Dense Localization and Mapping

Jürgen Sturm

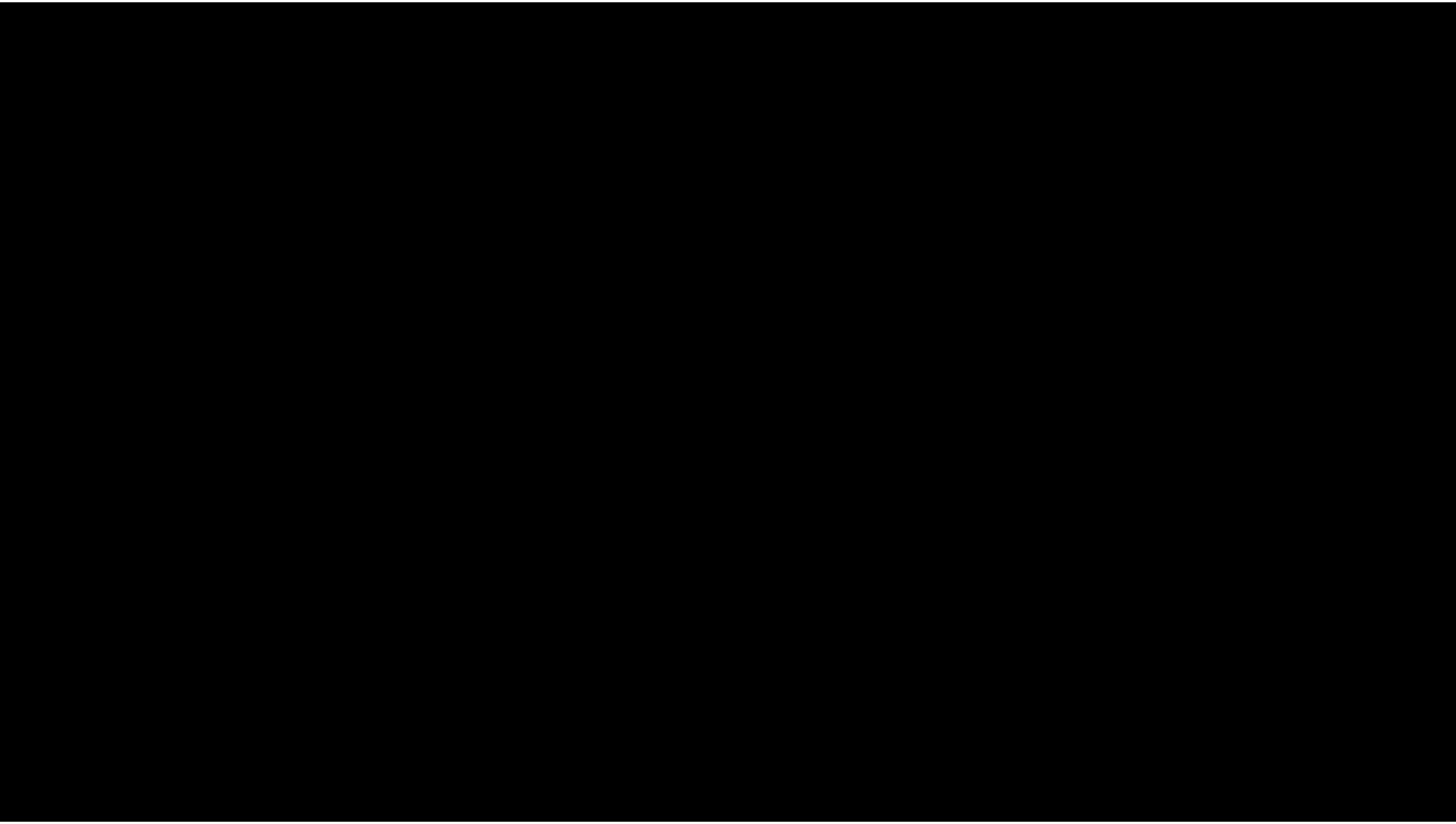
Joint work with Frank Steinbrücker, Jakob Engel, Christian Kerl, and Daniel Cremers

Introduction

- **(RGB-D) Cameras are rich sensors** that provide intensities, color, depth at video frame rates
- Lightweight and cheap
- **Many useful applications** in robotics: Localization, mapping, navigation, obstacle avoidance

Feature-based Visual Navigation

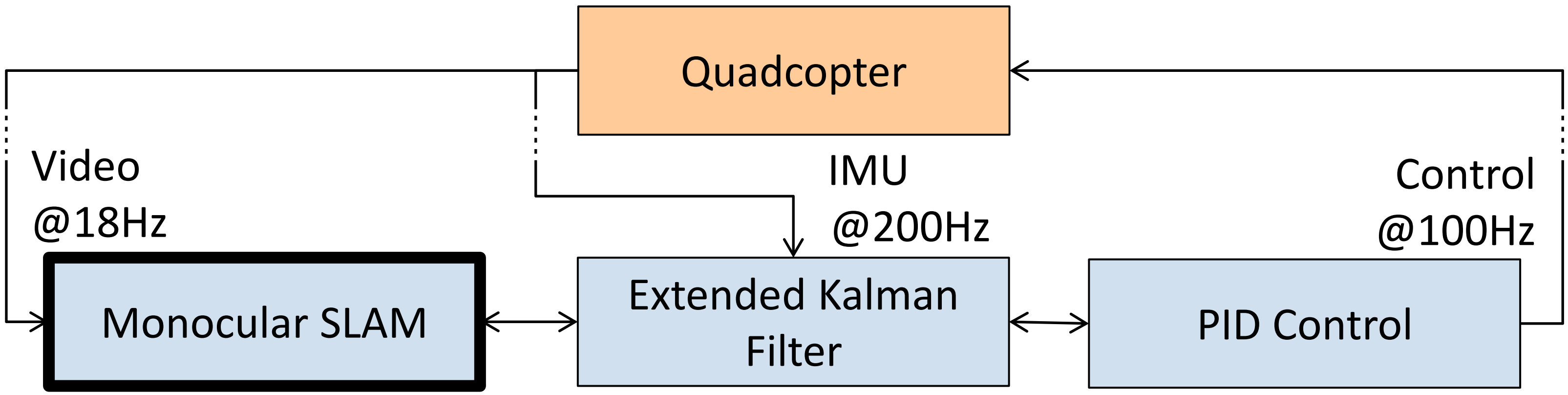
[Engel, Sturm, Cremers, IROS ‘12]



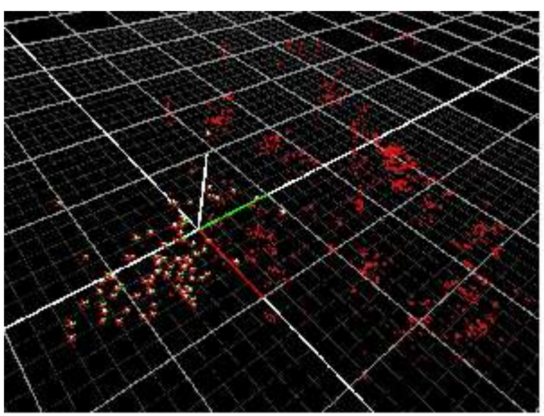
Feature-based Visual Navigation

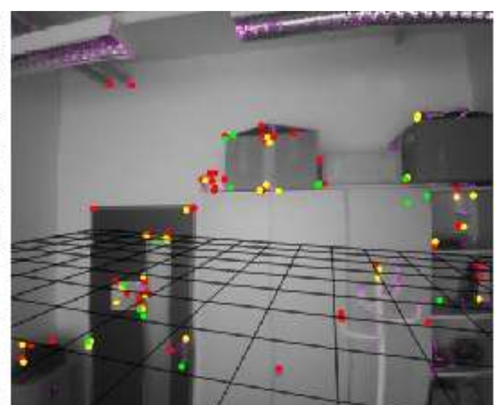
[Engel, Sturm, Cremers, IROS '12]

- Architecture



- Based on PTAM [Klein and Murray, ISMAR '07]



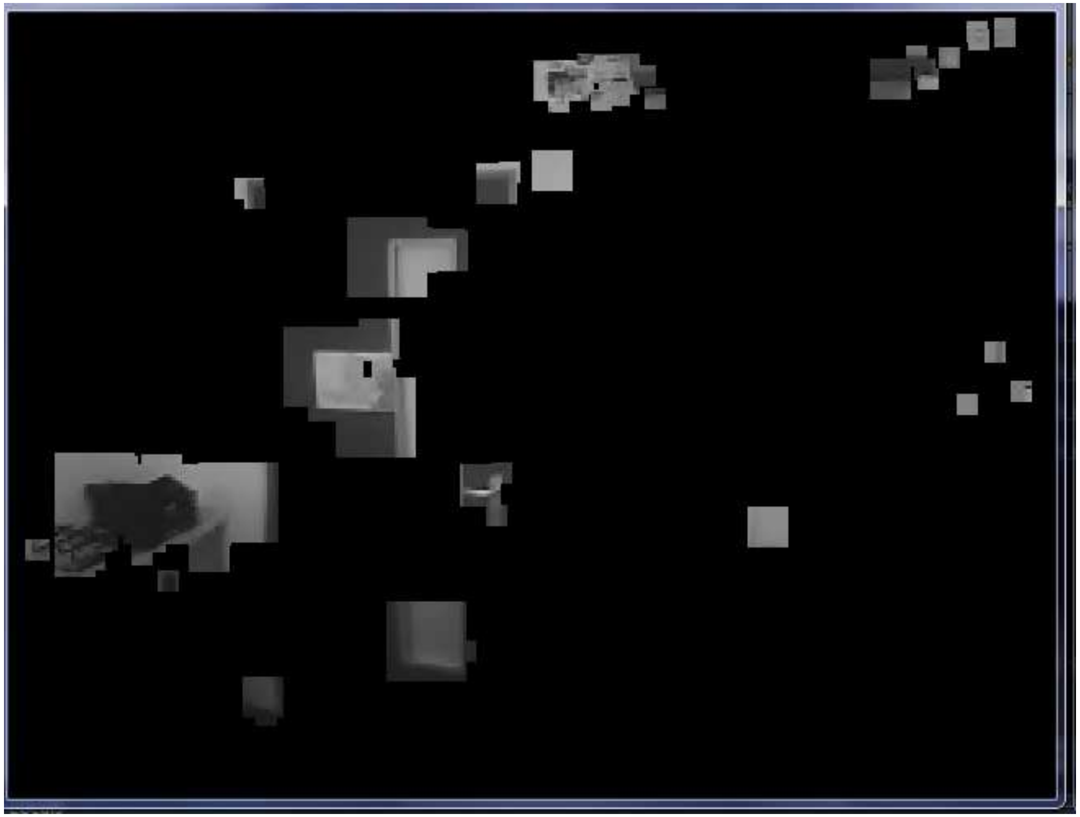


Motivation

- Video feed from quadrocopter

Motivation

- What PTAM actually sees



Motivation

- **Problem:** Most approaches only use a small fraction of the available data
  - Keypoint detection
  - Visual features
- **Question:** How can we use most/all information to maximize the performance?

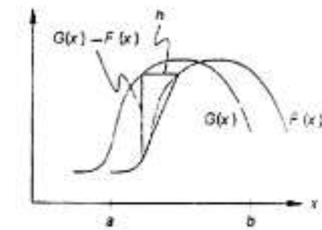
➡ In this talk: Dense methods for localization and mapping

Outline of the Talk

- Part 1: Dense tracking
- Part 2: Dense reconstruction
- Part 3: Evaluation and benchmarking

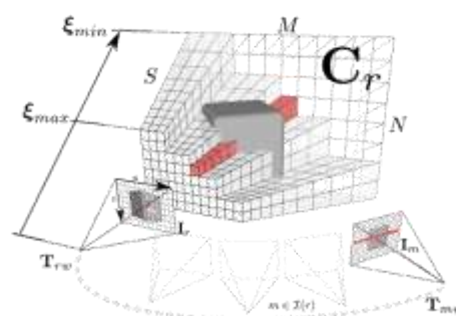
Related Work on Dense Tracking

- Lucas and Kanade (IJCAI’81)



- Lovegrove et al. (IV’11)

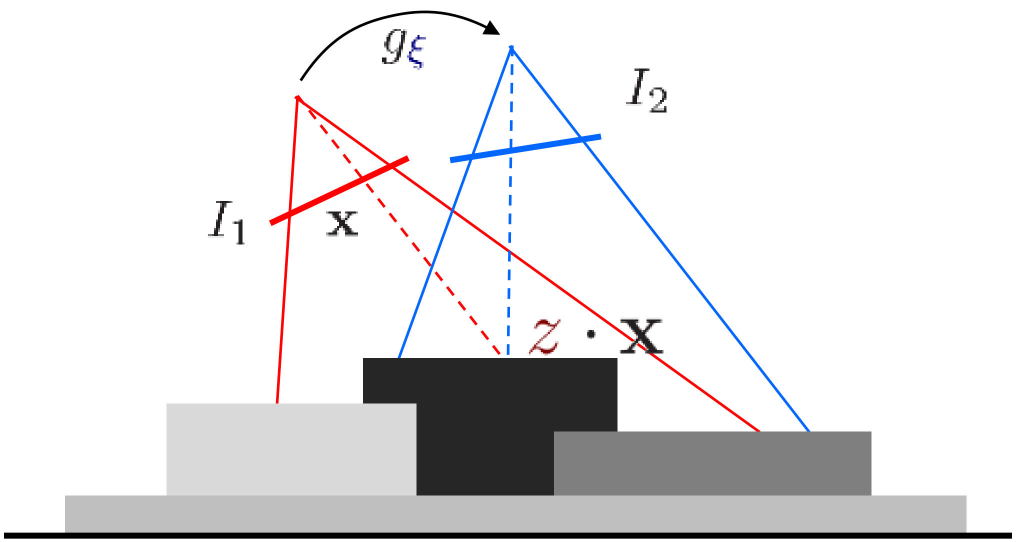
- Newcombe et al. (ICCV’11)



- Comport/Tykkälä et al. (ICCV’11)

Dense Tracking

- How can we exploit ALL data of the image?
- Idea



- Photo-consistency constraint

$$I_1(\mathbf{x}) = I_2\left(\pi(g_\xi(z \cdot \mathbf{x}))\right) \text{ for all pixels } \mathbf{x}$$

How to deal with noise?

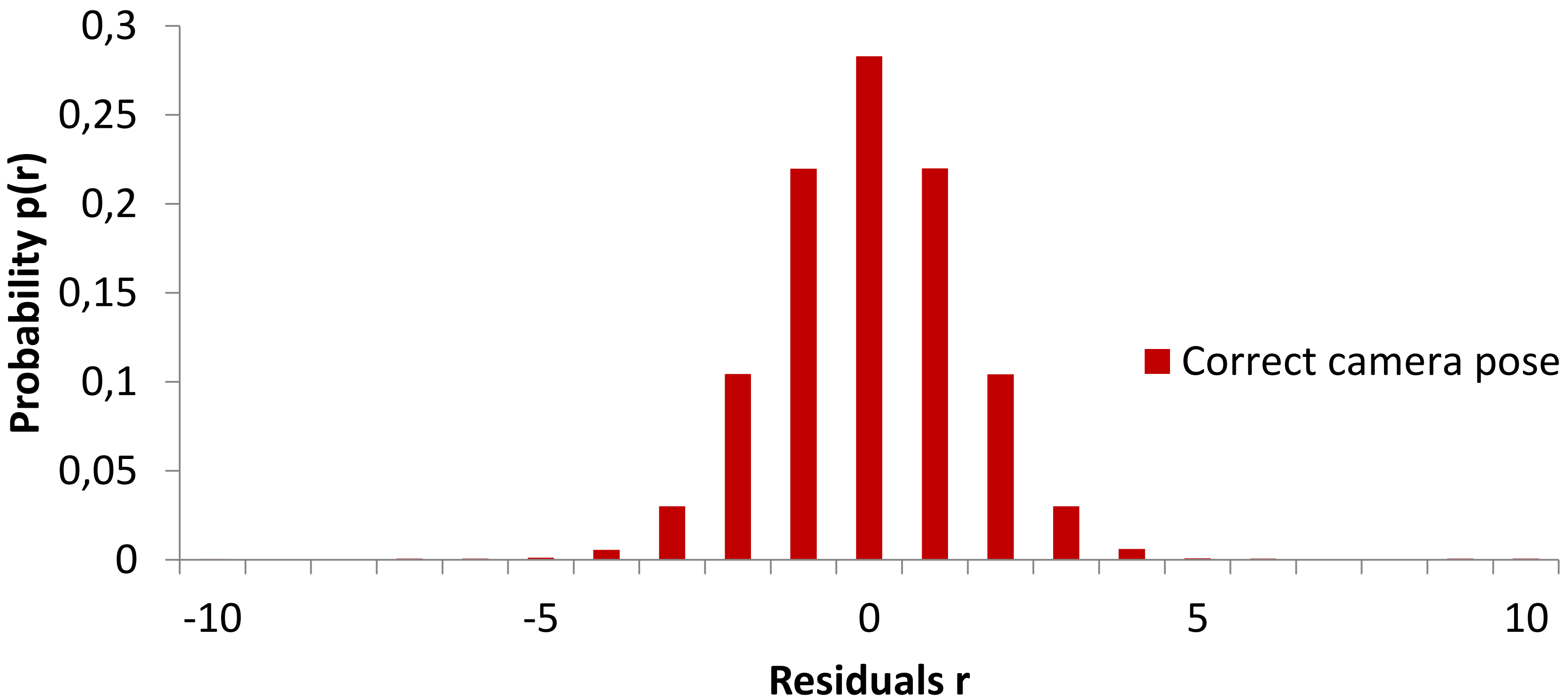
- Photo-consistency constraint will not perfectly hold
  - Sensor noise
  - Pose error
  - Reflections, specular surfaces
  - Dynamic objects (e.g., walking people)
- Residuals will be non-zero

$$r = I_1(\mathbf{x}) - I_2\left(\pi(g_\xi(z \cdot \mathbf{x})\right)$$

- Residual distribution $p(r)$

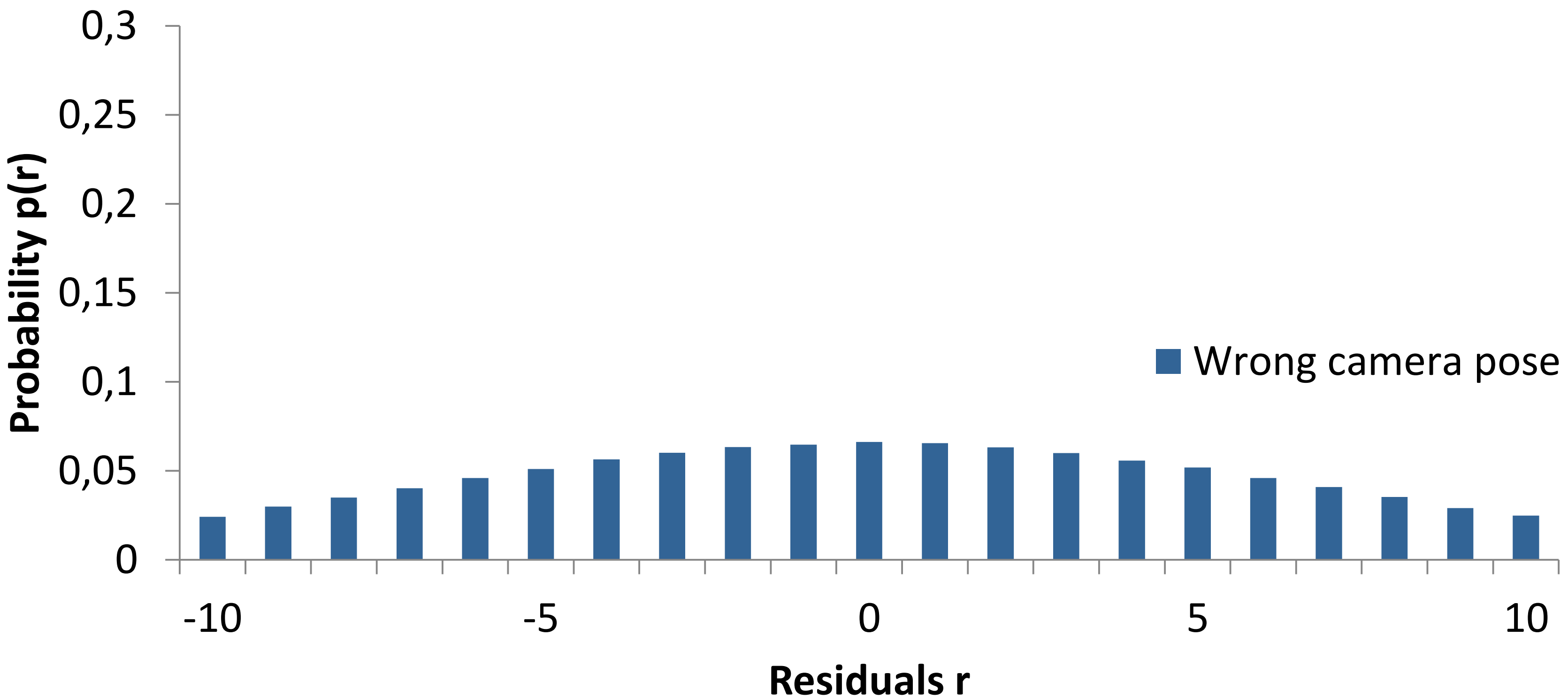
Residual Distribution

- Zero-mean, peaked distribution
- Example: Correct camera pose



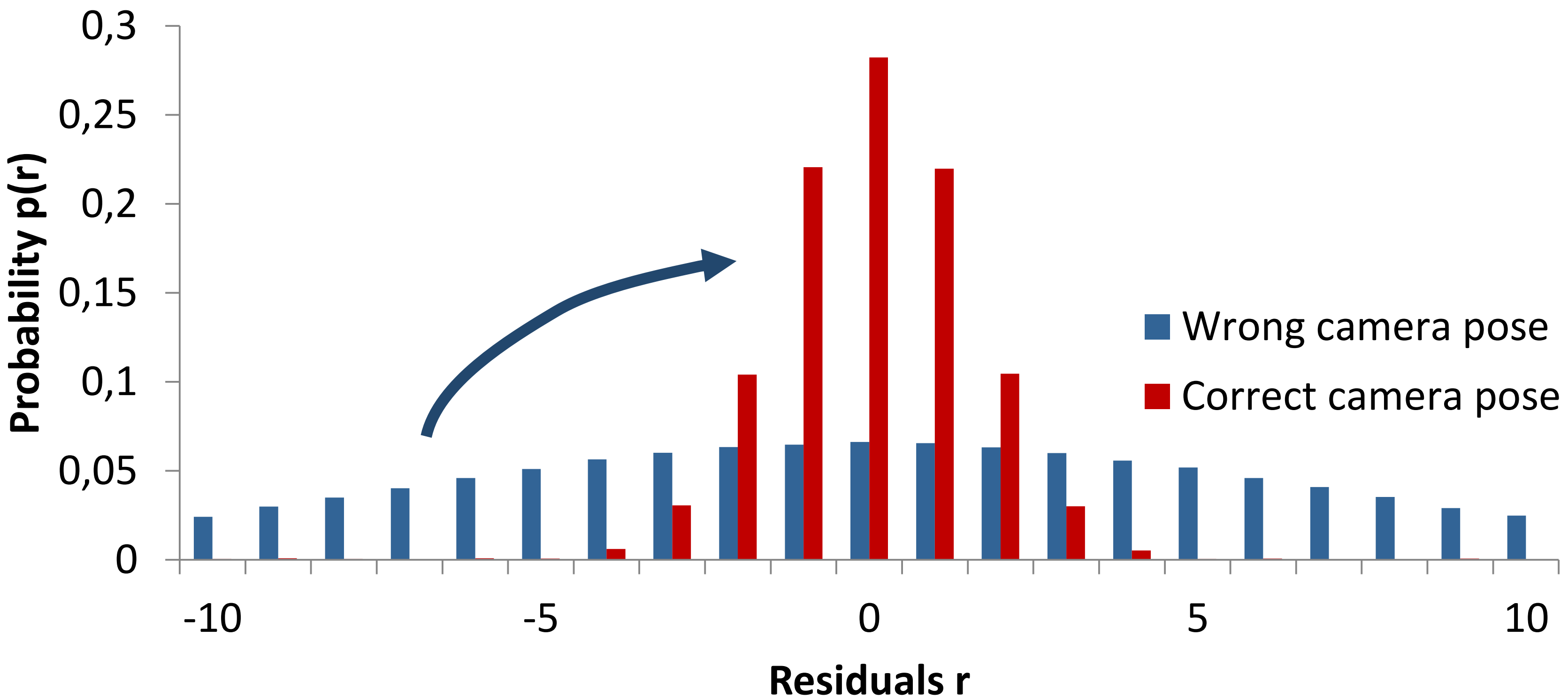
Residual Distribution

- Zero-mean, peaked distribution
- Example: Wrong camera pose

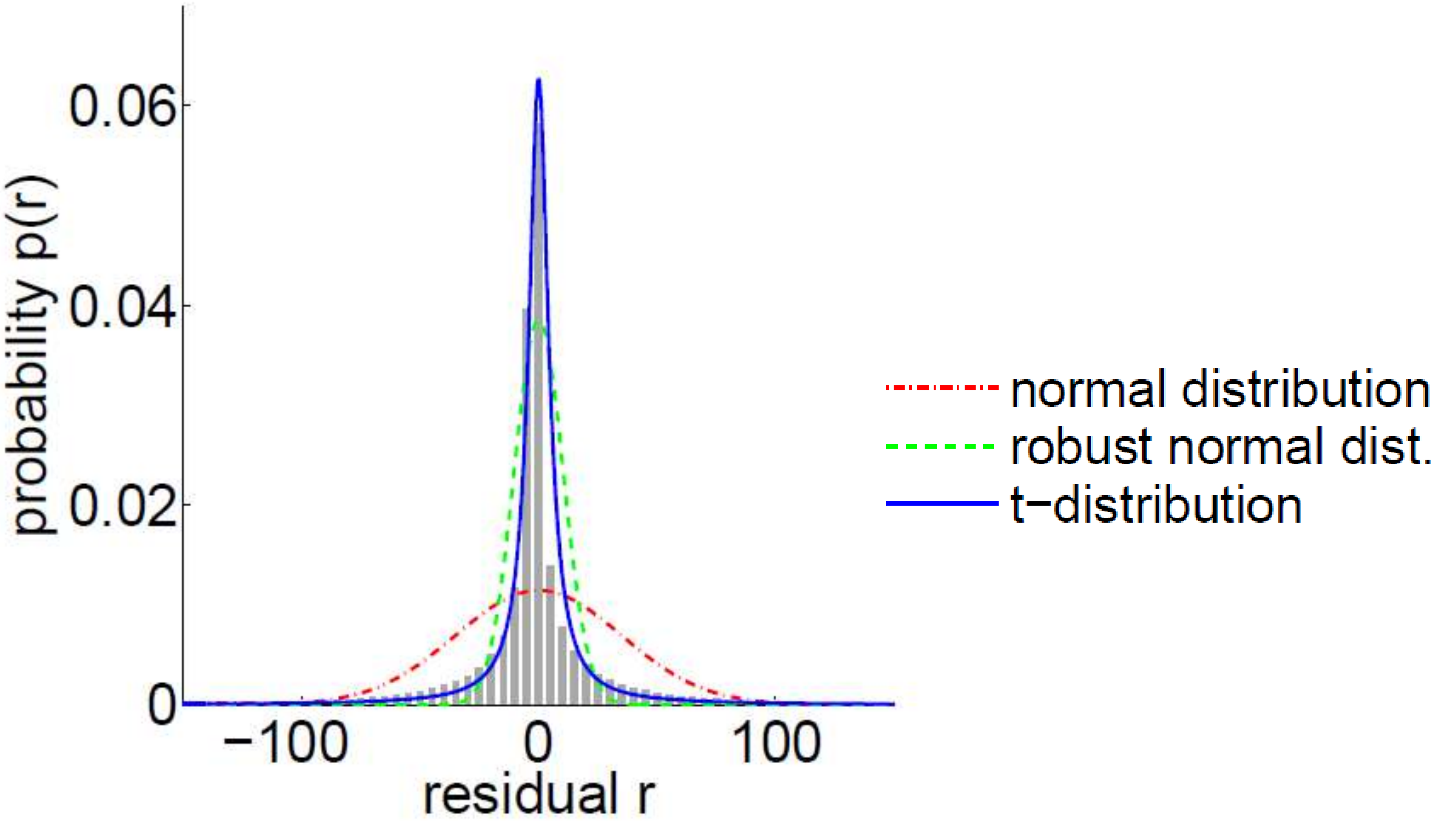


Residual Distribution

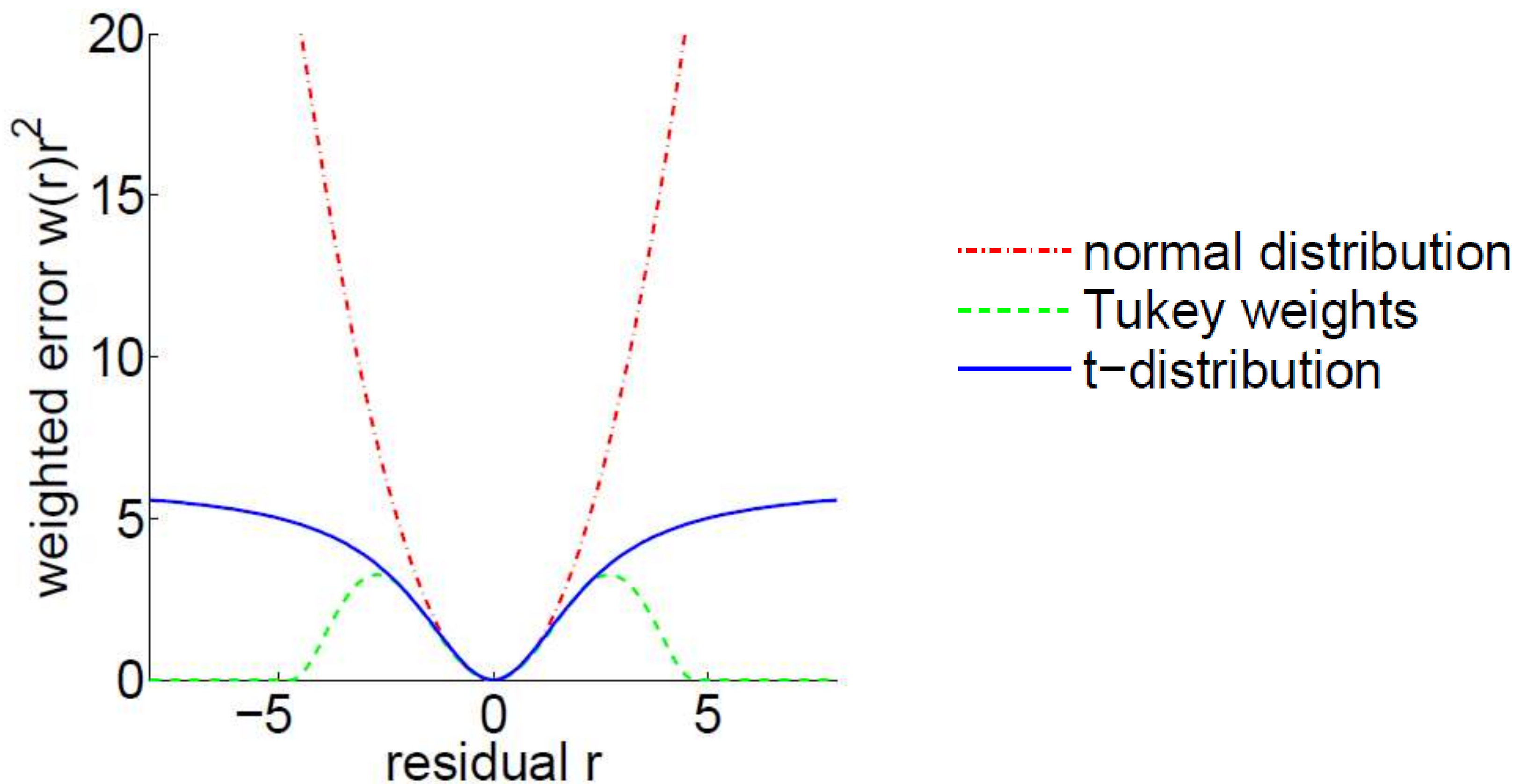
- **Goal:** Find the camera pose that maximizes the observation likelihood



What is a Good Model for the Residual Distribution?



Weighted Error

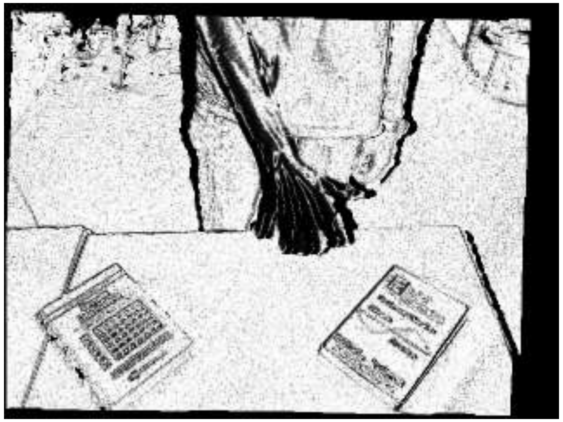


Example Weights

- Robust sensor model allows to down-weight outliers (dynamic objects, motion blur, reflections, …)

Scene

Residuals



Weights

Motion Estimation

- **Goal:** Find the camera pose that maximizes the observation likelihood

$$\xi^* = \arg\max_{\xi} \prod_i p(r_i(\xi))$$

compute over all pixels

- Assume pixel-wise residuals are conditionally independent
- How can we solve this optimization problem?

Example

First input image

Second input image

Residuals

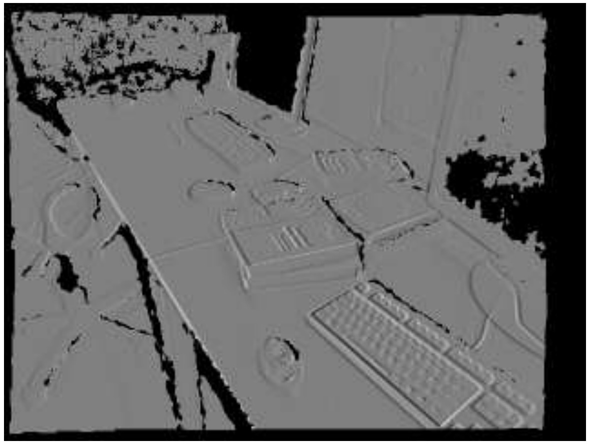


Image Jacobian for
Camera motion along x axis

Approach

- Take negative logarithm

$$\xi_{\text{MAP}} = \arg\min_{\xi} \sum_i -\log p(r_i(\xi))$$

- Set derivative to zero

$$\sum_i \frac{\partial \log p(r_i(\xi))}{\partial \xi} = \sum_i \frac{\partial \log p(r_i)}{\partial r_i} \frac{\partial r_i(\xi)}{\partial \xi} \stackrel{!}{=} 0$$

Approach (cont.d)

- This can be rewritten as a weighted least squares problem

$$\xi^* = \arg\min_{\xi} \sum_i w(r_i)(r_i(\xi))^2$$

with weights $w(r_i) = \frac{\partial \log p(r_i)}{\partial r_i} \frac{1}{r_i}$

- $r_i(\xi)$ is non-linear in ξ
- Need to linearize, solve, and iterate

Iteratively Reweighted Least Squares

Problem: $$\xi^* = \arg\min_{\xi} \sum_i w(r_i)(r_i(\xi))^2$$

Algorithm:

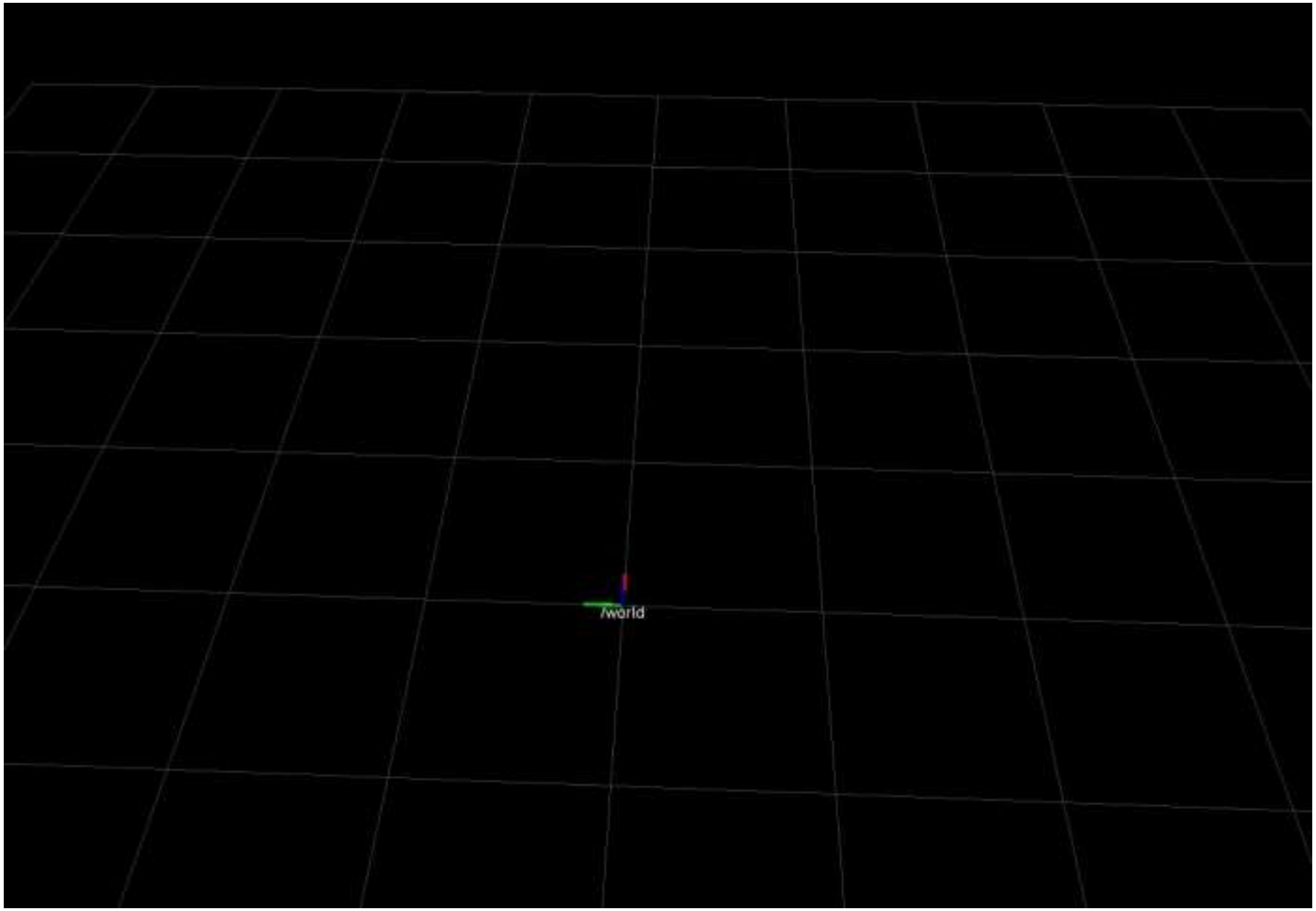
1. Compute weights $w(r_i) = \frac{\partial \log p(r_i)}{\partial r_i} \frac{1}{r_i}$
2. Linearize in the camera motion ξ
$$r_{\text{lin}}(\xi) = r(0) + J\Delta\xi$$
3. Build and solve normal equations
$$J^T W J \Delta\xi = -J^T W \mathbf{r}(\mathbf{0})$$
4. Repeat until convergence

Coarse-to-Fine

- Linearization only holds for small motions
- Coarse-to-fine scheme
- Image pyramids

Dense Tracking: Results

[Steinbrücker et al., ICCV LDRMC'11]

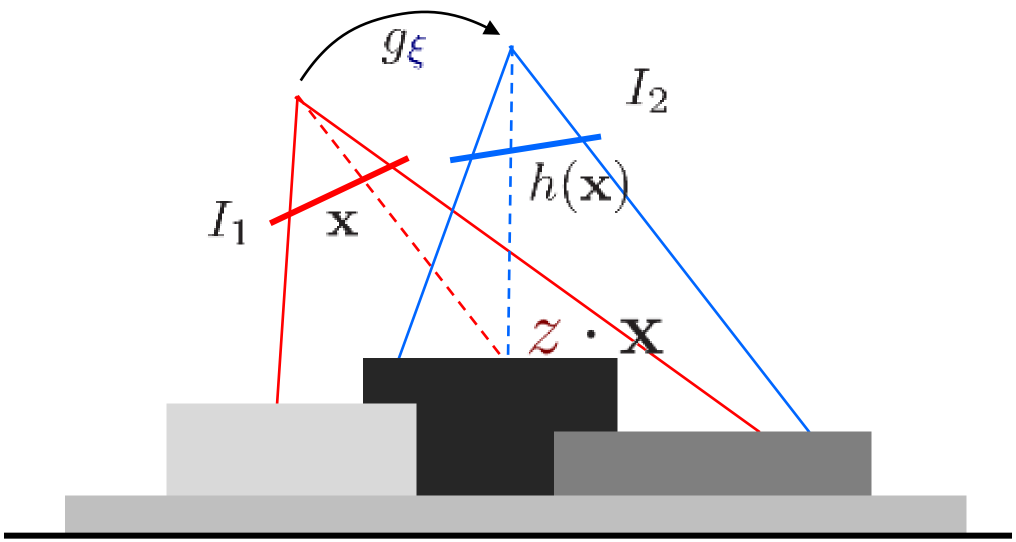


Summary: Dense tracking

- Pro
  - Super fast, highly accurate
  - Low memory consumption
- Con
  - Accumulates drift over time, sometimes diverges
- Next steps
  - Apply this method on the quadrocopter

Dense Reconstruction

- Can we use the same principle for 3D reconstruction?



- Photo-consistency constraint

$$I_1(\mathbf{x}) = I_2\left(\pi(g_\xi(z \cdot \mathbf{x}))\right) \text{ for all pixels } \mathbf{x}$$

Dense Reconstruction

- Dense tracking [Steinbrücker et al., ICCV LDRMC'11]
  - Given intensity images and depth maps
  - Estimate camera pose

$$\min_{\xi} \int_{\Omega} \left| I_1(\mathbf{x}) - I_2\left(\pi(g_{\xi}(z \cdot \mathbf{x})\right)\right|^2 \mathrm{d}\mathbf{x}$$

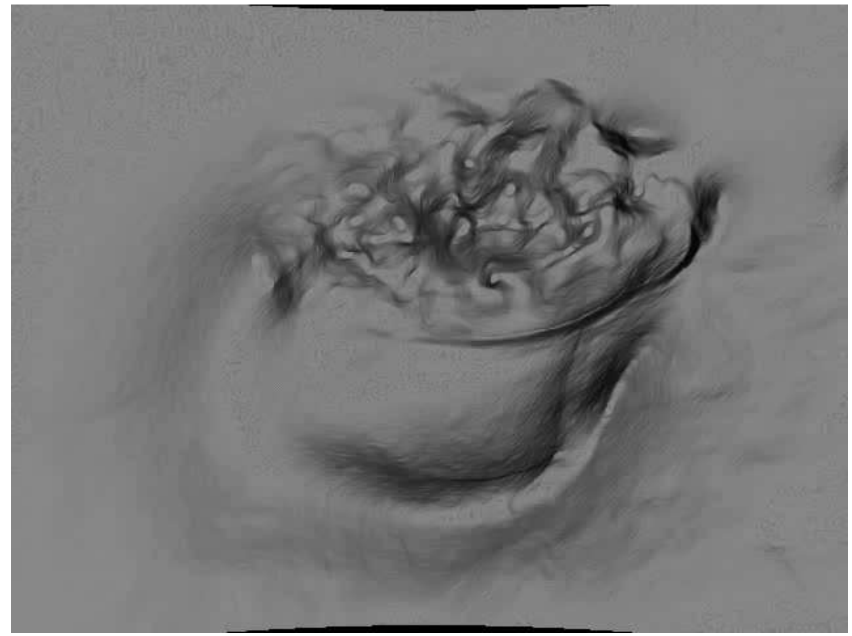
- Dense reconstruction [Stühmer et al., DAGM'10]
  - Given intensity images and camera poses
  - Estimate depth map

$$\min_{z} \int_{\Omega} \left| I_1(\mathbf{x}) - I_2\left(\pi(g_{\xi}(z \cdot \mathbf{x})\right)\right|^2 \mathrm{d}\mathbf{x}$$

Dense Reconstruction: Results

[Stühmer et al., DAGM'10]

Input: Intensity images + pose



Output: Estimated Geometry

Evaluation and Benchmarking

- How can we evaluate such methods?
- What are good evaluation criteria?
  - Accuracy of the estimated camera trajectory
  - Robustness to dynamic objects, noise, …
  - Accuracy of the 3D model

Existing Benchmarks

- Intel dataset: laser + odometry [Haehnel et al., 2004]
- New College dataset: stereo + omni-directional vision + laser + IMU [Smith et al., IJRR'2009]
- KITTI Vision benchmark: stereo [Geiger et al., CVPR'12]
- Our contribution: Dataset for RGB-D evaluation

Intel

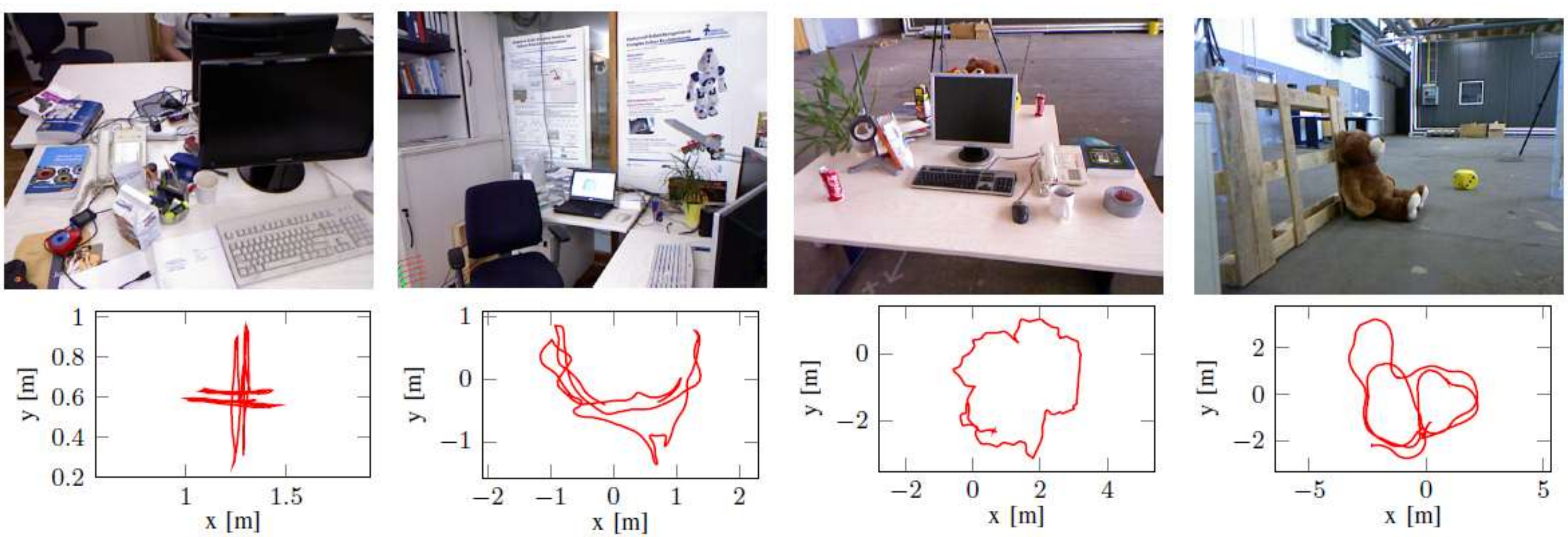
New College

KITTI

RGB-D

Recorded Scenes

- Different environments (office, industrial hall, …)
- Large variations in camera speed, camera motion, illumination, number of features, dynamic objects, …
- Handheld and robot-mounted sensor



(a) fr1/xyz (b) fr1/room (c) fr2/desk (d) fr2/slam

Dataset Acquisition

- Motion capture system
  - Camera pose (100 Hz)
- Microsoft Kinect (later: Asus Xtion Pro Live)
  - Color images (30 Hz)
  - Depth images (30 Hz)
- External video camera (for documentation)

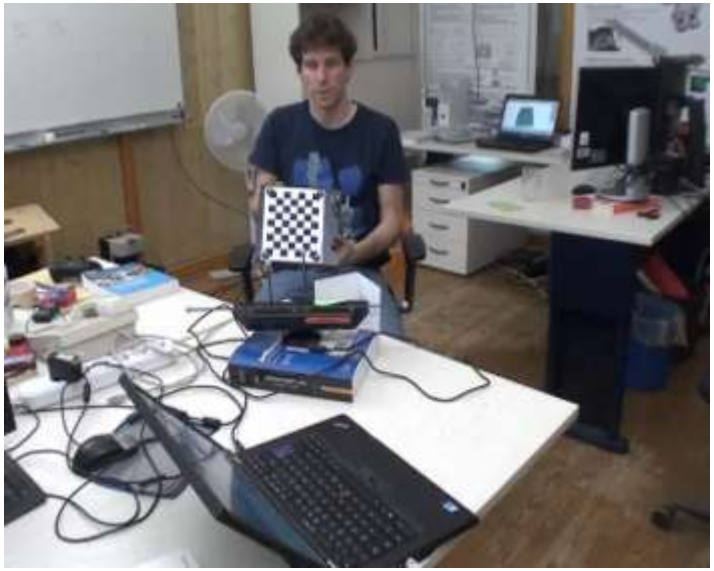
Motion Capture System

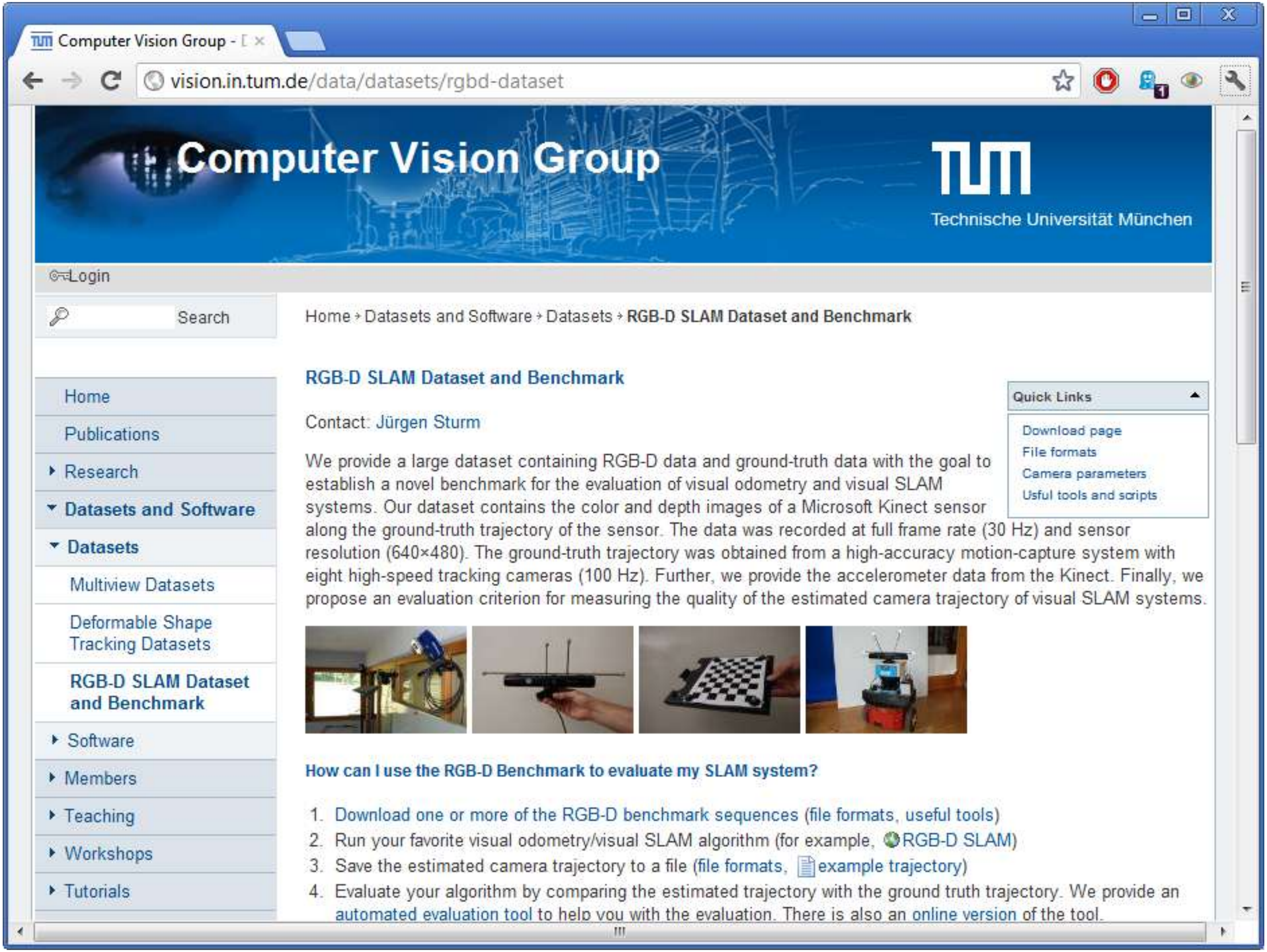
- 9 high-speed cameras mounted in room
- Cameras have active illumination and pre-process image (thresholding)
- Cameras track positions of retro-reflective markers

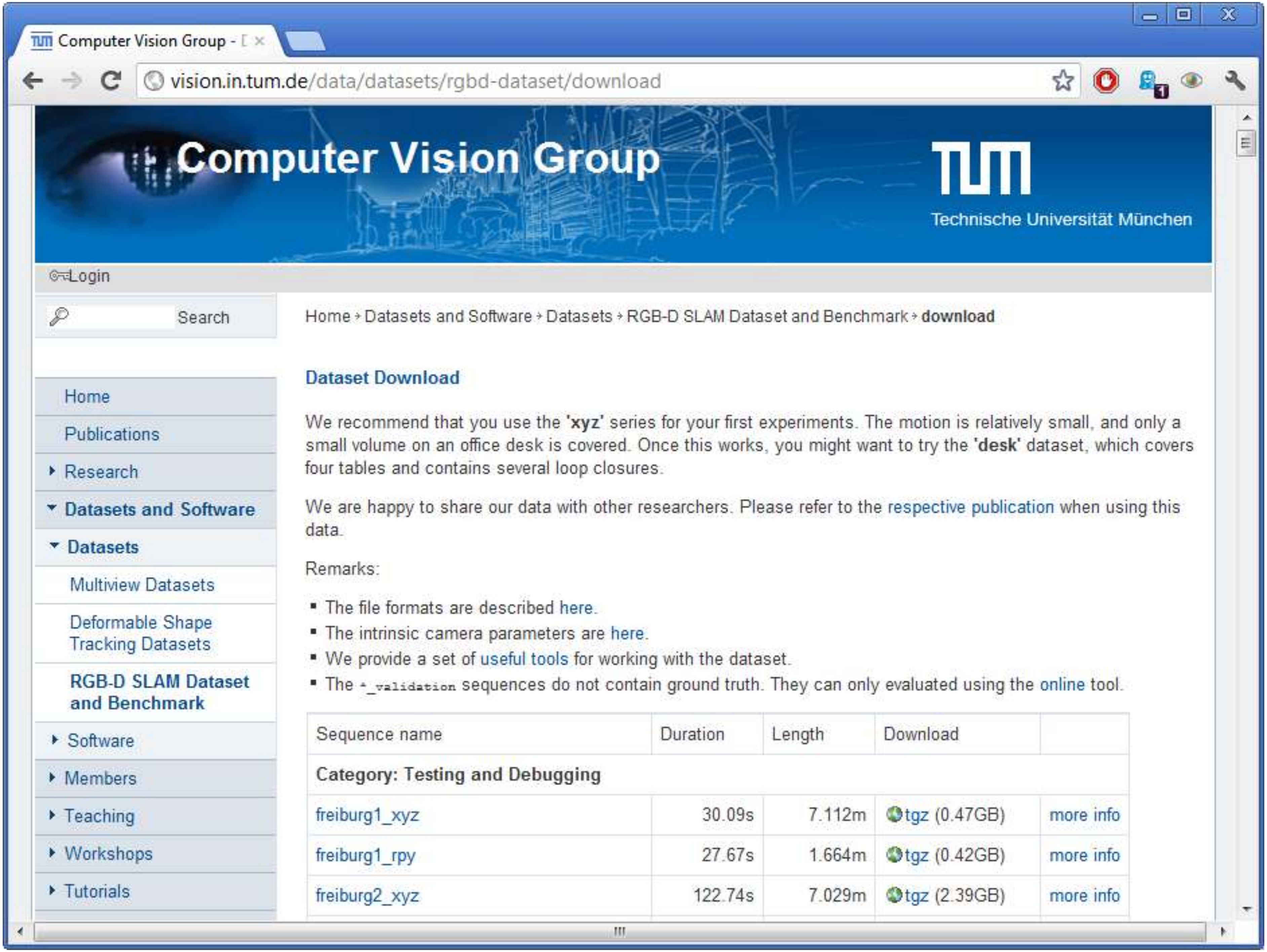
Calibration

Calibration of the overall system is not trivial:

1. Intrinsic calibration (Mocap + Kinect)
2. Extrinsic calibration (Kinect vs. Mocap)
3. Time synchronization (Kinect vs. Mocap)




Computer Vision Group - [
vision.in.tum.de/data/datasets/rgbd-dataset
Computer Vision Group
TUM
Technische Universität München
Login
Search
Home › Datasets and Software › Datasets › RGB-D SLAM Dataset and Benchmark
Home
Publications
Research
Datasets and Software
Datasets
Multiview Datasets
Deformable Shape Tracking Datasets
RGB-D SLAM Dataset and Benchmark
Software
Members
Teaching
Workshops
Tutorials
RGB-D SLAM Dataset and Benchmark
Quick Links
Download page
File formats
Camera parameters
Usful tools and scripts
Contact: Jürgen Sturm
We provide a large dataset containing RGB-D data and ground-truth data with the goal to establish a novel benchmark for the evaluation of visual odometry and visual SLAM systems. Our dataset contains the color and depth images of a Microsoft Kinect sensor along the ground-truth trajectory of the sensor. The data was recorded at full frame rate (30 Hz) and sensor resolution (640×480). The ground-truth trajectory was obtained from a high-accuracy motion-capture system with eight high-speed tracking cameras (100 Hz). Further, we provide the accelerometer data from the Kinect. Finally, we propose an evaluation criterion for measuring the quality of the estimated camera trajectory of visual SLAM systems.
How can I use the RGB-D Benchmark to evaluate my SLAM system?
1. Download one or more of the RGB-D benchmark sequences (file formats, useful tools)
2. Run your favorite visual odometry/visual SLAM algorithm (for example, RGB-D SLAM)
3. Save the estimated camera trajectory to a file (file formats, example trajectory)
4. Evaluate your algorithm by comparing the estimated trajectory with the ground truth trajectory. We provide an automated evaluation tool to help you with the evaluation. There is also an online version of the tool.



Computer Vision Group - [

vision.in.tum.de/data/datasets/rgbd-dataset/download

Computer Vision Group

TUM

Technische Universität München

Login

Search

Home › Datasets and Software › Datasets › RGB-D SLAM Dataset and Benchmark › **download**

- Home
- Publications
- Research
- **Datasets and Software**
- **Datasets**
- Multiview Datasets
- Deformable Shape Tracking Datasets
- **RGB-D SLAM Dataset and Benchmark**
- Software
- Members
- Teaching
- Workshops
- Tutorials

Dataset Download

We recommend that you use the **'xyz'** series for your first experiments. The motion is relatively small, and only a small volume on an office desk is covered. Once this works, you might want to try the **'desk'** dataset, which covers four tables and contains several loop closures.

We are happy to share our data with other researchers. Please refer to the respective publication when using this data.

Remarks:

- The file formats are described here.
- The intrinsic camera parameters are here.
- We provide a set of useful tools for working with the dataset.
- The `*_validation` sequences do not contain ground truth. They can only evaluated using the online tool.

Sequence name	Duration	Length	Download	
Category: Testing and Debugging				
freiburg1_xyz	30.09s	7.112m	tgz (0.47GB)	more info
freiburg1_rpy	27.67s	1.664m	tgz (0.42GB)	more info
freiburg2_xyz	122.74s	7.029m	tgz (2.39GB)	more info


Computer Vision Group - [
vision.in.tum.de/data/datasets/rgbd-dataset/download
Computer Vision Group
TUM
Technische Universität München
Login
Search
Home
Publications
Research
Datasets and Software
Datasets
Multiview Datasets
Deformable Shape Tracking Datasets
RGB-D SLAM Dataset and Benchmark
Software
Members
Teaching
Workshops
Tutorials
freiburg1_xyz: RGB movie
Download this movie as avi
Related movies: RGB movie and, depth movie, external camera view
wnload
n is relatively small, and only a
the 'desk' dataset, which covers
tive publication when using this
ed using the online tool.
freiburg1_xyz 30.09s 7.112m tgz (0.47GB) more info
freiburg1_rpy 27.67s 1.664m tgz (0.42GB) more info
freiburg2_xyz 122.74s 7.029m tgz (2.39GB) more info

File Formats

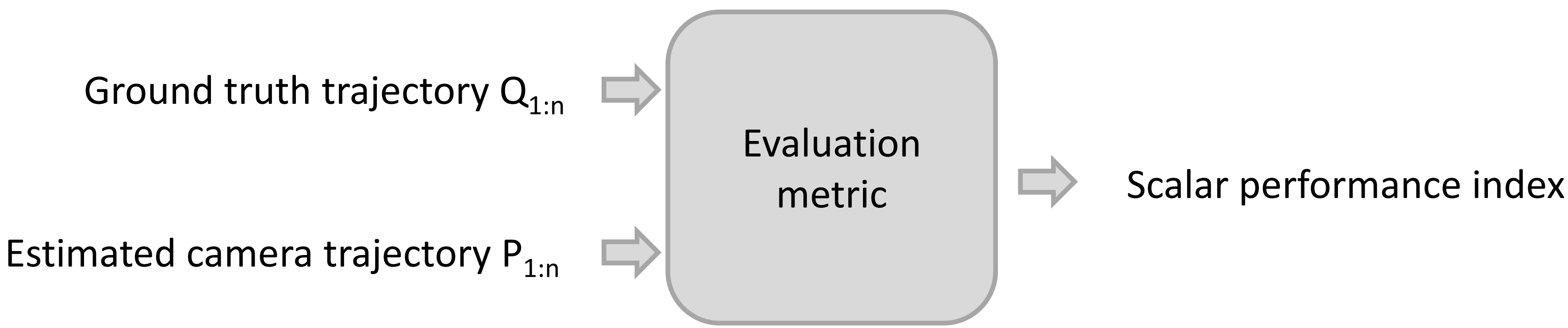
- In total: 69 sequences (33 training, 36 testing)
- One TGZ archive per sequence, containing
  - Color and depth images (PNG)
  - List of color images (timestamp filename)
  - List of depth images (timestamp filename)
  - List of camera poses (timestamp tx ty tz qx qy qz qw)

What Is a Good Evaluation Metric?

- Visual odometry system outputs
  - Camera trajectory (accumulated)
- Visual SLAM system outputs
  - Camera trajectory
  - 3D map
- Ground truth
  - Camera trajectory

What Is a Good Evaluation Metric?

- Trajectory comparison
  - Ground truth trajectory $Q_1, \ldots, Q_n \in \mathrm{SE}(3)$
  - Estimate camera trajectory $P_1, \ldots, P_n \in \mathrm{SE}(3)$
- Two evaluation metrics
  - Drift per second
  - Global consistency

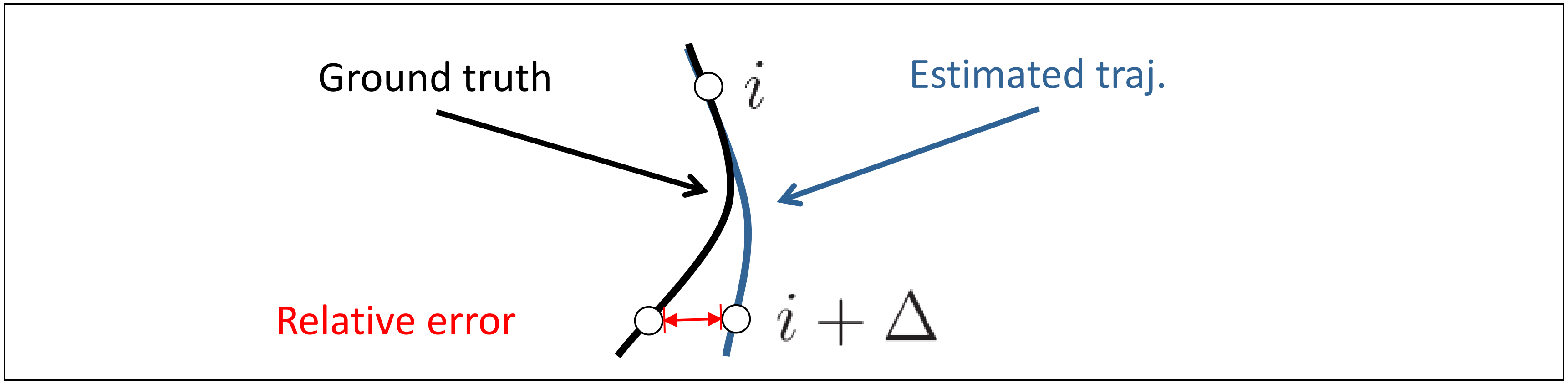


Relative Pose Error (RPE)

- Measures the (relative) drift between the i-th frame and the (i+Δ)-th frame

$$E_i := \left(Q_i^{-1} Q_{i+\Delta}\right)^{-1} \left(P_i^{-1} P_{i+\Delta}\right)$$

Relative error — True motion — Estimated motion



Relative Pose Error (RPE)

How to choose the time delta Δ?

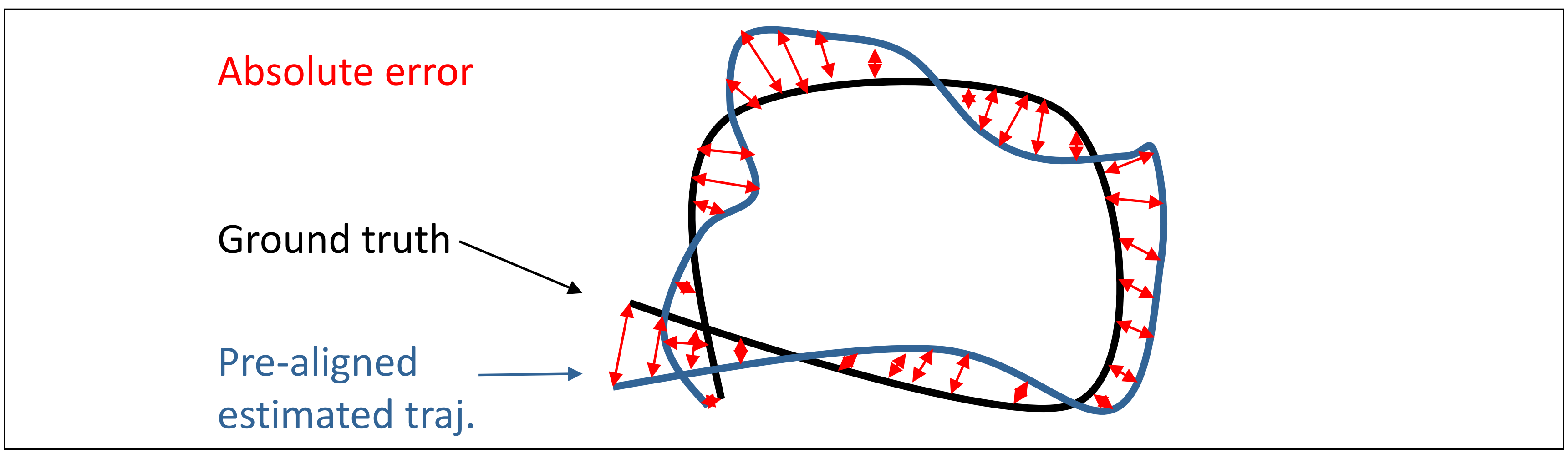
- For odometry methods:
  - Δ=1: Drift per frame
  - Δ=30: Drift per second
- For SLAM methods:
  - Average over all possible deltas
  - Measures the global consistency

Absolute Trajectory Error (ATE)

- Alternative method to evaluate SLAM systems
- Requires pre-aligned trajectories

$$E_i := Q_i^{-1} S P_i$$

Absolute error Groundtruth Alignment Estimated



Evaluation Tools

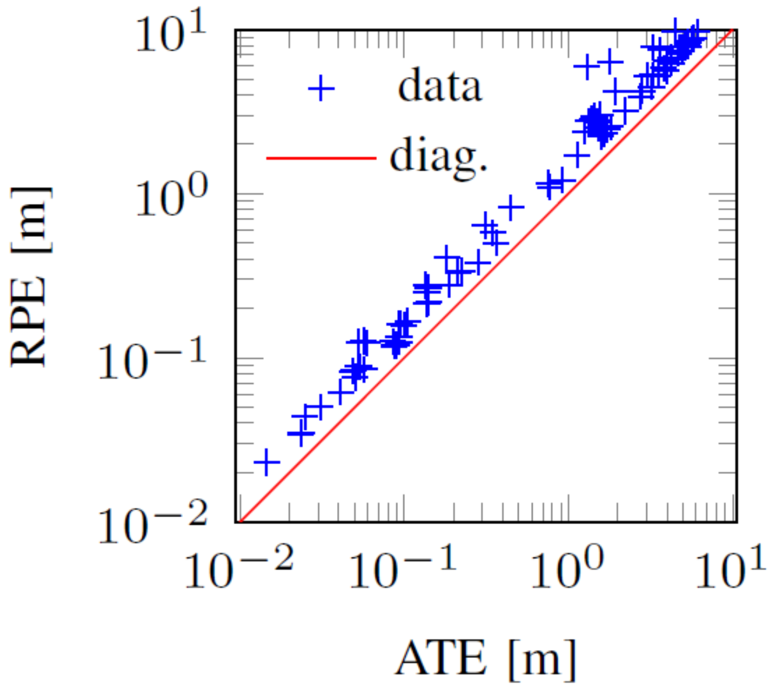
- Average over all time steps

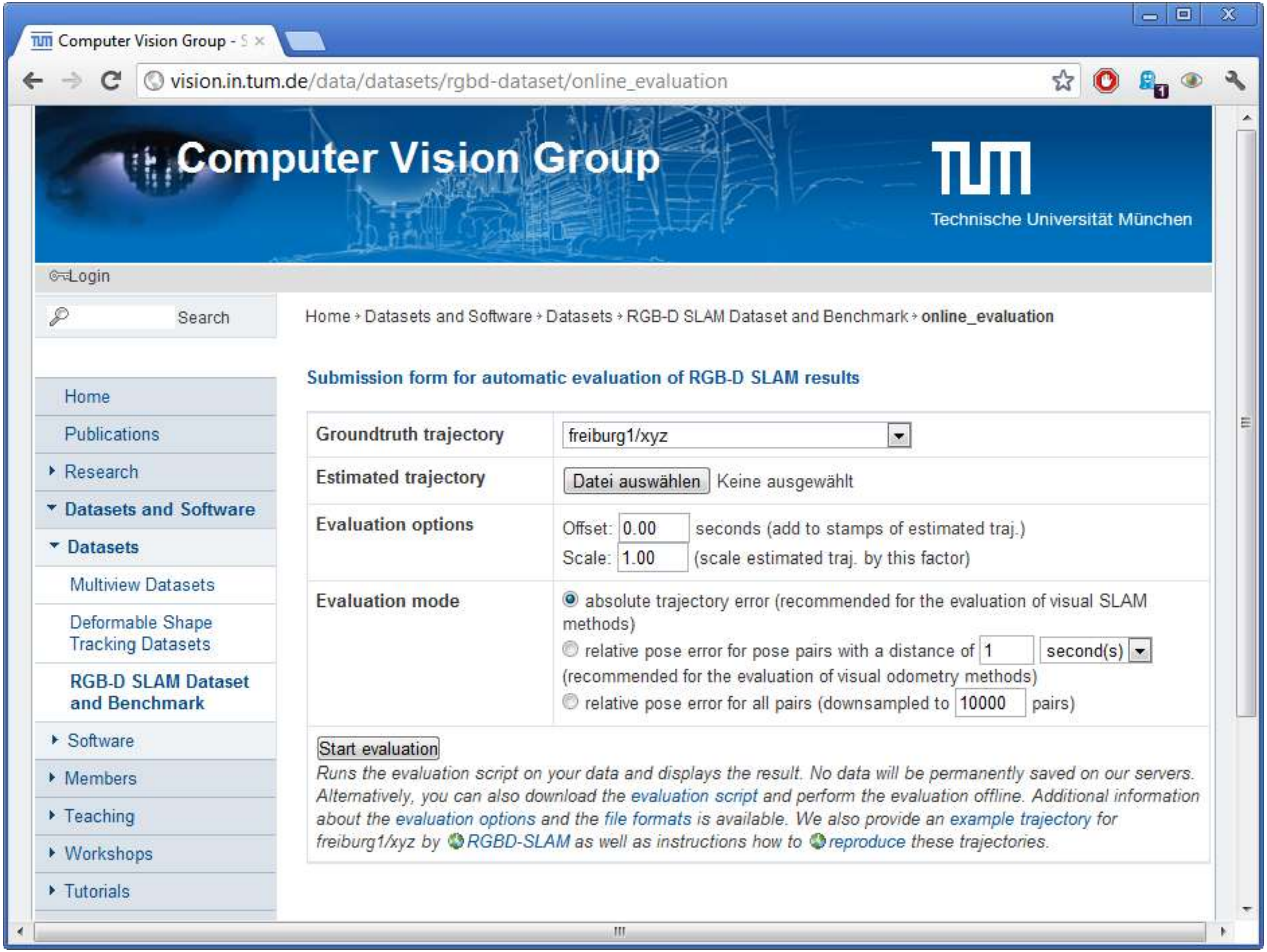
$$\mathrm{RMSE}(E_{1:n}) := \left(\frac{1}{m}\sum_{i=1}^{m} \|trans(E_i)\|^2\right)^{1/2}$$

- Evaluation scripts for both evaluation metrics available (Python)
- Output: RMSE, median, mean
- Plot to png/pdf (optional)

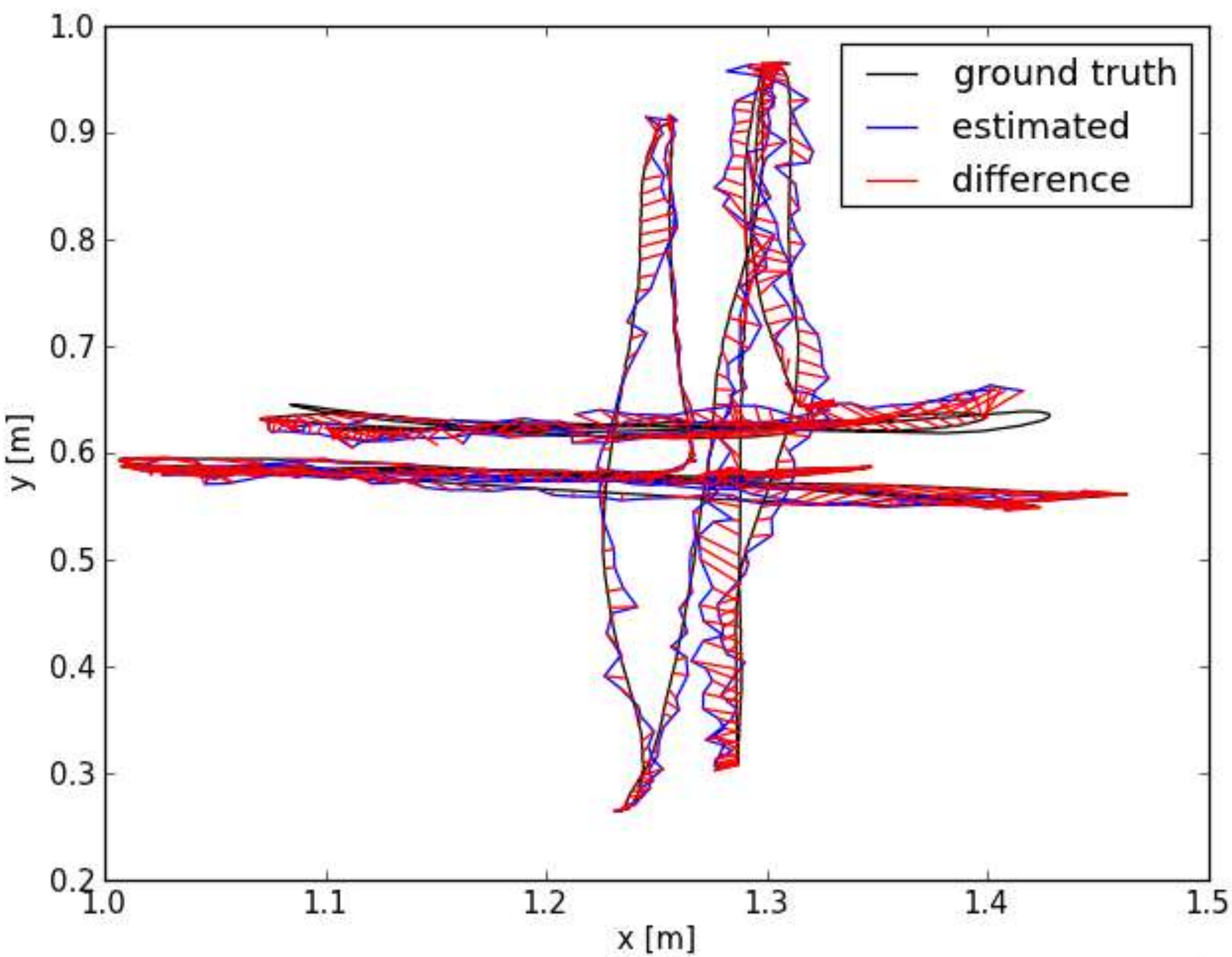
Comparison of RPE and ATE

- RPE and ATE are strongly related
- RPE considers additionally rotational errors
- RPE ≥ ATE




Computer Vision Group - S
vision.in.tum.de/data/datasets/rgbd-dataset/online_evaluation
Computer Vision Group
TUM
Technische Universität München
Login
Search
Home › Datasets and Software › Datasets › RGB-D SLAM Dataset and Benchmark › online_evaluation
Home
Publications
Research
Datasets and Software
Datasets
Multiview Datasets
Deformable Shape Tracking Datasets
RGB-D SLAM Dataset and Benchmark
Software
Members
Teaching
Workshops
Tutorials
Submission form for automatic evaluation of RGB-D SLAM results
Groundtruth trajectory
freiburg1/xyz
Estimated trajectory
Datei auswählen
Keine ausgewählt
Evaluation options
Offset: 0.00 seconds (add to stamps of estimated traj.)
Scale: 1.00 (scale estimated traj. by this factor)
Evaluation mode
absolute trajectory error (recommended for the evaluation of visual SLAM methods)
relative pose error for pose pairs with a distance of 1 second(s) (recommended for the evaluation of visual odometry methods)
relative pose error for all pairs (downsampled to 10000 pairs)
Start evaluation
Runs the evaluation script on your data and displays the result. No data will be permanently saved on our servers. Alternatively, you can also download the evaluation script and perform the evaluation offline. Additional information about the evaluation options and the file formats is available. We also provide an example trajectory for freiburg1/xyz by RGBD-SLAM as well as instructions how to reproduce these trajectories.

```
compared_pose_pairs 786 pairs
absolute_translational_error.rmse 0.013473 m
absolute_translational_error.mean 0.012029 m
absolute_translational_error.median 0.011176 m
absolute_translational_error.std 0.006068 m
absolute_translational_error.min 0.000939 m
absolute_translational_error.max 0.034727 m
```



Summary – TUM RGB-D Benchmark

- Dataset for the evaluation of RGB-D SLAM systems
- Ground-truth camera poses
- Evaluation metrics + tools available
- Since August 2011:
  - >17.000 visitors
  - >4.500 online trajectory evaluations
  - >15 published research papers using the dataset
- Next steps:
  - Possibility to upload own trajectories/publications
  - Results page and automated ranking

Conclusion

- Dense methods bear a large potential
  - Dense camera tracking
  - Dense 3D reconstruction
  - Open question: Estimate both at the same time?
- Benchmarks stimulate the comparison of alternative approaches
- Please contact us if you are interested in a collaboration!