Scanning and Printing Objects in 3D

Dr. Jürgen Sturm
metaio GmbH

(formerly Technical University of Munich)

My Research Areas

- Visual navigation for mobile robots

RoboCup

Kinematic Learning

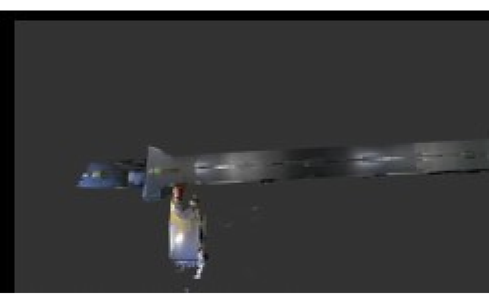
Articulated Objects

Quadrotors

- Camera tracking and 3D reconstruction

RGB-D SLAM

Visual Odometry



Large-Scale Reconstruction

3D Printing

My Research Areas

Camera-based Navigation of a Low-Cost Quadrotor

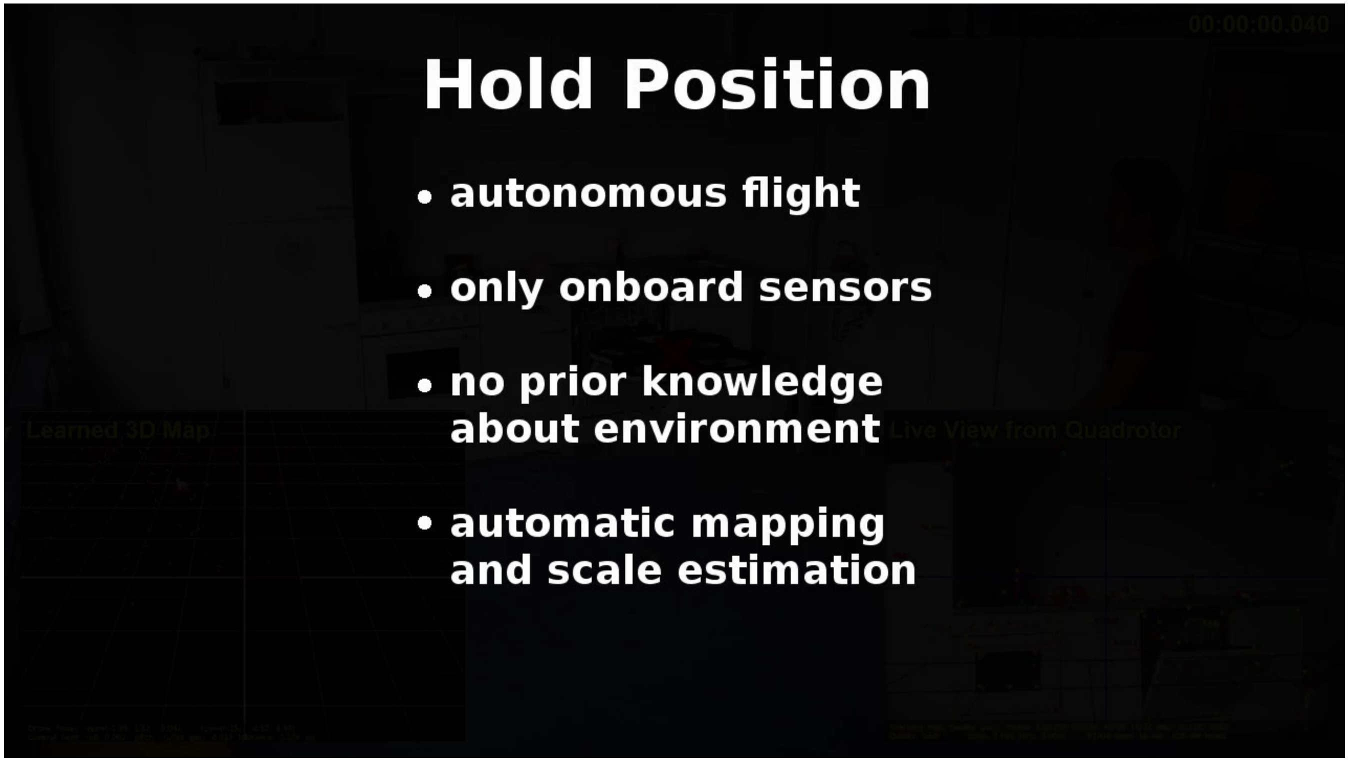
[IROS ’12, RSS ’13, UAV-g ‘13, RAS ’14]

TUM TeachInf Best Lecture Award 2012 and 2013

EdX Course AUTONAVx with 20k participants

My Research Areas

Camera-based Navigation of a Low-Cost Quadrotor



[IROS ’12, RSS ’13, UAV-g ‘13, RAS ’14]
TUM TeachInf Best Lecture Award 2012 and 2013
EdX Course AUTONAVx with 20k participants

My Research Areas

EdX Course „Autonomous Navigation for Flying Robots“

[IROS ’12, RSS ’13, UAV-g ‘13, RAS ’14]

TUM TeachInf Best Lecture Award 2012 and 2013

EdX Course AUTONAVx with 20k participants

My Research Areas

- Visual navigation for mobile robots

RoboCup

Kinematic Learning

Articulated Objects

Quadrotors

- Camera tracking and 3D reconstruction

RGB-D SLAM

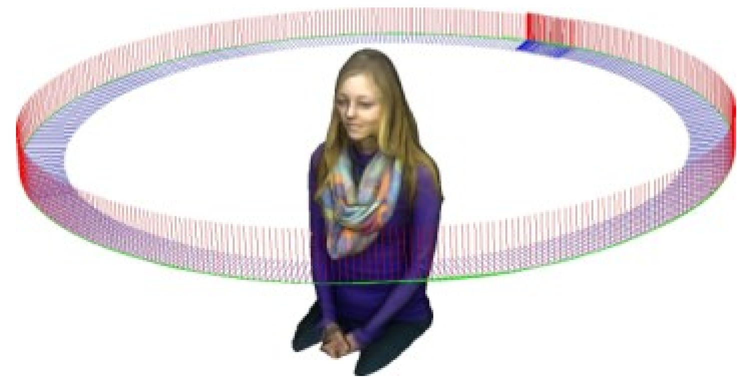
Visual Odometry

Large-Scale Reconstruction

3D Printing

Motivation

Wouldn‘t it be cool to have a 3D photo booth?



Questions:

- How to scan a person in 3D?
- How to prepare the model for 3D printing?

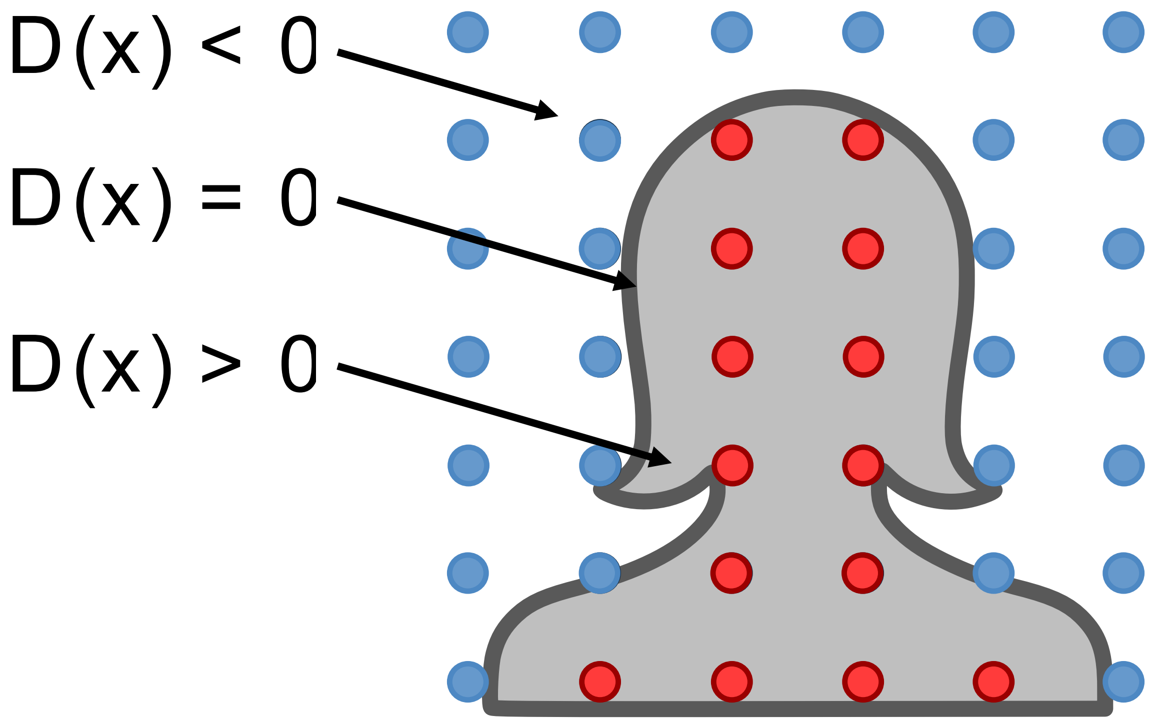
Problem Description

- Setup:
  Static RGB-D camera, person sitting on a swivel chair

- Given: A sequence of color and depth images
- Wanted: Accurate, watertight 3D model

Signed Distance Function (SDF)

[Curless and Levoy, '96]



Negative signed distance (=outside)

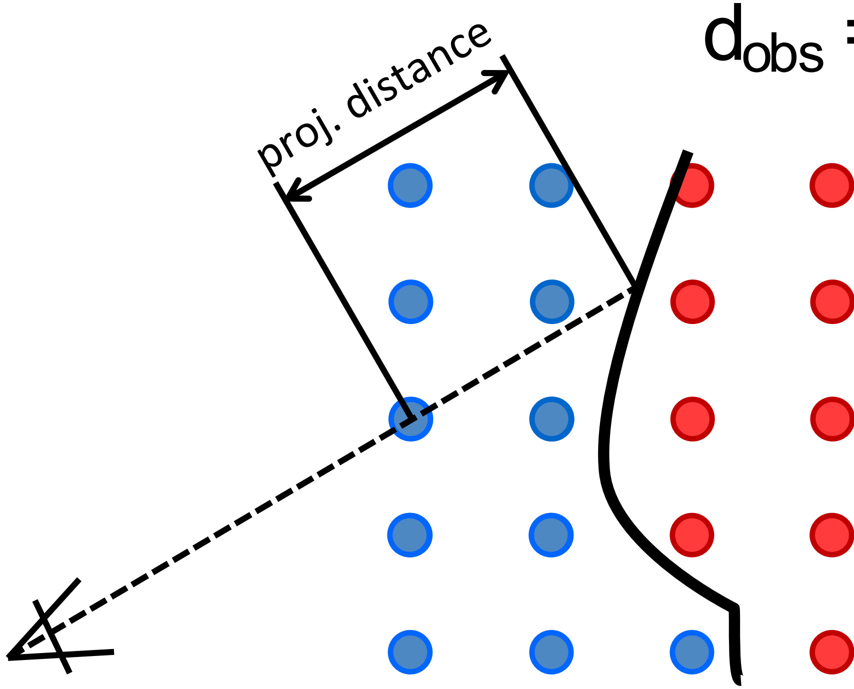
Positive signed distance (=inside)

Signed Distance Function (SDF)

[Curless and Levoy, '96]

- Compute SDF from a depth image
- Measure distance of each voxel to the observed surface
- Can be done in parallel for all voxels (→ GPU)

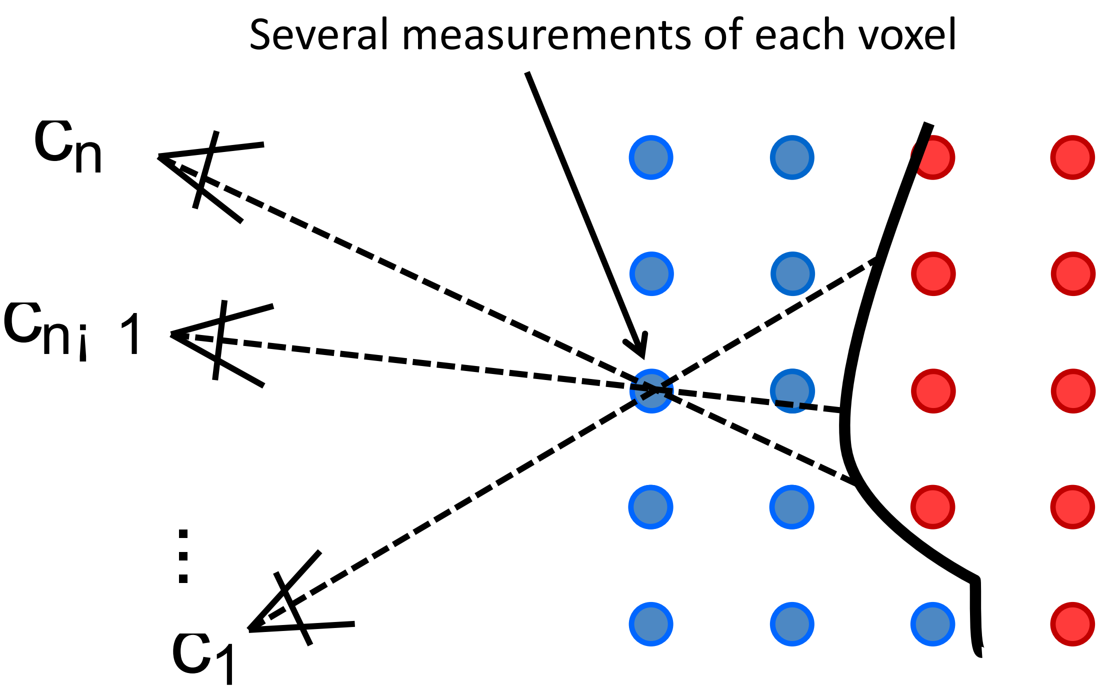
$$d_{obs} = z - I_Z(\pi(x; y; z))$$



Signed Distance Function (SDF)

[Curless and Levoy, '96]

- Calculate weighted average over all measurements
- Assume known camera poses (for now)



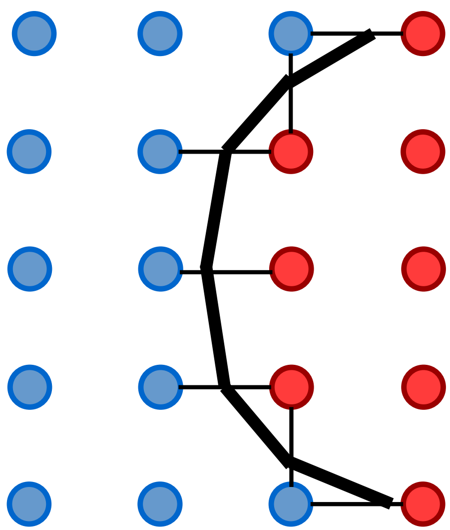
$$D \; \tilde{A} \; \frac{WD + wd}{W + w}$$

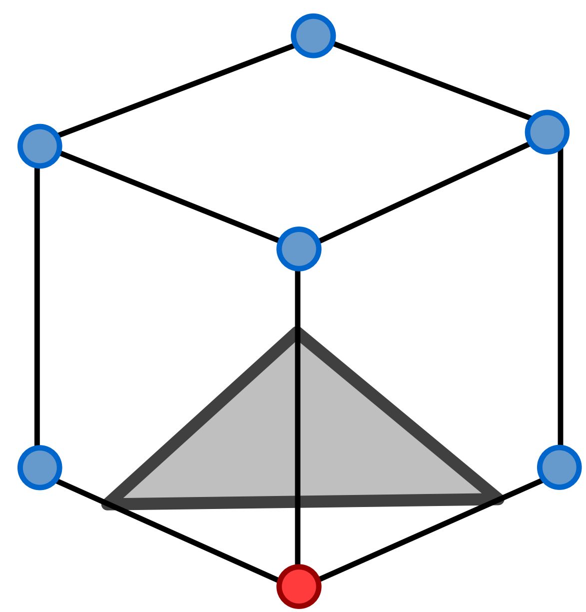
$$C \; \tilde{A} \; \frac{WC + wc}{W + w}$$

$$W \; \tilde{A} \; W + w$$

Mesh Extraction using Marching Cubes

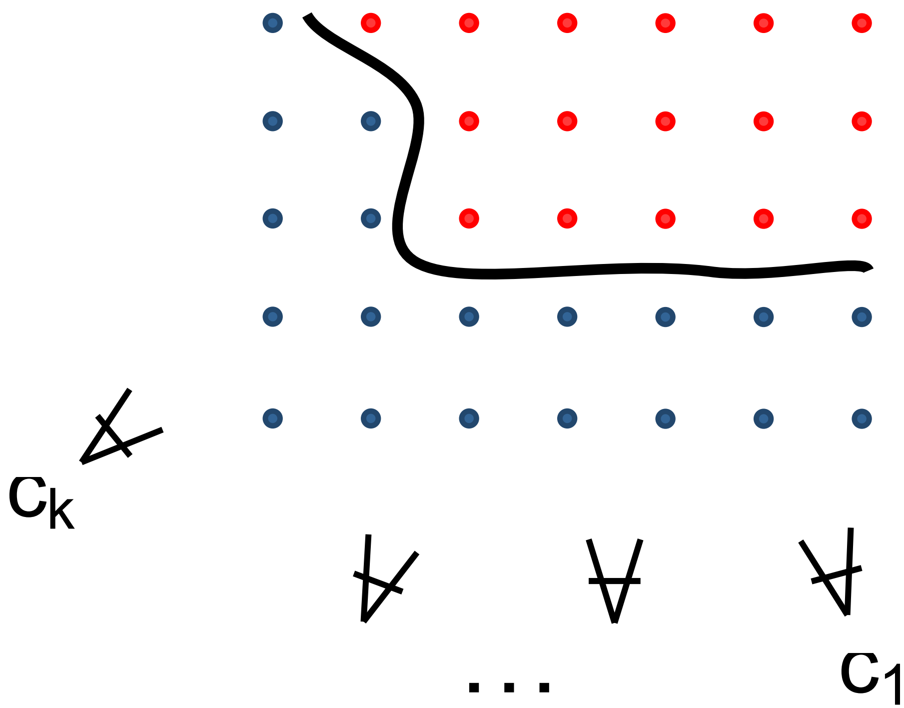
- Find zero-crossings in the signed distance function by interpolation





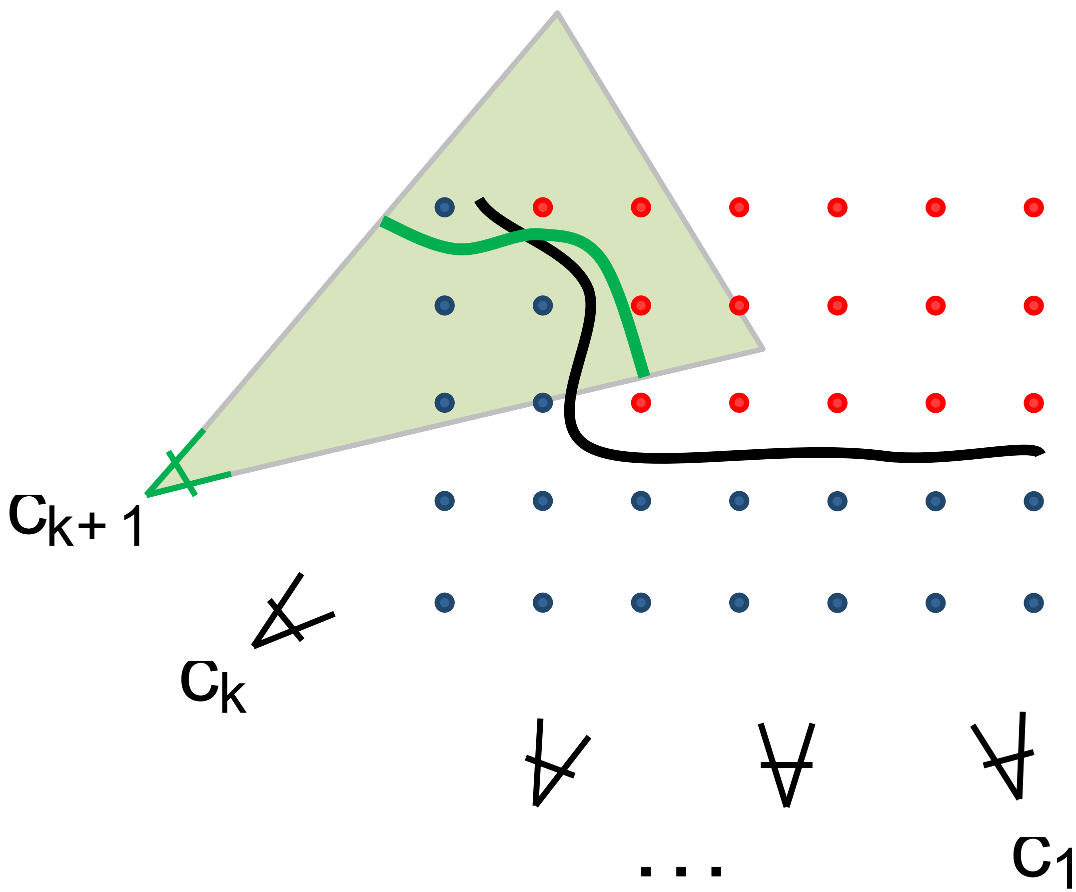
Estimating the Camera Pose

- SDF built from the first k frames



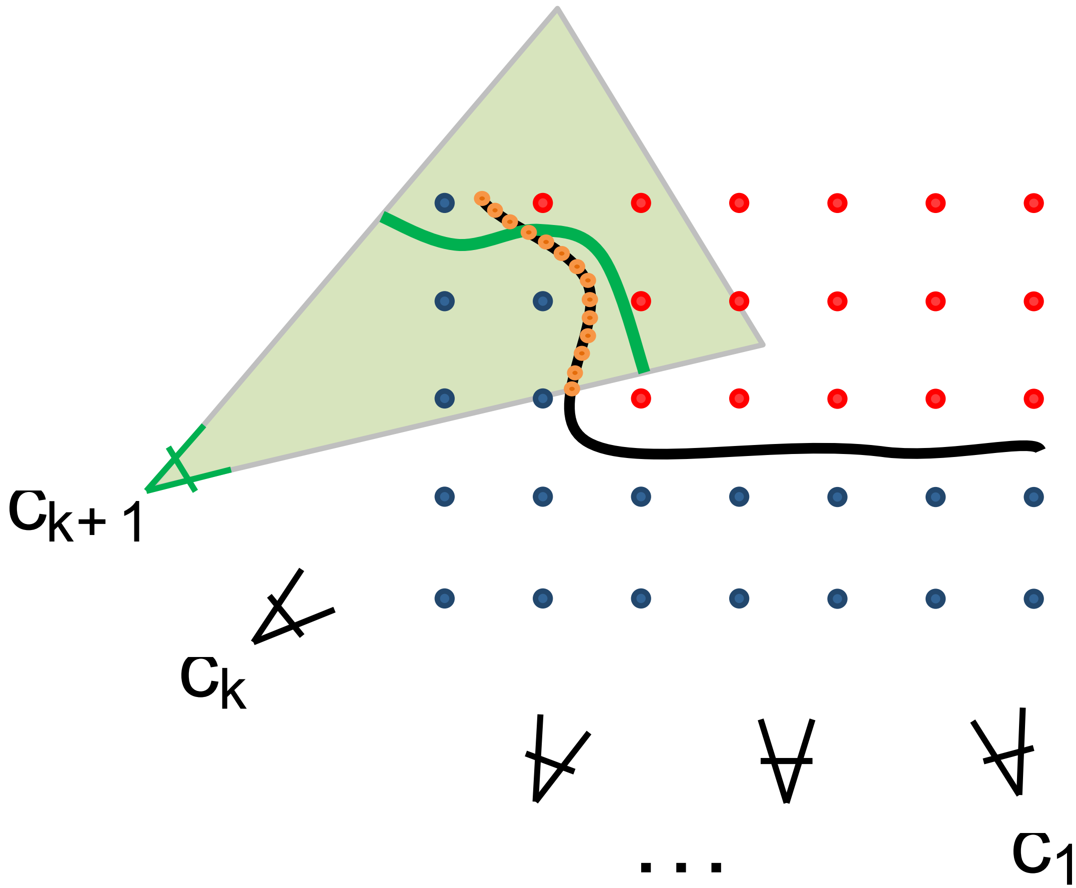
Estimating the Camera Pose

- We seek the next camera pose (k+1)



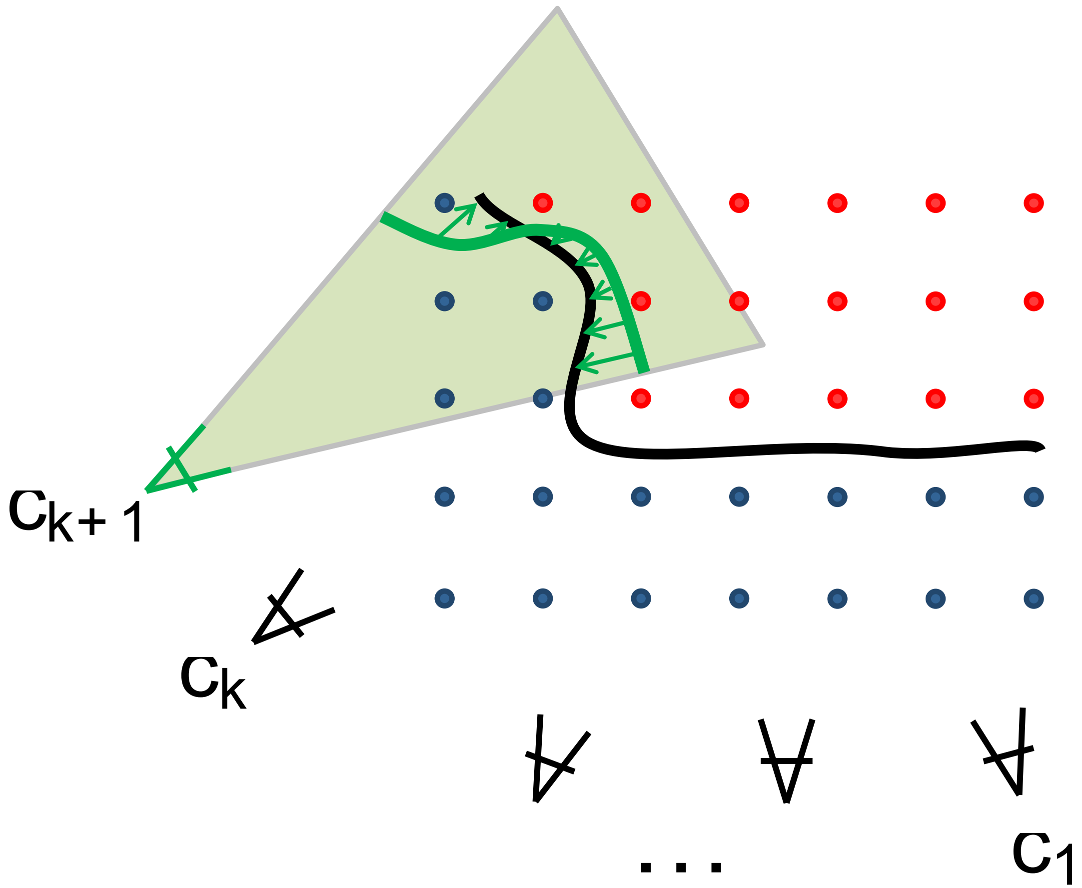
Estimating the Camera Pose

- KinectFusion generates a synthetic depth image from SDF and aligns it using ICP



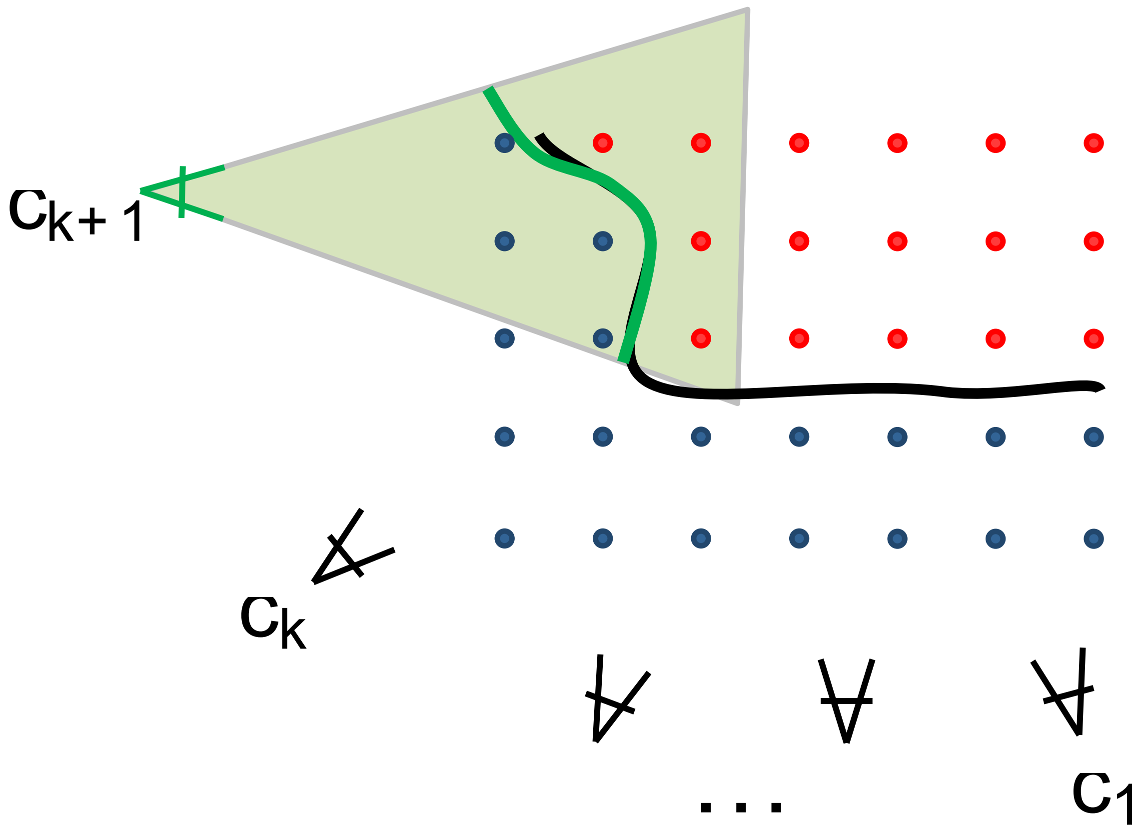
Estimating the Camera Pose

- Our approach: Use SDF directly during minimization



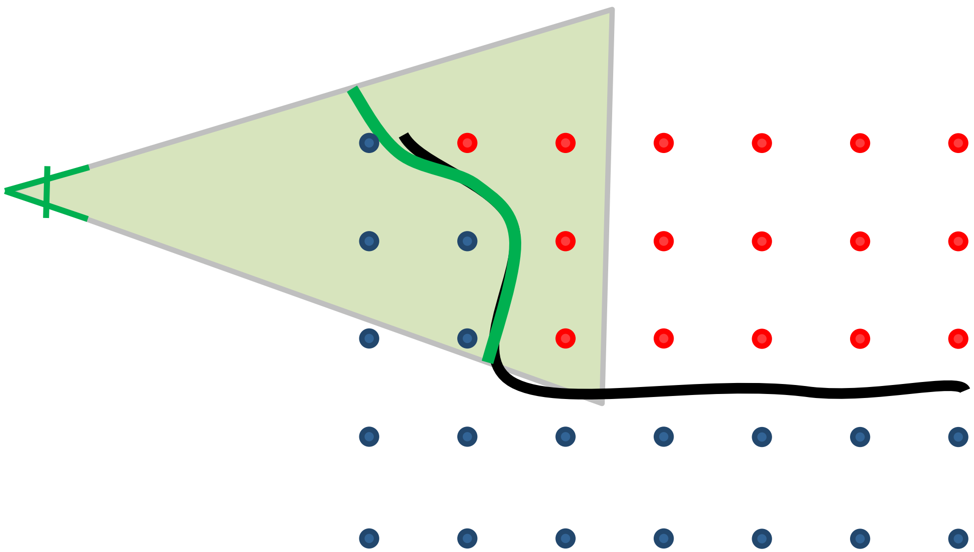
Estimating the Camera Pose

- Our approach: Use SDF directly during minimization



Estimating the Camera Pose

- Our approach: Use SDF directly during minimization



$$\arg\min_{R,\mathbf{t}} \sum_{ij} D(R\mathbf{x}_{ij} + \mathbf{t})^2$$

Evaluation on Benchmark

[Bylow, Sturm, Kerl, Kahl, Cremers; RSS 2013]

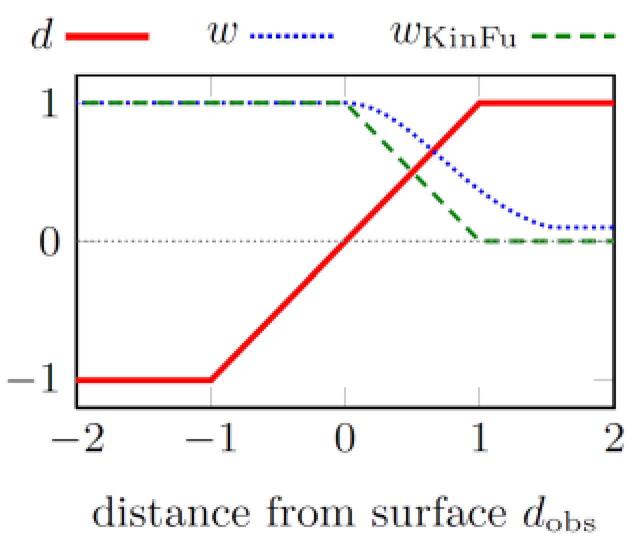
- Thorough evaluation on TUM RGB-D benchmark
- Comparison with KinFu and RGB-D SLAM
- Significantly more accurate and robust than IC

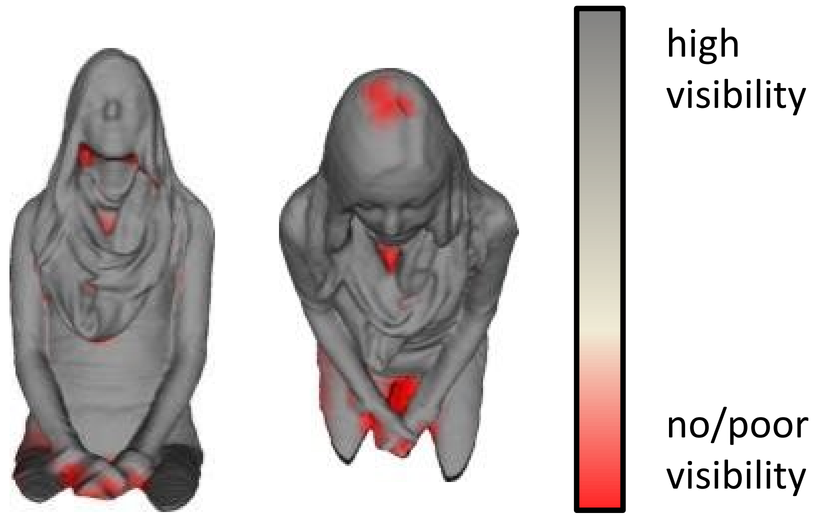
Algorithm	Resolution	Teddy (RMSE)	Desk (RMSE)	Plant (RMSE)
KinFu	256	0.156 m	0.057m	0.598 m
KinFu	512	0.337 m	0.068 m	0.281 m
Our	256	0.086 m	0.038 m	0.047 m
Our	512	0.080 m	0.035 m	0.043 m

Automatically Close Holes

[Sturm, Bylow, Kahl, Cremers; GCPR 2013]

- Certain voxels are never observed in near range
- Regions with no data result in holes
- Idea: Truncate weights to positive values

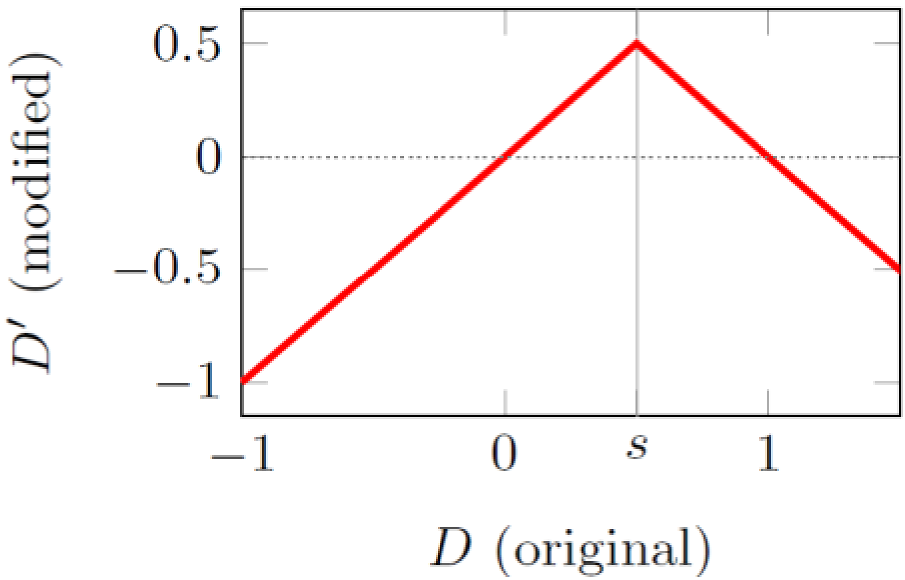




Hollowing Out

[Sturm, Bylow, Kahl, Cremers; GCPR 2013]

- Printing cost is mostly dominated by volume
- Idea: Make the model hollow



before

after

Video (real-time)
[Sturm, Bylow, Kahl, Cremers; GCPR 2013]

Examples of Printed Figures

[Sturm, Bylow, Kahl, Cremers; GCPR 2013]

More Examples

[Sturm, Bylow, Kahl, Cremers; GCPR 2013]

- Need a present?
- Live Demo after the talk

FabliTec 3D Scanner

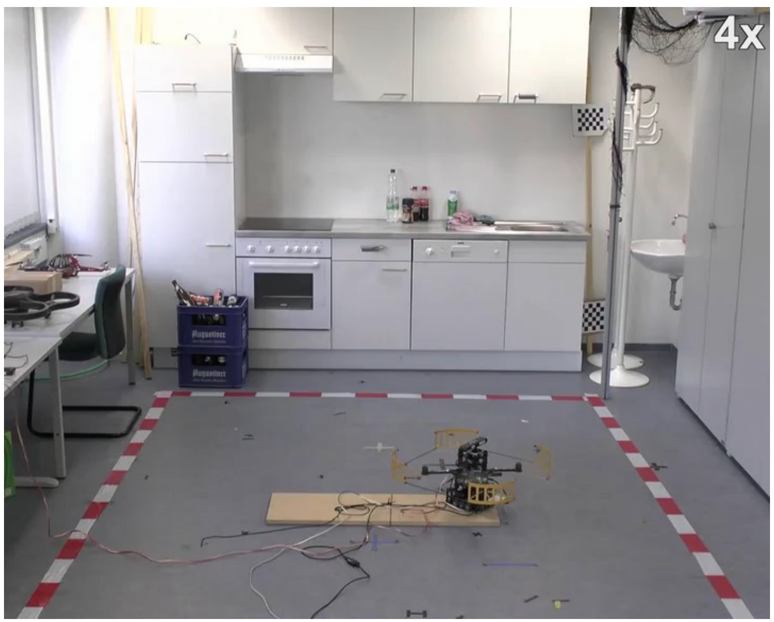
- 3D scanning software “FabliTec 3D Scanner”
- TUM spin-off, founded in 2013
- Targeting private users
- Sale and user support
- Prerequisites
  - Windows 7/8
  - Graphics card from Nvidia
  - Xbox Kinect camera
- Partners
  - German RepRap GmbH
  - Conrad Electronic
  - Volumental (formerly Kinect-at-home)
- Download free demo version from

http://www.fablitec.com

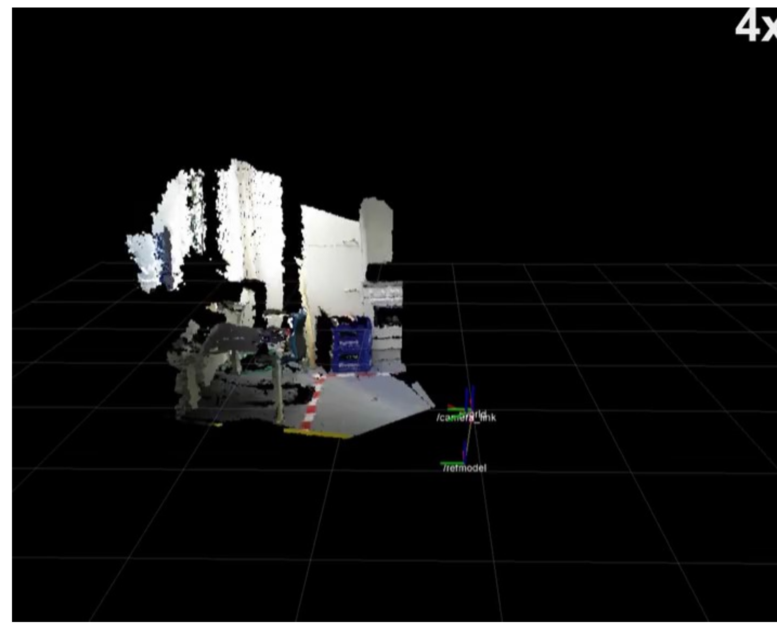
3D Reconstruction from a Quadrocopter

[Bylow et al., RSS 2013; Sturm et al., UAV-g 2013]

- AscTec Pelican quadrocopter
- Real-time 3D reconstruction, position tracking and control (external processing on GPU)



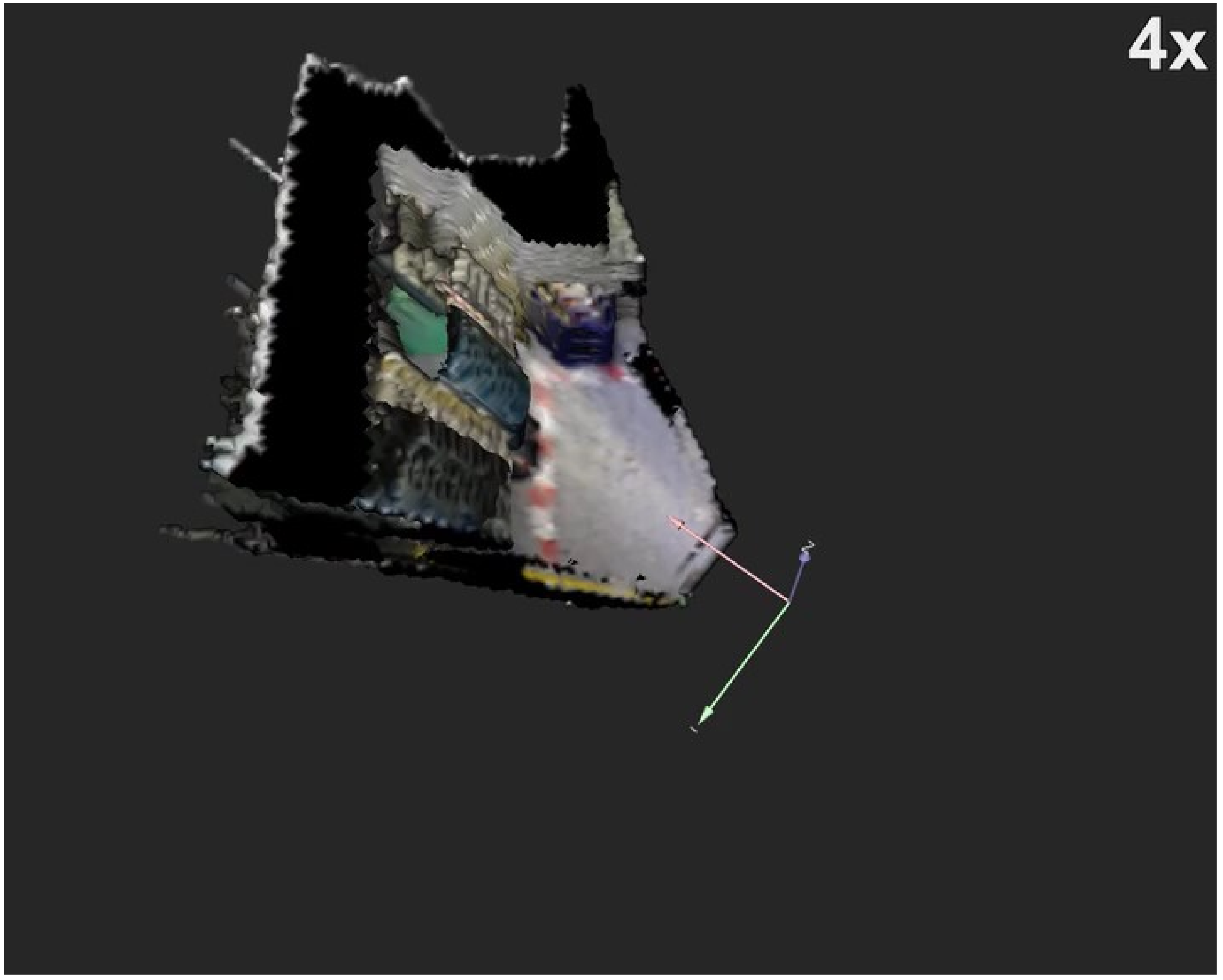
external view



estimated pose

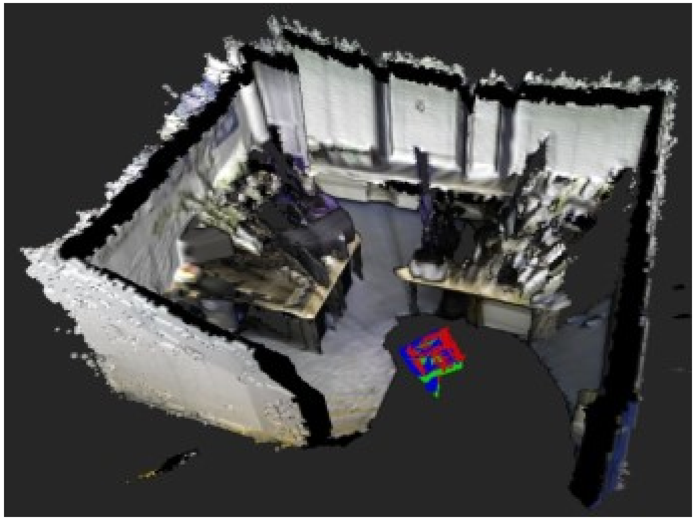
Resulting 3D Model

[Bylow et al., RSS 2013; Sturm et al., UAV-g 2013]



More Examples

[Sturm, Bylow, Kerl, Kahl, Cremers; UAV-g 2013]

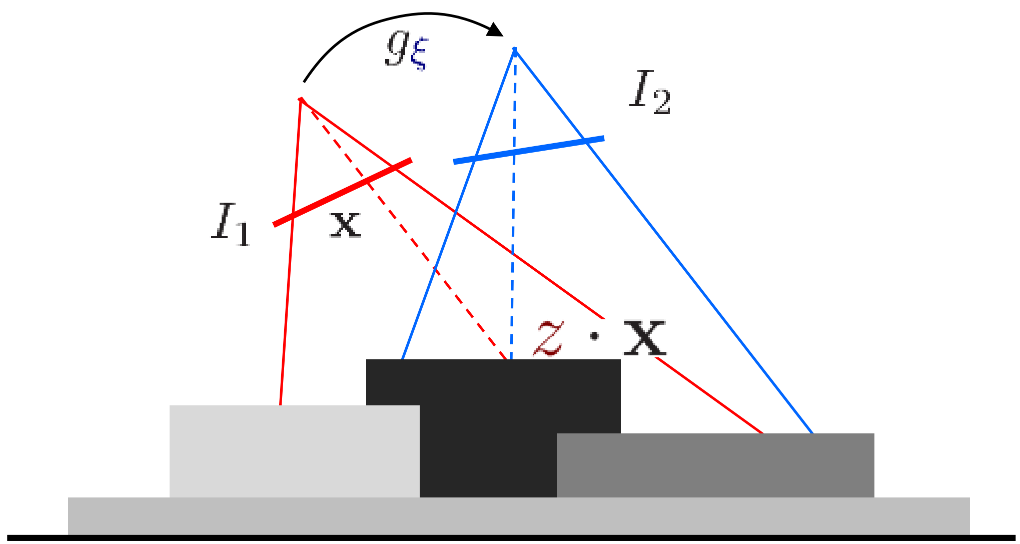


- Nice 3D models, but:
  - Large memory and computational requirements are suboptimal for use on quadrocopter
  - Significant drift in larger environments
- How can we improve on this?

Dense Visual Odometry

[Steinbrücker, Sturm, Cremers, ICCV LDRMC 2011; Kerl, Sturm, Cremers, ICRA 2013]

- Can we compute the camera motion directly?
- Idea



- Photo-consistency constraint $$I_1(\mathbf{x}) = I_2\left(\pi(g_\xi(z \cdot \mathbf{x})\right)$$
- Geometry-consistency constraint $$Z_2(\mathbf{x}') = \mathbf{p}'_z$$

How to deal with noise?

[Steinbrücker, Sturm, Cremers; ICCV LDRMC 2011]

- 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

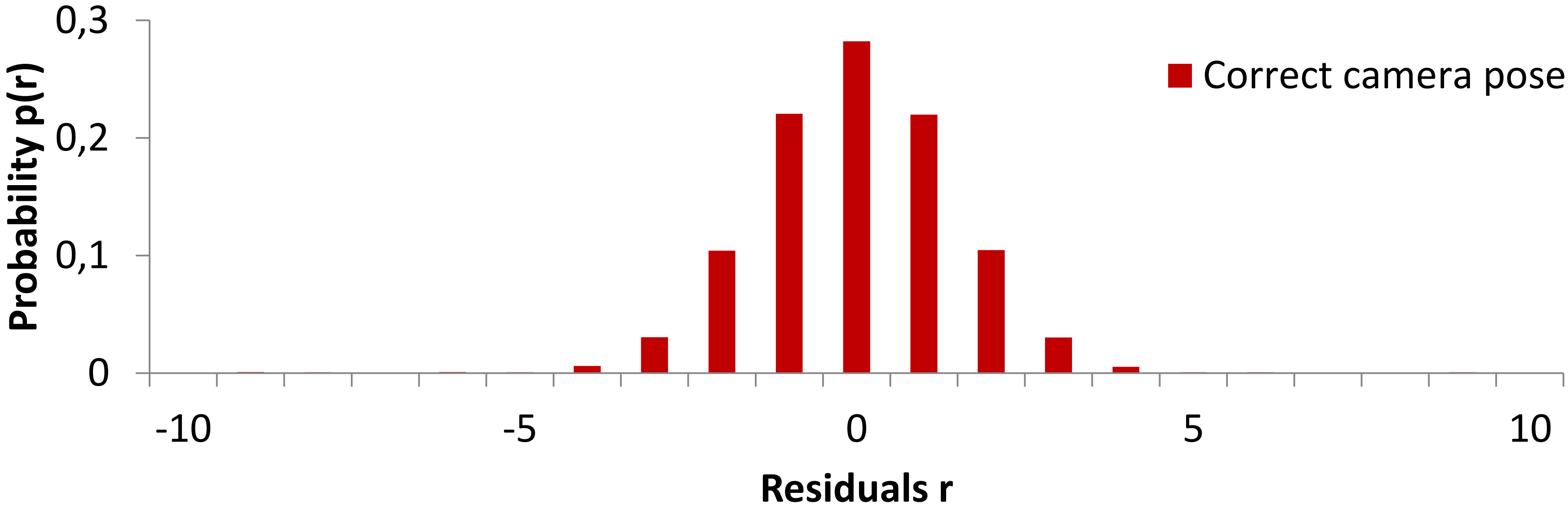
$$r = I_1(\mathbf{x}) - I_2\left(\pi(g_\xi(z \cdot \mathbf{x}))\right) \qquad \left(\mathbf{r} = \begin{pmatrix} I_2(\mathbf{x}') - I_1(\mathbf{x}) \\ Z_2(\mathbf{x}') - \mathbf{p}'_z \end{pmatrix}\right)$$

- How does the residual distribution $p(r)$ look like?

How to deal with noise?

[Steinbrücker, Sturm, Cremers; ICCV LDRMC 2011]

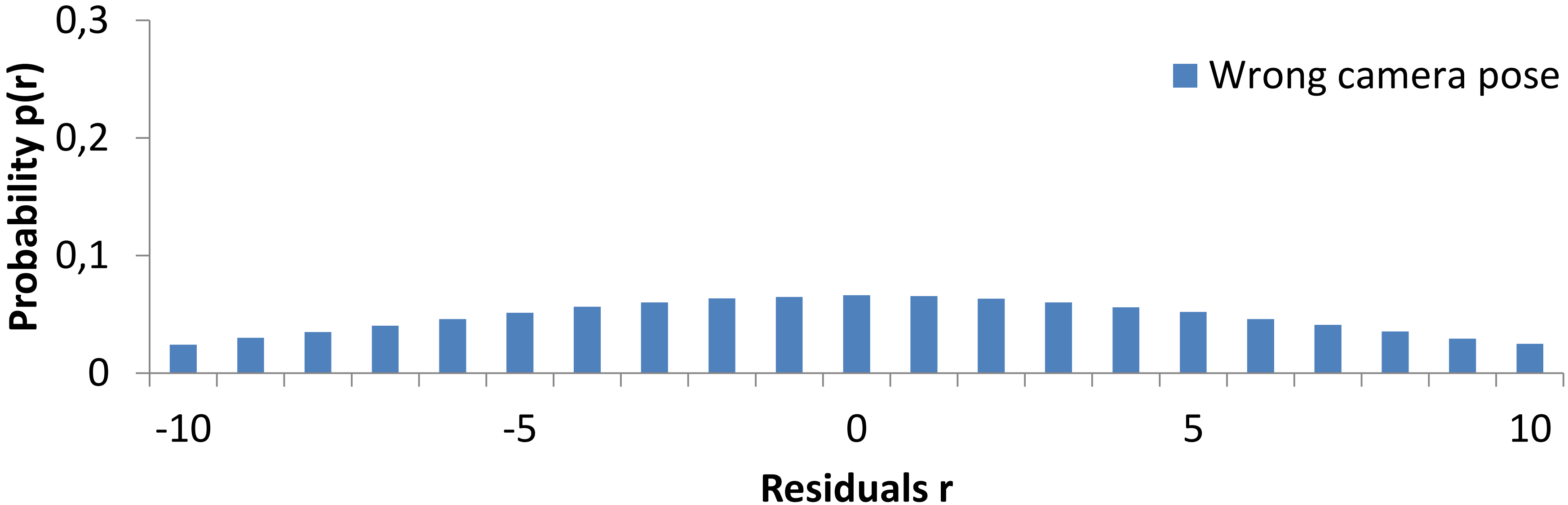
- Zero-mean, peaked distribution
- Example: Correct camera pose



How to deal with noise?

[Steinbrücker, Sturm, Cremers; ICCV LDRMC 2011]

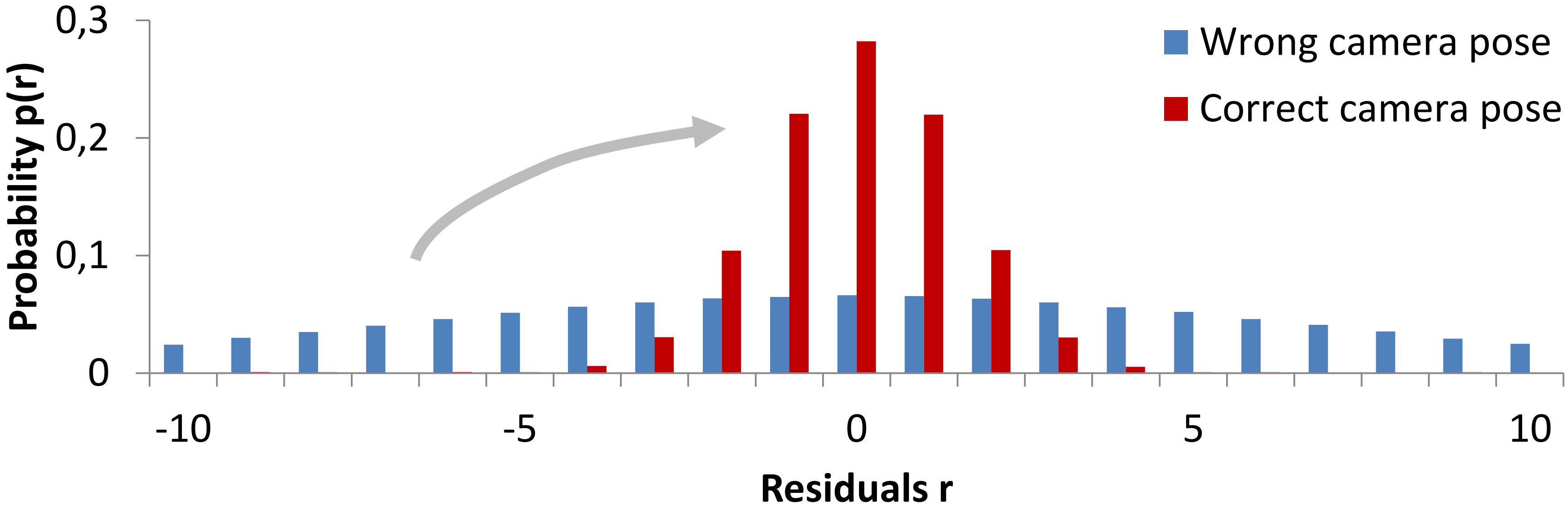
- Zero-mean, peaked distribution
- Example: Wrong camera pose



Residual Distribution

[Steinbrücker, Sturm, Cremers; ICCV LDRMC 2011]

- Our goal: Find the camera pose that maximizes the observation likelihood



Dense Alignment
[Steinbrücker, Sturm, Cremers; ICCV LDRMC 2011]

- Our goal: Find the camera pose that maximizes the observation likelihood

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

compute over all pixels

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

Dense Alignment

[Steinbrücker, Sturm, Cremers; ICCV LDRMC 2011]

- Take negative logarithm

$$\boldsymbol{\xi}^* = \arg\min_{\boldsymbol{\xi}} \sum_i -\log p(r_i(\boldsymbol{\xi}))$$

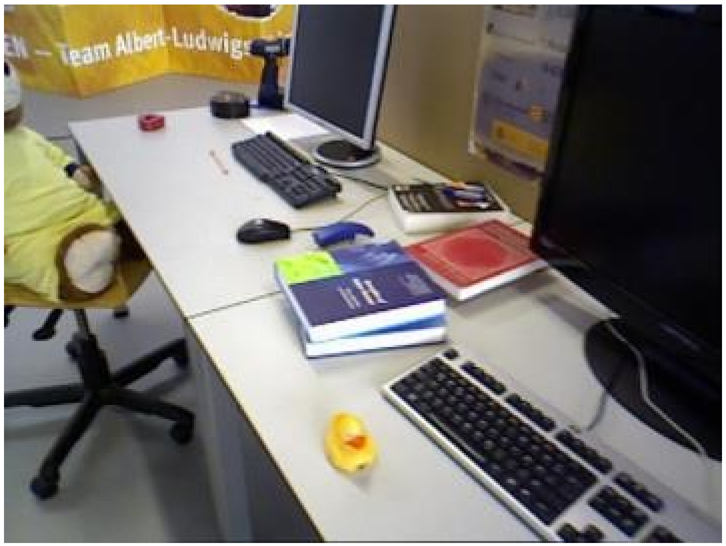
- Set derivative to zero

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

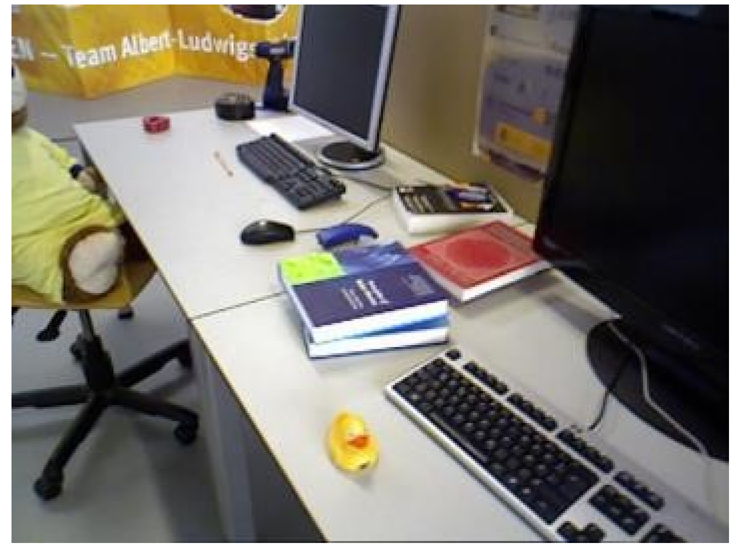
- $r_i(\boldsymbol{\xi})$ is non-linear in $\boldsymbol{\xi}$
- Solve using Gauss-Newton method (linearize, solve, repeat)

Example

[Kerl, Sturm, Cremers; ICRA 2013]



I_1



I_2

Example

[Kerl, Sturm, Cremers; ICRA 2013]

Residuals before registration

$(I_2(\mathbf{x}') - I_1(x))^2 \quad \boldsymbol{\xi} = \mathbf{0}$

Residuals after registration

$(I_2(\mathbf{x}') - I_1(x))^2 \quad \boldsymbol{\xi} = \boldsymbol{\xi}^*$

Coarse-to-Fine

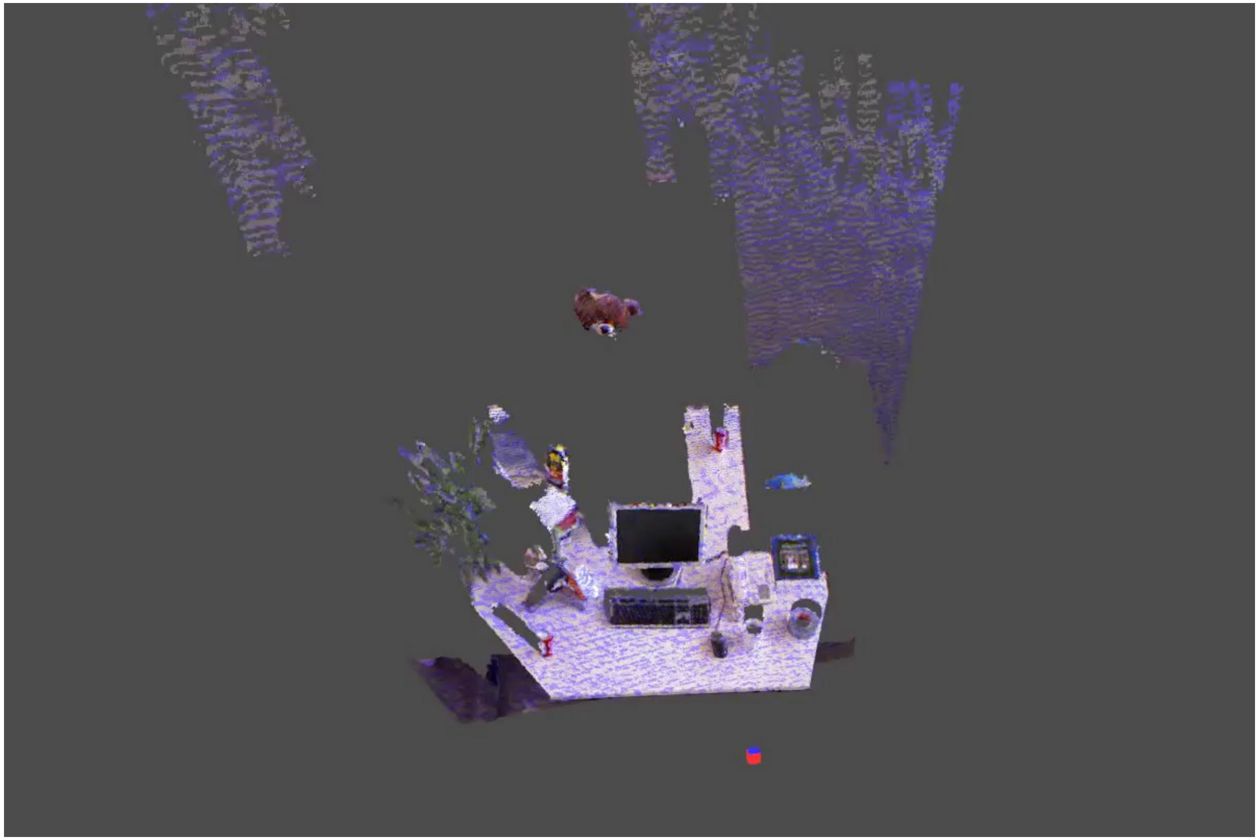
[Steinbrücker, Sturm, Cremers; ICCV LDRMC 2011]

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

Dense Visual Odometry: Results

[Steinbrücker, Sturm, Cremers, ICCV LDRMC 2011; Kerl, Sturm, Cremers, ICRA 2013]

- Runs in real-time on single CPU core (SSE optimized)
- Available as open-source
- Average drift: ~3cm/s

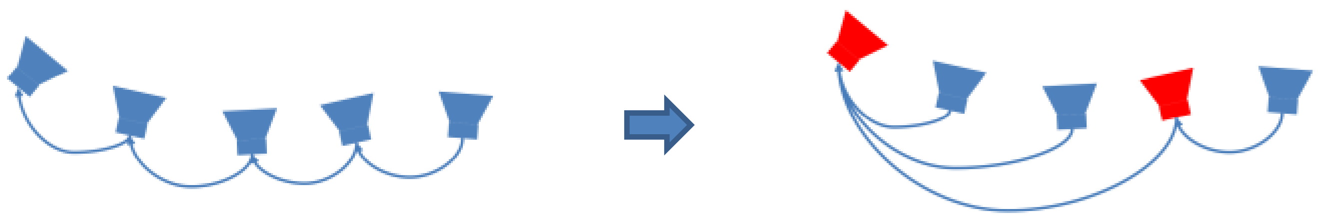


Dense Visual Odometry: Results

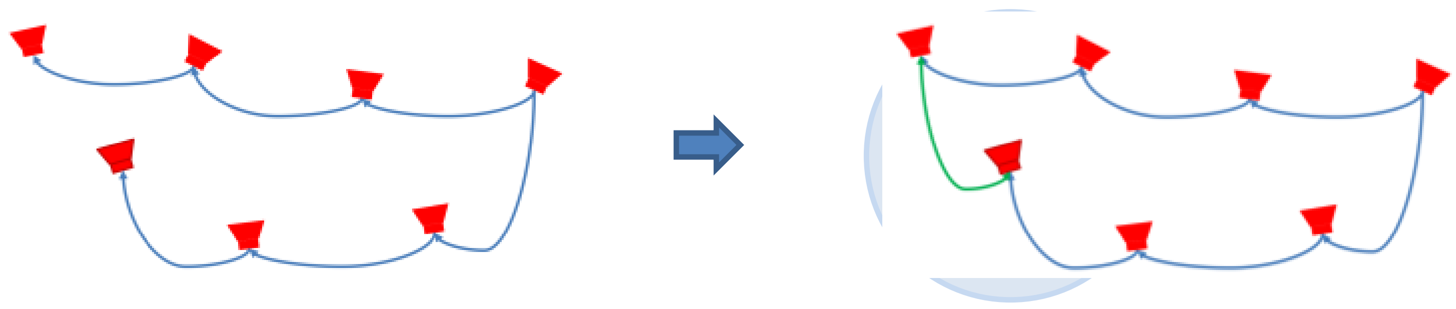
[Kerl, Sturm, Cremers; IROS 2013]

- Problem: Considerable drift accumulation (1.8m/min)
- How can we further reduce this drift?

- Local drift: Track w.r.t. key frames



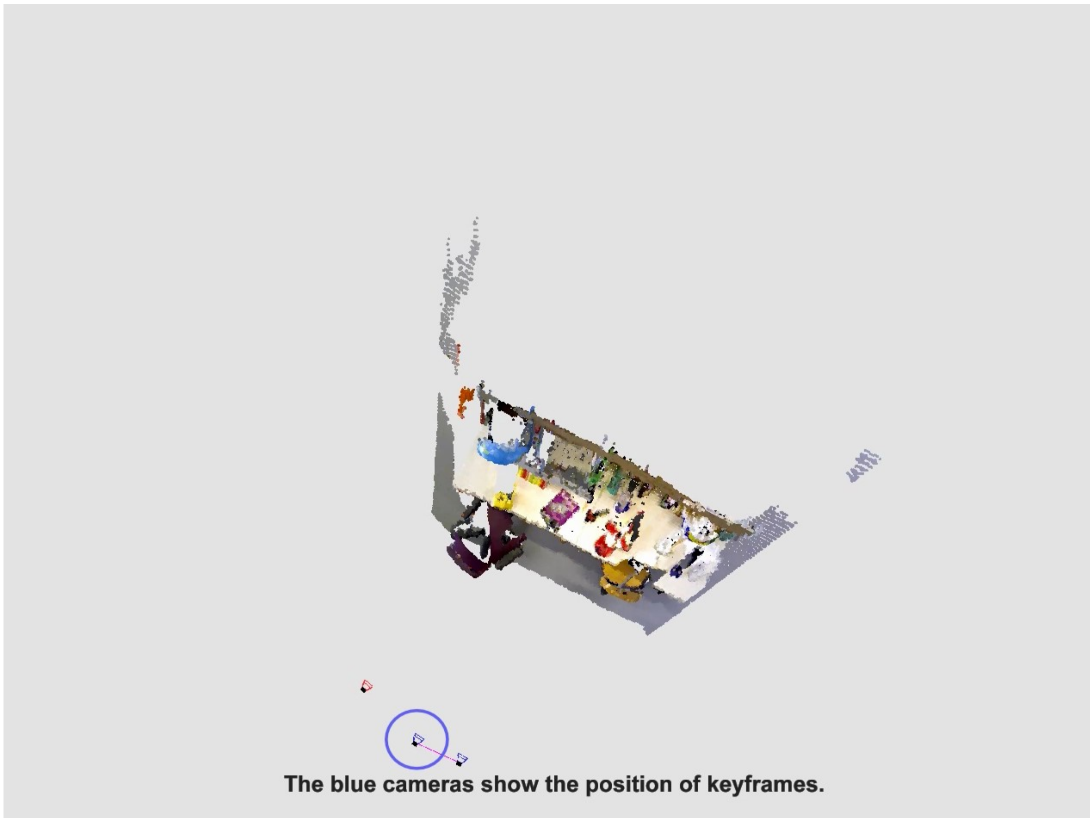
- Global drift: Detect loop closures and optimize pose graph



Dense Visual Odometry: Results

[Kerl, Sturm, Cremers; IROS 2013]

- Keyframes are added dynamically (based on entropy evaluation)
- Localize w.r.t to current keyframe (first CPU core/thread)
- Detect loop closures and optimize pose graph (second CPU core/thread)

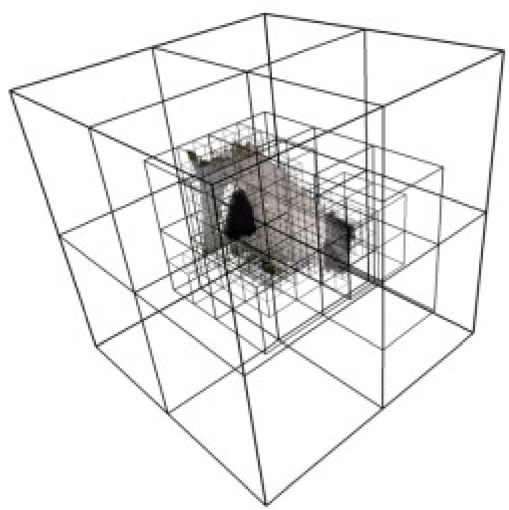


Large-Scale 3D Reconstruction
[Steinbrücker, Kerl, Sturm, Cremers; ICCV 2013]

- We have: Optimized pose graph
- We want: High-resolution 3D map

- Problem: High-resolution voxel grids consume much memory (grows cubically)
  - 512^3 voxels, 24 byte per voxel → 3.2 GB
  - 1024^3 voxels, 24 byte per voxel → 24 GB
  - …

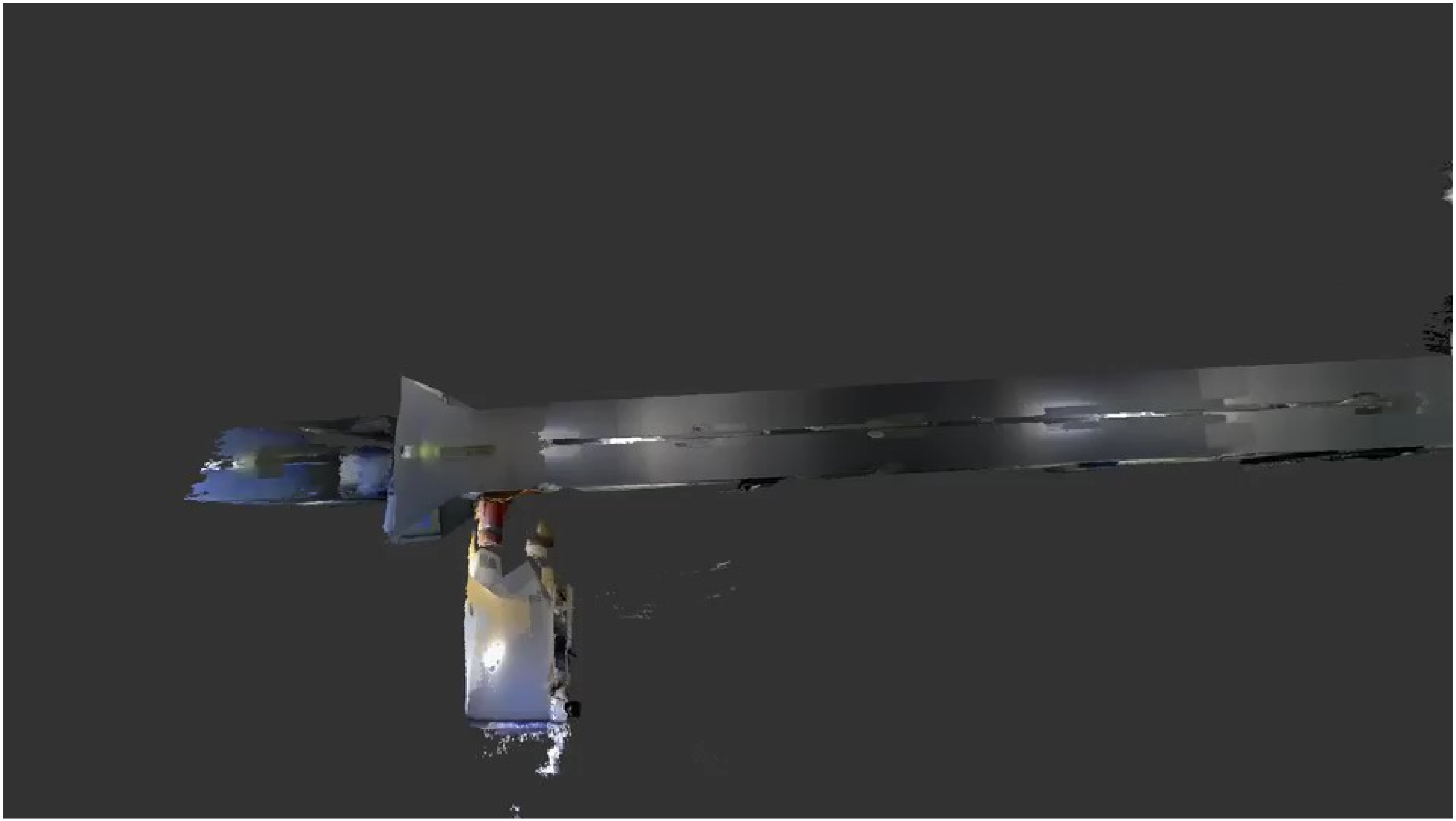
- Idea:
  - Save data in oct-tree data structure
  - Only allocate cells that are close to the surf
  - Store geometry at multiple resolutions
  - Tree can grow dynamically (no fixed size)



Large-Scale 3D Reconstruction

[Steinbrücker, Kerl, Sturm, Cremers; ICCV 2013]

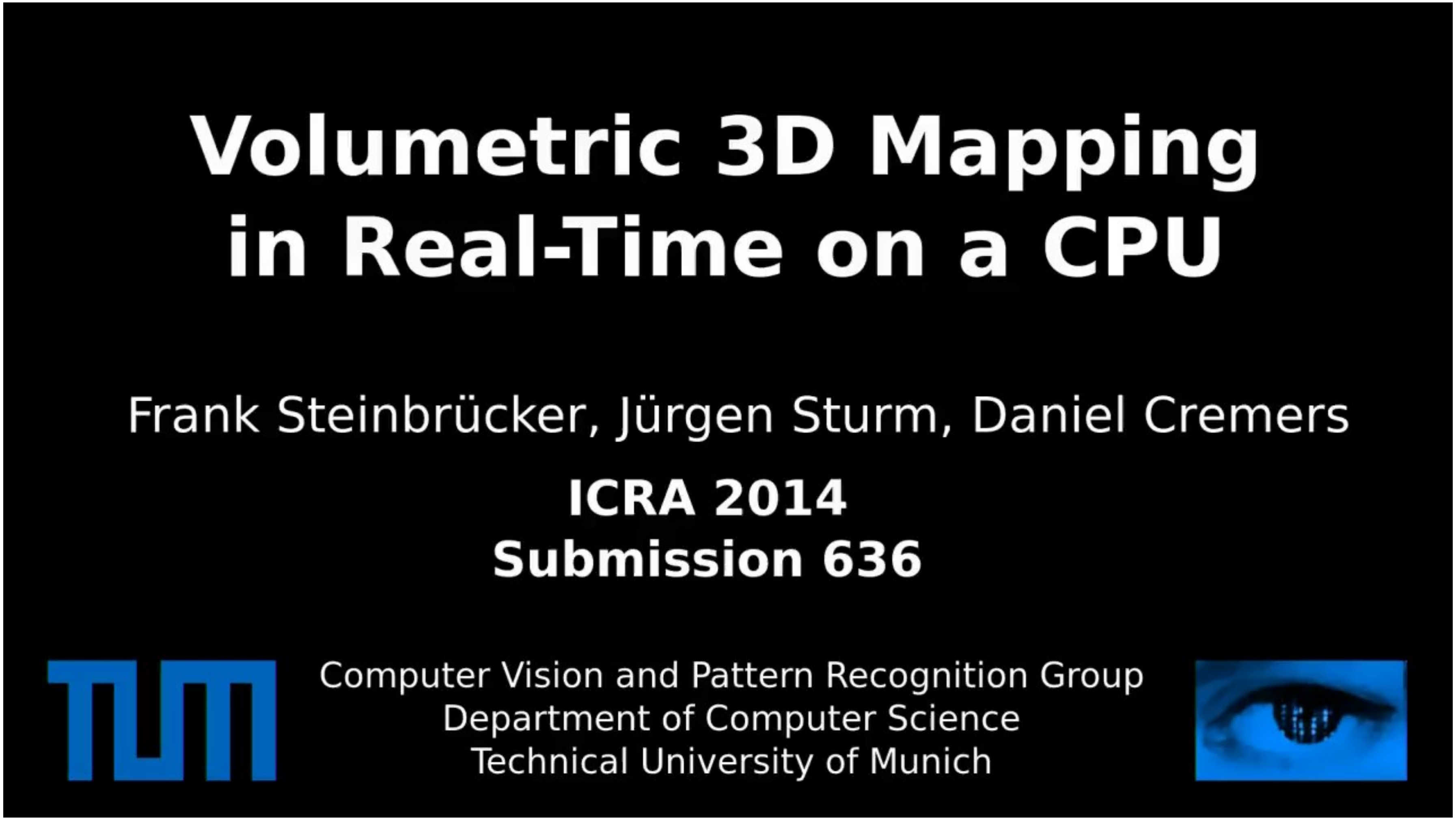
- Runs at 200 fps on a GPU (assuming camera poses are known)



3D Mapping in Real-Time on a CPU

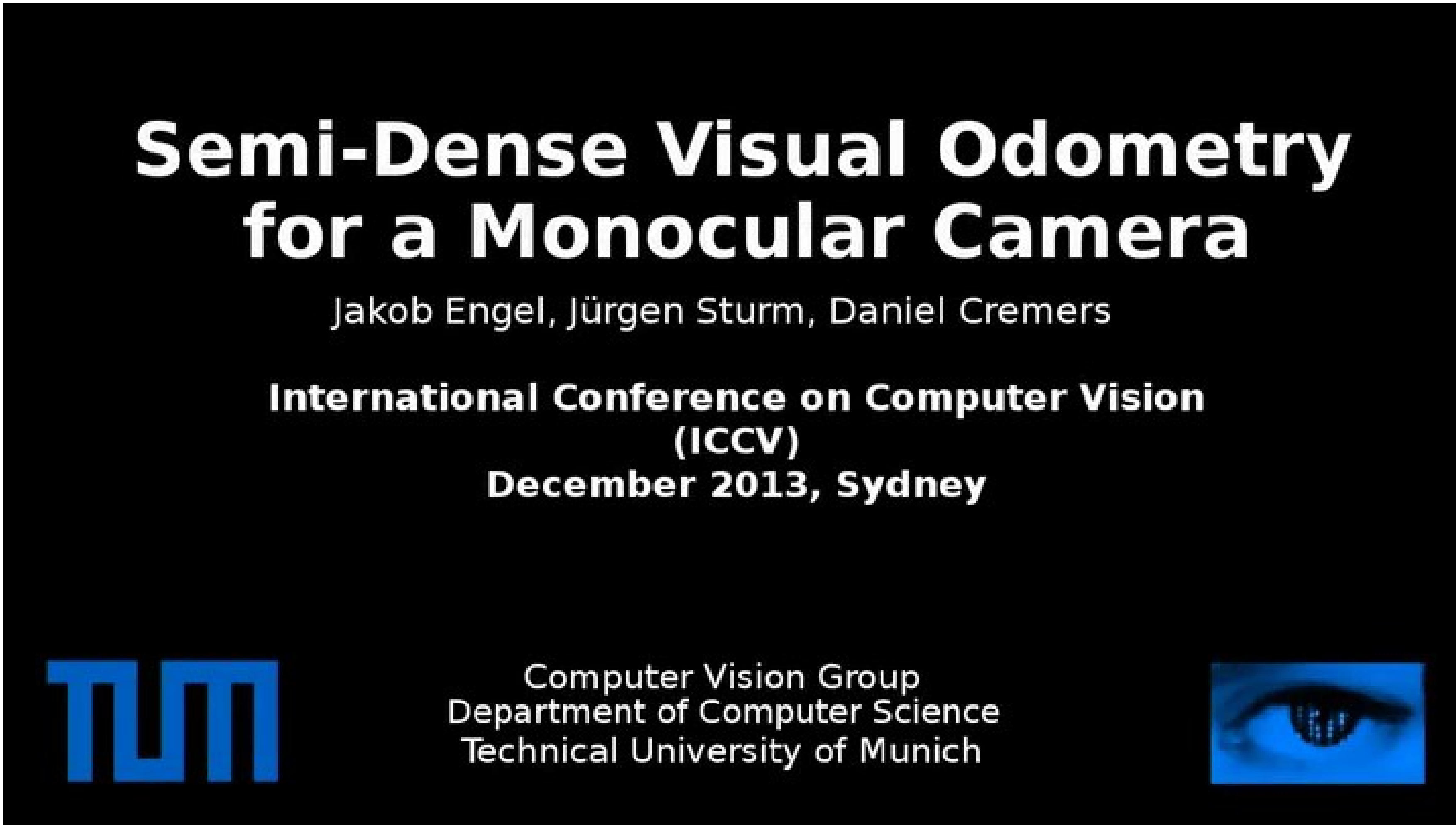
[Steinbrücker, Sturm, Cremers; ICRA 2014]

- Runs at 45 fps on CPU, available as open-source!



Same with a Monocular Camera?
[Engel, Sturm, Cremers; ICCV 2013]

- Soon available as open-source!



Summary

- (Scientific) Take home messages:
  - Dense methods make better use of available data
  - Supersede sparse/feature-based approaches
  - Real-time visual SLAM and 3D reconstruction is there

- Dense visual odometry: simple, fast, efficient
- Dense visual SLAM: eliminates drift
- Dense 3D reconstruction: nice models

- Nice, but..
  But what do we need this for??

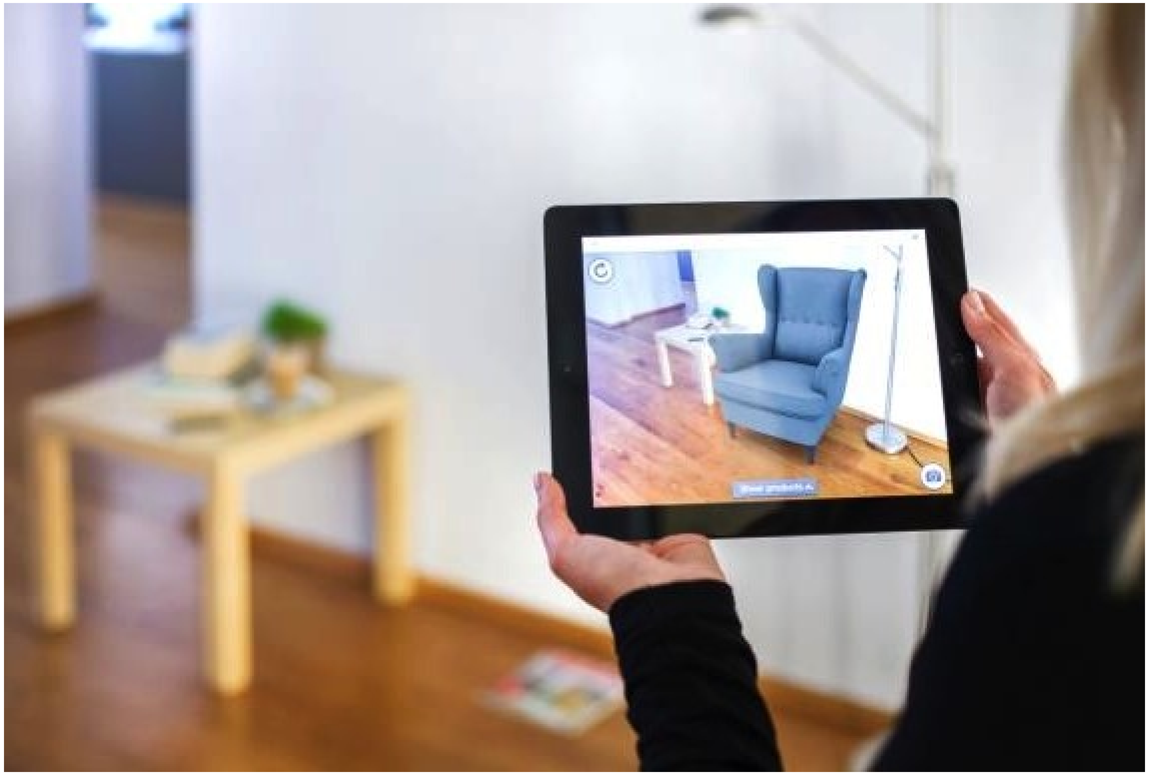
What do we need this for?

- Robotics
  - Laser scanners will eventually get replaced by (depth) cameras
  - Localization, mapping/SLAM, exploration, navigation
- Augmented reality (AR)
  - Games that play in your home
  - Virtual shopping: place furniture
  - User manuals: teach interactively how to repair/maintain a device

Key capabilities:

- Know how the camera is moving (odometry)
- Know where the camera is (absolute position)
- Know how the environment looks like (occlusion modeling, scene understanding)

The 2014 IKEA Catalog App (powered by metaio SDK)



Utilizes next-generation SLAM tracking to place furniture in home, easily and conveniently

Influences and educates purchasing decision while driving massive brand awareness

http://www.youtube.com/watch?v=vDNzTasuYEw

Volkswagen XL1 MARTA (powered by metaio SDK)

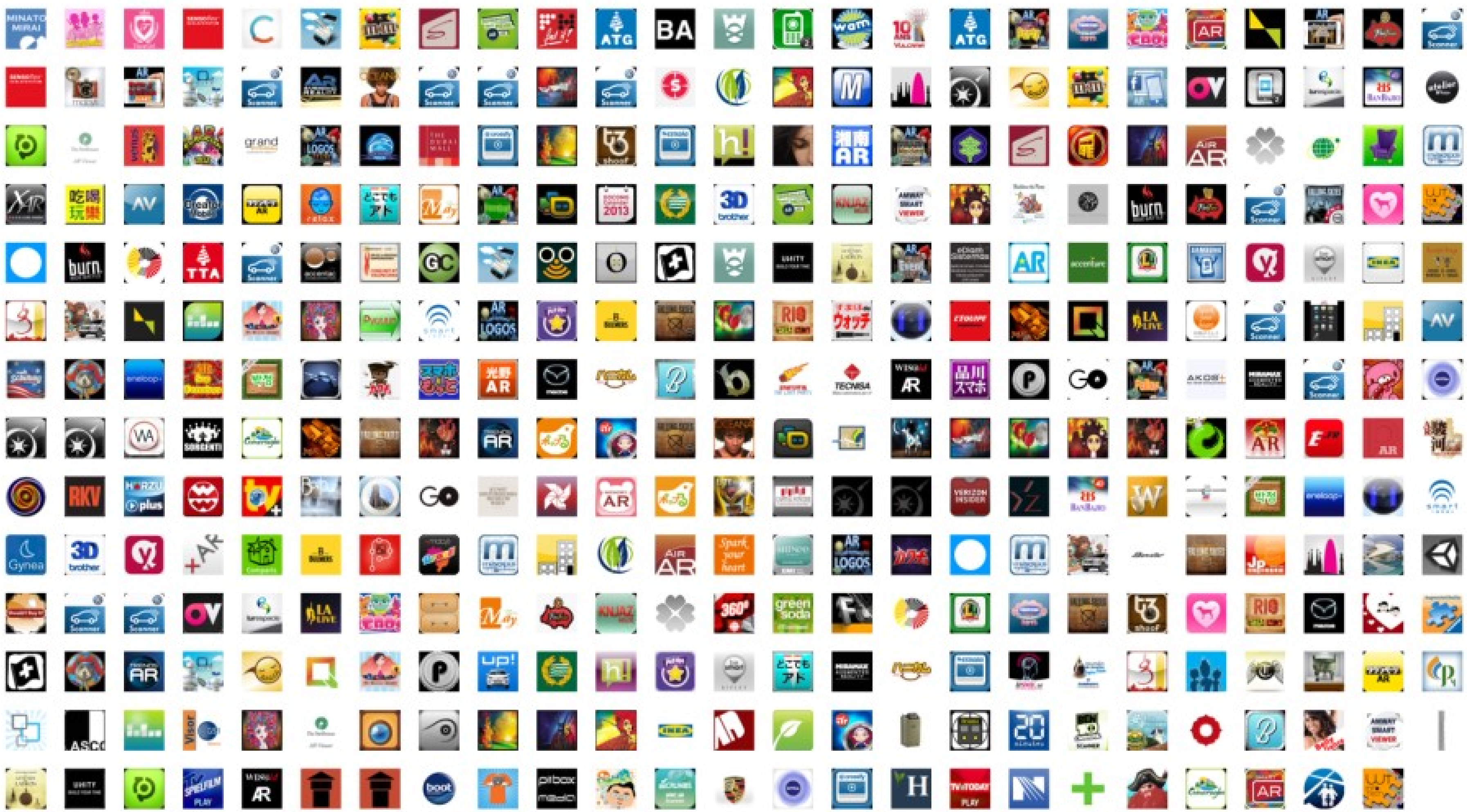
First-ever integrated AR support system for service technicians

Visualizes and overlays animated step-by-step service instructions

Utilizes Metaio's most robust 2D and 3D AR tracking technology.

http://www.youtube.com/watch?v=h2l3VzrkmRY

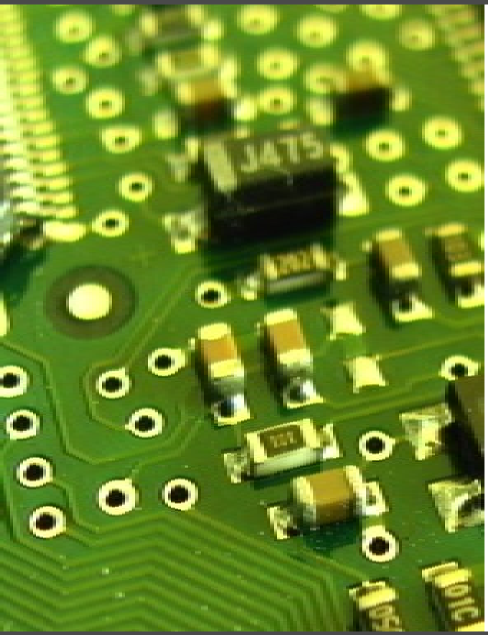
Some of the AR apps based on metaio SDK



metaio – A Brief introduction

- ✓ **ONLY dedicated company** to serve the entire AR value chain
- ✓ **10+ years of professional experience** in AR development
- ✓ **130+ people** working in Germany (HQ) and the USA
- ✓ **1000+ B2B** customers worldwide
- ✓ **100,000+ active developers** across the world
- ✓ **5million+ downloads** of metaio´s AR browser (junaio)

AR Food Chain – Always ON, Always Augmented!



Hardware	Software	Content	Users
(AREngine)	AR Applications	AR Usage	AR Content Access

We're hiring!

Metaio
Phone (EMEA): +49-89-5480-198-0
Phone (US): +1-415-814-3376
info@metaio.com
www.metaio.com

http://www.facebook.com/metaio

@twitt_AR
http://twitter.com/#!/twitt_AR

http://augmentedblog.wordpress.com/

http://www.youtube.com/user/metaioAR