Survey for Dynamic SLAM

Guan Weipeng
Email: wpguan@connect.hku.hk

The University of Hong Kong

Department of Mechanical Engineering

Adaptive Robotic Controls Lab (ArcLab)

Dynamic SLAM

Motivation:

- Most SLAM labels the dynamic contents as outliers, they assume the environment to be static or mostly static;
- While it is susceptible to failure in complex dynamic environment;
- **Objects can provide long-range geometric and scale constraints** to improve camera pose estimation and reduce monocular drift.
- Instead of treating dynamic regions as outliers, we can utilize **object representation and motion model constraints** to improve the camera pose estimation.

Three classification:

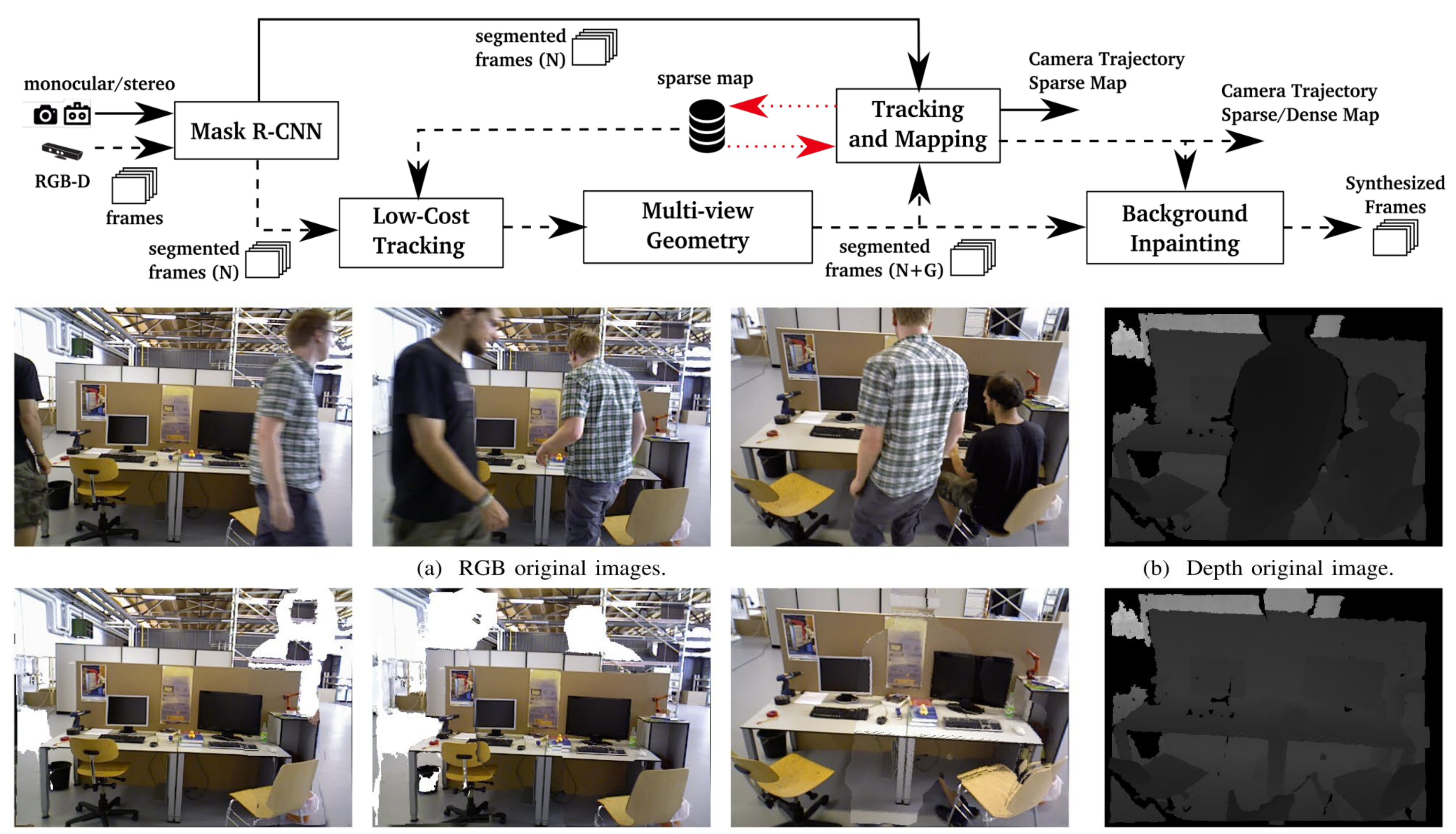
1. Detecting moving objects and **track them separately from the SLAM** formulation by using traditional multitarget tracking approaches; **Their accuracy highly depends on the camera pose estimation**, which is more susceptible to failure in complex dynamic environments where the presence of reliable static structure is not guaranteed (**only for 6 DoF dynamic object tracking**);
2. Focus on achieving an accurate ego-motion estimation from the static scene (**detecting and removing the dynamic region**);
3. Do not only solve the SLAM problem but also provide information about the poses of other dynamic agents (**jointly optimization**);

DynaSLAM: Tracking, Mapping and Inpainting in Dynamic Scenes

RAL 2018

香 港 大 學
THE UNIVERSITY OF HONG KONG

- Dynamic object detection and background inpainting;
- Classification 2;



(a) RGB original images. (b) Depth original image.

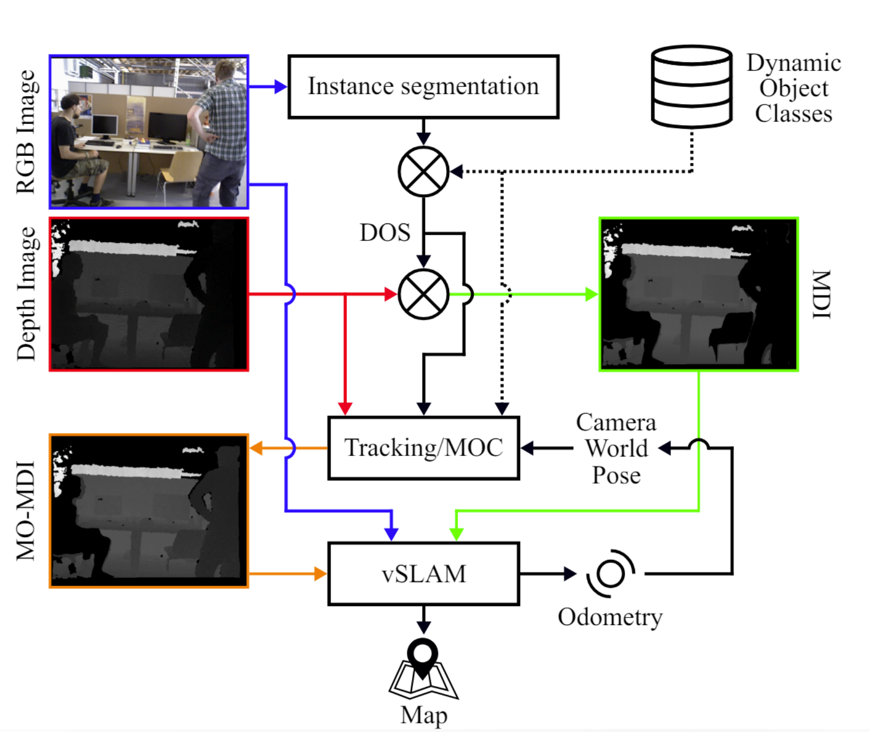
(c) Inpainted RGB images. (d) Inpainted depth image.

Dynamic object tracking and masking for visual SLAM

IROS 2020

香 港 大 學
THE UNIVERSITY OF HONG KONG

- Deep learning for semantically segment object;
- Enabling the identification, tracking and **removal of dynamic objects**, using EKF for localization and mapping;
- YOLACT (instance segmentation)+EkF+RTAB-MAP;
- Achieving **similar** localization performance compared to other state-of-the-art methods, while also providing the position of the tracked dynamic objects, a 3D map free of those dynamic objects; (Classification 1+2)



(a) fr3/sitting_static (b) fr3/walking_static (c) fr3/walking_rpy

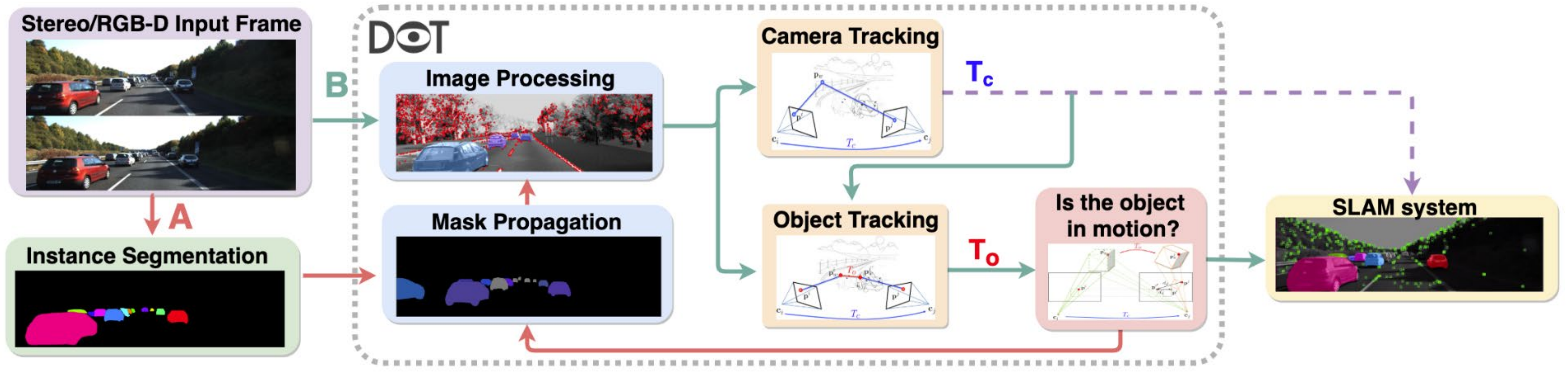
Fig. 3: RTAB-Map 3D rendered map from the TUM sequences, without (top) and with (bottom) DOTMask

DOT: Dynamic Object Tracking for Visual SLAM

ICRA 2021

香 港 大 學
THE UNIVERSITY OF HONG KONG

- DOT combines instance segmentation and multi-view geometry to generate masks for dynamic objects in order to allow SLAM systems based on rigid scene models to **avoid such image areas in their optimizations**;
- Tracking dynamic objects by minimizing the photometric reprojection error. This short-term tracking **improves the accuracy of the segmentation** with respect to other approaches;
- Classification 1;

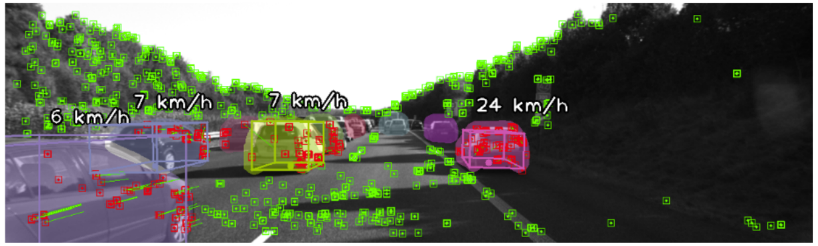


DynaSLAM II: Tightly-Coupled Multi-Object Tracking and SLAM

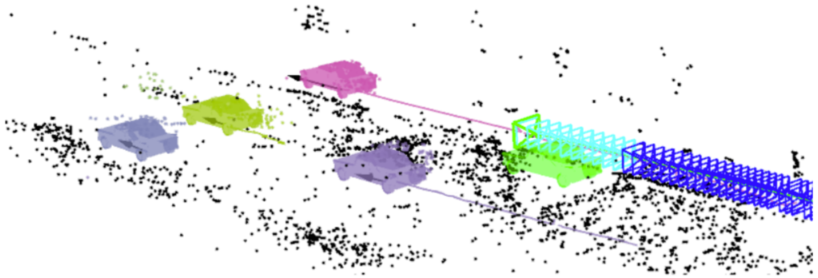
RAL 2021

香 港 大 學
THE UNIVERSITY OF HONG KONG

- Simultaneously estimates the poses of the **camera**, the **map** and the **trajectories** of the scene moving objects;
- The structure of the **static scene** and of the **dynamic objects** is **optimized jointly** with the trajectories of both the camera and the moving agents within a novel bundle adjustment;
- Classification 3;

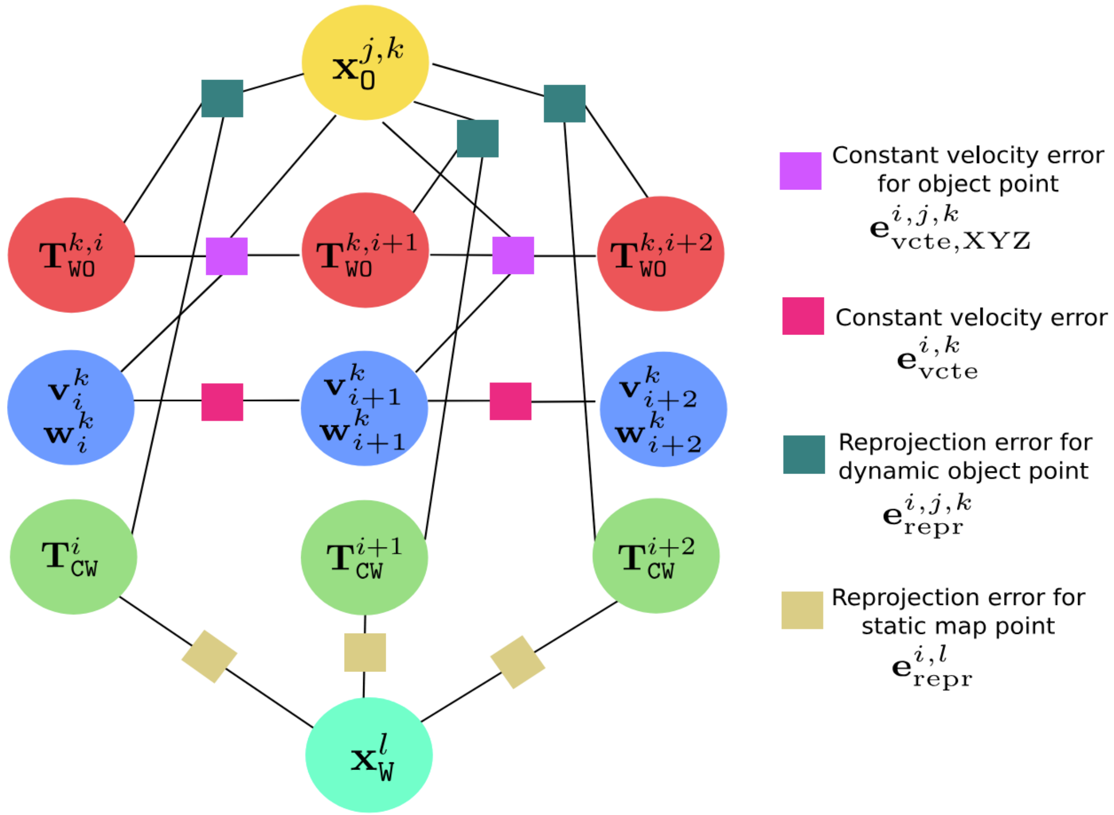


(a) The 3D bounding box and the speed of the objects are inferred in the image. Static and dynamic key points are in green and red respectively.



(b) Joint estimation of the camera ego motion (green car), the sparse static 3D map (black points) and the trajectories of the dynamic objects. The cyan key frames allow to optimize the map dynamic structure, whereas the blue ones only optimize the camera pose and the static structure.

Fig. 1: Qualitative results with the KITTI tracking dataset.



Results

seq	ORB-SLAM2			DynaSLAM			VDO-SLAM			Ours		
	ATE [m]	RPE_t[m/f]	RPE_R[°/f]	ATE [m]	RPE_t[m/f]	RPE_R[°/f]	ATE [m]	RPE_t[m/f]	RPE_R[°/f]	ATE [m]	RPE_t[m/f]	RPE_R[°/f]
0000	1.32	**0.04**	0.06	1.35	**0.04**	0.06	-	0.05	**0.05**	**1.29**	**0.04**	0.06
0001	**1.95**	**0.05**	0.04	2.42	**0.05**	0.04	-	0.12	0.04	2.31	**0.05**	0.04
0002	0.95	0.04	0.03	1.04	0.04	0.03	-	0.04	**0.02**	**0.91**	0.04	**0.02**
0003	0.74	0.07	0.04	0.78	0.07	0.04	-	0.09	0.04	**0.69**	**0.06**	0.04
0004	1.44	**0.07**	0.06	1.52	**0.07**	0.06	-	0.11	**0.05**	**1.42**	**0.07**	0.06
0005	1.23	**0.06**	0.03	**1.22**	**0.06**	0.03	-	0.10	**0.02**	1.34	**0.06**	0.03
0006	0.19	0.02	**0.04**	0.19	0.02	**0.04**	-	0.02	0.05	0.19	0.02	**0.04**
0007	**2.47**	0.05	0.07	2.69	0.05	0.07	-	-	-	3.10	0.05	0.07
0008	1.40	**0.08**	0.04	**1.29**	**0.08**	0.04	-	-	-	1.68	0.10	0.04
0009	4.00	0.06	**0.05**	**3.55**	0.06	**0.05**	-	-	-	5.02	0.06	0.06
0010	1.68	0.07	0.04	1.84	0.07	0.04	-	-	-	**1.30**	0.07	**0.03**
0011	**0.97**	0.04	0.03	1.05	0.04	0.03	-	-	-	1.03	0.04	0.03
0013	1.18	0.04	0.05	1.18	0.04	0.05	-	-	-	**1.10**	0.04	**0.04**
0014	0.13	0.03	0.08	0.13	0.03	0.08	-	-	-	**0.12**	0.03	0.08
0018	**0.89**	**0.05**	0.03	1.00	**0.05**	0.03	-	0.07	**0.02**	1.09	**0.05**	**0.02**
0019	2.31	0.05	0.03	2.35	0.05	0.03	-	-	-	**2.25**	0.05	0.03
0020	16.80	0.11	0.07	**1.10**	**0.05**	0.04	-	0.16	**0.03**	1.36	0.07	0.04
mean	2.33	0.055	0.046	**1.45**	**0.051**	0.045	-	0.084	**0.036**	1.54	0.053	0.043

TABLE I: Egomotion comparison on the KITTI tracking dataset. Results of sequences without egomotion are not shown.

seq	ORB-SLAM2			DynaSLAM			ClusterSLAM			ClusterVO			Ours		
	ATE [m]	RPE_t[m]	RPE_R[rd]	ATE [m]	RPE_t[m]	RPE_R[rd]	ATE [m]	RPE_t[m]	RPE_R[rd]	ATE [m]	RPE_t[m]	RPE_R[rd]	ATE [m]	RPE_t[m]	RPE_R[rd]
0926-0009	0.83	1.85	**0.01**	0.81	**1.80**	**0.01**	0.92	2.34	0.03	**0.79**	2.98	0.03	0.85	1.87	**0.01**
0926-0013	0.32	1.04	0.01	0.30	0.99	0.01	2.12	5.50	0.07	**0.26**	1.16	0.01	0.29	**0.93**	**0.00**
0926-0014	0.50	1.22	**0.01**	0.60	1.62	**0.01**	0.81	2.24	0.03	**0.48**	**1.04**	**0.01**	**0.48**	1.35	**0.01**
0926-0051	**0.38**	1.16	**0.00**	0.46	1.17	**0.00**	1.19	1.44	0.03	0.81	2.74	0.02	0.44	**1.14**	**0.00**
0926-0101	**2.97**	13.63	0.03	3.52	15.14	0.03	4.02	**12.43**	**0.02**	3.18	12.78	**0.02**	4.33	15.02	0.04
0929-0004	0.62	1.38	**0.01**	0.56	**1.36**	**0.01**	1.12	2.78	0.02	**0.40**	1.77	0.02	0.64	1.41	**0.01**
1003-0047	20.49	32.59	0.08	**2.87**	**5.95**	**0.02**	10.21	8.94	0.06	4.79	6.54	0.05	3.03	6.85	**0.02**
mean	3.73	7.55	0.02	**1.30**	**4.00**	**0.01**	2.91	5.10	0.04	1.53	4.14	0.02	1.44	4.08	**0.01**

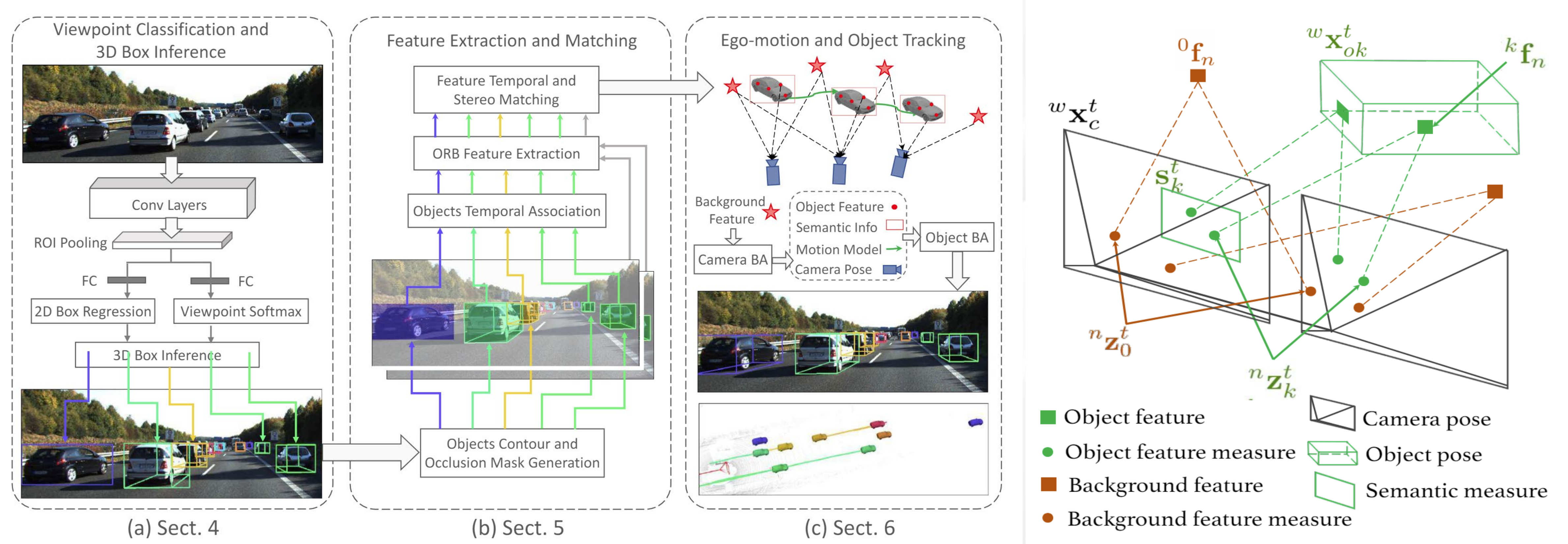
TABLE II: Egomotion comparison on the KITTI raw dataset

Stereo vision-based semantic 3D object and ego-motion tracking for autonomous driving

ECCV 2018

香 港 大 學
THE UNIVERSITY OF HONG KONG

- Using a CNN trained in an end-to-end manner to estimate the 3D pose and dimensions of cars (for lightweight purpose), which is further refined together with camera poses;
- Object-aware-aided camera pose tracking and dynamic object bundle adjustment approach;



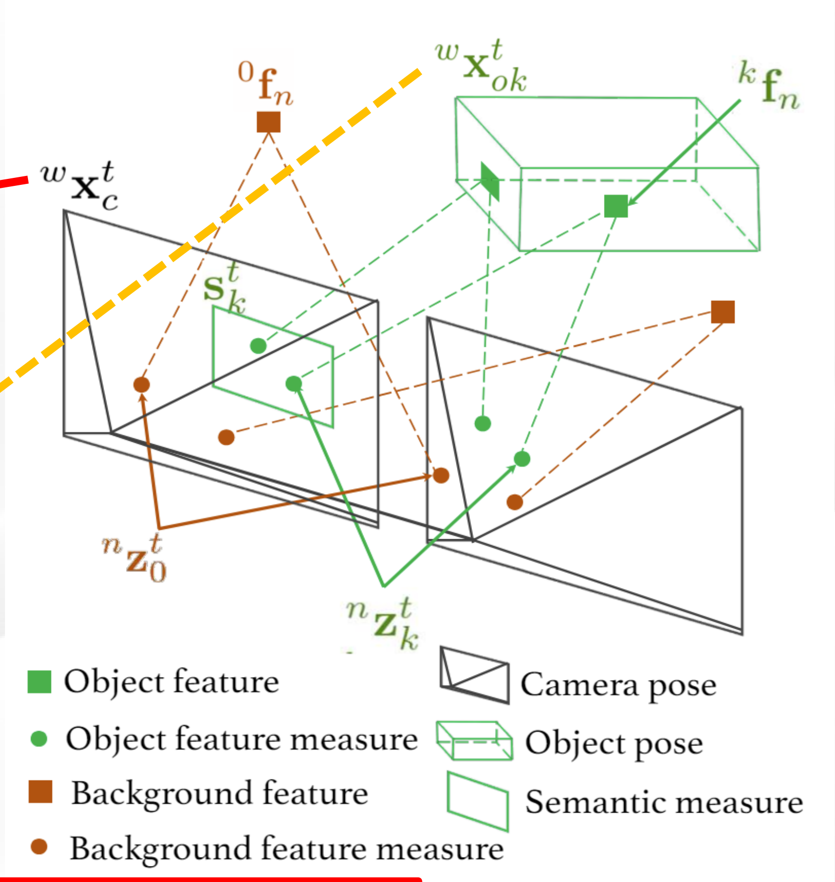
(a) Sect. 4 (b) Sect. 5 (c) Sect. 6

Stereo vision-based semantic 3D object and ego-motion tracking for autonomous driving

香 港 大 學
THE UNIVERSITY OF HONG KONG

$$
\begin{aligned}
{}^{w}\mathcal{X}_c, {}^{0}\mathbf{f} &= \arg\max_{{}^{w}\mathcal{X}_c, {}^{0}\mathbf{f}} \prod_{n=0}^{N_0}\prod_{t=0}^{T} p({}^{n}\mathbf{z}_0^t | {}^{w}\mathbf{x}_c^t, {}^{0}\mathbf{f}_n, {}^{w}\mathbf{x}_c^0) \\
&= \arg\max_{{}^{w}\mathcal{X}_c, {}^{0}\mathbf{f}} \sum_{n=0}^{N_0}\sum_{t=0}^{T} \log p({}^{n}\mathbf{z}_0^t | {}^{w}\mathbf{x}_c^t, {}^{0}\mathbf{f}_n, {}^{w}\mathbf{x}_c^0) \\
&= \arg\min_{{}^{w}\mathcal{X}_c, {}^{0}\mathbf{f}} \sum_{n=0}^{N_0}\sum_{t=0}^{T} \left\| r_{\mathcal{Z}}({}^{n}\mathbf{z}_0^t, {}^{w}\mathbf{x}_c^t, {}^{0}\mathbf{f}_n) \right\|^2_{{}^{0}\Sigma_n^t} .
\end{aligned}
$$

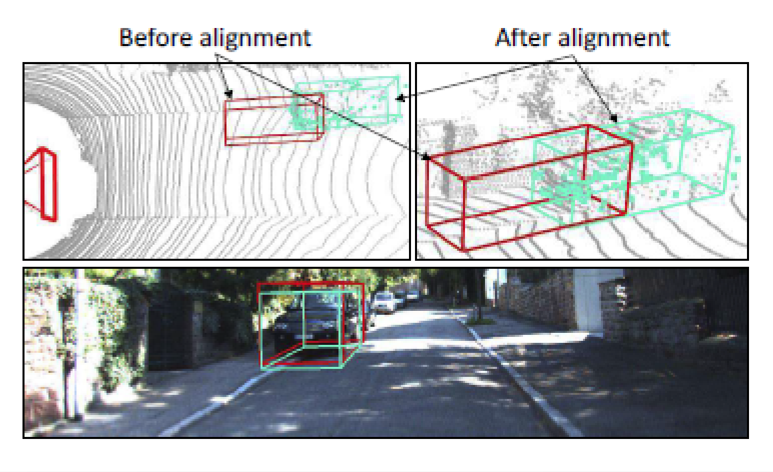
$$
\begin{aligned}
{}^{w}\mathbf{x}_{ok}, {}^{k}\mathbf{f} &= \arg\max_{{}^{w}\mathbf{x}_{ok}, {}^{k}\mathbf{f}} p({}^{w}\mathbf{x}_{ok}, {}^{k}\mathbf{f} \,|\, {}^{w}\mathbf{x}_c, \mathbf{z}_k, \mathbf{s}_k) \\
&= \arg\max_{{}^{w}\mathbf{x}_{ok}, {}^{k}\mathbf{f}} p(\mathbf{z}_k, \mathbf{s}_k | {}^{w}\mathbf{x}_c, {}^{w}\mathbf{x}_{ok}, {}^{k}\mathbf{f}) p(\mathbf{d}_k) \\
&= \arg\max_{{}^{w}\mathbf{x}_{ok}, {}^{k}\mathbf{f}} p(\mathbf{z}_k | {}^{w}\mathbf{x}_c, {}^{w}\mathbf{x}_{ok}, {}^{k}\mathbf{f}) p(\mathbf{s}_k | {}^{w}\mathbf{x}_c, {}^{w}\mathbf{x}_{ok}) p(\mathbf{d}_k) \\
&= \arg\max_{{}^{w}\mathbf{x}_{ok}, {}^{k}\mathbf{f}} \prod_{t=0}^{T}\prod_{n=0}^{N_k} p({}^{n}\mathbf{z}_k^t | {}^{w}\mathbf{x}_c^t, {}^{w}\mathbf{x}_{ok}^t, {}^{k}\mathbf{f}_n) p(\mathbf{s}_k^t | {}^{w}\mathbf{x}_c^t, {}^{w}\mathbf{x}_{ok}^t) p({}^{w}\mathbf{x}_{ok}^{t-1} | {}^{w}\mathbf{x}_{ok}^t) p(\mathbf{d}_k).
\end{aligned}
$$



$$
\begin{aligned}
{}^{w}\mathbf{x}_{ok}, {}^{k}\mathbf{f} = \arg\min_{{}^{w}\mathbf{x}_{ok}, {}^{k}\mathbf{f}} \Bigg\{ & \sum_{t=0}^{T}\sum_{n=0}^{N_k} \left\| r_{\mathcal{Z}}({}^{n}\mathbf{z}_k^t, {}^{w}\mathbf{x}_c^t, {}^{w}\mathbf{x}_{ok}^t, {}^{k}\mathbf{f}_n) \right\|^2_{{}^{k}\Sigma_n^t} + \left\| r_{\mathcal{P}}(d_k^l, \mathbf{d}_k) \right\|^2_{\Sigma^l} \\
& + \sum_{t=1}^{T} \left\| r_{\mathcal{M}}({}^{w}\mathbf{x}_{ok}^t, {}^{w}\mathbf{x}_{ok}^{t-1}) \right\|^2_{\Sigma_k^t} + \sum_{t=0}^{T} \left\| r_{\mathcal{S}}(\mathbf{s}_k^t, {}^{w}\mathbf{x}_c^t, {}^{w}\mathbf{x}_{ok}^t) \right\|^2_{\Sigma_k^t} \Bigg\}, \quad (10)
\end{aligned}
$$

Stereo vision-based semantic 3D object and ego-motion tracking for autonomous driving

香 港 大 學
THE UNIVERSITY OF HONG KONG



Estimating metric poses of dynamic objects using monocular visual-inertial fusion

IROS 2018

香 港 大 學
THE UNIVERSITY OF HONG KONG

- The whole system consists of a 2D object tracker, an **object region-based visual bundle adjustment (BA),** VINS and a correlation analysis-based metric scale estimator;
- Recovering the **metric scale** of an arbitrary dynamic object by optimizing the trajectory of the objects in the world frame, without motion assumptions (through **signal correlation analysis, rather than jointly BA,** classification 1);

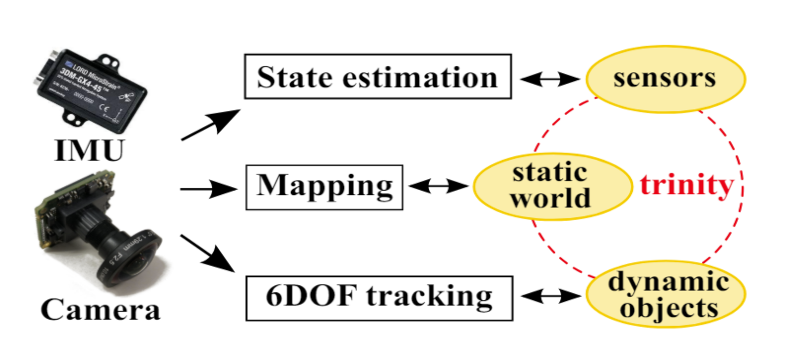
$$\min_{\mathcal{X}} \sum_{(l,j)\in\mathcal{C}} \left\| \mathbf{r}_{\mathcal{C}}(\hat{\mathbf{z}}_l^{c_j}, \mathcal{X}) \right\|_2^2,$$

accurate enough camera poses in the world frame, and up-to-scale object poses in the camera frame

Metric scale estimation

$$\begin{aligned} f(\hat{s}) &= \sum_{i;j\in x,y,z} \mathrm{Cov}^2(\hat{m}_o^i, m_c^j) \\ &= \sum_{i;j\in x,y,z} (\hat{s}\mathrm{Cov}(m_d^i, m_c^j) + \mathrm{Cov}(m_c^i, m_c^j))^2 \\ &= \left(\sum_{i;j\in x,y,z} \mathrm{Cov}^2(m_d^i, m_c^j)\right)\hat{s}^2 \\ &\quad + \left(\sum_{i;j\in x,y,z} 2\mathrm{Cov}(m_d^i, m_c^j)\mathrm{Cov}(m_c^i, m_c^j)\right)\hat{s} \\ &\quad + \sum_{i;j\in x,y,z} \mathrm{Cov}^2(m_c^i, m_c^j), \end{aligned} \tag{13}$$

[Demo for AR used]



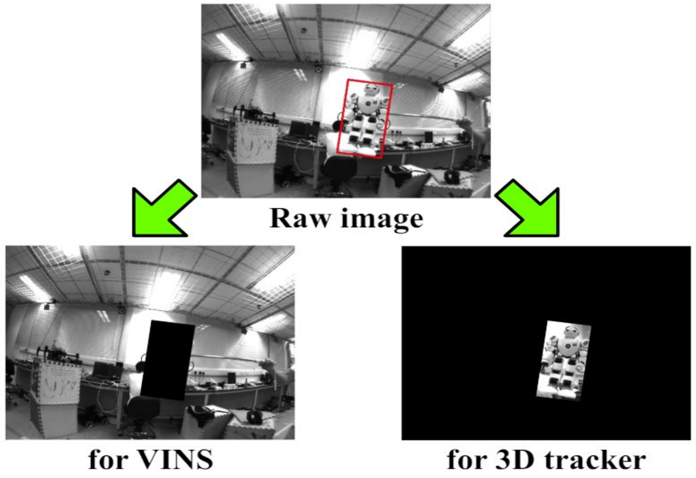
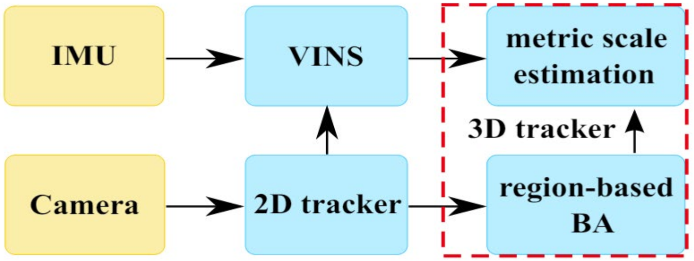


Fig. 2: 2D object tracking results with bounding boxes.



Tracking 3-D Motion of Dynamic Objects Using Monocular Visual-Inertial Sensing

TRO 2019

香 港 大 學
THE UNIVERSITY OF HONG KONG

- VINS-MONO+YOLO (2D detection)+Re-3 (2D tracker) for 6 DoF dynamic object tracking;
- Extension of their IROS 2018 paper;
- Achieving more accurate and robust scale estimation through more reliable correlation quantification method

$$\min_{\mathcal{X}} \sum_{(l,j)\in\mathcal{C}} \left\| \mathbf{r}_{\mathcal{C}}(\hat{\mathbf{z}}_l^{c_j}, \mathcal{X}) \right\|_2^2$$

$$\mathbf{r}_{\mathcal{C}}(\hat{\mathbf{z}}_l^{c_j}, \mathcal{X}) = \left(\begin{bmatrix} \hat{u}_l^{c_j} \\ \hat{v}_l^{c_j} \end{bmatrix} - \pi_c(\mathcal{P}_l^{c_j}) \right)$$

$$\mathcal{P}_l^{c_j} = \mathbf{R}_o^{c_j}\left(\mathbf{R}_{c_i}^{o} \frac{1}{\lambda_l} \pi_c^{-1}\left(\begin{bmatrix} \hat{u}_l^{c_i} \\ \hat{v}_l^{c_i} \end{bmatrix} \right) + \mathbf{p}_{c_i}^{o} \right) - \mathbf{p}_{c_j}^{o}$$

Up-to-scale Region BA

resentation. For 3-D object tracking, however, since the object is dynamic and the original point of the object coordinates represents the object location, the object coordinates must be located on the object described by the reconstructed object points.

To solve this problem, we further modify the object coordinates by estimating the initial object pose in the camera frame when the region-based BA finishes initialization. Since the absolute object orientation is unknown, we set the object orientation as an identity rotation matrix when the region-based BA is initialized, and estimate the relative 3-D rotation during tracking. As shown in Fig. 5, we move the object coordinates from the initial position to the object points along the direction denoted by the 2-D object region center, then normalize the object center depth to 1 by scaling all the object points simultaneously. In fact, the object coordinates can be anywhere around the object points. Once it is determined in the coordinate modification process, it will be continuously tracked in the BA framework.

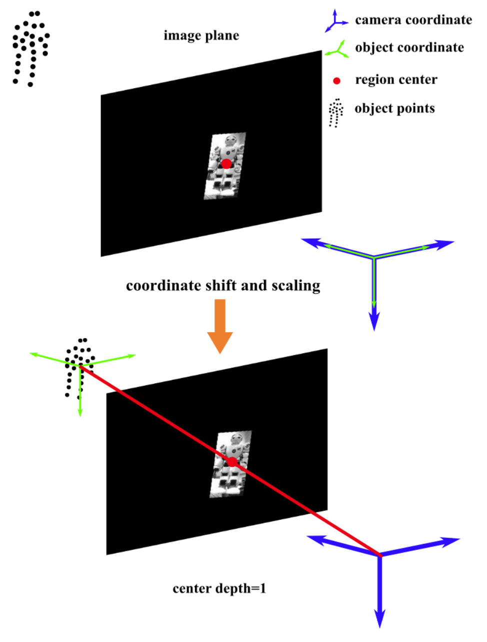
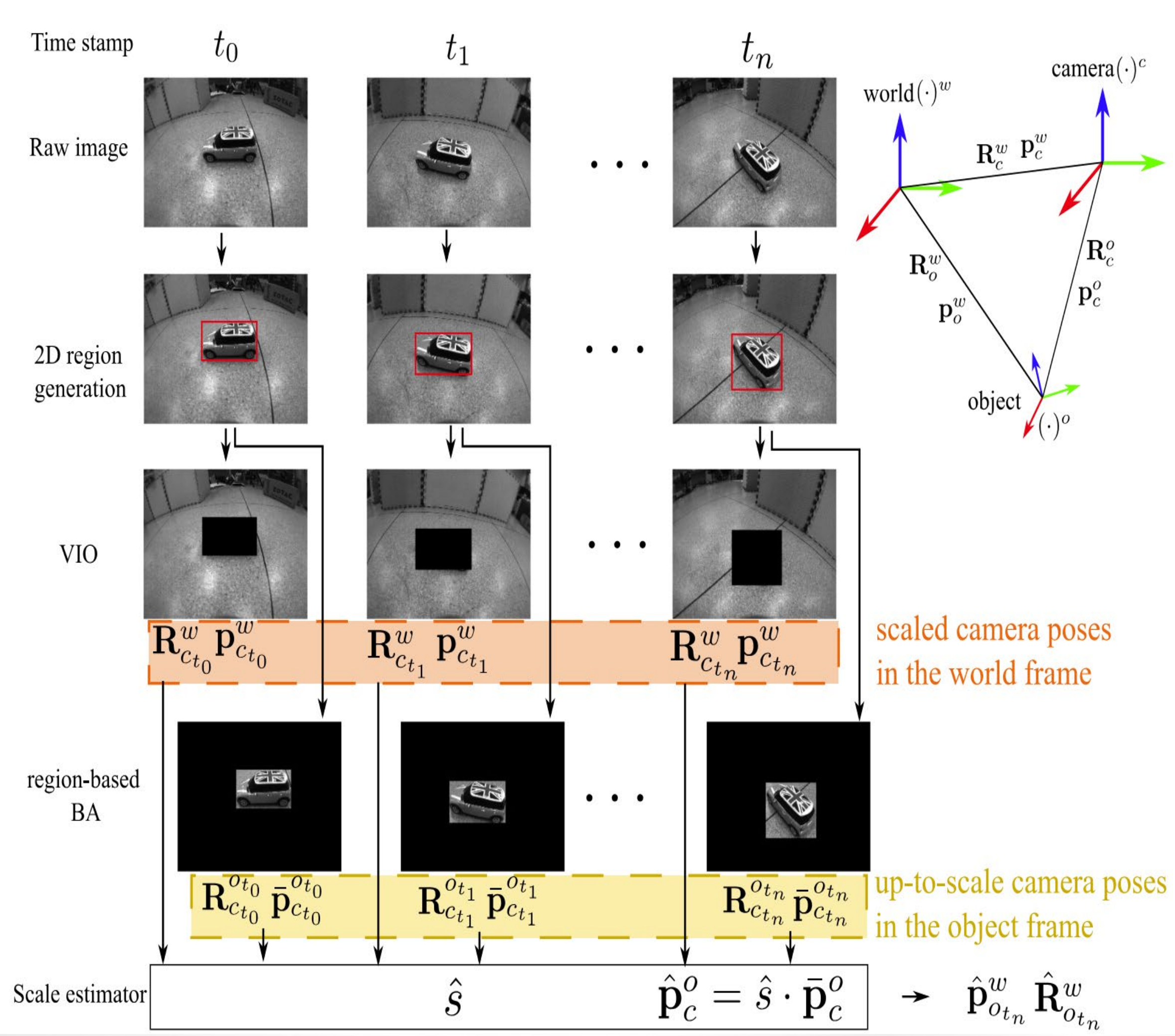
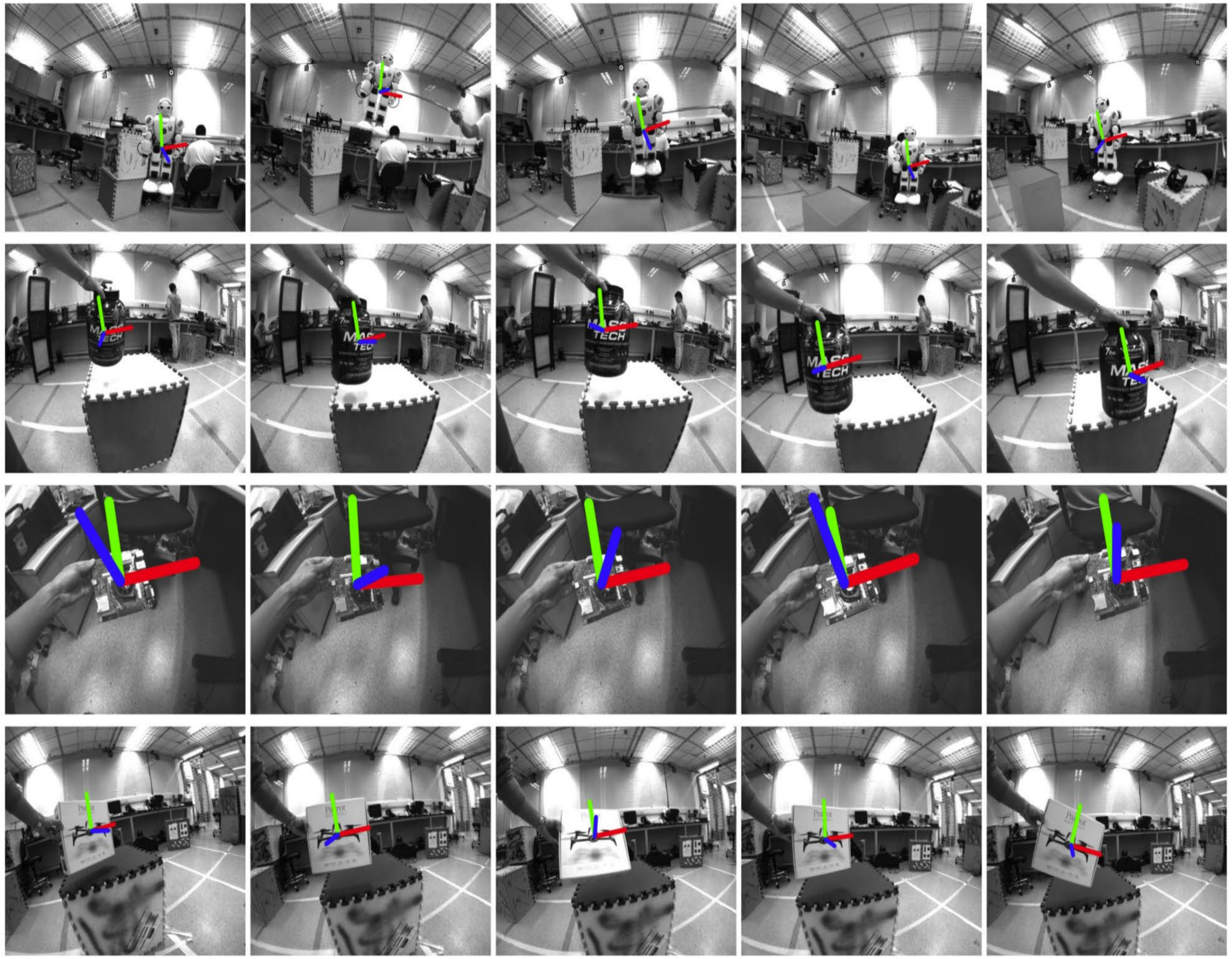


Fig. 5. Object coordinate modification process. The object coordinate will be attached with the object points and the depth of the region center will be normalized to 1 after modification.

Tracking 3-D Motion of Dynamic Objects Using Monocular Visual-Inertial Sensing for AR/VR

香 港 大 學
THE UNIVERSITY OF HONG KONG

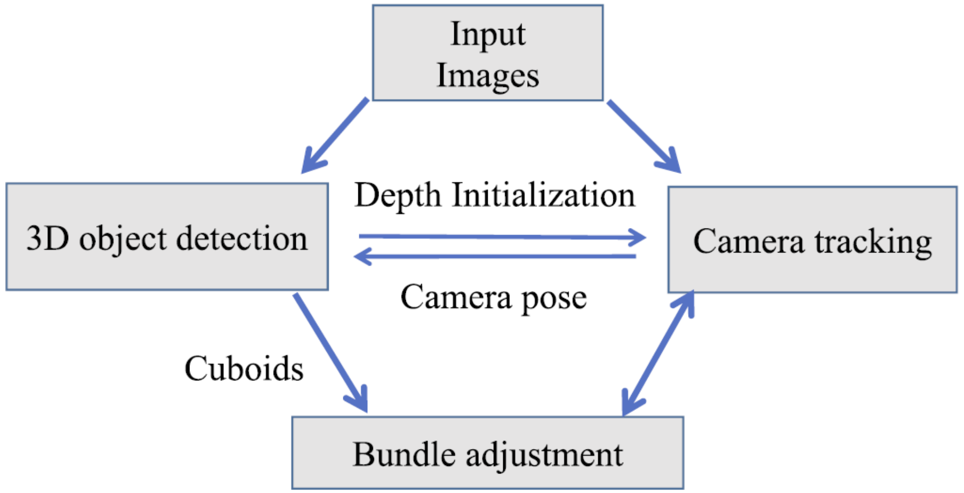


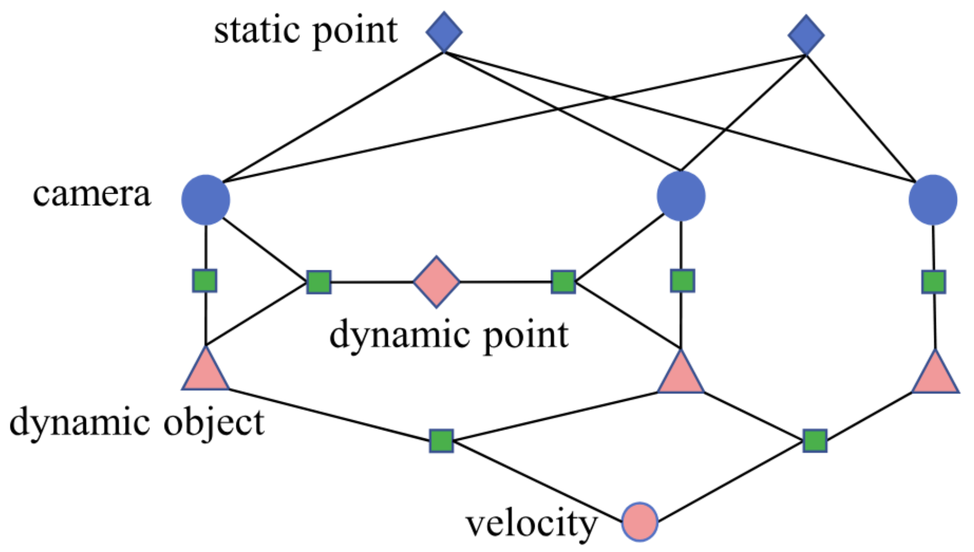


CubeSLAM: Monocular 3-D object SLAM

TRO 2019

- Single image 3-D cuboid object detection and Multiview object simultaneous localization and mapping in both static and dynamic environments, and demonstrate that the **two parts can improve each other**;
- Objects are utilized in two ways: to provide geometry and scale constraints in BA, and to provide depth initialization for points difficult to triangulate. The estimated camera poses from SLAM are also used for single-view object detection;
- Multiview bundle adjustment with new object measurements is proposed to **jointly optimize poses of cameras, objects, and points**; (Classification 3)
- Assuming that objects have a constant velocity within a hard-coded duration time interval and exploit object priors such as car sizes;





ClusterSLAM: A SLAM Backend for Simultaneous Rigid Body Clustering and Motion Estimation

香 港 大 學
THE UNIVERSITY OF HONG KONG

ICCV 2019

- Exploiting the consensus of 3D motions among the landmarks extracted from the same rigid body for **clustering** and estimating static and dynamic objects in a unified manner;
- Building a noise-aware motion affinity matrix upon landmarks, and uses agglomerative clustering for **distinguishing** those rigid bodies;
- A decoupled factor **graph optimization** for revising their shape and trajectory;

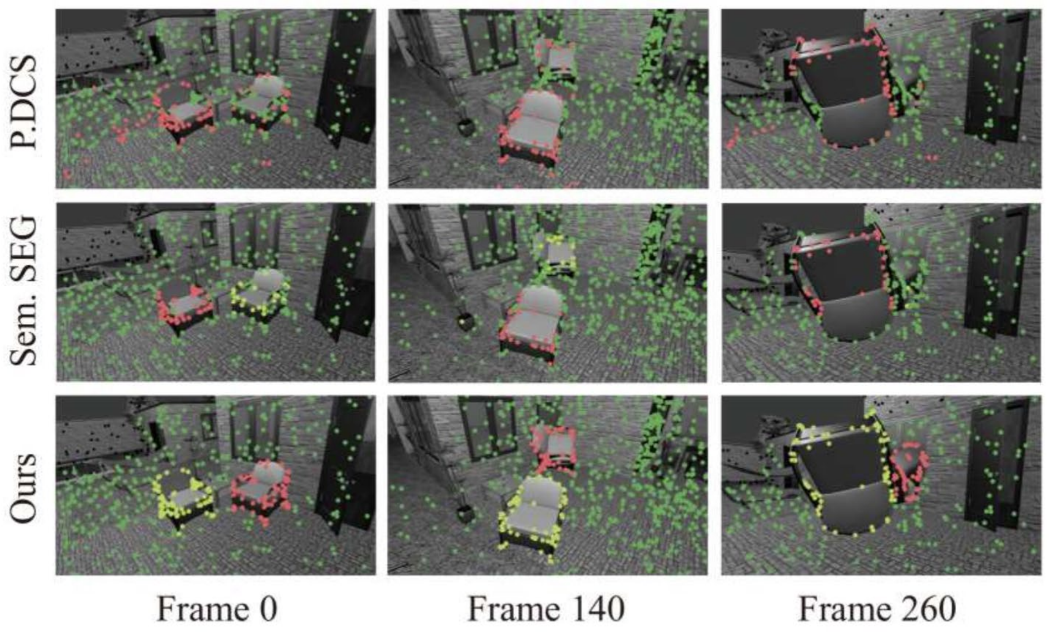
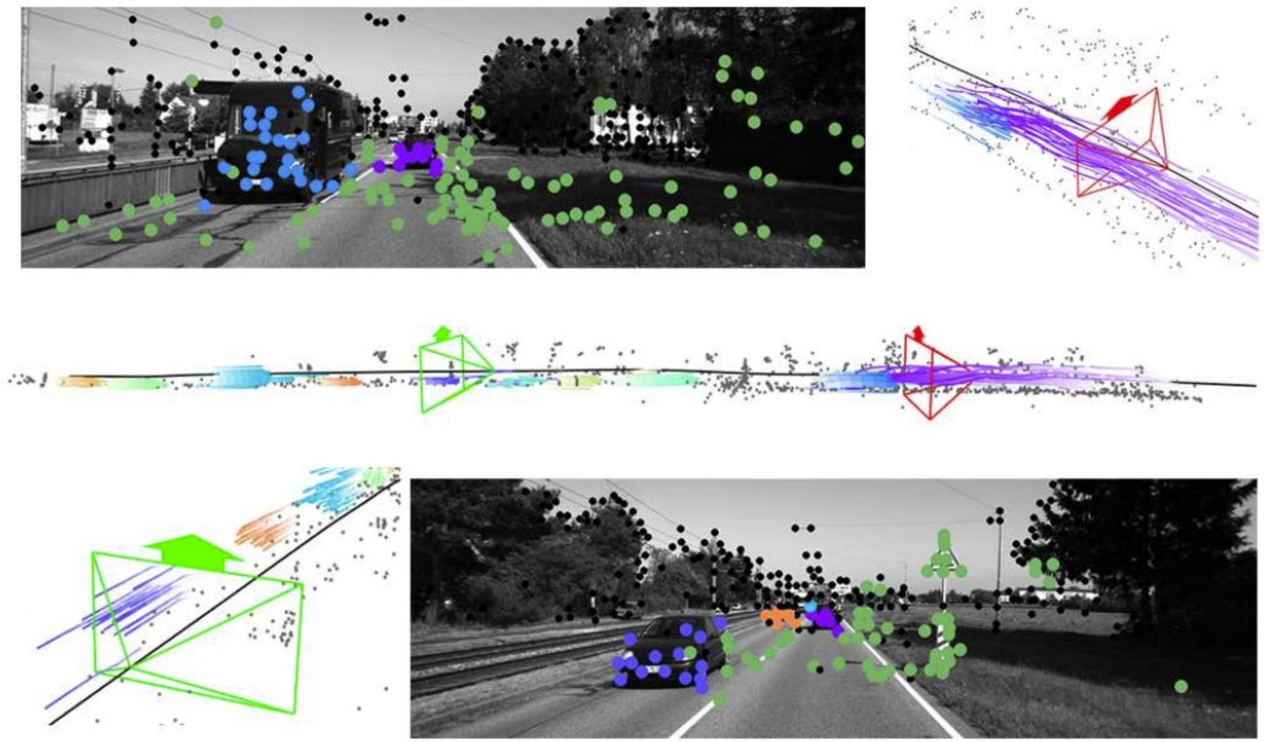


Figure 1. Visual comparisons on a SUNCG sequence. Landmarks are colorized by their index of cluster.

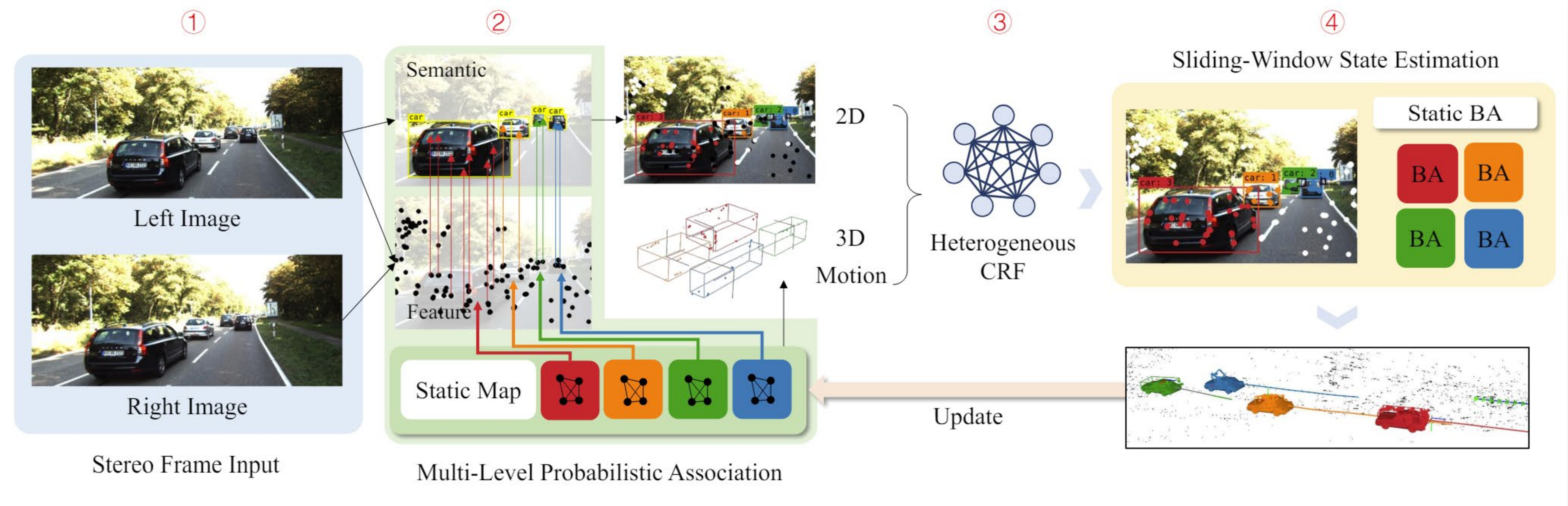


ClusterVO: Clustering Moving Instances and Estimating Visual Odometry for Self and Surroundings

香 港 大 學
THE UNIVERSITY OF HONG KONG

CVPR 2020

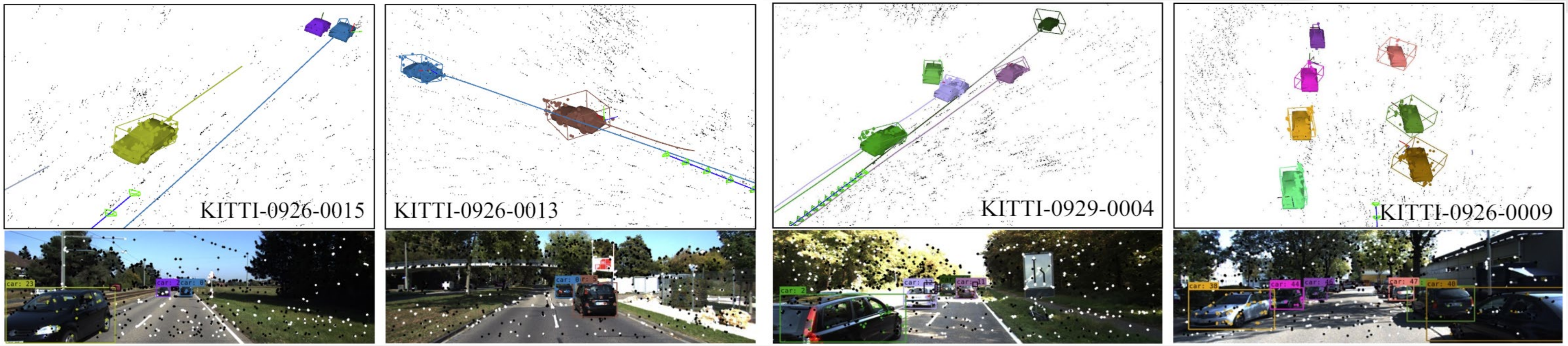
- Improvement of ClusterSLAM to include more scenarios and online; (Classification 3)
- No geometric or shape priors;
- Simultaneously optimizes the poses of camera and multiple moving objects, regarded as clusters of point landmarks, in a unified manner, achieving a competitive frame-rate with promising tracking and segmentation ability;
- Solely based on **sparse landmarks and 2D detections**, lightweight enough to track both low-level features and high-level detections over time in the 3D space;



Results

Table 2. Camera ego-motion comparison with state-of-the-art systems on KITTI raw dataset. The unit of ATE and T.RPE is meters and the unit for R.RPE is radians.

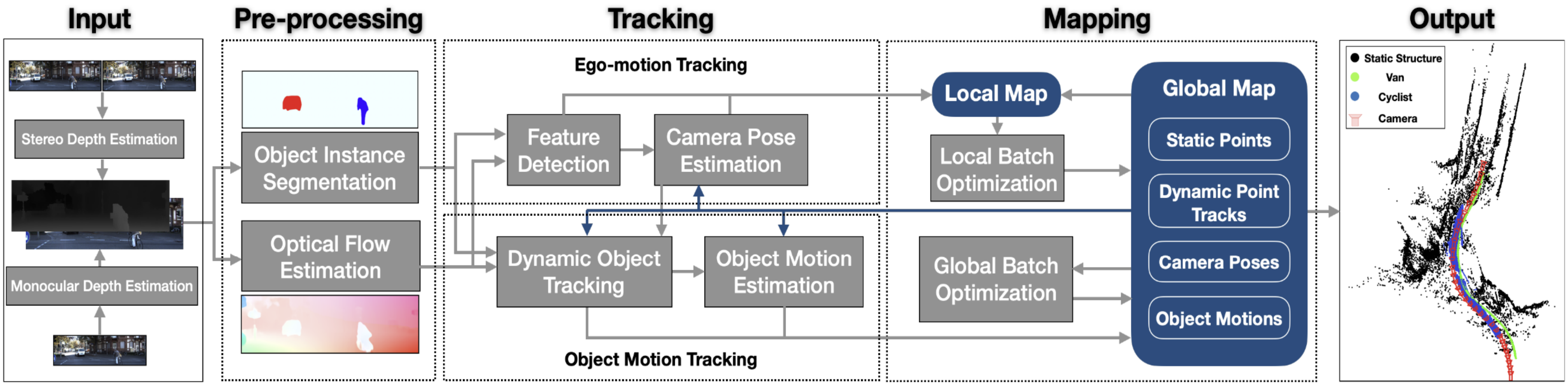
Sequence	ORB-SLAM2 [28]			DynSLAM [2]			Li *et al.* [24]	ClusterSLAM [15]			ClusterVO		
	ATE	R.RPE	T.RPE	ATE	R.RPE	T.RPE	ATE	ATE	R.RPE	T.RPE	ATE	R.RPE	T.RPE
0926-0009	0.91	**0.01**	**1.89**	7.51	0.06	2.17	1.14	0.92	0.03	2.34	**0.79**	0.03	2.98
0926-0013	0.30	**0.01**	**0.94**	1.97	0.04	1.41	0.35	2.12	0.07	5.50	**0.26**	**0.01**	1.16
0926-0014	0.56	**0.01**	1.15	5.98	0.09	2.73	0.51	0.81	0.03	2.24	**0.48**	**0.01**	**1.04**
0926-0051	**0.37**	**0.00**	**1.10**	10.95	0.10	1.65	0.76	1.19	0.03	1.44	0.81	0.02	2.74
0926-0101	3.42	0.03	14.27	10.24	0.13	**12.29**	5.30	4.02	**0.02**	12.43	**3.18**	**0.02**	12.78
0929-0004	0.44	**0.01**	**1.22**	2.59	0.02	2.03	**0.40**	1.12	0.02	2.78	**0.40**	0.02	1.77
1003-0047	18.87	0.05	28.32	9.31	0.05	6.58	**1.03**	10.21	0.06	8.94	4.79	0.05	**6.54**



VDO-SLAM: A Visual Dynamic Object-aware SLAM System

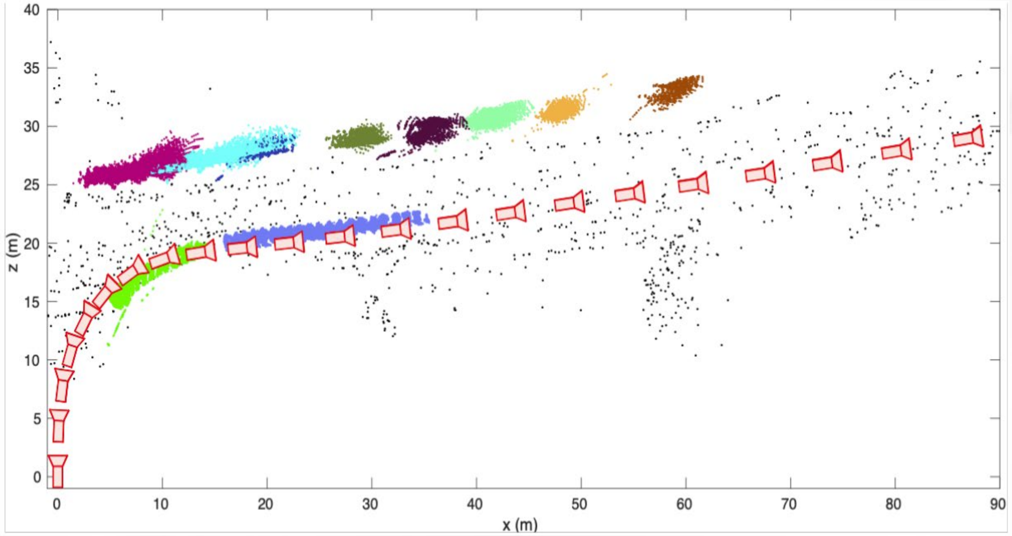
香 港 大 學
THE UNIVERSITY OF HONG KONG

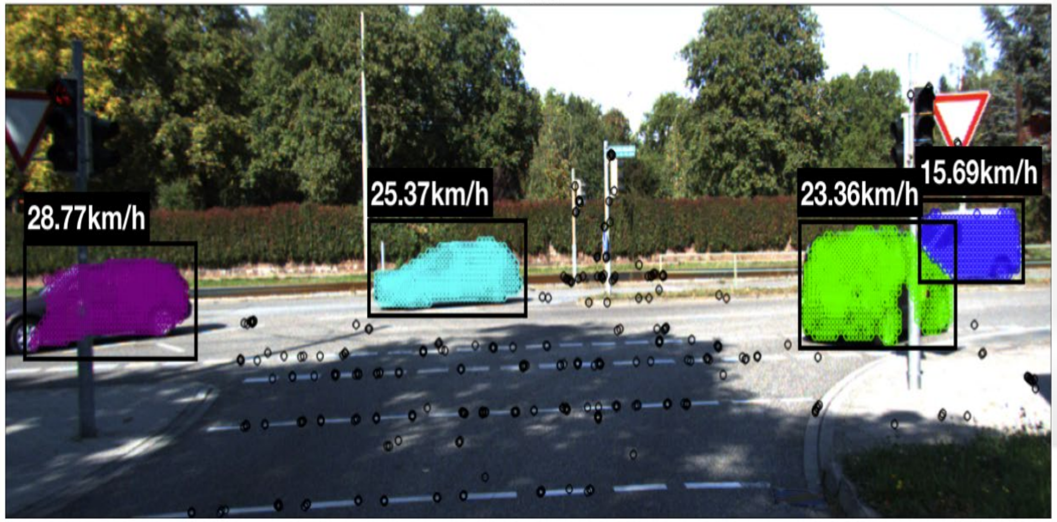
- Enable accurate motion estimation and tracking of dynamic rigid objects in the scene without any prior knowledge of the objects' shape or geometric models;
- Extract linear velocity estimates from objects (functionality for navigation and obstacle avoidance);
- Contribution points: model the dynamic scenes, robust tracking moving objects with semantic information; full system design;

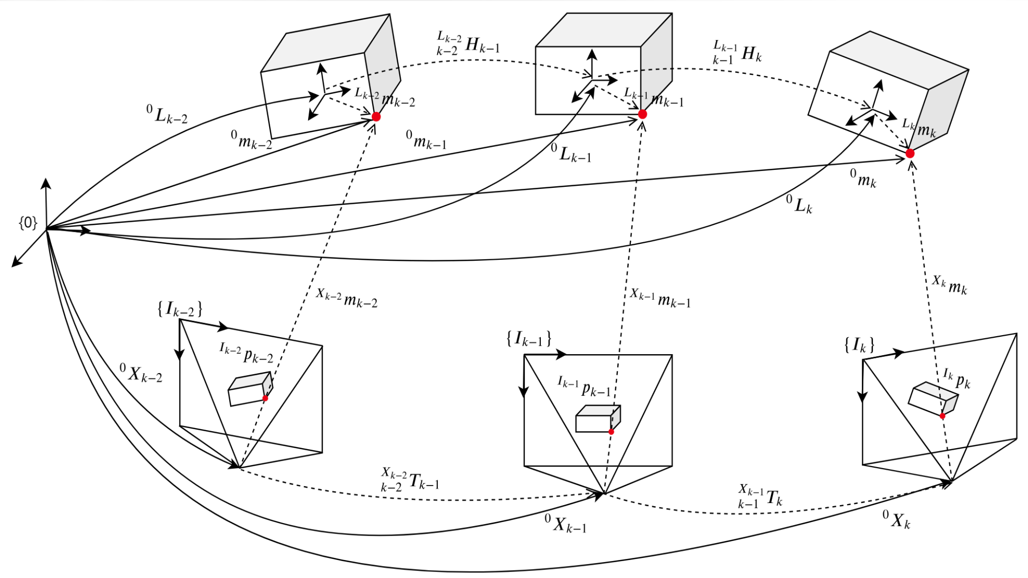


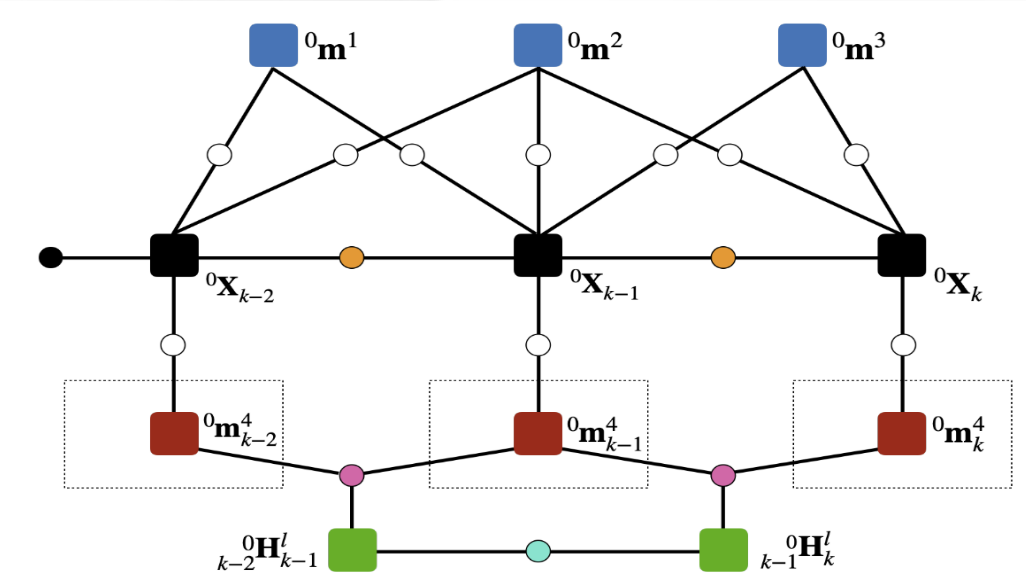
VDO-SLAM: A Visual Dynamic Object-aware SLAM System

香 港 大 學
THE UNIVERSITY OF HONG KONG







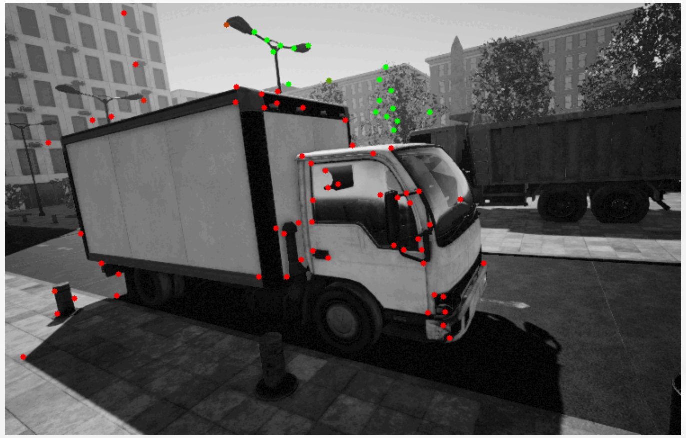


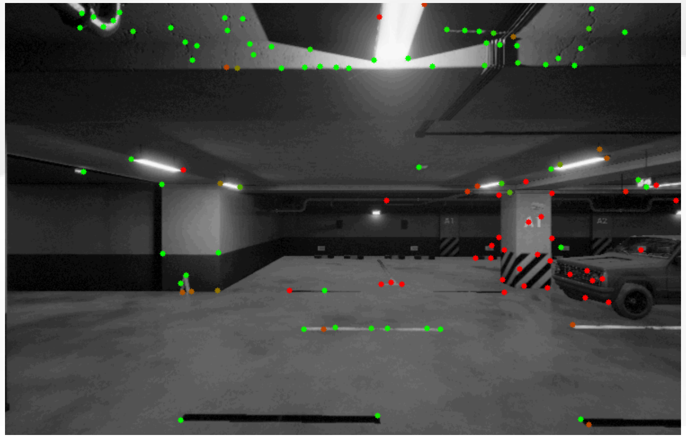
DynaVINS: A Visual-Inertial SLAM for Dynamic Environments

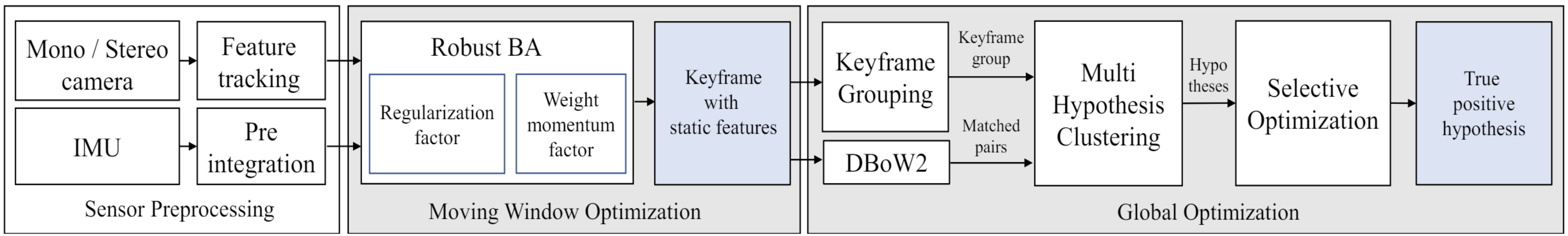
RAL 2022

香 港 大 學
THE UNIVERSITY OF HONG KONG

- A robust BA is applied to **discard tracked features from dynamic objects** and only the features from static objects will be remain (**motion prior**);
- In addition, **temporarily static objects**, which are static during observation but move when they are out of sight, **trigger false positive loop closings**;
- Keyframe grouping and **multi-hypothesis-based constraints grouping methods** are proposed to reduce the effect of temporarily static objects in the loop closing;

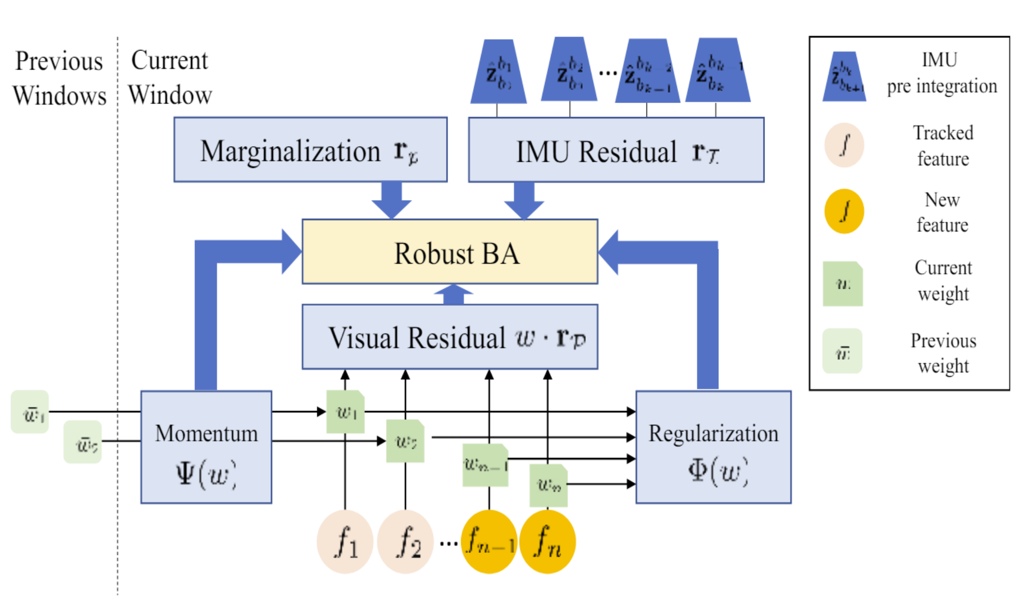


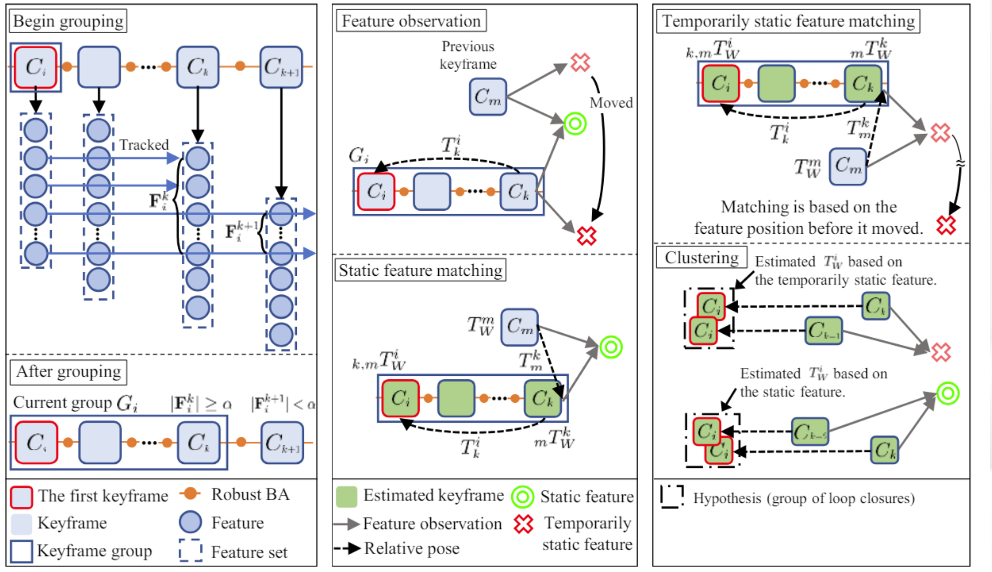




Discarding the features from the dynamic objects that deviate significantly from the motion prior

香 港 大 學
THE UNIVERSITY OF HONG KONG





(a) (b) (c)

TABLE II. Comparison with state-of-the-art methods (RMSE of ATE in [m]). *: Failure case (diverged), −M−I: Mono-inertial mode, −S: Stereo mode, −S−I: Stereo-inertial mode, SC: Switchable Constraints [16] Parameters for DynaVINS in VIODE: $\lambda_w = 1.0, \lambda_m = 0.2$ and in our dataset: $\lambda_w = 1.0, \lambda_m = 1.0, \lambda_l = 1.0$.

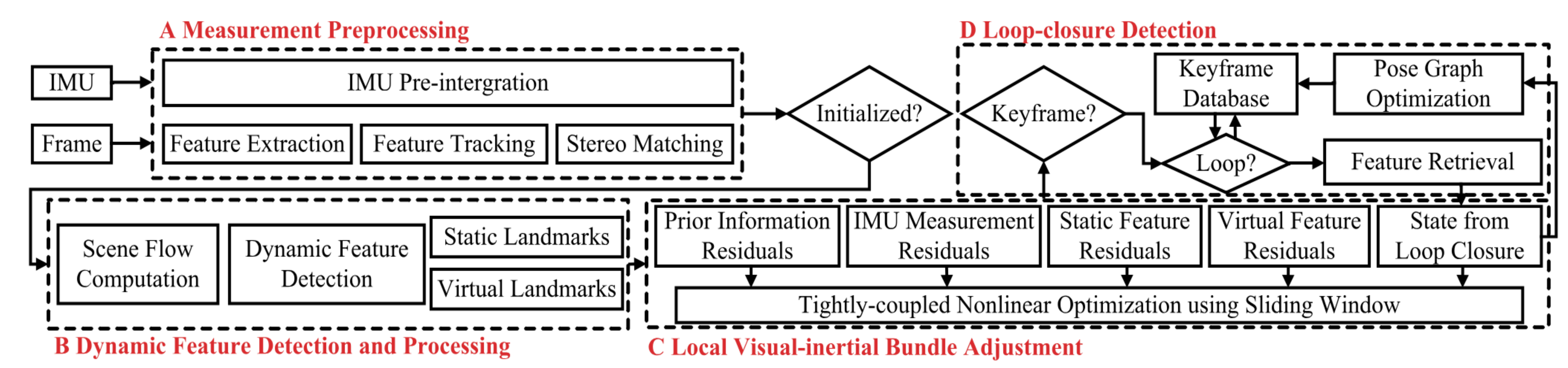
Method	VIODE [8]												Our dataset			
	city_day				city_night				parking_lot				Static	Dynamic follow	Temporal static	E-shape
	none	low	mid	high	none	low	mid	high	none	low	mid	high				
ORB-SLAM3−M−I	1.940	0.857	4.486	*	*	*	*	*	0.147	0.175	0.145	0.194	0.379	1.374	0.775	*
VINS-Fusion−M−I	**0.210**	0.182	0.560	0.510	0.328	0.371	0.457	0.464	0.102	0.138	0.707	1.135	0.080	0.463	0.414	0.727
VINS-Fusion−M−I with SC															0.091	0.736
DynaVINS−M−I	0.224	**0.167**	**0.154**	**0.364**	**0.189**	**0.181**	**0.184**	**0.256**	**0.097**	**0.120**	**0.118**	**0.149**	**0.048**	**0.141**	**0.051**	**0.107**
DynaSLAM−S	1.621	1.426	1.638	*	3.333	3.314	3.074	3.865	0.108	0.170	*	*	0.081	1.017	0.467	0.937
ORB-SLAM3−S−I	0.302	0.419	0.217	*	0.709	0.895	1.693	3.006	0.148	0.067	*	*	0.069	*	0.067	0.476
VINS-Fusion−S−I	**0.150**	0.203	0.234	0.373	0.317	0.507	0.494	0.828	0.121	0.121	0.212	0.278	**0.029**	0.383	0.229	0.711
VINS-Fusion−S−I with SC															0.034	0.686
DynaVINS−S−I	0.171	**0.178**	**0.091**	**0.148**	**0.213**	**0.182**	**0.201**	**0.198**	**0.049**	**0.042**	**0.064**	**0.042**	0.032	**0.038**	**0.025**	**0.029**

Dynam-SLAM: An Accurate, Robust Stereo Visual-Inertial SLAM Method in Dynamic Environments

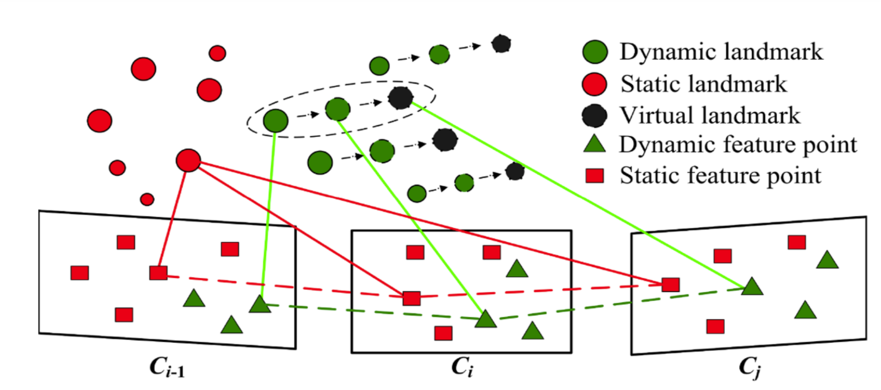
香 港 大 學
THE UNIVERSITY OF HONG KONG

TRO 2023

- Loosely coupling the stereo scene flow with IMU for dynamic feature detection;
- Tightly coupling the dynamic and static features with the IMU measurements for nonlinear optimization;
- Proposing a concept of virtual landmarks related to dynamic features and construct a nonlinear optimization model;
- Classification 3;



Dynam-SLAM: An Accurate, Robust Stereo Visual-Inertial SLAM Method in Dynamic Environments



Due to the short time interval between two adjacent frames, it can be approximated that the dynamic landmark moves at a uniform speed in the world frame. Therefore, the movement change of the dynamic landmark between each adjacent frame can be considered the same, which can be given by

$$\Delta \mathbf{M}_i^{i-1} = \mathbf{P}_n^{w_i} - \mathbf{P}_n^{w_{i-1}} \tag{33}$$

where $\mathbf{P}_n^{w_{i-1}}$ and $\mathbf{P}_n^{w_i}$, respectively, represent the world locations of the nth landmarks observed in frame $i-1$ and frame i, which are expressed as

$$\begin{aligned} \mathbf{P}_n^{w_{i-1}} &= \mathbf{R}_w^{b_{i-1}}\left(\mathbf{R}_b^c \mathbf{P}_n^{c_{i-1}} + \boldsymbol{\alpha}_b^c\right) + \mathbf{p}_w^{b_{i-1}} \\ \mathbf{P}_n^{w_i} &= \mathbf{R}_w^{b_i}\left(\mathbf{R}_b^c \mathbf{P}_n^{c_i} + \boldsymbol{\alpha}_b^c\right) + \mathbf{p}_w^{b_i}. \end{aligned} \tag{34}$$

The predicted world location of the virtual landmark in frame j is given by

$$\mathbf{P}_n^{w_j} = \mathbf{P}_n^{w_i} + \Delta \mathbf{M}_i^{i-1}. \tag{35}$$

The residual for the dynamic feature observation $\mathbf{p}_n^{\mathbf{I}_{j,l}}$ in the image frame $\mathbf{I}_{j,l}$ corresponding to the virtual landmark $\mathbf{P}_n^{w_i}$ is defined as

$$\mathbf{r}_{\mathcal{D}}\left(\hat{\mathbf{z}}_k^{c_j}, \chi\right) = \pi_c\left(\mathbf{R}_c^b\left(\mathbf{R}_{b_i}^w \mathbf{P}_n^{w_i} + \boldsymbol{\alpha}_{b_i}^w\right) + \boldsymbol{\alpha}_c^b\right) - \mathbf{p}_n^{\mathbf{I}_{j,l}} \tag{37}$$

$$\begin{aligned} \min_{\chi} \Bigg\{ & \underbrace{\left\|\mathbf{r}_p - \mathbf{H}_p \chi\right\|^2}_{\mathbf{R}_p} + \sum_{k \in \mathcal{B}} \underbrace{\left\|\mathbf{r}_{\mathcal{B}}\left(\hat{\mathbf{z}}_{b_{k+1}}^{b_k}, \chi\right)\right\|_{\Sigma_{\mathbf{X}_{k,k+1}}}^2}_{\mathbf{R}_{\mathcal{B}}} \\ & + \sum_{(l,j) \in \mathcal{S}} \underbrace{\rho\left(\left\|\mathbf{r}_{\mathcal{S}}\left(\hat{\mathbf{z}}_k^{c_j}, \chi\right)\right\|_{\mathbf{P}_l^{c_j}}^2\right)}_{\mathbf{R}_{\mathcal{S}}} \\ & \boxed{+ \sum_{(m,j) \in \mathcal{D}} \underbrace{\rho\left(\left\|\mathbf{r}_{\mathcal{D}}\left(\hat{\mathbf{z}}_k^{c_j}, \chi\right)\right\|_{\mathbf{P}_m^{c_j}}^2\right)}_{\mathbf{R}_{\mathcal{D}}} \Bigg\}} \end{aligned} \tag{38}$$

Results

TABLE II
RMSE ATE (M) COMPARISON IN THE STATIC EUROC DATASET FOR SEVERAL DIFFERENT METHODS[1]

Dataset	Our Method		Stereo-Inertial VISLAM		
	Dynam w/o DFT[2]	Dynam[2]	VINS-SI[3]	Kimera[4]	ORB3-SI[5]
MH_01	0.078	0.078	0.240	0.080	**0.037**
MH_02	0.066	0.067	0.180	0.090	**0.031**
MH_03	0.061	0.061	0.230	0.110	**0.026**
MH_04	0.077	0.077	0.390	0.150	**0.059**
MH_05	0.095	0.096	0.190	0.240	**0.086**
V1_01	0.052	0.052	0.100	0.050	**0.037**
V1_02	0.038	0.040	0.100	0.110	**0.014**
V1_03	0.042	0.051	0.110	0.120	**0.023**
V2_01	0.048	0.048	0.120	0.070	**0.037**
V2_02	0.055	0.056	0.100	0.100	**0.014**
V2_03	0.080	0.085	0.270	0.190	**0.029**
Avg	0.061	0.063	0.185	0.119	**0.036**

[1] The bold text indicates the best results among all evaluated methods. Abbreviations: VINS-SI—VINS-Fusion with stereo-inertial configuration, ORB3-SI—ORB-SLAM3 with stereo-inertial configuration.
[2] Errors obtained by ourselves, running the source code ten times and then taking the median.
[3,4,5] Errors reported at [14], [15], and [55], respectively.

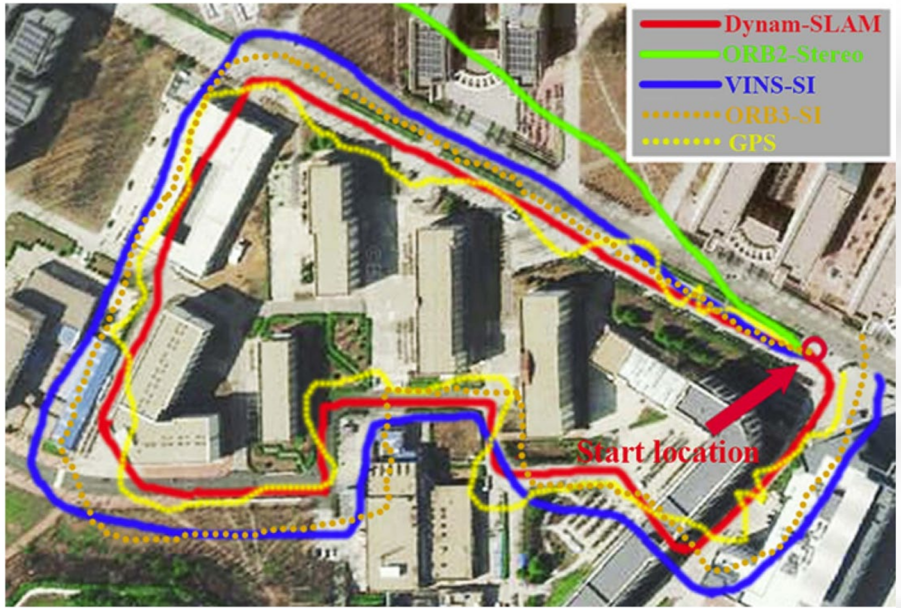


TABLE III
RMSE ATE (M) COMPARISON FOR MULTIPLE EVALUATED METHODS IN SELF-COLLECTED DYNAMIC BENCHMARK DATASETS OF VARYING DIFFICULTY[1]

Dataset	Our Method		Stereo VSLAM			Stereo-Inertial VISLAM		
	Dynam	Dynam w/o DFT	ORB2-Stereo	VINS-Stereo	ORB3-Stereo	VINS-SI	ORB3-SI	Kimera
SFL_easy	**0.048**	0.203	0.317	0.546	0.330	0.221	0.297	0.302
SSL_easy	**0.059**	0.253	0.323	1.732	0.360	0.362	0.321	0.347
SFH_easy	**0.055**	0.343	0.322	0.540	0.308	0.311	0.254	0.226
SSH_med	**0.078**	0.598	0.422	4.367	0.332	0.373	0.412	0.487
LFL_med	**0.081**	0.619	0.801	13.278	0.764	0.526	0.447	0.573
LSL_med	**0.107**	0.825	0.982	-	1.396	1.075	0.484	0.768
LFH_hard	**0.087**	0.514	-	-	-	0.988	0.897	0.961
LSH_hard	**0.118**	2.162	-	-	-	2.371	0.903	1.544
Avg[2]	**0.079**	0.690(↓89%)	0.528*	4.093*	0.582*	0.778(↓90%)	0.502(↓84%)	0.651(↓88%)

[1] The values we report are the median after ten executions. A dash indicates that the SLAM system fails to estimate the full trajectory in this dataset. The bold text indicates the best results among all evaluated methods. Abbreviations: ORB2-Stereo—ORB-SLAM2 with stereo configuration, VINS-Stereo—VINS-Fusion with stereo configuration, ORB3-Stereo—ORB-SLAM3 with stereo configuration, VINS-SI—VINS-Fusion with stereo-inertial configuration, ORB3-SI—ORB-SLAM3 with stereo-inertial configuration, SSL (small proportion, slow motion, and low frequency), · · ·, LFH (large proportion, fast motion, and high frequency).
[2] The average error of successful datasets for a given SLAM system. Systems that do not complete all datasets are denoted by*.

TABLE IV
RMSE RPE (TRANS. IN M/S, ROT. IN °/S) COMPARISON FOR MULTIPLE EVALUATED METHODS IN SELF-COLLECTED DYNAMIC BENCHMARK DATASETS OF VARYING DIFFICULTY[1]

Dataset	Our Method		Stereo VSLAM			Stereo-Inertial VISLAM		
	Dynam	Dynam w/o DFT	ORB2-Stereo	VINS-Stereo	ORB3-Stereo	VINS-SI	ORB3-SI	Kimera
	Trans. Rot.	Trans. Rot.	Trans. Rot.	Trans. Rot.	Trans. Rot.	Trans. Rot.	Trans. Rot.	Trans. Rot.
SFL_easy	**0.005** **0.013**	0.152 0.249	0.123 3.180	0.118 1.109	0.119 3.121	0.148 0.249	0.124 3.183	0.144 0.208
SSL_easy	**0.008** **0.131**	0.157 0.270	0.132 3.004	0.410 4.265	0.136 3.002	0.185 0.266	0.128 2.989	0.176 0.305
SFH_easy	**0.010** **0.099**	0.156 0.318	0.114 2.719	0.120 3.099	0.116 2.709	0.156 0.234	0.114 2.698	0.167 0.303
SSH_med	**0.009** **0.124**	0.136 2.251	0.126 2.474	0.781 3.647	0.127 2.590	0.183 0.388	0.126 2.453	0.186 0.553
LFL_med	**0.008** **0.041**	0.167 0.376	0.162 3.081	3.451 4.011	0.144 3.072	0.163 0.271	0.135 2.633	0.147 0.438
LSL_med	**0.005** **0.018**	0.256 0.512	0.224 4.736	-	0.182 3.329	0.237 0.389	0.145 3.182	0.187 0.368
LFH_hard	**0.021** **0.118**	0.232 0.391	-	-	-	0.224 0.285	0.139 3.215	0.197 0.368
LSH_hard	**0.021** **0.242**	0.361 0.605	-	-	-	0.298 0.448	0.149 3.240	0.174 0.598
Avg[2]	**0.011** **0.098**	0.202(↓95%) 0.622(↓84%)	0.147* 3.199*	0.976* 3.226*	0.137* 2.971*	0.199(↓94%) 0.316(↓69%)	0.133(↓92%) 2.949(↓97%)	0.172(↓94%) 0.393(↓75%)

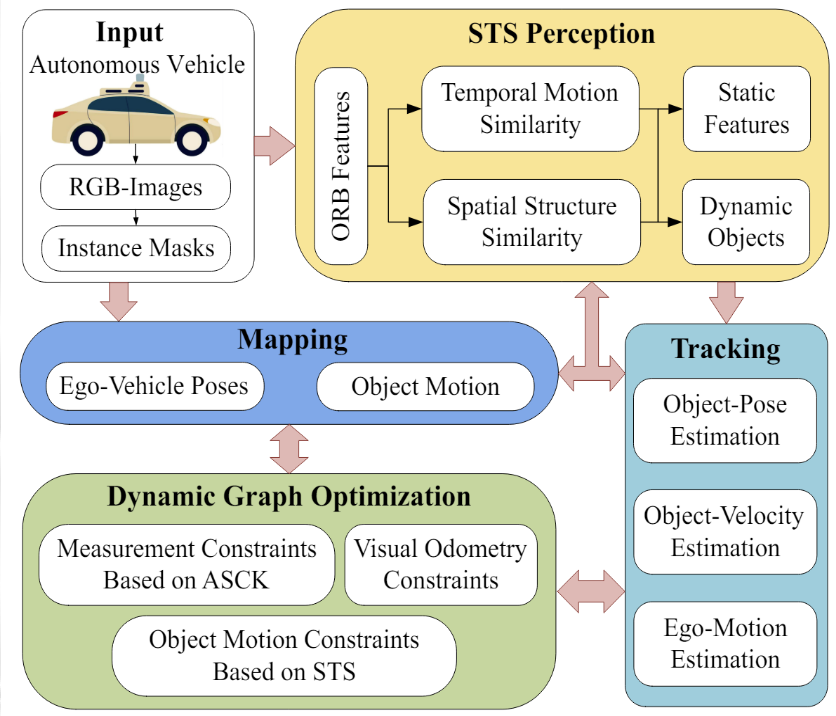
[1] The values we report are the median after ten executions. A dash indicates that the SLAM system fails to estimate the full trajectory in this dataset. The bold text indicates the best results among all evaluated methods. The abbreviations in the table are the same as in Tab. III.
[2] The average error of successful datasets for a given SLAM system. Systems that do not complete all datasets are denoted by *.

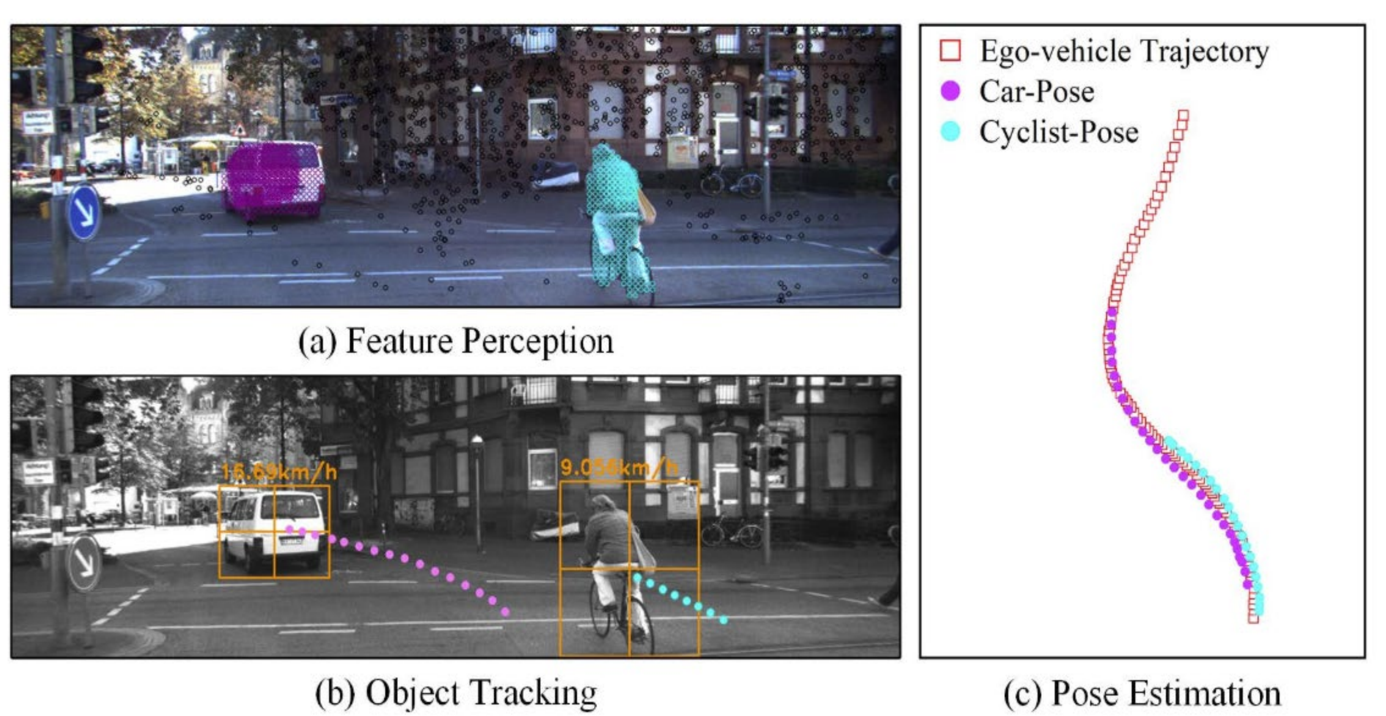
STS-SLAM: Joint Visual SLAM and Multi-Object Tracking Based on Spatio-Temporal Similarity

香 港 大 學
THE UNIVERSITY OF HONG KONG

TIV 2024

- Synchronously optimize the motion of the ego-vehicle and objects, and estimate object velocity without any prior information about the object;
- All static features are employed for ego-vehicle localization, and features with high spatio-temporal similarity on dynamic objects are robustly tracked;
- The STS-SLAM problem is modeled as a dynamic constraint factor graph for joint optimization of dynamic and static structures;(Classification 3)





(a) Feature Perception

(b) Object Tracking

(c) Pose Estimation

Optimizing the ego-vehicle poses, object poses, and STS-based scaling factor

$$T_c^*, S_i^*, S^* = \underset{T,S,S}{\operatorname{argmin}} \sum_{k,j} \|e_{cam}\|^2_{\Sigma_{kj}} + \sum_{k,i,j} \|e_{sim}\|^2_{\Sigma_{kij}} + \sum_{k,j,l} \|e_{mea}\|^2_{\Lambda_{kjl}} \quad (18)$$

where T_c^*, S_i^*, and S^* denote the ego-vehicle poses, object poses, and STS-based scaling factor, respectively. Σ and Λ represent the covariance of the pose estimation and sensor measurements, respectively. e_{mea} represents the error between the measured pose transformations z_{jl} and the predicted pose $\hat{z}(m_{k,j}, m_{k,l})$ for all features $m_{k,j}, m_{k,l}$.

Similarly, the poses $T_k^i \in SE(3)$ of object i at moment k can be solved by minimizing the reprojection error between the 3D pose $X_{k,j}^i \in \mathbb{R}^3$ of the object point $m_{k,j}^i$ and its corresponding 2D pose $x_{k,j}^i \in \mathbb{R}^2$:

$$e_{obj} = (x_{k,j}^i - \pi(T_k^i, X_{k,j}^i))P(m_{k,j}^i), \quad (2)$$

The Motion

$$e_{obj} = (x_{k,j}^i - \pi(T_k^i, {}_{k-1}\boxed{H_k^i}\, X_{k-1,j}^i))P(m_{k,j}^i).$$

Fig. 3 illustrates the pose transformation in 3D space. $X_{k,j}^s \in \mathbb{R}^3$ and $T_k^c \in SE(3)$ are the 3D pose of the j-th static point $m_{k,j}^s$ and that of the ego-vehicle at moment k, respectively. Ego-motion can be estimated by minimizing the reprojection error of static points:

$$e_{cam} = (x_{k,j}^s - \pi(T_k^c, X_{k,j}^s))P(m_{k,j}^s), \quad (1)$$

where $x_{k,j}^s \in \mathbb{R}^2$ is the 2D pose of $m_{k,j}^s$. π is the projection function. $P(m_{k,j}^s)$ is the STS probability of the static point.

4) Object Motion Constraints Based on STS: If the grid cells a and b belong to the same object, the Euclidean distance remains constant in consecutive frames motion, i.e., $d_{k-1,jl} = d_{k,jl}$. Meanwhile, the difference between the motion changes H^i of the two grid cells is minimal. The similarity error function can be defined as:

$$e_{sim} = \| \|C_a - C_b\| - d_{jl} \| + ({}_{k-1}H_k^{i,j})^{-1}\, {}_{k-1}H_k^{i,j+1}. \quad (17)$$

Thank you