QCon 全球软件开发大会
INTERNATIONAL SOFTWARE DEVELOPMENT CONFERENCE

BEIJING 2017

复杂环境下的视觉同时定位与地图构建

章国锋

浙江大学CAD&CG国家重点实验室

主办方 Geekbang. 极客邦科技 InfoQ

SLAM: 同时定位与地图构建

- 机器人和计算机视觉领域的基本问题
  - 在未知环境中定位自身方位并同时构建环境三维地图
- 广泛的应用
  - 增强现实、虚拟现实
  - 机器人、无人驾驶

SLAM常用的传感器

- 红外传感器：较近距离感应，常用于扫地机器人。
- 激光雷达：单线、多线等。
- 摄像头：单目、双目、多目等。
- 惯性传感器（英文叫IMU，包括陀螺仪、加速度计等）：智能手机标配。

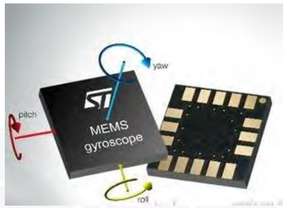
激光雷达

常见的单目摄像头

普通手机摄像头也可作为传感器

双目摄像头

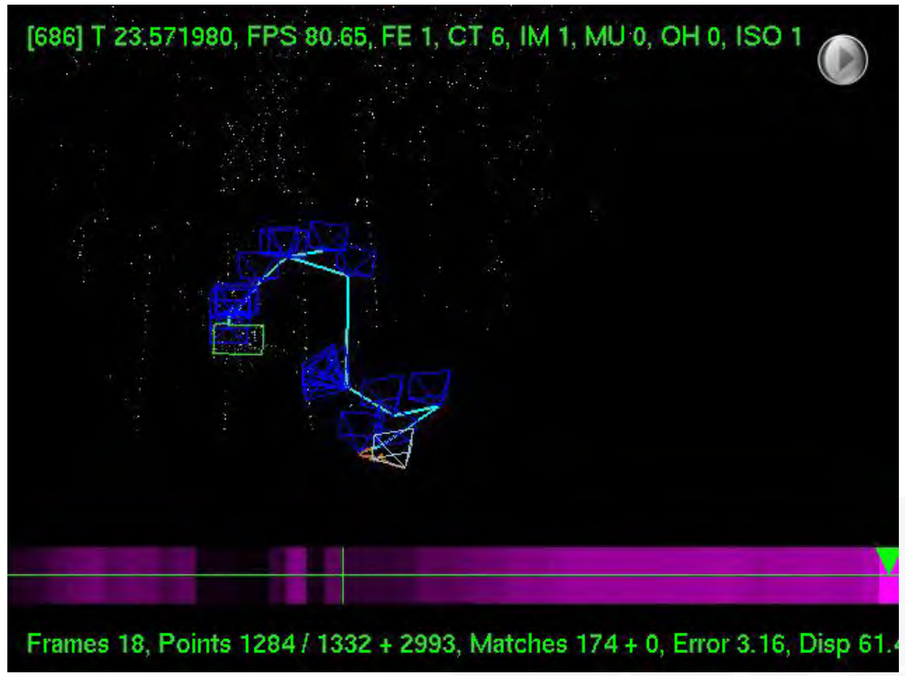
微软Kinect彩色-深度（RGBD）传感器



手机上的惯性传感器（IMU）

SLAM运行结果

- 设备根据传感器的信息
  - 计算自身位置（在空间中的位置和朝向）
  - 构建环境地图（稀疏或者稠密的三维点云）



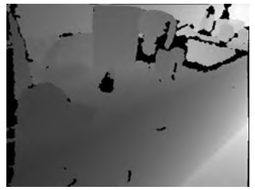
稀疏SLAM

稠密SLAM

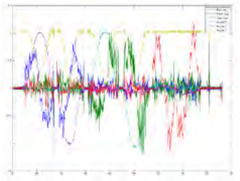
SLAM系统常用的框架

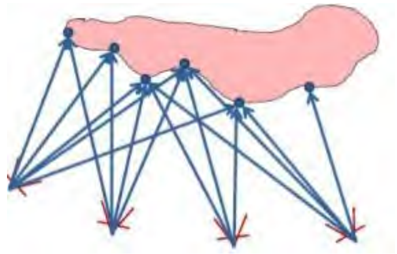
RGB图

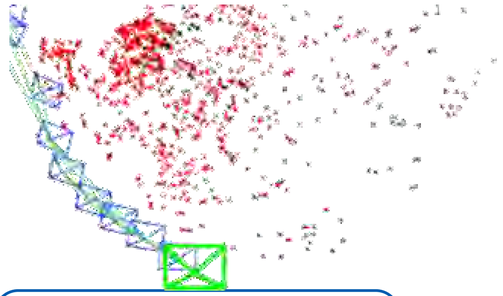
深度图



IMU测量值







输入

- 传感器数据

前台线程

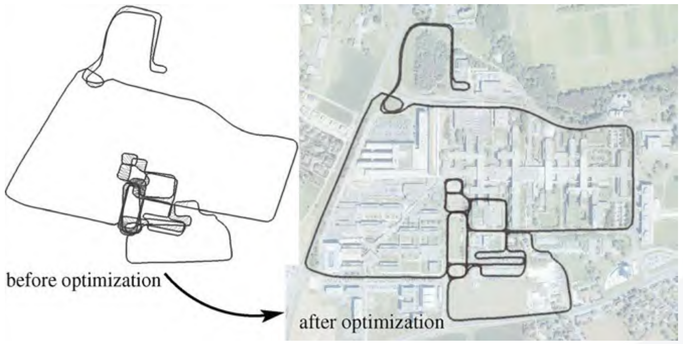
- 根据传感器数据进行跟踪求解，实时恢复每个时刻的位姿

输出

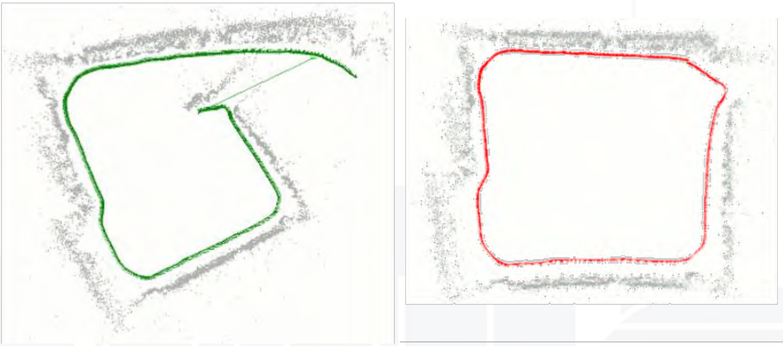
- 设备实时位姿
- 三维点云

后台线程

- 进行局部或全局优化，减少误差累积
- 场景回路检测



优化以减少误差累积



回路检测

SLAM应用介绍

- 扫地机器人

小米扫地机器人
以激光雷达为核心

戴森360°Eye扫地机器人
以视觉为核心（顶部有全景摄像头）

SLAM应用介绍

- 无人机

大疆Phantom4
结合双目立体视觉和超声波，实现空中精准悬停和安全航线自动生成

SLAM应用介绍

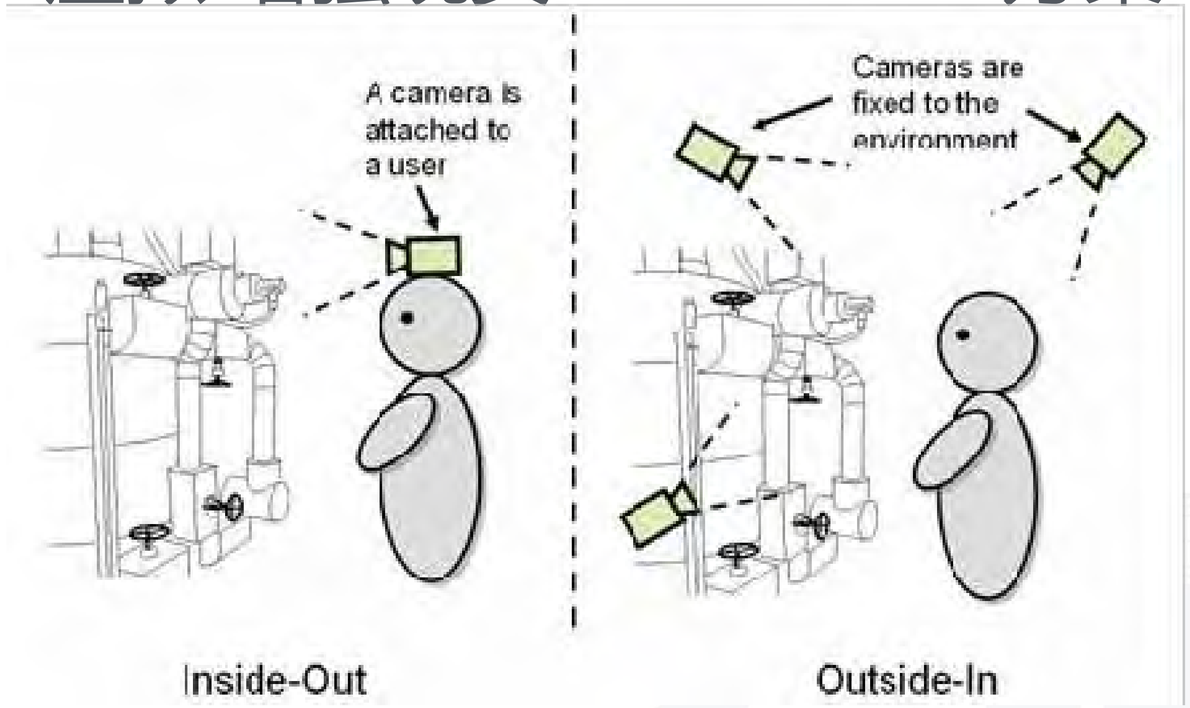
- 无人车

Google无人车项目Waymo
使用高精度激光雷达构建地图

MobileEye、特斯拉等自动驾驶方案
以廉价的摄像头为主

SLAM应用介绍

- 虚拟/增强现实：Inside-Out方案



《The Devices of VR: Part 3 - The Future of VR》

目前绝大多数VR头盔都采用Outside-In的定位方案，需要在环境中放置一个或多个传感器，活动范围受限，不支持大范围移动的定位。

基于SLAM技术的VR/AR可以实现Inside-Out方案：将传感器固定在使用者端。

优点：不需要提前布置环境中的传感器，且没有活动范围的限制。

SLAM应用介绍

- 增强现实：Google Tango

Google的Tango项目演示视频

Tango为终端开发者提供了从硬件到软件的整套AR开发套件

SLAM应用介绍

- 混合现实：微软HoloLens

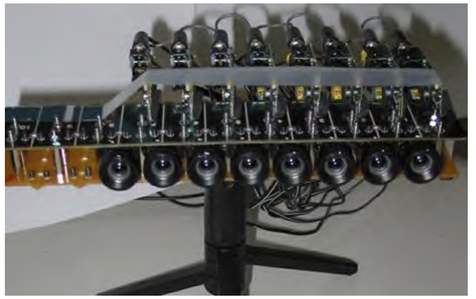
Hololens宣传视频

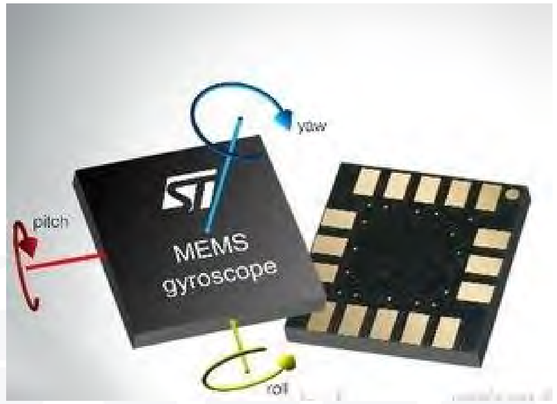
Hololens部分传感器
左右双目+前视RGB摄像头+深度传感器

HoloLens融合了场景位置感知和头盔显示技术，并提供了完整的软硬件解决方案。

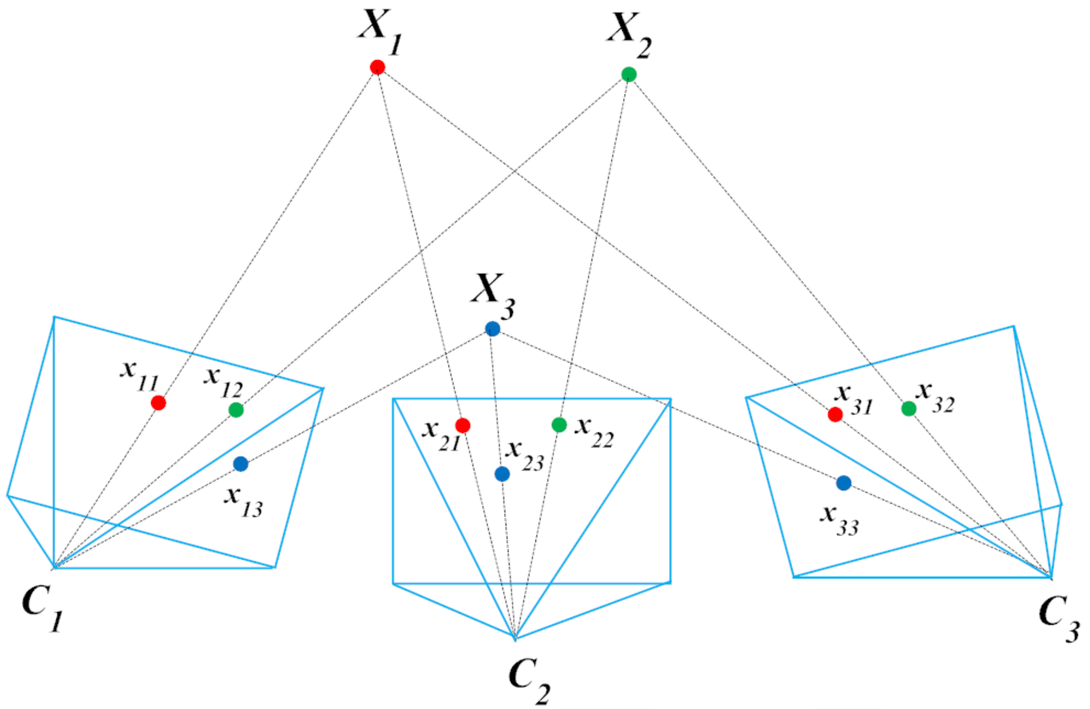
视觉SLAM

- 主要传感器
  - 单目摄像头
  - 双目摄像头
  - 多目摄像头
- 其它辅助传感器
  - 廉价IMU、GPS
  - 深度传感器
- 优势
  - 硬件成本低廉
  - 小范围内定位精度较高
  - 无需预先布置场景





基本原理：多视图几何



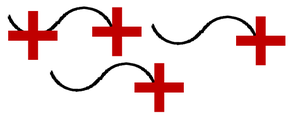
$$\mathbf{x}_{ij} = \pi(\mathbf{P}_i X_j)$$

投影函数 $\pi(x, y, z) = (x/z, y/z)$ $\mathbf{P}_i = \mathbf{K}_i[\mathbf{R}_i|\mathbf{T}_i]$

主要模块

- 特征跟踪
  - 获得一堆特征点轨迹

$$\mathcal{X} = \{\mathbf{x}_i | _{i=1,...,m}\}$$



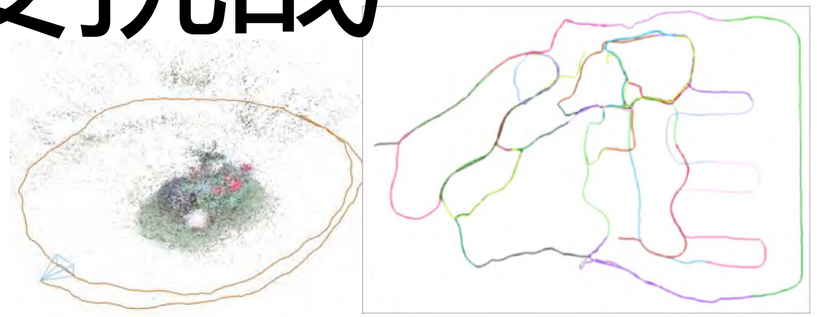
- 相机姿态恢复与场景三维结构恢复
  - 求解相机参数和三维点云

$$\mathbf{x}_{ij} = \pi(\mathbf{P}_i X_j) \qquad \mathbf{P}_i = \mathbf{K}_i[\mathbf{R}_i|\mathbf{T}_i]$$

$$E(\mathbf{P}_1, ..., \mathbf{P}_m, X_1, ..., X_n) = \sum_{i=1}^{m}\sum_{j}^{n} w_{ij}||\pi(\mathbf{P}_i X_j) - \mathbf{x}_{ij}||^2$$

复杂环境下的主要挑战

- 如何处理循环回路序列和多视频序列？
- 如何高效高精度地处理大尺度场景？
- 如何处理动态场景？
- 如何处理快速运动和强旋转？



我们课题组的工作

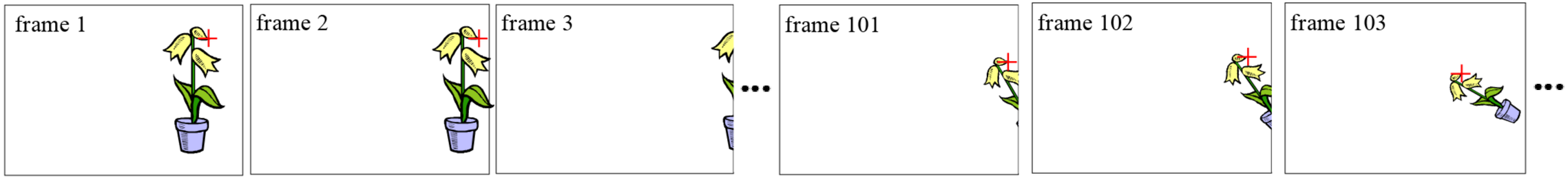
- 面向大尺度场景的运动恢复结构
  - ENFT-SFM：能够高效地处理大尺度场景下拍摄的循环回路和多视频序列。

- 单目视觉的同时定位与地图构建
  - ENFT-SLAM：能在大尺度场景下实时稳定工作、在线回路闭合；
  - RDSLAM：能在动态场景下稳定工作；
  - RKSLAM：可以实时运行在移动设备上，并能处理快速运动和强旋转 。

ENFT-SFM: Efficient Non-Consecutive Feature Tracking for Robust SFM

循环回路序列和多视频序列

- 如何将不同子序列上的相同特征点高效地匹配上？



- 如何高效地进行全局优化，消除重建漂移问题？

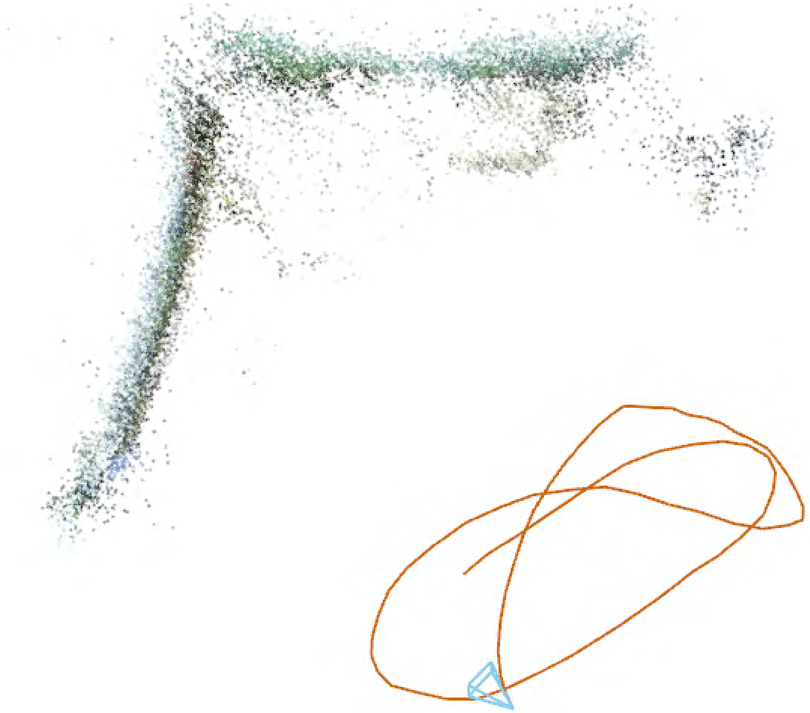
VisualSFM
结果

ENFT：高效的非连续帧特征跟踪

Consecutive Feature Tracking

Non-Consecutive Track Matching

Structure from Motion



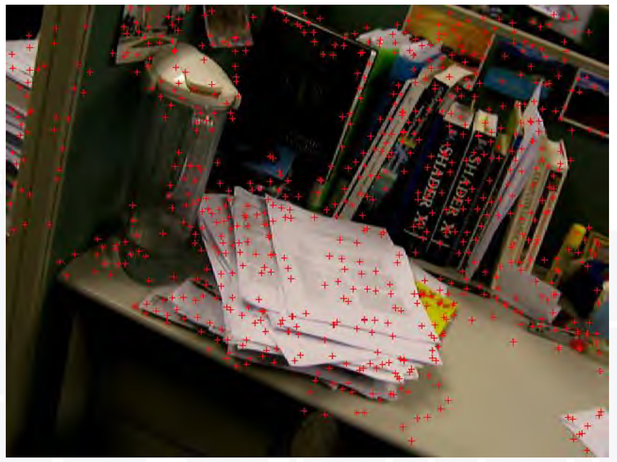
基于两道匹配的连续帧跟踪

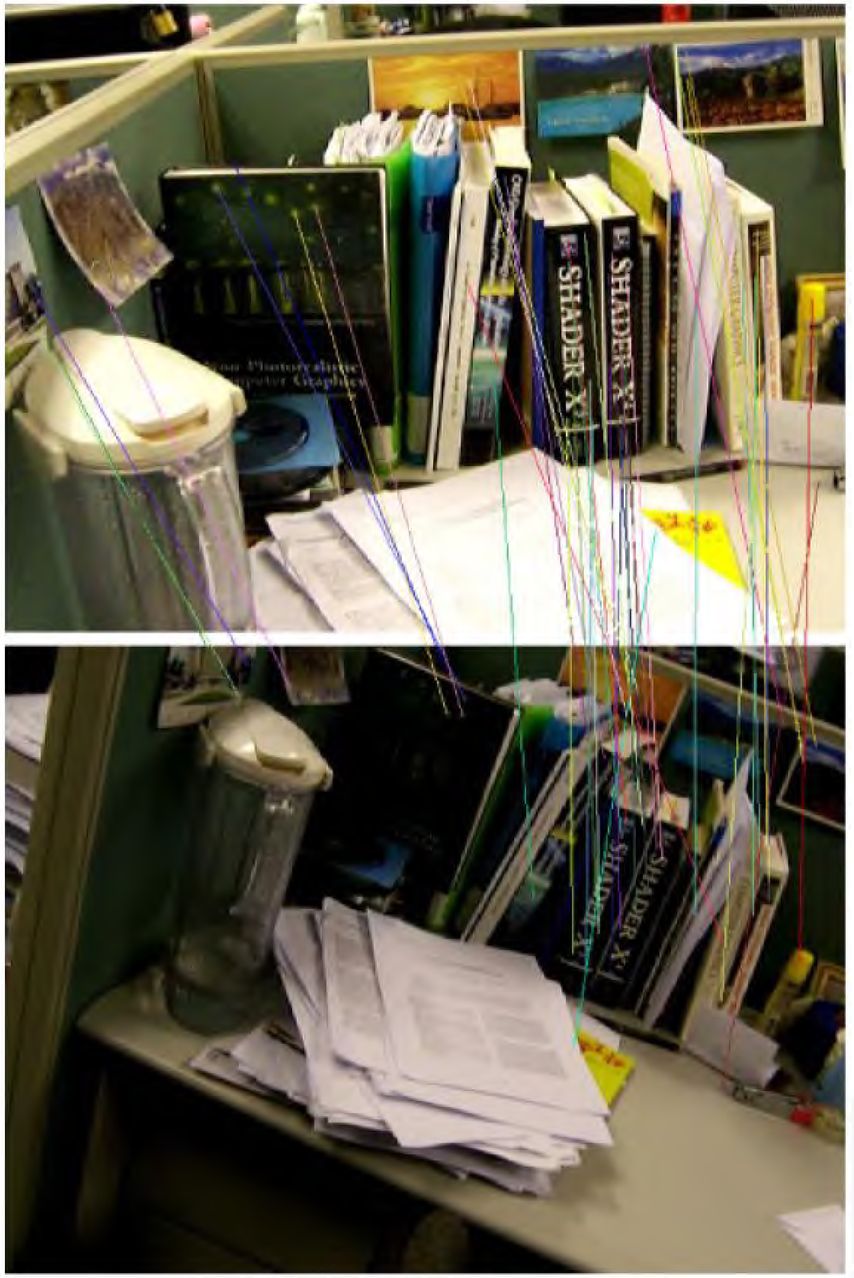
- 抽取SIFT特征

- 第一道匹配：比较描述量

$$c = \frac{||\mathbf{p}(\mathcal{N}_1^{t+1}(\mathbf{x}_t)) - \mathbf{p}(\mathbf{x}_t)||}{||\mathbf{p}(\mathcal{N}_2^{t+1}(\mathbf{x}_t)) - \mathbf{p}(\mathbf{x}_t)||}$$

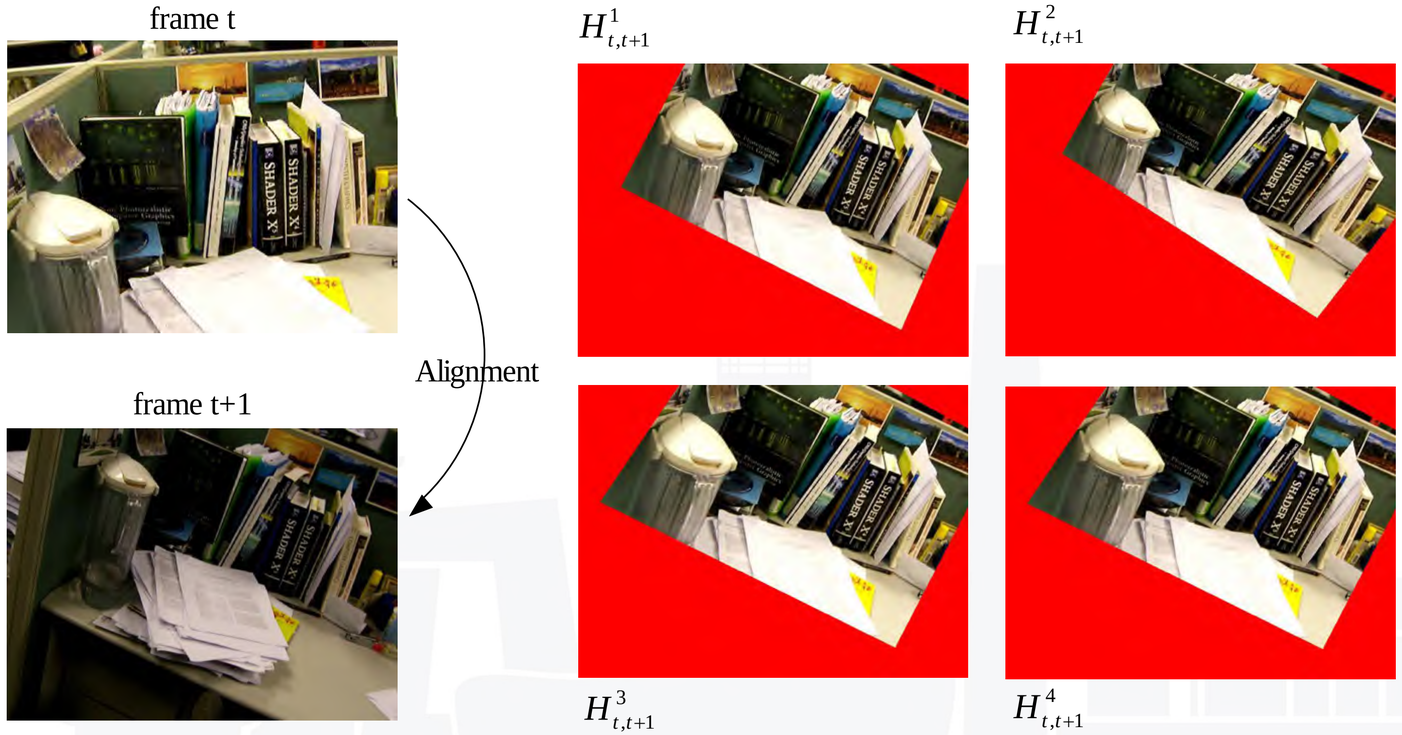
$c < \varepsilon$ Global distinctive




SHADER X
SHADER X

平面运动分割

- 估计若干个平面运动 $\{H^k_{t,t+1}|k=1,...,N\}$
  - 使用第一道匹配得到的内点匹配对（inlier matches）



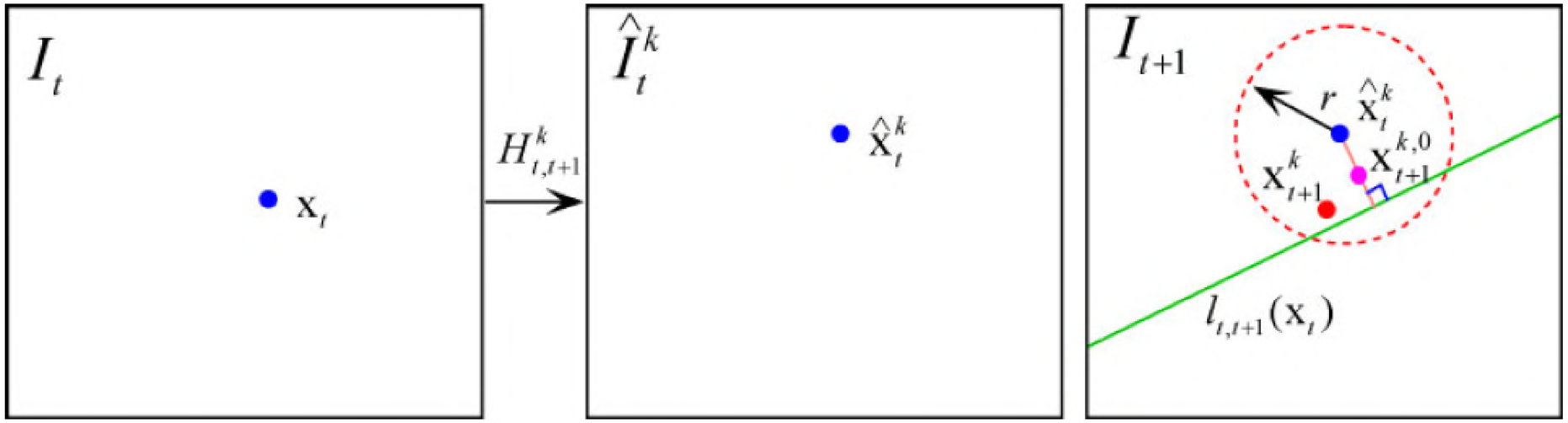
第二道匹配

- 根据估计的平面运动进行匹配

$$S^k_{t,t+1}(\mathbf{x}^k_{t+1}) = \sum_{\mathbf{y}\in W} ||\hat{I}^k_t(\hat{\mathbf{x}}^k_t + \mathbf{y}) - I_{t+1}(\mathbf{x}^k_{t+1} + \mathbf{y})||^2 +$$

$$\lambda_e d(\mathbf{x}^k_{t+1}, l_{t,t+1}(\mathbf{x}_t))^2 + \lambda_h ||\hat{\mathbf{x}}^k_t - \mathbf{x}^k_{t+1}||^2$$

Epipolar constraint　　Homography constraint



匹配结果比较

第一道匹配
(53个匹配对)

直接极线上搜索
(增加了11个匹配对)

第二道匹配
(增加了346个匹配对)

非连续帧上的特征点轨迹匹配

- 快速匹配矩阵估计

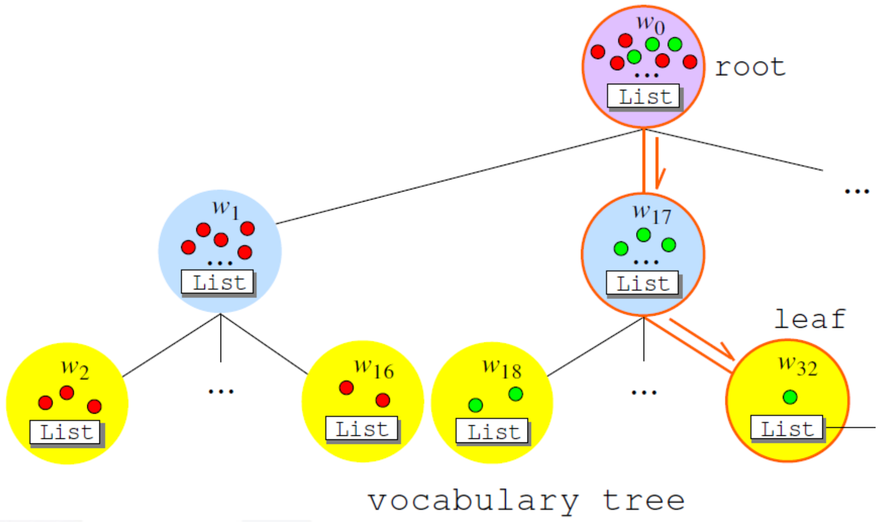
- 检测有公共内容的子序列进行特征轨迹匹配

快速匹配矩阵估计

- 每个轨迹有一组描述向量

$$\mathcal{P}_{\mathcal{X}} = \{\mathbf{p}(\mathbf{x}_t) | t \in f(\mathcal{X})\}$$

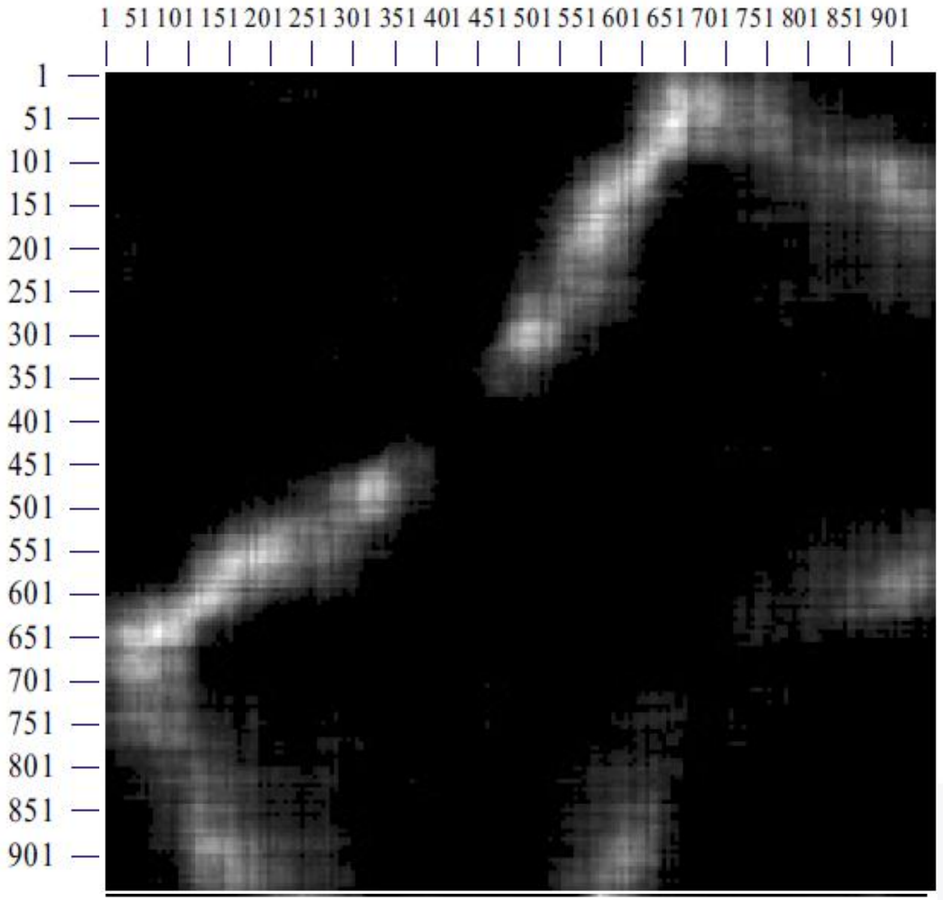
- 特征轨迹描述量 $\mathbf{p}(\mathcal{X})$



- 采用分层的K-means方法进行轨迹描述量聚类

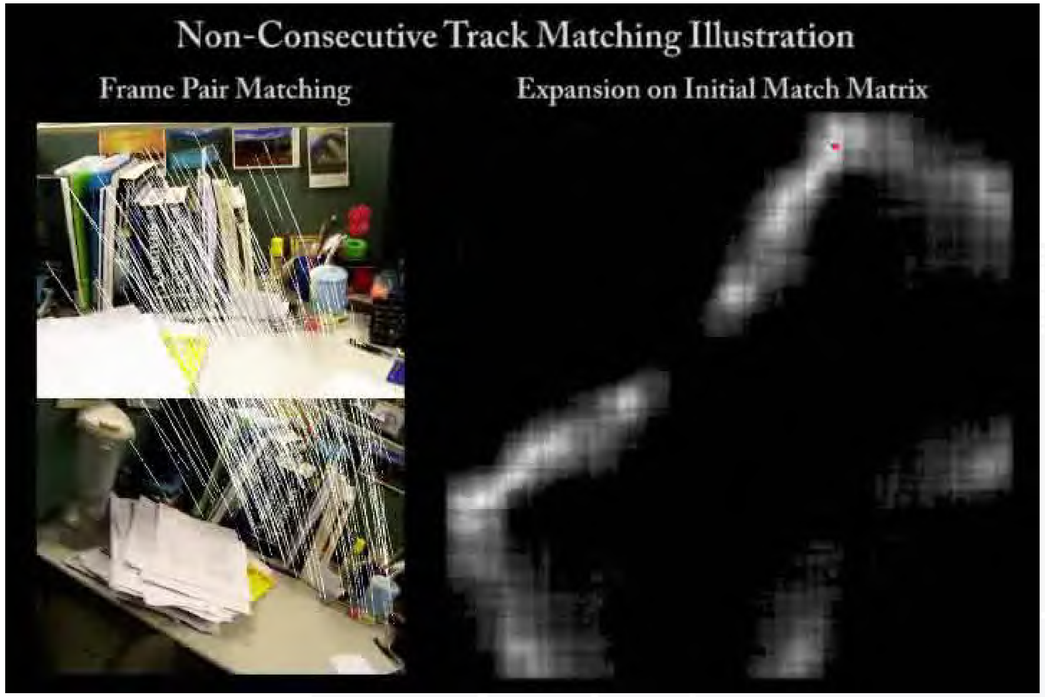
快速匹配矩阵估计

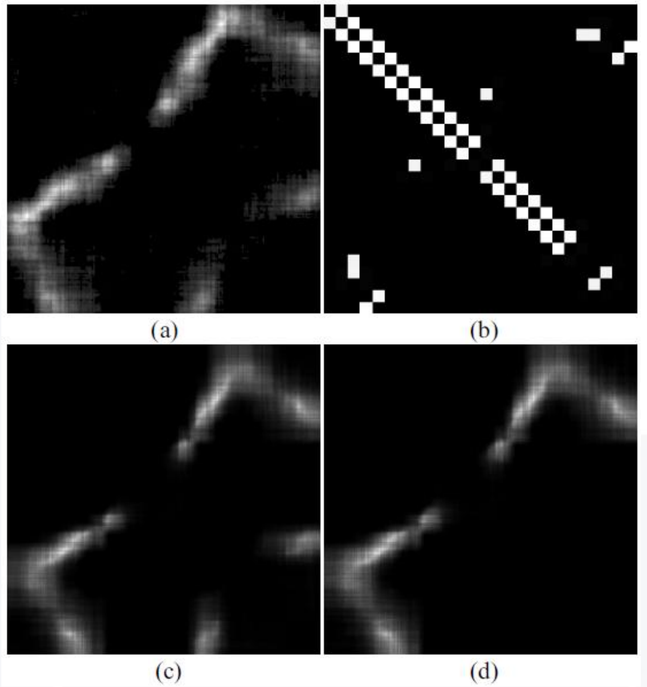

1
51
101
151
201
251
301
351
401
451
501
551
601
651
701
751
801
851
901
941


1 51 101 151 201 251 301 351 401 451 501 551 601 651 701 751 801 851 901
1
51
101
151
201
251
301
351
401
451
501
551
601
651
701
751
801
851
901

非连续特征轨迹匹配

- 同时进行图像对的特征匹配和优化匹配矩阵
  - 根据选择的图像对的特征匹配结果对匹配矩阵进行优化；
  - 根据更新的匹配矩阵更可靠地选择出有公共内容的图像对进行特征匹配。



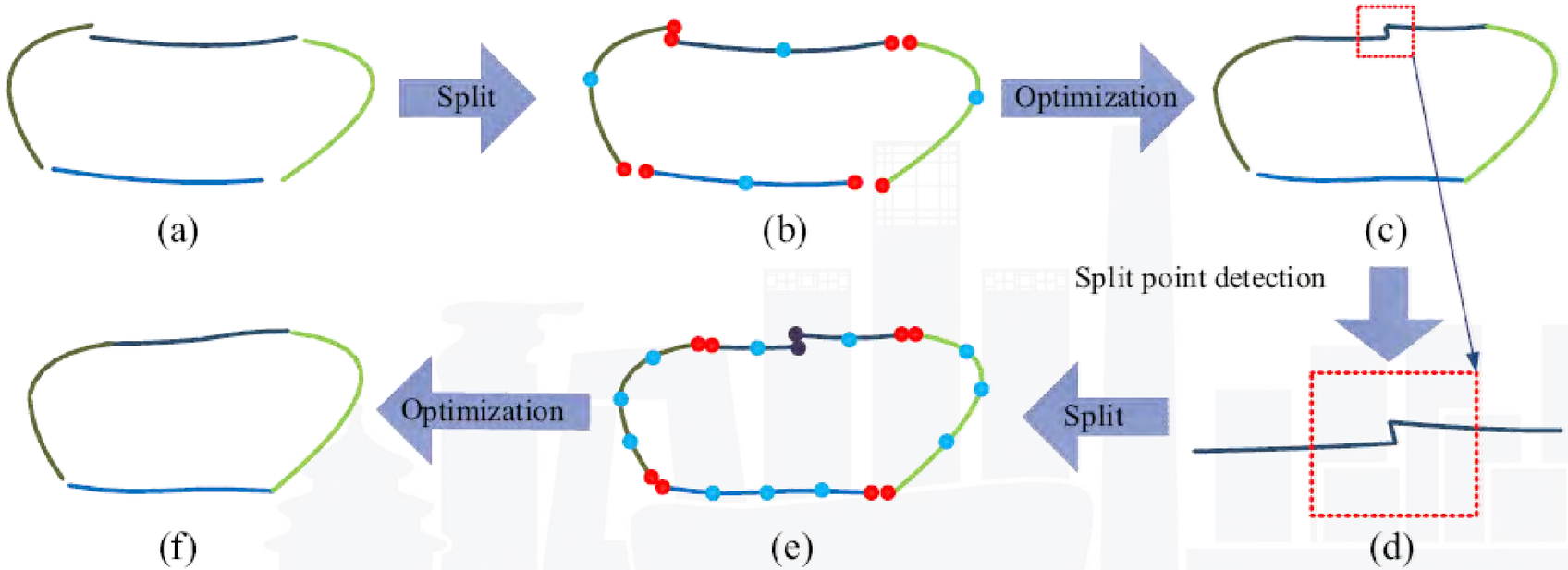


大尺度运动恢复结构的难点

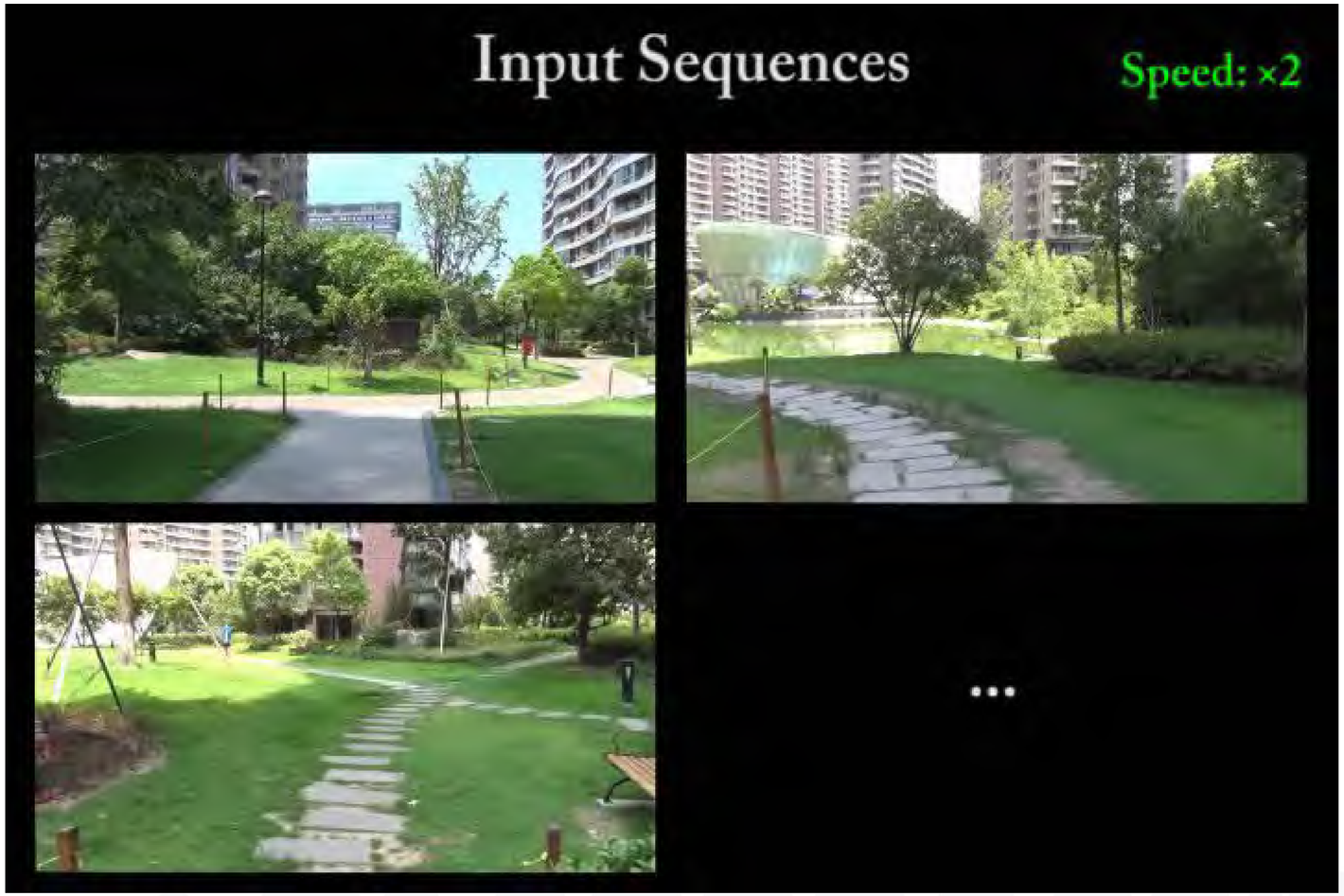
- 全局集束调整（Global Bundle Adjustment）
  - 变量数目非常庞大
  - 内存空间需求大
  - 计算耗时
- 迭代的局部集束调整
  - 大误差难以均匀扩散到整个序列
  - 极易陷入局部最优
- 姿态图优化（Pose Graph Optimization）
  - 只优化相机之间的相对姿态，三维点都消元掉；
  - 是集束调整的一个近似，不是最优解。

基于自适应分段的集束调整

- 将长序列分成若干段短序列；
- 每个短序列进行独立的SfM并根据公共匹配对进行对齐，每个段由7个自由度的相似变换控制；
- 如果投影误差比较大，检测分裂点将序列分段，然后优化；
- 重复上述步骤直至投影误差小于阈值或不能再分裂为止。



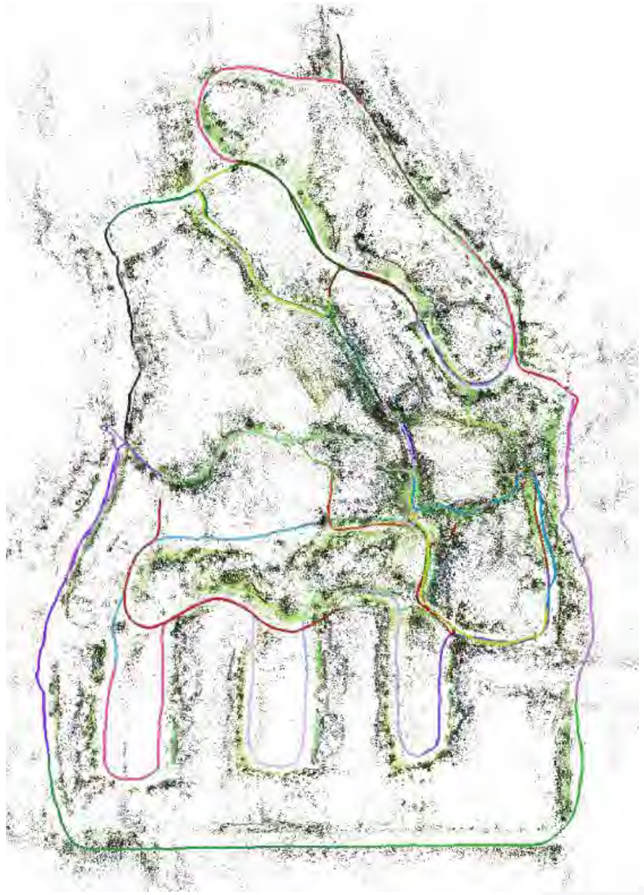
Garden数据集的SfM结果



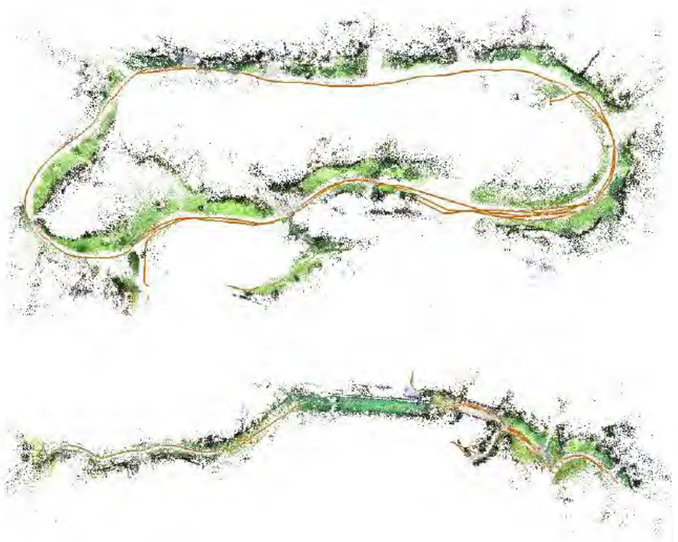
6段长视频序列，将近10万帧，特征匹配74分钟，SfM求解16分钟（单线程），平均17.7fps

VisualSFM：SfM求解 57 分钟 （GPU加速）

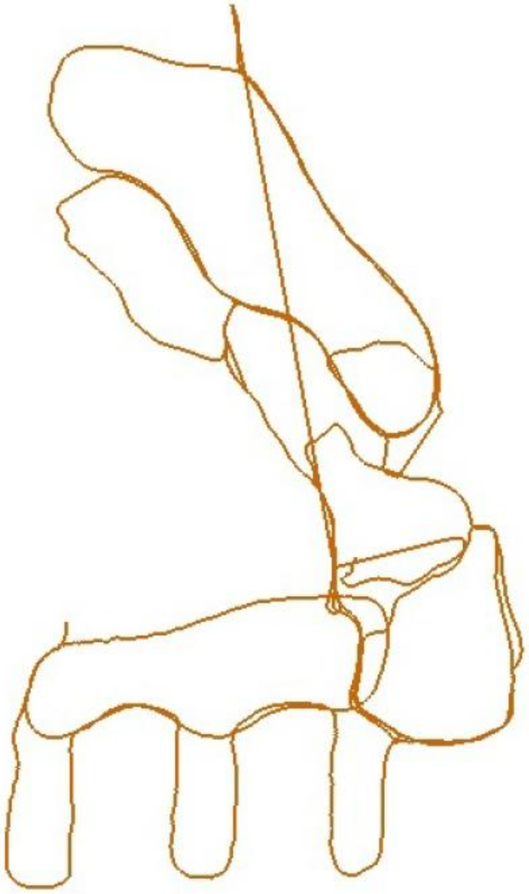
Garden数据集上的比较



ENFT-SFM



VisualSFM



ORB-SLAM

KITTI数据集上的定量比较

TABLE V
LOCALIZATION ERROR (RMSE (M)/COMPLETENESS) COMPARISON IN KITTI DATASET.

Seq.	ENFT-SFM	ENFT-SFM (Keyframes)	ORB-SLAM	VisualSFM (Keyframes)	OpenMVG (Keyframes)
00	4.58 / 100%	4.76 / 100%	5.33	2.78 / 3.71%	5.83 / 0.7%
01	57.20 / 100%	53.96 / 100%	X	52.34 / 12.46%	8.79 / 2.08%
02	28.13 / 100%	28.26 / 100%	21.28	1.77 / 4.53%	50.36 / 3.74%
03	2.82 / 100%	2.94 / 100%	1.51	0.28 / 12.05%	3.53 / 8.43%
04	0.66 / 100%	0.66 / 100%	1.62	0.76 / 23.44%	5.14 / 14.06%
05	2.88 / 100%	3.48 / 100%	4.85	9.77 / 7.42%	22.42 / 9.07%
06	14.24 / 100%	14.43 / 100%	12.34	8.58 / 7.41%	3.16 / 3.37%
07	1.83 / 100%	2.03 / 100%	2.26	3.85 / 7.78%	7.75 / 5%
08	30.74 / 100%	28.32 / 100%	46.68	0.81 / 0.90%	17.82 / 2.58%
09	5.63 / 100%	5.88 / 100%	6.62	0.90 / 4.92%	14.26 / 3.36%
10	19.53 / 100%	18.49 / 100%	8.8	5.70 / 6.05%	27.06 / 7.01%

TUM数据集上的定量比较

TABLE VI
LOCALIZATION ERROR (RMSE (CM)/COMPLETENESS) COMPARISON IN TUM RGB-D BENCHMARK.

Sequence	ENFT-SFM	ENFT-SFM (Keyframes)	ORB-SLAM	VisualSFM (Keyframes)	OpenMVG (Keyframes)
fr1_desk	2.71/99.84%	2.96/100%	1.69	2.74 / 100%	X
fr1_floor	4.08/96.70%	3.93/100%	2.99	53.11/69.23%	0.52/6.92%
fr1_xyz	1.25/100%	1.59/100%	0.9	1.43/100%	X
fr2_360_kidnap	13.57/91.47%	15.31/100%	3.81	10.08/50.91%	5.21/14.55%
fr2_desk	2.43/100%	2.27/100%	0.88	1.79/100%	1.38/13.95%
fr2_desk_person	2.46/100%	2.55/100%	0.63	1.92/100%	2.16/97.01%
fr2_xyz	0.81/100%	0.73/100%	0.3	0.71/100%	5.74/97.6%
fr3_long_office	1.21/100%	1.44/100%	3.45	1.15/100%	2.94/32.74%
fr3_nst_tex_far	3.60/86.58%	7.76/100%	X	7.29/100%	35.64/3.79%
fr3_nst_tex_near	1.87/100%	1.66/100%	1.39	1.13/100%	3.4/39.13%
fr3_sit_half	1.50/100%	1.55/100%	1.34	2.30/100%	0.68/9.3%
fr3_sit_xyz	0.84/100%	1.39/100%	0.79	1.28/100%	1.03/100%
fr3_str_tex_far	0.94/100%	0.95/100%	0.77	2.15/100%	1.12/100%
fr3_str_tex_near	1.86/100%	1.82/100%	1.58	0.95/100%	0.97/19.74%
fr3_walk_half	2.08/100%	2.21/100%	1.74	1.88/100%	X
fr3_walk_xyz	1.30/100%	1.74/100%	1.24	1.62/100%	X

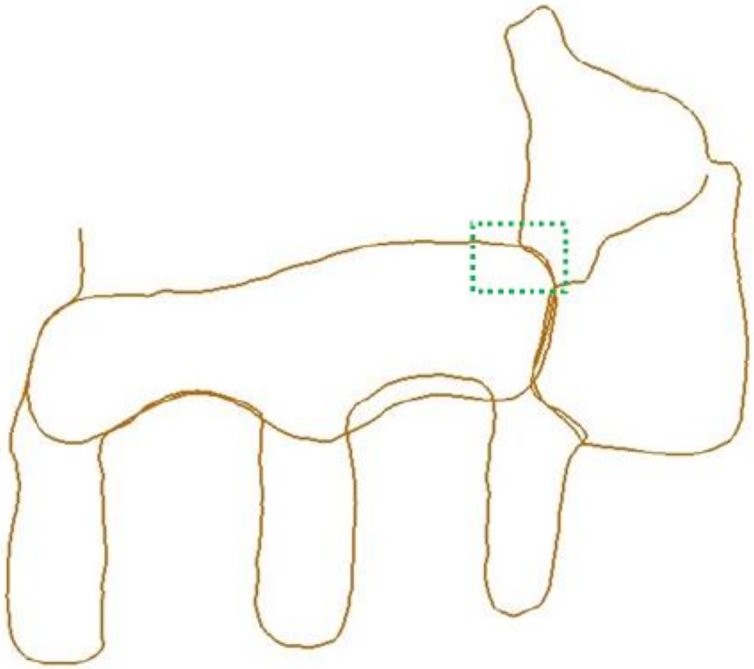
ENFT-SLAM: ENFT-based Large-Scale Monocular SLAM

ENFT-SLAM

- 特征跟踪
  - 直接采用ENFT特征跟踪
- 回路检测与闭合
  - 对原来的非连续特征轨迹匹配进行修改
    - 计算当前帧与历史关键帧的相似度，并选择相似度高的关键帧进行匹配
  - 采用基于分段的集束调整进行优化

Garden序列上的实时SLAM

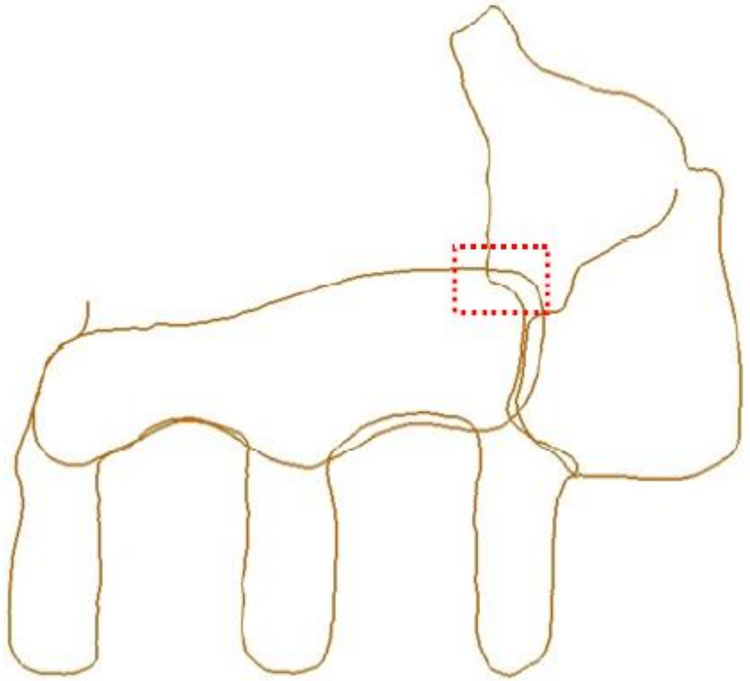
Garden序列结果比较



ENFT-SLAM

Non-consecutive Track Matching

Segment-based BA

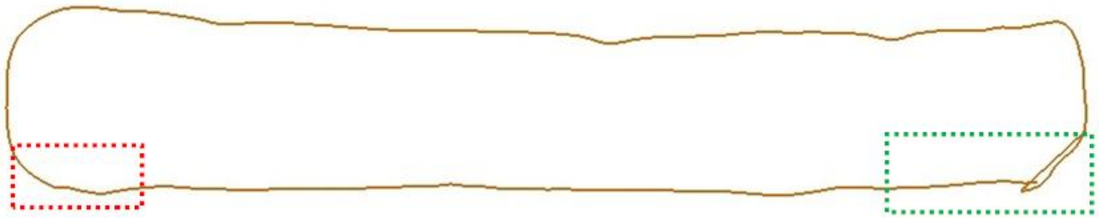


ORB-SLAM

Bag-of-words Place Recognition

Pose Graph Optimization + Traditional BA

Street序列结果比较



ENFT-SLAM

Non-consecutive Track Matching

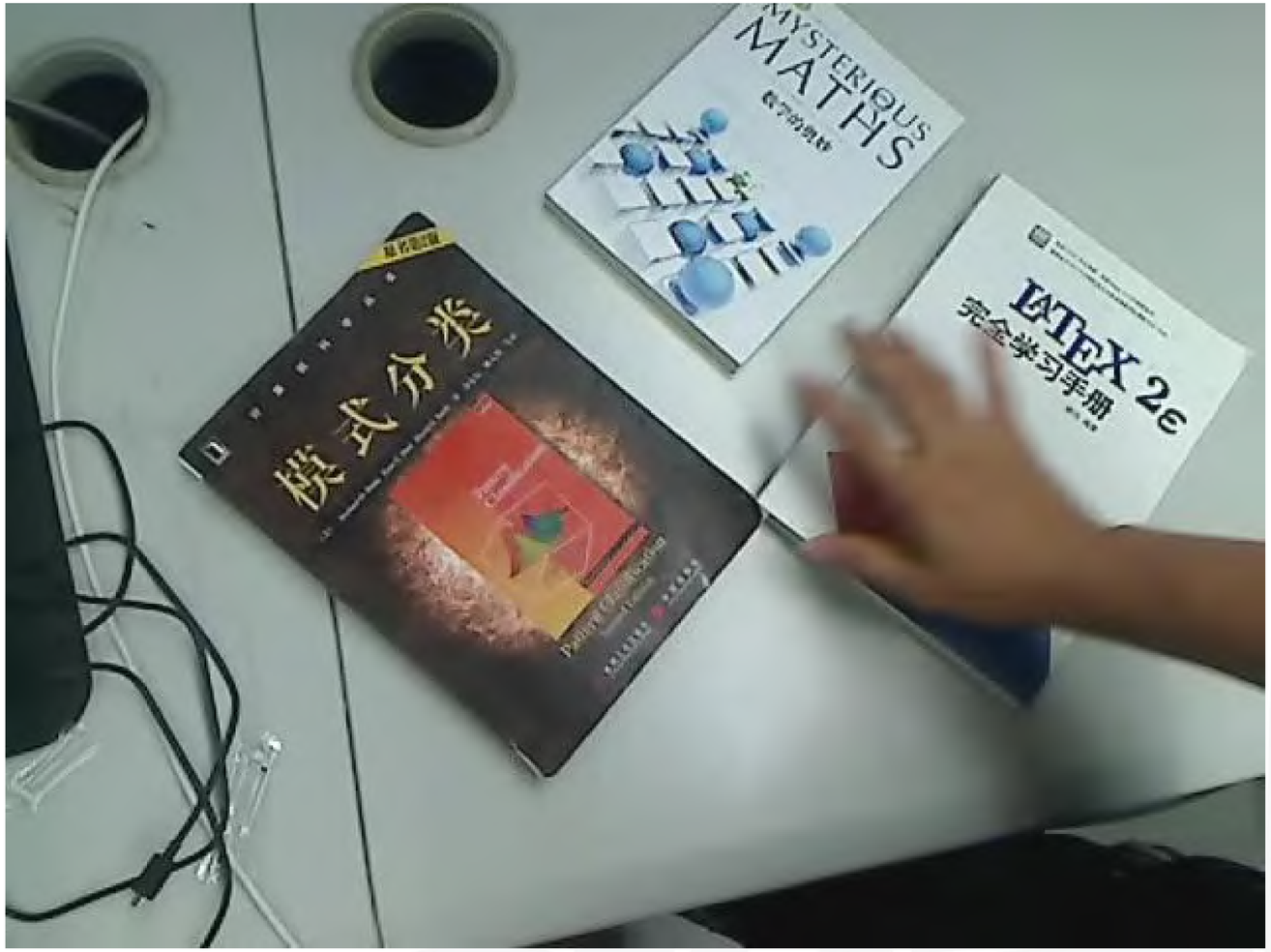
Segment-based BA

ORB-SLAM

Bag-of-words Place Recognition

Pose Graph Optimization + Traditional BA

动态场景SLAM的主要问题（1）



场景逐渐在改变

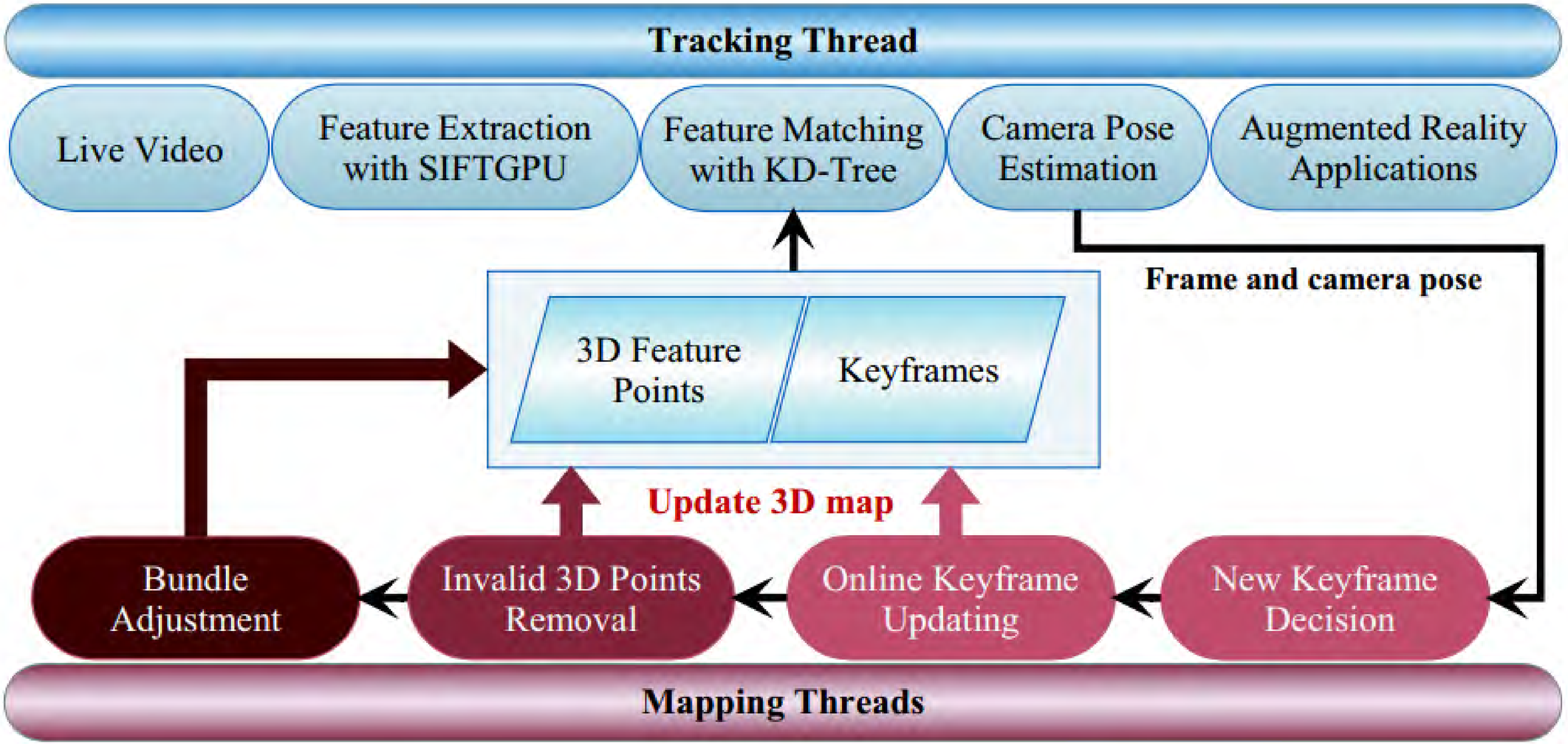
可能有大量的错误匹配

动态场景SLAM的主要问题（2）

视点改变造成的遮挡

运动物体造成的遮挡

RDSLAM框架

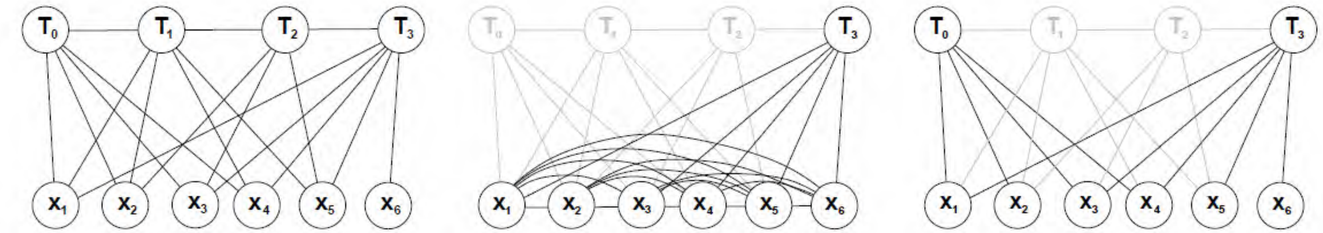

Tracking Thread
Live Video
Feature Extraction with SIFTGPU
Feature Matching with KD-Tree
Camera Pose Estimation
Augmented Reality Applications
Frame and camera pose
3D Feature Points
Keyframes
Update 3D map
Bundle Adjustment
Invalid 3D Points Removal
Online Keyframe Updating
New Keyframe Decision
Mapping Threads

结果与比较

RKSLAM: Robust Keyframe-based Monocular SLAM for Augmented Reality

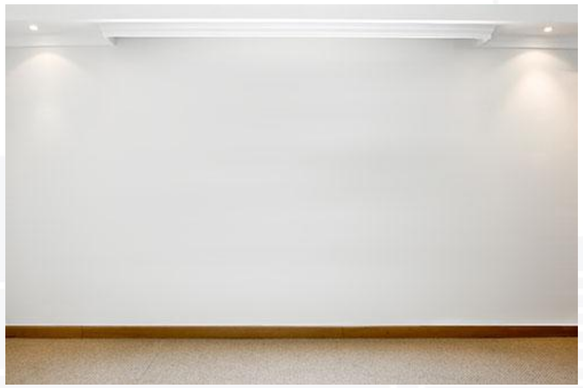
Keyframe-based SLAM vs Filtering-based SLAM

- 优点
  - 精度高
  - 效率高
  - 扩展性好
- 缺点
  - 对强旋转很敏感
- 共同的挑战
  - 快速运动
  - 运动模糊
  - 特征不够丰富



(a) Markov Random Field (b) Filter (c) Keyframe BA

H. Strasdat, J. Montiel, and A. J. Davison. Visual SLAM: Why filter? Image and Vision Computing, 30:65-77, 2012.



Visual-Inertial SLAM

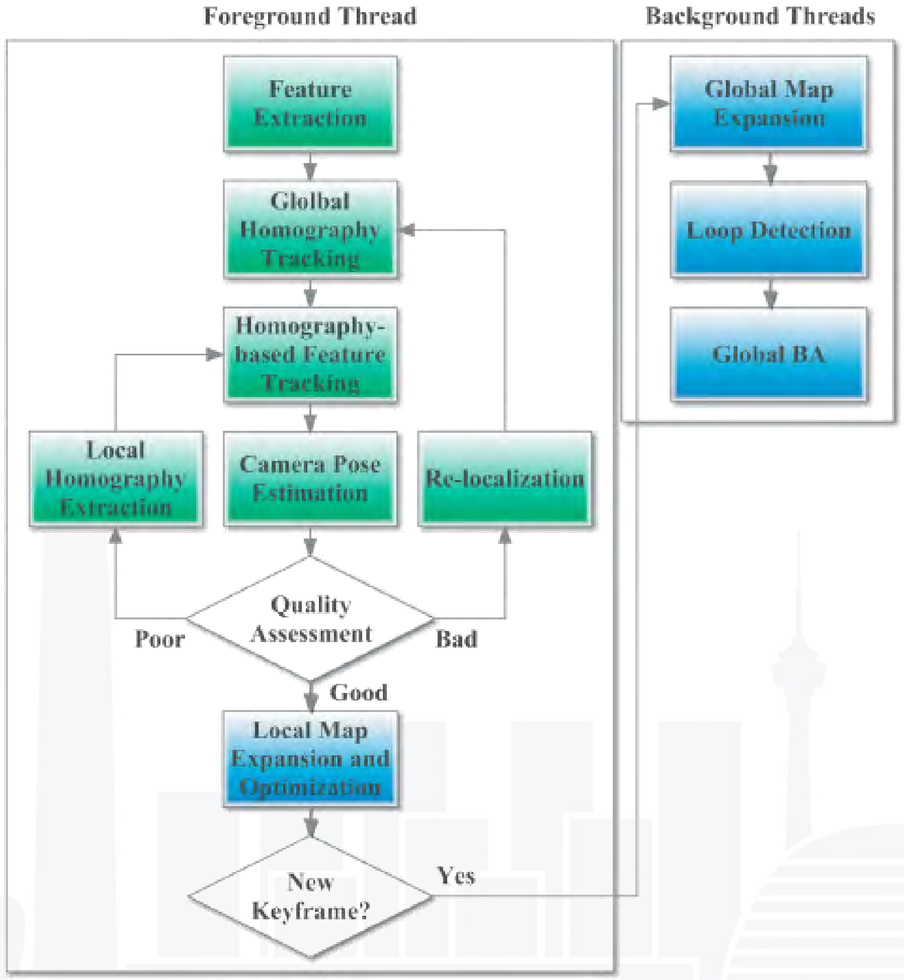
- 使用IMU数据提高鲁棒性
  - 基于滤波的方法
    - MSCKF, SLAM in Project Tango, …
  - 基于非线性优化的方法
    - OKVIS, …
- 没有真实IMU数据的情况下，是否能够通过视觉的方法来模拟IMU数据?

RKSLAM

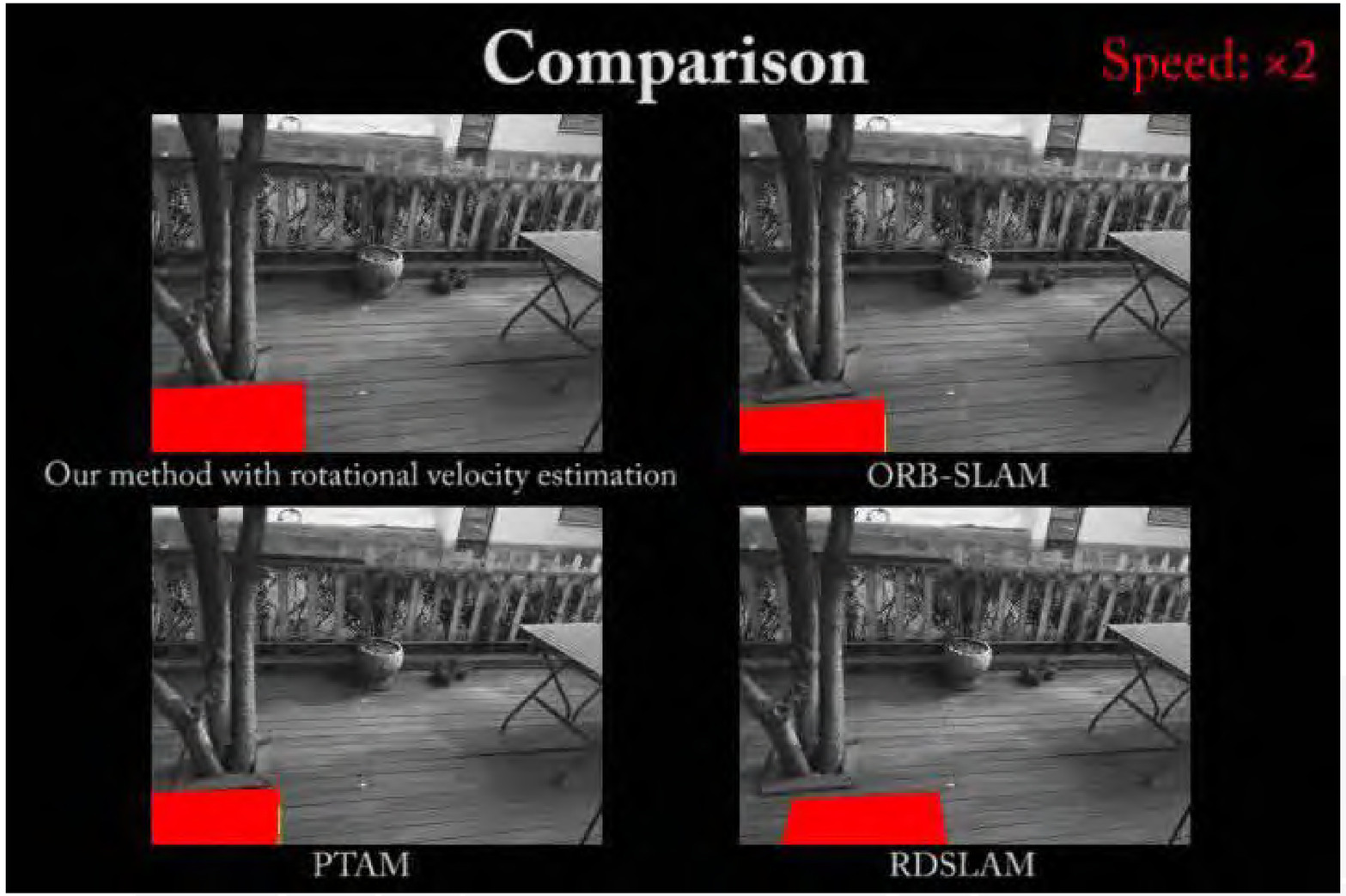
- 基于多单应矩阵的跟踪
  - 全局单应矩阵
  - 三维平面单应矩阵
  - 局部单应矩阵
- 基于滑动窗口的姿态优化
  - 用整张图像对齐来估计旋转角速度

$$\hat{\omega}_i = \arg\min_{\omega}\Big(\sum_{x\in\Omega}||\tilde{I}_i(\mathbf{x}) - \tilde{I}_{i+1}(\pi(\mathbf{K}\mathbf{R}_\Delta(\boldsymbol{\omega}, t_{\Delta_i})\mathbf{K}^{-1}\mathbf{x}^h))||_{\delta_\mathbf{I}}$$
$$+ \sum_{(\mathbf{x}_i,\mathbf{x}_{i+1})\in M_{i,i+1}} \frac{1}{\delta_\mathbf{x}}||\pi(\mathbf{K}\mathbf{R}_\Delta(\boldsymbol{\omega}, t_{\Delta_i})\mathbf{K}^{-1}\mathbf{x}_i^h) - \mathbf{x}_{i+1}||_2^2\Big)$$

  - 采用模拟的IMU数据进行姿态优化



结果与比较


Comparison
Speed: ×2
Our method with rotational velocity estimation
ORB-SLAM
PTAM
RDSLAM

TUM RGB-D数据集上的定量比较

Group	Sequence	RKSLAM	ORB-SLAM	PTAM	LSD-SLAM
A	fr1_xyz	0.61/0%/100%	1.05/0%/100%	1.29/0%/100%	7.64/0%/100%
A	fr2_xyz	0.43/0%/100%	0.23/0%/100%	0.29/0%/100%	6.32/0%/100%
A	fr3_sitting_xyz	1.98/0%/92%	1.31/5%/100%	X	9.12/0%/100%
B	fr1_desk	1.69/0%/100%	1.40/12%/100%	2.71/0%/44%	3.86/27%/100%
B	fr2_desk	10.10/0%/97%	0.78/6%/100%	0.55/0%/20%	17.41/0%/100%
B	fr3_long_office	2.48/0%/100%	2.17/0%/100%	0.82/0%/31%	36.04/30%/100%
C	fr1_rpy	1.26/0%/100%	5.53/4%/84%	X	3.26/0%/11%
C	fr2_rpy	0.41/0%/100%	0.23/32%/100%	0.56/0%/100%	3.71/0%/25%
C	fr3_sitting_rpy	1.44/0%/100%	0.19/93%/100%	2.44/0%/93%	3.36/0%/89%
D	fr1_360	11.81/0%/95%	8.16/5%/11%	X	8.25/0%/5%
D	fr2_360_hemisphere	17.48/0%/88%	12.27/1%/65%	76.50/0%/33%	25.64/0%/19%
D	fr2_pioneer_360	20.24/0%/86%	1.40/69%/46%	59.09/0%/98%	30.62/0%/41%

From left to right: RMSE (cm) of keyframes, the starting ratio (i.e. dividing the initialization frame index by the total frame number), and the tracking success ratio after initialization.

Group A: simple translation
Group B: there are loops
Group C: slow and nearly pure rotation
Group D: fast motion with strong rotation

时间统计

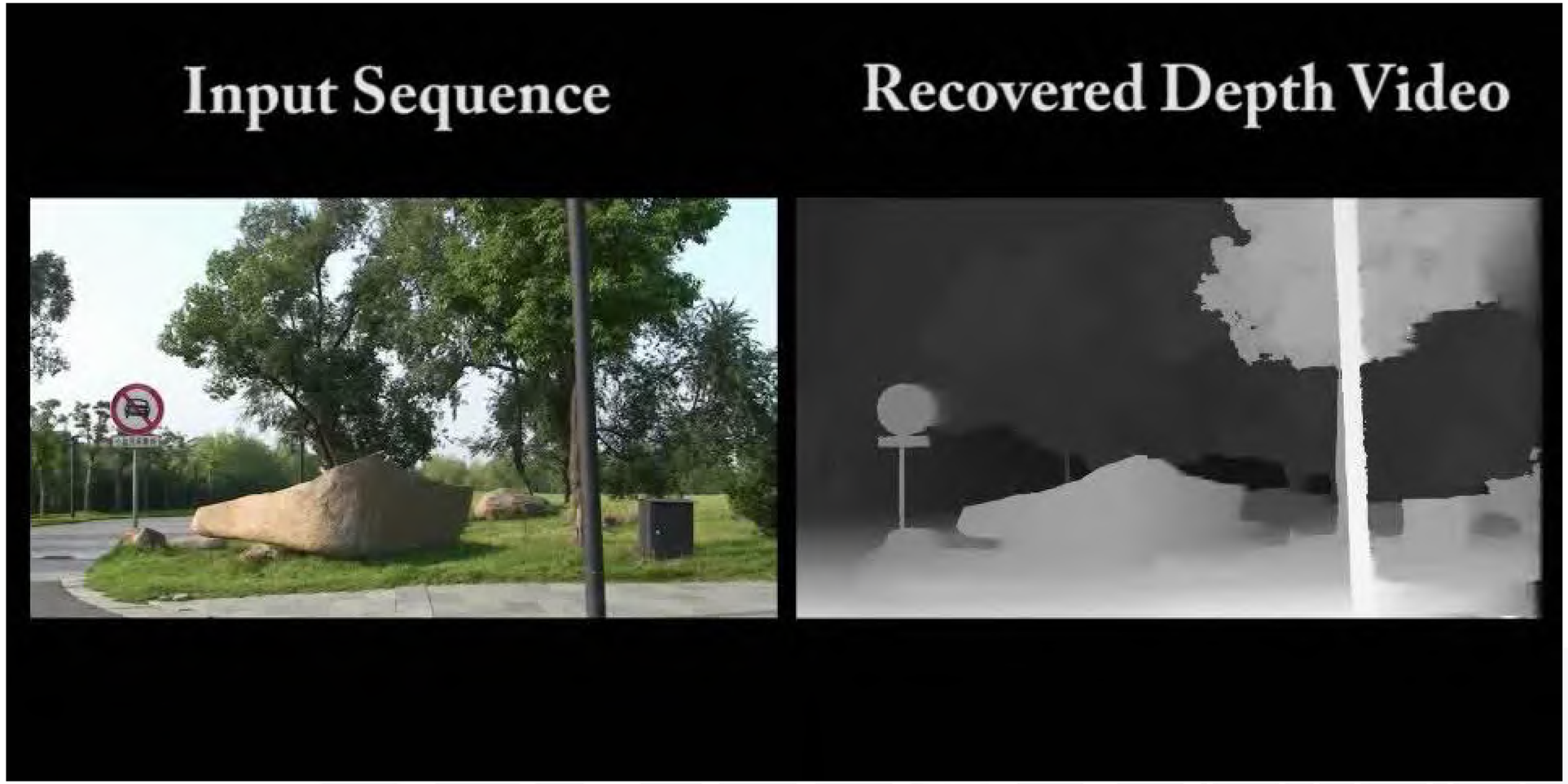
- 台式机上的计算时间

Module	Time per frame
Feature extraction	$\sim$ 2 ms
Feature tracking	2 $\sim$ 8 ms
Local map expansion and optimization	2 $\sim$ 4 ms

Table 1: Process time per frame with a single thread.

- 移动终端上
  - 20~50 fps on an iPhone 6.

时空一致性深度恢复

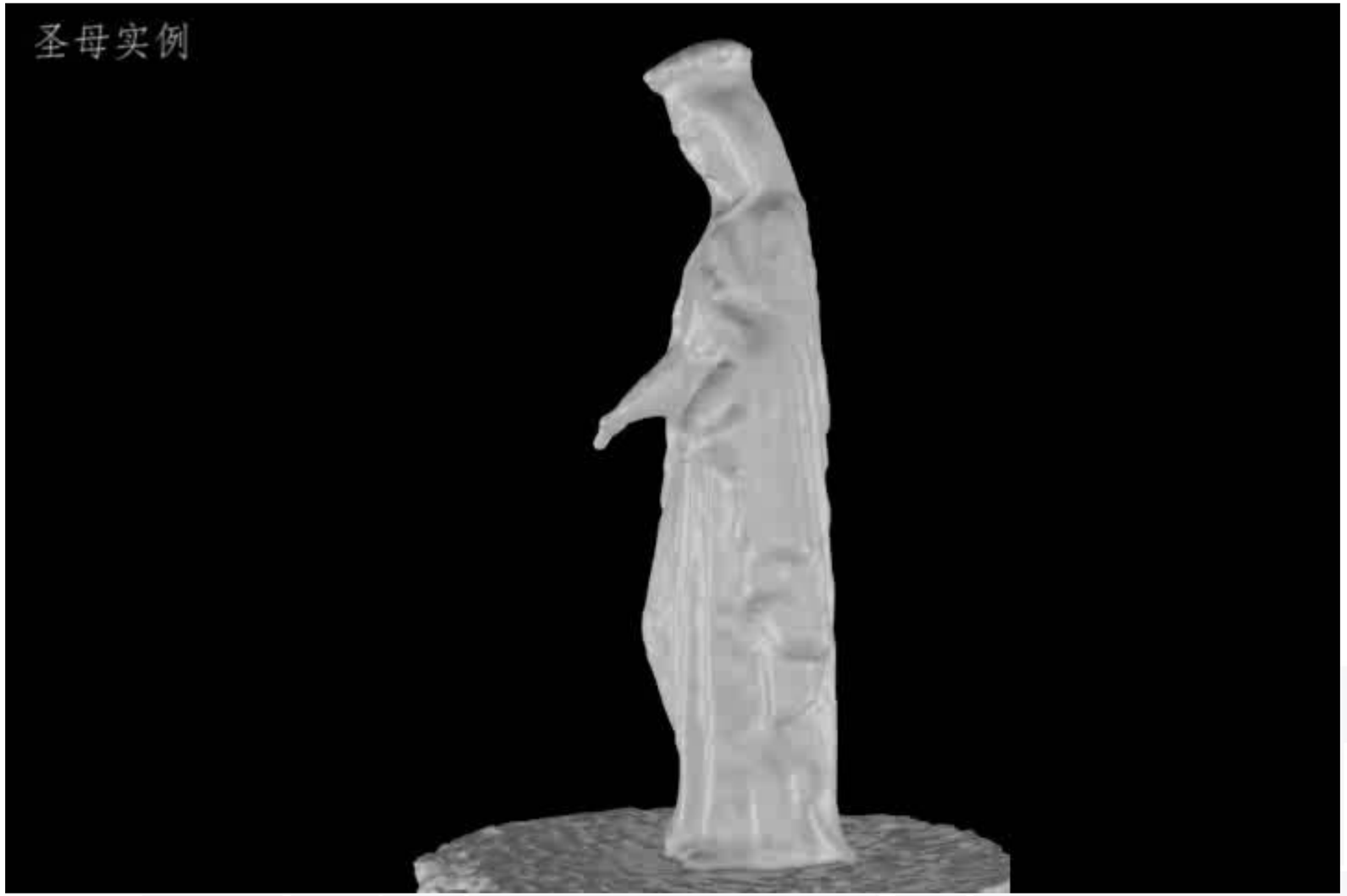


- Guofeng Zhang, Jiaya Jia, Tien-Tsin Wong, and Hujun Bao. Consistent Depth Maps Recovery from a Video Sequence. IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI), 31(6):974-988, 2009.

典型应用

- 三维重建
- 视频场景编辑

三维重建



三维重建

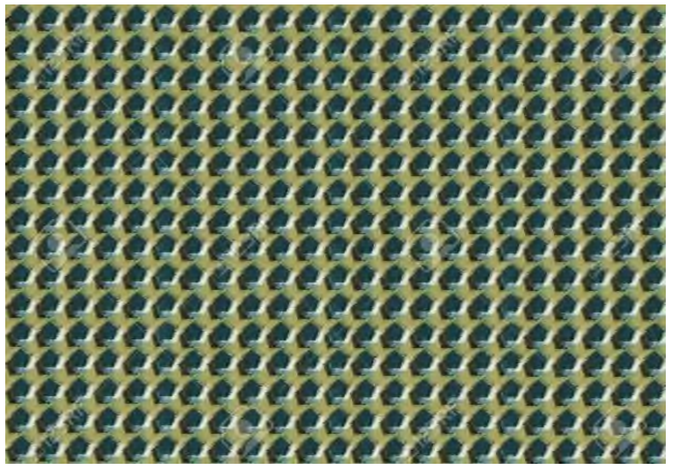
视频场景编辑

软件或源代码

- ENFT-SFM or LS-ACTS
  - http://www.zjucvg.net/ls-acts/ls-acts.html
- RKSLAM:
  - http://www.zjucvg.net/rkslam/rkslam.html
- RDSLAM:
  - http://www.zjucvg.net/rdslam/rdslam.html
- ACTS:
  - http://www.zjucvg.net/acts/acts.html

总结与讨论

- 极度缺乏特征和大量重复纹理场景下还工作得不好
  - 结合边跟踪或直接稠密跟踪
  - 融合其它传感器



- 目前只能实现实时的稀疏重建
  - 加速稠密深度恢复
  - 采用RGB-D相机

视觉SLAM技术发展趋势

- 缓解特征依赖
  - 结合边的跟踪
  - 直接图像跟踪或半稠密跟踪
- 朝实时稠密三维重建发展
  - 单目实时三维重建
  - 多目实时三维重建
  - 基于深度相机的实时三维重建
- 多传感器融合
  - 结合IMU、GPS、深度相机、光流计、里程计等

未来工作展望

- 协同SLAM
- 稠密SLAM
- 场景分析和理解
- 在VR/AR、机器人和无人驾驶领域进行应用

ZJUCVG Group Website: http:www.zjucvg.net

Personal Homepage: http:www.cad.zju.edu.cn/home/gfzhang

Email: zhangguofeng@cad.zju.edu.cn

关注QCon微信公众号，
获得更多干货！

Thanks!

INTERNATIONAL SOFTWARE DEVELOPMENT CONFERENCE

主办方 Geekbang. 极客邦科技 InfoQ