软件学报 ISSN 1000-9825, CODEN RUXUEW
*Journal of Software*,2016,27(10):2661−2675 [doi: 10.13328/j.cnki.jos.005083]

E-mail: jos@iscas.ac.cn
http://www.jos.org.cn
Tel: +86-10-62562563

# 基于光学图像的多粒度随动环境感知算法*

陈昊升, 张 格, 叶阳东

(郑州大学 信息工程学院,河南 郑州 450052)

通信作者: 叶阳东, E-mail: ieydye@zzu.edu.cn

**摘 要**: 针对快速三维建模中的室内外随动环境感知问题,提出一种基于光学图像的多粒度随动环境感知算法.该算法根据多种光学图像生成拟合真实三维环境的多粒度点云模型,然后通过概率八叉树压缩并统一表示已生成的多粒度三维模型.进而伴随相机轨迹每个时间节点,通过卡尔曼滤波动态融合多粒度点云模型的概率八叉树表示.最终生成唯一的时态融合概率八叉树三维模型,简称 TFPOM,使 TFPOM 能够在较少的噪声影响下以任意粒度动态拟合真实环境.该算法配合剪枝和归并策略能够适应多粒度融合和多粒度表示的环境建模要求,有效压缩环境模型存储空间,实现鲁棒的随动环境感知,便于基于环境模型的视觉导航,增强现实等应用.实验结果表明,该算法能够在以可穿戴设备为代表的内含多种异构光学图像传感器、低计算效能的平台上实时地得到充分拟合真实动态环境的多粒度 TFPOM,基于该模型的视觉导航具有较小的轨迹误差.

**关键词**: 随动环境感知;概率八叉树;多粒度;快速三维建模

**中图法分类号**: TP391

中文引用格式: 陈昊升,张格,叶阳东.基于光学图像的多粒度随动环境感知算法.软件学报,2016,27(10):2661−2675. http://www. jos.org.cn/1000-9825/5083.htm

英文引用格式: Chen HS, Zhang G, Ye YD. Optical image based multi-granularity follow-up environment perception algorithm. Ruan Jian Xue Bao/Journal of Software, 2016,27(10):2661−2675 (in Chinese). http://www.jos.org.cn/1000-9825/5083.htm

## Optical Image Based Multi-Granularity Follow-Up Environment Perception Algorithm

CHEN Hao-Sheng, ZHANG Ge, YE Yang-Dong

(School of Information Engineering, Zhengzhou University, Zhengzhou 450052, China)

**Abstract**: An optical image based multi-granularity follow-up environment perception algorithm is proposed to address the follow-up environment perception issue from indoor to outdoor in the field of rapid 3D modeling. The algorithm generates multi-granularity 3D point cloud models which perfectly fit the ground-truth according to different types of optical image. A probabilistic octree representation is proposed to uniformly express the 3D point cloud models. Finally, the expected TFPOM is generated through dynamic ground-truth fitting at any granularity, and probabilistic octree representation of multi-granularity point cloud models are dynamically fused through implementation of Kalman filter along with the camera trajectory. Benefiting from pruning and merging strategies, the proposed algorithm meets requirements of multi-granularity fusion and multi-granularity representation. As a result, the storage space of environment models can be effectively compressed and robust follow-up environment perception can be achieved, which are essential in environment model based visual navigation and augmented reality. Experiment results show that the algorithm can generate multi-granularity TFPOM which perfectly fits ground-truth in real time with fewer errors in model based navigation on platforms, such as wearable devices, that are equipped with multiple optical sensors and low computing capability.

**Key words**: follow-up environment perception; probabilistic octree; multi-granularity; rapid 3D modeling

* 基金项目: 国家自然科学基金(61170223, 61502434, 61502432)
Foundation item: National Natural Science Foundation of China (61170223, 61202207, 61502432)
收稿时间: 2016-01-20; 修改时间: 2016-03-25; 采用时间: 2016-05-09; jos 在线出版时间: 2016-08-08
CNKI 网络优先出版: 2016-08-09 15:38:19, http://www.cnki.net/kcms/detail/11.2560.TP.20160809.1538.020.html

基于光学图像的环境感知是指通过多种异构的光学图像传感器,例如单目相机、全景相机、双目相机、红外相机等对真实环境的描述构建高度拟合环境的三维模型.随动环境感知不同于以批处理模式离线构建环境模型的基于运动结构恢复(structure from motion,简称 SfM)的传统方法,它强调在相机运动过程中根据相机轨迹实时生成环境模型.随动环境感知问题实质上是一个基于视觉的同步定位与建图(vision based simultaneous localization and mapping,简称 visual SLAM)问题,通常使用基于滤波器的方法或基于关键帧的捆集调整法(bundle adjustment,简称 BA)求解.而传统的基于滤波器,如扩展卡尔曼滤波器的方法因受累计误差和协方差矩阵计算复杂度的限制,已逐步被基于关键帧的捆集调整法取代.

在 visual SLAM 领域,文献[1]首次提出一种使用单目广角相机作为图像采集设备、基于扩展卡尔曼滤波的 MonoSLAM 算法,为 visual SLAM 算法的可行性和易用性奠定了坚实的基础.文献[2,3]提出一种基于关键帧且定位和建图分离的单目 visual SLAM 算法 PTAM.PTAM 保持了 visual SLAM 结构,并使用运动结构重建相关算法对 MonoSLAM 进行改进.其中,分离后的建图工作使用捆集调整法代替扩展卡尔曼滤波,在具有更好的鲁棒性的同时避免了累计误差以及协方差矩阵计算量等限制.

伴随 visual SLAM 体系结构的逐步完善,相关工作的重心转移到如何使算法能够获取更加丰富的环境信息和具有更好的精确性以及更强的实时性上.其中,文献[4]提出一种逐个像素遍历处理的 PTAM 改进算法,算法虽然能够构建较为精确的三维稠密点云地图,但受计算复杂度约束,其应用场景仅限于较小的桌面场景.文献[5−7]提出了一种基于图像梯度不依赖于特定图像特征点的半稠密建图方法,并根据单目彩色相机、全景相机、多目立体相机分别提出了相关算法.这种方法相对于基于图像特征点的建图方法,可以较大程度地恢复场景细节,但是,由于半稠密区域所恢复的深度采用中心特征点深度近似方法,使得这类算法具有较大的固有和随机噪声.文献[8]提出了一种基于 ORB[9]算子的 ORB-SLAM 算法,该算法利用闭环检测机制减少了累计误差,在实际应用中具有较好的鲁棒性,但存在生成的环境模型较为稀疏、缺少较多细节等问题.文献[10]对文献[8]进一步加以补充,提出了 ORB 特征的半稠密表示.文献[11−14]通过主动红外深度相机构建精确稠密的点云地图,其中,文献[12]考虑到动态场景的恢复问题,文献[13]则关注先验模型库为定位与建图所带来的增益.受限于现有深度相机支持的最大有效工作景深一般仅为 4m 且易受室外自然红外线干扰,该类算法一般受限于室内场景.

环境三维模型存储表示是用于环境感知的 visual SLAM 中的重要一环,对算法精确性和时间效率起决定性作用.其中,文献[15,16]将空间离散划分为刚体的体素,这种方法存在存储空间占有大、需要预先确定环境的内容等问题.另一种被 visual SLAM 广泛使用的表示形式是点云形式[17,18],这种表示形式的主要缺陷在于无法明确表示状态为未知(摄像机尚未捕捉到的区域)和空闲(确定为未被实体占用的区域)的区域,对环境的动态性容忍度较低.文献[19]首次提出使用八叉树用于存储表示,之后,文献[20,21]开始关注是否处于被占用的二值状态.文献[22−24]提出一种概率方法以表示状态为占用和空闲的区域.上述八叉树表示没有解决存储消耗和置信度阈值问题.文献[25,26]设计了一种概率八叉树的表示和相关剪枝等策略,较好地解决了上述问题.同时,文献[27]还讨论了利用八叉树结构进行有效纹理映射的方法.

在实际的视觉同步定位与建图中,一般使用以可穿戴设备为代表的硬件平台.这些硬件平台大多具有内含多种异构光学图像传感器、低计算效能等特点.在环境场景由室内向室外转换时,难以由一种同构图像传感器统一感知.例如,在室内具有较高精度的红外相机由于其最小及最大可测距离的限制,无法适用于室外场景,如图 1 所示,需要双目相机或单目相机辅助.另外,各种异构相机衍生出相应不同粒度的环境建图方式.目前常用的方法有单目或双目相机的基于图像特征点的稀疏建图方法[1−3,8],单目、双目或全景相机的半稠密建图方法[5−7,10],单目、双目或主动深度相机的基于深度图的稠密建图方法[4,11−14]这 3 类.这些客观因素对统一表示多粒度地图模型提出了挑战.另一方面,在相机运动过程中相机姿态不断更新,相应的环境地图需要同步动态增量更新,此外还需要考虑不同图像传感器在不同时刻的工作状态,如运动模糊所造成的影响等.如何针对不同环境模型的内在特性给予相应的置信度进行有机融合和根据相机时间序列状态动态更新置信度的问题亟待解决.

本文在深入研究现有算法的基础上,提出了基于光学图像的多粒度随动环境感知算法.该算法采集多种异

构图像传感器及相应 visual SLAM 算法所生成的稀疏、半稠密、稠密多种粒度的异构三维点云模型,同时提出一种概率八叉树统一表示生成的若干三维模型.然后,通过卡尔曼滤波,在相机运动期间不断融合多种异构点云的置信度,并更新时态融合概率八叉树模型(temporal fused probabilistic octree model,简称 TFPOM),这种融合方法既可以保证环境模型的时空一致性,又可以满足后续增量更新的需求.同时,利用一种有效的剪枝和归并策略,在压缩模型存储空间时使环境模型能够以任意粒度动态拟合真实环境,最终实现鲁棒的随动环境感知.已有的 visual SLAM 技术均可借鉴本文算法,融合计算真实场景的动态时空模型,从而避免单源固有噪声和随机噪声的影响,以提高算法鲁棒性,并通过概率八叉树模型压缩特性和本文所提出的剪枝归并策略提高算法效率.本文给出其中一种融合稀疏、半稠密、稠密 visual SLAM 环境模型并给出统一表示的算法实现.据我们所知,目前尚未出现融合多源多粒度环境模型的相关算法.

(a) 彩色图 (b) 深度图 (c) 彩色深度配准图

Fig.1 Color image, depth image and registered image of infrared camera under the same scene. Registered area of the scene is highlighted by red ellipse

图 1 红外深度相机在同一场景下的彩色、深度和彩色深度配准图.图中椭圆标出区域是受相机景深限制唯一能够配准的区域

本文主要贡献如下:

(1) 对真实环境的若干异构点云模型提出一种统一的概率八叉树表示.这种统一的概率八叉树表示可以有效地表达状态为占有、空闲和未知的区域.然后为这种环境模型的概率八叉树表示提出了剪枝和归并策略,压缩模型存储空间,扩展模型多粒度表示.

(2) 通过为多种异构光学图像传感器提出一种合理的置信度概率模型,伴随相机运动,使用卡尔曼滤波器对上述环境模型的概率八叉树表示提供时态融合,进而生成 TFPOM,用于多粒度动态感知真实环境.

(3) 为算法更新过程提出一种模型增量更新的方法,这种方法具有较低的计算复杂度,可以在以穿戴式设备为主的低计算效能的设备上实时运行,便于随动环境感知.

## 1 相关知识

### 1.1 多粒度环境感知原理

当前 visual SLAM 所构建的环境模型根据体素的密度分为 3 种类型:稀疏模型、半稠密模型和稠密模型.相关的构建方法高度依赖于具有不同特性的传感器和相关算法实现.例如,单目相机一般用于构建稀疏模型和半稠密模型,双目和红外深度相机则用于构建稠密模型.现将本文所使用的模型构建方法中的相关原理描述如下.

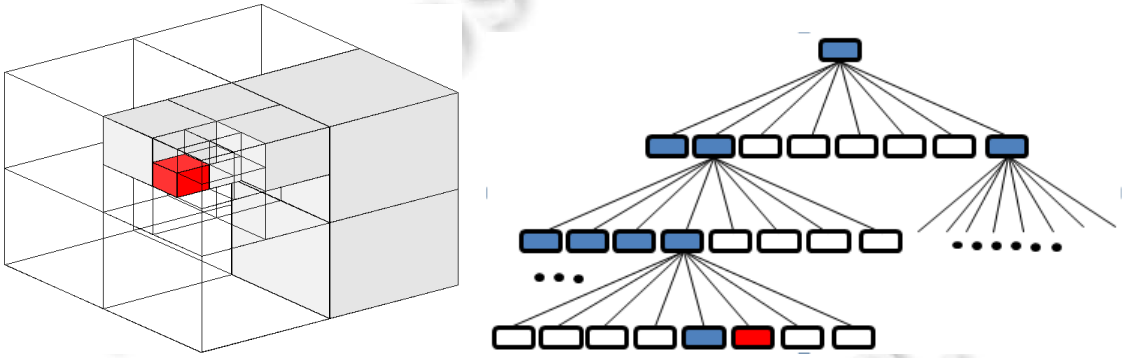
稀疏模型的构建原理参考文献[8],通过对视频帧提取 ORB 图像特征点并在视频帧序列的相邻帧之间或双目图像之间寻找特征匹配,接着使用图像特征点匹配和三角测量从图像中恢复像素的深度信息,使之成为体素.与此同时,使用捆集调整法不断优化所有体素的位姿,使其达到最优化,优化后的体素整体构建成环境稀疏地图.半稠密模型的构建原理参考文献[5],通过对图像剧烈变化的区域,也就是图像梯度处于极值的区域做图像配准

来更新相机位姿.更新后的相机位姿配合相关的若干视频帧,通过误差能量函数最小化估计像素深度,这是一种光流的思想.获取到图像深度信息后,半稠密模型将图像极值区域的像素群构建相应体素群,因为图像一般包含较多的极值区域,能够恢复较多的图像细节,所以这种体素群构建环境地图的方法称为半稠密建图.稠密模型的构建原理是通过视频帧的彩色图和配准的深度图在视频帧序列中不断更新当前相机姿态,并根据相机姿态,将新加入的具有直接深度的体素插入环境点云,从而直接获得稠密环境三维模型.

### 1.2 概率八叉树模型

八叉树是一种将环境空间迭代细分为八维子空间的存储表示方法.在环境感知中,八叉树的叶子节点存储所有的环境体素,非叶子节点称为内点,在插入、剪枝、归并或变更分辨率等树操作后可能转换为叶子节点或高层节点,如图 2 所示.环境空间内的立体度量单位在转换成三维模型中的一个体素后可以分为 3 种状态.其中一种称为未知状态,表示该体素在当前时刻之前相机尚未捕捉到的环境区域中.剩余两种相互制约状态分别称为占有和空闲,表示该体素对应的立体度量单位在当前时刻被认为有实体占有和被认为无实体占有.在实际对环境的感知过程中,受各种异构相机内部噪声和环境外部噪声的共同影响,当前时刻体素的状态难以由确定的三元状态完全解释,应使用更灵活的概率八叉树表示.本文所用概率八叉树的实现可参考文献[25,26],在第 $i$ 个叶子节点使用 log$Odds$ 值 $L(i)$,如式(1)所示,表示对占有状态的置信度.$P(i)$的初始值一般设置为 0.5,表示处于置信度不明确状态.根据三元状态的定义,该叶子节点空闲状态的置信度为 1–$L(i)$,而未知状态则会被预先标识.

$$L(i)=\text{log}it(p(i))=\log\left[\frac{P(i)}{1-P(i)}\right] \tag{1}$$



(a) 概率八叉树空间模型 (b) 概率八叉树结构

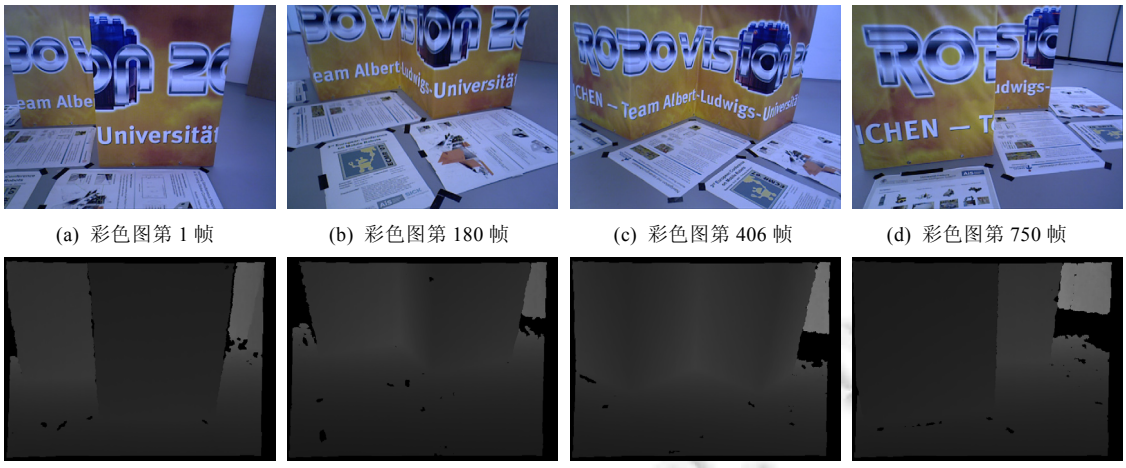
Fig.2 A typical example of probabilistic octree

图 2 概率八叉树示例

## 2 基于概率八叉树的多粒度环境模型的统一表示

传统环境模型多以单粒度模型为主,而单粒度模型自身存在各异构图像传感器的固有不足,且受模型固有噪声、随机噪声的影响较大.多粒度模型的主要困难则在于如何统一多粒度模型的表示,使其具有统一的语义背景.综上,本文首先提出一种多粒度环境点云模型及其概率八叉树的统一表示,具体如下所述.

本文所使用的多粒度的环境模型由稀疏、半稠密、稠密模型三者组成.为了较好地可视化算法每一步的结果,选择 TUM RGB-D 数据集[28]的 Fr3_structure_texture_far 序列,如图 3 所示,作为被感知的标准环境.图 4 展示了标准序列的稀疏、半稠密和稠密模型.从图中可以显著看出,如文献[8]等构建的稀疏模型可以较为真实地恢复场景的少量细节,如文献[5]等构建的半稠密模型虽然可以获得较多的环境细节,但噪声较大,在实际运动融合中极易出现地图信息不一致.只有如文献[11]等构建的稠密模型可以得到相机扫描区域较为平滑的三维模型.另一方面,稀疏和半稠密模型可以在无 GPU 加速的情况下实时运行,而稠密模型难以实时生成[4],在不同的硬件平台上可能会导致不同粒度模型的时序异步或地图不一致.

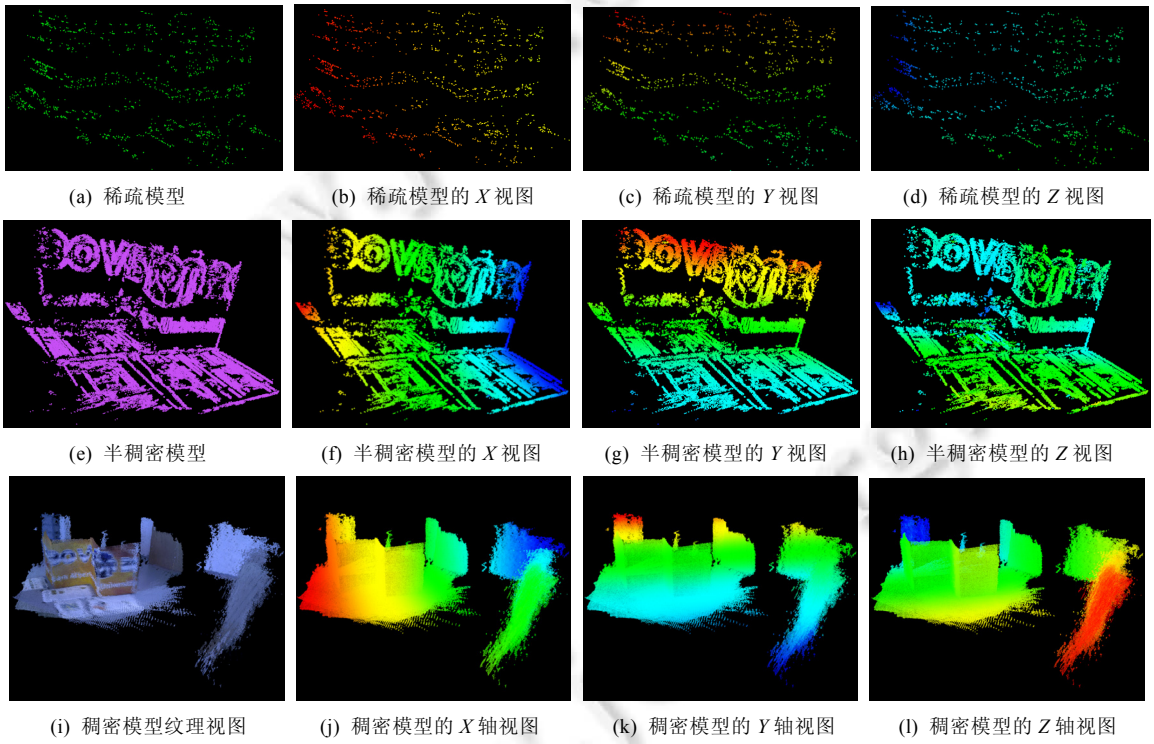


(a) 彩色图第 1 帧 (b) 彩色图第 180 帧 (c) 彩色图第 406 帧 (d) 彩色图第 750 帧

(e) 深度图第 1 帧 (f) 深度图第 180 帧 (g) 深度图第 406 帧 (h) 深度图第 750 帧

Fig.3 Fr3_structure_texture_far sequence of TUM RGB-D dataset (938 frames in total)

图 3 TUM RGB-D 数据集 Fr3_structure_texture_far 序列(共 938 帧)



(a) 稀疏模型 (b) 稀疏模型的 *X* 视图 (c) 稀疏模型的 *Y* 视图 (d) 稀疏模型的 *Z* 视图

(e) 半稠密模型 (f) 半稠密模型的 *X* 视图 (g) 半稠密模型的 *Y* 视图 (h) 半稠密模型的 *Z* 视图

(i) 稠密模型纹理视图 (j) 稠密模型的 *X* 轴视图 (k) 稠密模型的 *Y* 轴视图 (l) 稠密模型的 *Z* 轴视图

Fig.4 A comparison among sparse, semi dense and dense models.

Image color is set from dark blue to dark red according to its value along corresponding axis

图 4 3 种建模方法所建的多粒度点云模型比对,上层图片所示为稀疏模型,中层图片所示为半稠密模型,底层图片所示为稠密模型.其中 *XYZ* 视图根据体素相应的 *X*、*Y*、*Z* 值由小到大配以深蓝色到深红色的渐变色

本文使用概率八叉树模型统一表示 3 种模型.概率八叉树模型构建时,式(1)所表示的置信度关系随着相机的移动可能会产生 3 种状态的置信度变化,例如状态为未知的环境区域由摄像机在后续时刻捕捉到而转换为

其他两种状态或者非刚体环境中体素不定时在占有和未知状态相互转换.概率八叉树使用式(2)更新时间序列节点 $T$ 上的 log$Odds$ 值 $L(i|z_{1:t})$,其中,$z_{1:t}$ 为传感器在当前时刻的累计度量,$L(n|z_t)$根据异构模型的固有置信特性设置,如式(3)所示.因为这种 log$it$ 转换可逆,更新过程可以被解释为一种概率度量更新.其中,log$Odds$ 值通过反 log$it$ 变换,如式(4)所示,转换成 $P(n|z_{1:t})$,并由式(5)表示概率更新过程.在实际存储过程中还可以转换为占有、倾向占有、空闲、未知 4 种状态保存以压缩存储空间.传统八叉树存储表示方法不具有表示对多种体素状态的置信关系的能力,也不具有伴随时间序列更新的特性.相较于文献[25,26],本文方法充分考虑到模型自身固有噪声的影响,同时也考虑到模型的存储空间因素,因而更为精确和灵活.

$$L(n|z_{1:t}) = L(n|z_{1:t-1}) + L(n|z_t) \tag{2}$$

$$L(n|z_t)=\begin{cases} l^1_{occ}=0.7, & \text{特征点节点} \\ l^1_{free}=-0.3, & \text{非特征点节点} \\ l^2_{occ}=0.6, & \text{灰度极值节点} \\ l^2_{free}=-0.3, & \text{非灰度极值节点} \\ l^3_{occ}=0.85, & \text{红外已捕捉节点} \\ l^3_{free}=-0.4, & \text{红外未捕捉节点} \end{cases} \tag{3}$$

$$P(i) = \text{log}it^{-1}(L(i)) = \frac{1}{1+\exp(-L(i))} \tag{4}$$

$$P(i|z_{1:t}) = \left[1+\frac{1-P(i|z_t)}{P(i|z_t)}\frac{1-P(i|z_{1:t-1})}{P(i|z_{1:t-1})}\frac{P(i)}{1-P(i)}\right]^{-1} \tag{5}$$

概率八叉树中非叶子节点也称为内点,这种节点在优化剪枝、变更分辨率导致的节点合并等操作后可能转换为叶子节点或高层节点.因此,如何将八维孩子节点状态均为占有、空闲或未知的内点剪枝,以内点表示内点为根的子树整体,优化查询存储,即称为剪枝问题.如何将若干叶子节点合并为高层节点并把高层节点作为叶子节点参与表示或存储称为合并问题.

为了解决概率八叉树的剪枝问题,首先需要确定叶子节点的置信度上下限,也就是说,对于内点所有子节点的某种状态的置信度均达到了确信状态(overconfidence)时,才可以被剪枝.与此同时,剪枝后的内点代表整个子树成为叶子节点,如何为内点 $n$ 分配一个合理的 log$Odds$ 值使 $L(n)$代表整个子树成为另一个需要考虑的问题.为了解决第 1 个问题,本文使用式(6)在更新过程中配合置信上下限边界 $l_{\max}$ 和 $l_{\min}$ 以区分占有状态的确信和空闲状态的确信.当全部子节点处于同一状态下被确信后就可以触发剪枝操作.

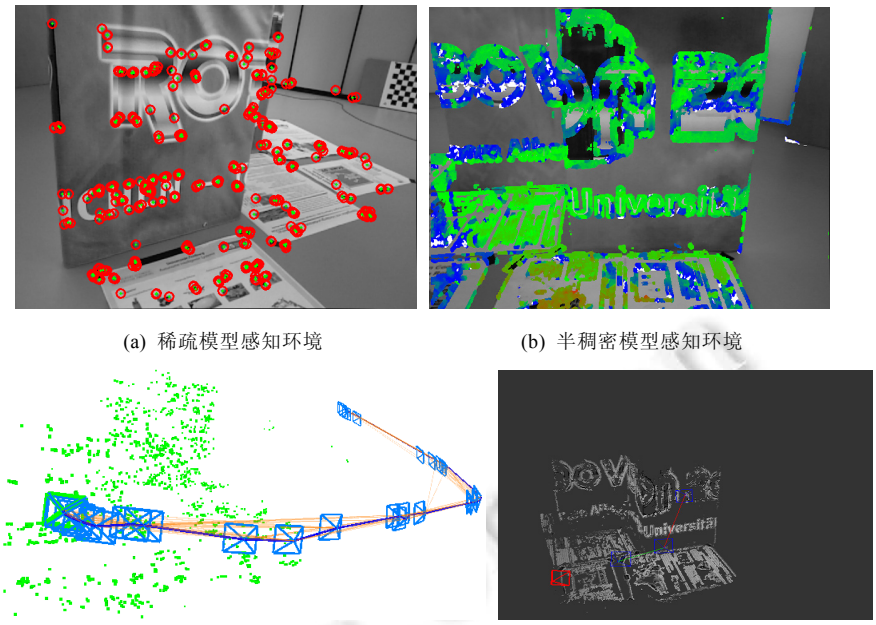
$$L(n|z_{1:t}) = \max(\min(L(n|z_{1:t-1}) + L(n|z_t), l_{\max}), l_{\min}) \tag{6}$$

多粒度也就是多分辨率模型,具体实现依靠概率八叉树子节点迭代归并动态生成.多粒度模型实现和上述剪枝问题一样,存在内点在剪枝后 log$Odds$ 值如何设置的问题.这种问题存在如:子节点最大值、子节点最小值等诸多解决方法.本文选择式(7)所示子节点均值方法,旨在避免在建图过程中由于内部或外部噪声所带来的最大、最小值噪声影响.

$$\bar{l}(n) = \frac{1}{8}\sum_{i=1}^{8} L(n_i) \tag{7}$$

## 3 概率八叉树模型的时间序列融合

稀疏、半稠密和稠密模型经过上述过程后转化为统一的概率八叉树表示.在相机时间序列中,若干稀疏、半稠密和稠密的概率八叉树模型在同步定位与建图的每个时刻不断生成,如图 5 所示,相机在沿轨迹运动的过程中,算法不断以多粒度感知环境.因此,在时间序列中的每个时间节点,由不同异构模型所生成的若干概率八叉树模型如何有机融合成统一的时态融合概率八叉树模型 TFPOM 成为下一步需要解决的问题.



(a) 稀疏模型感知环境 (b) 半稠密模型感知环境

(c) 稀疏模型的相机姿态轨迹 (d) 半稠密模型的相机姿态轨迹

Fig.5 Sparse and semi dense multi time spatial probabilistic octree models

图 5 稀疏和半稠密的多时态概率八叉树模型

本文提出一种基于卡尔曼滤波的置信融合方法,设第 $i$ 个概率八叉树模型的第 $j$ 个叶子节点在 $k$ 时刻的形式化表示为 $L_j^i(k)$, 其中,$i$=0 表示时态融合概率.置信度融合算法首先引入一个离散系统的控制模型和测量方程,分别如式(8)、式(9)所示.其中, $L_j^0(k)$ 为系统在 $k$ 时刻的状态变量, $z_j^i(k)$ 为系统在 $k$ 时刻第 $i$ 个观测样本的状态变量观测值,$w(k)$,$v^i(k)$分别为过程激励噪声和第 $i$ 个观测样本的观测噪声,协方差分别为 $Q$,$R^i$.

$$L_j^0(k+1)=L_j^0(k)+w(k) \tag{8}$$

$$z_j^i(k)=L_j^i(k)+v^i(k) \tag{9}$$

下一步使用系统的过程模型,如式(10)所示,预测下一时刻的系统状态.

$$L_j^0(k+1|k)=L_j^0(k|k) \tag{10}$$

更新状态 $L_j^0(k+1|k)$ 的协方差 $P_j^0(k+1|k)$, 如式(11)所示.

$$P_j^0(k+1|k)=P_j^0(k|k)+Q \tag{11}$$

计算每个观测值的卡尔曼增益 $K_g^i$ ,如式(12)所示.

$$K_g^i=P_j^0(k+1|k)(P_j^0(k+1|k)+R^i)^{-1} \tag{12}$$

结合观测值更新状态估计 $L_j^0(k+1|k+1)$, 如式(13)所示,其中,$N$ 为观测值个数,$obv$ 函数将模型置信度融入,如式(14)所示.

$$L_j^0(k+1|k+1)=L_j^0(k+1|k)+\frac{1}{N}\sum_{i=1}^{N}obv(K_g^i(z^i(k+1)-L_j^0(k+1|k)) \tag{13}$$

$$obv(x(k_g^i))=\begin{cases}1.2x, & \text{模型 } i \text{ 属于稠密模型}\\ 0.8x, & \text{模型 } i \text{ 属于半稠密模型}\\ x, & \text{模型 } i \text{ 属于稀疏模型}\end{cases} \tag{14}$$

最后更新 $k+1$ 时刻的误差估计 $P_j^0(k+1|k+1)$, 如式(15)所示,并返回式(10),使卡尔曼滤波器随时间序列不断迭代下去.

$$P_j^0(k+1|k+1)=\left(I-\frac{1}{N}\sum_{i=1}^{N}K_g^i\right)P_j^0(k+1|k) \tag{15}$$

## 4 算法流程与复杂度分析

本文算法的整体步骤可概括如下.

Step 1. 更新当前时刻 $k$;

Step 2. 更新时刻 $k$ 的稀疏、半稠密、稠密模型的统一八叉树表示 $oct^1(k)$、$oct^2(k)$、$oct^3(k)$作为观测模型;

Step 3. 更新 TFPOM 当前节点索引 $j$;

Step 4. 根据 Step 2 的结果更新 TFPOM 当前叶子节点各观测模型的观测值 $z_j^i(k)$;

Step 5. 引入 TFPOM 节点系统状态的控制模型和测量方程,如式(8)、式(9)所示;

Step 6. 使用式(10)预测下一时刻系统状态;

Step 7. 以式(11)预测下一时刻系统状态的误差 $P_j^0(k+1|k)$;

Step 8. 通过式(12)计算每个观测值的卡尔曼增益 $K_g^i$;

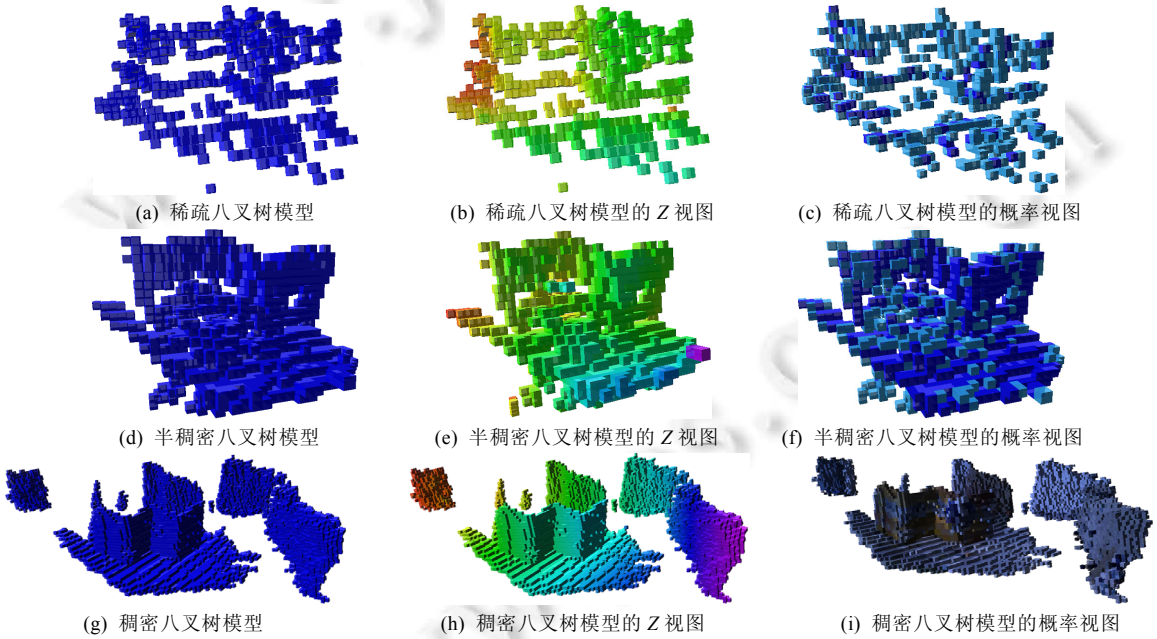
Step 9. 通过式(14)融入模型置信度,并通过式(13)更新下一时刻系统状态估计 $L_j^0(k+1|k+1)$;

Step 10. 使用式(15)更新系统状态的误差估计 $P_j^0(k+1|k+1)$;

Step 11. 如果没有可以插入或者更新的叶子节点,则进入 Step 12,否则返回 Step 3;

Step 12. 返回 Step 1.

为可视化算法的完整过程,对共 938 帧的 TUM RGB-D 数据集 Fr3_structure_texture_far 序列生成的点云,如图 3 所示,对应生成统一的概率八叉树表示,如图 6 所示.接着对上一步每个时间节点所生成的若干概率八叉树融合.最后通过图像时间序列地不断增量融合,形成完整的多粒度环境模型,如图 7 所示.



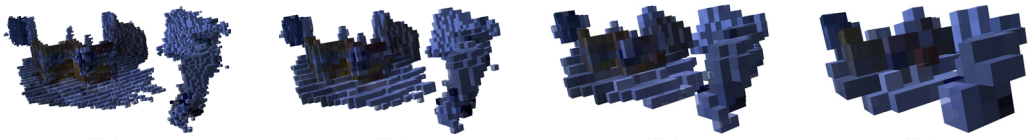
(a) 稀疏八叉树模型 (b) 稀疏八叉树模型的 $Z$ 视图 (c) 稀疏八叉树模型的概率视图

(d) 半稠密八叉树模型 (e) 半稠密八叉树模型的 $Z$ 视图 (f) 半稠密八叉树模型的概率视图

(g) 稠密八叉树模型 (h) 稠密八叉树模型的 $Z$ 视图 (i) 稠密八叉树模型的概率视图

Fig.6 Uniform probabilistic octree representation corresponding to its point cloud model. The dark cube represent over-confident occupied node, the light cube represent probabilistic occupied node and the white cube represent over-confident free node in probabilistic view

图 6 多粒度点云模型相应的统一概率八叉树表示.其中概率视图中深色方块代表确信占有节点,浅色为非确信占有节点,空白区域为确信空闲节点



(a) 分辨率 0.05m (b) 分辨率 0.1m (c) 分辨率 0.2m (d) 分辨率 0.4m

Fig.7 Multi-Granularity form of temporal fused environment probabilistic octree model

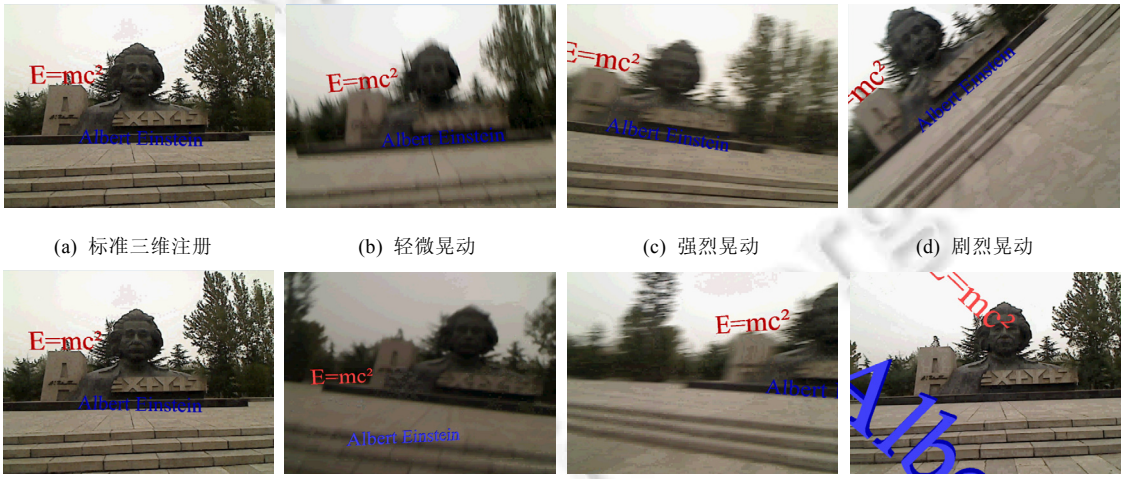
图 7 时态融合概率八叉树环境模型的多粒度形态

本文算法在实际时态融合的过程中,仅增量更新在当前时刻标记为变动状态的内节点与叶子节点,对于未处于变动状态的节点,只更新此节点的卡尔曼滤波器相关变量,不进行概率八叉树的搜索和更新.算法整体复杂度为 $O(N_2\log_8 N_1)$,其中,$N_1$ 代表环境模型的八叉树节点数,$N_2$ 代表标记为变动状态的节点的个数.

## 5 实 验

### 5.1 增强现实

增强现实(augmented reality,简称 AR)是一种将虚拟信息三维注册在真实场景中,追求沉浸式融合虚拟信息和真实场景的计算机视觉技术,是快速三维建模的一个重要应用.如今这种技术已被广泛应用于工业、娱乐、游戏、医疗、军事等领域.增强现实具有沉浸感的前提是具有较好的三维注册和虚拟遮挡能力,也就是对环境具有较高精度的感知能力.所以本文首先使用室外增强现实系统,以融合单目稀疏环境模型和双目稠密环境模型后的 TFPOM 作为系统的虚拟环境模型,测试算法的环境感知能力.实验在配有 ZED 双目相机和 Logitech Pro9000 单目相机(均选择常用分辨率 640×480)的 Nvidia Jetson TX1 Developer Kit 嵌入式硬件平台上进行测试,该硬件处理平台核心硬件包含 ARM 架构的 A57 型号 4 核处理器,Maxwell 架构含有 256 个 cuda 单元的 Nvidia 显卡和 4GB LPDDR4 内存.实验结果如图 8 所示.




(a) 标准三维注册 (b) 轻微晃动 (c) 强烈晃动 (d) 剧烈晃动

(e) 标准三维注册 (f) 轻微晃动 (g) 强烈晃动 (h) 剧烈晃动

Fig.8 A comparison between temporal fused probabilistic octree model based AR (upper) and point cloud model based AR (lower)

图 8 基于时态融合概率八叉树模型的增强现实(上层图片)与基于点云模型的增强现实(下层图片)在相机动态模糊情况下的效果对比

记本文算法生成的时态融合概率八叉树作为环境模型的增强现实系统称为系统 A,则系统 A 的三维注册效果如图 8(a)~图 8(d)所示.记图像特征点云作为环境模型的增强现实系统称为系统 B,则系统 B 的三维注册效

果如图 8(e)~图 8(h)所示,系统 A、B 均能够实时运行(30 帧每秒以上).从对比图可以较为明显地看出,系统 A 在经历轻微至剧烈晃动的过程中,虚拟物体(红色的能量方程、蓝色的爱因斯坦英文全名)仍然精确三维注册在标准三维注册时的三维位姿.而轻微晃动导致的运动模糊就已经使系统 B 中虚拟物体相对于原先注册的三维位姿大幅漂移,即使相机恢复原始相机位姿,虚拟物体仍然不能恢复原始虚拟物体所注册时的三维位姿.导致这种现象出现的原因主要是因为系统 B 在相机剧烈运动过程中所出现的不同强度的运动模糊影响下导致定位漂移,进一步影响了特征点所恢复出深度的准确性.这种现象甚至导致了环境地图的整体不一致性,如图 8(h)所示,即使相机恢复了原始位姿,虚拟物体也无法恢复原始三维注册的位姿.而系统 A 则因为滤波融合了双目及单目数据源,显著减少了单源的运动模糊随机噪声和硬件固有噪声,使得系统 A 能够在持续运动模糊中始终保持虚拟物体原始三维注册时的三维姿态.

### 5.2 视觉里程计

TUM RGB-D 数据集[28]包含若干室内视频序列相应的彩色图和配准后的深度图,详细信息见表 1,这些视频序列由复杂多样的相机轨迹对不同特点的环境场景拍摄而成,为评价视觉定位与建图算法的定位精度做出了重要贡献.相对于 TUM 数据集,KITTI 数据集[29]包含若干室外视频序列的双目彩色图、GPS 数据、IMU 数据、激光数据等,详细信息见表 2,这些视频序列由车载摄像拍摄而成,是评价自动导航系统,如自动驾驶汽车、多旋翼无人机等优劣的重要数据平台.这两种数据集的评价都是检测视觉里程计(visual odometry,简称 VO)优劣的主流方法.同步定位与建图理论中定位与建图相互依附、相互影响,相机定位精度直接决定建图体素的三维位姿误差,而体素的位姿误差则会反向影响三角测量时相机的三维位姿定位,是一种典型的 VO 思想,所以定位精度的评价可以反映建图质量的优劣.其中,TUM 数据集的模型构造效果本文已有介绍,KITTI 数据集的稀疏地图构建过程如图 9 所示.

**Table 1** Detail information of TUM RGB-D dataset

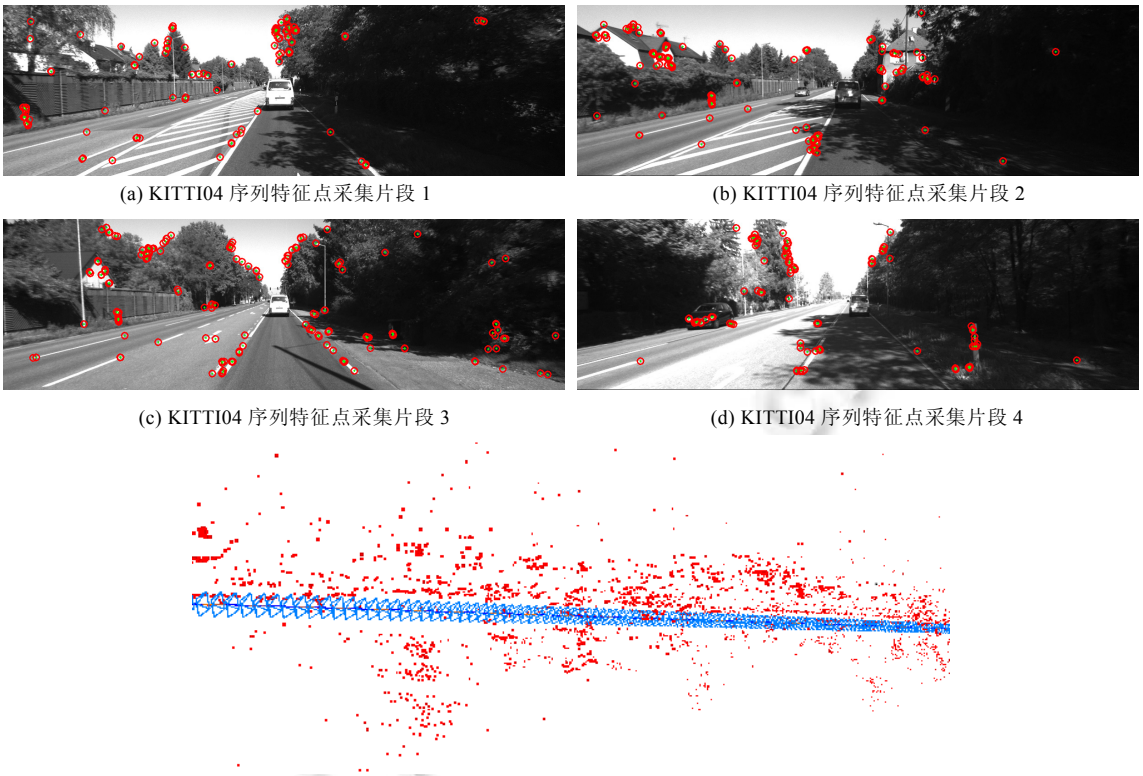
**表 1** TUM RGB-D 数据集详细信息

| 视频序列 | 序列编号 | 数据集规模(image) | 感知时间长度(s) |
|---|---|---|---|
| Fr1_desk | 1 | 613 | 20.43 |
| Fr1_floor | 2 | 1 242 | 41.4 |
| Fr1_xyz | 3 | 798 | 26.6 |
| Fr2_360_kidnap | 4 | 1 431 | 47.7 |
| Fr2_desk | 5 | 2 965 | 98.83 |
| Fr2_desk_with_person | 6 | 4 067 | 135.56 |
| Fr2_xyz | 7 | 3 669 | 122.3 |
| Fr3_long_office_household | 8 | 2 585 | 86.16 |
| Fr3_nostructure_texture_far | 9 | 465 | 15.5 |
| Fr3_nostructure_texture_near_withloop | 10 | 1 682 | 56.06 |
| Fr3_sitting_halfsphere | 11 | 1 110 | 37 |
| Fr3_sitting_xyz | 12 | 1 261 | 42.03 |
| Fr3_structure_texture_far | 13 | 938 | 31.26 |
| Fr3_structure_texture_near | 14 | 1 099 | 36.63 |
| Fr3_walking_halfsphere | 15 | 1 067 | 35.56 |
| Fr3_walking_xyz | 16 | 859 | 28.63 |

**Table 2** Detail information of KITTI outdoor dataset

**表 2** KITTI 室外数据集详细信息

| 视频序列 | 序列编号 | 环境大小(m×m) | 数据集规模(image) | 感知时间长度(s) |
|---|---|---|---|---|
| KITTI 00 | 1 | 564×496 | 4 541 | 908.2 |
| KITTI 01 | 2 | 1157×1827 | 1 101 | 220.2 |
| KITTI 02 | 3 | 599×946 | 4 661 | 932.2 |
| KITTI 03 | 4 | 471×199 | 801 | 160.2 |
| KITTI 04 | 5 | 0.5×394 | 271 | 54.2 |
| KITTI 05 | 6 | 479×426 | 2 761 | 552.2 |
| KITTI 06 | 7 | 23×457 | 1 101 | 220.2 |
| KITTI 07 | 8 | 191×209 | 1 101 | 220.2 |
| KITTI 08 | 9 | 808×391 | 4 071 | 814.2 |
| KITTI 09 | 10 | 465×568 | 1 591 | 318.2 |
| KITTI 10 | 11 | 671×177 | 1 201 | 240.2 |



(a) KITTI04 序列特征点采集片段 1

(b) KITTI04 序列特征点采集片段 2

(c) KITTI04 序列特征点采集片段 3

(d) KITTI04 序列特征点采集片段 4

(e) KITTI 04 视频序列建图轨迹

Fig.9 Feature extraction and mapping trajectory of KITTI 04

图 9 KITTI 数据集 04 序列的特征点采集和建图轨迹示例

实验基于上文的分析,使用本文算法所建模型与其他最先进的定位与建图算法[2,5,8,30]所建模型的定位轨迹数据和真实轨迹的绝对均方根误差,简称 RMSE[28],评价方法度量本文算法的精确性和时间效率.由于 KITTI 数据集处于室外环境,缺少深度图信息,故只与文献[8]方法进行对比.所使用的硬件平台核心包含 Intel Core i7-4700MQ 处理器,Nvidia GT755M 显卡和 8GB 内存.TUM 数据集的 RMSE 对比结果如图 10 所示,为便于显示,所有高于 20cm 的误差将被显示为 20cm,rgbdslam 数据源自文献[30],仅包含前 7 个序列数据.KITTI 数据集的 RMSE 对比结果如图 11 所示,部分相机轨迹误差如图 12 所示.相关时间消耗分列于表 1、表 2 的最后一列,TUM 数据集官方推荐帧率为 30 帧每秒,KITTI 数据集官方推荐频率为 10 帧每秒.

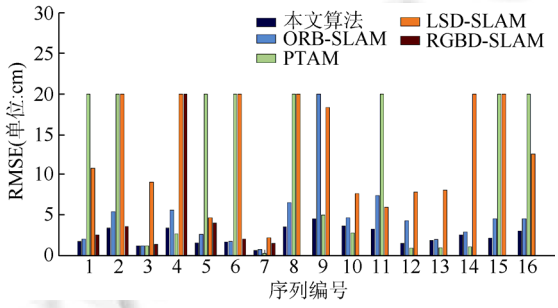



Fig.10 A RMSE comparison among some sequences of TUM dataset

图 10 TUM 数据集若干视频序列的 RMSE 误差对比

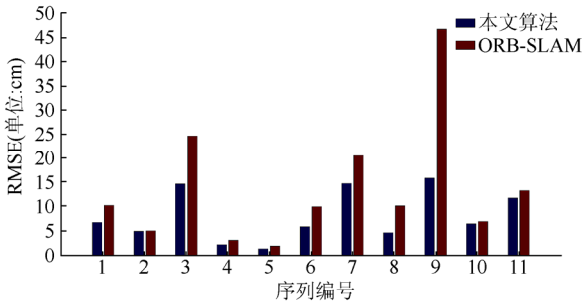
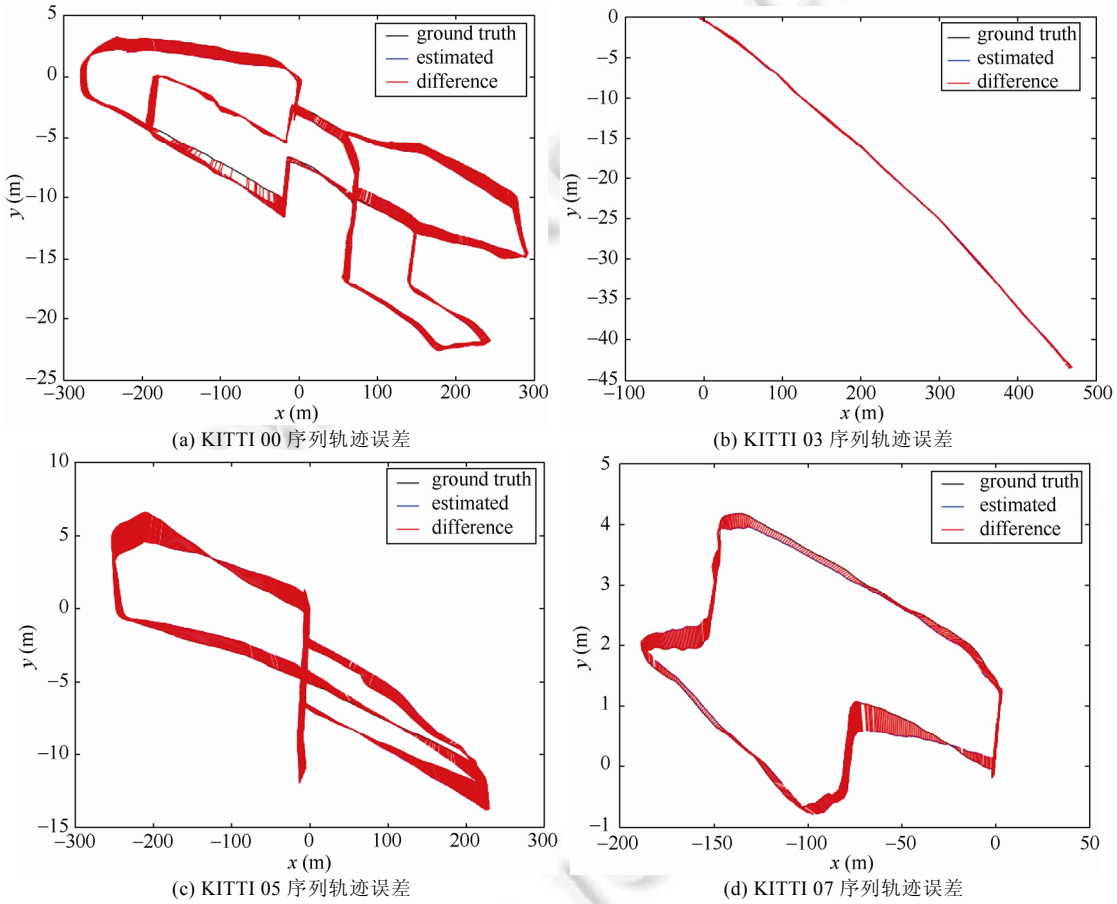



Fig.11 A RMSE comparison among sequences of KITTI dataset

图 11 KITTI 数据集视频序列的 RMSE 对比




(a) KITTI 00 序列轨迹误差 (b) KITTI 03 序列轨迹误差

(c) KITTI 05 序列轨迹误差 (d) KITTI 07 序列轨迹误差

Fig.12 Trajectory error between ground truth and estimated of four typical sequences of KITTI dataset

图 12 KITTI 数据集 4 个典型序列真实轨迹与预测轨迹的轨迹误差

在如图 10 所示的 TUM 数据集测试中,本文所对比的算法均为国内外 visual SLAM 领域不同建图类型的当今前沿工作.由于 TUM 数据集包含诸如纹理单一的 Fr1_floor 序列、内含行人的 Fr2_desk_with_person 序列和缺少结构、纹理并包含闭环的 Fr3_nostructure_texture_near_withloop 序列等复杂场景和跳帧问题,这些都加大了对算法精确度和鲁棒性的要求.通过对比可知,本文算法不仅在大部分序列中有效地使 RMSE 误差有所下降,并且没有在任一数据集上产生被认为是跟踪失败的大于 20cm 的 RMSE 误差.在图 11 所示的 KITTI 数据集测

试中,这种效果得以保持并验证了本文算法对室外环境的有效性.另外,在这里的KITTI数据集对比中,为了与其他单目算法保持一致,本文算法仍使用相对尺度以及Huber估计拟合ground truth,并计算RMSE.

为了进一步分析本文算法的时间效率,我们进一步对 visual SLAM 算法中较为关心和运算频率较高的任务:TFPOM 模型的构建和遍历与点云模型遍历进行测试,相关时间数据见表 3.本文选取了体素个数分别为 9 336、67 230 和 450 253 的由数据集生成的典型稀疏、半稠密和稠密环境模型,如图 3(a)、图 3(e)和图 3(i)所示.所用点云模型采用 0.01m$^3$ 的分辨率,对比的 TFPOM 采用常用的 0.05m$^3$ 和 0.1m$^3$ 的分辨率进行测试.通过表 3 可以看出,点云模型遍历从稀疏到稠密分别需要 68ms~1 916ms,而所对应的 TFPOM 模型在处理最复杂的稠密模型时也仅需要 39ms,在时间消耗上普遍下降了 1~2 个数量级，并可以通过提高分辨率而进一步下降.在构建 TFPOM 任务中最复杂的稠密模型也仅需要 58ms.另一个需要关注的现象是,即使半稠密模型的体素较之稀疏模型大量上升,但 TFPOM 的构建时间却反而下降.这主要是因为相关点大量聚集在小簇中使得空间八叉树的时间效率优势得以进一步体现所致.可以预见的是,相关visual SLAM算法在使用TFPOM模型后将会具有更多的空闲时间用于捆集调整,以进一步提升算法精确性.

**Table 3** Time statistics with TFPOM

**表 3** TFPOM 模型相关时间统计

| 任务 | 环境模型类型 | 体素分辨率(m$^3$) | 体素个数(per) | 时间消耗(ms) |
|---|---|---|---|---|
| 遍历点云 | 稀疏 | 0.01 | 9 336 | 68.24±10.683 |
| 遍历点云 | 半稠密 | 0.01 | 67 230 | 332.08±38.317 |
| 遍历点云 | 稠密 | 0.01 | 450 253 | 1 916±38.996 |
| 构建 TFPOM | 稀疏 | 0.05 | 9 336 | 6.218±1.284 |
| 构建 TFPOM | 稀疏 | 0.1 | 9 336 | 4.613±1.521 |
| 构建 TFPOM | 半稠密 | 0.05 | 67 230 | 5.919±0.015 |
| 构建 TFPOM | 半稠密 | 0.1 | 67 230 | 4.193±0.017 |
| 构建 TFPOM | 稠密 | 0.05 | 450 253 | 58.554±0.369 |
| 构建 TFPOM | 稠密 | 0.1 | 450 253 | 31.567±0.15 |
| 遍历 TFPOM | 稀疏 | 0.05 | 9 336 | 13.45±1.514 |
| 遍历 TFPOM | 稀疏 | 0.1 | 9 336 | 7.775±0.583 |
| 遍历 TFPOM | 半稠密 | 0.05 | 67 230 | 3.81±0.063 |
| 遍历 TFPOM | 半稠密 | 0.1 | 67 230 | 1.332±0.583 |
| 遍历 TFPOM | 稠密 | 0.05 | 450 253 | 39.521±0.554 |
| 遍历 TFPOM | 稠密 | 0.1 | 450 253 | 8.68±0.218 |

## 6 结 论

本文提出了基于光学图像的多粒度随动环境感知算法.算法针对稀疏、半稠密、稠密多种粒度的异构三维点云模型,同时提出一种概率八叉树表示,将所生成的若干三维模型统一表示.再通过剪枝、归并及卡尔曼滤波,在相机运动期间不断融合多种异构点云的置信度,生成唯一的时态融合概率八叉树环境模型TFPOM.实验结果表明,本文算法在压缩环境模型存储空间的同时,使其可以任意粒度动态拟合真实环境,实现鲁棒的随动环境感知.基于该环境模型的视觉导航具有较小的轨迹误差和较高的时间效率.

**References**:

[1] Davison AJ, Reid ID, Molton ND, Stasse O. MonoSLAM: Real-Time single camera SLAM. IEEE Trans. on Pattern Analysis & Machine Intelligence, 2007,29(6):1052–1067. [doi: 10.1109/TPAMI.2007.1049]

[2] Klein G, Murray D. Parallel tracking and mapping for small AR workspaces. IEEE and ACM Int'l Symp. on Mixed and Augmented Reality. 2007. 225–234. [doi: 10.1109/ISMAR.2007.4538852]

[3] Castle RO, Klein G, Murray DW. Wide-Area augmented reality using camera tracking and mapping in multiple regions. Computer Vision and Image Understanding, 2011,115(6):854–867. [doi: 10.1016/j.cviu.2011.02.007]

[4] Newcombe RA, Lovegrove SJ, Davison AJ. DTAM: Dense tracking and mapping in real-time. In: Proc. of the Int'l Conf. on Computer Vision. IEEE Computer Society, 2010. 2320–2327. [doi: 10.1109/ICCV.2011.6126513]

[5] Engel J, Schöps T, Cremers D. LSD-SLAM: Large-Scale direct monocular SLAM. In: Proc. of the Computer Vision (ECCV 2014). Springer Int'l Publishing, 2014. 834–849. [doi: 10.1007/978-3-319-10605-2_54]

[6] Caruso D, Engel J, Cremers D. Large-Scale direct SLAM for omnidirectional cameras. In: Proc. of the 2015 IEEE/RSJ Int'l Conf. on Intelligent Robots and Systems (IROS). IEEE, 2015. 141–148. [doi: 10.1109/IROS.2015.7353366]

[7] Engel J, Stuckler J, Cremers D. Large-Scale direct slam with stereo cameras. In: Proc. of the 2015 IEEE/RSJ Int'l Conf. on Intelligent Robots and Systems (IROS). IEEE, 2015. 1935–1942. [doi: 10.1109/IROS.2015.7353631]

[8] Mur-Artal R, Montiel JMM, Tardos JD. ORB-SLAM: A versatile and accurate monocular SLAM system. IEEE Trans. on Robotics, 2015,1147–1163. [doi: 10.1109/TRO.2015.2463671]

[9] Rublee E, Rabaud V, Konolige K, Bradski G. ORB: An efficient alternative to SIFT or SURF. In: Proc. of the 2011 IEEE Int'l Conf. on Computer Vision (ICCV). IEEE, 2011. 2564–2571. [doi: 10.1109/ICCV.2011.6126544]

[10] Mur-Artal R, Tardos J. Probabilistic semi-dense mapping from highly accurate feature-based monocular SLAM. In: Proc. of the Robotics: Science and Systems. Rome, 2015. [doi: 10.15607/RSS.2015.XI.041]

[11] Newcombe RA, Izadi S, Hilliges O, Molyneaux D, Kim D, Davison AJ, Fitzgibbon A. KinectFusion: Real-Time dense surface mapping and tracking. In: Proc. of the 10th IEEE Int'l Symp. on Mixed and Augmented Reality (ISMAR). IEEE, 2011. 127–136. [doi: 10.1109/ISMAR.2011.6092378]

[12] Salas-Moreno R, Newcombe R, Strasdat H, Kelly P, Davison A. Slam++: Simultaneous localisation and mapping at the level of objects. In: Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition. 2013. 1352–1359. [doi: 10.1109/CVPR.2013.178]

[13] Newcombe RA, Fox D, Seitz SM. DynamicFusion: Reconstruction and tracking of non-rigid scenes in real-time. In: Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition. 2015. 343–352. [doi: 10.1109/CVPR.2015.7298631]

[14] Kahler O, Prisacariu VA, Ren CY, Sun X, Torr P, Murray D. Very high frame rate volumetric integration of depth images on mobile devices. IEEE Trans. on Visualization and Computer Graphics, 2015,21(11):1241–1250. [doi: 10.1109/TVCG.2015.2459891]

[15] Moravec H. Robot spatial perceptionby stereoscopic vision and 3D evidence grids. Perception, 1996.

[16] Roth-Tabak Y, Jain R. Building an environment model using depth information. Computer, 1989,22(6):85–90. [doi: 10.1109/2.30724]

[17] Cole DM, Newman PM. Using laser range data for 3D SLAM in outdoor environments. In: Proc. of the 2006 IEEE Int'l Conf. on Robotics and Automation. IEEE, 2006. 1556–1563. [doi: 10.1109/ROBOT.2006.1641929]

[18] Surmann H, Nüchter A, Lingemann K., Hertzberg J. 6D SLAM—Mapping outdoor environment. Journal of Field Robotics, 2007,24:699–722. [doi: 10.1002/rob.20209]

[19] Meagher D. Geometric modeling using octree encoding. Computer Graphics and Image Processing, 1982,19(2):129–147. [doi: 10.1016/0146-664X(82)90104-6]

[20] Wilhelms J, Van Gelder A. Octrees for faster isosurface generation. ACM Trans. on Graphics (TOG), 1992,11(3):201–227. [doi: 10.1145/130881.130882]

[21] Dai Z, Cha JZ, Ni ZL. A fast decomposition algorithm of octree node in 3D-packing. Ruan Jian Xue Bao/Journal of Software, 1995,6(11):679–685 (in Chinese with English abstract). http://www.jos.org.cn/1000-9825/19951106.htm

[22] Payeur P, Hébert P, Laurendeau D, Gosselin, CM. Probabilistic octree modeling of a 3D dynamic environment. In: Proc. of the IEEE Int'l Conf. on Robotics and Automation. IEEE, 1997,2:1289–1296. [doi: 10.1109/ROBOT.1997.614315]

[23] Fournier J, Ricard B, Laurendeau D. Mapping and exploration of complex environments using persistent 3D model. In: Proc. of the 4th Canadian Conf. on Computer and Robot Vision (CRV 2007). IEEE, 2007. 403–410. [doi: 10.1109/CRV.2007.45]

[24] Pathak K, Birk A, Poppinga J, Schwertfeger S. 3D forward sensor modeling and application to occupancy grid based sensor fusion. In: Proc. of the 2007 IEEE/RSJ Int'l Conf. on Intelligent Robots and System (IROS 2007). IEEE, 2007. 2059–2064. [doi: 10.1109/IROS.2007.4399406]

[25] Wurm KM, Hornung A, Bennewitz M, Stachniss C, Burgard W. OctoMap: A probabilistic, flexible, and compact 3D map representation for robotic systems. In: Proc. of the ICRA 2010 Workshop on Best Practice in 3D Perception and Modeling for Mobile Manipulation. 2010, 2.

[26] Hornung A, Wurm KM, Bennewitz M, Stachniss C, Burgard W. OctoMap: An efficient probabilistic 3D mapping framework based on octrees. Autonomous Robots, 2013,34(3):189–206. [doi: 10.1007/s10514-012-9321-0]

[27] Benson D, Davis J. Octree textures. ACM Trans. on Graphics (TOG), 2002,21(3):785–790. [doi: 10.1145/566654.566652]

[28] Sturm J, Engelhard N, Endres F, Burgard W, Cremers D. A benchmark for the evaluation of RGB-D SLAM systems. In: Proc. of the 2012 IEEE/RSJ Int'l Conf. on Intelligent Robots and Systems (IROS). IEEE, 2012. 573–580. [doi: 10.1109/IROS.2012.6385773]

[29] Geiger A, Lenz P, Stiller C, Urtasun R. Vision meets robotics: The KITTI dataset. The Int'l Journal of Robotics Research, 2013, 0278364913491297. [doi: 10.1177/0278364913491297]

[30] Endres F, Hess J, Sturm J, Cremers D, Burgard W. 3-d mapping with an RGB-d camera. IEEE Trans. on Robotics, 2014,30(1):177–187. [doi: 10.1109/TRO.2013.2279412]

**附中文参考文献**:

[21] 戴佐,查建中,倪仲力.三维布局中八叉树节点的快速分解算法.软件学报,1995,6(11):679–685. http://www.jos.org.cn/1000-9825/19951106.htm

**陈昊升**(1992－),男,河南洛阳人,硕士生,主要研究领域为计算机视觉.

**叶阳东**(1962－),男,博士,教授,博士生导师,CCF 高级会员,主要研究领域为智能系统,机器学习,数据库.

**张格**(1984－),男,博士生,实验师,主要研究领域为增强现实,人工智能.