University of Zurich[UZH] | ETH zürich
Institute of Informatics – Institute of Neuroinformatics
ROBOTICS & PERCEPTION GROUP

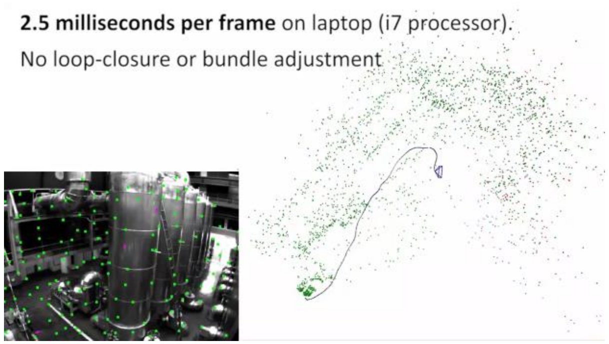
Benchmarking SLAM:

Current Status and the Road Ahead

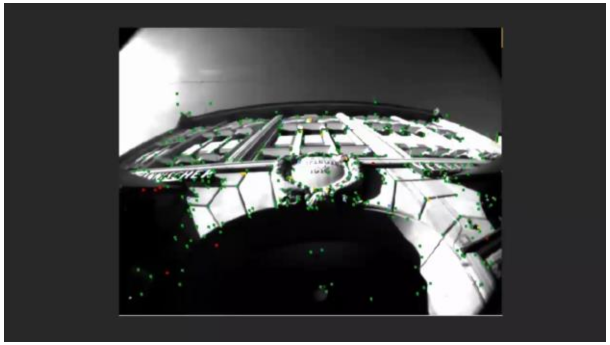
Davide Scaramuzza & Zichao Zhang

Slides, Publications, Videos, Code: http://rpg.ifi.uzh.ch/

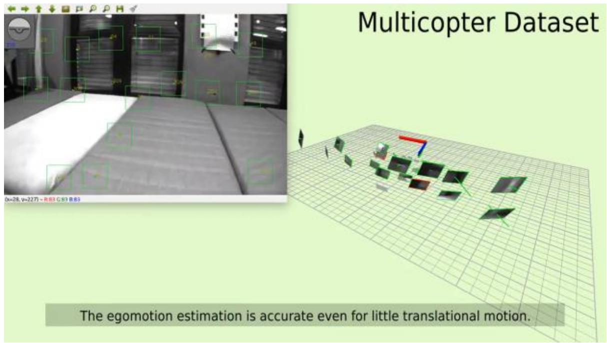
There are more and more VIO-VISLAM algorithms



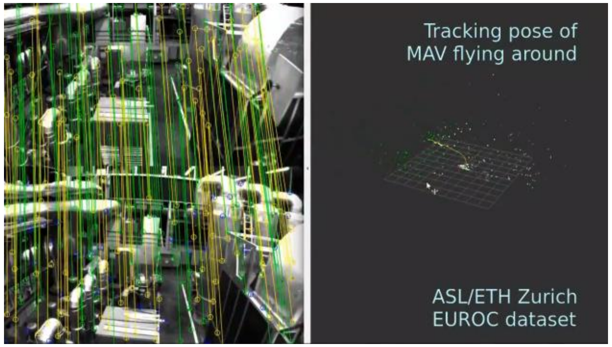
SVO+MSF



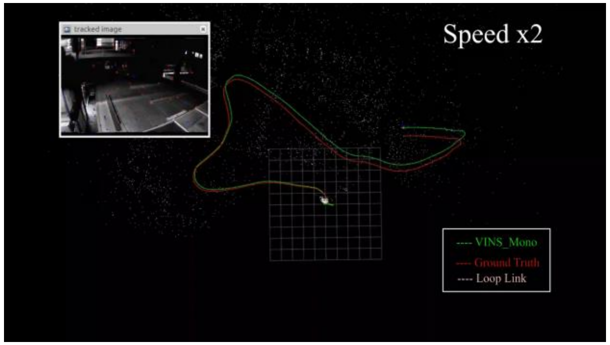
MSCKF



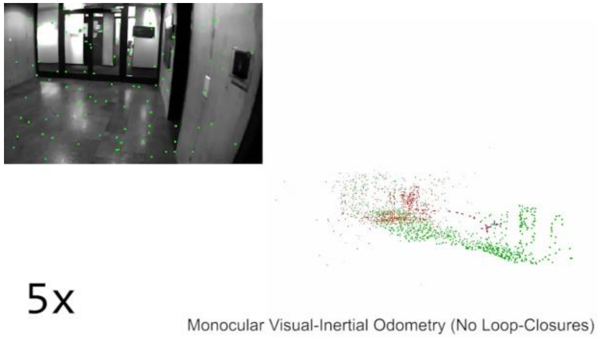
ROVIO



OKVIS



VINS-Mono



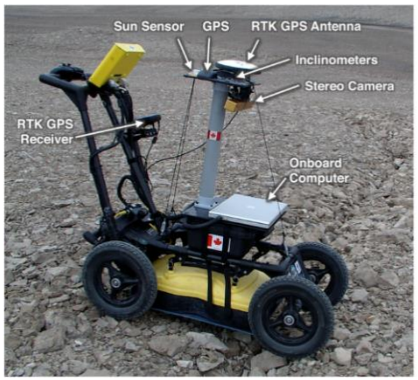
SVO+GTSAM

How do we compare them?

Example Real-World Datasets

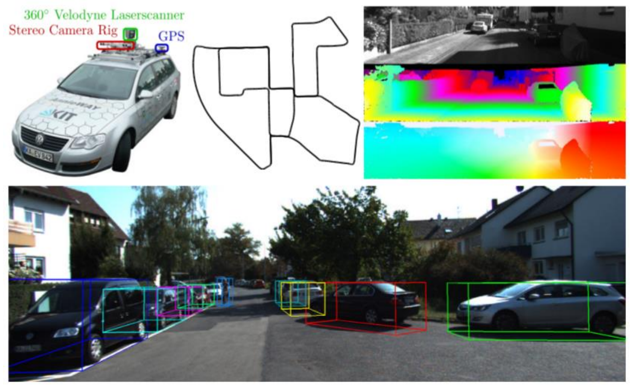
Devon Island [Furgale'11]

Stereo + D-GPS + inclinometer + sun sensor



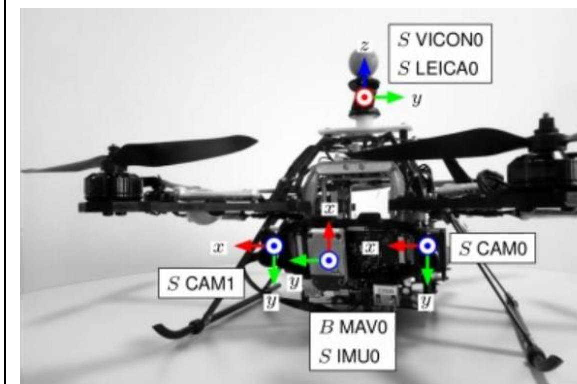
KITTI [Geiger'12]

Automobile, Laser + stereo + GPS, multiple tasks



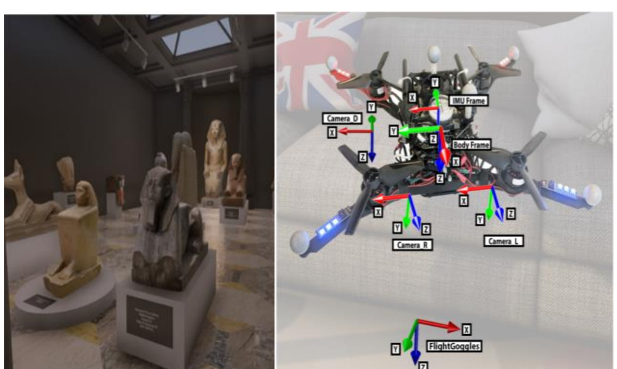
EuRoC [Burri'16]

MAV with synchronized IMU and stereo



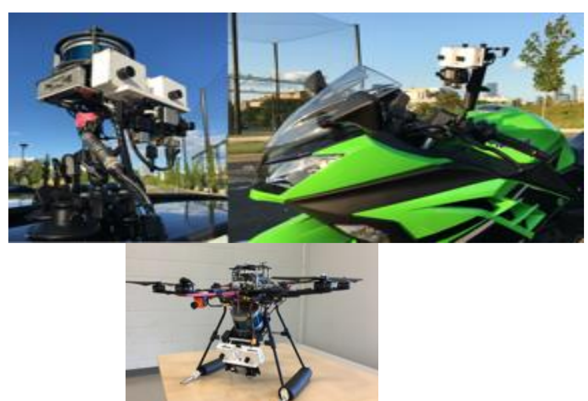
Blackbird [Antonini'18]

MAV indoor aggressive flight with rendered images and real dynamics + IMU



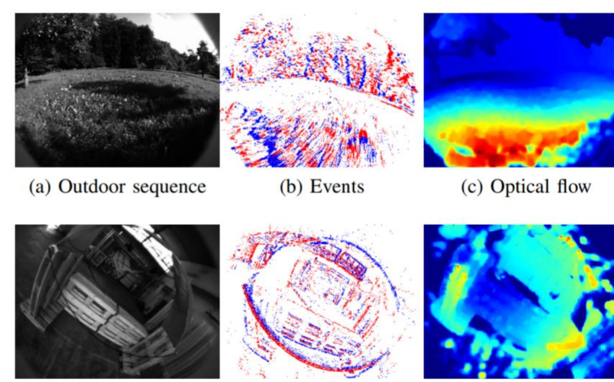
MVSEC [Zhu'18]

Events, frames, lidar, GPS, IMU from cars, drones, and motorcycles



UZH Drone Racing [Delmerico'19]

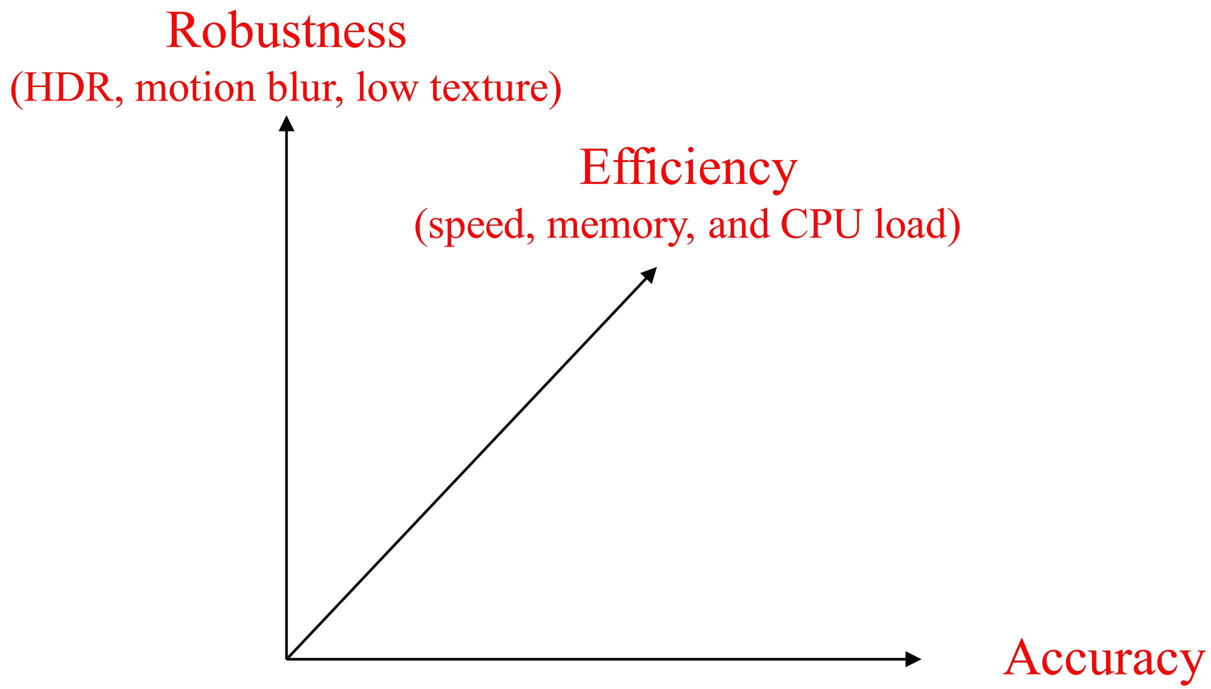
MAV aggressive flight, standard + event cameras, IMU, indoors and outdoors



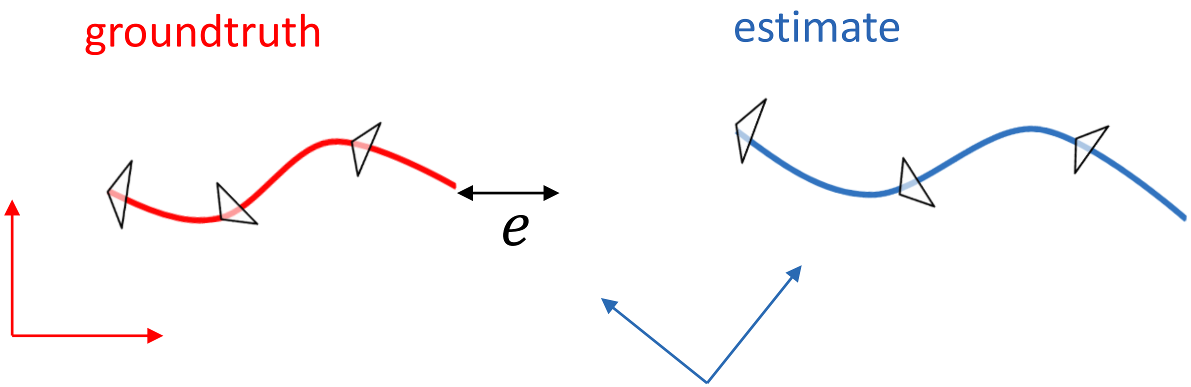
(a) Outdoor sequence (b) Events (c) Optical flow

(d) Indoor sequence (e) Events (f) Optical flow

What metrics should be used?



Evaluation is a non-trivial task...



Direct difference?

- Different reference frame
- Different scale
- Different times stamps
- ...

Maybe align the first poses and measure the *end-pose error*?

- **How many poses** should be used for the alignment?
- **Not robust:**
  - **Most VIOs are non-deterministic** (e.g., RANSAC, multithreading) → every time you run your VIO on the same dataset, you get different results
  - **Not meaningful:**
    - too sensitive to the trajectory shape
    - does not capture the error statistics

Metric 1: Absolute Trajectory Error (ATE)

Absolute Trajectory Error
RMSE of the aligned estimate and the groundtruth.

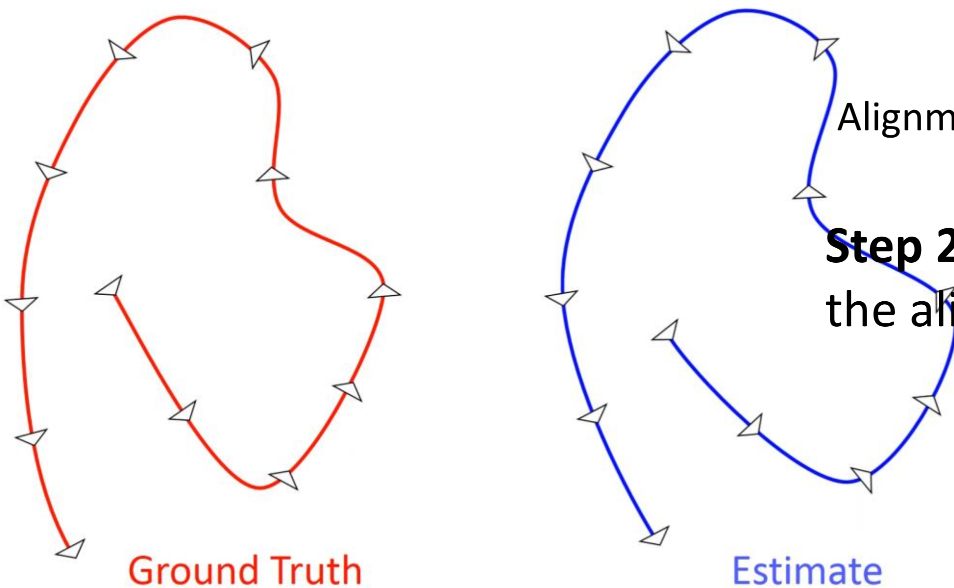
Step 1: Align the trajectory

groundtruth positions

$$\underset{R,T,s}{\operatorname{argmin}} \sum_{i=0}^{N} \|\hat{t}_i - sRt_i - T\|^2$$

estimated positions

Alignment parameters



Step 2: Root mean squared errors between the aligned estimate and the groundtruth.

$$\sqrt{\frac{\sum_{i=1}^{N} \|\hat{t}_i - sRt_i - T\|^2}{N}}$$

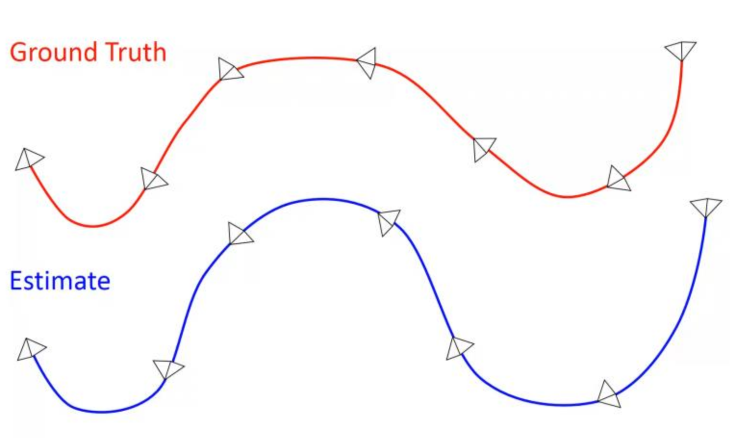
✓ Single number metric
✗ Many parameters to specify

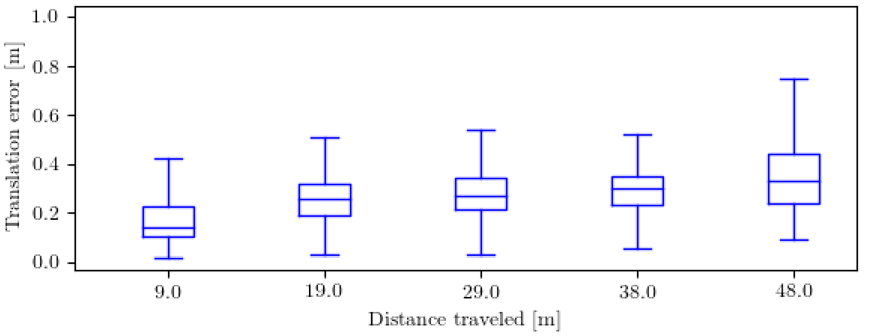
- Sturm et al., "A benchmark for the evaluation of RGB-D SLAM systems." IROS 2012.
- Zhang et al., "A tutorial on quantitative trajectory evaluation for visual (-inertial) odometry." IROS'18. PDF

Metric 2: Relative Trajectory Error (RTE)

Relative Error (Odometry Error)
Statistics of sub-trajectories of specified lengths.

➢ Calculate errors for all the subtrajectories of certain lengths.



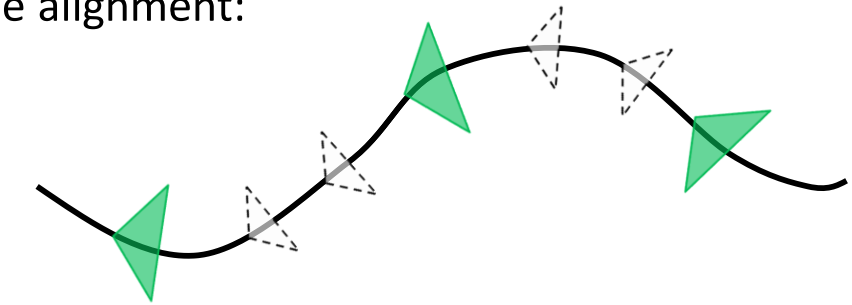


✓ Informative statistics
✗ Complicated to compute and rank

- Geiger et al. "Are we ready for autonomous driving? the KITTI vision benchmark suite." CVPR 2012.
- Zhang et al., "A tutorial on quantitative trajectory evaluation for visual (-inertial) odometry." IROS'18. PDF

Trajectory Accuracy: Error Metrics

- Both ATE and RTE are widely used in practice, but:
  - **Many details** need to be specified which are **often omitted in papers**
    - **Number of poses** used for the alignment (also, **frames or keyframes**?)
    - **Type of transformation used** for the alignment:
      - **SE(3)** for stereo VO
      - **Sim(3)** for monocular VO
      - **4DOF** for VIO
    - **Sub-trajectory lengths** in RTE

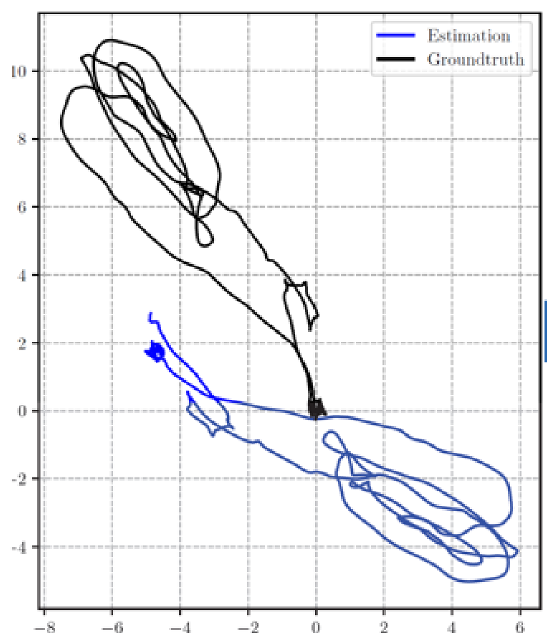


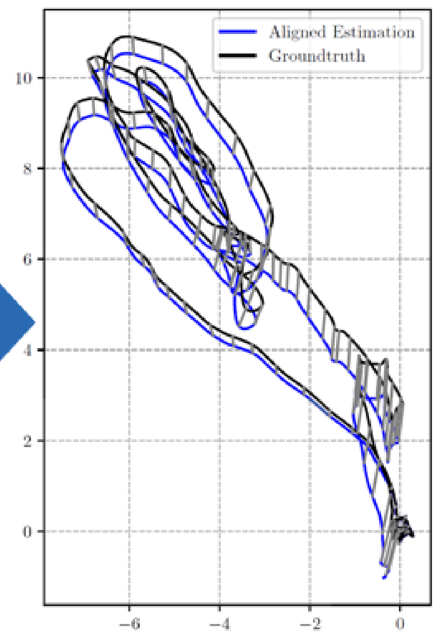
- White: Normal frames (used for real time pose update)
- Green: Keyframes (usually updated after BA)

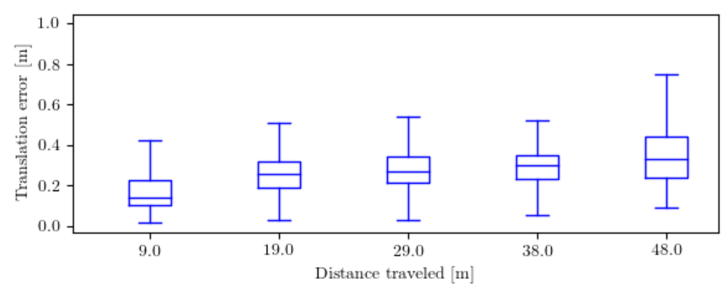
- Results are not directly comparable with different settings
  - Report the evaluation settings in detail.
  - Use/develop of publicly available evaluation tools to facilitate reproducible evaluation.

Trajectory Evaluation Toolbox

- **Designed to make trajectory evaluation easy!**
  - Implements **different alignment methods** depending on the sensing modalities: **SE(3)** for stereo, **sim(3)** for monocular, **4DOF** for VIO.
  - Implements **Absolute Trajectory Error** and **Relative Error**.
  - Automated evaluation of different algorithms on multiple datasets (for N runs).
- Code: https://github.com/uzh-rpg/rpg_trajectory_evaluation [Zhang, IROS'18]

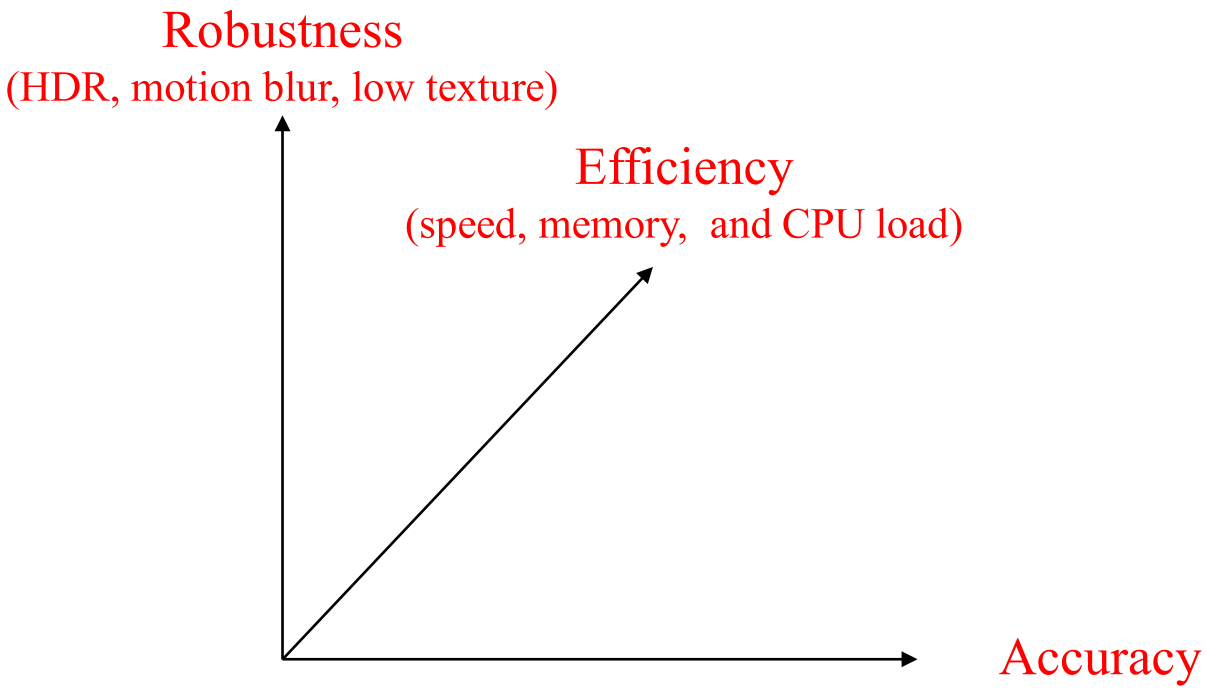






Zhang et al., "A tutorial on quantitative trajectory evaluation for visual (-inertial) odometry." IROS'18. [PDF]

What metrics should be used?

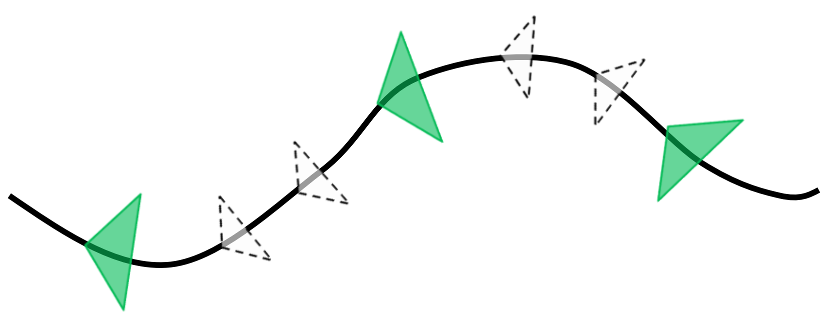


Benchmarking Efficiency

- Different computational resources
  - Memory
  - CPU load
  - Processing time

Depends not only on algorithm design, but also implementation, platforms, etc.

- There are different definitions of processing time in SLAM systems.



- White: Normal frames (used for real time pose update)
- Green: Keyframes (usually updated after BA)

- Processing time for real-time pose:
  $$t_{pose\ output} - t_{image\ arrival}$$
- Processing time for asynchronously executed threads (e.g., bundle adjustment)
- ……

Case study: VIO for Flying Robots [ICRA'18]

- Algorithms: MSCKF, OKVIS, ROVIO, VINS-Mono, SVO+MSF, SVO+GTSAM, VINS-Mono w/ and w/o loop closure

- Hardware: consider the limitation of flying robots

Intel NUC

Odroid XU4

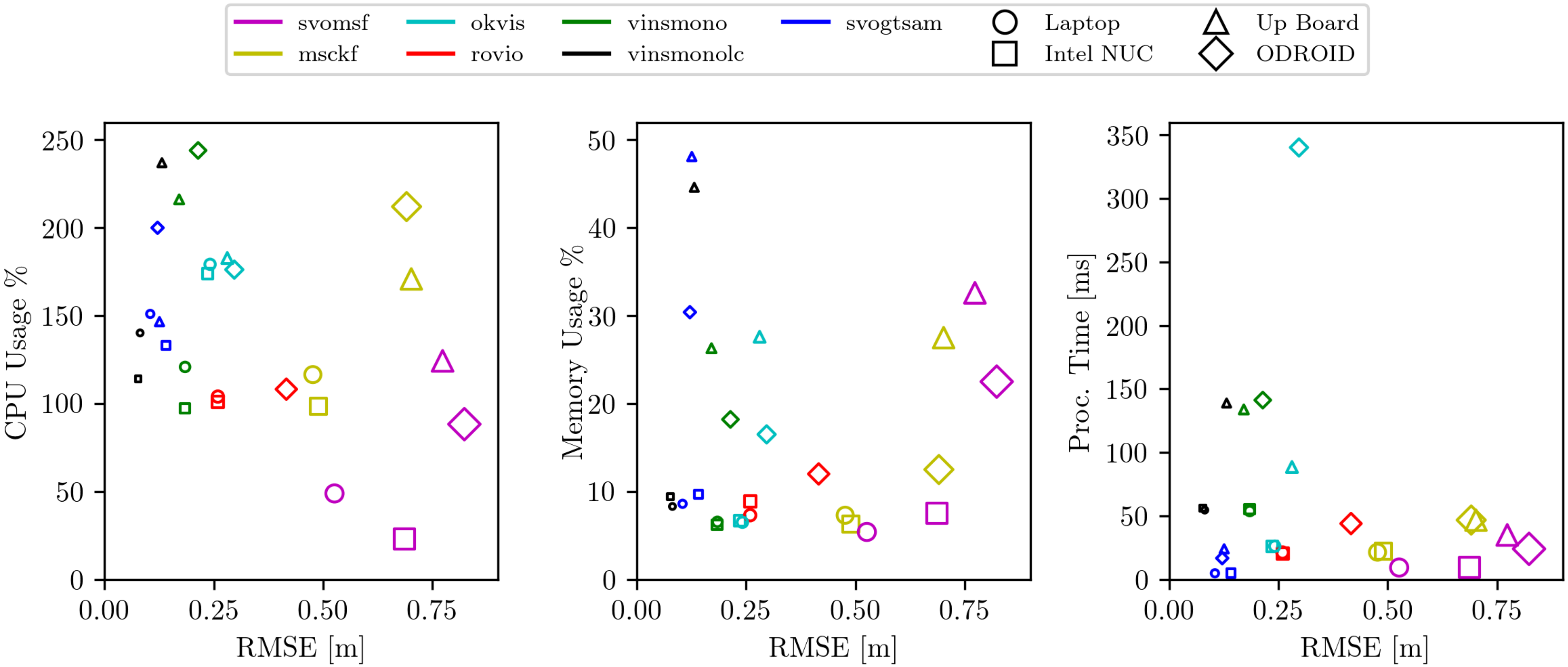
Up Board

Intel Lenovo W540 i7laptop

- Evaluation
  - **Absolute odometry error** – RMSE after sim(3) trajectory alignment (7DoF)
  - **Relative odometry error** – error distribution of the subtrajectories
  - **CPU usage** – total load of CPU
  - **Memory usage** – total percentage of available RAM
  - **Time per frame** – from input until pose is updated

Delmerico, Scaramuzza, A Benchmark Comparison of Monocular Visual-Inertial Odometry Algorithms for Flying Robots, ICRA'18. PDF. Video.

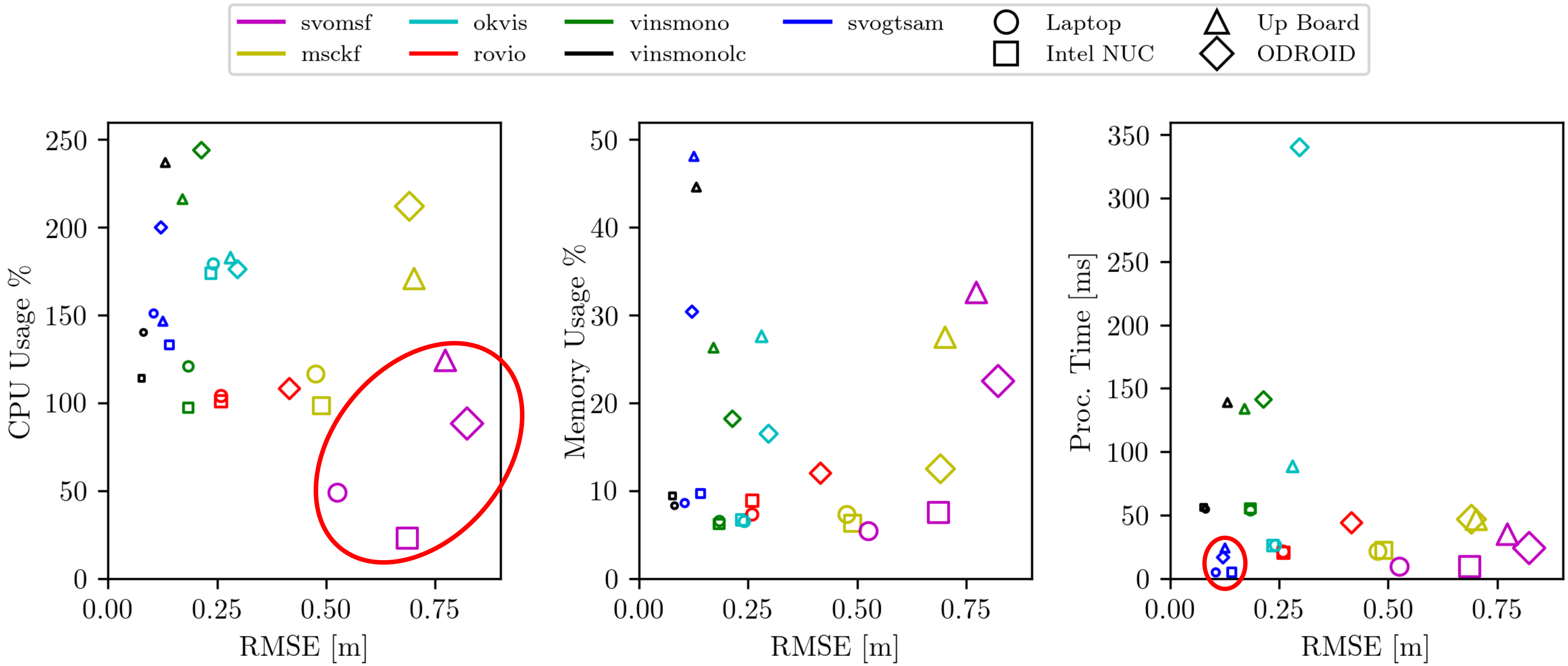
Case study: VIO for Flying Robots [ICRA'18]



No free lunch: more computation → better accuracy

Delmerico, Scaramuzza, A Benchmark Comparison of Monocular Visual-Inertial Odometry Algorithms for Flying Robots, ICRA'18. PDF. Video

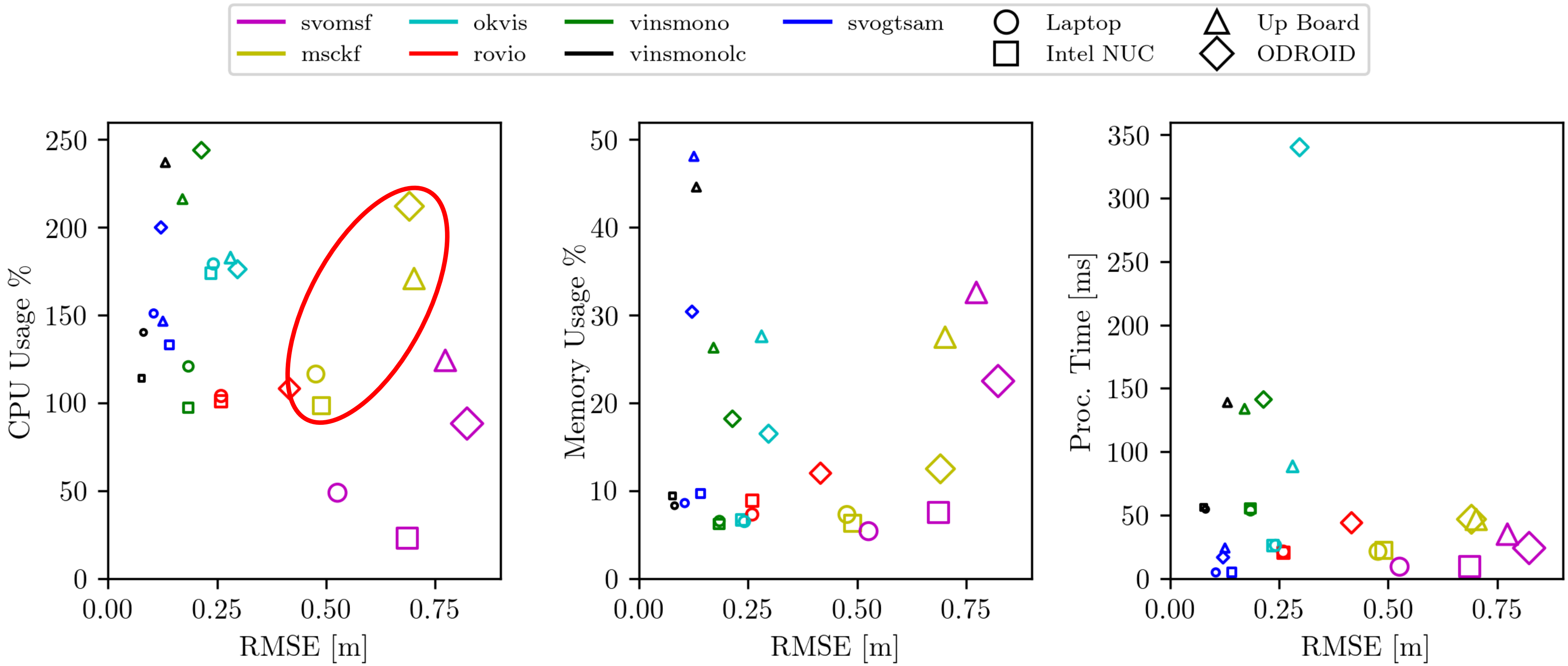
Case study: VIO for Flying Robots [ICRA'18]



SVO+MSF: most efficient but least accurate.

Delmerico, Scaramuzza, A Benchmark Comparison of Monocular Visual-Inertial Odometry Algorithms for Flying Robots, ICRA'18. PDF. Video

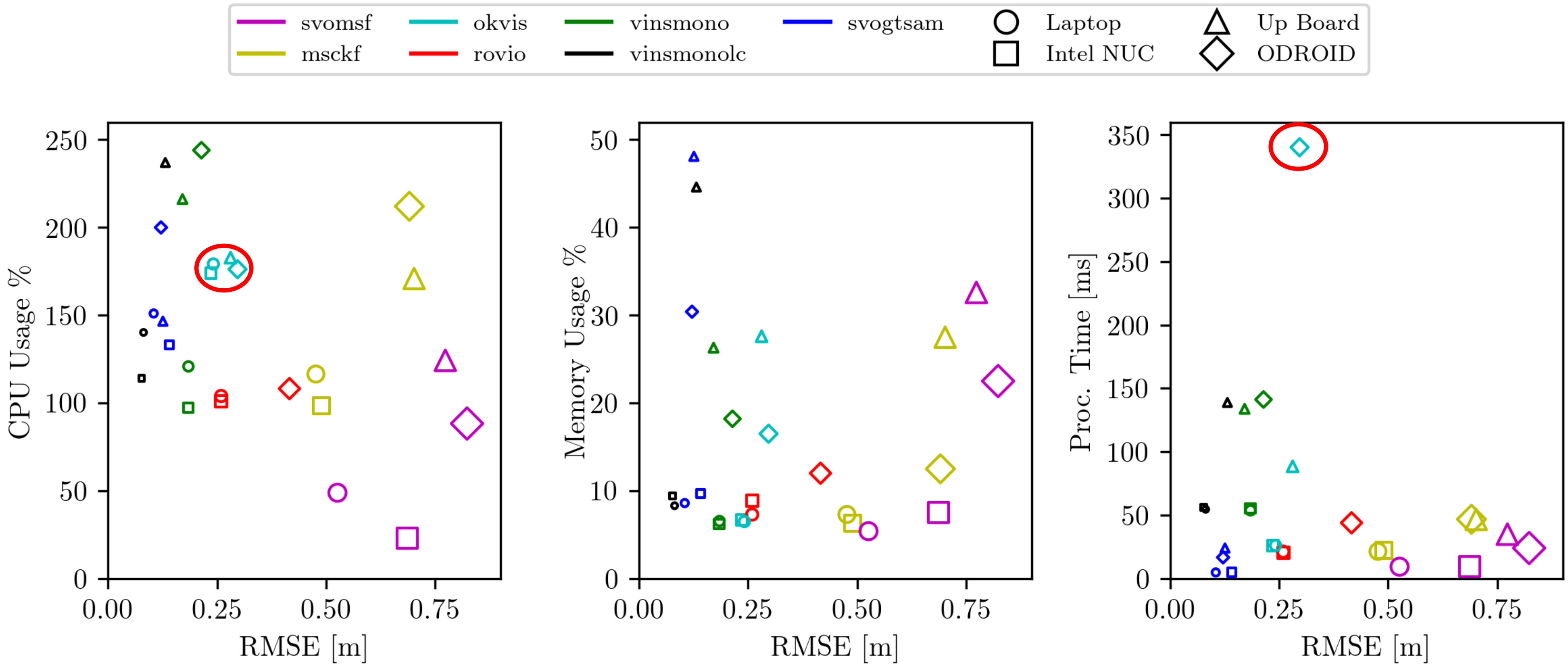
Case study: VIO for Flying Robots [ICRA'18]



MSCKF: successful on all sequences, but achieves lower accuracy than smoothing algorithms.

Delmerico, Scaramuzza, A Benchmark Comparison of Monocular Visual-Inertial Odometry Algorithms for Flying Robots, ICRA'18. PDF. Video

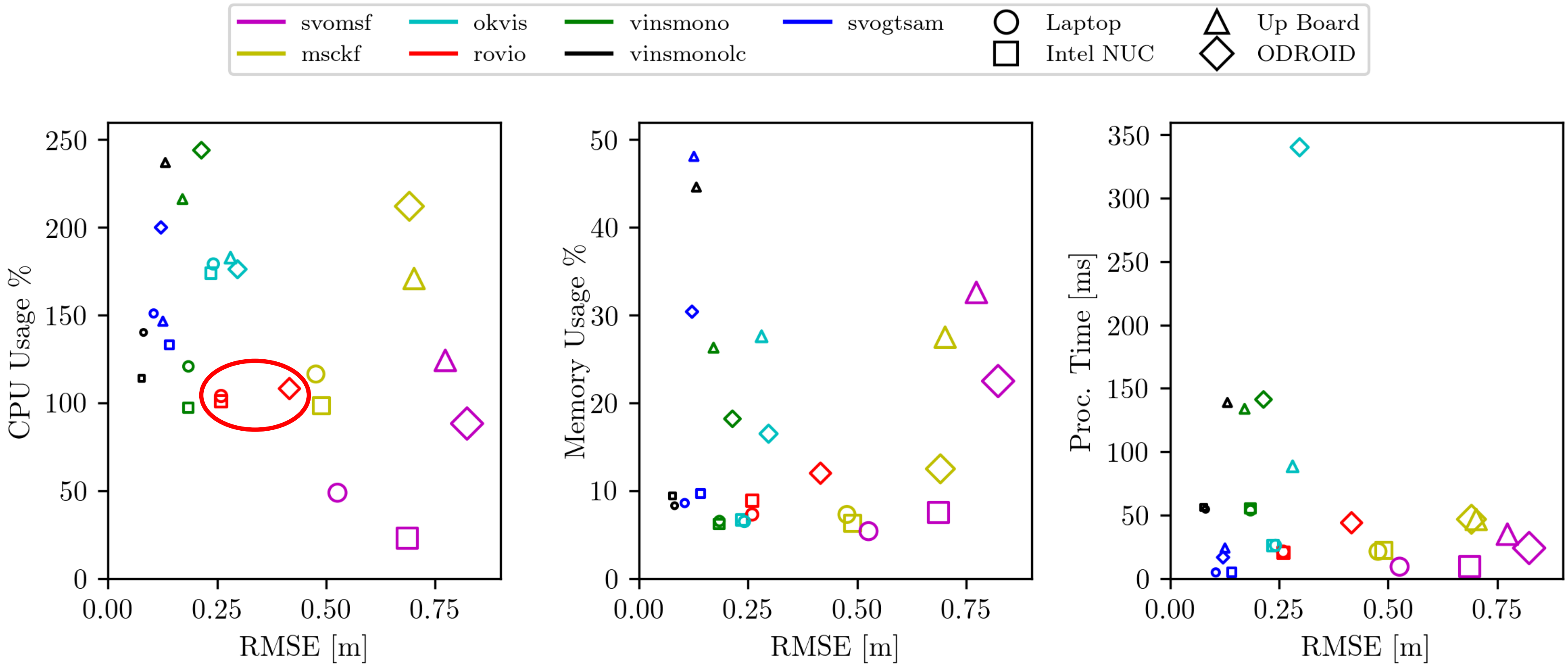
Case study: VIO for Flying Robots [ICRA'18]



OKVIS: consistent performance across all HW platforms, but low update rate on the ODROID.

Delmerico, Scaramuzza, A Benchmark Comparison of Monocular Visual-Inertial Odometry Algorithms for Flying Robots, ICRA'18. PDF. Video

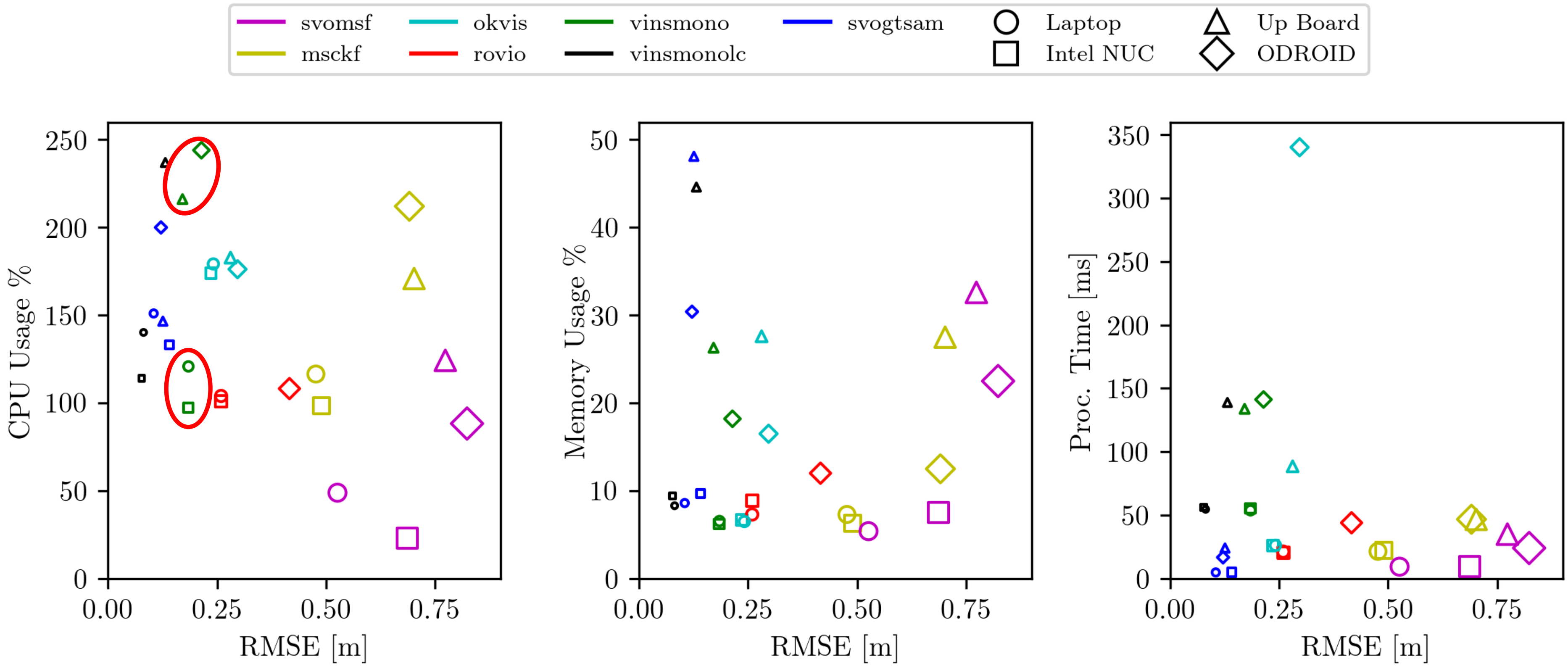
Case study: VIO for Flying Robots [ICRA'18]



ROVIO: tight bound on resource usage, competitive accuracy, but unable to run on Up Board due to its low clock speed.

Delmerico, Scaramuzza, A Benchmark Comparison of Monocular Visual-Inertial Odometry Algorithms for Flying Robots, ICRA'18. PDF. Video

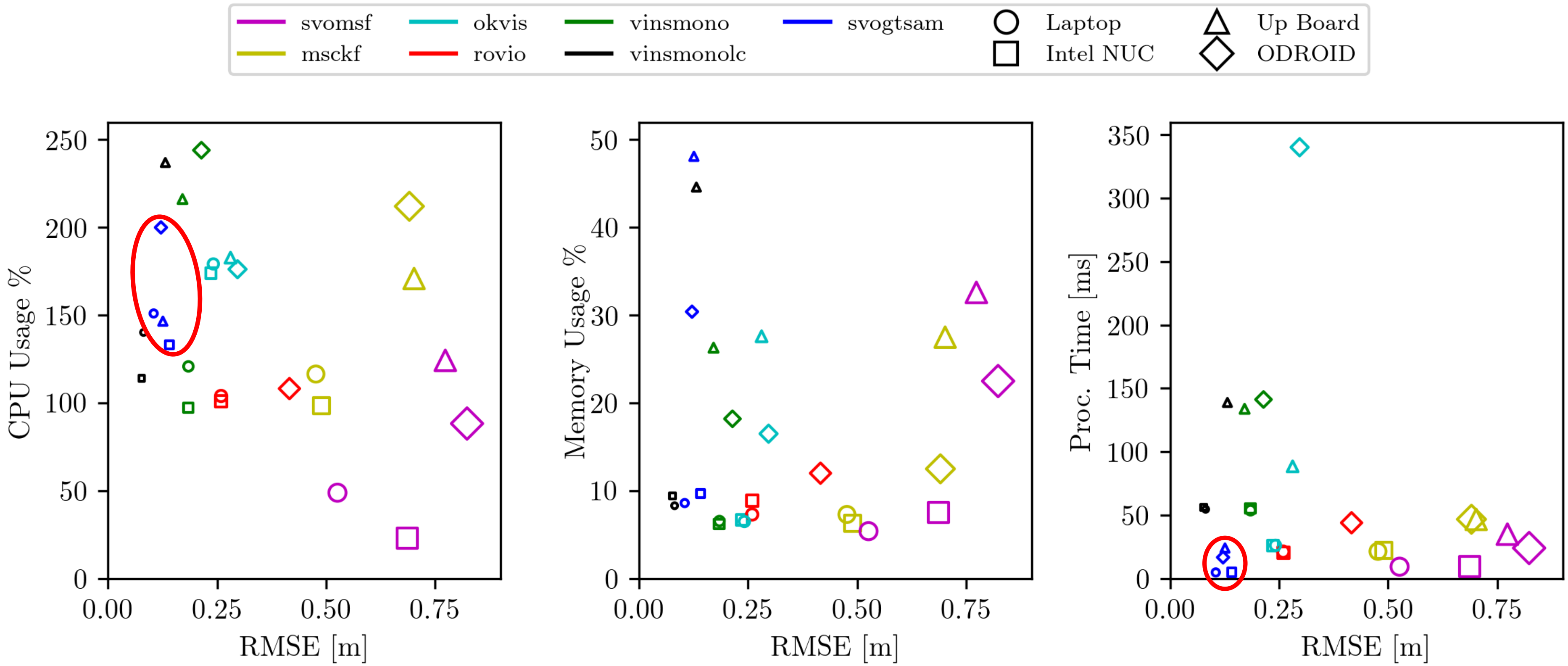
Case study: VIO for Flying Robots [ICRA'18]



VINS-Mono: consistently robust and accurate, even more with loop closure enabled, but high resource usage.

Delmerico, Scaramuzza, A Benchmark Comparison of Monocular Visual-Inertial Odometry Algorithms for Flying Robots, ICRA'18. PDF. Video

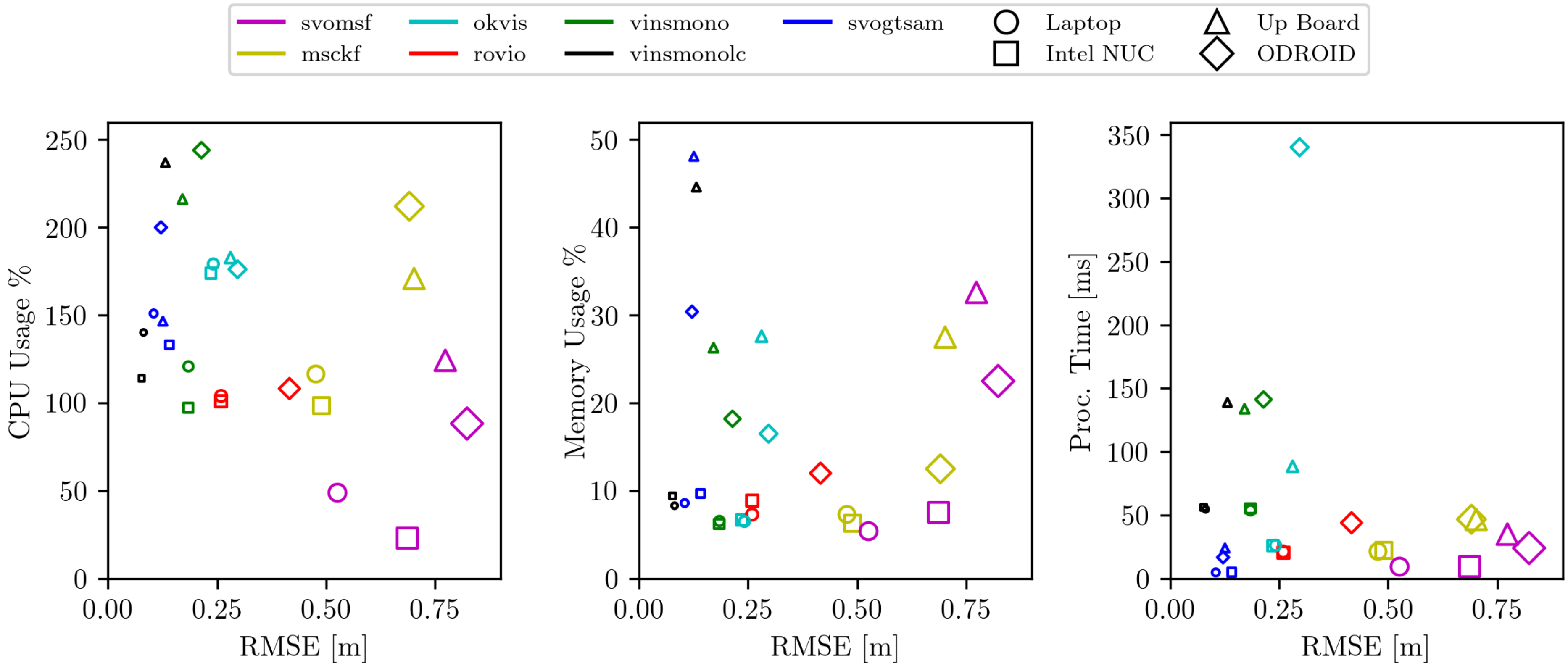
Case study: VIO for Flying Robots [ICRA'18]



SVO+GTSAM: high accuracy and modest resource use, but lack of robustness due to numerical instability during GTSAM optimization.

Delmerico, Scaramuzza, A Benchmark Comparison of Monocular Visual-Inertial Odometry Algorithms for Flying Robots, ICRA'18. PDF. Video

Case study: VIO for Flying Robots [ICRA'18]



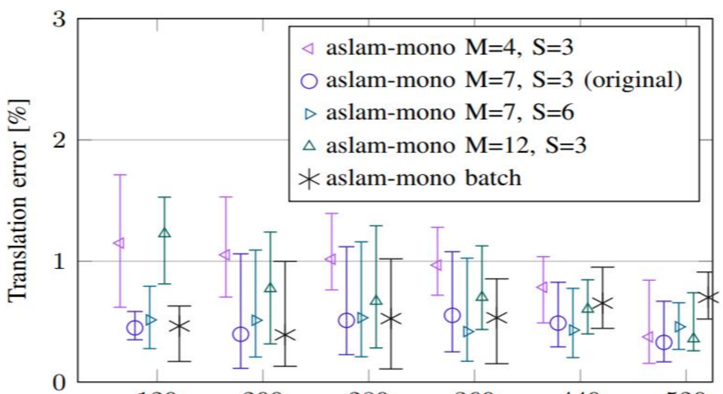
Findings:

- Results (accuracy & efficiency) vary **depending on the platforms**
- **No free lunch**: more computation → better accuracy

Delmerico, Scaramuzza, A Benchmark Comparison of Monocular Visual-Inertial Odometry Algorithms for Flying Robots, ICRA'18. PDF. Video

To recap:

- Do not interpret the results outside the evaluation context
  - **Performance varies** depending on:
    Specific algorithms + **specific** datasets + **specific** platforms
- Be very careful about (many, many) details
  - Parameters: how many keyframes in the sliding window?
  - Are we interested in real-time poses or refined poses ?



Error depending on sliding window size

Leutenegger et al. "Keyframe-based visual–inertial odometry using nonlinear optimization." *IJRR 2015*.

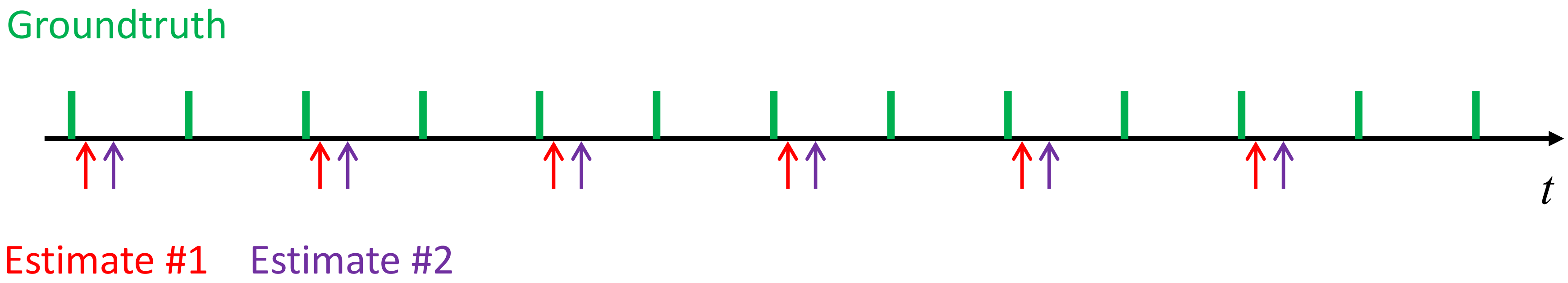
Bottom line: be very specific when reporting the results!

Research question: can we design a theoretically grounded trajectory error metric?

Trajectory Accuracy: open problems?

- ATE and RTE are observed to be correlated in practice, but their theoretical connections are not clear → **a unified metric**?

- In practice, the temporal association is usually done by finding the nearest groundtruth → **a more principled way**?



We can model the trajectory evaluation problem more rigorously in a probabilistic and continuous-time formulation and show theoretical connection between conventional ATE and RTE.

Zichao Zhang, Davide Scaramuzza, "*Rethinking Trajectory Evaluation for SLAM: a Probabilistic, Continuous-Time Approach*", **Best Paper Award** at the ICRA'19 SLAM Benchmarking workshop.

Algorithm Design Choices: Fair comparison?

Algorithm Design Choices: Fair comparison?

How can we evaluate the pros and cons of different algorithm design choices?

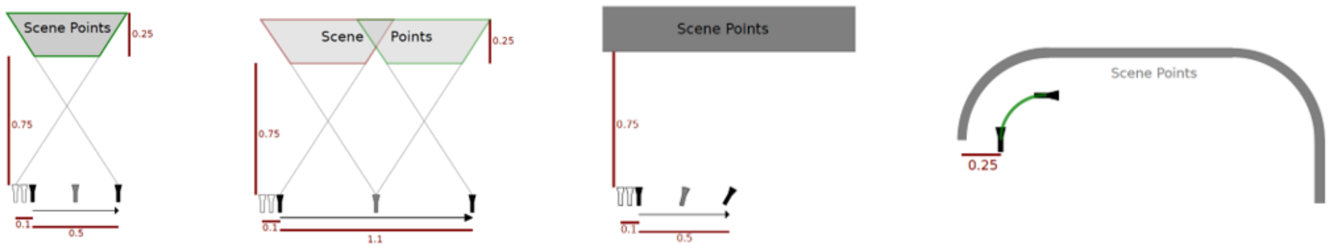
- Does the difference come from specific implementation details or the algorithm choice?
- Does the observed difference generalize to different situations?
- ....

It is **tricky but important** to separate the influence of the factors of interest by:

  - Standard implementations
  - Well-controlled simulation
  - …

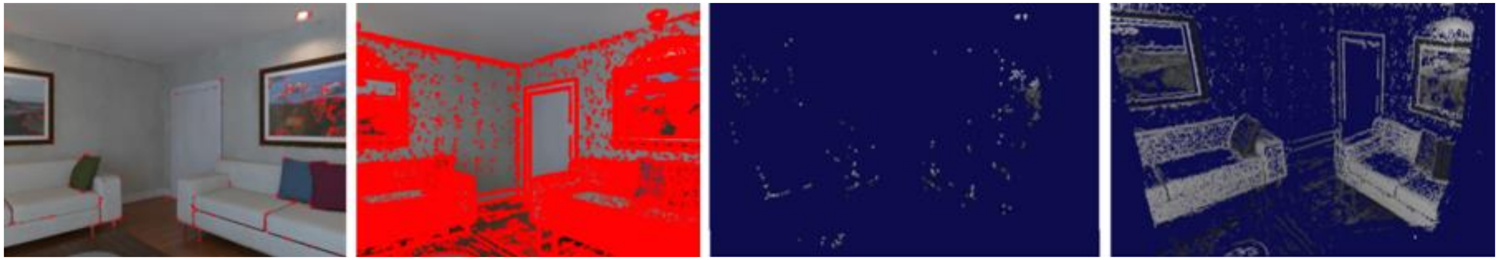
Algorithm Choices: Success stories

- Filter vs. Keyframe: representative, canonical setups



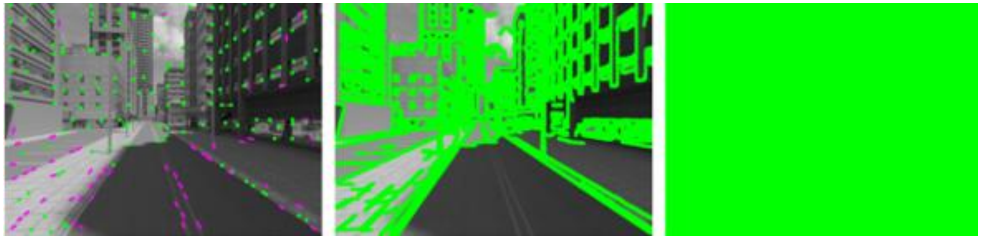
Strasdat et al. "Visual SLAM: why filter?." Image and Vision Computing 30, no. 2 (2012): 65-77.

- Sparse Joint Optimization vs. Dense Alternation: custom VO for comparison



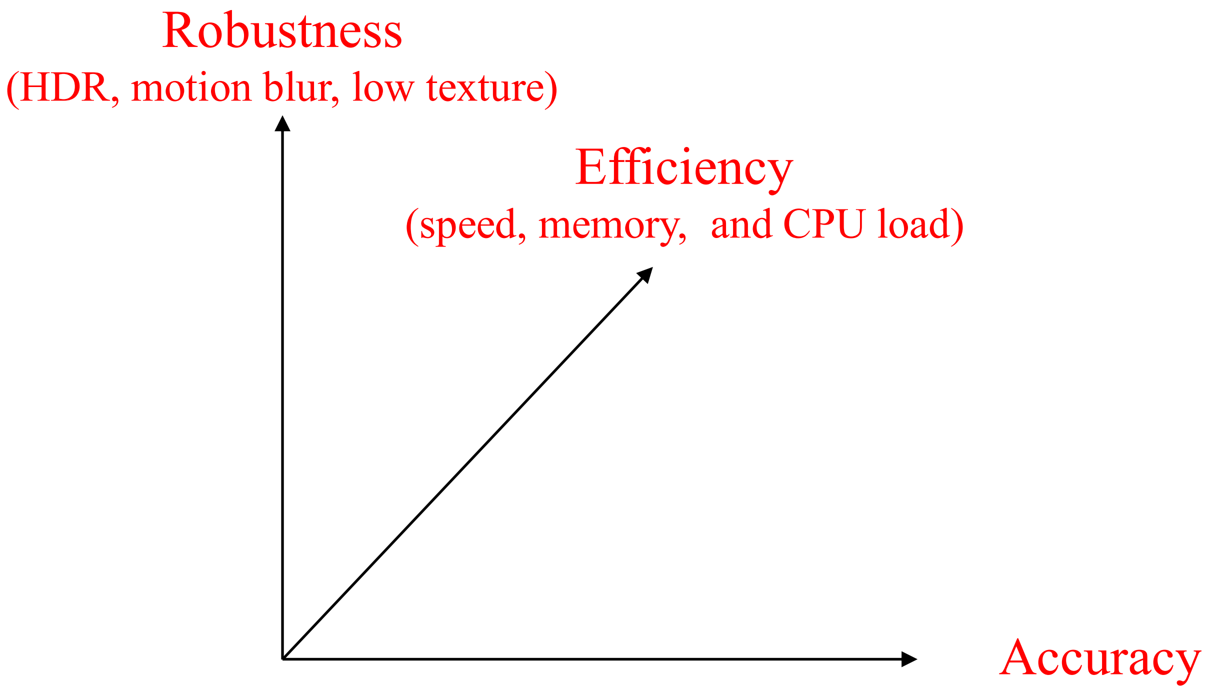
Platinsky et al. "Monocular visual odometry: Sparse joint optimisation or dense alternation?." ICRA 2017.

- Dense vs. Semi-dense vs. Sparse Image Alignment : specific algorithm modules



Forster et al.. "SVO: Semidirect visual odometry for monocular and multicamera systems." TRO 2017. PDF. Code.

What metrics should be used?



Robustness is the greatest challenge for SLAM today!

How to cope & quantify robustness to:

- low texture
- High Dynamic Range (HDR) scenes
- motion blur
- dynamically changing environments
- large latencies

How can we quantify the robustness of algorithms to such situations?

High Dynamic Range

Motion blur

Latency

Cadena, Carlone, Carrillo, Latif, Scaramuzza, Neira, Reid, Leonard, *Past, Present, and Future of Simultaneous Localization and Mapping: Toward the Robust-Perception Age,* IEEE Transactions on Robotics, 2016. [PDF](PDF)

Robustness is the greatest challenge for SLAM today!

How to cope & quantify robustness to:

- low texture
- High Dynamic Range (HDR) scenes
- motion blur
- dynamically changing environments
- large latencies

How can we quantify the robustness of algorithms to such situations?

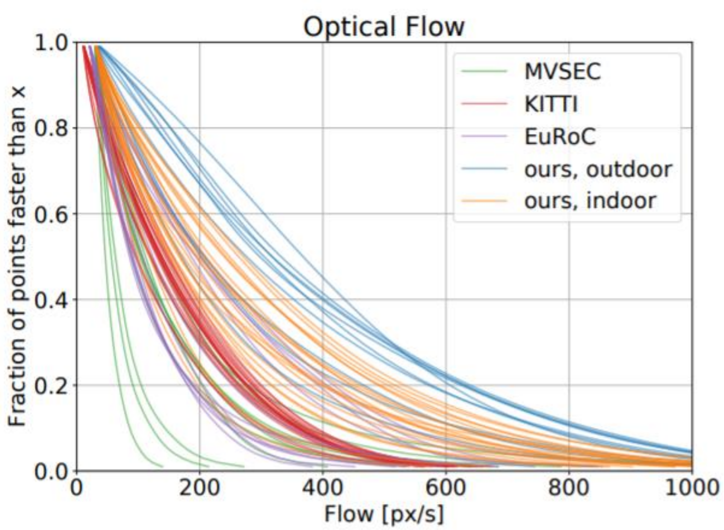
Also, most algorithms have random components

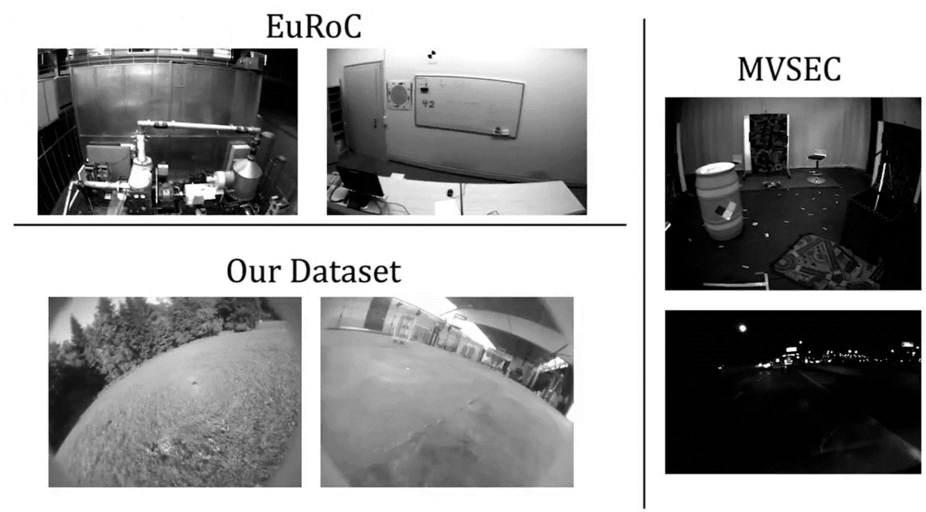
- RANSAC
- Feature selection to constrain computation
- …

Is the performance robust to algorithmic randomness?

Robustness

- Quantify the level of the challenge properly
  - E.g., optical flow for the aggressiveness for vision algorithms



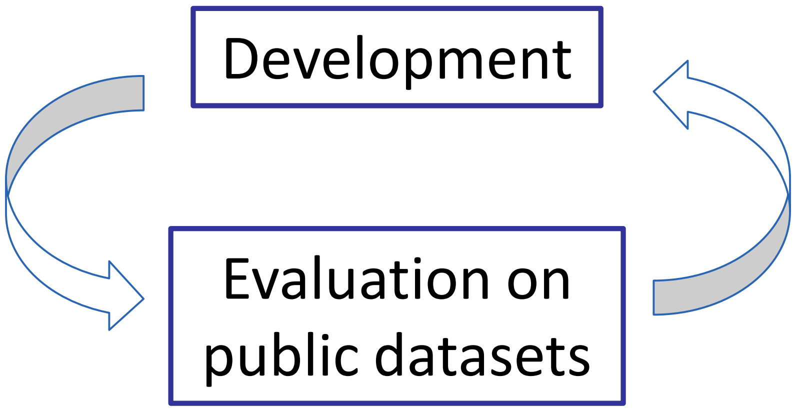


- Repeated experiments to get statistically meaningful results
  - Success rate
  - Mean/Median error
  - …

Delmerico et al. "Are We Ready for Autonomous Drone Racing? The UZH-FPV Drone Racing Dataset" ICRA'19
PDF. Video. Datasets.

Dataset Bias

Typical workflow of developing VO/VIO/SLAM algorithms:



As a community, we are overfitting the public dataset.

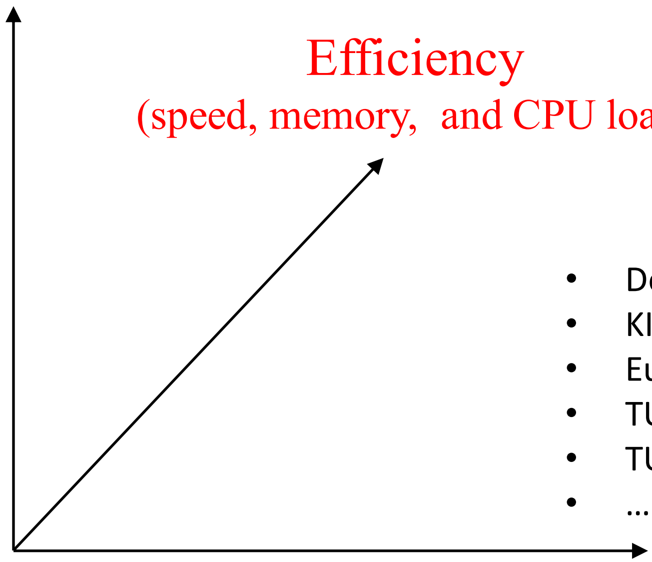
Potential problems:

- **Generalizability:** Performance on one does not guarantee to generalize to others
  - E.g., KITTI → low frame rate, not friendly for direct methods

- **Old datasets (e.g., KITTI) are already saturated:**
  - It becomes more and **more difficult to tell whether we are making real progress** or just overfitting the datasets.
  - E.g., **does 1 or 2 cm improvement in RMSE** over a 100 meter trajectory **really mean something**?

Dataset Bias

We need more datasets to evaluate the performance of SLAM algorithms along different axes



Robustness
(HDR, motion blur, low texture)

- BlackBird [Antonini'18]
- UZH-FPV dataset [Delmerico'18]
- Event Camera [Mueggler'17]
- MVSEC [Zhu'18]
- …

Efficiency
(speed, memory, and CPU load)

Accuracy

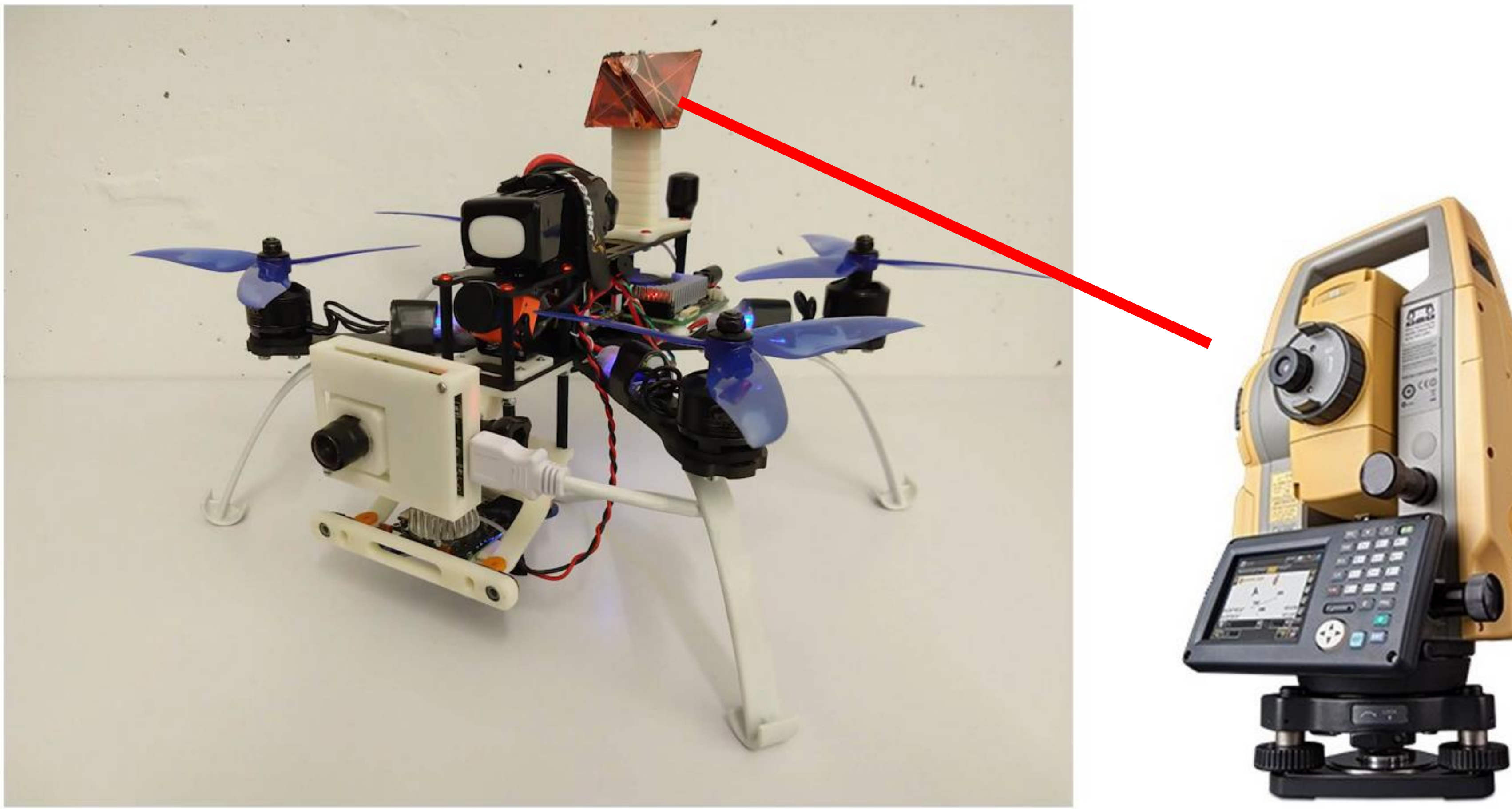
- Devon Island [Furgale'13]
- KITTI [CVPR'12]
- EuRoC [Burri'16]
- TUM-RGBD [Sturm'12]
- TUM VI Benchmark [Schubert'18]
- …

Realistic simulators:

- AirSim
- FlightGoggles [Guerra'19]
- ESIM [Rebecq'18]
- ……

UZH-FPV Drone Racing Dataset

Contains data recorded by a drone flying up to over 20m/s indoors and outdoors frown by a professional pilot. Contains frames, events, IMU, and Ground Truth from a Robotic Total Station: http://rpg.ifi.uzh.ch/uzh-fpv.html



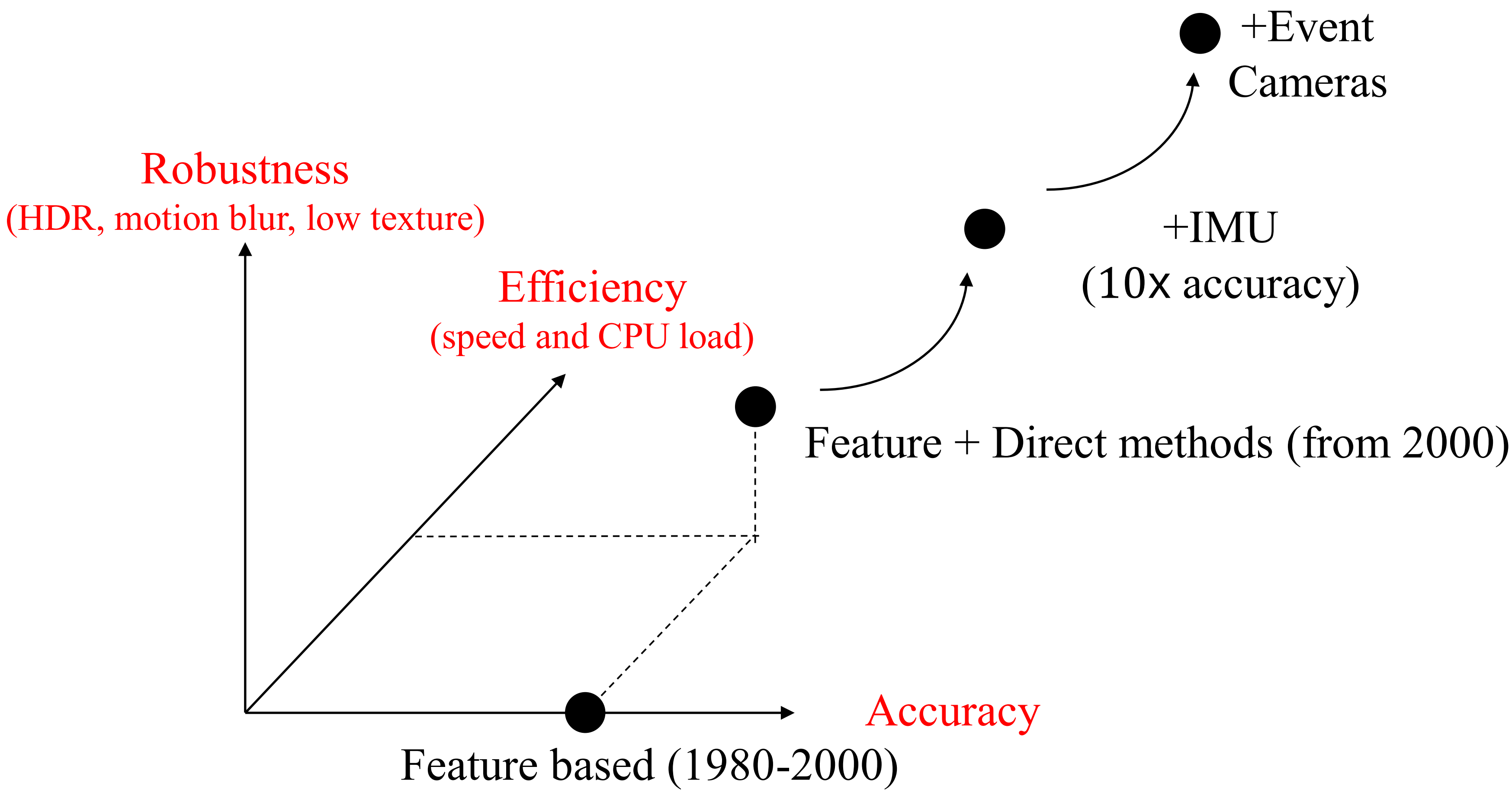
Delmerico et al. "Are We Ready for Autonomous Drone Racing? The UZH-FPV Drone Racing Dataset" ICRA'19
PDF. Video. Datasets.

UZH-FPV Drone Racing Dataset

- Recorded with a drone flown by a **professional pilot up to over 20m/s**
- Contains **images, events, IMU**, and **ground truth from a robotic total station**: http://rpg.ifi.uzh.ch/uzh-fpv.html

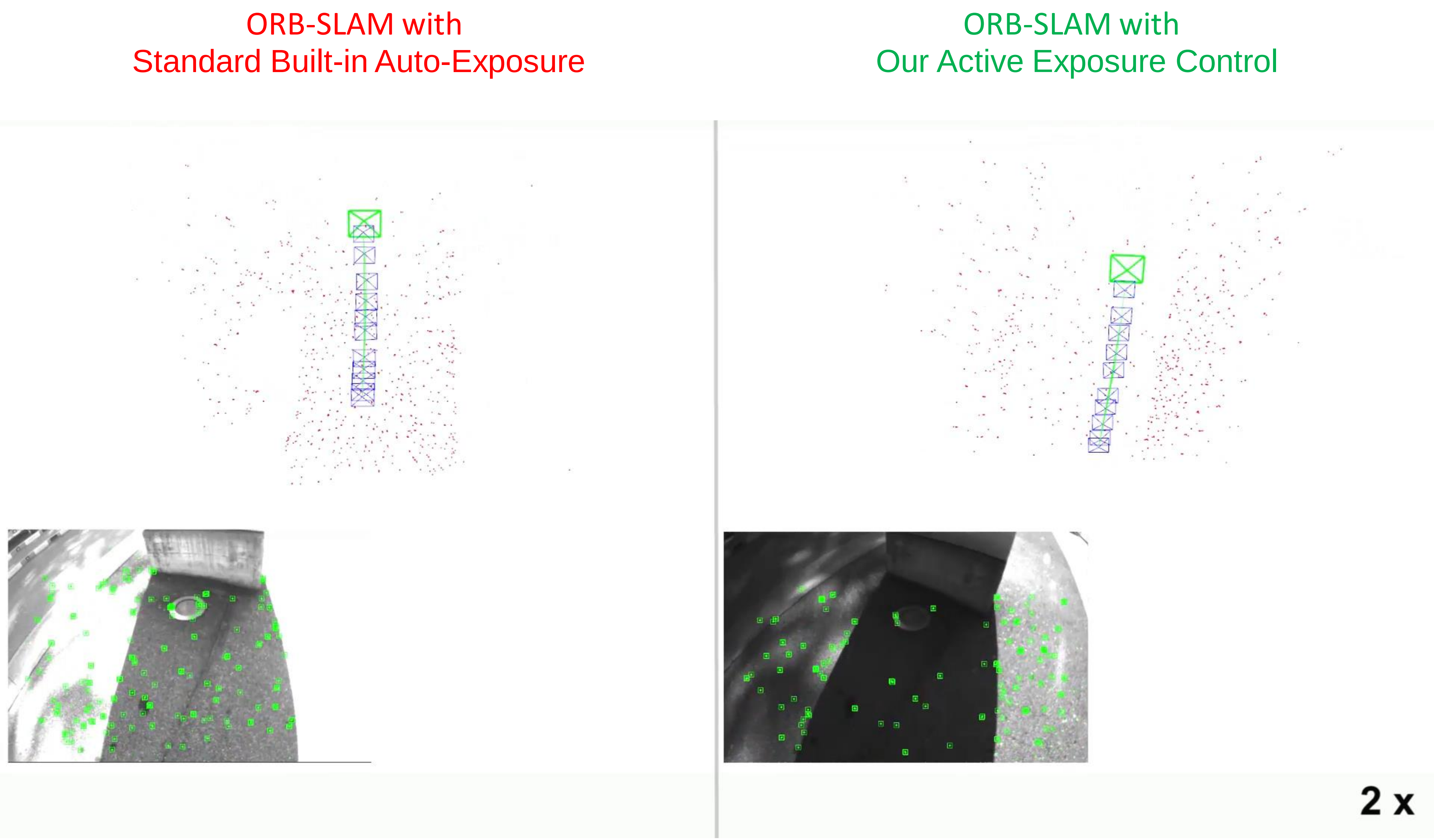
Delmerico et al. "Are We Ready for Autonomous Drone Racing? The UZH-FPV Drone Racing Dataset" ICRA'19
PDF. Video. Datasets.

My Personal View of the last 30 years of Visual Inertial SLAM



Opportunities

Active Exposure Control for Robustness in HDR scenes



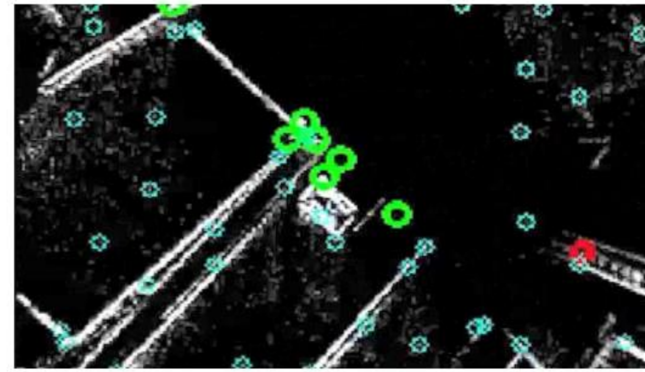
Zhang, et al., Active Exposure Control for Robust Visual Odometry in HDR Environments, ICRA'17. PDF. Video

“UltimateSLAM”: Frames + Events + IMU

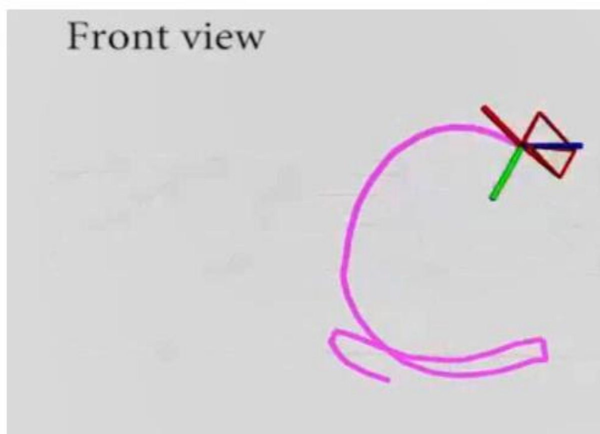
85% accuracy gain over standard visual-inertial SLAM in HDR and high speed scenes!

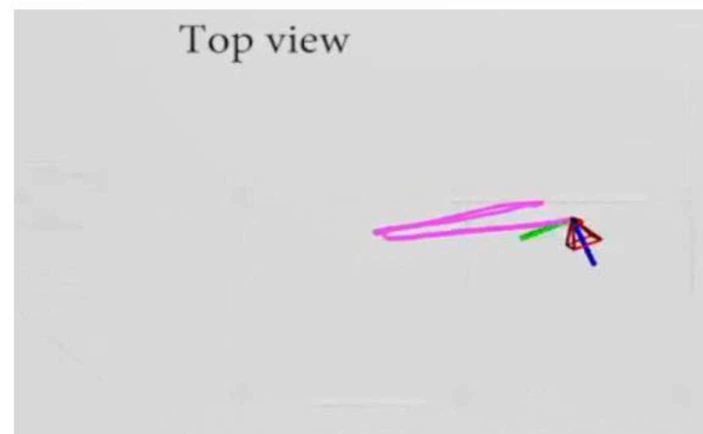
Standard camera

Event camera



Estimated trajectory





Rosinol et al., *Ultimate SLAM?* **IEEE RAL’18 Best Paper Award Honorable Mention** PDF. Video. IEEE Spectrum.
Mueggler et al., *Continuous-Time Visual-Inertial Odometry for Event Cameras*, **IEEE T-RO’18**. PDF

Conclusion

- Current SLAM evaluation
  - **Many existing metrics**, reflecting different aspects of the algorithms
  - Evaluation is a non-trivial task: **many little details affect the results**
  - Check out our **tutorial and toolbox**: https://github.com/uzh-rpg/rpg_trajectory_evaluation [Zhang, IROS'18]

- How to push forward SLAM research
  - Take robustness into consideration
  - Do not stick to a few datasets: use more diverse ones
  - Take advantage of photo **realistic simulators**, but if you do, please share the datasets!
  - Take the chance to
    - Actively change the parameters of the algorithm to improve robustness
    - Work on new sensors (e.g., event cameras)
      - Event camera dataset: http://rpg.ifi.uzh.ch/davis_data.html
      - MVSEC dataset: https://daniilidis-group.github.io/mvsec/
      - UZH-FPV Drone Racing dataset: http://rpg.ifi.uzh.ch/uzh-fpv.html
      - Event-camera Simulator (ESIM): https://github.com/uzh-rpg/rpg_esim

Checklist for Reproducible (meaningful) SLAM Results

Running experiments

- What are the crucial parameters (# features, # keyframes, etc.)?
- Does the starting and ending time in the dataset have an obvious impact on the results?
- Am I running the experiments in a real-time setup (or processing new measurements only when the previous processing is done)?
- Have I ran the algorithm multiple times to have repeatable results/meaningful statistics?

Reporting results

<u>Accuracy</u>
- Am I reporting the accuracy of real-time poses or refined poses?
- Absolute error: how is the trajectory aligned with the groundtruth?
- Which frames are evaluated? All the frame or only keyframes?

<u>Efficiency</u>
- What are the experimental platforms?
- What are the exact starting and end point of the processing time?
- Is there any special optimization used that has a big impact?

How should we report results in papers?

What not to write in a paper:

"We aligned the estimated trajectory with the groundtruth and calculated the Root Mean Square Error (RMSE) to indicate the estimation accuracy." [Author names hidden for privacy]

- What type of alignment was used?
- What method was used for calculate the alignment transformation?

How to write in a paper:

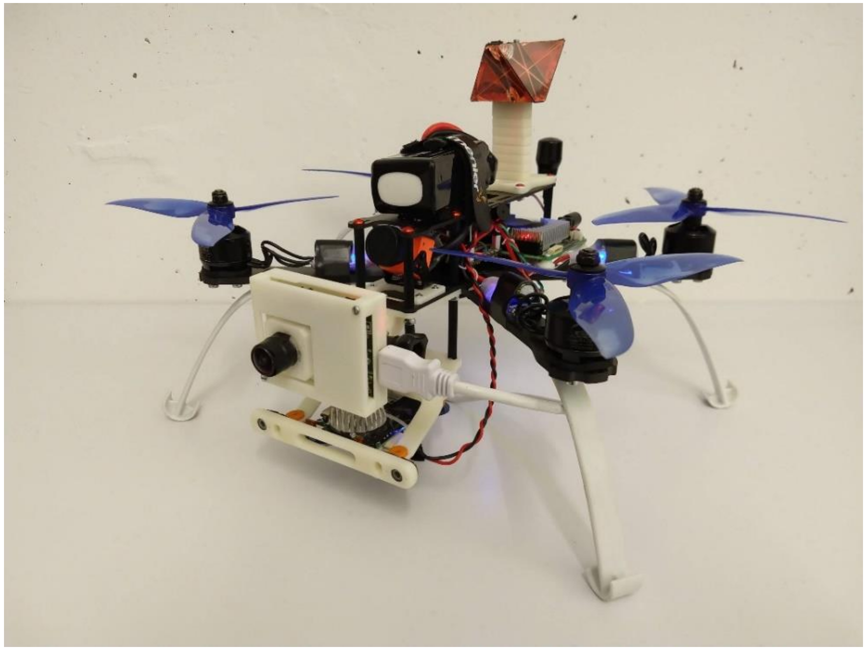
"To obtain a measure of accuracy of the different approaches, we aligned the final trajectory of keyframes with the ground-truth trajectory using the least-squares approach proposed in [Umeyama, 1991]. Since scale cannot be recovered using a single camera, we also rescaled the estimated trajectory to best fit with the ground-truth trajectory. Subsequently, we computed the Euclidean distance between the estimated and ground-truth keyframe poses and compute the mean, median, and Root Mean Square Error (RMSE) in meters." [Author names hidden for privacy]

"We used the relative error metrics proposed in [KITTI] to obtain error statistics. The metric evaluates the relative error by averaging the drift over trajectory segments of different length {10; 40; 90; 160; 250; 360 } meter." [Author names hidden for privacy]

→ Necessary references and details.

IROS 2019 FPV VIO Competition

Dataset: UZH-FPV Drone Racing Dataset



- **Aggressive motion**: First-person view (FPV) drone racing quadrotor flown by expert pilots.
- **Rich sensors**: Time-synchronized stereo/monocular standard/event cameras + inertial measurement units.

Delmerico et al, Are We Ready for Autonomous Drone Racing? The UZH-FPV Drone Racing Dataset, *ICRA2019.*

- Why another dataset?
  - Existing datasets with ground truth trajectories are slow and not aggressive.
  - VIO has become mature and handles non-aggressive situations well.

More difficult/discriminative datasets are necessary to push the state-of-the art.

Dataset: UZH-FPV Drone Racing Dataset

Outdoor

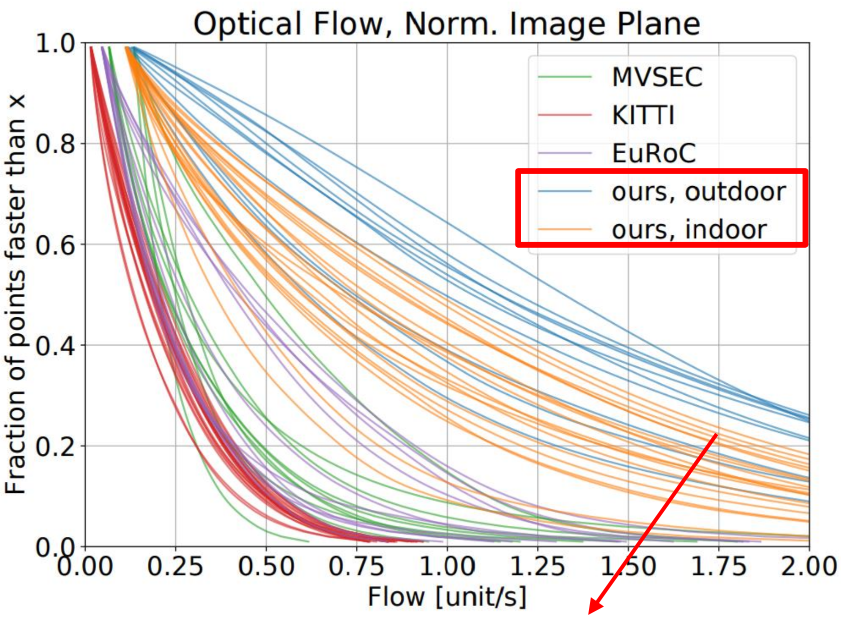
Indoor

Comparison with Existing Datasets

Our Dataset

The Most Aggressive Drone Dataset

	EuRoC MAV [17]	UPenn Fast Flight [24]	Zurich Urban MAV [19]	Blackbird [20]	**UZH-FPV Drone Racing**
Environments	2	1	3	**5**[a]	2
Sequences	11	4	1	**186**	27
Camera (Hz)	20	40	20	**120**	30/50[d] + events
IMU (Hz)	200	200	10	100	**500/1000**[e]
Motor Encoders (Hz)	n/a	n/a	n/a	**~190**	n/a
Max. Distance (m)	130.9	700	**2000**	860.8	340.1/923.5[f]
Top Speed (m/s)	2.3	17.5	3.9[b]	7.0	**12.8/23.4**[g]
mm Ground Truth (Hz)	20/100[c]	n/a	n/a	**360**	20



Highest optical flow

Challenges in the dataset:

- High flight speed
- High optical flow

The dataset contain various sensors (both conventional and novel sensors), providing different possibilities to deal with these challenges.

The 1st FPV Drone Racing VIO competition

- 6 sequences (no public groundtruth) from the UZH-FPV datasets

Dataset	Length	V_{max}	Difficulty
indoor-forward-11	85.68 m	10.32 m/s	Easy
indoor-45-3	119.82 m	3.53 m/s	Easy
outdoor-forward-9	314.41 m	10.68 m/s	Medium
outdoor-forward-10	455.63 m	12.58 m/s	Medium
indoor-forward-12	124.07 m	15.28 m/s	Hard
indoor-45-16	58.72 m	7.69 m/s	Hard

https://github.com/uzh-rpg/IROS2019-FPV-VIO-Competition

The Participants

- We received 5 submissions, 3 of which agreed to disclose their submission information.
  - **Patrick Geneva**, Robot Perception and Navigation group, University of Delaware.
  - **Thomas Mörwald**, Leica Geosystems.
  - **Vladyslav Usenko**, Computer Vision Group, Technical University of Munich.

- The reports and links to open source code are publicly available with the consent of the participants.

Competition Results

- Evaluation: the relative pose error as in KITTI
  - Average relative pose error over sub-trajectory lengths of 40, 60, 80, 100, 120 meters.

Ranking	Name	Sensors	Trans. Error (%)	Rot. Error (deg/m)
1	Patrick Geneva	binocular; inertial	7.023	0.264
2	Thomas Mörwald	monocular; inertial	7.034	0.266
3	Vladyslav Usenko	stereo; Inertial	7.778	0.285
4	a-u	stereo; inertial	11.869	0.619
5	r-u	stereo; inertial	36.048	1.894

Detailed results available at http://rpg.ifi.uzh.ch/uzh-fpv.html

The Winner: Patrick Geneva

- OpenVINS (https://github.com/rpng/open_vins)
  - Sensors: Binocular
    - Stereo matching is not used due to the poor matching performance
  - Frontend: Optical flow
    - FAST detector
    - Lucas-Kanade optical flow (OpenCV implementation)
  - Backend: MSCKF
    - Siding window of 15 frames
  - Loop closing: No
- Hardware/Processing Time
  - E3-1505M @ 3.00GHz: ~ 1.5 x real-time

The Runner-Up: Thomas Mörwald

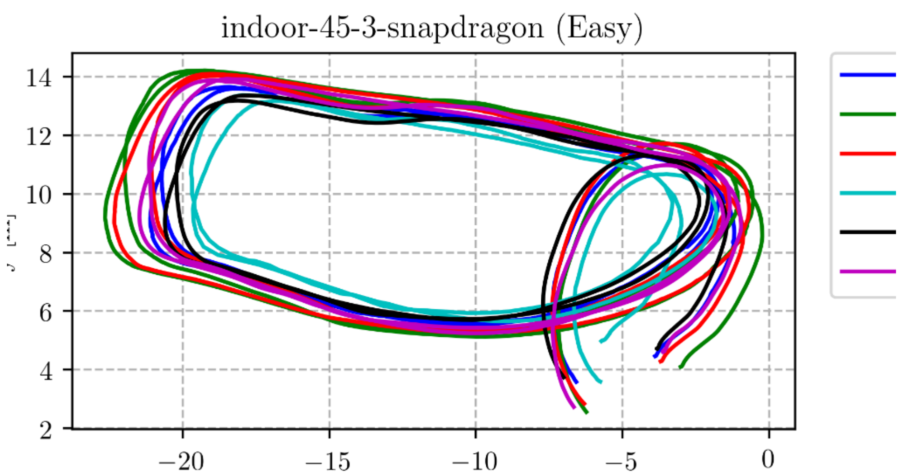
- Optimization-based VIO
  - Sensors: Monocular
  - Frontend: Optical flow
    - Shi-Tomasi Detector (OpenCV implementation)
    - Lucas-Kanade optical flow
  - Backend: Fixed-lag optimization (GTSAM iSAM2)
    - Sliding window size: 0.5 second (= 15 frames)
  - Loop closing: No

- Hardware/Processing Time
  - i7-8650U CPU @ 1.9 GHz: ~ 1.3 x real-time
  - Most time consuming: Backend optimization > Detection > Optical flow

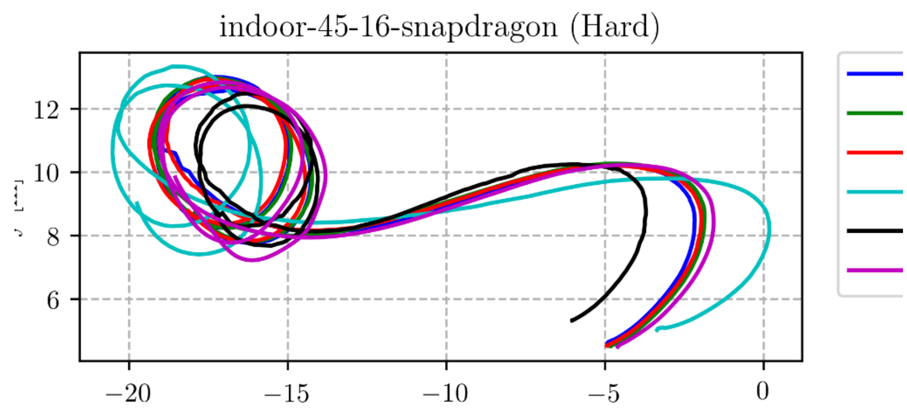
Conclusion and Outlook

- “Worse” performance than existing datasets
  - Best **7 %** vs. commonly seen **1 %** translation error (e.g., EuRoC)
- None of the participants utilized the event camera.

The UZH-FPV dataset is far from saturated compared to existing ones. New algorithms, possibly combined with novel sensing modalities, can potentially push the performance.



Easy: OK tracking



Hard: erroneous tracking