Local and Global Mapping for Direct SLAM

Erkam Uyanik

Master's Thesis in Informatics

Advisor: Nikolaus Demmel, M.Sc.

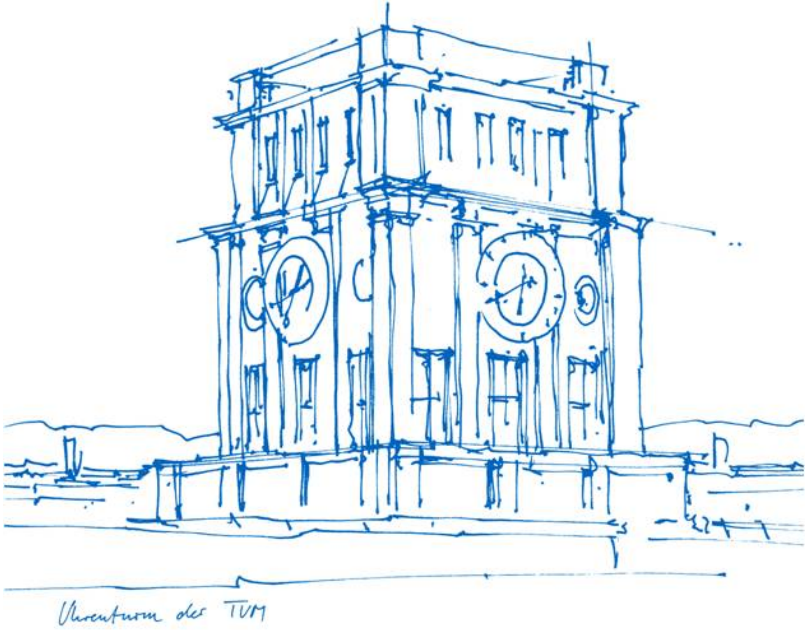
Supervisor: Prof. Dr. Daniel Cremers

Technical University of Munich

Department of Informatics

Chair of Computer Vision & Artificial Intelligence

Munich, 23 June 2021



Motivation

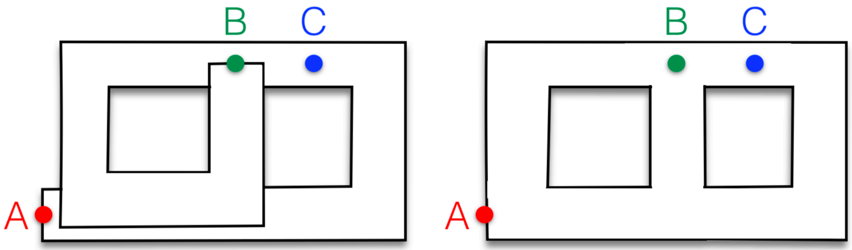
Direct methods

- raw sensor measurements
- photometric error

Goal

create a direct SLAM system with

- local mapping
- global mapping



Odometry SLAM

Direct Sparse Odometry (DSO) [2]

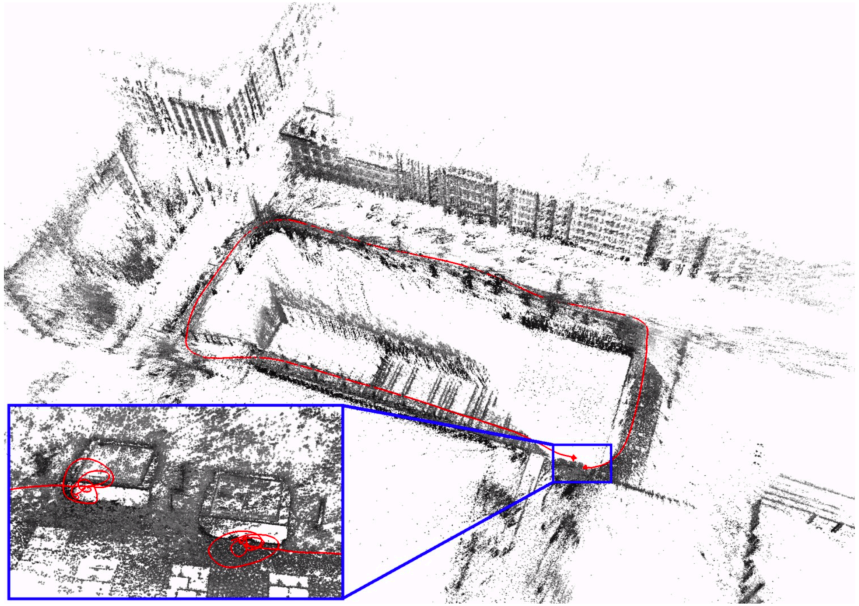
photometric bundle adjustment (PBA)

has advantages of direct approaches & sparse data

uses a sliding window
marginalizes old frames and points

cannot reuse information

accumulated drift on long trajectories



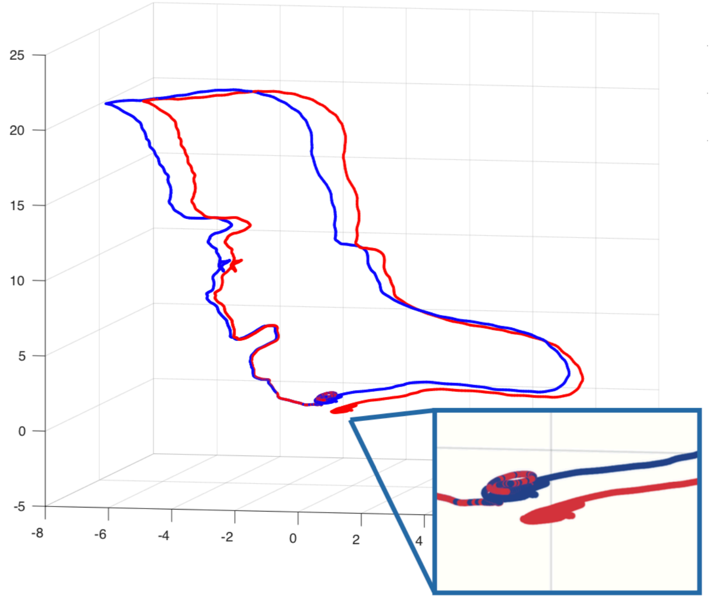
Direct Sparse Odometry with Loop Closure (LDSO) [3]

extends DSO with

- loop closure detection (LC)
- pose graph optimization (PGO)

Detecting revisited scenes helps reduce drift

Reobservations are not used in local optimization



Direct Sparse Mapping (DSM) [4]

global map

Local Map Covisibility Window (LMCW)

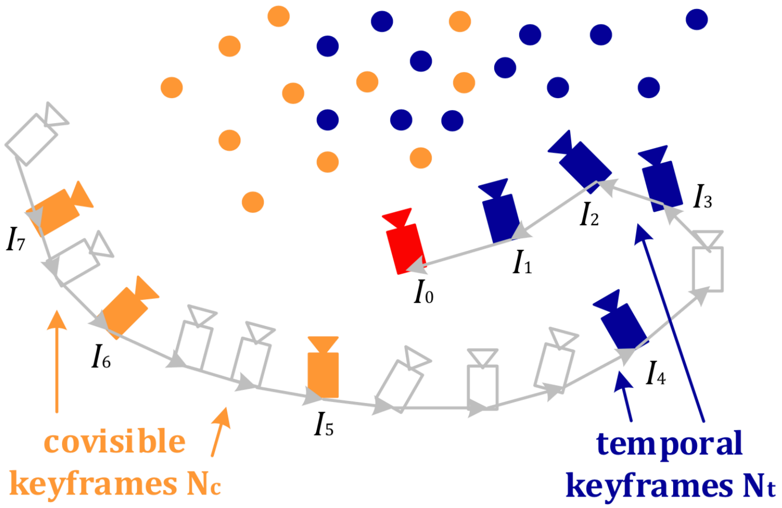
reuse of keyframes and map points

a coarse-to-fine optimization scheme

a robust influence function

outlier management using t-distribution

no explicit compensation of the drift for larger loops



Direct Sparse Mapping with Loop Closure (LDSM)

So far, we have seen
- DSO: a direct VO
- LDSO: a direct SLAM with loop closure
- DSM: a direct SLAM with map reuse

We propose an extention of DSM, **LDSM**, a direct SLAM system with
- loop closure detection
- pose graph optimization
- map reuse

Main Method

Feature Point Selection

We need feature points for loop closure detection and verification

DSM uses image gradients to select points

- not usually repeatable, e.g. points from weakly textured regions or edges

select a portion of points as repeatable feature points

do not select non-feature points near feature points

We use Shi-Tomasi corners [5] and ORB descriptors [6]

Loop Closure Detection

DBoW3 [7]

- converts images into bag of words vectors
- implements a database enabling queries

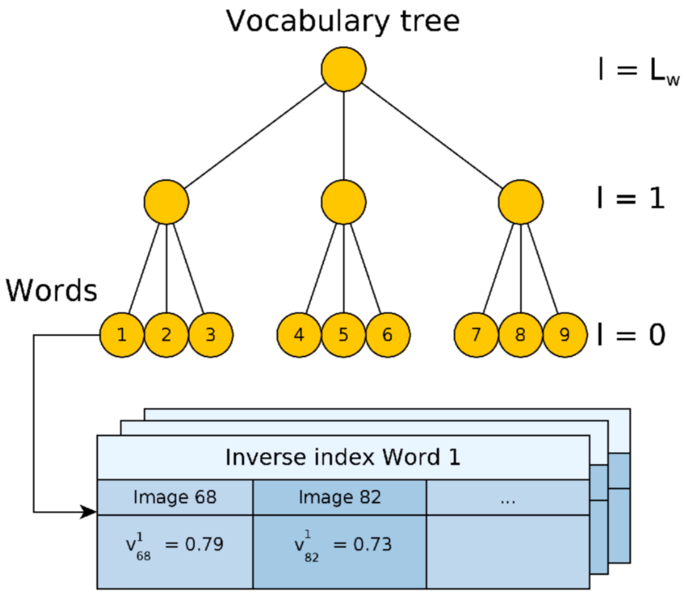
query the database for similar keyframes to I_r

filter the neighborhood N_r

- temporal connections
- covisible connections with >100 shared points

take 10 best candidates $\{I_c\}$

future work: use an adaptive threshold



[8]

Loop Closure Detection

Relative Pose Estimation

corner matches: descriptor similarity

estimate $Sim(3)$ relative pose S_{rc} by solving the PnP problem

create & use depth maps to estimate depth of 2D points
3D points: active points, optimized candidate points, points with estimated depth

solve PnP both ways to derive relative scale s_{rc}

Relative Pose Optimization

corner matches: descriptor similarity + spatial closeness

allows locally unique matches

$$E_{LOOP} = \sum_{p_i, q_i \in P} \left(|p_i - S_{rc} * q_i|_{\gamma_1} + |\Pi(p_i) - \Pi(S_{rc} * q_i)|_{\gamma_2} \right)$$

S_{rc}: $Sim(3)$ transformation from I_c to I_r

$P = \{p_i, q_i\}$: set of 3D point matches between I_r and I_c

$\Pi(\cdot)$: projection function

$|\cdot|_{\gamma_1}, |\cdot|_{\gamma_2}$: Huber norms

Relative Pose Optimization

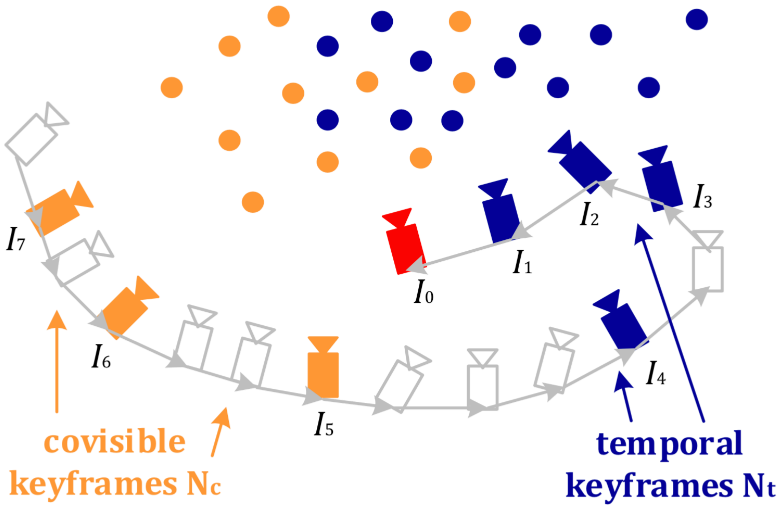
Changes to LMCW

Covisible window selection procedure originally

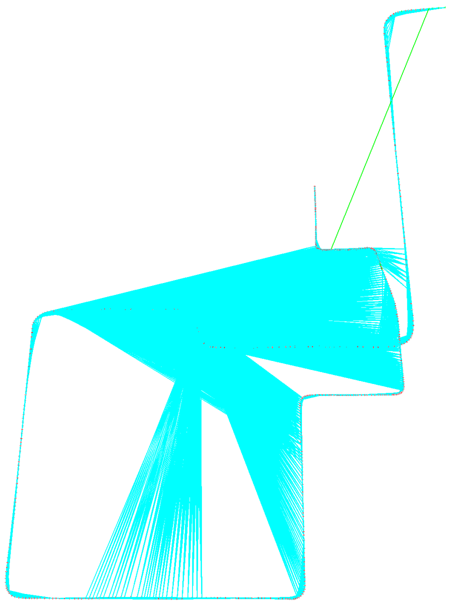
- traverses all inactive keyframes
- only applies a geometric check

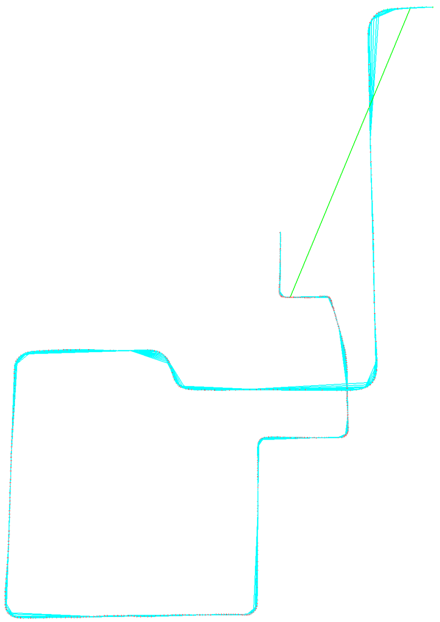
limit search range to the inactive neighborhood

add a feature-based control step
    to prevent false covisibility connections



Changes to LMCW





Pose Graph Optimization

relative pose constraints from loop closures and covisibility graph

relative poses from covisibility graph are based on local BA, not based on camera poses

to not use PGO results to define subsequent PGO problems

$$E_{PGO} = \sum_{i,j} w_{ij} \log_{Sim(3)}(S_{ij} * S_{wj}^{-1} * S_{wi})$$

S_{wi}: $Sim(3)$ pose of I_i

S_{ij}: $Sim(3)$ relative pose from I_j to I_i

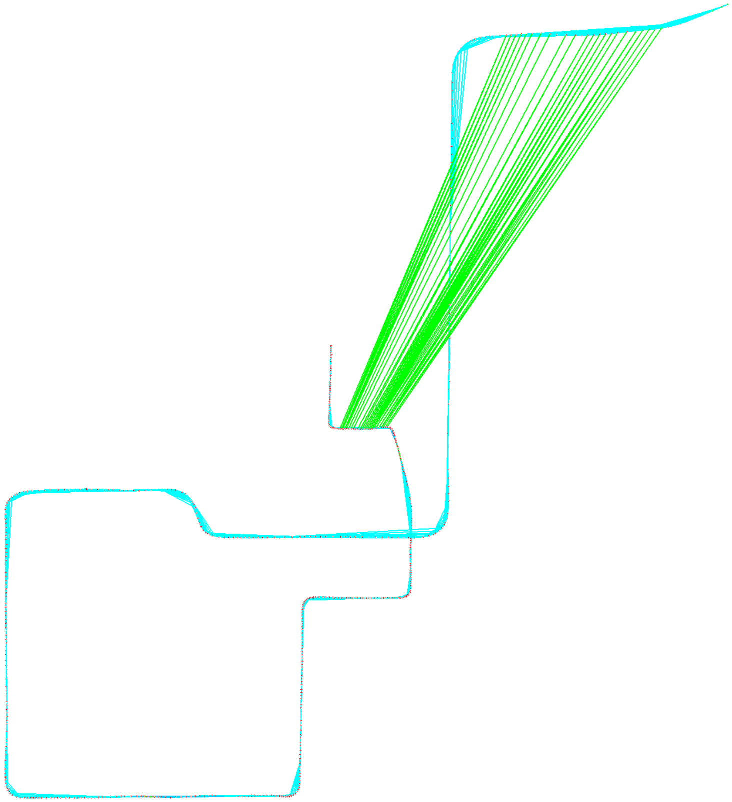
$\log_{Sim(3)}$: maps $Sim(3)$ to its tangent space, $\mathbb{R}^7$

w_{ij}: 100 for loop closures, 1 otherwise

Pose Graph Optimization

We wait 5 keyframes without a loop closure to run PGO

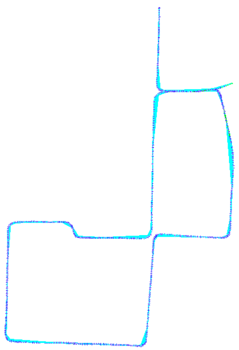
We update frame poses, map points, and relative pose constraints based on PGO results



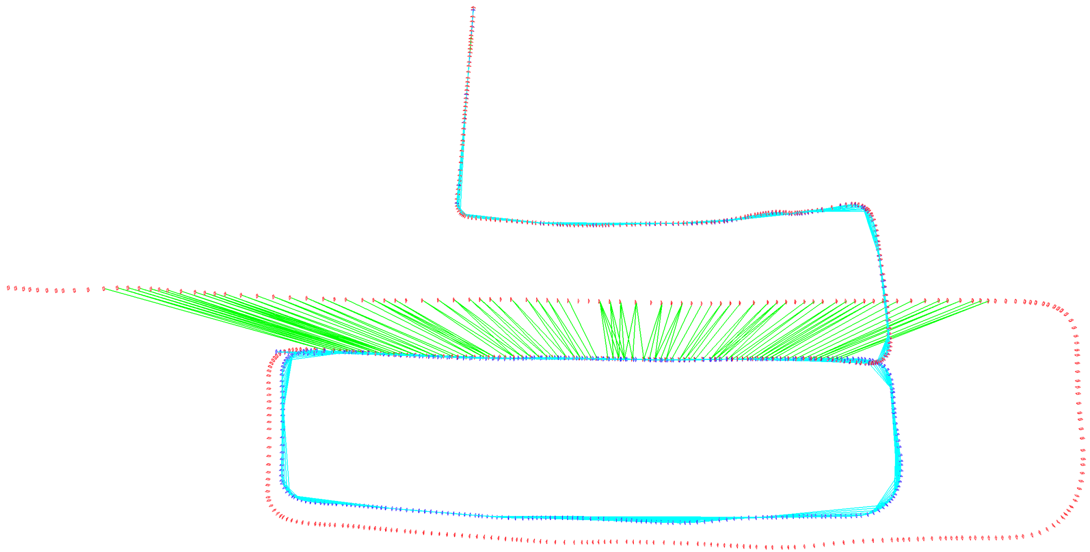
Pose Graph Optimization

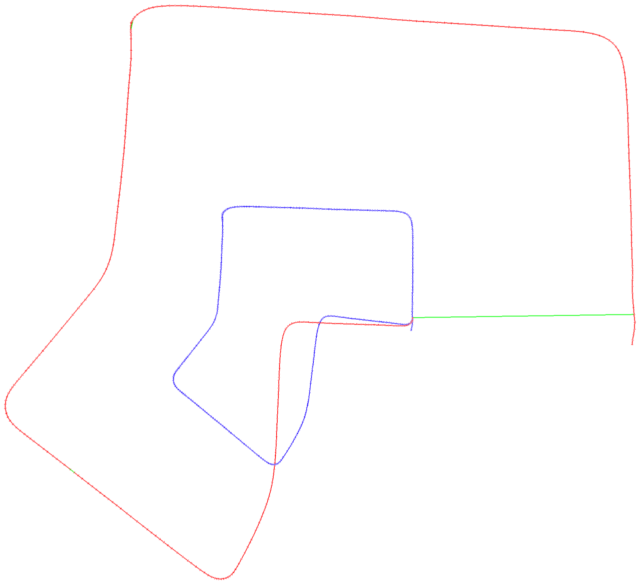
We wait 5 keyframes without a loop closure to run PGO

We update frame poses, map points, and relative pose constraints based on PGO results



Pose Graph Optimization



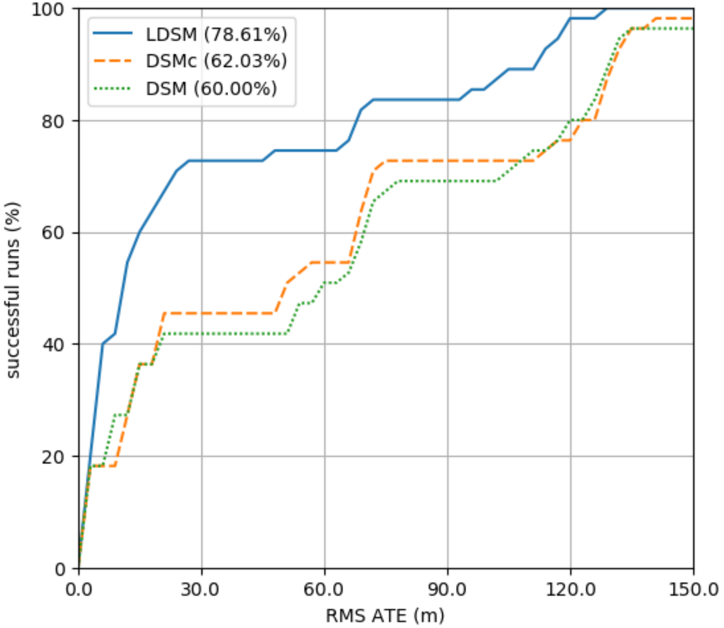


Results

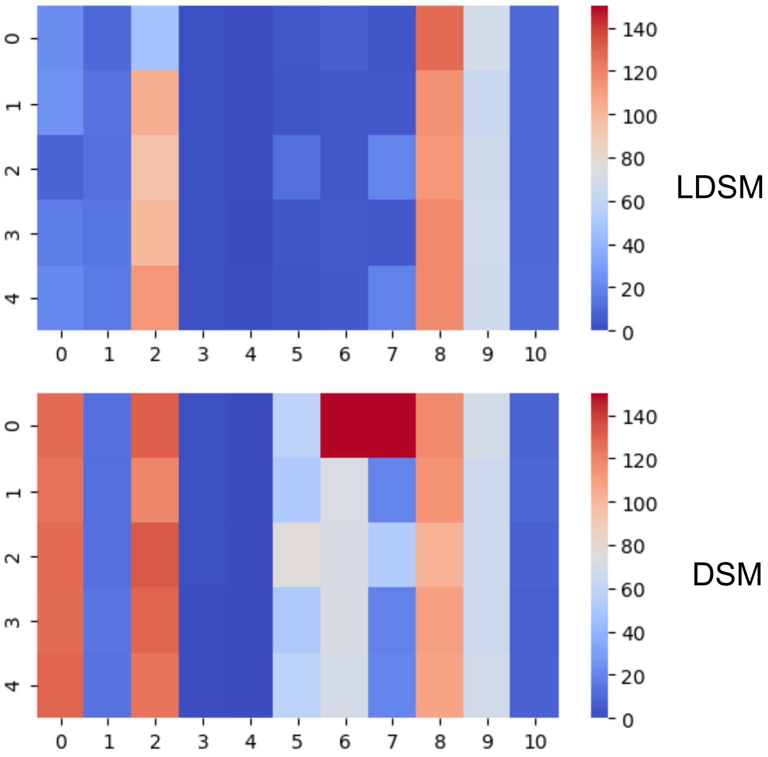
Kitti Odometry dataset [9]

11 sequences from a driving car

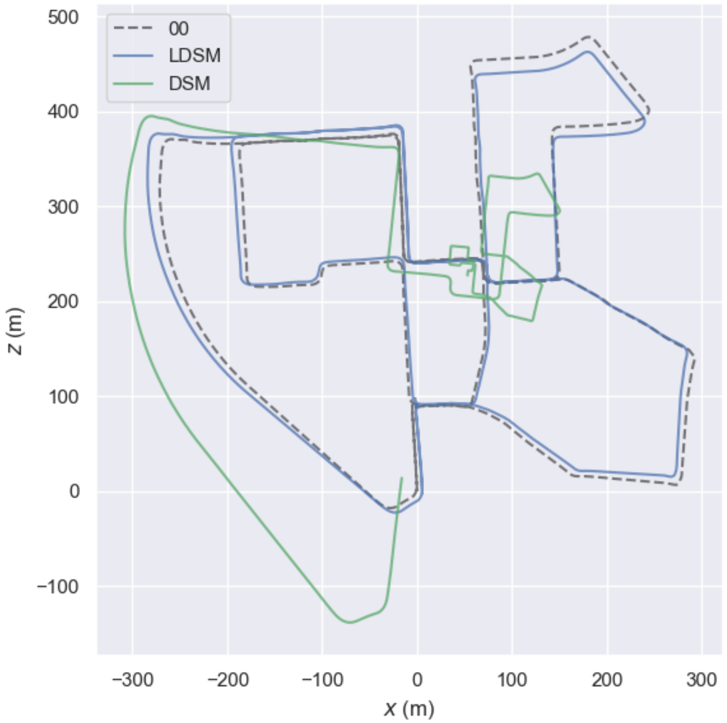
6 sequences with one or more loops: 00, 02, 05, 06, 07, 09



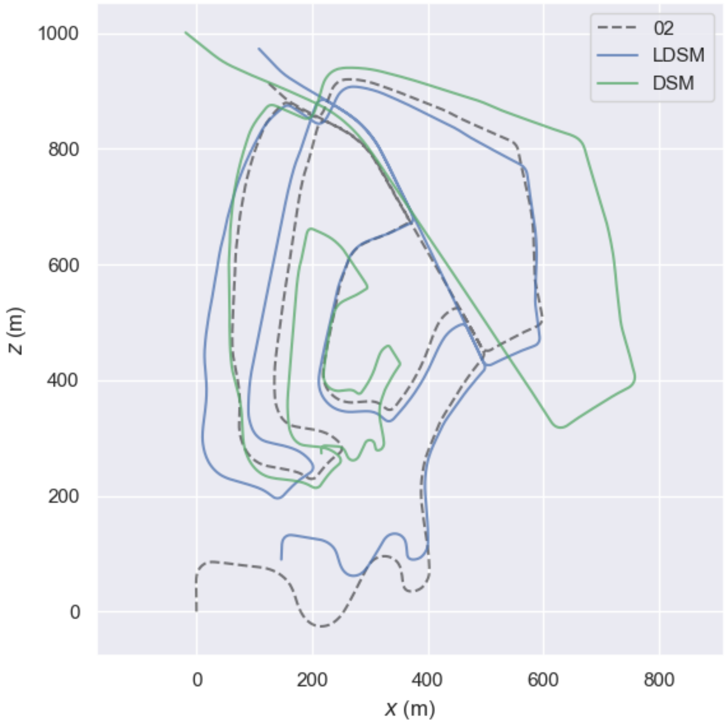
LDSM (78.61%)
DSMc (62.03%)
DSM (60.00%)
successful runs (%)
RMS ATE (m)
100
80
60
40
20
0
0.0
30.0
60.0
90.0
120.0
150.0



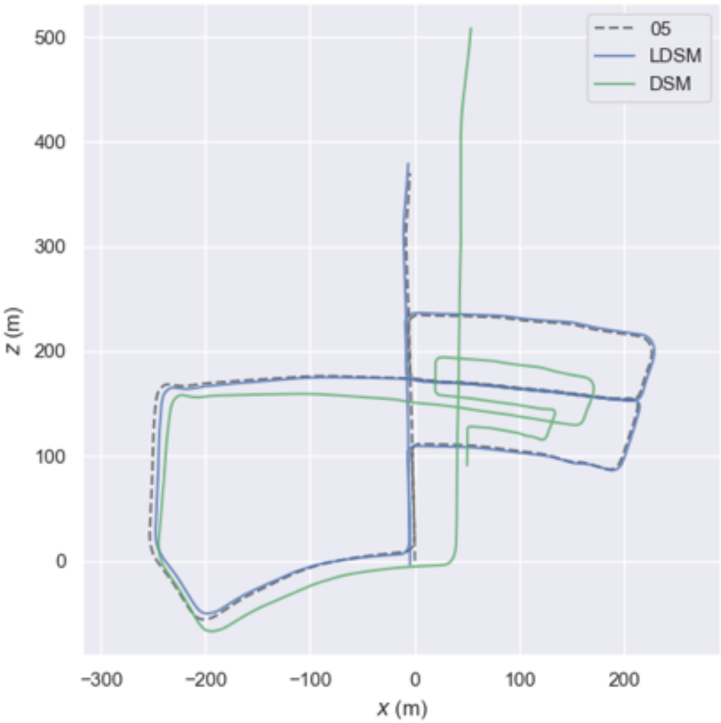
LDSM
DSM
0
1
2
3
4
5
6
7
8
9
10
140
120
100
80
60
40
20
0



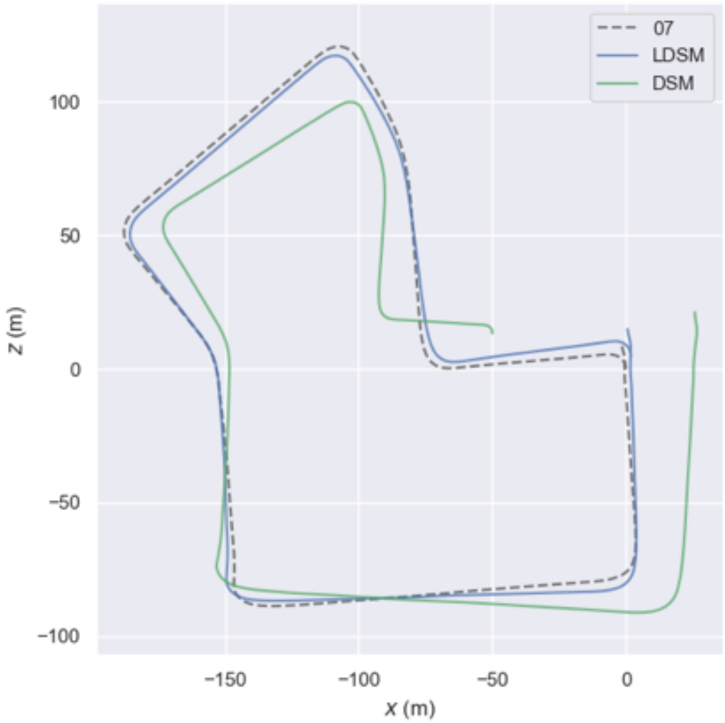
00
LDSM
DSM
500
400
300
200
100
0
−100
z (m)
−300
−200
−100
0
100
200
300
x (m)



02
LDSM
DSM
1000
800
600
400
200
0
z (m)
0
200
400
600
800
x (m)



05
LDSM
DSM
500
400
300
200
100
0
z (m)
−300
−200
−100
0
100
200
x (m)



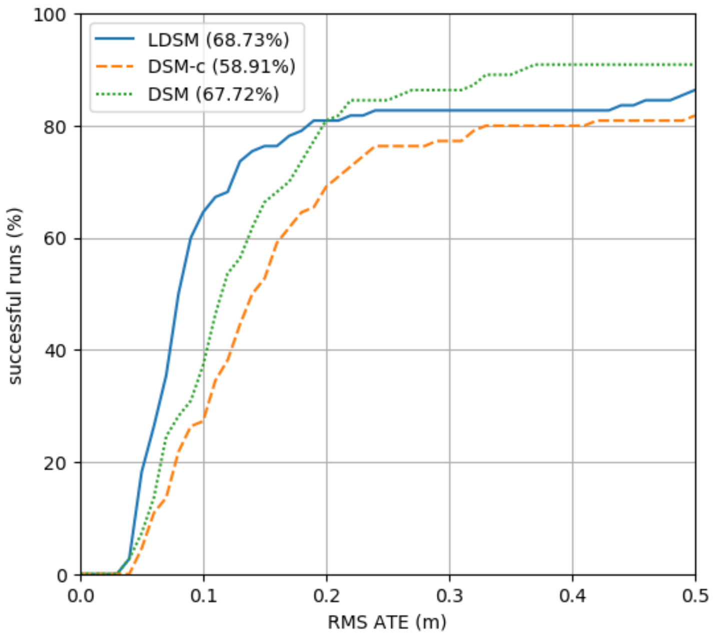
07
LDSM
DSM
100
50
0
−50
−100
z (m)
−150
−100
−50
0
x (m)

Seq.	DSM	LDSM	LDSO	ORB-SLAM	# loops	# PGOs
00	127.87	18.87	9.32	8.27	403.6	9.4
01	13.16	13.50	11.68	-	2.2	1.0
02	127.82	91.65	31.98	26.86	138.2	5.6
03	1.27	1.34	2.85	1.21	14.0	2.6
04	0.35	0.67	1.22	0.77	9.6	2.2
05	59.60	5.00	5.10	7.91	222.8	5.8
06	71.62	4.85	13.55	12.54	102.2	2.8
07	28.27	9.89	2.96	3.44	1.0	1.0
08	110.63	118.50	129.02	46.81	5.6	2.8
09	67.13	67.21	21.64	76.54	1.0	1.0
10	7.71	10.42	17.36	6.61	0.4	0.4

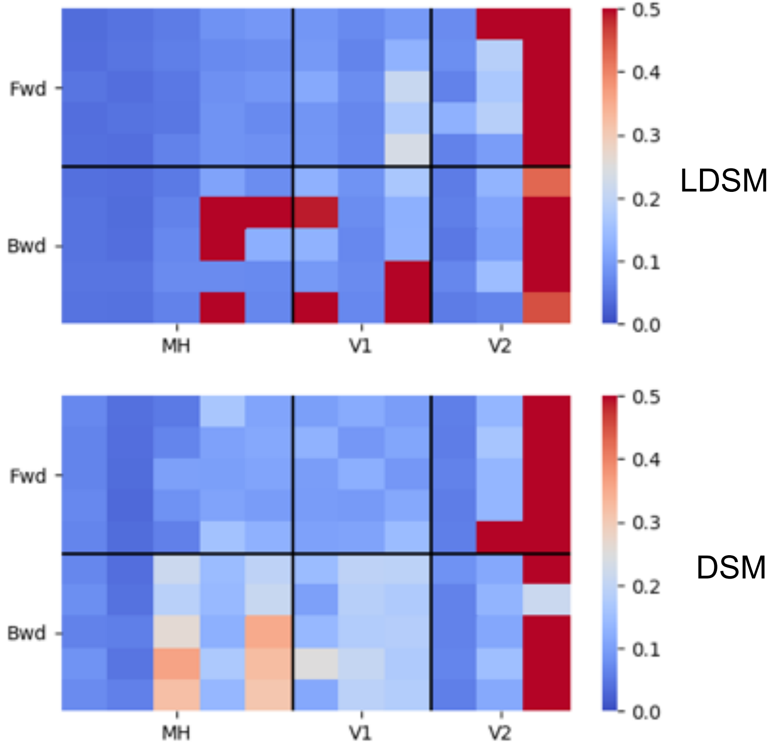
Results

EuRoC MAV dataset [10]

11 sequences in 3 indoor environments



LDSM (68.73%)
DSM-c (58.91%)
DSM (67.72%)
successful runs (%)
RMS ATE (m)
100
80
60
40
20
0
0.0
0.1
0.2
0.3
0.4
0.5



Fwd
Bwd
MH
V1
V2
0.5
0.4
0.3
0.2
0.1
0.0
LDSM
Fwd
Bwd
MH
V1
V2
0.5
0.4
0.3
0.2
0.1
0.0
DSM

Conclusion

Contributions

- LDSM, a direct SLAM system with local and global mapping
- a significant improvement over DSM for long trajectories with large loops
- comparable performance to LDSO and ORB-SLAM [11]

Future Work

- map maintenance strategy to remove redundant keyframes and map points
- runtime performance

References

[1] C. Cadena et al., “Past, present, and future of simultaneous localization and mapping: Toward the robust-perception age,” IEEE Transactions on robotics, 2016

[2] J. Engel, V. Koltun, and D. Cremers, “Direct sparse odometry,” in PAMI, 2017

[3] X. Gao, R. Wang, N. Demmel, and D. Cremers, “Ldso: Direct sparse odometry with loop closure,” in IROS, 2018.

[4] J. Zubizarreta et al., “Direct sparse mapping,” IEEE Transactions on Robotics, 2020.

[5] J. Shi et al., “Good features to track,” in CVPR, 1994

[6] E. Rublee et al., “Orb: An efficient alternative to sift or surf,” in ECCV, 2011

[7] R. Muñoz-Salinas, DBoW3, https://github.com/rmsalinas/DBoW3

[8] D. Gálvez-López et al., “Bags of binary words for fast place recognition in image sequences,” IEEE Transactions on Robotics, 2012.

[9] A. Geiger et al., “Are we ready for autonomous driving? the kitti vision benchmark suite,” in CVPR, 2012

[10] M. Burri et al., “The euroc micro aerial vehicle datasets”, The International Journal of Robotics Research, 2016.

[11] R. Mur-Artal et al., “Orb-slam: A versatile and accurate monocular slam system,” IEEE transactions on robotics, 2015