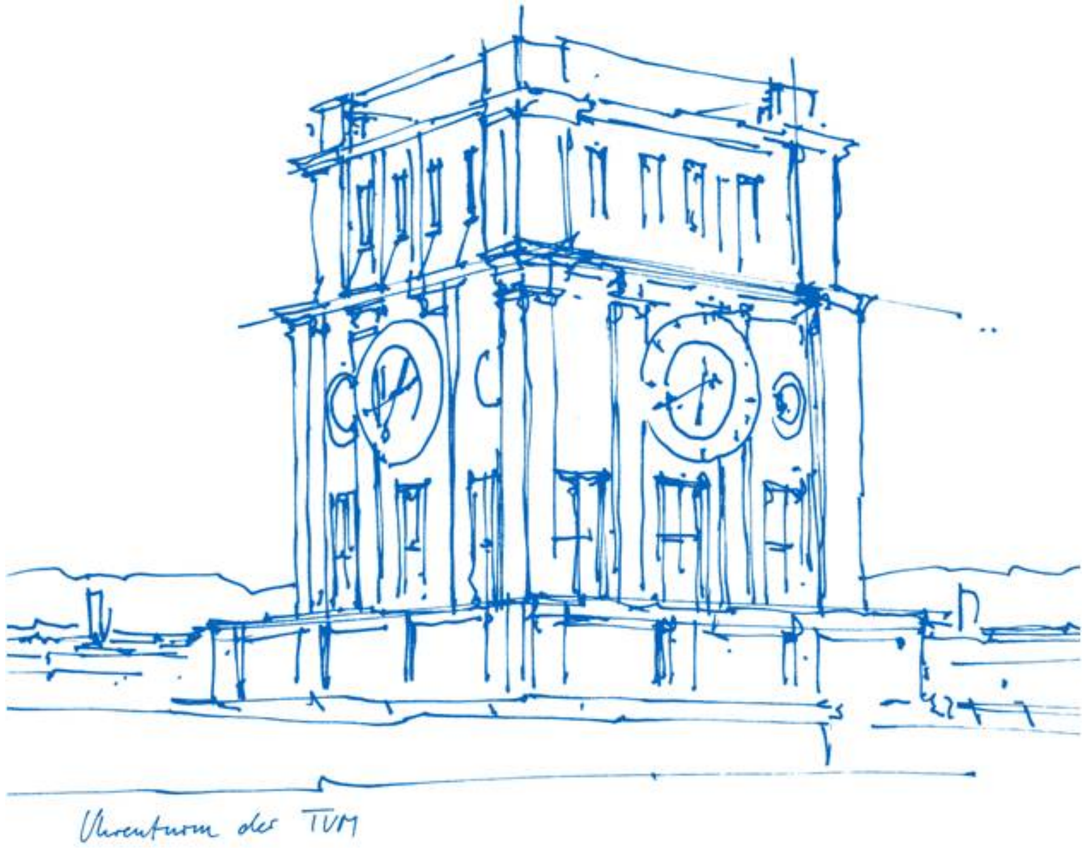
Photometric Bundle Adjustment

Nikolaus Demmel

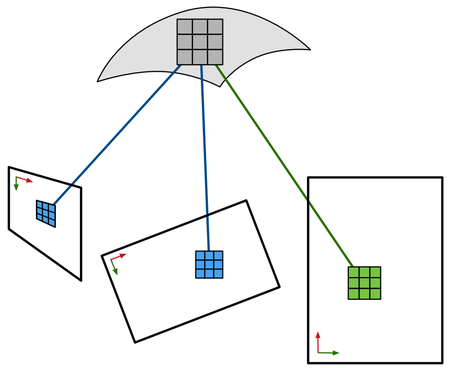
10.06.2020
CVG Internal Group Seminar



Agenda

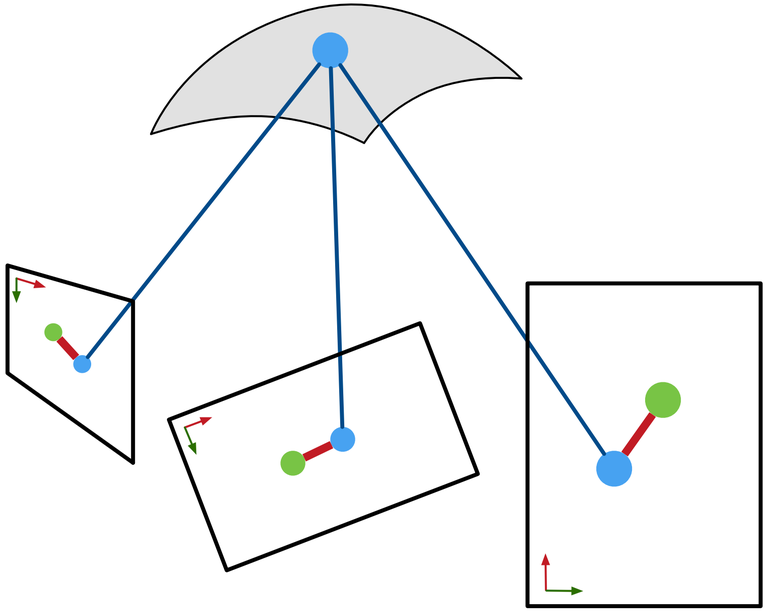
- What is Photometric Bundle Adjustment?
  - Cost Function, Factor Graph
  - Applications
  - Implementation
- Distributed PBA
  - Splitting approach with penalty method (ECCV submission)
  - Limitations and outlook
- Ongoing Student Projects

Photometric Bundle Adjustment Primer



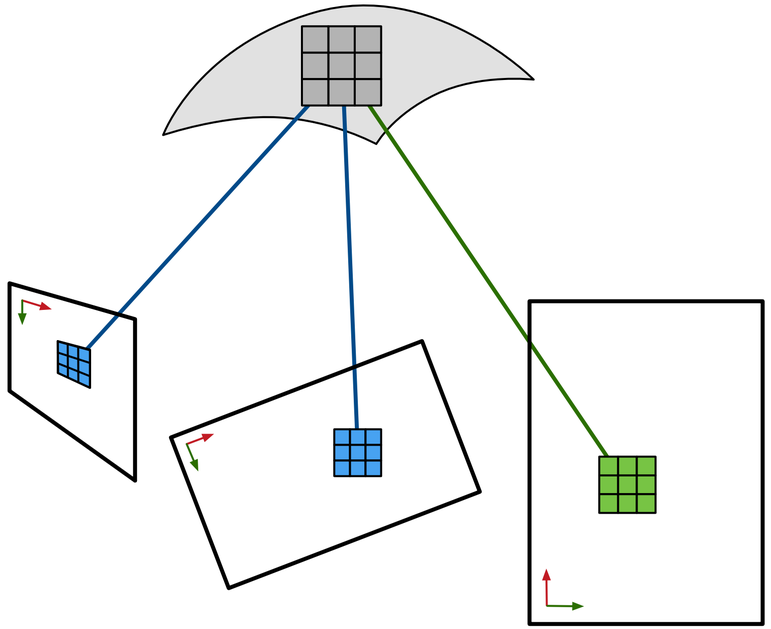
Cost Function

Geometric BA



reprojection error: $\sum_{\text{i,k}} \|\mathbf{y}_{ik} - \pi(\mathbf{T}_i \mathbf{x}_k)\|^2$

Photometric BA



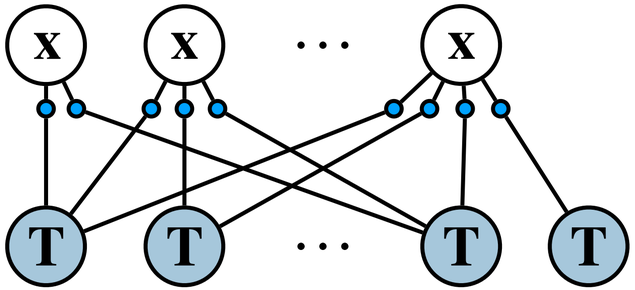
photometric error: $\sum_{\text{i,j,k}} \sum_{\mathbf{p} \in P_k} \|I_j(w(\mathbf{p})) - I_i(\mathbf{p})\|^2$

patch warp: $w(\mathbf{p}; \mathbf{T}_i, \mathbf{T}_j, d_k) = \pi(\mathbf{T}_j \mathbf{T}_i^{-1} \pi^{-1}(\mathbf{p}, d_k))$

Factor Graph and Normal Equations

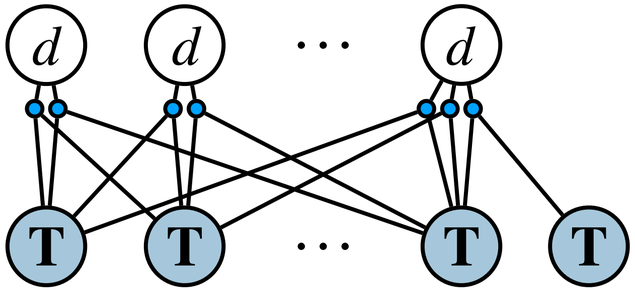
Geometric BA

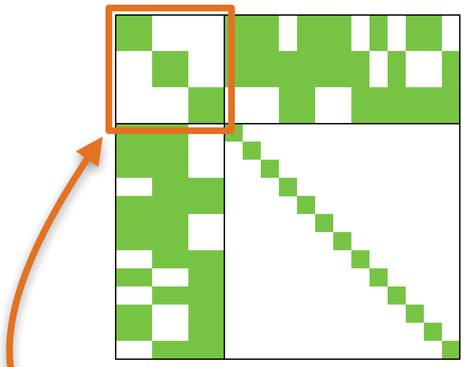
$$\sum_{\text{i,k}} \|\mathbf{y}_{ik} - \pi(\mathbf{T}_i \mathbf{x}_k)\|^2$$



Photometric BA

$$\sum_{\text{i,j,k}} \sum_{\mathbf{p} \in P_k} \|I_j(w(\mathbf{p}; \mathbf{T}_i, \mathbf{T}_j, d_k)) - I_i(\mathbf{p})\|^2$$





Normal Equations:

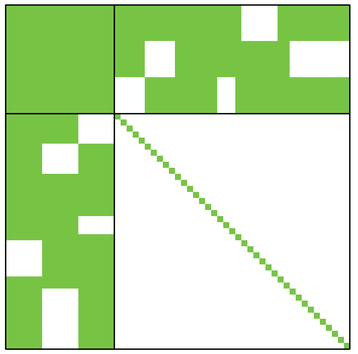
$$Hx = b$$

$$\begin{pmatrix} H_{pp} & H_{pl} \\ H_{pl}^\top & H_{ll} \end{pmatrix} \begin{pmatrix} x_p \\ x_l \end{pmatrix} = \begin{pmatrix} b_p \\ b_l \end{pmatrix}$$

Schur Complement:

$$(H_{pp} - H_{pl} H_{ll}^{-1} H_{pl}^\top) x_p = (b_p - H_{pl} H_{ll}^{-1} b_l)$$

$$x_l = H_{ll}^{-1}(b_l - H_{pl}^\top x_p)$$

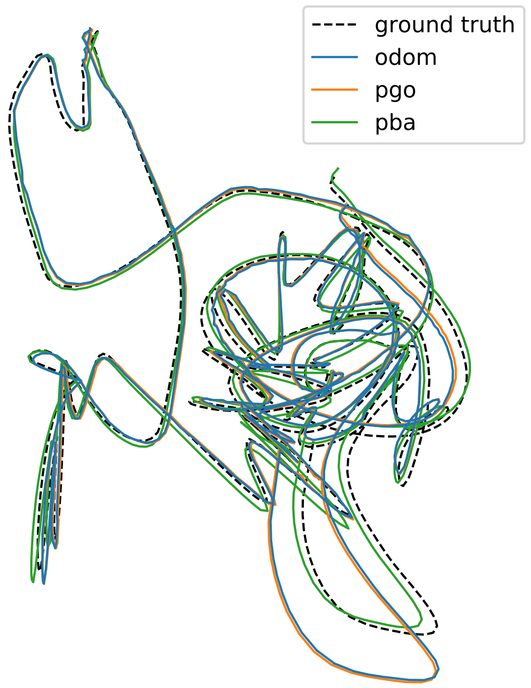


Applications

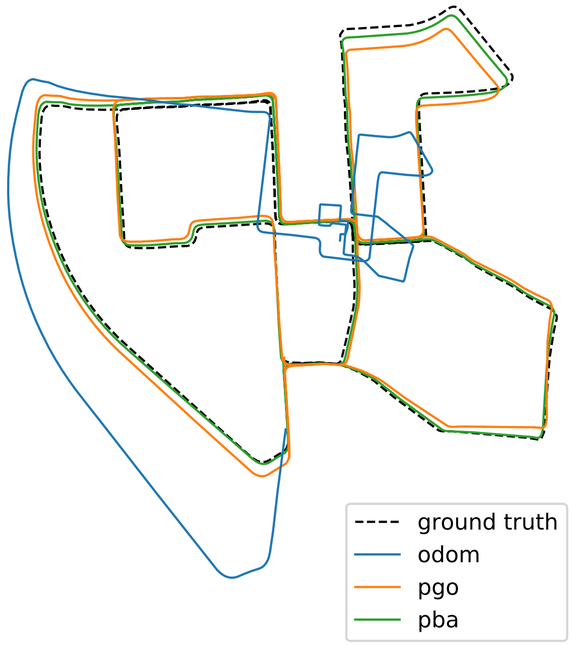
- Literature
  - Delaunoy et al. (CVPR14) "Photometric Bundle Adjustment for Dense Multi-view 3d Modeling”
  - Alismail et al. (ACCV16) "Photometric Bundle Adjustment for Vision-based SLAM"
  - Engel et al. (PAMI17) "Direct Sparse Odometry"
  - Zubizarreta et al. (TRO20) "Direct Sparse Mapping”

- Our Scenario
  - Initialize with LDSO (after loop closure / PGO)
  - Global PBA over all keyframes

EuRoC V103

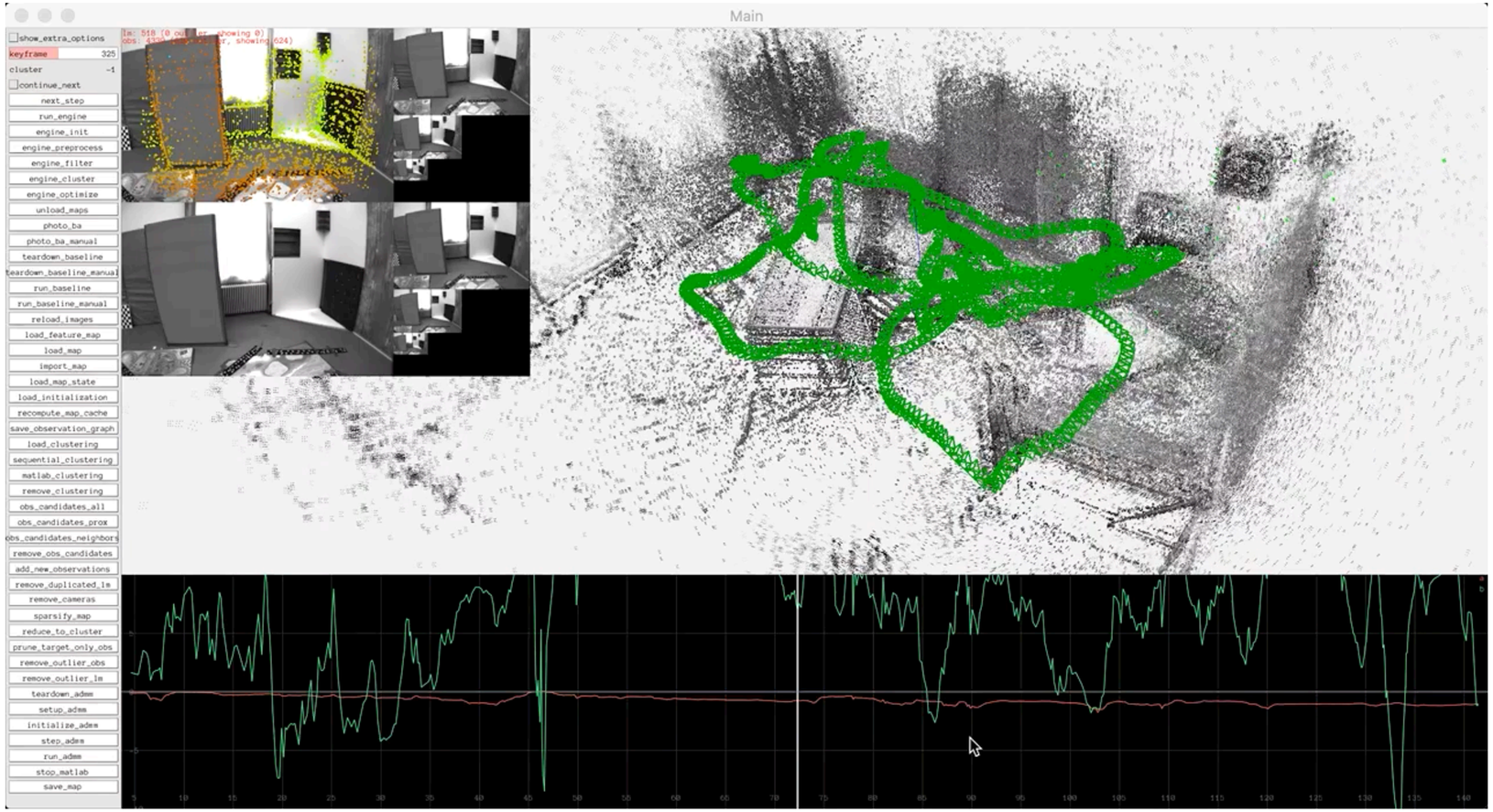


Kitti 00



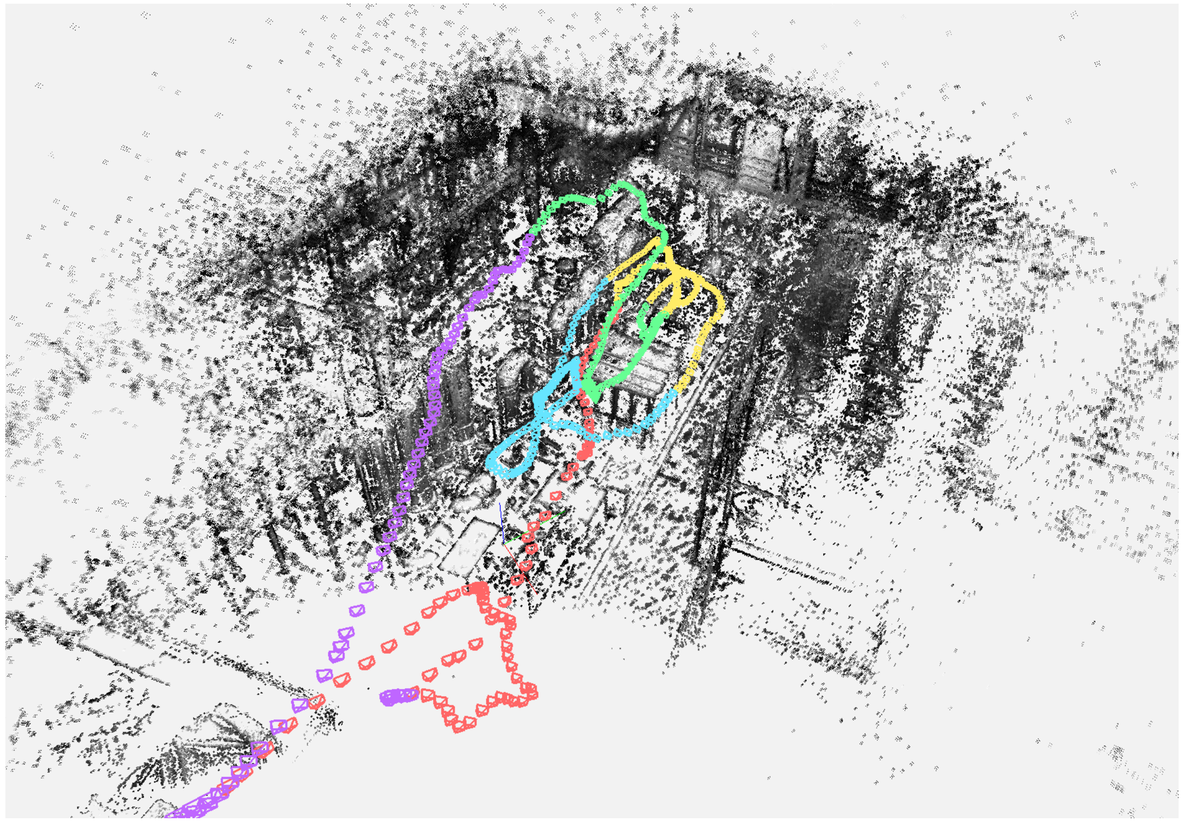
Implementation

- flexible framework: **dbatk**
  - camera models (distorted images)
  - residual definitions
  - visualization
  - load datasets
  - import LDSO result
  - graph manipulation
  - extensive logging, plotting, …
  - batch execution
  - ceres solver
  - manual solver (3-4x faster)

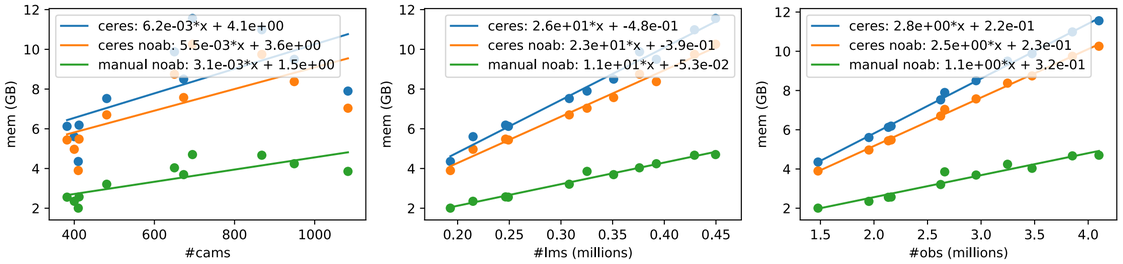




Distributed Photometric Bundle Adjustment

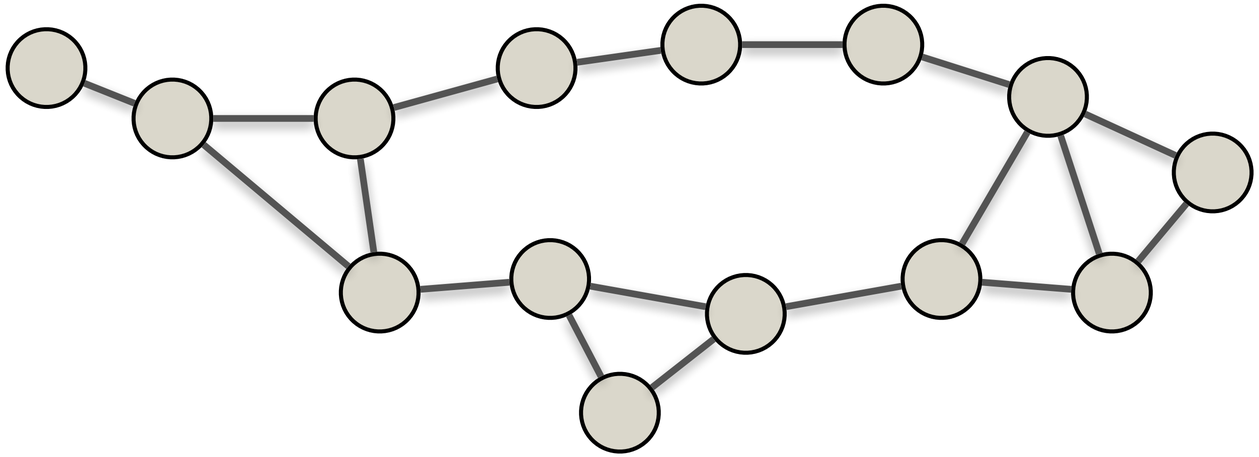


Distributed Photometric Bundle Adjustment

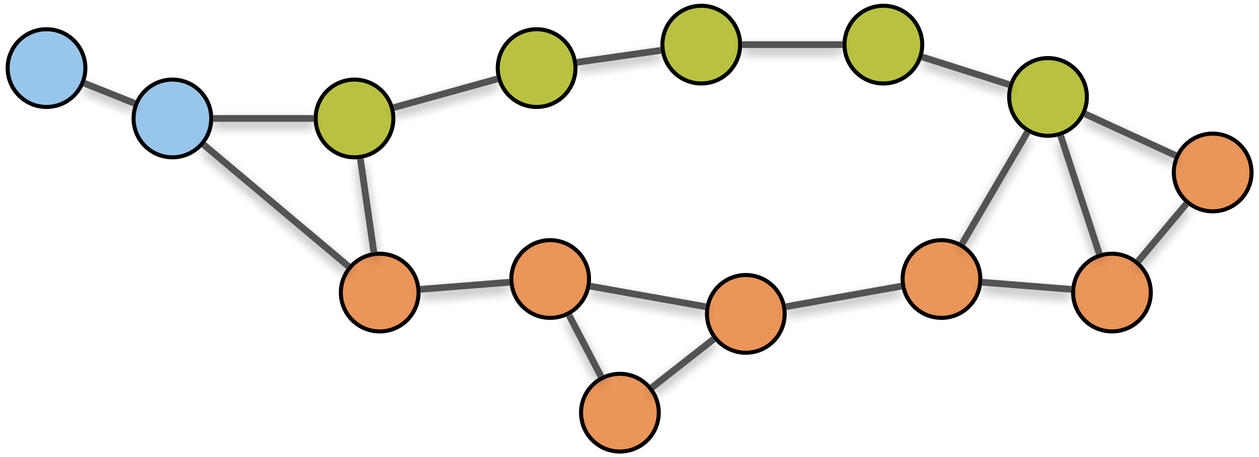


- PBA is expensive:
  - DSO w/ 7 KF 1x realtime; DSM w/ 7 KF 0.25x realtime
  - Number of points and residuals much higher than GBA
  - Residuals have higher dimension (e.g. 8 for DSO patch, vs 2 for GBA)
  - Cost is more expensive to evaluate (image gradient and interpolation)
  - Need to store images, not just 2D locations
  - Solving linear system has cubic runtime with growing number of cameras
- Idea:
  - Split up factor graph into subproblems that can be solved independently on “workers” (expensive)
  - Regularly aggregate information on the “master” (cheap)
  - Make sure required communication is limited: Only exchange camera states

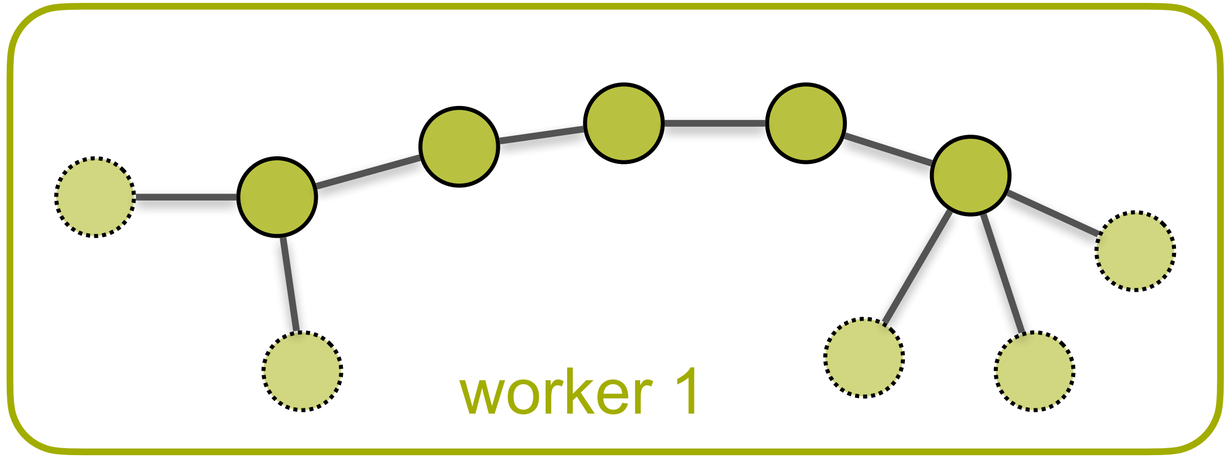
Splitting

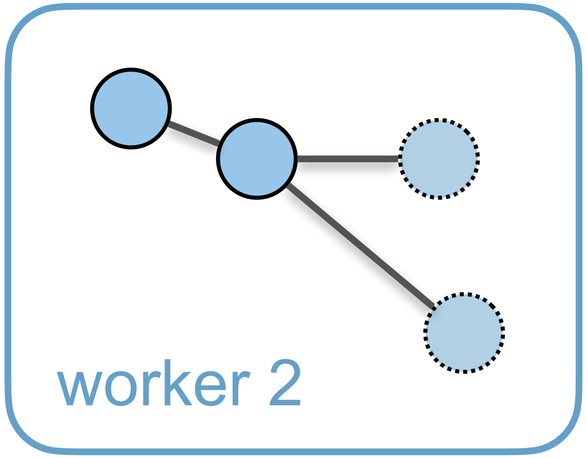


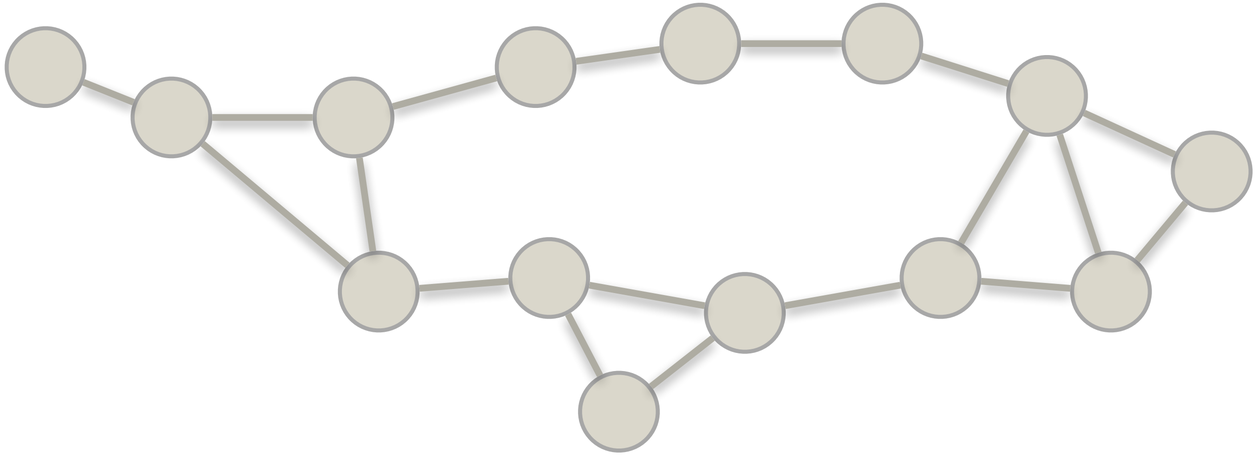
Splitting

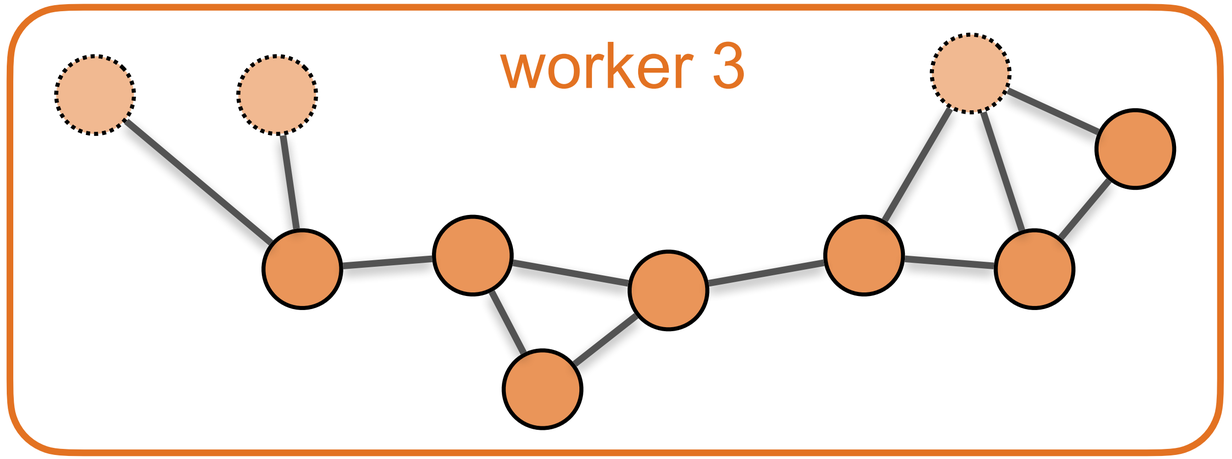


Splitting

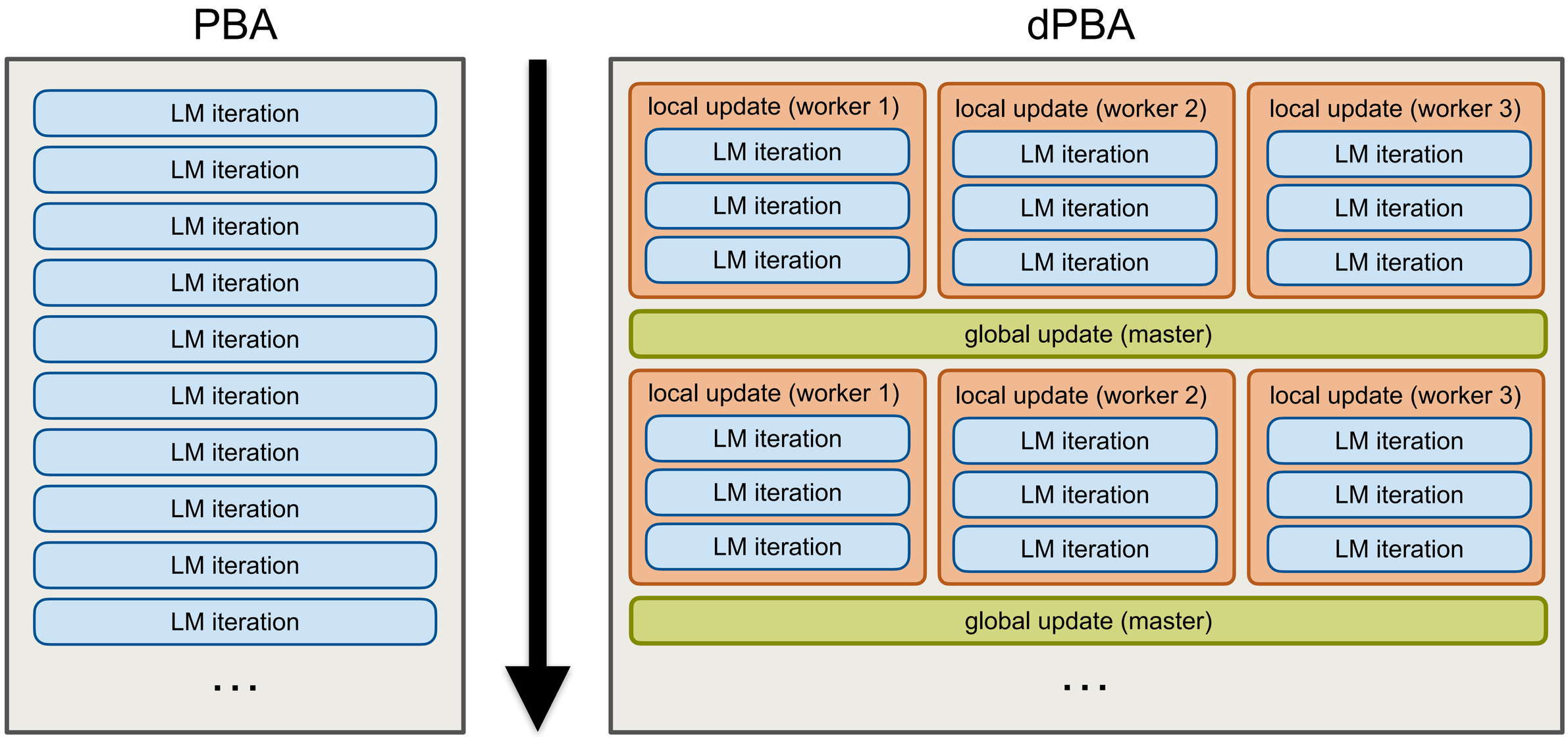






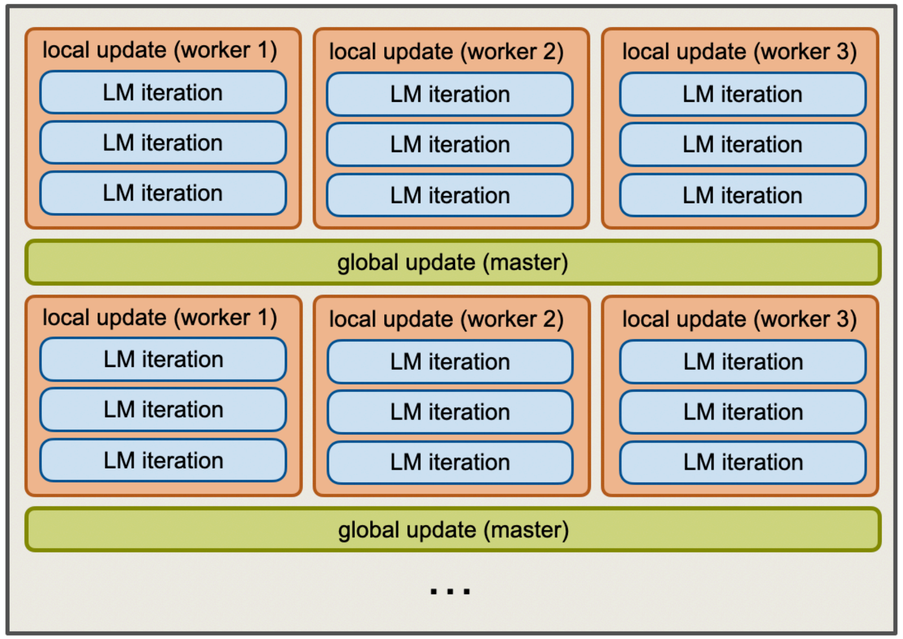


Penalty Method: Algorithm



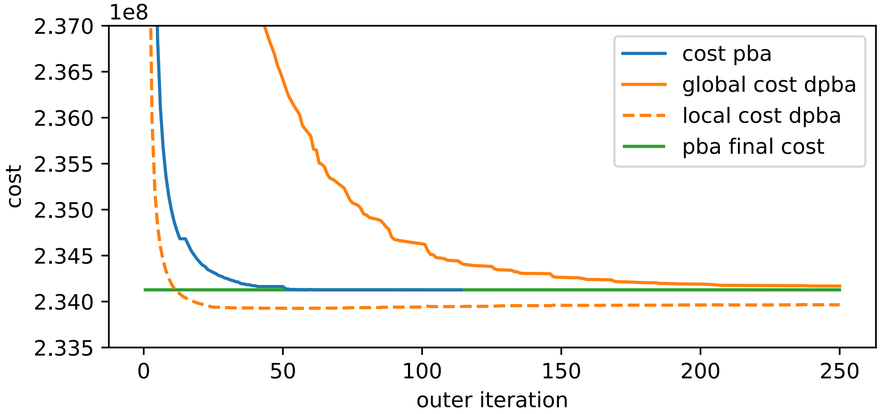
Penalty Method: Discussion

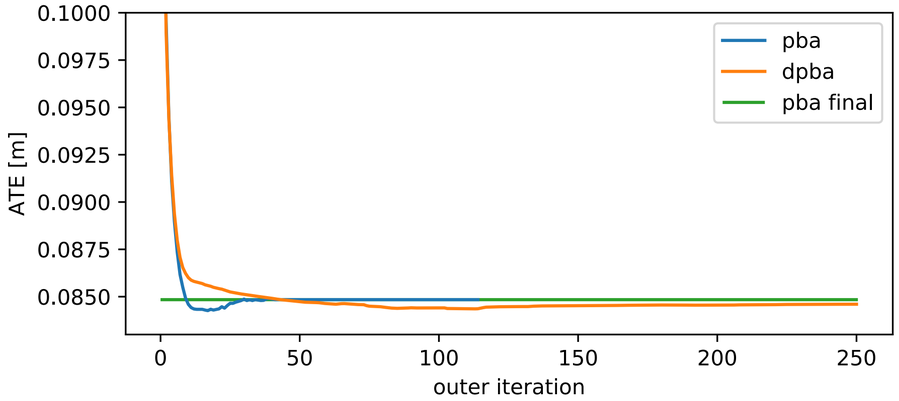
- local update: regular PBA using LM with additional quadratic penalties
  - several LM iterations
  - much smaller than original problem
- global update: “averaging” multiple local copies
  - only poses are copied
- penalty parameter ρ increased over time



Results

EuRoC V103: Cost and ATE



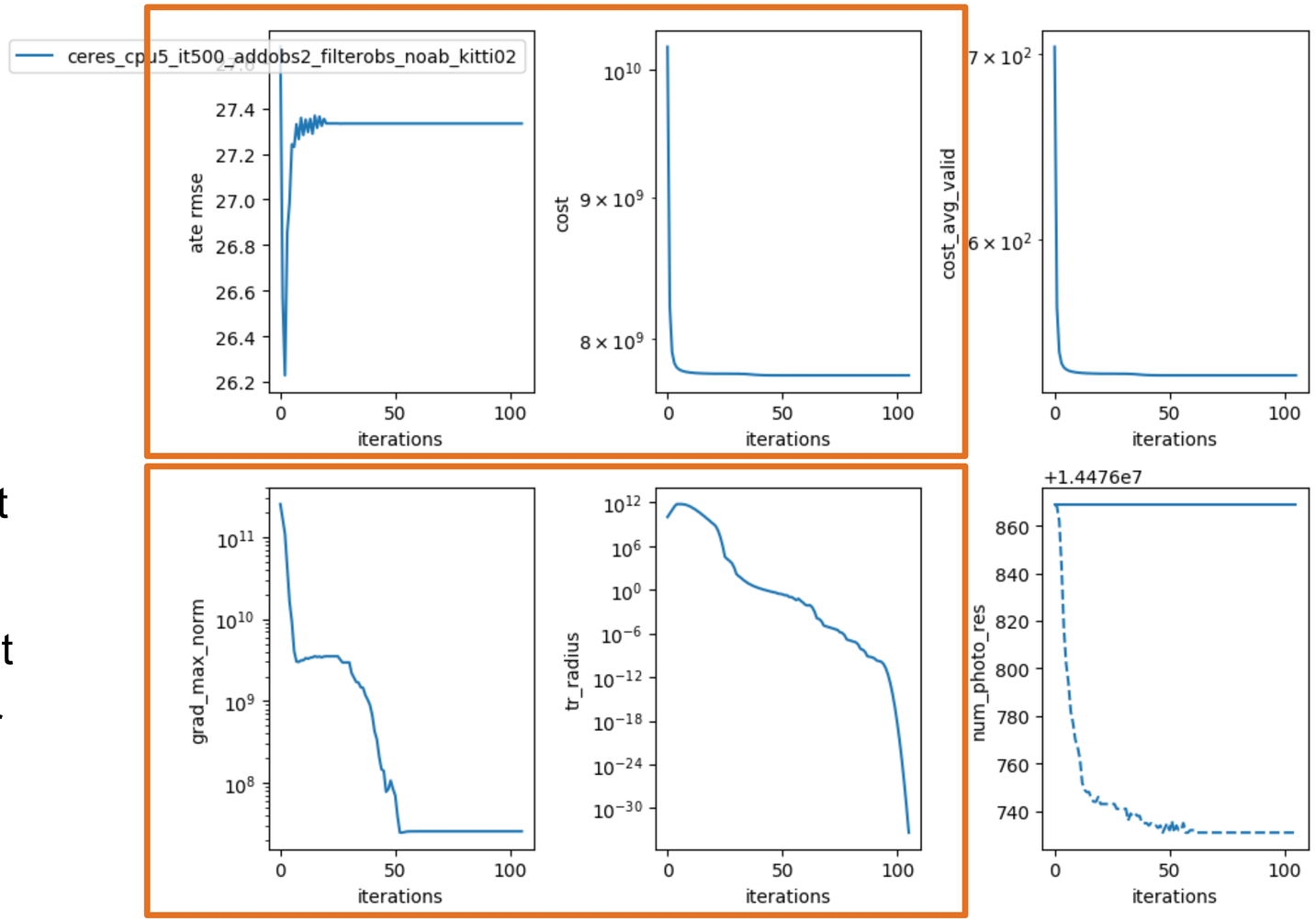


EuRoC: All Sequences

	pba	dpba			nogauge
ρ-increase rate β		1.06	1.1	1.2	1.06
# iterations (avg)	148	250	150	75	250
fraction of *odom*-ATE	0.363	0.376	0.379	0.380	0.386
fraction of *pgo*-ATE	0.644	0.667	0.673	0.673	0.685
cost global	1.0000	1.0022	1.0038	1.0076	1.0085
cost local	1.0000	0.9987	1.0003	1.0040	1.0040

Limitations and Outlook

- ATE not always meaningful
  - other metrics?
  - evaluate landmarks
- dPBA sometimes far from ATE of PBA (Kitti)
- numeric issues?
  - algorithm terminates with “big lambda”, not “small gradient”
  - (preconditioned) gradient descent does not work well —> dPBA is related to first-order methods
  - ill-conditioned linear problem?
  - manual rescaling of variables?
  - different linear solvers?



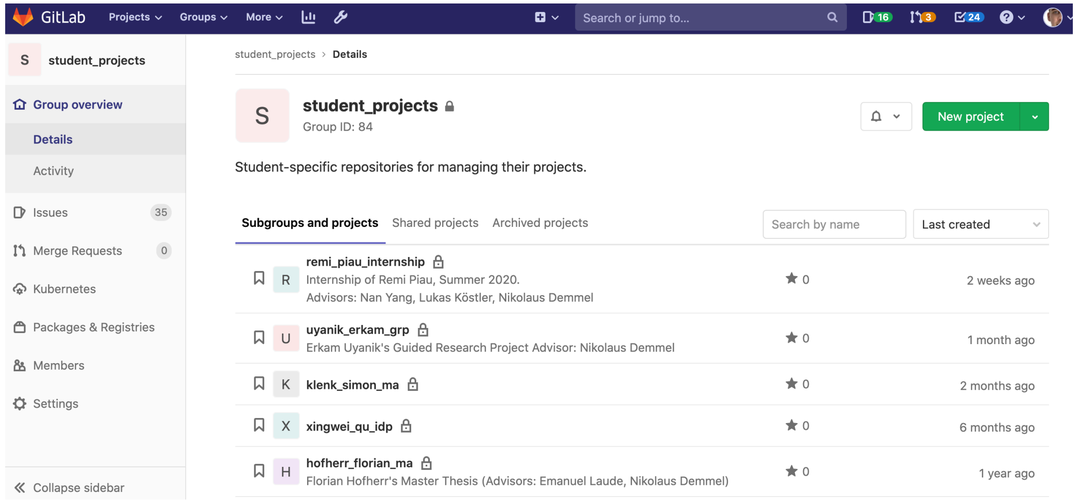
Limitations and Outlook 2

- runtime of dPBA bad
  - multiple iterations in local update don't help much —> number of outer iterations still high
  - runtime dominated by linearization / computation of SC —> not linear solver!
- Conclusion: Full 2nd order method (LM) uncontested in speed and accuracy
- Future: Focus on parallelisation of a single LM iteration
  - Exploit structure of PBA problem
  - Parallelise PCG to avoid too large communication overhead
  - Can also be applied to GBA
- Investigate alternative initialisations: Postprocess SfM, feature-based SLAM, …
  - initialise new landmarks, deduplicate landmarks, detect occlusions, handle inliers/outliers

timing cluster sizes (1 thread; avg over first 10 it; relative)

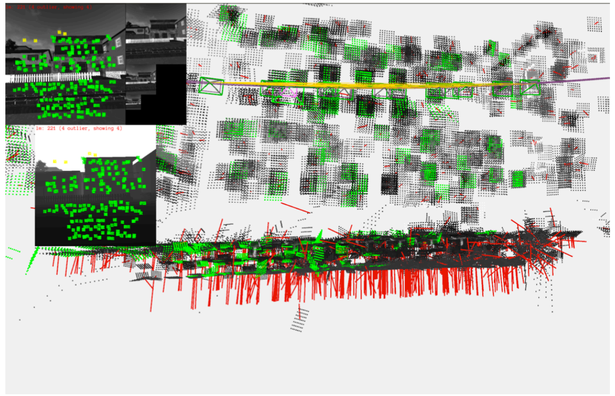
	pba	2	3	5	10	15	20	30
time/it total fake parallel	1.000	2.022	1.604	1.221	0.871	0.770	0.677	0.657
time/it total	1.000	5.331	5.851	6.234	7.185	8.211	9.295	11.104

Student Projects



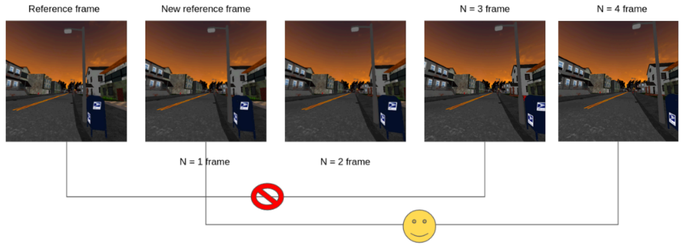
Student Projects

- **Simon Klenk**: Global PBA details: Residual formulation, Jacobian approximations, estimating patch normals, image interpolation, LM dampening, occlusion detection
  - Presentation: Mo, 15.06., 11:00



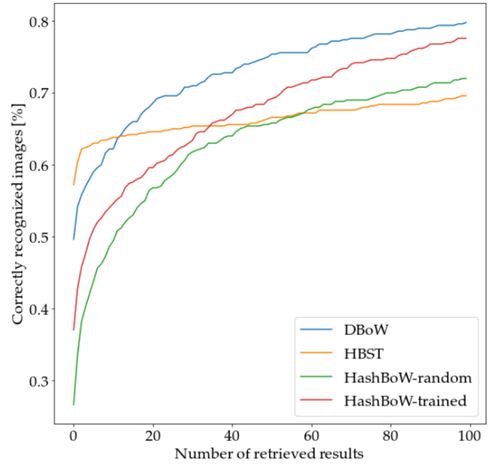
- **Xingwei Qu**: Photometric error for relative pose estimation for loop closures

- **Mariia Gladkova**: DSO initialisation using feature correspondences and PBA



- **Erkam Uyanik**: Extend DSM with loop closure and global PBA

- **Tim Stricker** (w/ Vlad): Fast and accurate visual place recognition with BoW and related techniques
  - Presentation: Thu, 25.06., 11:00



- **Rémi Piau** (w/ Lukas K, Nan): Visual place recognition with deep learning: Balancing runtime and accuracy