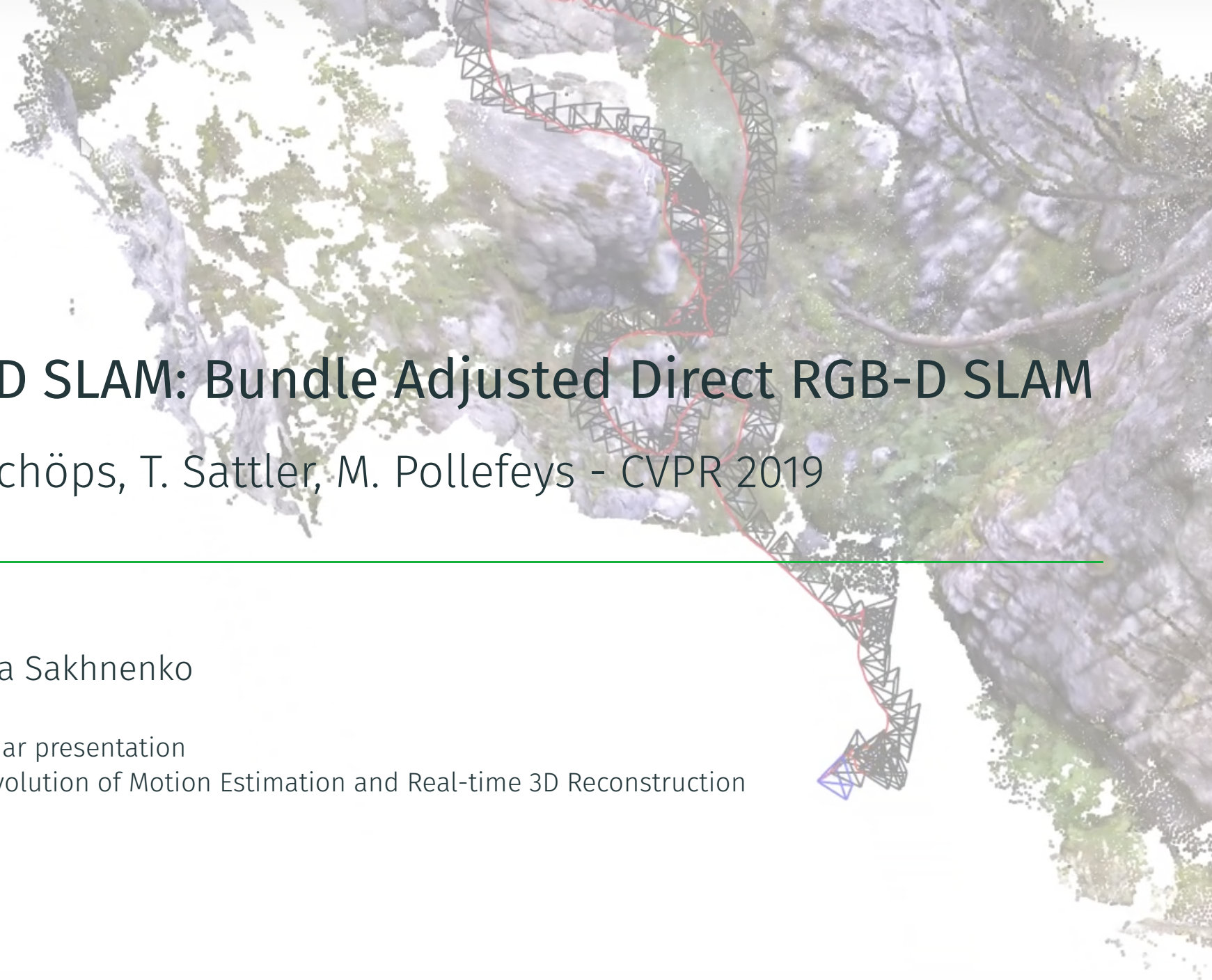


BAD SLAM: Bundle Adjusted Direct RGB-D SLAM

T. Schöps, T. Sattler, M. Pollefeys - CVPR 2019

Alona Sakhnenko

Seminar presentation
The Evolution of Motion Estimation and Real-time 3D Reconstruction

Introduction

Motivation

Motivation # 1

In previous works on dense RGB-D SLAM,
bundle adjustment (BA) was only **approximated**.

↓

We can improve the performance by using BA **directly**.

Motivation # 2

Direct RGB-D SLAM systems are highly sensitive to
rolling shutter, RGB-D asynchronization and calibration errors.

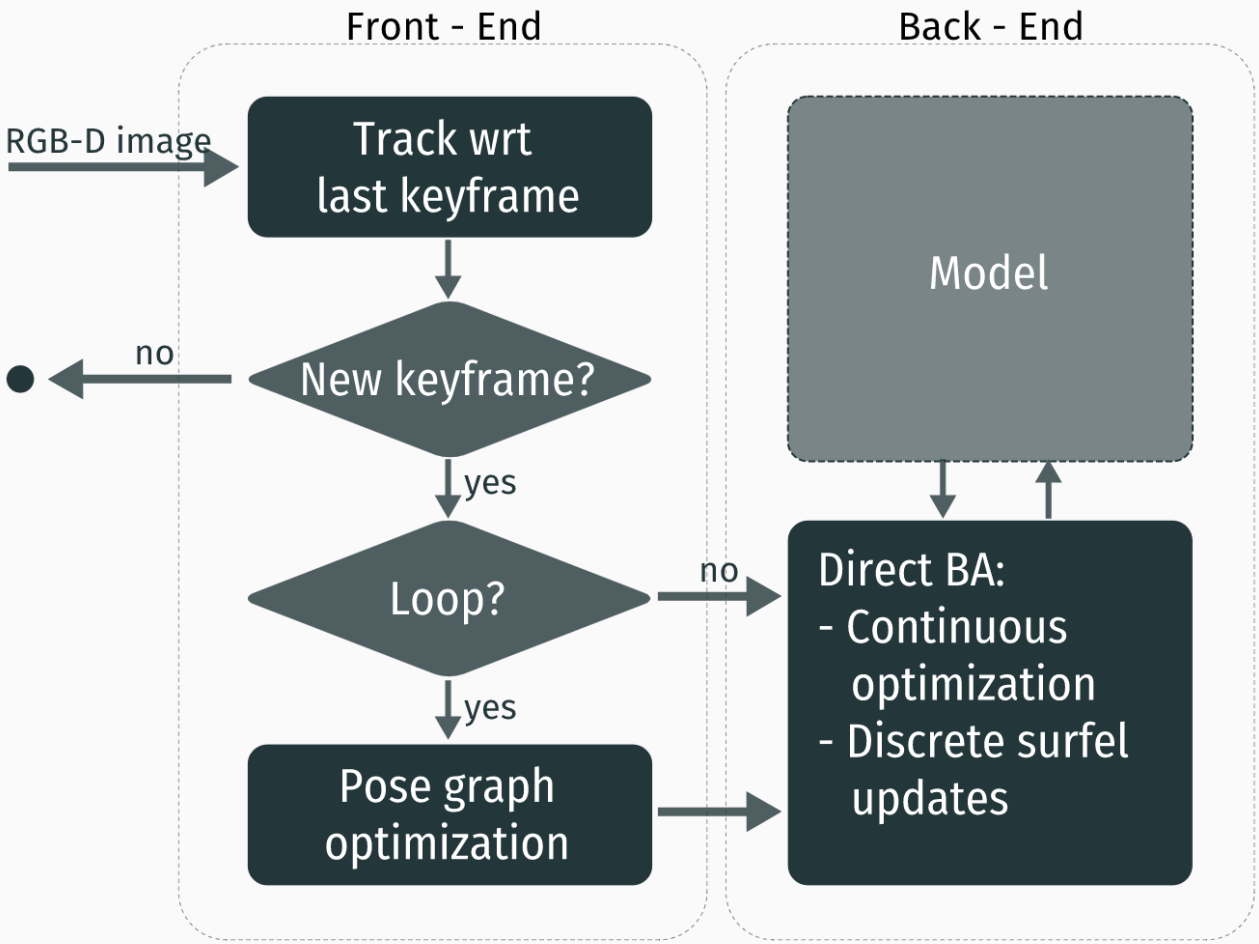
↓

These problems are easier to fix
on the hardware than model them out.

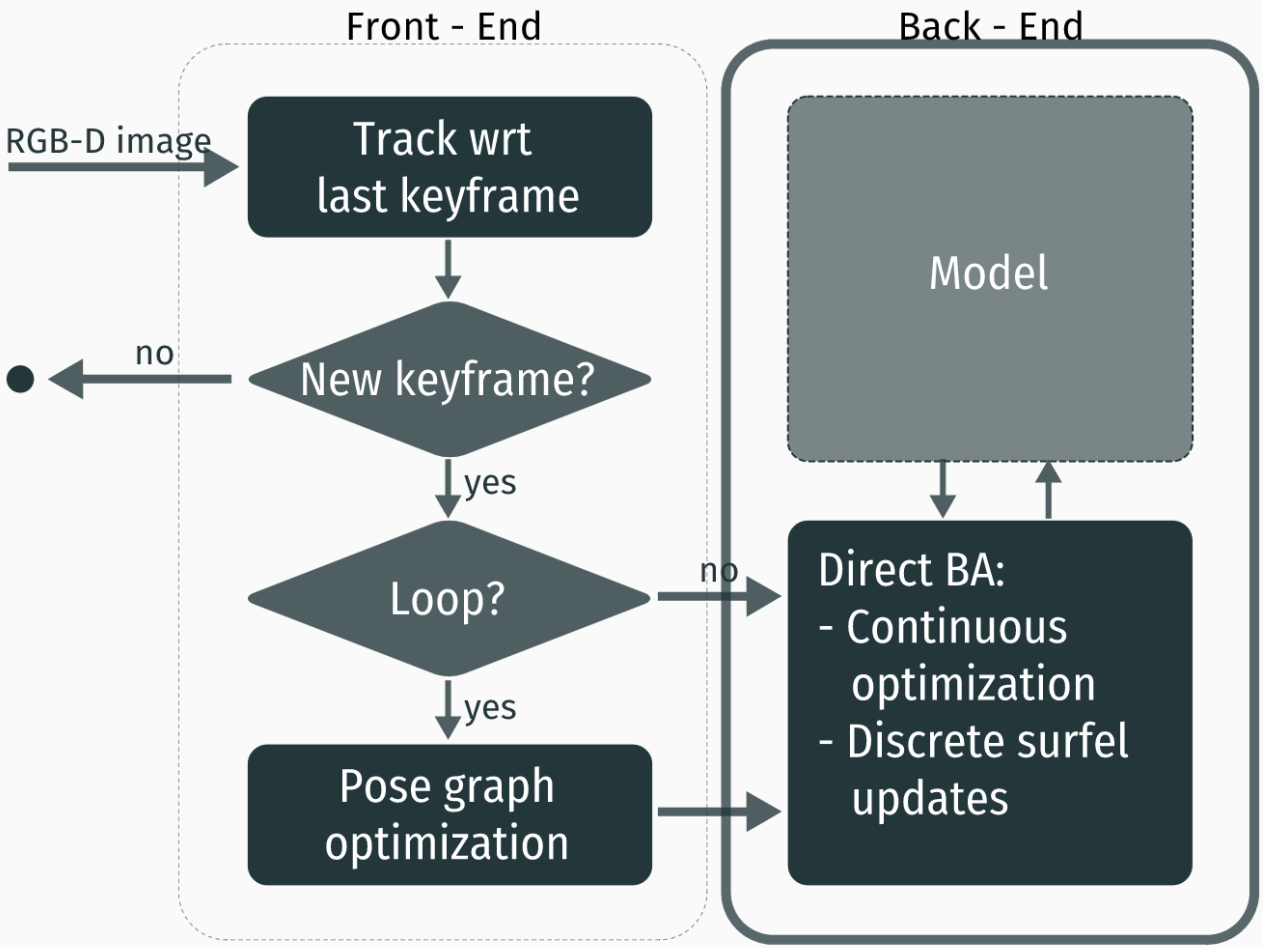
Table of contents

BAD SLAM: The algorithm

SLAM Architecture Overview



SLAM Architecture Overview



Model representation

Keyframe
is defined by an RGB-D frame and its 6 DoF camera pose.

Meshes:
+ arbitrary precision
- expensive topology update
- sensitive to noise

Voxels:
+ resilient to noise
+ efficient topology update
- cannot represent thin surfaces

Surfels:
+ arbitrary precision
+ efficient topology update
- sensitive to noise

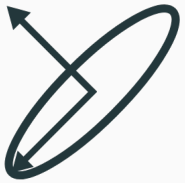
Model representation

Keyframe
is defined by an RGB-D frame and its 6 DoF camera pose.

Meshes:
+ arbitrary precision
- expensive topology update
- sensitive to noise

Voxels:
+ resilient to noise
+ efficient topology update
- cannot represent thin surfaces

Surfels:
+ arbitrary precision
+ efficient topology update
- sensitive to noise



Cost function definition

Cost function

The cost is computed by projecting each surfel into each keyframe to establish correspondences with pixel measurements.

$$C(K,S) = \sum_{k}^{K} \sum_{s}^{S_k} \underbrace{\rho_{Tukey}(\sigma_D^{-1} r_{geom}(s,k))}_{\text{geometric constraint}} + \overbrace{\omega}^{10^{-2}} \underbrace{\rho_{Huber}(\sigma_p^{-1} r_{photo}(s,k))}_{\text{photometric constraint}}$$

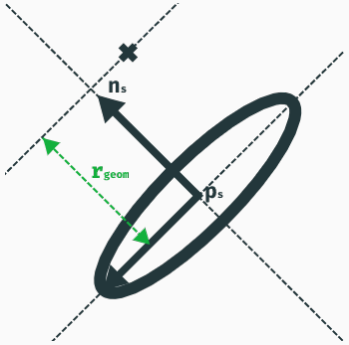
Geometric constraint

$$C(K,S) = \sum_k^K \sum_s^{S_k} \rho_{Tukey}(\sigma_D^{-1} r_{geom}(s,k)) + \omega \rho_{Huber}(\sigma_p^{-1} r_{photo}(s,k))$$

Geometric residual

The distance between the surfel and measurement positions along the surfel's normal direction.

$$r_{geom}(s,k) = (\mathbf{T}_G^k \mathbf{n}_s)^T (\pi_{D,k}^{-1}(\hat{\pi}_{D,k}(\mathbf{T}_G^k \mathbf{p}_s)) - \mathbf{T}_G^k \mathbf{p}_s)$$



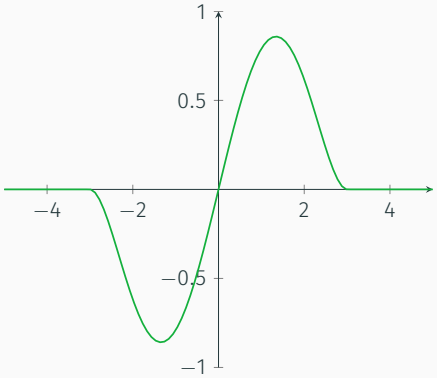
Normalisation factor σ_D^{-1}

Uncertainty propagation of the measurement uncertainty in the depth direction.

Geometric constraint

$$C(K,S) = \sum_k^K \sum_s^{S_k} \rho_{Tukey}(\sigma_D^{-1} r_{geom}(s,k)) + \omega \rho_{Huber}(\sigma_p^{-1} r_{photo}(s,k))$$

Tukey biweight loss function



$$\rho_{Tukey,\delta}(x) = \begin{cases} x\left(1 - \frac{x^2}{\delta^2}\right)^2 & \text{for} |x| < \delta \\ 0 & \text{for} |x| > \delta \end{cases}$$

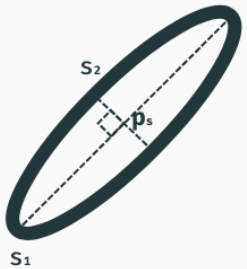
Photometric constraint

$$C(K,S) = \sum_k^K \sum_s^{S_k} \rho_{Tukey}(\sigma_D^{-1} r_{geom}(s,k)) + \omega \rho_{Huber}(\sigma_p^{-1} r_{photo}(s,k))$$

Photometric residual

Comparing the surfel's descriptor d_s to the surfel's projection into the image.

$$r_{photo}(s,k) = \left\| \begin{pmatrix} I(\pi_{I,k}(\mathbf{s}_1)) - I(\pi_{I,k}(\mathbf{p}_s)) \\ I(\pi_{I,k}(\mathbf{s}_2)) - I(\pi_{I,k}(\mathbf{p}_s)) \end{pmatrix} \right\|_2 - d_s$$



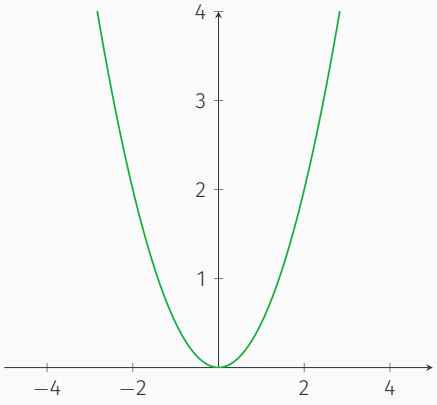
Normalisation factor σ_p^{-1}

It's hard to model reflections and illumination changes $\rightarrow \sigma_p^{-1} = \frac{1}{180}$

Photometric constraint

$$C(K,S) = \sum_k^K \sum_s^{S_k} \rho_{Tukey}(\sigma_D^{-1} r_{geom}(s,k)) + \omega \rho_{Huber}(\sigma_p^{-1} r_{photo}(s,k))$$

Huber robust loss function



$$\rho_{Huber,\delta}(x) = \begin{cases} \frac{1}{2}x^2 & \text{for } |x| \leq \delta \\ \delta\left(|x| - \frac{1}{2}\delta\right) & \text{otherwise} \end{cases}$$

Surfel-based alternating direct BA

Create new surfels for all keyframes

Update surfel normals $\mathbf{n}_s$

Optimize surfel positions $\mathbf{p}_s$ and descriptors d_s

Optimize keyframe poses

Optimize camera intrinsics

Merge similar surfels

Clean up surfels and update radii

Surfel-based alternating direct BA

Create new surfels for all keyframes

```
if (no pixel in 4x4 cell correspond to a surfel):
    randomly choose one depth measurement to create s;
    p_s = T_k^G π_{D,k}^{-1}(p);
    n_s = centered finite differences on depth img;
    r_s = min (p_s - 3D points of 4-neighbothood of p);
    d_s = r_photo(s, k);
```

Update surfel normals $\mathbf{n}_s$

Optimize surfel positions $\mathbf{p}_s$ and descriptors d_s

Optimize keyframe poses

Optimize camera intrinsics

Merge similar surfels

Clean up surfels and update radius

Surfel-based alternating direct BA

Create new surfels for all keyframes

Update surfel normals $\mathbf{n}_s$

For efficiency reasons:
→ `Average the normals of all corr. measurements;`
→ `Renormalize to unit vectors;`

Optimize surfel positions $\mathbf{p}_s$ and descriptors d_s

Optimize keyframe poses

Optimize camera intrinsics

Merge similar surfels

Clean up surfels and update radius

Surfel-based alternating direct BA

Create new surfels for all keyframes

Update surfel normals $\mathbf{n}_s$

Optimize surfel positions $\mathbf{p}_s$ and descriptors d_s

Position param: A surfel moves only along $\mathbf{n}_s \rightarrow$ optimize $\mathbf{p}_s + t\mathbf{n}_s$ for t.

Jointly optimize $\mathbf{p}_s$ and d_s with a Gauss-Newton iteration of the cost function.

Optimize keyframe poses

Optimize camera intrinsics

Merge similar surfels

Clean up surfels and update radius

Surfel-based alternating direct BA

Create new surfels for all keyframes

Update surfel normals $\mathbf{n}_s$

Optimize surfel positions $\mathbf{p}_s$ and descriptors d_s

Optimize keyframe poses

Pose update: $\epsilon \in \mathfrak{se}(3)$

Transformation param: T_G^k is updated as $T_k^g \cdot exp(\hat{\epsilon})$

Jointly optimize $\forall k$ poses with a Gauss-Newton iteration of the cost function.

Optimize camera intrinsics

Merge similar surfels

Clean up surfels and update radius

Surfel-based alternating direct BA

Create new surfels for all keyframes

Update surfel normals $\mathbf{n}_s$

Optimize surfel positions $\mathbf{p}_s$ and descriptors d_s

Optimize keyframe poses

Optimize camera intrinsics

Merge similar surfels

Merge surfels with similar attributes:

```
project surfels into all keyframes
if (projected to the same cell):
    consider for merging
```

Clean up surfels and update radius

Surfel-based alternating direct BA

Create new surfels for all keyframes

Update surfel normals $\mathbf{n}_s$

Optimize surfel positions $\mathbf{p}_s$ and descriptors d_s

Optimize keyframe poses

Optimize camera intrinsics

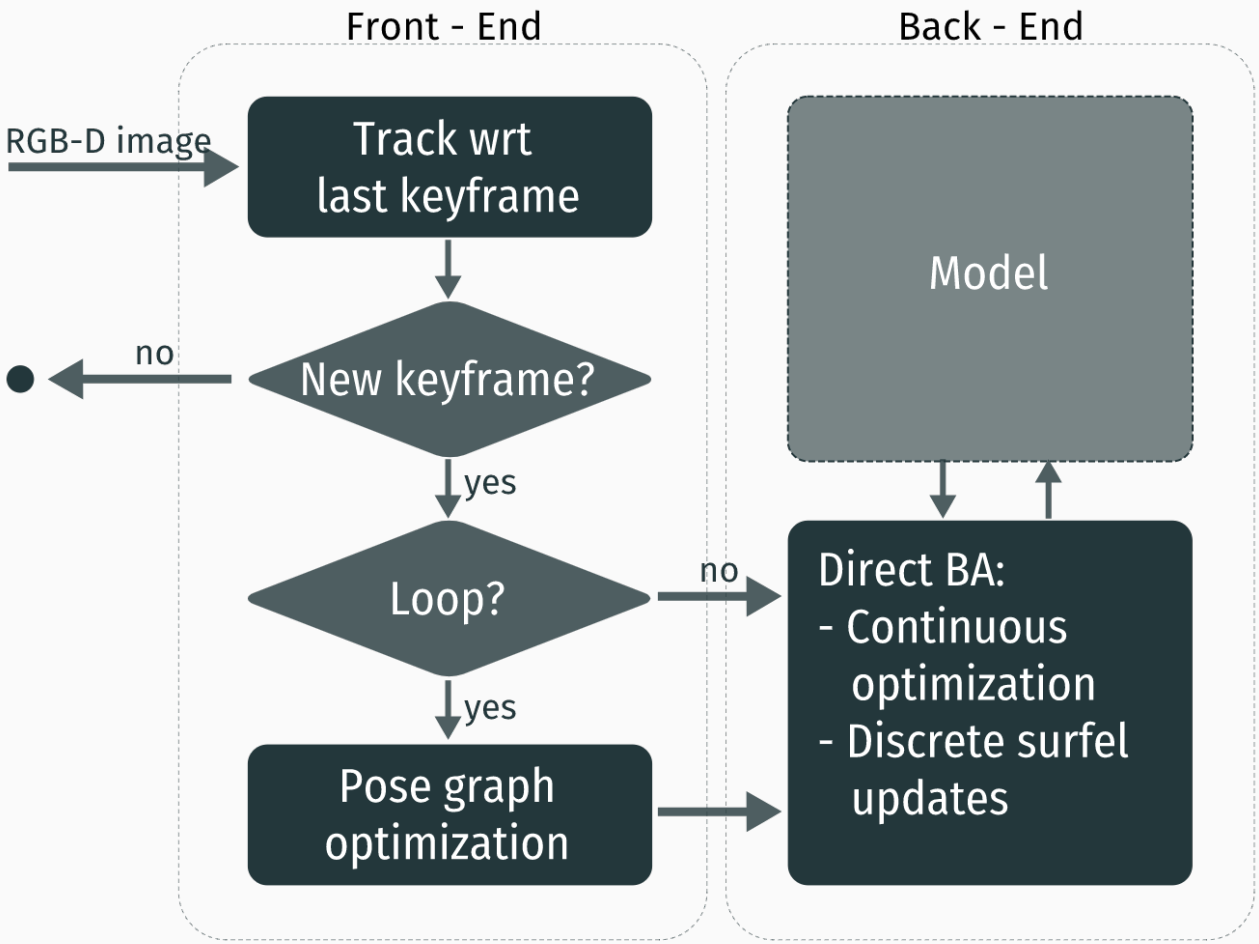
Merge similar surfels

Clean up surfels and update radius

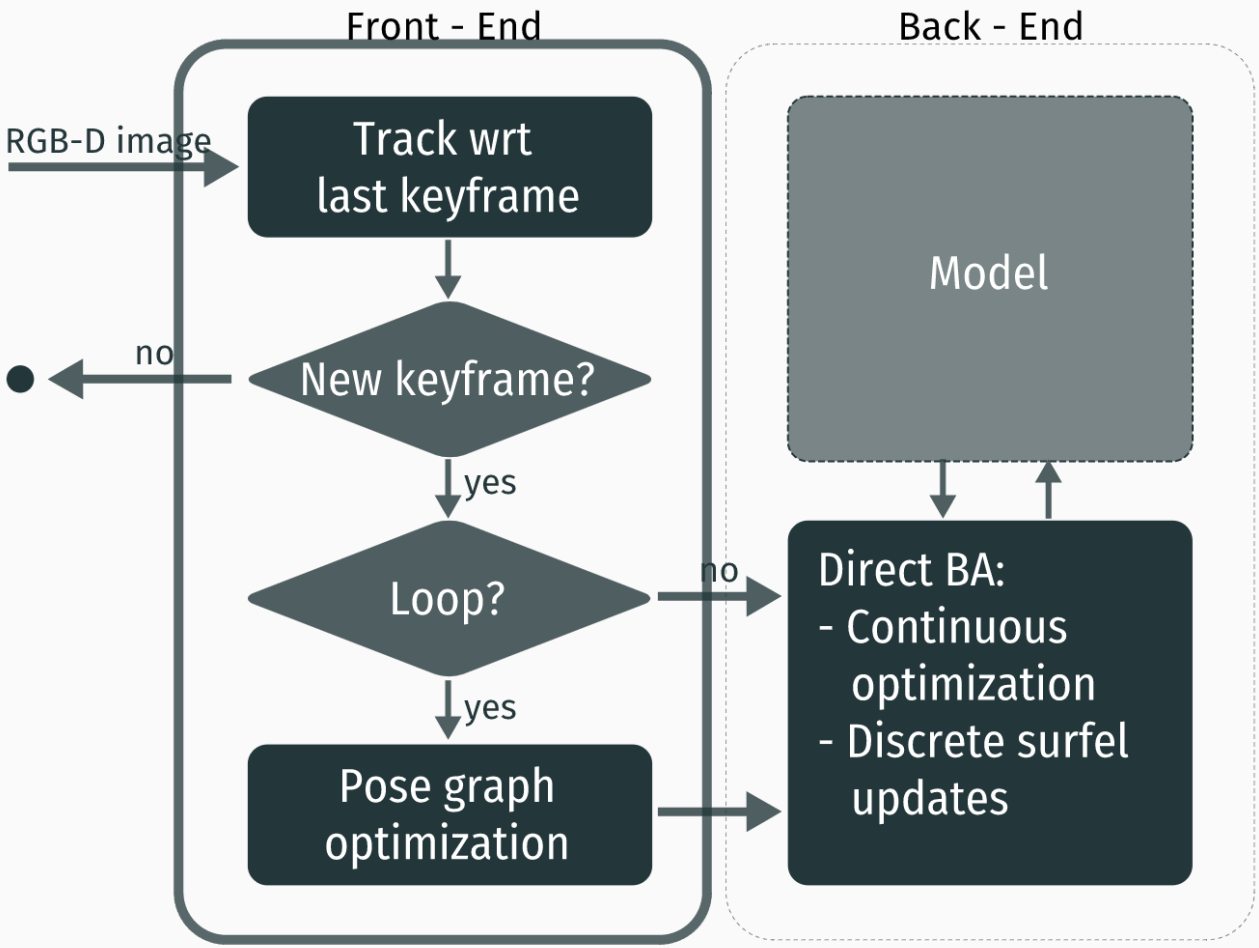
→ `filter outlier surfels`

→ `radius = min radius of all its corr. measurements`

SLAM Architecture Overview



SLAM Architecture Overview



SLAM Front-End

Preprocessing

→ `apply bilateral filter to smooth depth map`
→ `remove large depth measurements`

Odometry

Estimate new RGB-D frame pose relative to the last keyframe:
→ `apply standard direct photometric and geometric image alignment in` $SE(3)$

Keyframe selection

It's not adressed in the paper:

→ `select every` 10th `frame as a keyframe`

Loop closure

Identify the keyframe m most similar to the lastest keyframe k:
→ `extract binary independent elementary features`
→ `apply bag-of-words approach`

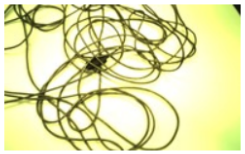
New benchmark dataset

Benchmark dataset

Main features of the new dataset:

- **Camera:** global shutter, well-calibrate, synchronized frames
- **Datasets:** 61 training, 35 test.
- **Benchmark:** non-public ground truth, an online leaderboard.

Training datasets

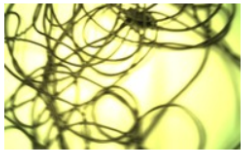


cables_1 - 1180 frames
The camera views some cable clutter on a table. This arrangement may make it hard to detect reliable features in the RGB images.

cables_1_mono.zip
cables_1_stereo.zip
cables_1_rgbd.zip
cables_1_imu.zip
cables_1_raw.zip

cables_2 - 98 frames
The camera views some cable clutter on a table while moving quickly. This arrangement may make it hard to detect reliable features in the RGB images.

cables_2_mono.zip
cables_2_stereo.zip
cables_2_rgbd.zip
cables_2_imu.zip
cables_2_raw.zip



cables_3 - 313 frames
The camera views some cable clutter on a table while moving quickly. This arrangement may make it hard to detect reliable features in the RGB images.

cables_3_mono.zip
cables_3_stereo.zip
cables_3_rgbd.zip
cables_3_imu.zip
cables_3_raw.zip

camera_shake_1 - 318 frames
The camera is shaken quickly. Using IMU data is probably very helpful here.

camera_shake_1_mono.zip
camera_shake_1_stereo.zip
camera_shake_1_rgbd.zip
camera_shake_1_imu.zip

Performance

Performance of different components

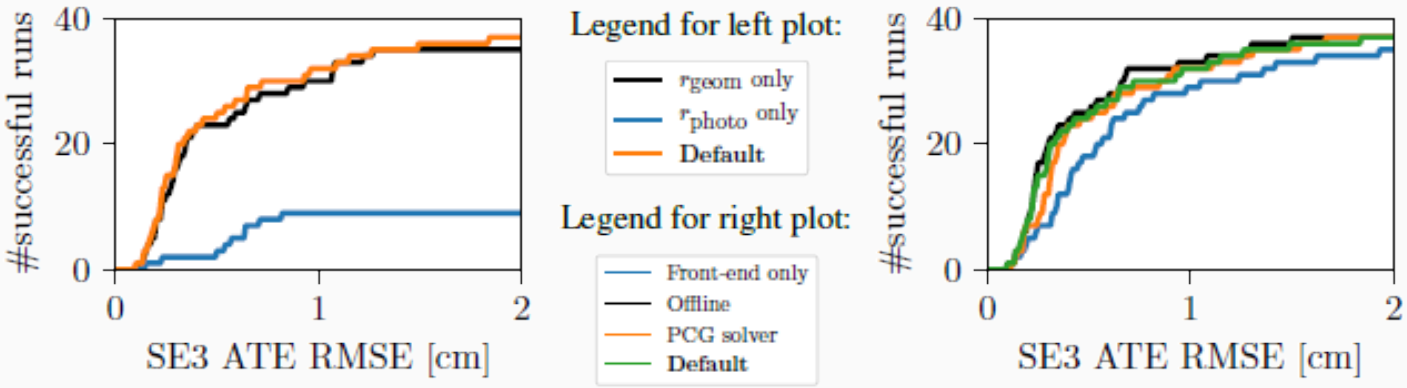


Figure 1: Ablation study

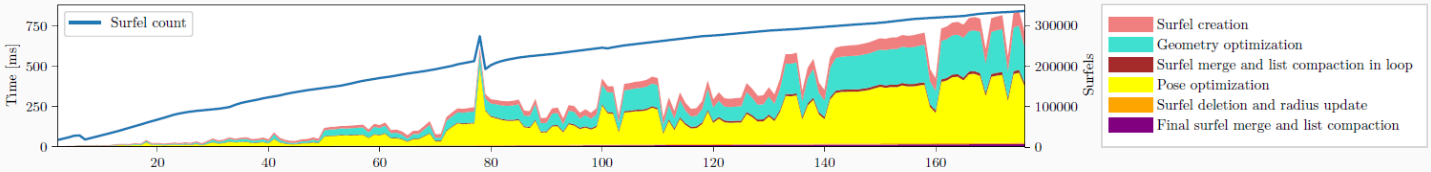


Figure 2: Runtime of the BA scheme

Performance comparison

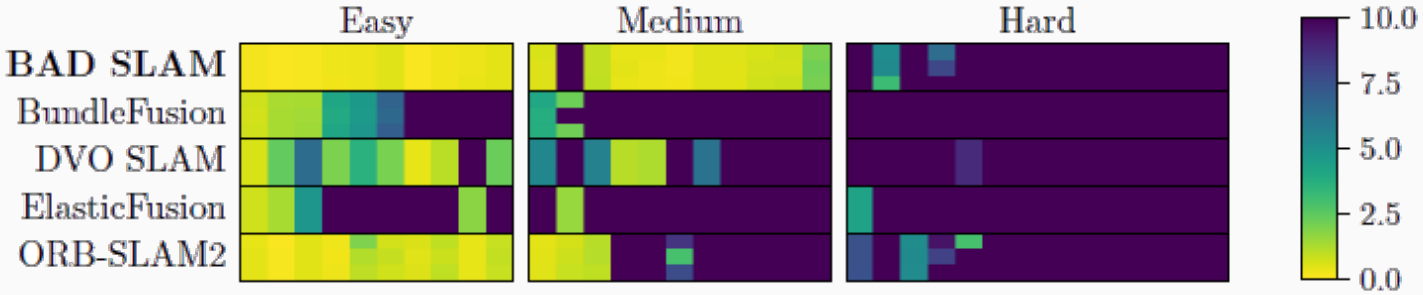


Figure 3: Evaluation results on new benchmark (ATE RMSE)

Rank	Error	Algorithm
1	1.0	ORB-SLAM2
2	2.7	BundleFusion
2	2.7	BAD SLAM
3	3.0	ElasticFusion
4	6.3	DVO SLAM

Table 1: Average ranking on TUM RGB-D benchmark (ATE RMSE)

Conclusion

Major contributions of the paper

- Real-time direct bundle adjustment method
- Algorithm with state-of-the-art performance
- New public benchmarkt dataset

Algorithm prerequisites

- Very specific camera requirements
- No rapid illumination changes, no moving objects in the environment etc.

References i

R. I. H. Bill Triggs, Philip F. McLauchlan and A. W. Fitzgibbon.
Bundle adjustment — a modern synthesis.
Vision Algorithms: Theory and Practice, 1883:298–372, 1999.

C. Cadena, L. Carlone, H. Carrillo, Y. Latif, D. Scaramuzza, J. Neira, I. Reid, and J. Leonard.
Past, present, and future of simultaneous localization and mapping: Towards the robust-perception age.
IEEE Transactions on Robotics, 32(6):1309–1332, 2016.

J. Engel, V. Koltun, and D. Cremers.
Direct sparse odometry.
CoRR, abs/1607.02565, 2016.

References ii

T. Schöps, T. Sattler, and M. Pollefeys.
BAD SLAM: Bundle adjusted direct RGB-D SLAM.
2019.