

濟南大學

毕业论文外文资料翻译

题 目 一种基于感知质量的滚动

 优化无人机导航系统

学 院 机械工程学院

专 业 机械工程专业

班 级 机工 1709 班

学 生 徐磊

学 号 20170421282

指导教师 吴长忠

二〇二一 年 三 月二十九 日

2018 IEEE International Conference on Robotics and Automation
(ICRA), 2018, 2534-2541

一种基于感知质量的滚动优化无人机导航系统

Zhang, Zichao ; Scaramuzza, Davide

苏黎世大学

摘要：为了安全、准确地到达指定的目的地，微型飞行器需要能够避开障碍物，同时最小化其状态估计的不确定性。为了达到这个目的，我们提出了一种感知滚动优化方法。在我们的方法中，一个前向单目相机用于状态估计和建图。利用单目状态估计和地图系统的信息，我们生成一个候选轨迹库，并从感知质量、碰撞概率和到目标的距离等方面对其进行评估。然后根据这三个指标，作为使奖励函数最大化的轨迹，选择最佳的执行轨迹。据我们所知，这是第一个将主动视觉整合到滚动优化导航框架中，以实现目标的任务。我们在一个实际的四旋翼机上通过仿真和实际实验证明，与纯反应式导航系统相比，我们的主动式导航方法能够提高目标到达任务的状态估计精度，特别是在困难场景（例如，弱纹理）下。

1.介绍

微型飞行器（MAVs）既灵活又通用，适用于工业检测、农业和货物运输等各种任务。为了使无人机能够在未知环境中自主运行，需要可靠的车载状态估计。在用于状态估计的不同传感器中，摄像机重量轻且节能，因此，由于其有效载荷和电池寿命有限，因此非常适合于微型飞行器。

对于基于视觉的状态估计，众所周知，摄像机的运动对估计精度有重要影响^[1]。因此，在规划无人机的运动时，既要考虑以上的任务，又要考虑感知质量。

在这项工作中，我们专注于在避免环境中的障碍物的同时，以最高的精确度达到给定的目标。在部分未知环境中规划轨迹时，常用的方法是将全局规划器与滚动优化方法相结合：在全局轨迹执行过程中，使用局部规划器在以机器人为中心的局部环境地图中生成和搜索可行的、无碰撞的轨迹。虽然这种方法已经成功的用在一些研究中^{[2]、[3]、[4]、[5]、[6]}，但是他们没有将感知约束考虑在内。（例如，有利于纹理丰富的区域最小化状态估计的不确定性）

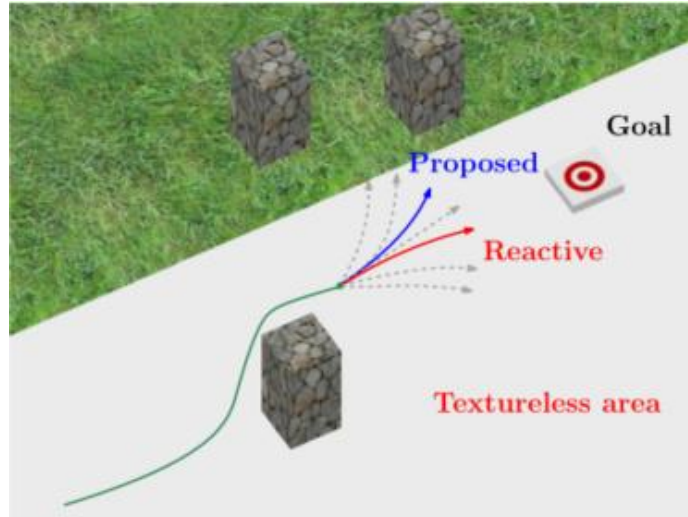


图 1: 所提出的感知感知滚动优化导航系统示意图。我们的方法能够选择一个合适的运动（蓝色线），它可以同时避开障碍物，到达指定的目的地并且最小化状态估计的不确定性。相比之下，纯反应式导航方案（红色线）会进入无纹理区域，导致状态估计误差大，无法到达指定的目的地。

很少有人考虑感知质量及其对视觉里程计精度的影响。选择使状态估计精度最大化的运动的问题称为主动 SLAM。这个问题通常是通过优化全局地图中的运动轨迹来解决的，这通常是一个昂贵的计算。相反，我们建议用滚动的方式来解决主动 SLAM 问题。特别地，我们将问题的范围限制在一个局部映射和一个短时间范围内，并且不断地重新计算一个合适的局部轨迹来进行状态估计。这样，可以有效地解决主动撞击问题，并且可以自然地整合到一个滚动优化中。

为了证明该方法的有效性，我们实现了一个基于感知质量的滚动优化导航系统。具体来说，我们使用单目里程计和建图系统进行状态估计和建图。我们没有直接优化运动参数，而是使用一种有效的轨迹生成方法（最小加加速度^[7]）在短时间内生成候选轨迹库。对于每一个轨迹，我们利用单目里程测量和绘图系统的信息来评估其感知质量、碰撞概率以及与目标的距离。然后选择执行的轨迹作为基于这三个指标的奖励函数最大化的轨迹。轨迹生成和评估过程是在线重复的。

1.1 相关工作

（1）无人机的滚动视觉规划：滚动视觉规划已被广泛用于在线生成无碰撞轨迹。刘等。^[5]开发了一个滚动视觉规划框架，在一个安全的飞行走廊内持续规划轨迹。陈等人。^[2]还通过对自由空间使用类似的道路表示来优化轨迹。不同的是，Landry 等人。^[4]将自由空间表示为凸体，并强制每个轨迹段保持在凸体中以避免碰撞。Mohta 等人。^[6]首先规划一条由直线段组成的安全路径，然后在有限的时间范围内通过加强轨迹接近安全路径来优化安全轨迹。

不同于求助于优化的方法。基于抽样的方法也得到了广泛的研究。Florence 等人。^[8]生成有限时间范围内的候选轨迹库，并根据成本函数（包括碰撞风险和到指定目的地的距离）选择要执行的轨迹。Matthies 等人。^[3]开发了一种快速探索随机树（RRT）

规划器，通过将采样轨迹投影到视差空间来进行碰撞检测。

上述方法主要是为无碰撞运动规划而设计的。然而，他们中没有一个考虑到感知质量，正如我们后来所展示的，这在包含视觉退化区域（即纹理差）的环境中极为重要。

(2) 无人机的主动 SLAM: 针对无人机提出了不同的主动 SLAM 方法。Mostegel 等人。^[9] 设计了一套启发式指标来评估定位质量和地图生成的可能性。根据当前的状态估计质量，他们的控制方案决定是否最大化定位质量或地图生成的可能性。Sabdat 等人。^[10] 提出了一种结合可见特征数量和可视表面视角的感知质量度量，并将该度量纳入快速探索随机树（RRT*）^[11] 中进行规划。Alzugaray 等人。^[12] 首先根据摄像机接近障碍物特征时姿态估计误差较小的直觉，对障碍物附近的位置进行采样。然后根据采样位置进行路径规划。不同于以往的基于几何信息的方法，Costante 等。^[13] 另外还结合了场景的光度信息来计算摄像机的定位不确定度。最近的一些研究还考虑了不同的传感器配置，例如全球定位系统（GPS）^[14] 和惯性测量单元（IMU）^[15]。作为一个补充问题，研究人员还提出了最小化重建不确定性的方法^{[16][17]}。

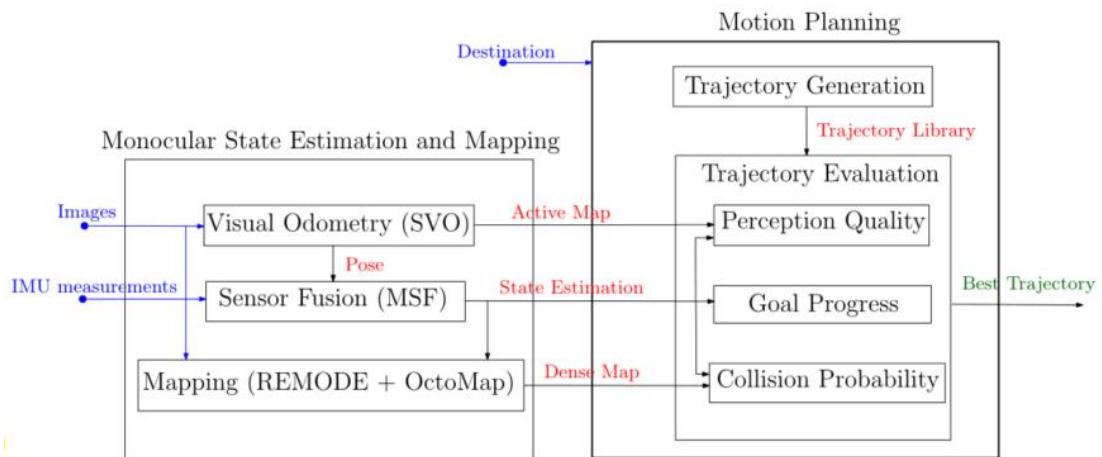


图 2：系统概述系统输入标识为蓝色，模块间通信标识为红色，输出标识为绿色

虽然上述方法维护了一个全局规划地图，但最近的一些工作侧重于基于短期精度的规划。Rong 等人。^[18] 用经验可观测性格拉姆矩阵评价短期知觉质量。他们证明了他们的测量方法能够成功地反映视觉退化区域感知质量的变化。Papachristos 等人。^[19] 提出了知觉感知探索的两步规划策略。他们首先规划了一个在探索空间中最大化信息增益的视图，然后第二个规划者对视图进行局部采样并选择状态估计不确定性最小的视图。该方法与文献^[19]相似，我们也在有限的时间范围内评估候选运动的感知质量。差异有两层。首先，我们把重点放在到达指定目的地的任务上，而不是探索。其次，在他们的方法中，避障是在自由空间中规划完成的，而我们的方法是在一个统一的滚动优化导航框架中考虑的。

1.2 稿件及提纲

据我们所知，这是第一个将主动视觉整合到滚动优化导航框架中，以实现目标的任务。我们通过仿真和实际实验证明，与纯反应式导航系统相比，我们的主动方法可以提高状态估计的精度，特别是在困难的场景中（例如，弱纹理）

论文的其余部分结构如下。在第二节中，我们对感知感知滚动优化导航系统进行了概述。在第三节中，我们描述了我们的单目状态估计和建图系统。在第四节中，我们详细介绍了如何利用第三节中的系统信息来规划四旋翼的运动。为了证明所提出方法的有效性，我们在第五节中进行了仿真和实际实验。

2.系统概述

图 2 概述了所提出的系统，该系统包括单目状态估计和映射系统以及运动规划系统。

单目状态估计和建图系统为多个目的去负责提供无人机和构建地图的状态估计。我们首先使用 SVO^[20]来估计摄像机的 6 自由度(DOF)姿态。利用多传感器融合(MSF)软件^[21]将姿态估计与 IMU 测量进一步融合，得到正确的尺度和额外的速度估计，这是轨迹生成和控制所必需的。然后，将状态估计值和图像输入 REMODE 的变体^[22]，得到正面视图的密集三维地图。在使用稠密地图进行运动规划之前，我们利用 OctoMap^[23]来降低地图中的噪声。除了状态估计和稠密地图之外，我们还从 SVO 得到了一个活动地图，这将在第三节中详细介绍。

运动规划系统由轨迹生成模块和轨迹评估模块组成。我们使用一种有效的轨迹生成方法^[7]，根据当前状态估计生成候选轨迹库。然后，我们根据三个指标评估每个轨迹：

- 碰撞概率，基于 REMODE 和 Octomap 的密集 3D 地图（第 IV-C 节）；
- 感知质量，基于 SVO 的活动地图（第 IV-D 节）；
- 目标进度（目标距离函数），基于当前状态估计和给定目标（第 IV-E 节）

根据评估结果，我们选择最佳轨迹执行，并将期望状态发送给控制器。

3. 单目状态估计与建图

我们的单目状态估计和映射与^[24]中提出的类似。采用 SVO+MSF 进行状态估计，通过 REMODE 生成稠密点云进行避障。SVO 是一种非常有效的 VO 算法，适用于资源受限的系统，如无人机。另一方面，REMODE 最初设计为在图形处理单元(GPU)上运行。在^[24]中，作者提出了一些修改，以使 REMODE 能够在智能手机处理器上执行。我们请读者参考^[24]了解更多细节。

与^[24]相比，我们的系统在以下方面有所不同。首先，为了评估感知质量，我们还从 SVO 中提取一个活动地图。在内部 SVO 维护一组稀疏点，这些点可以分为两类：

标志和种子。标志点是从不同的帧多次观测到的三维点，它们的位置已经得到了很好的估计。相比之下，种子是三维点，其位置尚未精确估计。

直观地说，标志点对帧姿态估计的准确性贡献最大。因此，我们从与当前帧重叠的关键帧中提取可见的标志。我们将这些标志记为活动地图（参见图 3）。稍后，我们将展示如何使用活动地图来评估感知质量（第 IV-D 节）。其次，我们使用 Octomap 来进一步降低 REMODE 输出中的噪声。如果密集点云包含过多的离群值，轨迹评估模块会错误估计碰撞概率，导致不必要的避碰机动。图 4 展示了活动地图和稠密地图，分别用 MA 和 MD 表示

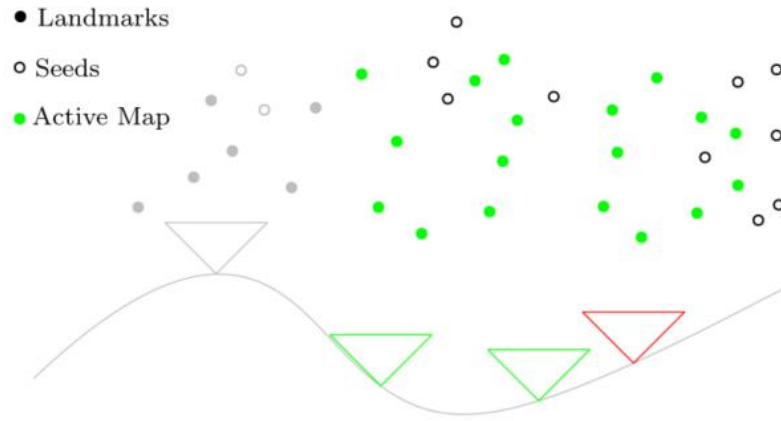


图 3：活动地图的提取。对于当前帧（红色三角形），我们找到与其重叠的关键帧（绿色三角形）。然后，在这些关键帧中可见的地标被提取为活动地图（绿色实心点）。活动地图中不考虑没有重叠（灰色三角形）和种子（圆）的关键帧。

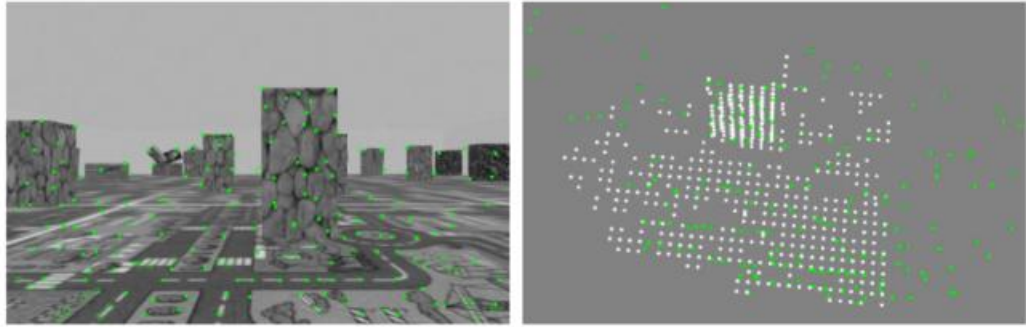


图 4：活动地图和稠密地图的示例。左边的一张是来自相机的图像，绿色实点是地标，圆圈是种子。右侧显示相应的活动地图（绿色）和稠密地图（白色）。

4. 轨迹生成与评估

在本节中，我们将描述如何生成候选轨迹库并选择要执行的最佳轨迹。

4.1 符号

前下标表示表示量的帧。我 $T_{ab} \in SE(3)$ 们用来表示框架 a 中 b 框架的刚体变换，它将框架 b 中的一个点转换为框架 a 中的一个点表示为 ${}_a\mathbf{p} = T_{ab} \cdot {}_b\mathbf{p}$ 。如果一个量/变换是在世界框架中表示的，为了简单起见，我们省略了（前）下标。

4.2 轨迹生成

我们用^[7]来生成候选轨迹。从概念上讲，轨迹生成过程用 $g(\cdot)$ 表示，是

$$f(t) \leftarrow g(\mathbf{p}_0, \mathbf{v}_0, \mathbf{p}_f, t_f).$$

\mathbf{p}_0 和 \mathbf{v}_0 是轨迹在世界坐标系中的初始位置和速度，来自当前状态估计。 \mathbf{p}_f 是轨迹的期望终点， t_f 是四旋翼跟随轨迹到达 \mathbf{p}_f 所需的时间。输出是函数 $f(t)$ 。对于 $t \in [0, t_f]$ ， $f(t)$ 给出了时刻 t 时轨道上的状态（位置、方向、速度）。

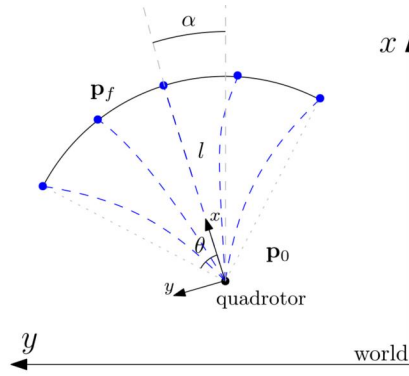


图 5：轨迹生成。我们在四边形转子前面的一条由 θ 和 l 定义的弧上均匀采样 N 个点。候选轨迹以蓝色显示。 α 是四旋翼的偏航角。

现在我们详细说明如何选择 \mathbf{p}_f 和 t_f 来生成轨迹库。我们通过选择四旋翼前面一段弧上的端点来规划轨迹，因为使用了前视摄像机我们希望在可见的方向移动，为此，我们需要知道圆弧的半径 l 和角度 θ ，如图 5 所示。当 l 只是一个设计参数时， θ 的计算公式为

$$\theta = \max(k_\theta \|\mathbf{v}_0\|, \theta_{\max}),$$

其中 k_θ 是常数。直观地说， θ 随着速度的增加而增加，直到达到最大值 θ_{\max} 。使用 θ 和 l ，我们均匀地采样弧上的端点。

对于 t_f ，我们使用以下公式：

$$t_f = \min\left(\frac{l}{\|\mathbf{v}_0\| + \Delta v}, \frac{l}{v_{\max}}\right),$$

式中 Δv 为常数， v_{\max} 为允许的最大速度。（3）意味着我们要增加速度直到达到最大值。

在生成 N 个候选轨迹后，我们以一个固定的时间间隔从每个轨迹上采样 J 个姿态。最后，对于每个候选轨迹，我们有一组采样姿态，可以表示为

$$C = \{c_i\}_{i=1}^N,$$

$$c_i = \{T_j\}_{j=1}^J,$$

其中 $T_j \in \text{SE}(3)$ 是在世界坐标系中表示的轨迹上的第 j 个姿态。由于四旋翼机

的偏航不受约束，我们只需将偏航设置为与速度方向相同，以便于避免障碍物（即，前视摄像头将始终朝运动方向看）

接下来，我们需要为给定的任务选择最佳轨迹 **c_{best}**。我们选择以下标准：

$$c_{best} = \arg \max_{c \in C} (1 - p_{col})(R_{perc} + R_{goal}) + p_{col}R_{col},$$

其中 p_{col} 是碰撞障碍物的概率（即密集地图）。 R_{col} 、 R_{perc} 和 R_{goal} 分别给出了与碰撞风险、感知质量和目标进度相关的奖励，下面我们将对此进行描述。为了简单起见，我们将去掉（5）中的下标，并将要求值的轨迹称为 c 。

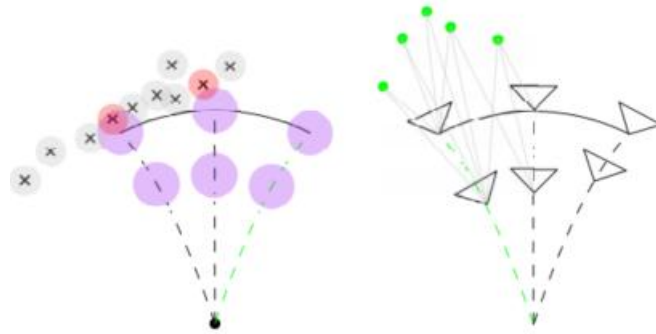


图 6：碰撞概率（左）和感知质量（右）的评估。左：十字是密集地图上的点，虚线曲线是候选轨迹。为了计算弹道上某一位置的碰撞概率，我们在稠密地图中找到最近的点（用红圈交叉），并用多元高斯分布计算碰撞概率。右：绿色实点是活动地图中的地标。对于所有沿轨迹采样的姿态，我们收集可见的路标并构造信息矩阵来评估感知质量。两种情况下的最佳轨迹都是绿色的。

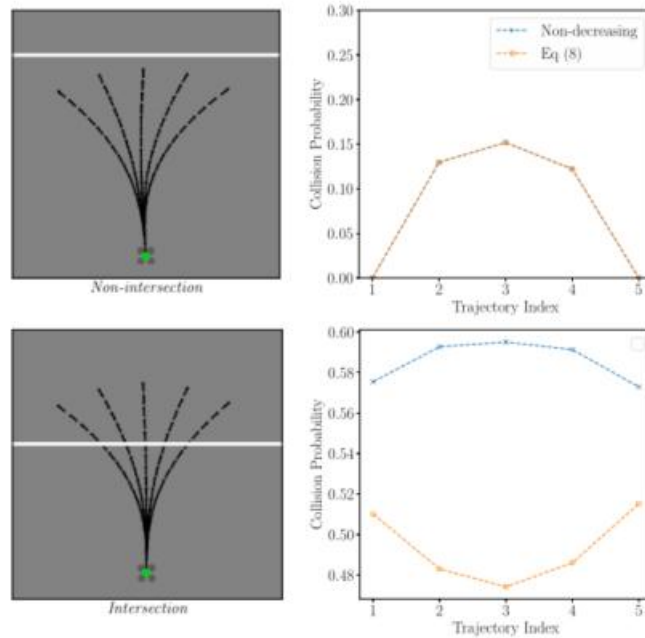


图 7：碰撞概率的计算。左栏显示模拟轨迹（黑色箭头）和障碍物（水平白带），右栏显示使用不同方法计算的碰撞概率。轨迹指数从左向右增加

4.3 碰撞概率

我们使用类似于^[8]的方法来计算碰撞概率，如图 6 所示。特别是，对于每个采样

位置 \mathbf{p}_j ，我们在稠密地图 MD 中找到最近的点：

$$\mathbf{d}_j = \arg \min_{\mathbf{d} \in M_D} \|\mathbf{d} - \mathbf{p}_j\|$$

然后计算了多元高斯分布的碰撞概率

$$p_j = V \times \frac{1}{\sqrt{2\pi}\Sigma} \exp\left[-\frac{1}{2}(\mathbf{d}_j - \mathbf{p}_j)^\top \Sigma^{-1}(\mathbf{d}_j - \mathbf{p}_j)\right],$$

式中， V 是我们要保持在四旋翼周围的安全球的体积。在 (8) 中， $\Sigma = (\sigma_d + \sigma_p)$ 3×3 ，其中 σ_d 和 σ_p 分别表示地图点和轨迹上位置的不确定性。那么轨道 c 的碰撞概率是

$$p_{\text{col}} = 1 - \prod_{j=1}^K (1 - p_j),$$

碰撞奖励是

$$R_{\text{col}} = k_{\text{col}},$$

其中 k_{col} 是一个负常数，表示我们需要对碰撞风险进行多少惩罚。

用上述方法计算碰撞概率时，有一个缺点，例如当轨迹超过平面障碍物时，障碍物后面位置的碰撞概率很小，这是不现实的。严格地说，我们没有关于障碍物后面空间的信息。然而，后面的空间也经常被占用。因此，与文献^[8]中的 (8) 不同，我们计算了 p_j 的碰撞概率

$$p_j = \begin{cases} \text{eq (8)} & p_j \geq p_{j-1} \\ p_{j-1} & p_j < p_{j-1} \end{cases}$$

基本上，我们假设沿轨道位置的碰撞概率是非递减的。为了证明 (11) 的效果，我们比较了当四旋翼接近平面障碍物时两种方法的碰撞概率，如图 7 所示。在第一行，我们可以看到当候选轨迹不与障碍物相交时，(8) 和 (11) 的结果相同。当有交叉时，如第二行所示，(8) 的结果表明中心轨迹与障碍物相撞的可能性较小。相比之下，我们的方法为中心轨迹分配了更高的碰撞概率。有人可能会争辩说，这两种方法无论如何都是不正确的，因为没有关于障碍物后面空间的信息。然而，当我们选择远离中心的轨迹时，我们的方法仍然是最有利的。

4.4 感知质量

在给定轨迹 c 和主动建图 $\mathbf{MA} = \{\mathbf{l}_k\}_{k=1}^K$ 的情况下，如果四旋翼跟踪轨迹，则需要量化位姿估计误差。位姿估计误差越小， R_{perc} 在 (6) 越大。为此，我们首先模拟采样姿态 $\{\mathbf{T}_j\}_{j=1}^J$ 的观测值，然后构造一个最小二乘问题，并从最小二乘问题的信息矩阵中估计估计误差。

对于每个姿势 T_j ，我们可以在 MA 中找到可见的路标，并将它们的索引统称为 O_j ，如图 6 所示。如果四旋翼移动到 T_j ，它的姿态通常是通过求解以下最小二乘问题来估计

$$T_j^* = \arg \min_T \sum_{k \in O_j} \|\text{proj}(T_{cb} T^{-1} l_k) - \tilde{u}_{jk}\|^2,$$

式中， T_{cb} 是从摄像机框架 c 到主体框架 b 的相对变换， $\text{proj}(\cdot)$ 是摄像机的投影函数，而 \tilde{u}_{jk} 是第 k 个地标的噪声观测值。(12) 通常用迭代优化方法如高斯-牛顿算法求解。从形式上讲，我们考虑以下问题

$$\xi^* = \arg \min_{\xi} \sum_{k \in O_j} \|\text{proj}(T_{cb}(T_j \text{Exp}(\xi))^{-1} l_k) - \tilde{u}_{jk}\|^2$$

T_j 将每次迭代更新为 $T_j \leftarrow T_j \text{Exp}(\xi)$ 。 ξ 是 $\text{se}(3)$ 中的元素， $\text{Exp}(\cdot)$ 将 $\text{se}(3)$ 中的元素映射到 $\text{se}(3)$ 。(13) 通过将 T_j 的当前估计值线性化来解决：

$$\xi^* = \arg \min_{\xi} \sum_{k \in O_j} \|\mathbf{u}_{jk} - \tilde{\mathbf{u}}_{jk} + \mathbf{J}_{\xi}^k \xi\|^2,$$

式中， $\mathbf{u}_{jk} = \text{项目}(T_{cb} T_j^{-1} l_k)$ 和 $\mathbf{J}_{\xi}^k = \partial \mathbf{u}_{jk} / \partial \xi$ 。(14) 可以用封闭形式求解。估计参数的协方差为^[25]

$$\Sigma_j = (\mathbf{J}_j^T \Sigma_u^{-1} \mathbf{J}_j)^{-1} = (\mathbf{H}_j)^{-1},$$

其中 \mathbf{J}_j 是由 \mathbf{J}_{ξ}^k 叠加而成， Σ_u 是 $\{\mathbf{u}_{jk}\}_{k \in O_j}$ 的协方差矩阵，当观测值独立且噪声水平相同时，它通常是对角线上具有相同值 σ_u 的对角矩阵。 \mathbf{h}_j 是信息矩阵

我们不仅要尽量减小轨迹上某个位姿的位姿估计误差，而且要尽量减小所有采样姿态的位姿估计误差。直觉上，沿轨迹精确的位姿估计有助于更好地三角化新的地标，有利于以后的位姿估计。因为活动地图只包含估计良好的点并且是固定的，所以姿态 $\{T_j\}_{j=1}$ 实际上是独立的（即，条件独立，以地标位置为条件）。因此，我们可以将整个信息矩阵写成

$$\mathbf{H} = \begin{bmatrix} \mathbf{H}_1 & 0 & \dots & 0 \\ 0 & \mathbf{H}_2 & \dots & 0 \\ 0 & \dots & \dots & 0 \\ 0 & \dots & 0 & \mathbf{H}_J \end{bmatrix},$$

全协方差是 $\Sigma = \mathbf{H}^{-1}$ ，这也是一个对角块矩阵。使用 (16)，我们使用 $D\text{-opt}$ 标准的模拟形式来计算报酬^[26]

$$R_{\text{perc}} = k_{\text{perc}} \exp(\log([\det(\mathbf{H})]^{\frac{1}{6J}})),$$

其中 k_{perc} 是确定感知质量权重的参数。

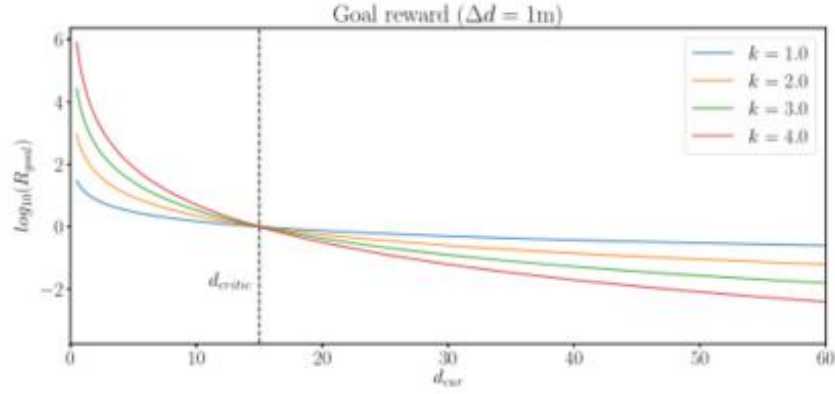


图 8: 当 $d_{critic}=15m$, $\Delta d=1m$ 时的目标奖励函数。对于相同的 Δd , 当四旋翼接近目标时, 目标奖励更高。当与目标的距离小于 d_{critic} 时, 目标奖励的增加速度更快。

4.5 目标进度

一种直接的方法来评估一条轨迹接近目的地的程度是使用从轨迹起点到终点的距离减少 Δd 。然而, 在实践中, 我们发现这种方法并不能很好地推广到不同的情况。一个原因是, 目标进度的评估应该与当前距目标的距离有关。例如, 将距离从 2 米缩短到 1 米, 应比从 100 米缩短到 99 米, 因此, 我们将目标奖励定义为

$$R_{goal} = k_{goal} \Delta d \times \left(\frac{d_{critic}}{d_{cur}} \right)^k,$$

其中 d_{critic} 是一个参数, 它控制目标附近强吸引区域的大小, k 控制权重的距离相关属性的强度。图 8 示出了不同距离处的目标奖励的示例。综上所述, 我们结合碰撞概率 (10)、感知质量 (17) 和目标进度 (18), 选择由 (6) 定义的最佳轨迹。在下一节中, 我们将在仿真和实际实验中证明所提出系统的有效性。

5. 实验

为了验证该方法的有效性, 我们在仿真环境和真实环境中进行了实验。与计算总奖励 (6) 相关的参数如表一。

表一: 模拟和真实世界实验的参数。它们共享相同的参数, 但 d_{critic} 和 l 取决于场景尺寸。

	k_{col}	k_{perc}	k_{goal}	d_{critic}	k	l (meter)
Simulation	-10000	1.5	10	15	3	5.0
Real-world	-	-	-	1	-	2.5

5.1 模拟

我们在不同的场景下进行了仿真测试, 以证明我们的系统不会过度适应特定的环境。从多个运行中总结统计结果。

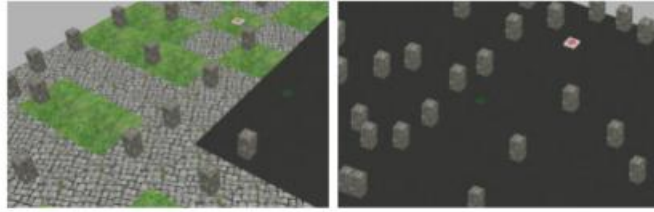


图 9：用于模拟的两个场景（L 形和障碍物）的近景。

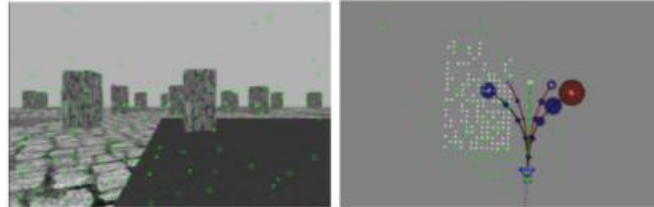


图 10：感知感知后退地平线导航在执行中的瞬间。左边是来自前视摄像头的图像，绿色的实心点和圆圈分别是 SVO 中的地标和种子。右边是轨迹生成和评估的可视化。在这种情况下，会生成五条轨迹。每个轨迹上的蓝色/红色球体表示采样姿势处的协方差，其中红色表示对应的姿势不受约束（信息矩阵是奇异的）。我们的方法正确识别了最右边的轨迹，它将四旋翼引导到无纹理区域，这是感知质量最差的。

我们利用 **Rotors** 模拟器^[27]。使用的场景的例子如图 9 所示，其尺寸在表中报告。模拟的无人机是一个 **AscTec** 蜂鸟四旋翼机和一个前视摄像头。在每个场景中，我们从稍微不同的起始位置启动四旋翼机，并命令它飞到指定的目的地 10 次。在我们的仿真中，四旋翼机很少撞上障碍物，即使状态估计误差也很大，这是只利用局部信息进行运动规划的优势。因此，我们使用状态估计误差来定义成功的标准。在每次运行中，一旦状态估计偏离地面真相超过 5 米，我们就终止执行并报告试验失败。一旦状态估计到给定目的地的距离小于 3 米，我们报告试验成功。为了消除视觉里程计初始化所带来的误差，我们利用模拟器中的 **groundtruth** 初始化 SVO。

我们比较了我们的方法和纯反应式导航方法（即没有 **Rpercin** (6)）的性能。图 12 示出了叠加在模拟场景上的轨迹和在行驶距离上的状态估计误差。图 11 示出了当模拟终止时的最终状态估计误差。表。我报告了每一个场景的成功试验次数。

第一个场景 L 形状由纹理强的区域（草和石头）和纹理较弱的区域（左下角的黑色区域）组成。我们可以观察到，纯反应式导航方法指令四旋翼机直接飞向目标，但轨迹经过视觉退化部分，导致状态估计误差较大。相比之下，

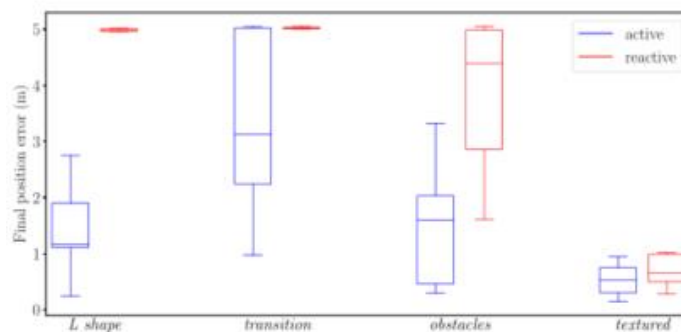


图 11：每次运行结束时的位置误差（要么达到目标，要么状态估计距离地面 5 米）

我们的方法避免了四转子进入纹理较少的区域，并且能够保持合理的状态估计。类似地，第二个场景转换在中间包含一个视觉退化区域，该区域位于起点和终点之间。虽然我们的方法能够使四旋翼机远离该区域并到达目的地，但反应式导航无法成功完成任务。图 10 显示了我们的系统正在执行的示例。

在前两个场景中，我们的方法比反应式导航有很大的优势。造成这种巨大差异的主要原因是这两个场景有两个明显分离的区域，纹理好和差（左下，其余为 L 形，中间和环绕过渡），视觉退化区域位于起始位置和目标之间。因此，反应式导航不可避免地会进入纹理较差的区域，导致性能较差。为了说明我们的方法在更一般的设置中的有用性，我们在另外两个场景中进行了测试。在第三场景障碍物中，地面几乎没有纹理，大部分视觉信息来自障碍物。在这个场景中，我们可以看到，我们的方法和反应式导航的轨迹与前两个场景差别不大，但是我们的方法的状态估计误差仍然明显较小，如图 12 和图 11 所示。如果仔细检查轨迹，我们可以看到，我们的方法引导四旋翼移动到更靠近障碍物，而不是反应式。这意味着在我们的方法中，具有视觉特征的障碍物既是排斥物又是吸引子：靠近这些障碍物会降低状态估计的不确定性，但也会增加碰撞风险。第四个场景纹理完全，没有明显的视觉退化部分。反应式导航和我们的方法在这个场景中都表现得很好，但是我们仍然可以从我们的方法中观察到更好的性能。

表二：10 个不同场景的试验成功。

Scene	Dimension (m)	Reactive	Ours
<i>L shape</i>	100 × 100	1	5
<i>transition</i>	100 × 100	0	5
<i>obstacles</i>	80 × 80	6	9
<i>textured</i>	60 × 60	9	10

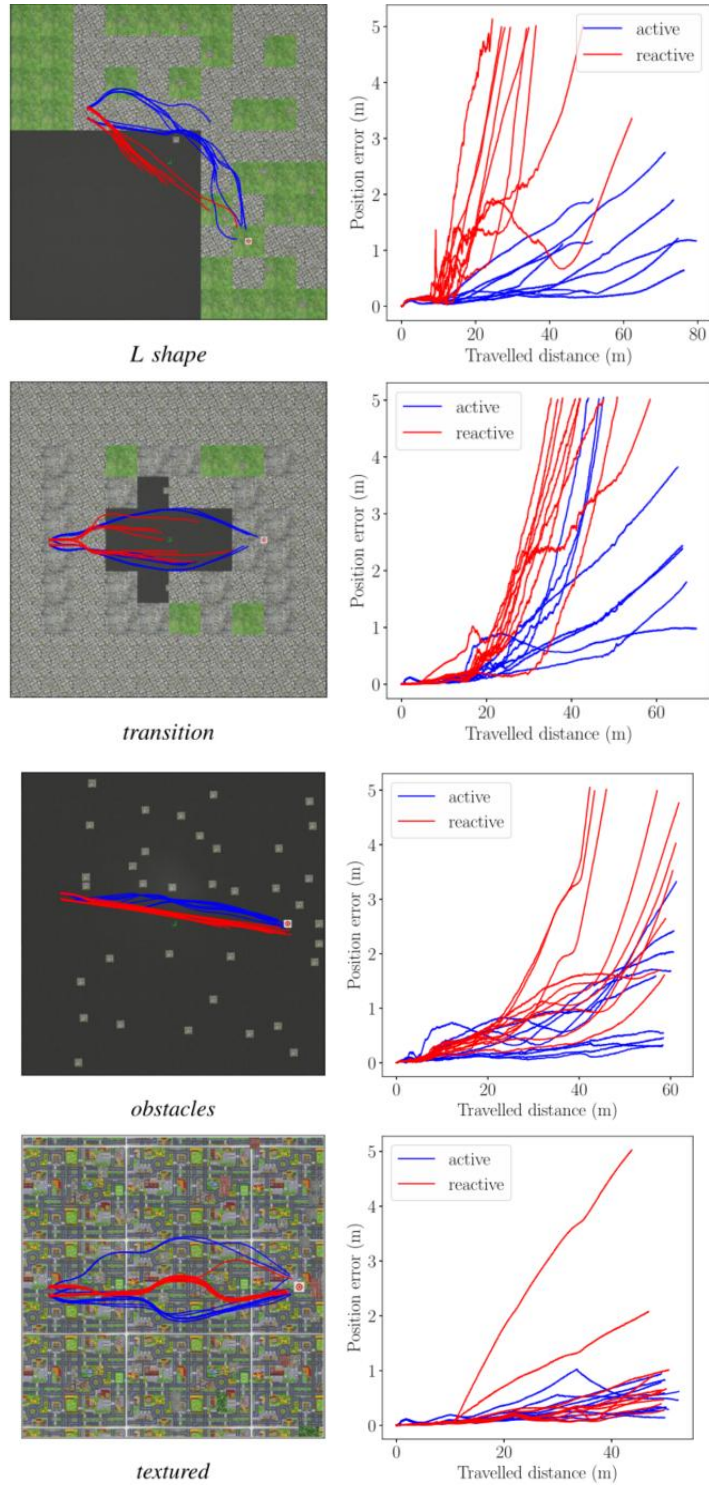


图 12：不同场景下叠加在仿真环境上的轨迹（左栏）和相对于行驶距离的位置误差（右栏）。每个导航策略在同一场景中执行 10 次，起始位置略有不同。

5.2 真实世界的实验

实际实验中使用的 Quadroot 配备了前视矩阵 mvBlueFOXMLC200w 单色全局快门相机。车载计算机为 ODROID-XU4。它还携带了一个来自 Pixhawk 的 PX4FMU 自

动驾驶仪板，包括一个 IMU。在我们的实验中，单目状态估计和映射系统是在船上完成的，而轨迹生成和评估则是在笔记本电脑上以 50 赫兹的频率进行计算的。



图 13: 用于测试我们的方法的真实场景。起点在左下角，目标在右上角。右边的图显示了四旋翼在我们的方法控制下的轨迹的俯视图。轨迹被纹理区域“吸引”，而不是直接从起始位置到目标。

我们在一个包含纹理和无纹理区域的场景中测试了我们的方法。图 13 显示了场景的照片和我们的方法中的几个示例轨迹。与仿真结果类似，我们的系统能够命令四旋翼机沿着一条信息更丰富的路径来达到目标。

6. 结论和今后的工作中

在这项工作中，我们建议将主动知觉整合到一个渐行渐远的视界中，以达成目标。特别地，我们设计了一个感知感知后退地平线导航系统，使用单目视觉里程计 SVO 和密集重建算法 REMODE 为导航提供必要的信息。利用这些信息，我们生成了一个轨迹库，并从碰撞概率、感知质量和目标进度等方面对其进行了评估，以选择 MA-Vs 的下一个运动，这自然地结合了不同的性能指标。我们通过大量的仿真和实际实验证明了该系统的有效性：除了能够避开障碍物之外，我们的感知感知后退地平线导航系统还能够选择运动以提高状态估计精度，尤其是在视觉退化的地区。

未来的工作将包括在不同的环境中进行进一步的实际验证，以更好地理解该方法的完整性和失败案例。生成信息丰富的运动原语也很有趣。

参考文献

- [1] Davison A J , Murray D W . Simultaneous localization and map-building using active vision[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2002, 24(7):865-880.
- [2] Jing C , Liu T , Shen S . Online generation of collision-free trajectories for quadrotor flight in unknown cluttered environments[C]// IEEE International Conference on Robotics & Automation. IEEE, 2016.
- [3] Matthies L , Brockers R , Kuwata Y , et al. Stereo vision-based obstacle avoidance for micro air vehicles using disparity space[C]// IEEE International Conference on Robotics & Automation. IEEE, 2014.

- [4] Landry B , Deits R , Florence P R , et al. Aggressive quadrotor flight through cluttered environments using mixed integer programming[C]// IEEE International Conference on Robotics & Automation. IEEE, 2016.
- [5] Liu S , Watterson M , Mohta K , et al. Planning Dynamically Feasible Trajectories for Quadrotors Using Safe Flight Corridors in 3-D Complex Environments[J]. IEEE Robotics & Automation Letters, 2017:1-1.
- [6] Mohta K , Watterson M , Mulgaonkar Y , et al. Fast, Autonomous Flight in GPS-Denied and Cluttered Environments[J]. Journal of Field Robotics, 2017, 35(1):101-120.
- [7] Mueller M W , Hehn M , D 'Andrea R . A computationally efficient algorithm for state-to-state quadcopter trajectory generation and feasibility verification[C]// Intelligent Robots and Systems (IROS), 2013 IEEE/RSJ International Conference on. IEEE, 2013.
- [8] Florence P R P R . Integrated perception and control at high speed. 2017.
- [9] Mostegel C , Wendel A , Bischof H . Active monocular localization: Towards autonomous monocular exploration for multirotor MAVs[C]// 2014 IEEE International Conference on Robotics and Automation (ICRA 2014). IEEE, 2014.
- [10] Sadat S A , Chutskoff K , D Jungic, et al. Feature-rich path planning for robust navigation of MAVs with Mono-SLAM[C]// 2014 IEEE International Conference on Robotics and Automation (ICRA). IEEE, 2014.
- [11] Karaman S , Frazzoli E . Sampling-based Algorithms for Optimal Motion Planning[J]. The International Journal of Robotics Research, 2011, 30(7):846-894.
- [12] Alzugaray I , Teixeira L , Chli M . Short-term UAV Path-Planning with Monocular-Inertial SLAM in the Loop[C]// 2017 IEEE International Conference on Robotics and Automation (ICRA). IEEE, 2017.
- [13] Costante G , Delmerico J , Werlberger M , et al. Exploiting Photometric Information for Planning Under Uncertainty[M]. 2017.
- [14] Hausman K , Preiss J , Sukhatme G , et al. Observability-Aware Trajectory Optimization for Self-Calibration with Application to UAVs[J]. IEEE Robotics & Automation Letters, 2016, PP(99):1-1.
- [15] Preiss J , Hausman K , Sukhatme G , et al. Trajectory Optimization for Self-Calibration and Navigation[C]// Robotics: Science & Systems. 2017.
- [16] Forster C , Pizzoli M , Scaramuzza D . Appearance-based Active, Monocular, Dense Reconstruction for Micro Aerial Vehicles[C]// Robotics: Science and Systems (RSS), Berkeley, 2014. 2014.
- [17] Isler S , Sabzevari R , Delmerico J , et al. An Information Gain Formulation for Active Volumetric 3D Reconstruction[J]. Proceedings - IEEE International Conference on Robotics and Automation, 2016.
- [18] Rong Z , Michael N . Detection and prediction of near-term state estimation degradation via online nonlinear observability analysis[J]. IEEE, 2016:28-33.
- [19] Papachristos C , Alexis K . Augmented reality-enhanced structural inspection using aerial robots[J]. IEEE, 2016:1-6.
- [20] Forster C , Zhang Z , Gassner M , et al. SVO: Semidirect Visual Odometry for Monocular and Multicamera Systems[J]. IEEE Transactions on Robotics, 2017, 33(2):249-265.
- [21] Lynen S , Achtelik M W , Weiss S , et al. A Robust and Modular Multi-Sensor Fusion Approach Applied to MAV Navigation[J]. IEEE, 2013.
- [22] Pizzoli M , Forster C , Scaramuzza D . REMODE: Probabilistic, monocular dense reconstruction in real time[C]// IEEE International Conference on Robotics & Automation. IEEE, 2014.
- [23] Hornung A , Wurm K M , Bennewitz M , et al. OctoMap: An efficient probabilistic 3D mapping framework based on octrees[J]. Autonomous Robots, 2013, 34(3):189-206.
- [24] Forster C , Faessler M , Fontana F , et al. Continuous On-Board Monocular-Vision-based Elevation Mapping Applied to Autonomous Landing of Micro Aerial Vehicles[C]// IEEE International Conference on Robotics & Automation. IEEE, 2015.
- [25] Hartley R , Zisserman A . Multiple view geometry in computer vision[C]// Cambridge University Press. Cambridge University Press, 2000

- [26] Carrillo H , Reid I , Castellanos J A . On the comparison of uncertainty criteria for active SLAM[J]. IEEE, 2012.
- [27] F. Furrer, M. Burri, M. Achtelik, and R. Siegwart, "Robot operating system (ros)," Studies Comp.Intelligence Volume Number:625, vol. The Complete Reference (Volume 1), no. 978-3-319-26052-5, p. Chapter 23, 2016, iSBN:978-3-319-26052-5