathbf{e}(\mathbf{T}_i, \mathbf{T}_k, \hat{\mathbf{T}}_{ik})\|_2^2 \quad (10)$$

其中， \mathcal{E}^i 是与 \mathcal{F}_i 相关的关键帧索引集合， \mathbf{T}_i (\mathbf{T}_k) 代表关键帧 \mathcal{F}_i (\mathcal{F}_k) 的位姿， $\hat{\mathbf{T}}_{ik}$ 是 \mathcal{F}_i 和 \mathcal{F}_k 之间的相对位姿约束， \mathcal{T}^j 代表 \mathcal{S}^j 中所有关键帧的相应位姿， $\mathbf{e}(\mathbf{T}_i, \mathbf{T}_k, \hat{\mathbf{T}}_{ik})$ 则是四自由度 (偏航角和平移) 位姿误差，其具体形式为：

$$\begin{aligned} \mathbf{e}(\mathbf{T}_i, \mathbf{T}_k, \hat{\mathbf{T}}_{ik}) &= [e_{ik}^{yaw}, (\mathbf{e}_{ik}^t)^T]^T \\ e_{ik}^{yaw} &= \theta_k^{yaw} - \theta_i^{yaw} - \hat{\theta}_{ik}^{yaw} \\ \mathbf{e}_{ik}^t &= \mathbf{R}_i(\mathbf{t}_k - \mathbf{t}_i) - \hat{\mathbf{t}}_{ik} \end{aligned} \quad (11)$$

其中， θ_i^{yaw} 、 θ_k^{yaw} 和 $\hat{\theta}_{ik}^{yaw}$ 分别是位姿 \mathbf{T}_i 、 \mathbf{T}_k 和 $\hat{\mathbf{T}}_{ik}$ 所对应的偏航角， \mathbf{R}_i 是 \mathbf{T}_i 所对应的旋转矩阵， \mathbf{t}_i 、 \mathbf{t}_k 和 $\hat{\mathbf{t}}_{ik}$ 则是相应位姿的平移向量。我们所提出的平滑方案基于一个重要的不等式，即：

$$\|\mathbf{e}(\mathbf{T}_i, \mathbf{T}_k, \hat{\mathbf{T}}_{ik})\|_2^2 \leq \frac{1}{2} (\|\mathbf{e}_i(\mathbf{T}_i, \hat{\mathbf{T}}_{ik})\|_2^2 + \|\mathbf{e}_k(\mathbf{T}_k, \hat{\mathbf{T}}_{ik})\|_2^2) \quad (12)$$

其中， $\mathbf{e}_i(\mathbf{T}_i, \hat{\mathbf{T}}_{ik})$ 和 $\mathbf{e}_k(\mathbf{T}_k, \hat{\mathbf{T}}_{ik})$ 分别被定义为：

$$\begin{aligned} \mathbf{e}_i(\mathbf{T}_i, \hat{\mathbf{T}}_{ik}) &= [\theta_i^{yaw} + \hat{\theta}_{ik}^{yaw} - {}_E\hat{\theta}_{ik}^{yaw}, \mathbf{R}_i(\mathbf{t}_i - {}_E\hat{\mathbf{t}}_{ik}) - \hat{\mathbf{t}}_{ik}]^T \\ \mathbf{e}_k(\mathbf{T}_k, \hat{\mathbf{T}}_{ik}) &= [\theta_k^{yaw} - {}_E\hat{\theta}_{ik}^{yaw}, \mathbf{R}_i(\mathbf{t}_k - {}_E\hat{\mathbf{t}}_{ik})]^T \end{aligned} \quad (13)$$

其中， ${}_E\hat{\theta}_{ik}^{yaw}$ 和 ${}_E\hat{\mathbf{t}}_{ik}$ 可以是任意常量。为了便于表示，本项目分别使用 \mathbf{e}_i 和 \mathbf{e}_k 来表示 $\mathbf{e}_i(\mathbf{T}_i, \hat{\mathbf{T}}_{ik})$ 和 $\mathbf{e}_k(\mathbf{T}_k, \hat{\mathbf{T}}_{ik})$ 。根据公式 12，可以得到问题式 10 的近似版本：

$$\min_{T^j} \sum_{i=I^j}^{I^j+N^j+1} \sum_{(i,k) \in \mathcal{E}^i} (\|\mathbf{e}_i\|_2^2 + \|\mathbf{e}_k\|_2^2) \quad (14)$$

问题式 10 与问题式 14 最优解相同当且仅当：

$$\begin{aligned} {}_E\hat{\theta}_{ik}^{yaw} &= (\tilde{\theta}_i^{yaw} + \tilde{\theta}_k^{yaw} - \hat{\theta}_{ik}^{yaw})/2 \\ {}_E\hat{\mathbf{t}}_{ik} &= (\tilde{\mathbf{t}}_i + \tilde{\mathbf{t}}_k - \mathbf{R}_i^T \hat{\mathbf{t}}_{ik})/2 \end{aligned} \quad (15)$$

其中， $\tilde{\theta}_i^{yaw}$ 、 $\tilde{\theta}_k^{yaw}$ 、 $\tilde{\mathbf{t}}_i$ 和 $\tilde{\mathbf{t}}_k$ 均为问题式 10 的最优解。由于这些最优解均不可获得，因此可采用 EM 框架来迭代平滑所有位姿，交替更新所有关键帧的位姿与近似最优解，以最终实现全局位姿优化的收敛。

(3) 多模态表征的稠密地图构建

本项目提出了几何、语义和外观信息统一的体素高斯地图表征模型。该方案结合了点云的离散扩展性和神经辐射场的可微优化性，将地图表征为一组不同粒度的体素高斯椭球，可以快速优化不同模态的信息，以构建全局一致的稠密地图。当获得了各异构传感器的相对位姿和智能体的全局位姿后，先利用 LiDAR 点云初始化八叉树体素地图，然后在不同体素内部构建三维高斯椭球。这种由粗粒度到细粒度的地图构建方式允许我们快速优化地图信息，并可以在体素层面进行无效信息的剪枝，在高斯椭球层面精细优化感兴趣区域，从而在增量式重建过程中自适应地滤除冗余信息，有效防止地图内存占用过高。课题组所提出的体素高斯地图表征模型的架构如图 5 所示。

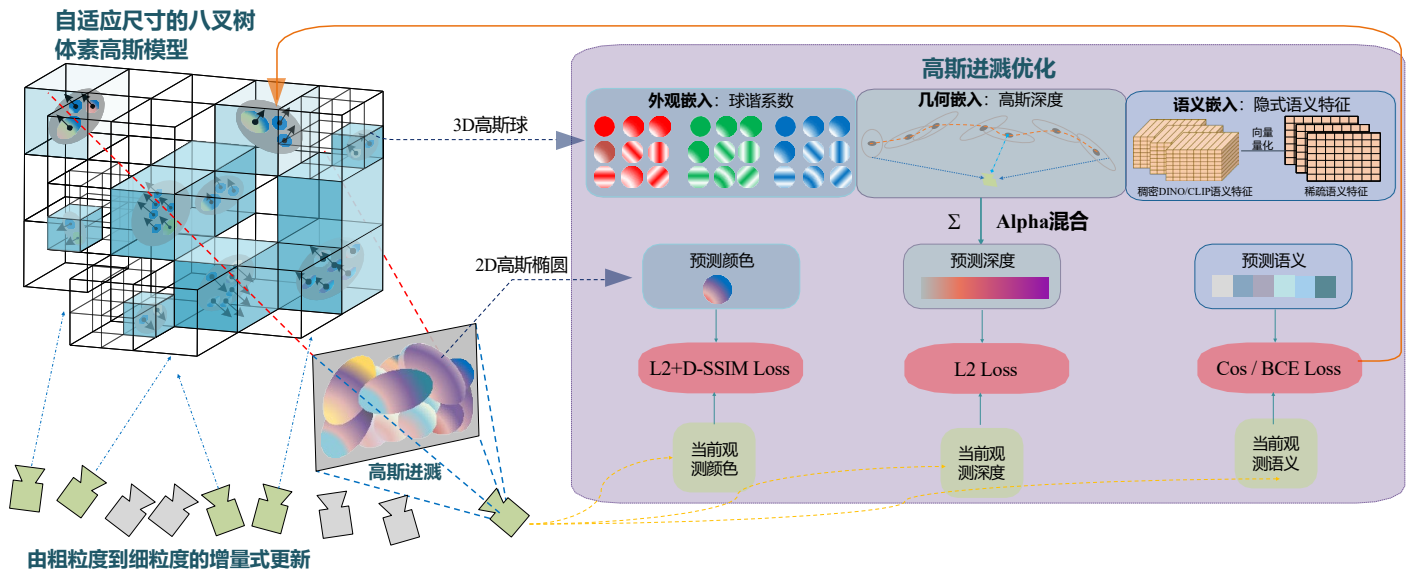


图 5：本项目提出的体素高斯地图表征模型的架构图。

• 体素高斯初始化

八叉树体素可以通过其平均位置 $\bar{\mathbf{p}}$ 、法向量 \mathbf{n} 和体素内部的协方差矩阵 $\Sigma_{\mathbf{n}, \bar{\mathbf{p}}}$ 来表征：

$$\bar{\mathbf{p}} = \frac{1}{N} \sum_{i=1}^N {}^w \mathbf{p}_i \quad (16)$$

体素的协方差矩阵 $\Sigma_{\mathbf{n}, \bar{\mathbf{p}}}$ 可以通过点 ${}^w \mathbf{p}_i$ 的分布来计算：

$$\Sigma_{\mathbf{n}, \bar{\mathbf{p}}} = \frac{1}{N} \sum_{i=1}^N \left({}^w \mathbf{p}_i - \bar{\mathbf{p}} \right) \left({}^w \mathbf{p}_i - \bar{\mathbf{p}} \right)^T \quad (17)$$

将协方差 $\Sigma_{\mathbf{n}, \bar{\mathbf{p}}}$ 的特征向量 \mathbf{n} 定义为假设高斯椭球平面的法向量，其对应的特征值表示该高斯椭球平面在每个方向上的分布。为了精细化表征几何表面，我们通过平面特征参数 η 细化体素。当 η 较大时，通过高斯椭球平面分布来更新参数 η ：

$$\eta = \frac{\lambda_{\min}}{\sqrt{\lambda_{\text{mid}}^2 + \lambda_{\min}^2 + \lambda_{\max}^2}} \quad (18)$$

其中， λ_{\min} 、 λ_{mid} 和 λ_{\max} 分别表示 $\Sigma_{\mathbf{n}, \bar{\mathbf{p}}}$ 的特征值的最小值、中值和最大值。

为每个点引入了一个缩放因子 α_i ，该因子由体素内的密度决定，可以被用于缩放三维高斯椭球：

$$\Sigma_{{}^w \mathbf{p}_i} = \alpha_i \Sigma_{\mathbf{n}, \bar{\mathbf{p}}} \quad (19)$$

综上，从八叉树体素中，我们可以得到细化的三维高斯椭球：

$$G_i^{3D}({}^w \mathbf{p}) = e^{-\frac{1}{2}({}^w \mathbf{p} - {}^w \mathbf{p}_i)^T \Sigma_{{}^w \mathbf{p}_i}^{-1}({}^w \mathbf{p} - {}^w \mathbf{p}_i)} \quad (20)$$

• 体素高斯模型

根据体素高斯初始化策略，场景最初被表示为一个密集的体素点云，其中填充了三维高斯椭球，每个高斯椭球包括三维位置 ${}^w \mathbf{p}_i$ 和协方差矩阵 $\Sigma_{{}^w \mathbf{p}_i}$ 。为了在地图中嵌入语义信息，我们为每个高斯椭球“挂载”语义特征向量 $\mathbf{f}_{{}^w \mathbf{p}_i}$ ，同时，其对应的几何深度 $d_{{}^w \mathbf{p}_i}$ 可以自然地用点 ${}^w \mathbf{p}_i$ 的 Z 轴坐标来表示。

对于外观信息，我们使用球谐函数为每个高斯椭球“挂载”谐波系数，从而可以渲染非朗伯表面。对于世界坐标系中的点 ${}^w \mathbf{p}_i$ ，假设某智能体系统的位姿为 ${}^w \mathbf{T}_{C_n}$ ，则点 ${}^w \mathbf{p}_i$ 从姿态 ${}^w \mathbf{T}_{C_n}$ 的视角来看，其观察方向为：

$$\begin{aligned} {}^{C_n} \mathbf{v}_i &= \frac{{}^w \mathbf{T}_{C_n}^{-1} \cdot {}^w \mathbf{p}_i}{\| {}^w \mathbf{T}_{C_n}^{-1} \cdot {}^w \mathbf{p}_i \|} \\ \theta &= \arccos \left(\frac{{}^{C_n} \mathbf{v}_{iz}}{\sqrt{{}^{C_n} \mathbf{v}_{ix}^2 + {}^{C_n} \mathbf{v}_{iy}^2 + {}^{C_n} \mathbf{v}_{iz}^2}} \right) \\ \phi &= \arctan 2({}^{C_n} \mathbf{v}_{iy}, {}^{C_n} \mathbf{v}_{ix}) \end{aligned} \quad (21)$$

球谐函数 $c(\theta, \phi)$ 是观察方向 (θ, ϕ) 的函数，

$$c(\theta, \phi) = \sum_{\ell=0}^{\infty} \sum_{m=-\ell}^{\ell} k_{\ell}^m \sqrt{\frac{2\ell+1}{4\pi} \frac{(\ell-m)!}{(\ell+m)!}} P_{\ell}^m(\cos \theta) e^{im\phi} \quad (22)$$

其中， $P_{\ell}^m(\cos \theta) e^{im\phi}$ 表示勒让德多项式， k_{ℓ}^m 是每个高斯椭球的球谐系数。

• 体素高斯模型的进溅优化

LiDAR 点云点 ${}^w \mathbf{p}_i$ 在图像平面上的投影 ${}^{C_n} \mathbf{q}_i$ 可以表示为：

$$C_n \mathbf{q}_i = \pi(w \mathbf{T}_{C_n}^{-1} \cdot w \mathbf{p}_i) \quad (23)$$

相机视角 $w \mathbf{T}_{C_n}$ 下所有的投影点 $C_n \mathbf{q}_i$ 构成了集合 \mathcal{R} 。为了训练预测真值图像 \mathbf{I}_n 的模型，使用如下的损失函数来优化高斯椭球的结构和球谐系数：

$$\mathcal{L}_c = (1 - \lambda) \sum_{n=1}^N \sum_{\mathbf{q} \in \mathcal{R}} \|\mathbf{I}_n(\mathbf{q}) - \hat{\mathbf{I}}_n(\mathbf{q})\| + \lambda \mathcal{L}_{\text{D-SSIM}} \quad (24)$$

其中， λ 是一个权重系数，用于平衡 MSE 和 D-SSIM 损失的贡献， N 为所有训练图像的数量。而预测的图像 $\hat{\mathbf{I}}_n(\mathbf{q})$ 可以通过 α 混合技术获得：

$$\hat{\mathbf{I}}_n(\mathbf{q}) = \sum_{i=1}^M \left[c_i \sigma_i G_i^{2\text{D}}(\mathbf{q}) \prod_{j=1}^{i-1} (1 - \sigma_j G_j^{2\text{D}}(\mathbf{q})) \right] \quad (25)$$

其中， $G_i^{2\text{D}}(\mathbf{q})$ 是通过局部仿射变换从 $G_i^{3\text{D}}(w \mathbf{p}_i)$ 映射得到的 2D 平面高斯椭圆， $\sigma_i \in [0, 1]$ 是高斯椭球的透明度， M 是影响该像素的 2D 高斯数量。

同理，几何深度可以通过如下损失函数进行优化：

$$\mathcal{L}_d = \sum_{n=1}^N \sum_{\mathbf{q} \in \mathcal{R}} \|\mathbf{D}_n(\mathbf{q}) - \hat{\mathbf{D}}_n(\mathbf{q})\| \quad (26)$$

其中， $\mathbf{D}_n(\mathbf{q})$ 为 LiDAR 点投影到图像平面上的深度，而 $\hat{\mathbf{D}}_n(\mathbf{q})$ 也可以通过 α 混合技术获得：

$$\hat{\mathbf{D}}_n(\mathbf{q}) = \sum_{i=1}^M \left[d_i \sigma_i G_i^{2\text{D}}(\mathbf{q}) \prod_{j=1}^{i-1} (1 - \sigma_j G_j^{2\text{D}}(\mathbf{q})) \right] \quad (27)$$

其中， d_i 为三维高斯椭球位置的 Z 轴值。

而对于语义信息优化，我们通过 α 混合技术获得与三维高斯椭球的语义特征 \mathbf{f}_i 对应的二维语义特征图：

$$\hat{\mathbf{S}}_n(\mathbf{q}) = \sum_{i=1}^M \left[\mathbf{f}_i \sigma_i G_i^{2\text{D}}(\mathbf{q}) \prod_{j=1}^{i-1} (1 - \sigma_j G_j^{2\text{D}}(\mathbf{q})) \right] \quad (28)$$

语义特征的优化可采用伪标签监督或语义特征监督方法。如果将 CLIP 或 DINO 提取的语义特征 $\mathbf{S}_n(\mathbf{q})$ 作为真值，则可以使用余弦相似性损失函数来监督语义优化：

$$\mathcal{L}_s = \sum_{n=1}^N \sum_{\mathbf{q} \in \mathcal{R}} \left(1 - \cos(\mathbf{S}_n(\mathbf{q}) \cdot \hat{\mathbf{S}}_n(\mathbf{q})) \right) \quad (29)$$

如果采用 SAM 等语义分割模型的伪语义标签 $\mathbf{S}_n^L(\mathbf{q})$ 作为语义真值，则先利用 CNN 解码器将语义特征图 $\hat{\mathbf{S}}_n(\mathbf{q})$ 解码为语义标签：

$$\hat{\mathbf{S}}_n^L(\mathbf{q}) = \text{softmax}(\text{CNN}(\hat{\mathbf{S}}_n(\mathbf{q}))) \quad (30)$$

然后使用交叉熵损失函数来监督语义优化：

$$\mathcal{L}_s = - \sum_{n=1}^N \sum_{\mathbf{q} \in \mathcal{R}} \left(\mathbf{S}_n^L(\mathbf{q}) \cdot \log(\hat{\mathbf{S}}_n(\mathbf{q})) + (1 - \mathbf{S}_n^L(\mathbf{q})) \cdot \log(1 - \hat{\mathbf{S}}_n(\mathbf{q})) \right) \quad (31)$$

• 由粗粒度到细粒度的快速增量更新算法

为了增量式更新多体素高斯地图，先利用粗粒度的八叉树体素进行快速的结构更新，然后再更新细粒度的三维高斯椭球的几何、语义和外观信息。

当接收到新的一帧时，八叉树体素先将该帧的点云信息融入到全局体素地图中，进行结构粗对齐。由于从 LiDAR 点云中得到的结构并非完美，它可能无法准确测量玻璃等表面，或者扫描的区域过度或不足。当几何特征尚未得到很好重建（即重建不足）时，在视图空间中可能会出现明显的错位。因此，需要根据体素密度阈值来识别需要细化的区域。对于这些区域，我们克隆相邻的三维高斯椭球，并联合几何、语义和外观来优化其位置以保证精确的体素结构。此外，对于因重复扫描而导致的过度密集的点云，可定期评估其透明度，并去除透明度较低的过度密集区域。这可有效减少地图中的冗余点，提高优化效率。

（4）基础模型驱动的多智能体局部信息共享和全局统筹决策框架

本项目提出了基础模型驱动的多智能体全局运动规划策略，如图 6 所示。该策略首先于边缘智能体节点识别关键视觉观测帧，并提取该帧中的视觉特征点。随后，该策略凭借当前关键帧中视觉特征点与历史关键帧中对应视觉特征点的关联关系压缩待传输的特征信息。通过上述两个步骤，该策略能够降低多智能体间信息共享的频率与数据量，从而在带宽受限的情况下实现高效可靠的协作感知。在接收到边缘智能体传输的局部特征信息后，中央智能体节点迭代更新全局地图，借助基础模型的强大推理能力判断全局稠密重建任务的完成程度，并选择性地调整边缘智能体节点的探索任务，从而实现多智能体系统全局任务的实时重决策。同时，边缘智能体节点接收来自中央智能体节点分配的探索任务，在局部区域规划行进路径，并依赖多源传感信息实现动态物体检测与主动避障，进一步提高了未知环境下多智能体协同运动规划的鲁棒性。

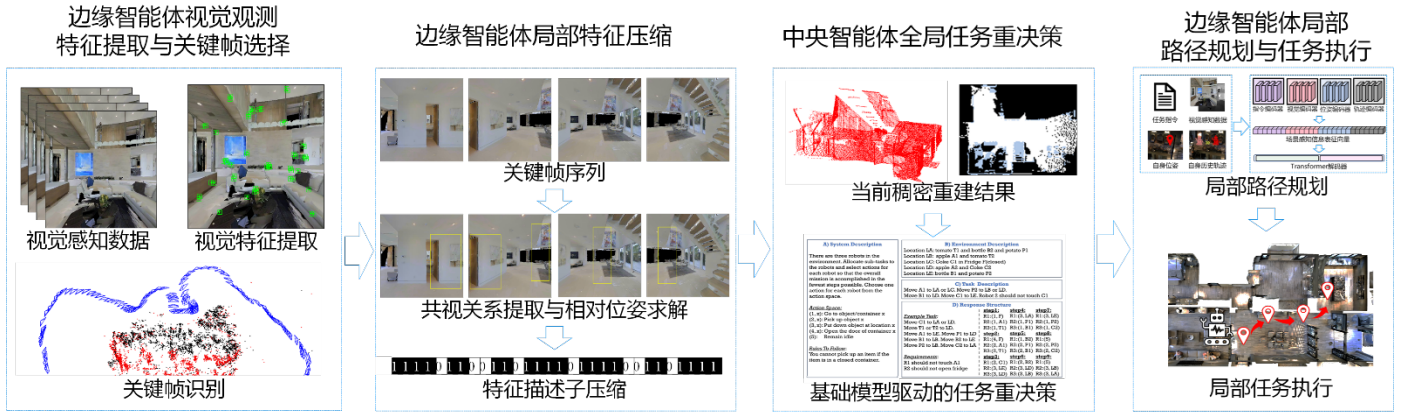


图 6：基础模型驱动的多智能体全局运动规划策略架构图。

• 视觉观测特征提取与关键帧识别

对于边缘智能体节点捕获的每一帧图像，我们首先从中提取 ORB 特征。该特征由两部分组成，特征点属性与特征点描述符。具体来说，对于第 n 张图像上的第 i 个特征点，其属性 $\mathbf{k}_{n,i}$ 可以表示为：

$$\mathbf{k}_{n,i}=[x, y, \sigma, \theta] \quad (32)$$

其中， x 和 y 表示该特征点在图片像素坐标系下的坐标， σ 表示特征点的特征尺度， θ 表示特征点的方向。同时，该特征点的描述符 $\mathbf{d}_{n,i}$ 可以表示为一个二进制字符串：

$$\mathbf{d}_{n,i} \in [0, 1]^D \quad (33)$$

其中， D 为特征编码长度。

通常情况下，从一帧图像中提取的特征点可分为两类：一类是已在历史观测中被计算过的特征，另一类是当前图像新增观测区域的特征。在对当前帧的特征点进行分类后，根据如下准则选择出关键帧：

- 平均视差：如果当前帧与上一关键帧的平均视差超过阈值，该帧将被视为新的关键帧；
- 历史特征点数量：如果当前帧中能够被追踪到的历史特征点数量低于阈值，该帧将被视为新的关键帧。

边缘智能体节点仅在检测到新的关键帧时，才尝试向中央智能体节点发送当前感知数据，从而降低了多智能体间信息共享的频率。

- **局部特征压缩**

可以利用两关键帧之间的空间相关性寻找两关键帧上特征点的对应关系。在多智能体信息共享过程中，边缘智能体只需要向中央智能体节点发送这些特征点在上一关键帧中的索引以及差异信息。具体来说，对于第 n 张图像上的第 i 个特征点，其编码成本 $R_{n,i}$ 可以表示为：

$$R_{n,i} = R_{n,i}^{\text{ref}} + R_{n,i}^{\text{diff}} + R_{n,i}^{\text{kpt}} \quad (34)$$

其中， $R_{n,i}^{\text{ref}}$ 是参考特征索引的编码成本， $R_{n,i}^{\text{diff}}$ 是描述差异信息的编码成本， $R_{n,i}^{\text{kpt}}$ 是该特征点属性信息的编码成本。差异信息由当前特征的特征描述符 $\mathbf{d}_{n,i}$ 与参考特征的特征描述符 \mathbf{d}_{ref} 经过二进制异或得到。因此， $R_{n,i}^{\text{diff}}$ 可由如下公式计算得出：

$$R_{n,i}^{\text{diff}} = -(D - h_{n,i}^{\text{ref}}) \cdot \log_2(p_0) - h_{n,i}^{\text{ref}} \cdot \log_2(1 - p_0) \quad (35)$$

其中， $h_{n,i}^{\text{ref}}$ 是当前特征的特征描述符 $\mathbf{d}_{n,i}$ 与参考特征的特征描述符 \mathbf{d}_{ref} 之间的汉明距离， p_0 是差异信息中每个二进制位为 0 的概率。

- **全局任务重决策**

在接收到来自边缘智能体节点传输的感知信息后，中央智能体节点使用这些数据更新全局地图，判断全局稠密重建任务的进度，并选择性地更新边缘智能体节点的探索任务。考虑一个包含 $N \geq 1$ 个边缘智能体节点与一个中央智能体节点的多智能体系统，每个边缘智能体节点的状态变化可由如下公式表示：

$$\mathbf{p}_j(t+1) = \mathcal{F}_j(\mathbf{p}_j(t), s_j(t)), j \in 1, 2, \dots, N \quad (36)$$

其中， $\mathbf{p}_j(t)$ 是边缘智能体 j 在离散时间步 t 的状态， $s_j(t)$ 是该智能体接收到的任务指令， $\mathcal{F}_j(\cdot)$ 表示该智能体的状态转移函数。假设每个边缘智能体系统都是同质的，即它们拥有相同的候选动作空间 \mathcal{A} ，其中，每一个候选动作 $a \in \mathcal{A}$ 是边缘智能体节点可执行动作的文本表示。边缘智能体 j 在时间步 t 于坐标 \mathbf{x} 执行动作 a 的操作可以表示为 $s_j(a, \mathbf{x}, t)$ （简称为 $s_j(t)$ ），那么多智能体全局任务重决策可以表示为寻找当前状态 $\mathbf{p}(t) = [\mathbf{p}_1(t), \mathbf{p}_2(t), \dots, \mathbf{p}_M(t)]$ 下所有边缘智能体的下一个最优操作 $\mathbf{s}(t) = [s_1(t), s_2(t), \dots, s_N(t)]$ 。

为了实现针对未知环境下多智能体稠密重建的全局任务重决策，借助于基础模型的推理能力，我们将其建模成一个多选项的问答过程。其中，“问题”包含对于任务的文本描述 Φ 以及决策过程的历史信息，“选项”则是边缘智能体在当前状态下的可行决策 \mathcal{S} 的文本描述。该框架确保基础模型只能从有效可行的选项中进行选择，从而降低了产生无意义指令的风险。在接收到问题描述 Φ 、多智能体系统当前状态 $\mathbf{p}(t)$ 以及备选任务指令 \mathcal{S} 后，基础模型将输出各个决策的置信度。具有最高置信度的决策将被选择为当前时间步的最优决策。