

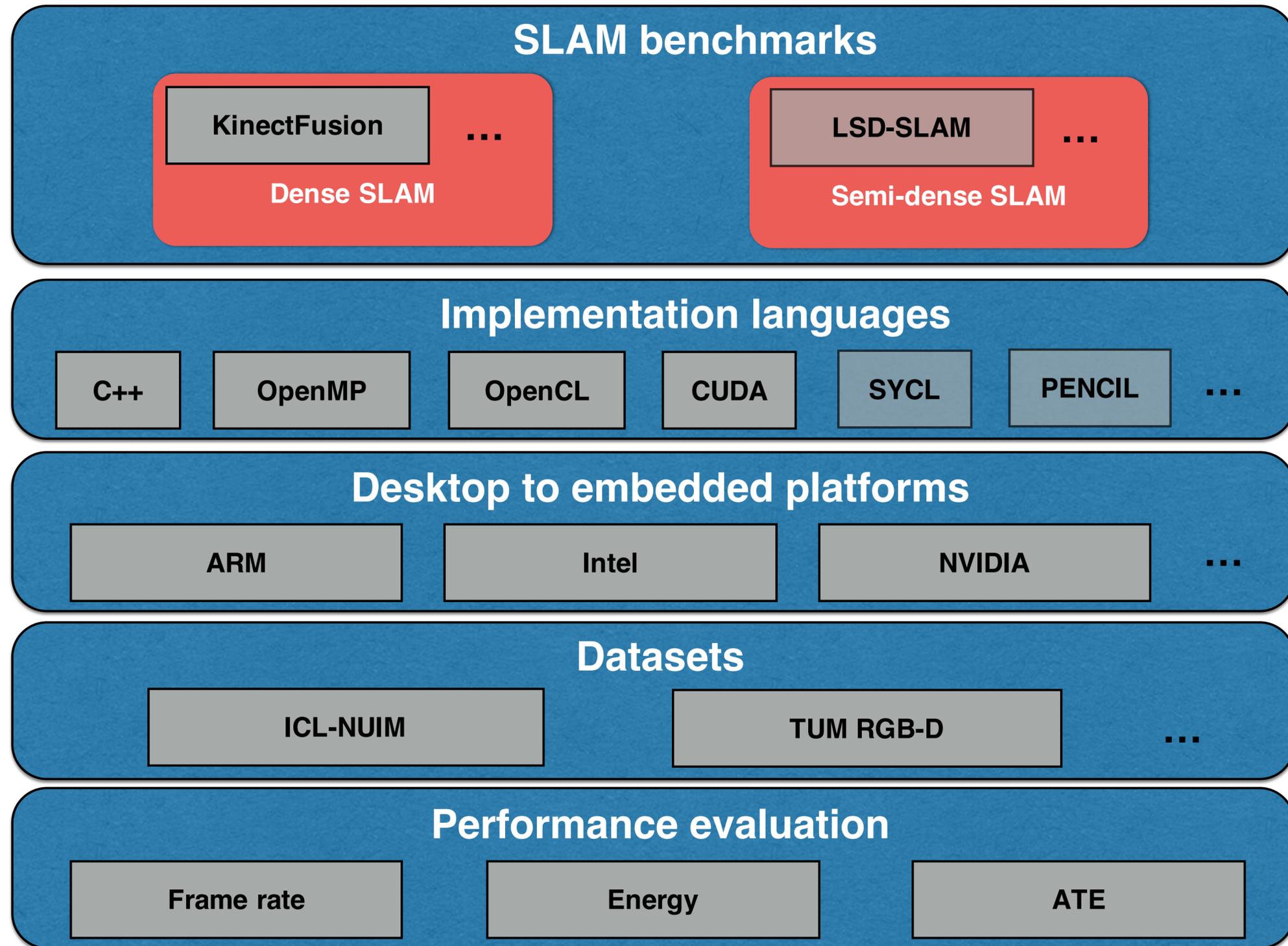
Comparative design space exploration of dense and semi-dense SLAM

Imperial College
London

Z. Zia, L. Nardi, A. Jack, E. Vespa, B. Bodin, P. H. J. Kelly, A. J. Davison

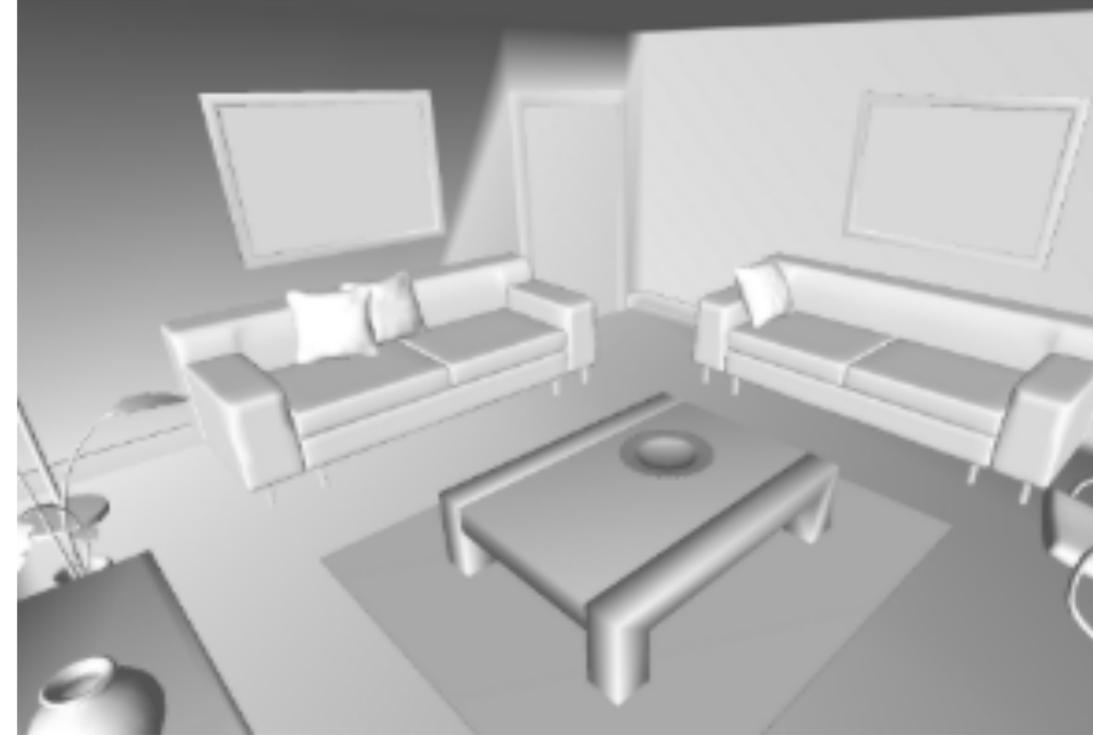


- ▶ Enabling end-to-end quantitative and reproducible benchmarking of SLAM pipelines
- ▶ SLAM as a multi-objective optimisation problem
 - Absolute Trajectory Error (ATE)
 - Relative Pose Error (RPE)
 - Frame rate
 - Energy per frame
 - Reconstruction accuracy (coming...)



- ▶ ICL-NUIM synthetic indoor scenes:
 - Living room synthetic environment
 - Human generated trajectories
 - Trajectory and world model ground-truth
 - A. Handa *et al.* A Benchmark for RGB-D Visual Odometry, 3D Reconstruction and SLAM.

- ▶ TUM real RGB-D dataset:
 - Handheld camera sequence plus trajectory ground-truth
 - J. Sturm *et al.* A benchmark for the evaluation of RGB-D SLAM systems

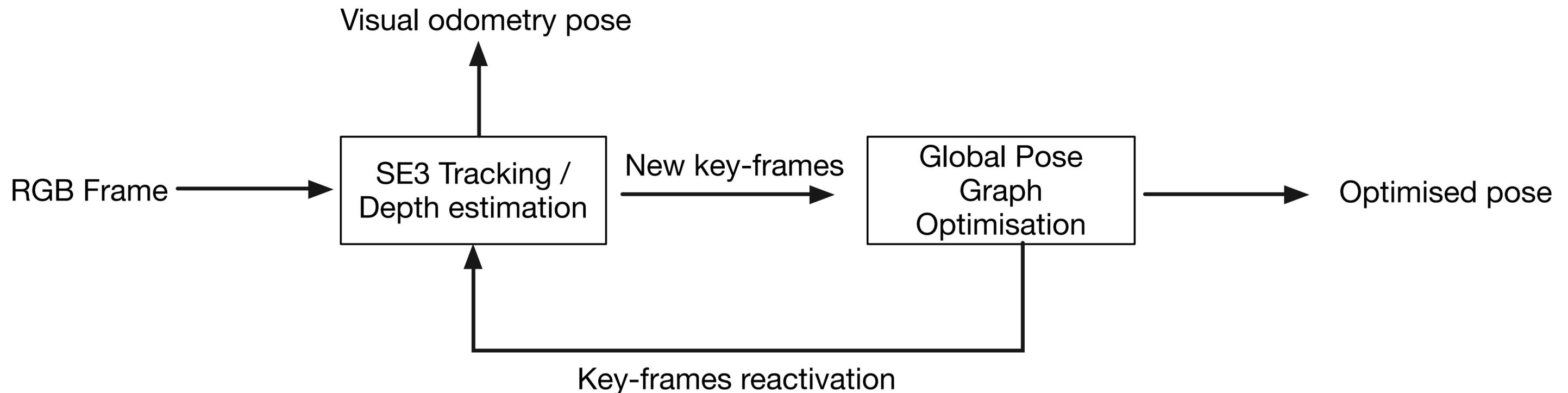


ICL-NUIM living room <http://www.doc.ic.ac.uk/~ahanda/VaFRIC/iclnuim.html>



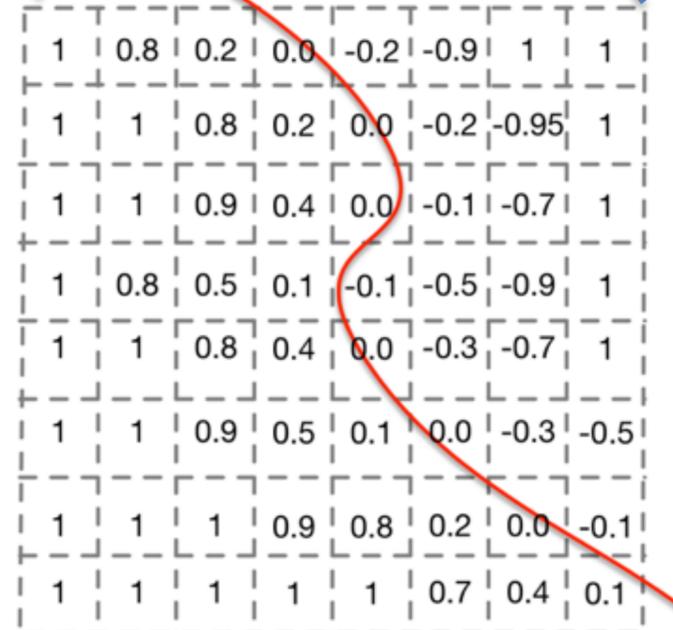
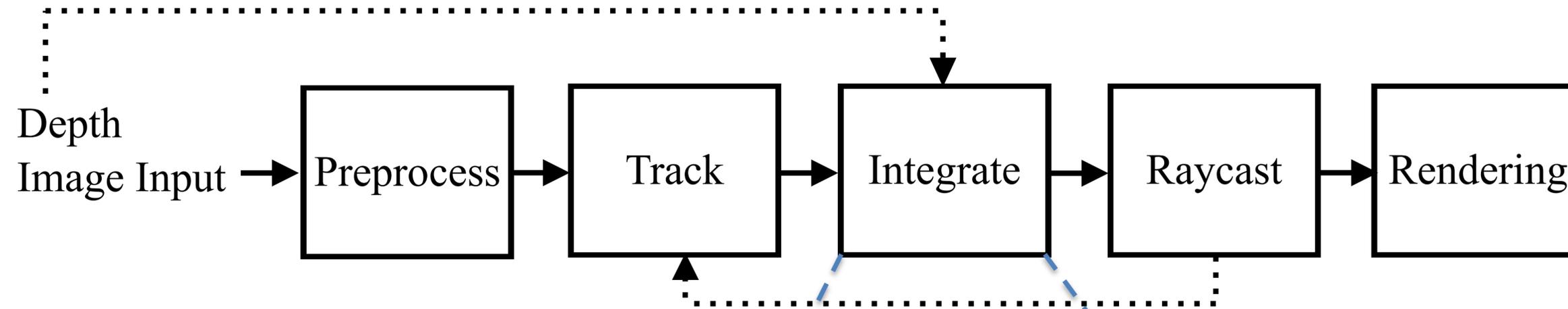
Screenshot from TUM fr2/desk sequence

- ▶ Semi-dense tracking and mapping front-end
 - Track new images against high gradient patches of reference key-frame
 - Estimate and refine depth for such patches
 - When current frame too far, finalise key-frame and initialise a new one
- ▶ Pose graph optimisation on back-end
 - Loop closure detection
 - Global optimisation



Thread name	Major kernels	Description	Pattern	Percent
Tracking (vectorized)	<i>Calc. Residuals</i>	} Calculate components of the Levenberg–Marquardt (LM) algorithm Evaluate the LM algorithm given the above calculations	Map	72%
	<i>Calc. Weights and Residuals</i>		Map	4%
	<i>Calc. Jacobians</i>		Map-Reduce	9%
	<i>Solve</i>		External	0%
<i>Total</i>				34 s
Depth	<i>Stereo Line Search</i>	Epipolar line search	Map	43%
	<i>Fill Holes</i>	Increase density of depth map	Stencil	20%
	<i>Regularize Depth Map</i>	Denoise the depth map	Stencil	28%
	<i>Copy Depth Map to Frame</i>	Implementation specific overhead	Map	6%
<i>Total</i>				48 s
Constraint Search	<i>Find Euclidean Overlaps</i>	Get neighbour frames from graph, to insert new constraints	Search	6%
	<i>Filter and Sorting</i>	Remove less optimal frames from results	Map	4%
	<i>Calc. Residuals</i>	} Calculate components of the Levenberg—Marquardt (LM) algorithm between keyframe and neighbour frames	Map	71%
	<i>Calc. Weights and Residuals</i>		Reduce	7%
<i>Calc. Jacobian Matrix</i>	External		12%	
<i>Total</i>				19 s
Optimization	<i>g2o Call</i>	Run iterations of global optimization	External	99%
	<i>Update Graph</i>	Incorporate improvements from g2o into graph	Map	1%
<i>Total</i>				3 s

- ▶ Dense geometry estimation encoded in a *truncated signed-distance function* (TSDF)
- ▶ Dense tracking via frame-to-model alignment: synthetic point cloud obtained by ray-casting the TSDF



Truncated signed-distance function

The red line shows the zero iso-surface representing the best estimate of the observed surface

Major kernels	Block	Pattern	Percent
<i>Convert mm to meters</i>	Preprocess	Gather	0%
<i>Bilateral Filter</i>	Preprocess	Stencil	4%
<i>Half Sample</i>	Track	Stencil	0%
<i>Depth to Vertex</i>		Map	0%
<i>Vertex to Normal</i>		Stencil	0%
<i>Track</i>		Map/Gather	2%
<i>Reduce</i>		Reduction	2%
<i>Solve</i>		Sequential	0%
<i>Integrate</i>	Integrate	Map/Gather	73%
<i>Raycast</i>	Raycast	Search/Stencil	17%

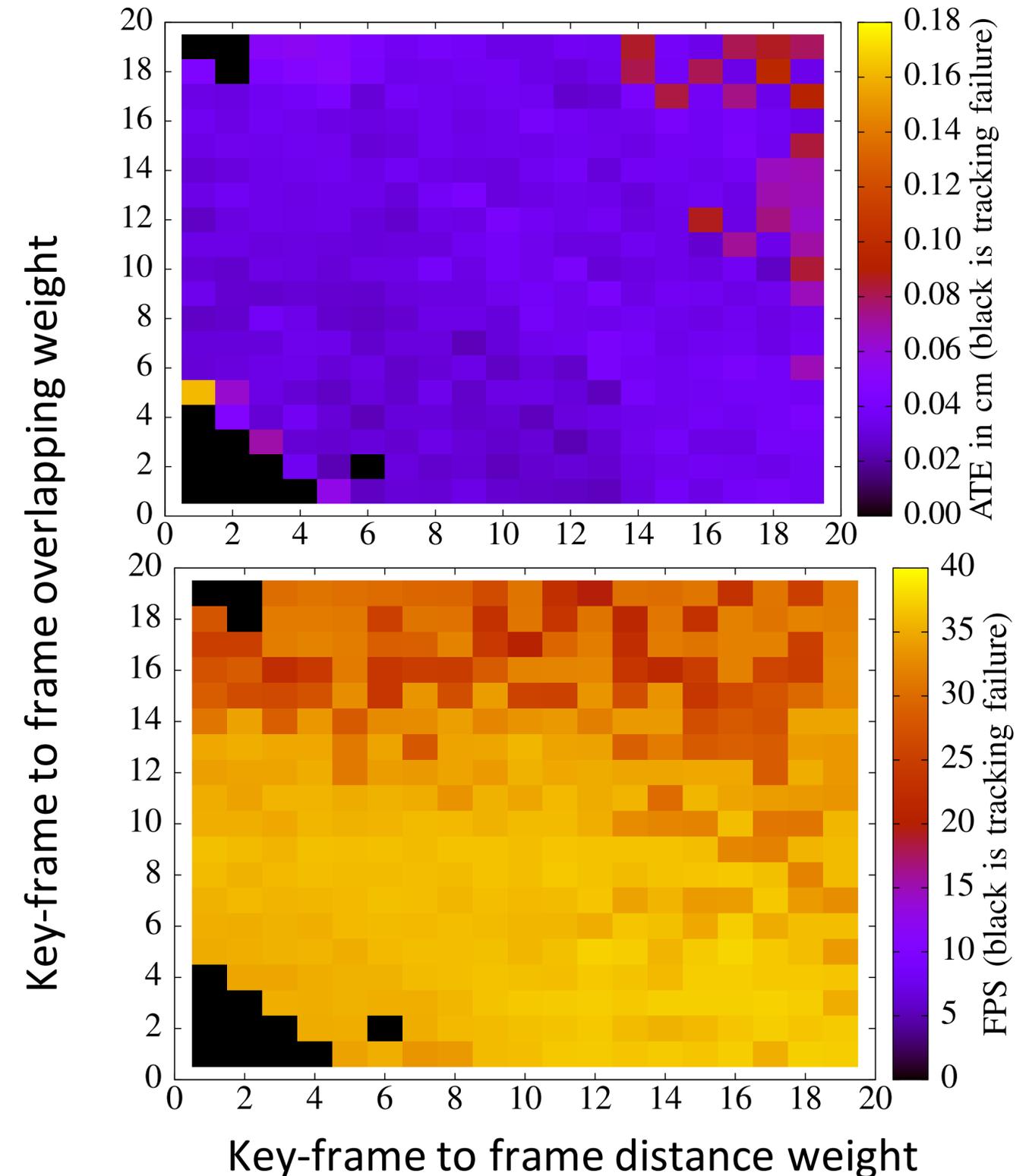
▶ LSD-SLAM:

- Number of key-frames on ATE and frame-rate
- Depth map density
- Hardware characteristics (frequency + number/type of cores)

▶ KinectFusion:

- ATE versus voxel size
- Frame-rate versus voxel size
- Hardware characteristics (frequency + number/type of cores)

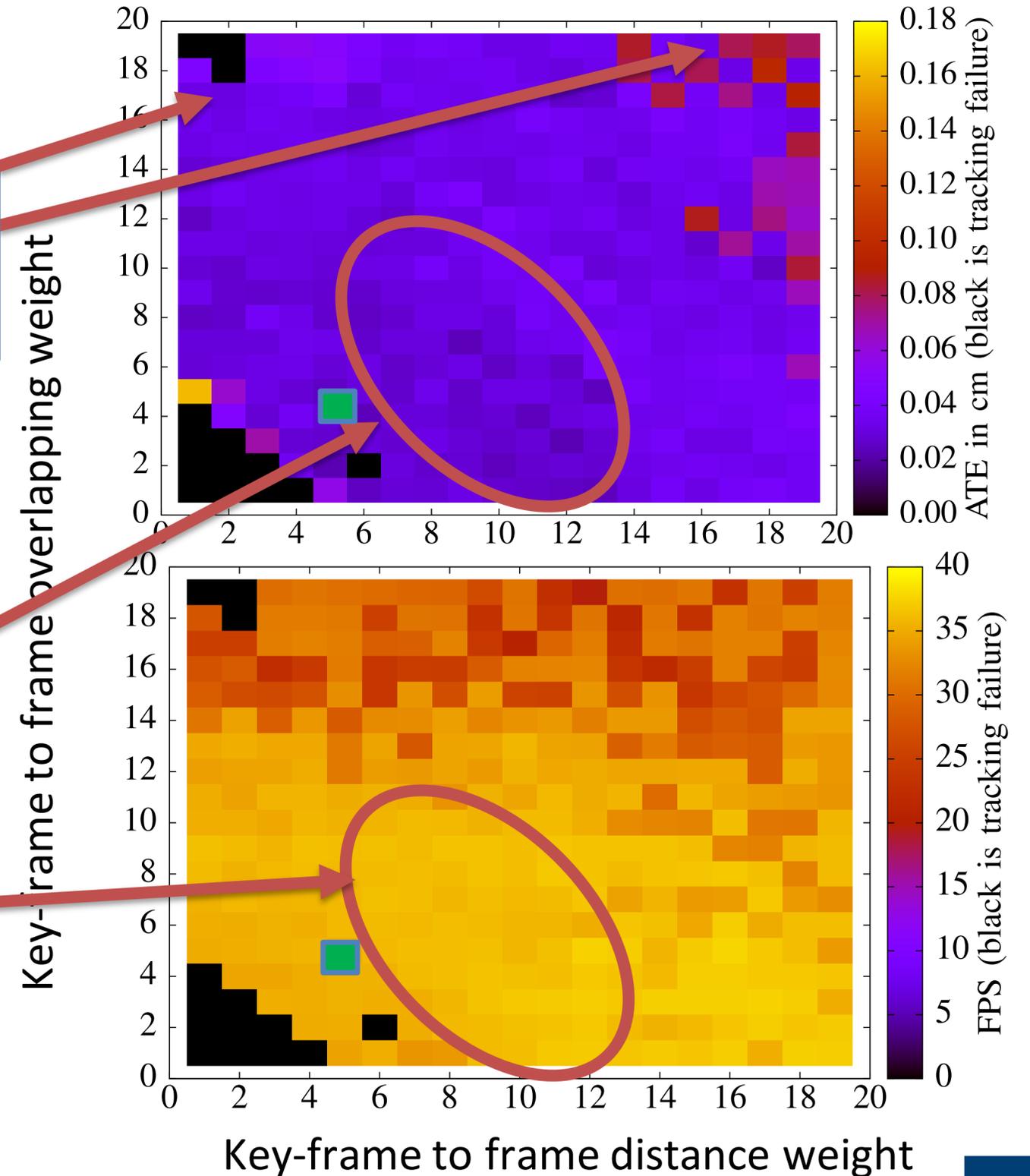
- ▶ Impact of key-frames number on ATE and frame-rate.
- ▶ Parametric weights that dictate how often new key-frames are created:
 - X axis: weight assigned to Euclidean distance between current frame and reference key-frame
 - Y axis: weight assigned to current frame and reference key-frame overlapping
- ▶ Higher values imply more key-frames.
- ▶ Black regions represent configurations that make the algorithm lose track



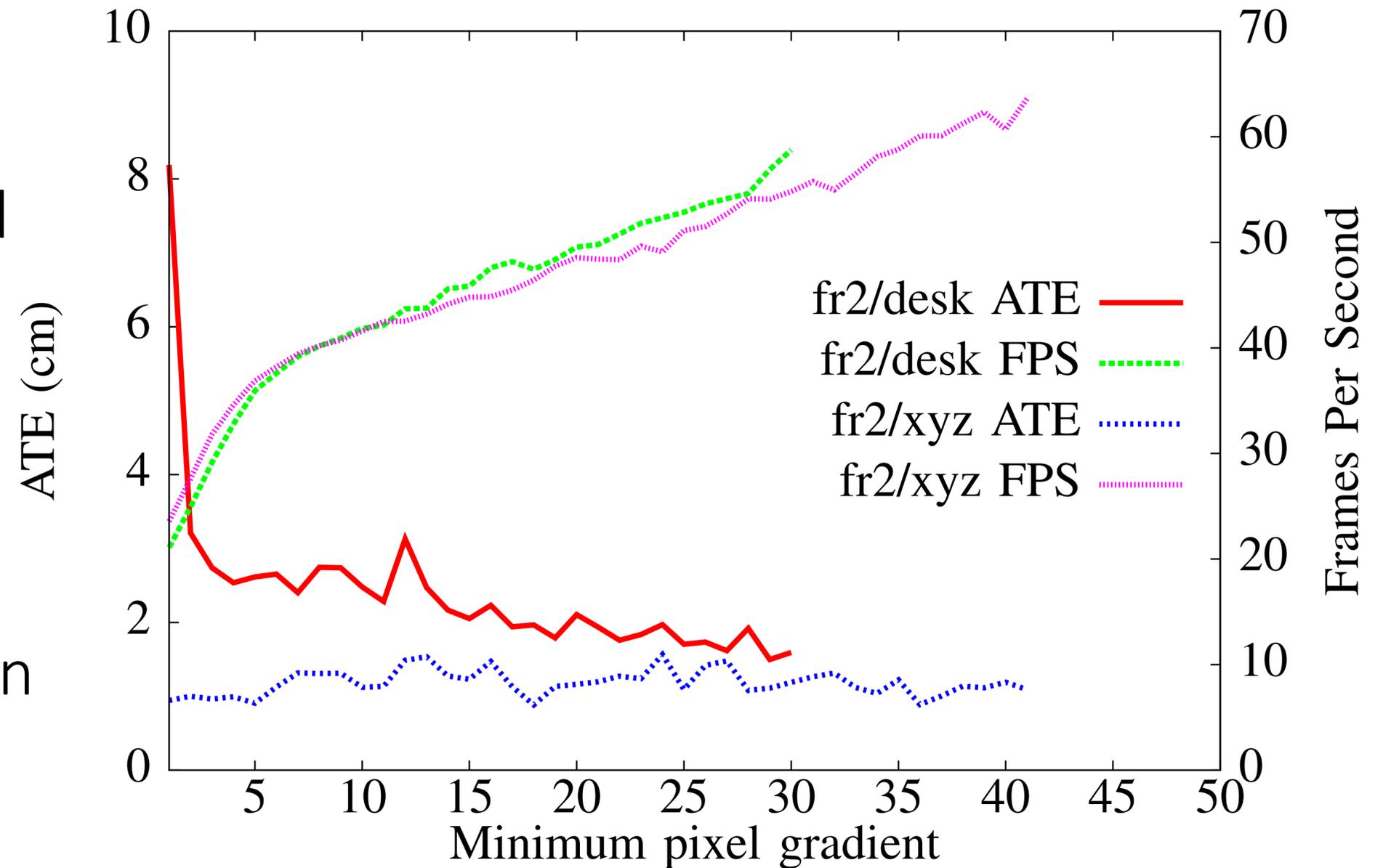
- ▶ Impact of key-frames number on ATE and frame-rate
- ▶ Parametric weights:
 - Euclidean distance
 - Frame to key-frame overlapping
- ▶ Default configuration ■

Too many, poor depth refinements imply bad ATE

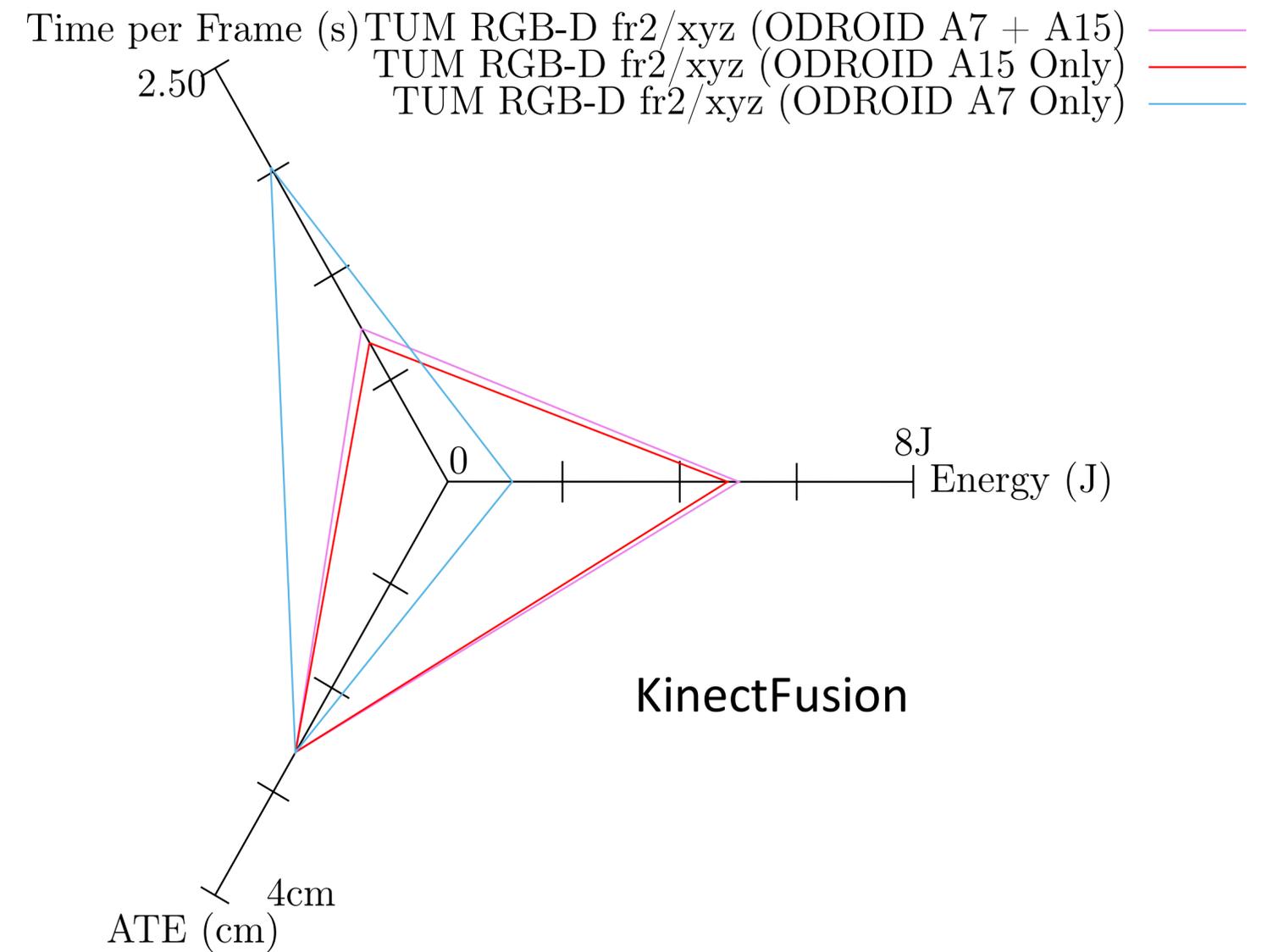
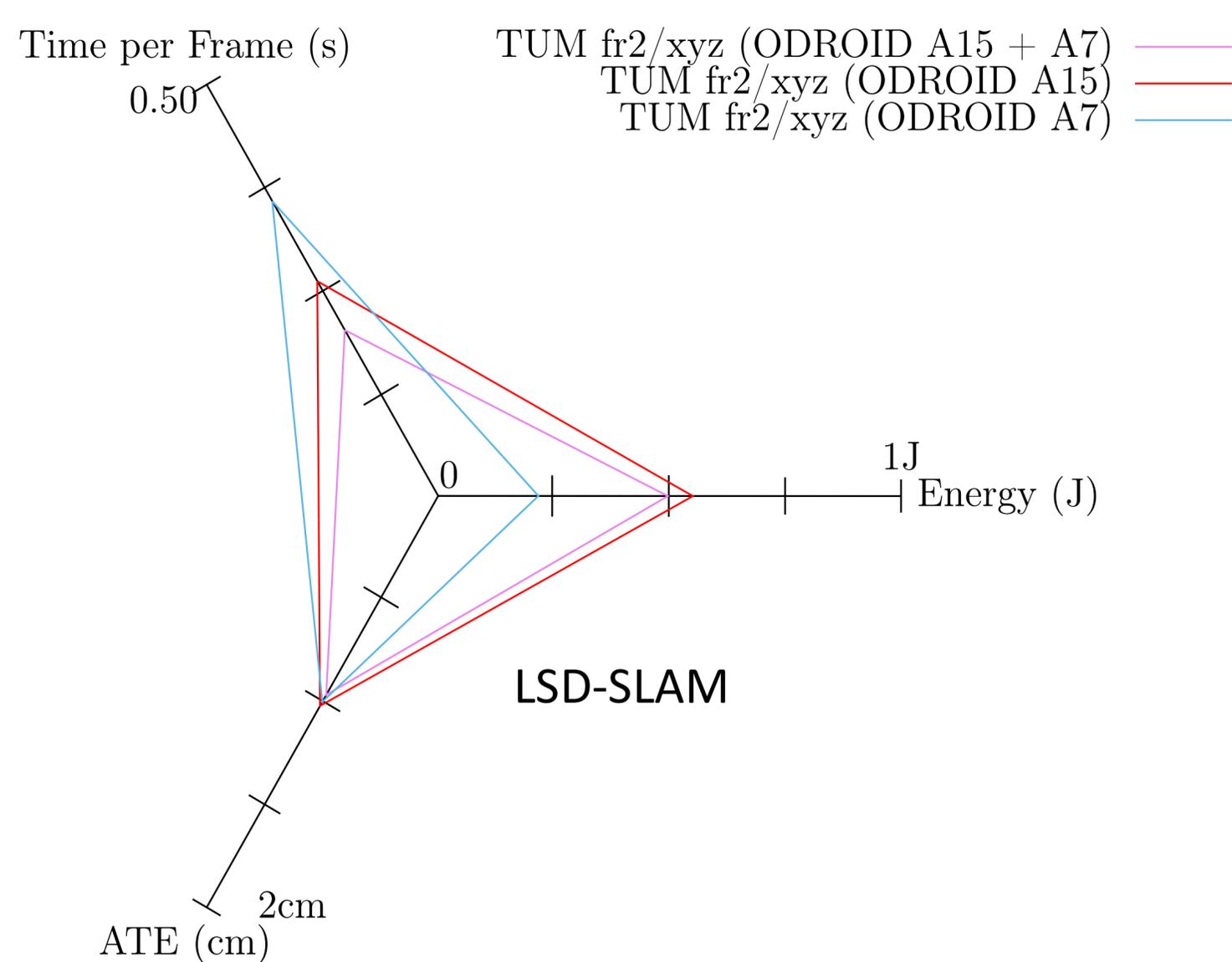
Regions where you attain best ATE and frame-rate



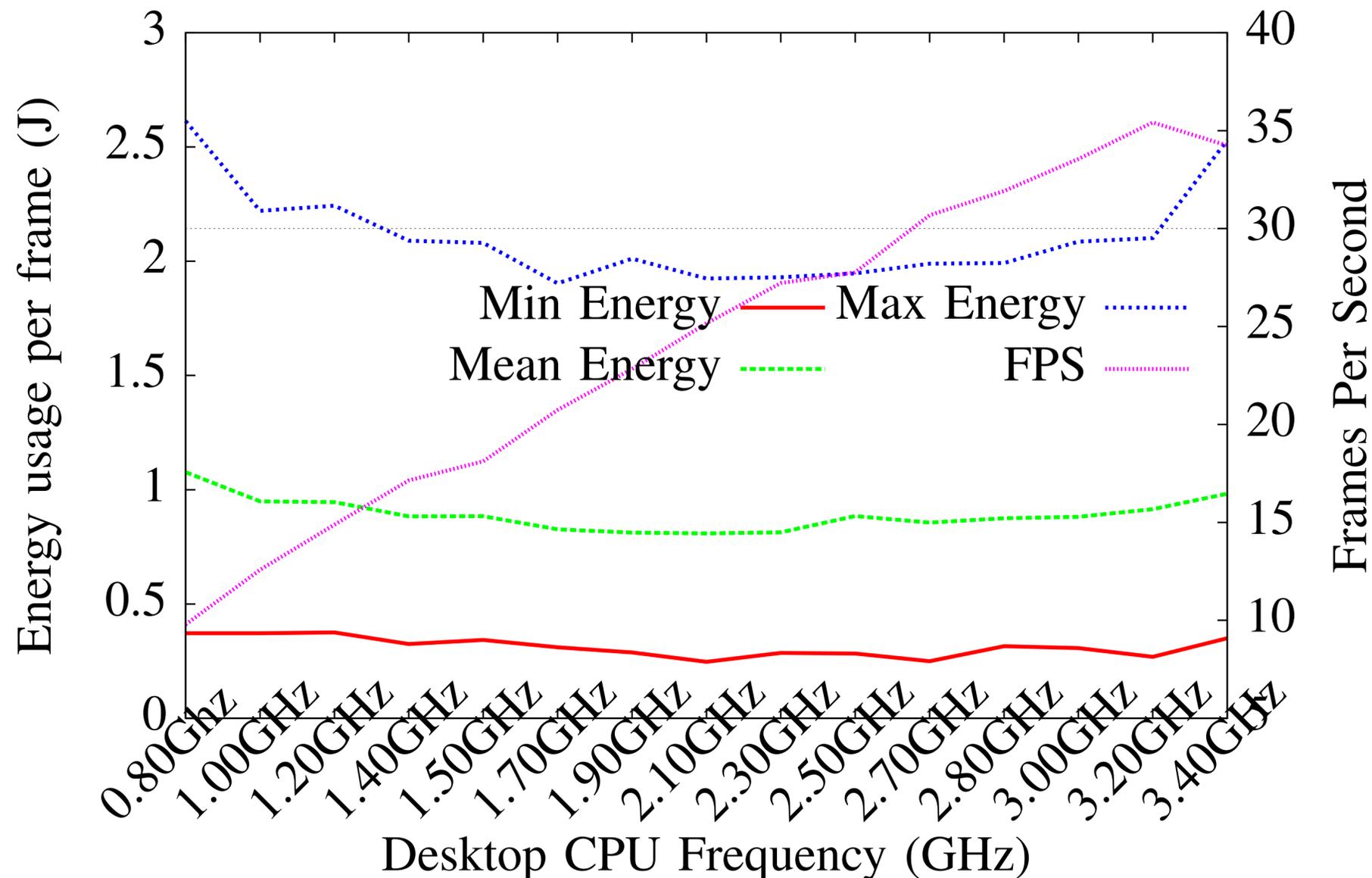
- ▶ LSD-SLAM depth estimation
 - Higher gradient threshold implies less pixels selected for tracking and mapping
 - Higher frame-rate given from the reduced number of epipolar searches
- ▶ Accuracy heavily depends on sequence.



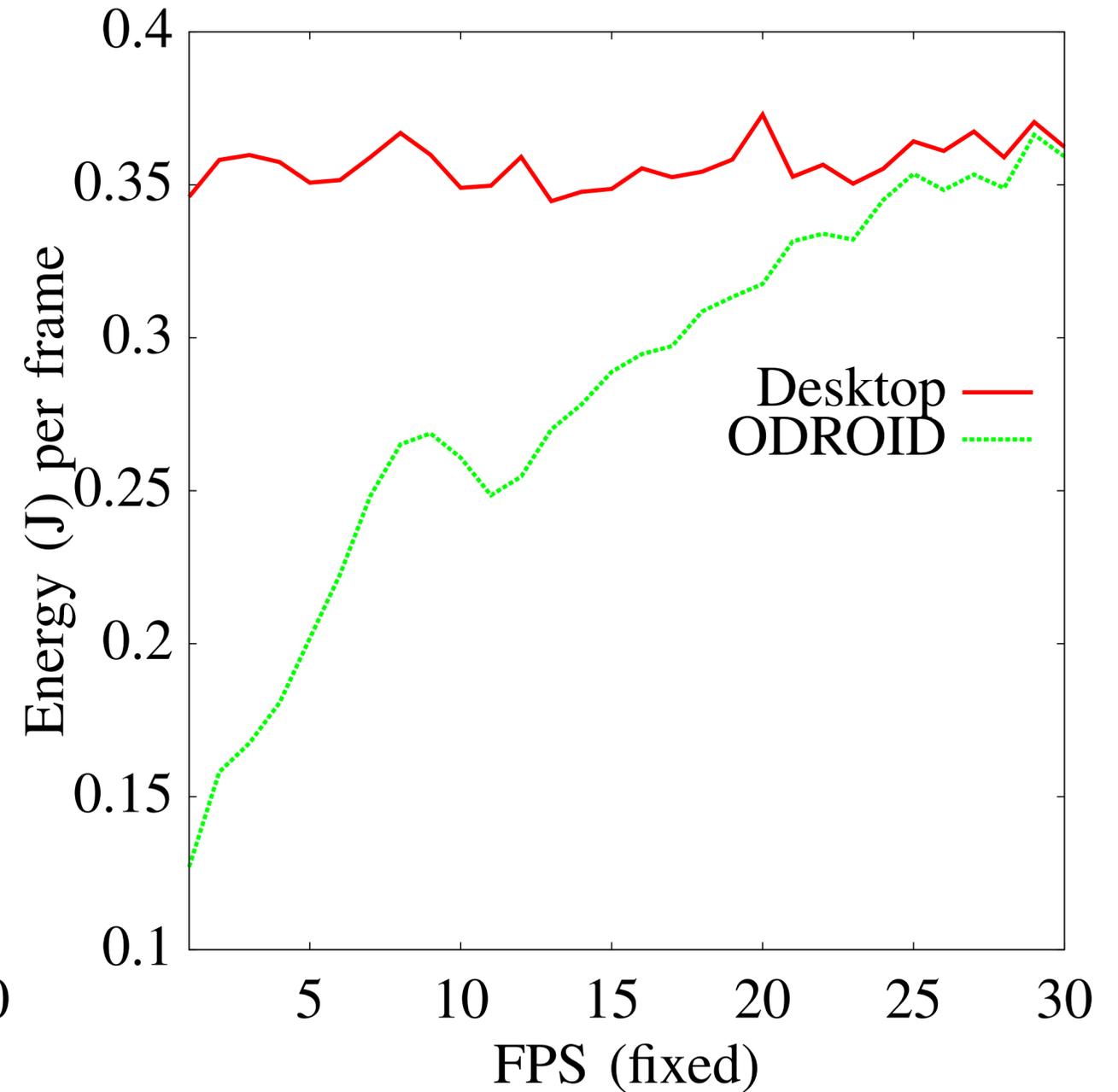
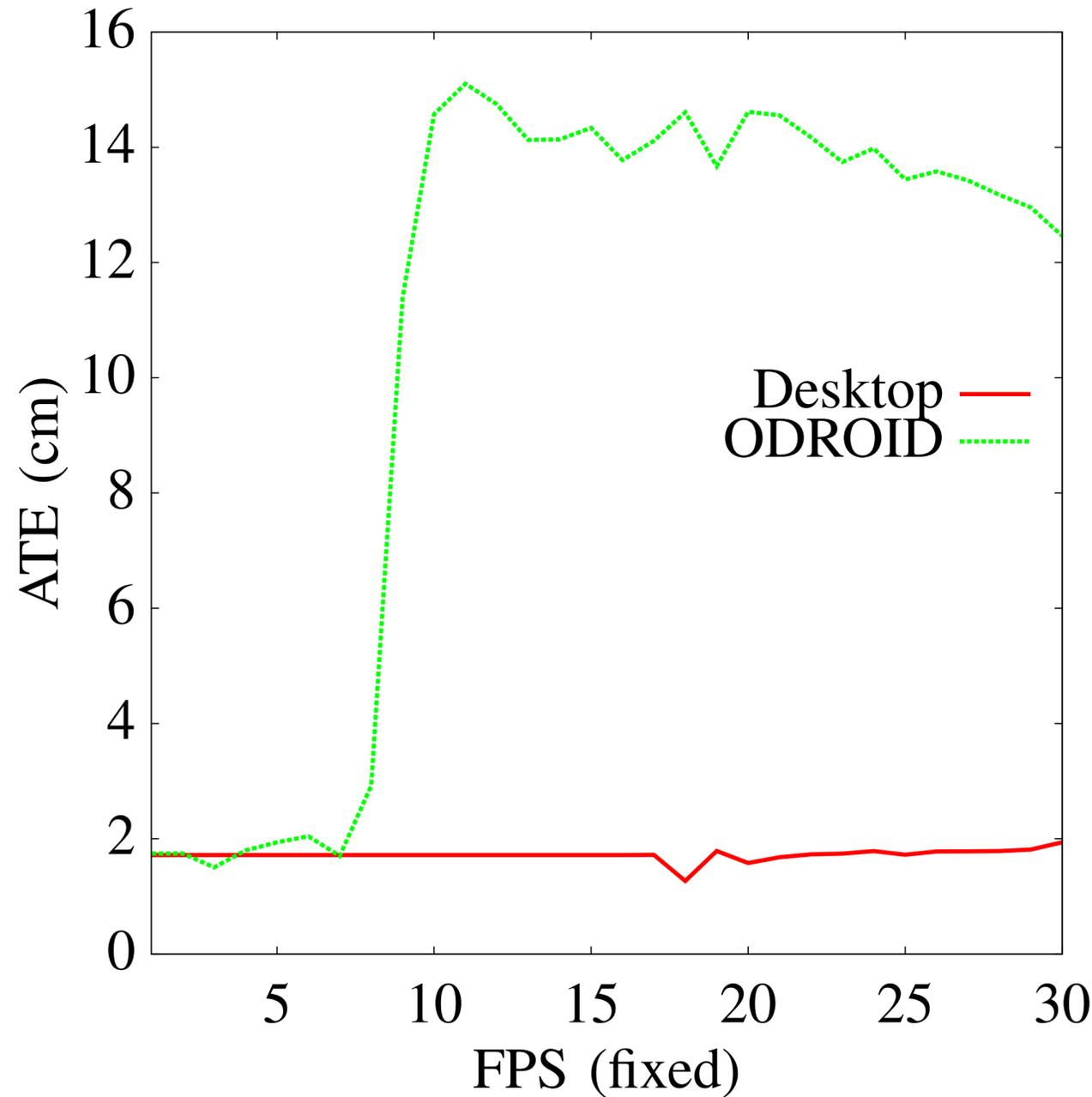
- ▶ Hardware configurations exploration on the ODROID board
 - ARM big.LITTLE architecture: 4 A7 + 4 A15 cores
 - Holistic comparison varying the number of cores



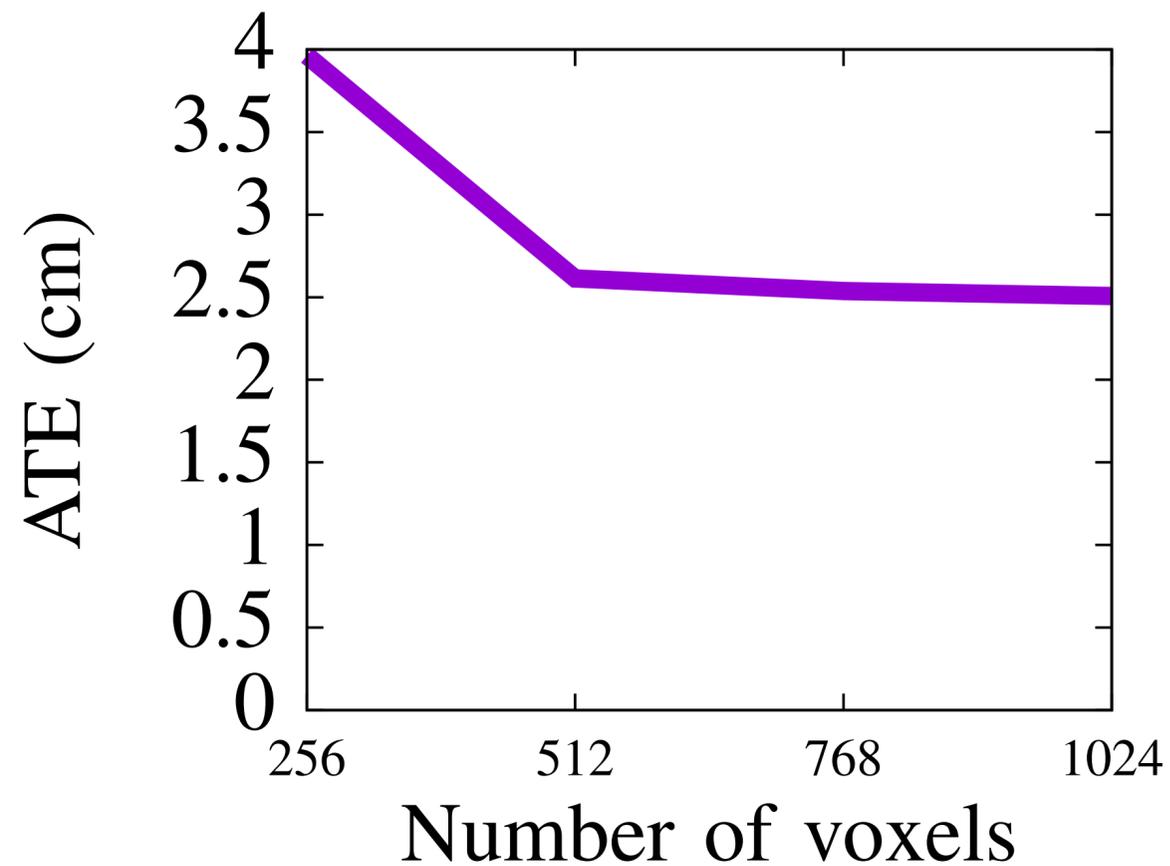
- ▶ Frequency scaling on a Haswell i7-4770 desktop processor
 - Mean energy per frame stays constant, frame rate increases sub-linearly



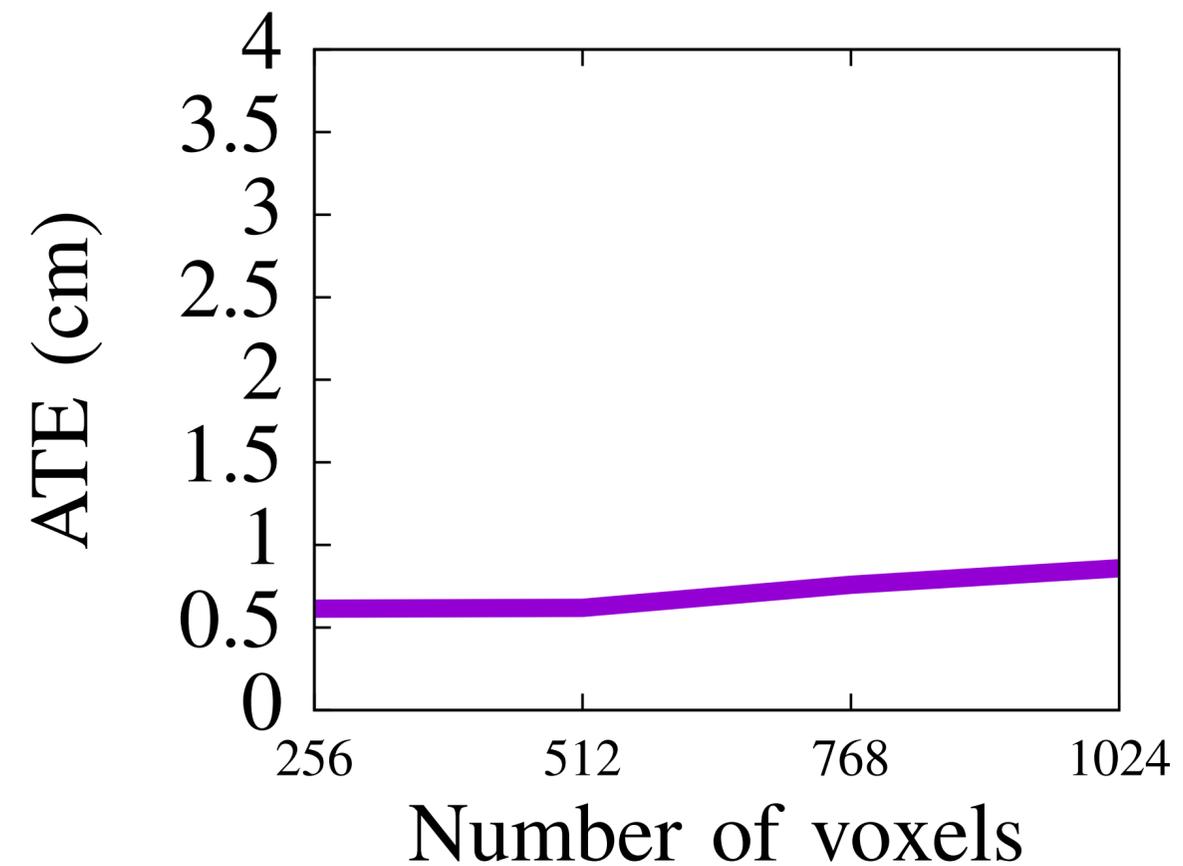
- ▶ Removing the constraining of process – every-frame mode
 - Fix frame rate - frames can be dropped
- ▶ Test platforms:
 - ODROID board (A7 + A15 ARM cores)
 - Desktop: Intel Haswell i7 4770
- ▶ Interesting impact of frame-rate on LSD-SLAM accuracy on the ODROID board
 - Frame dropping considerably impacts tracking accuracy



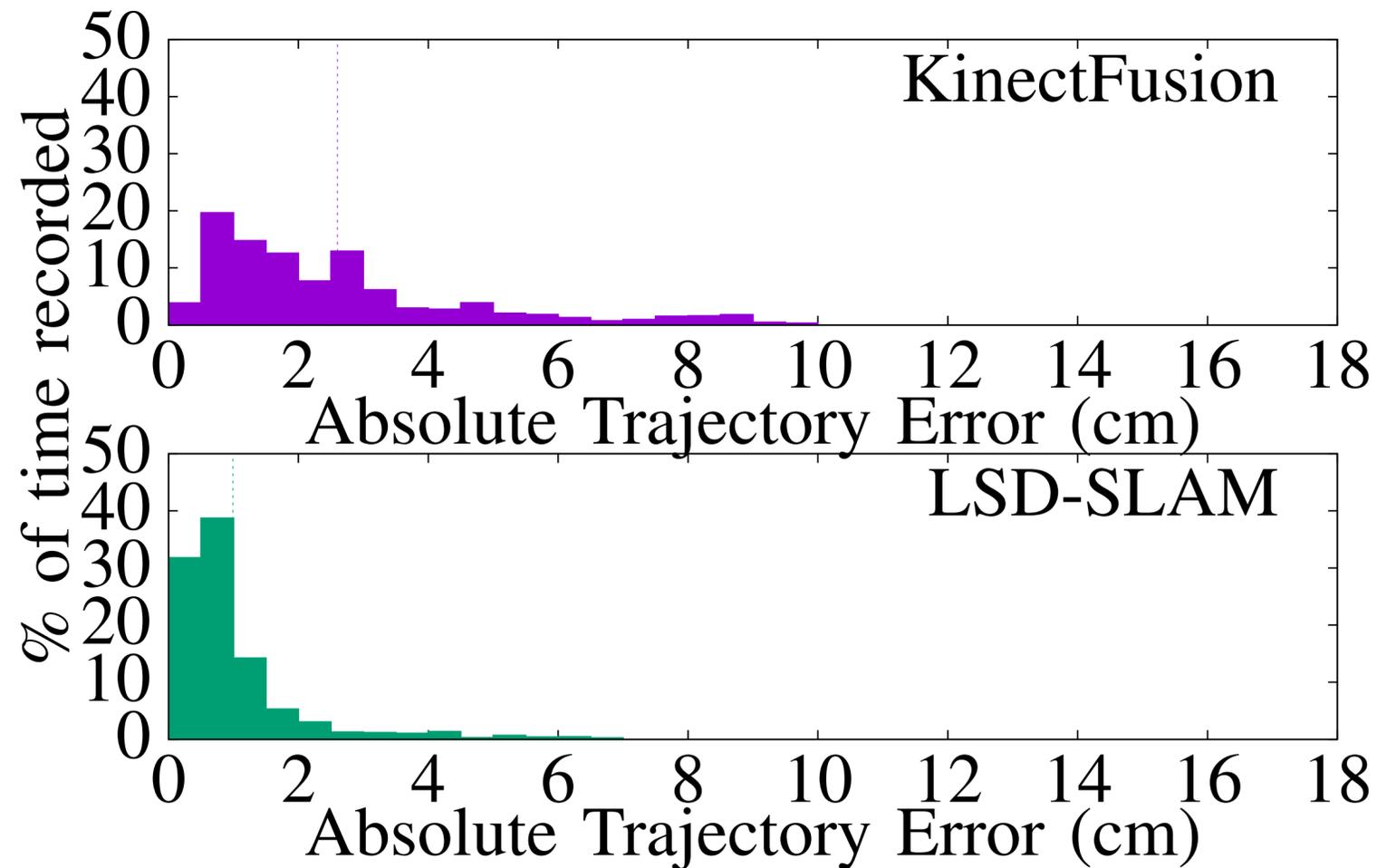
- ▶ Scaling up the resolution does not always imply a better accuracy
 - Coarser voxels might have a noise smoothing effect leading to better tracking



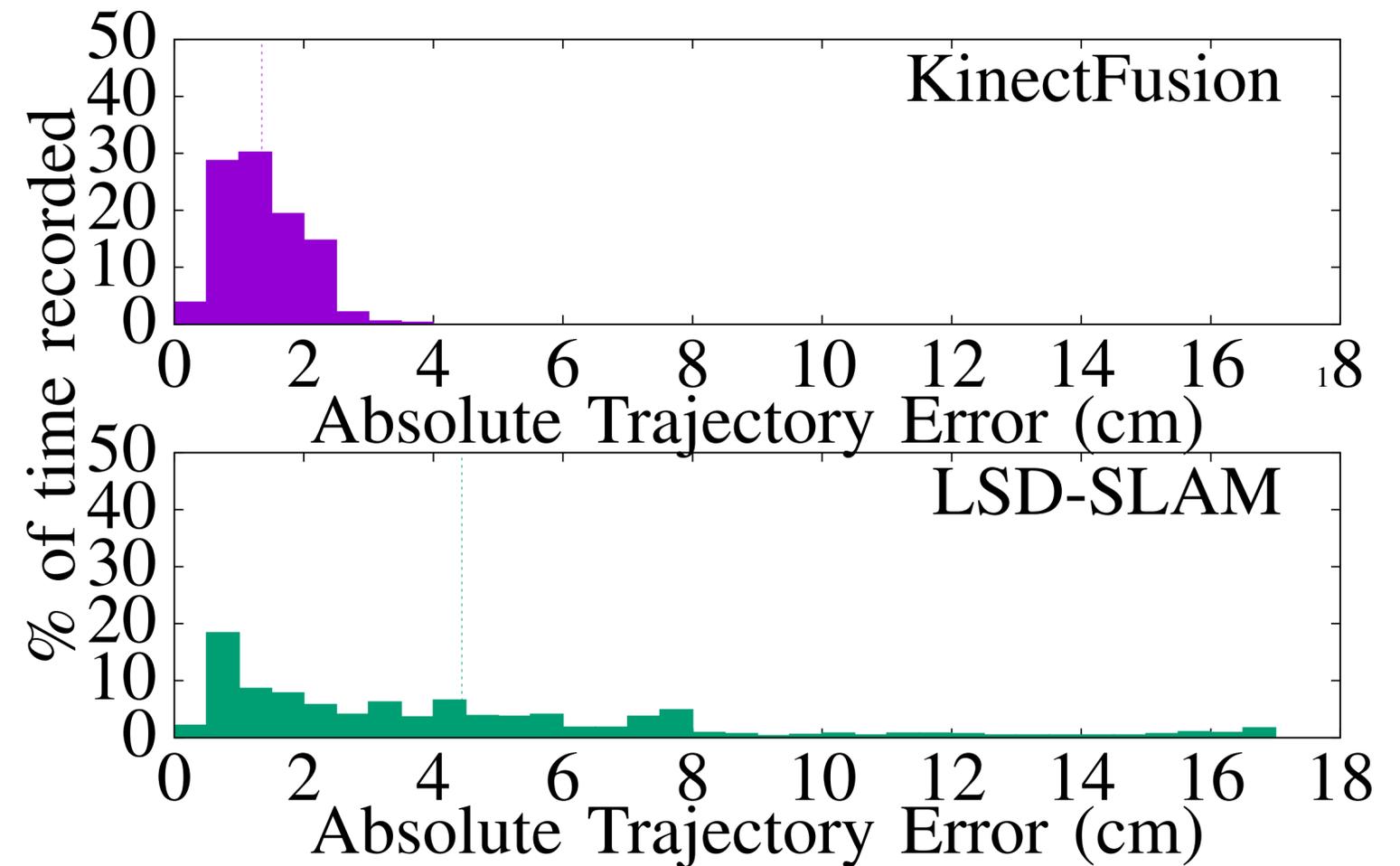
(a) Real Scene



(b) Synthetic Scene



(a) Real Scene

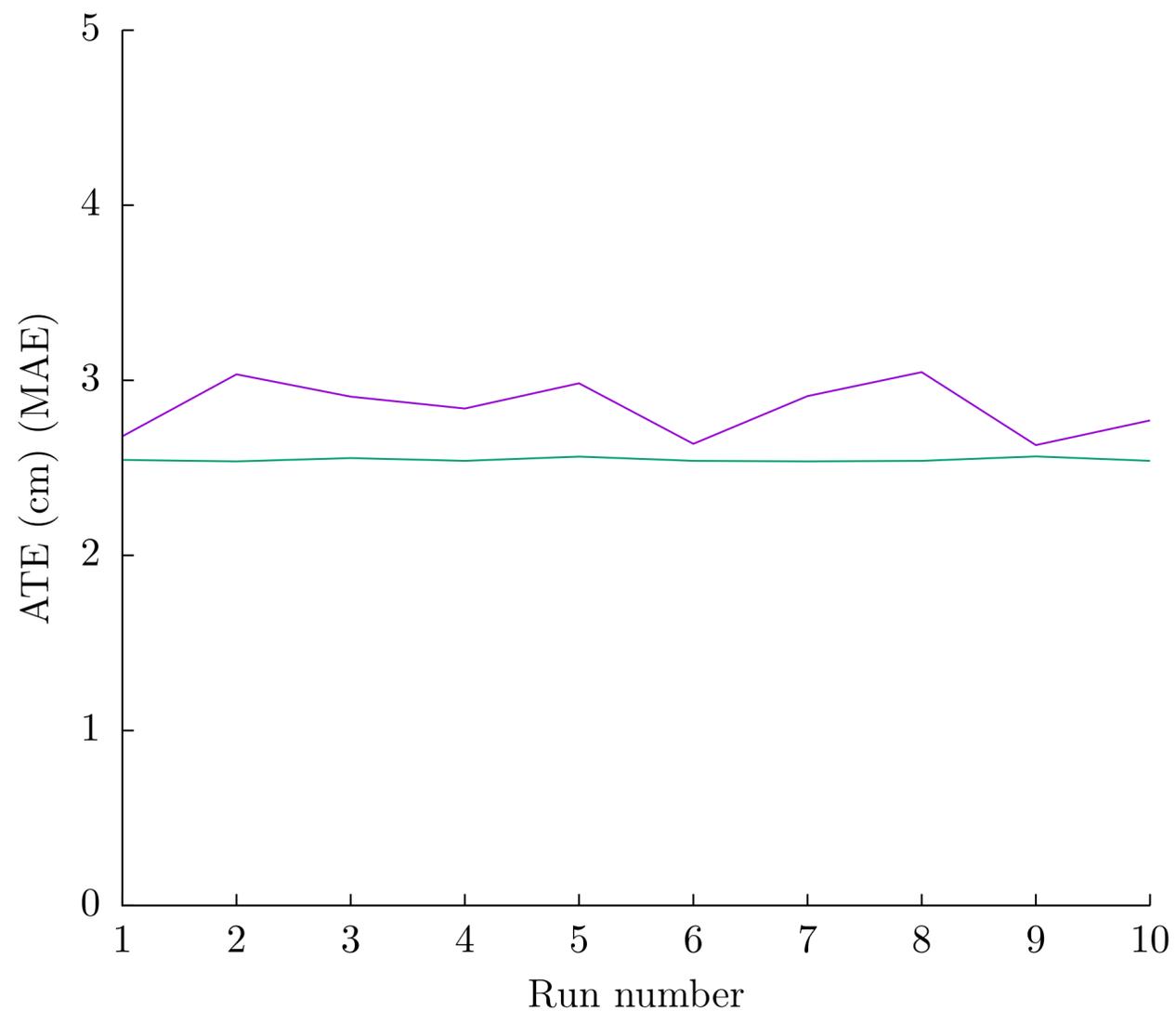


(b) Synthetic Scene

- ▶ Comparing KinectFusion and LSD-SLAM
 - Absolute trajectory error distribution over entire trajectory
 - Real scene vs synthetic scene
 - LSD-SLAM possibly affected by lack of realism in synthetic RGB data

A note on result reproducibility

- ▶ In this work we enforced *process-every-frame* mode for reproducibility purposes
- ▶ LSD-SLAM exposed significant fluctuations across repeated executions



- ▶ We would like to thank EPSRC for funding this research, PAMELA grant EP/K008730.
- ▶ Jacob Engel for useful discussion and feedback on LSD-SLAM
- ▶ Andy Nisbet and John Mawyer of University of Manchester for their contributions to the SLAMBench framework