LEAP-VO: Long-term Effective Any Point Tracking for Visual Odometry

Weirong Chen[1,2 *], Le Chen[3], Rui Wang[4], Marc Pollefeys[4]

* Work done for his Master's thesis at Microsoft

1 TUM 2 MCML Munich Center for Machine Learning 3 4

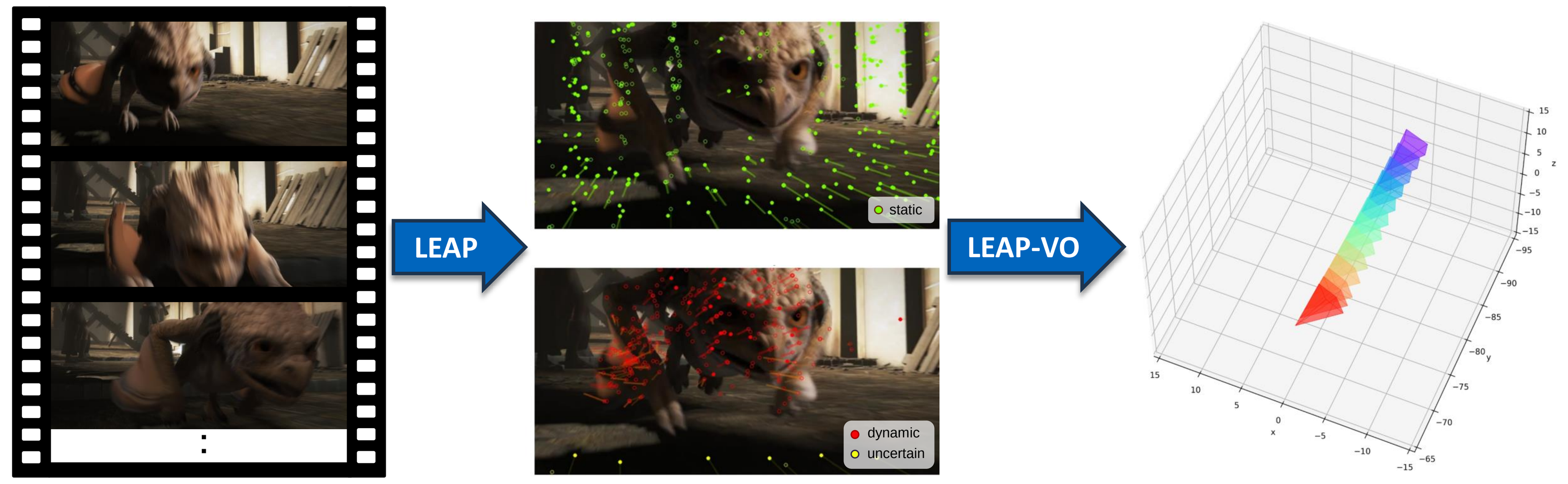
CVPR SEATTLE, WA JUNE 17-21, 2024

Paper, Code and Demo are available at
chiaki530.github.io/projects/leapvo

Overview

Problem: Monocular Visual Odometry

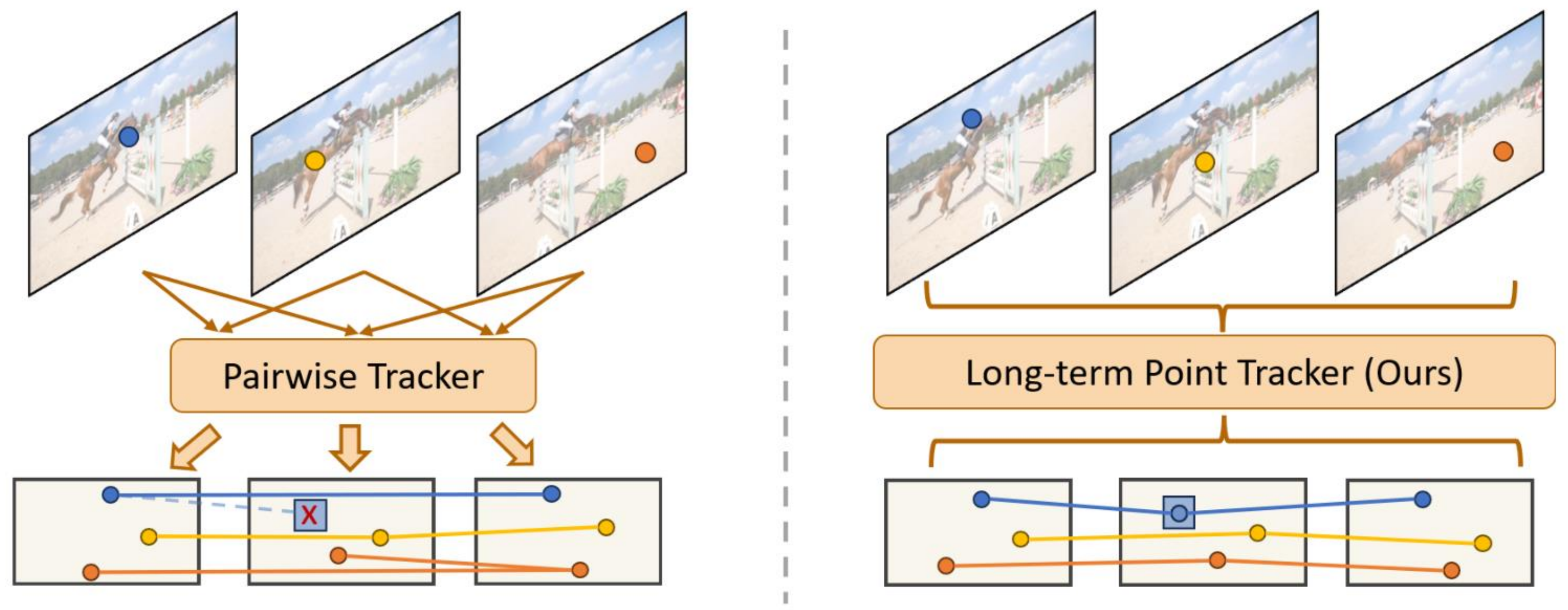
RGB Video → Sparse Point Trajectory → Camera Motion



Challenges
- Occlusion
- Dynamic Scenes
- Low-texture Area

Our Solution: Temporal Context
- Long-term Point Tracking
- Anchor-based Motion Estimation
- Temporal Probabilistic Modeling

Motivations

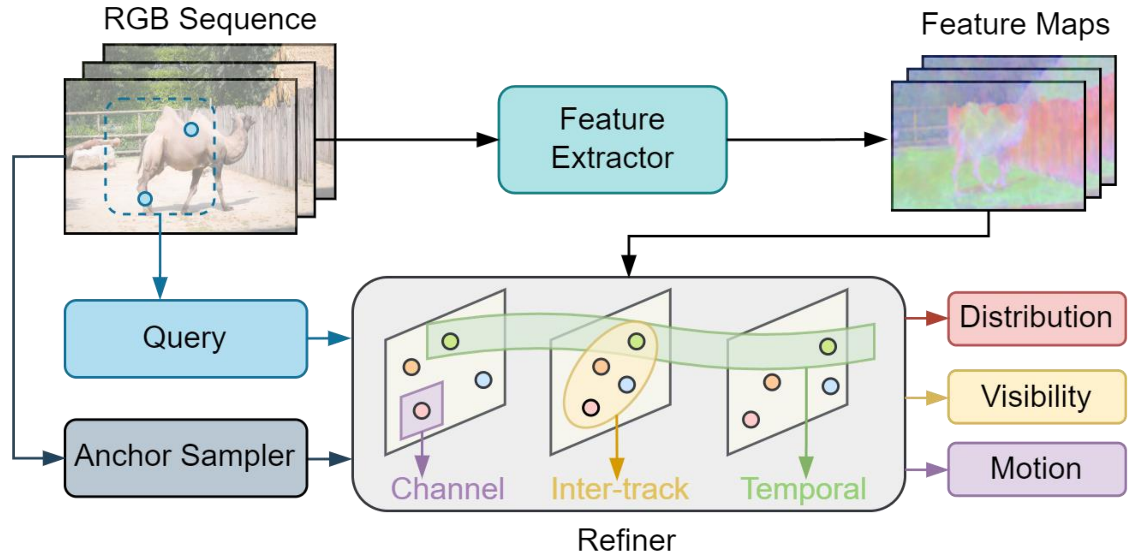


Method	Dynamic Detection	Occlusion Handling	Reliability Estimation
Two-view	Hard	Mostly Implicit	Per matching
LEAP (Ours)	Easy	Explicit	Per trajectory

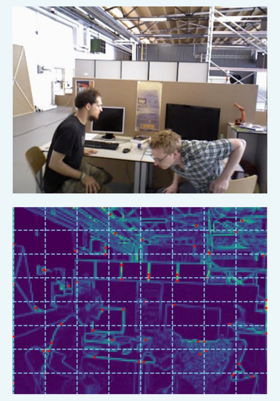
By leveraging **temporal context** and **continuous motion**, our LEAP can provide more reliable and accurate static trajectories for VO.

Point Tracking Front-end (LEAP)

- Extract feature map for each image
- Sample extra anchors to aid tracking
- Iteratively update point states by aggregating channel-wise, inter-track and temporal information
- Output trajectory distribution, visibility and dynamic motion label



(a) Anchor-based Motion Estimation



To better capture global motion patterns, we adopt distributed image gradient-based sampling.

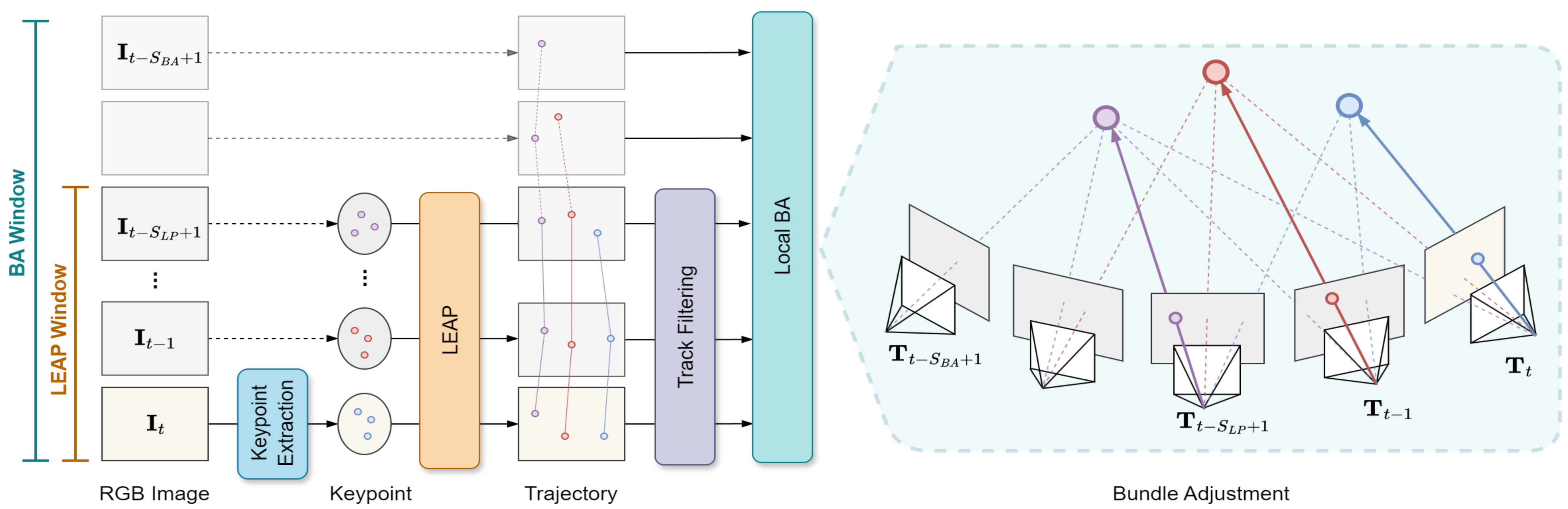
The selected anchors are **easy to track** and **well-distributed.**

(b) Temporal Probability Modeling

We model the distribution of point trajectories using two multivariate Cauchy distribution for 2D positions.

We apply **kernel method** on MLP features for predicting scale matrix.

LEAP-VO Pipeline

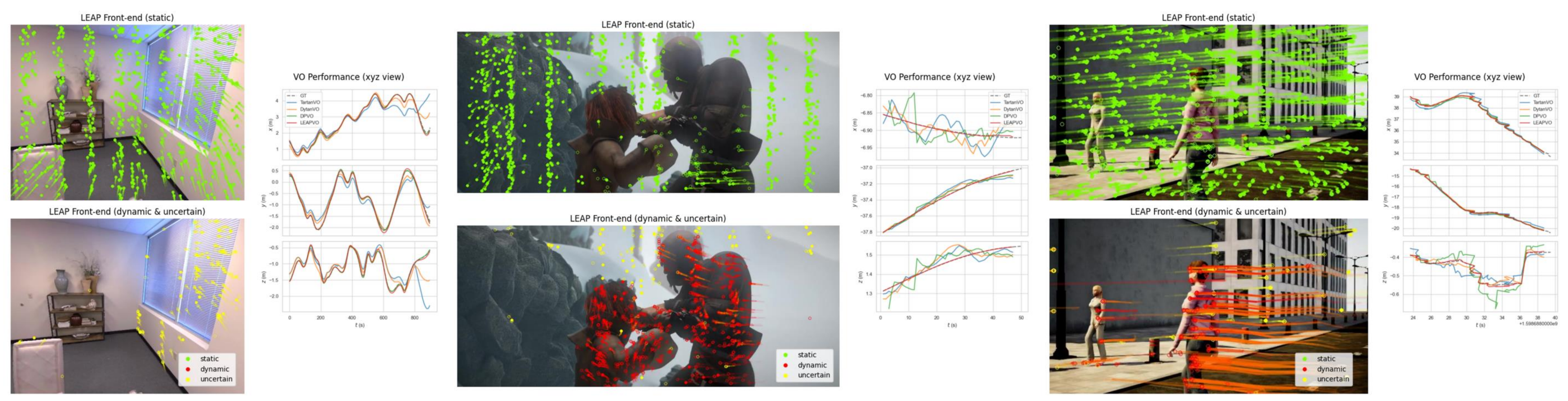


1. Given a new image, the **keypoint extractor** extracts new keypoints from this frame.
2. All keypoints are tracked by **LEAP front-end** across all other frames within the current LEAP window, followed by a **track filtering module** to remove dynamic and unreliable points.
3. The **local BA module** is applied on the current BA window to update the camera poses and 3D positions of the extracted keypoints by minimizing the reprojection error.

Results

Camera Tracking Performance

Method	Replica			MPI Sintel			TartanAir Shibuya
	ATE (m)	RPE trans (m)	RPE rot (deg)	ATE (m)	RPE trans (m)	RPE rot (deg)	ATE (m)
ORB-SLAM2	0.086	0.030	0.650	X	X	X	0.304
DynaSLAM	0.039	0.017	0.366	X	X	X	X
DROID-SLAM	0.267	0.036	2.631	0.175	0.084	1.912	0.124
TartanVO	0.406	0.036	2.063	0.238	0.093	1.305	0.246
DytanVO	0.289	0.035	2.146	0.131	0.097	1.538	0.061
DPVO	0.257	0.036	2.635	0.076	0.078	1.722	0.151
LEAP-VO (Ours)	**0.204**	**0.030**	**1.992**	**0.037**	**0.055**	**1.263**	**0.029**

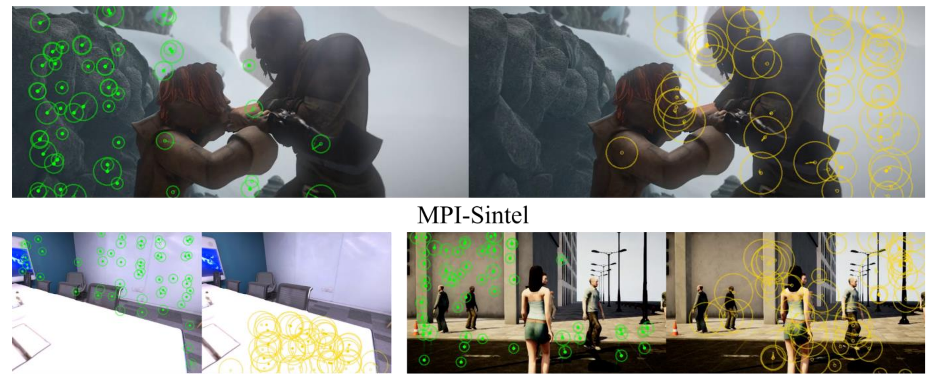


Replica MPI Sintel TartanAir Shibuya

Quantitative and qualitative comparisons on Replica, MPI Sintel and TartanAir Shibuya datasets. Our method shows better camera tracking performance.

More Studies

(a)



Replica (b) TartanAir-Shibuya

Method	Track Filtering			MPI Sintel		
	Vis.	Dyn.	Conf.	ATE	RPE trans	RPE rot
LEAP-VO	✓	-	-	0.118	0.078	1.340
	✓	✓	-	0.085	0.062	1.281
	✓	-	✓	0.054	0.058	1.274
	✓	✓	✓	**0.037**	**0.055**	**1.263**

(c)

(a) Qualitative results for dynamic track estimation. (b) Qualitative results of point-wise temporal uncertainty measurements. (c) Effect of track filtering module.