

# Autonomous Navigation for Flying Robots

## Lecture 8.2: Tracking and Mapping using Signed Distance Functions

Jürgen Sturm

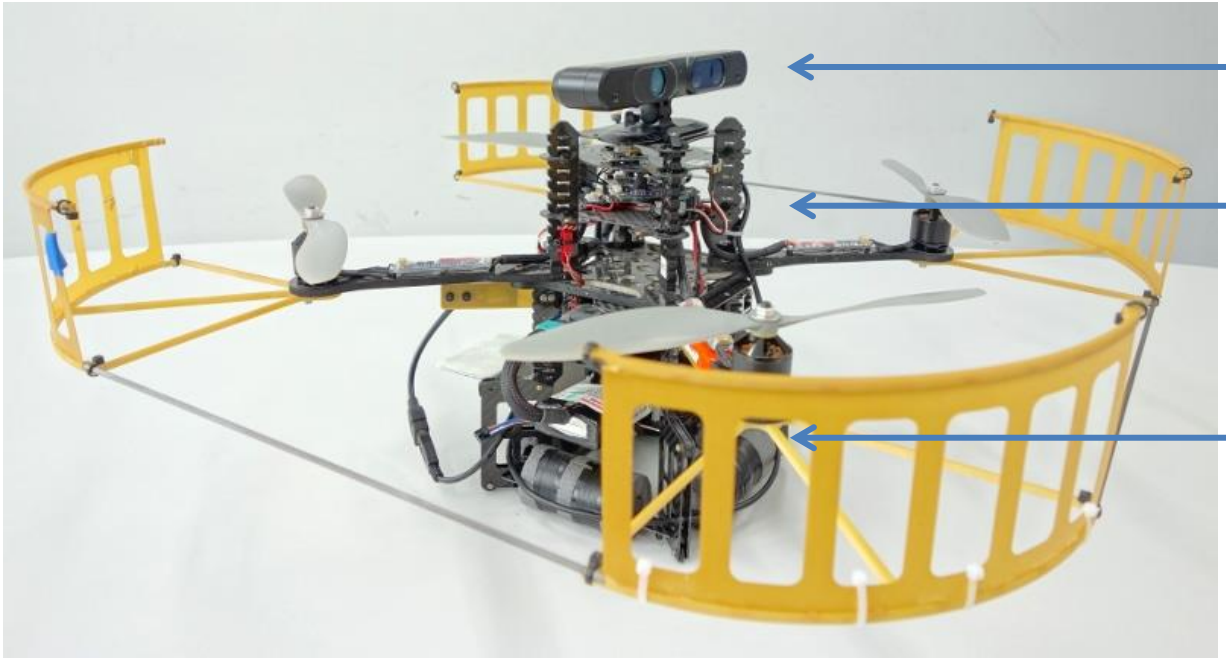
Technische Universität München

# 3D Reconstruction with a Quadrotor

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


**Real-Time Camera Tracking and 3D Reconstruction Using Signed Distance Functions** (E. Bylow, J. Sturm, C. Kerl, F. Kahl, D. Cremers), *In Robotics: Science and Systems Conference (RSS)*, 2013. http://youtu.be/MzLdRFSrtuI

# AscTec Pelican Platform



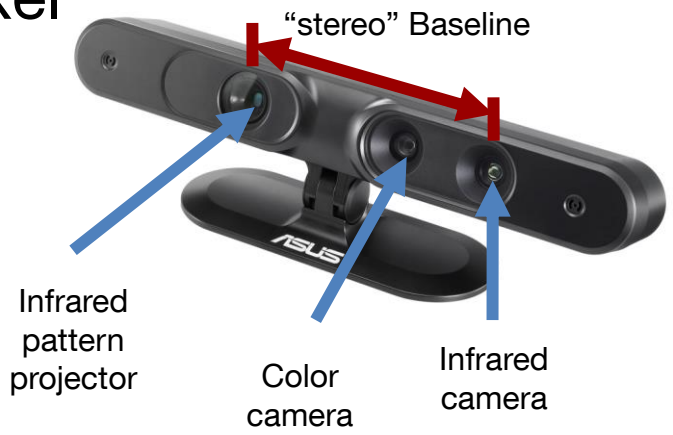
RGB-D sensor
(color and depth)

Autopilot board
(IMU, attitude
& position control)

Intel Core2Duo
running Ubuntu
(state estimation)

# Depth Cameras

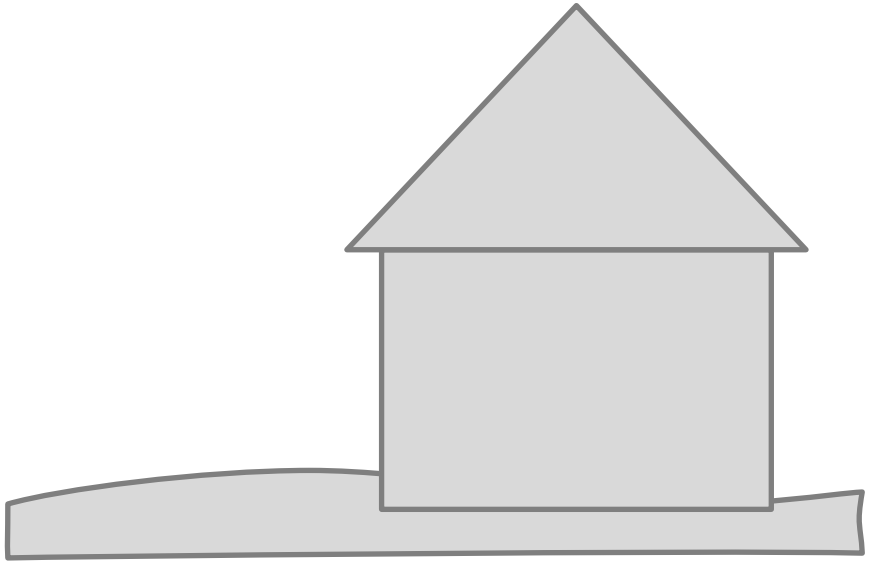
- Camera measures depth of every pixel
- Different sensing principles exist
  - Stereo cameras
  - Time-of-flight
  - Structured light (e.g., Kinect)



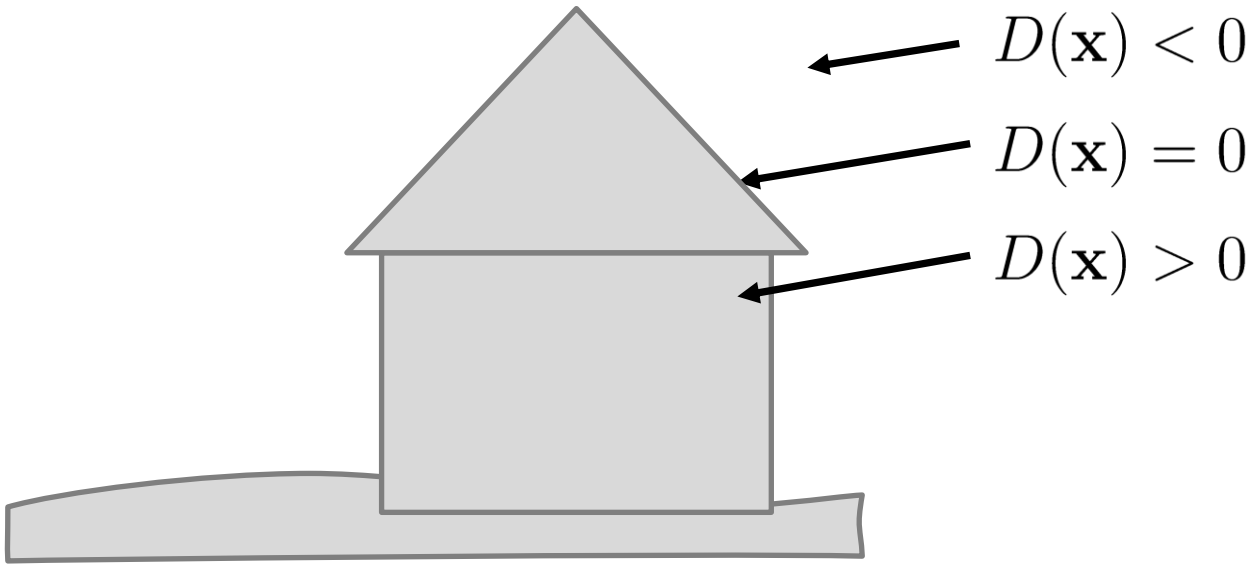
# Related Work

- Signed Distance Functions [Curless and Levoy, 1996]
  - Represent distance to surface in a voxel grid
  - Data fusion of depth images
- KinectFusion [Newcombe et al., 2011]
  - Generate synthetic depth image from SDF
  - ICP between current and synthetic image
- Our approach
  - Estimate the camera pose **directly** using the SDF

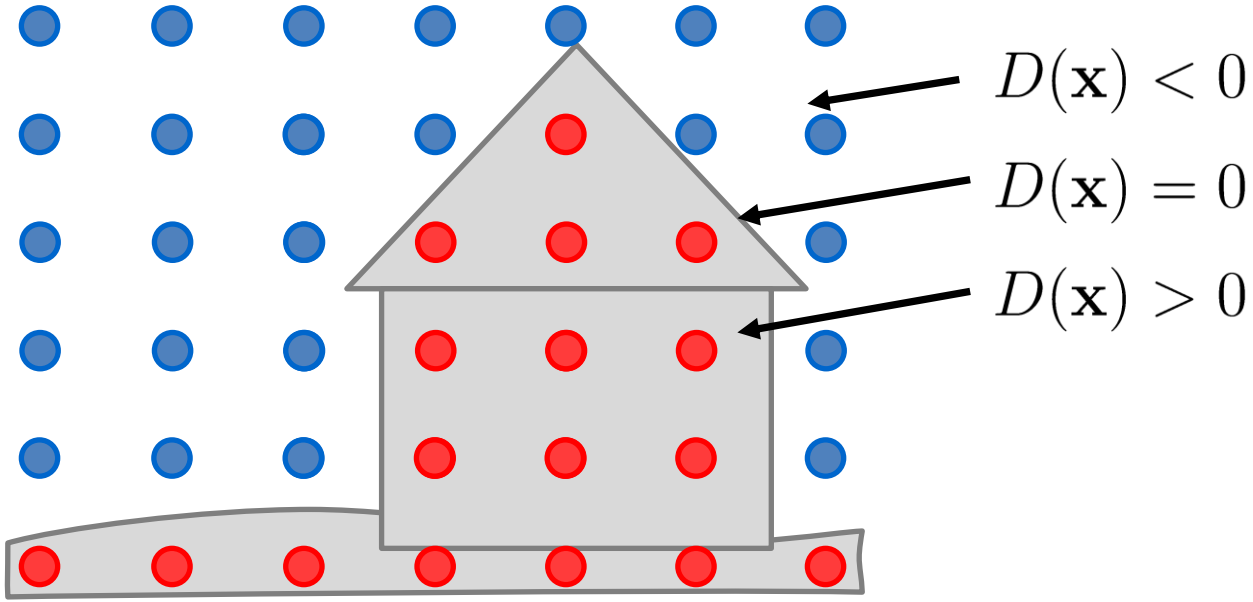
# Signed Distance Functions



# Signed Distance Functions



# Signed Distance Functions

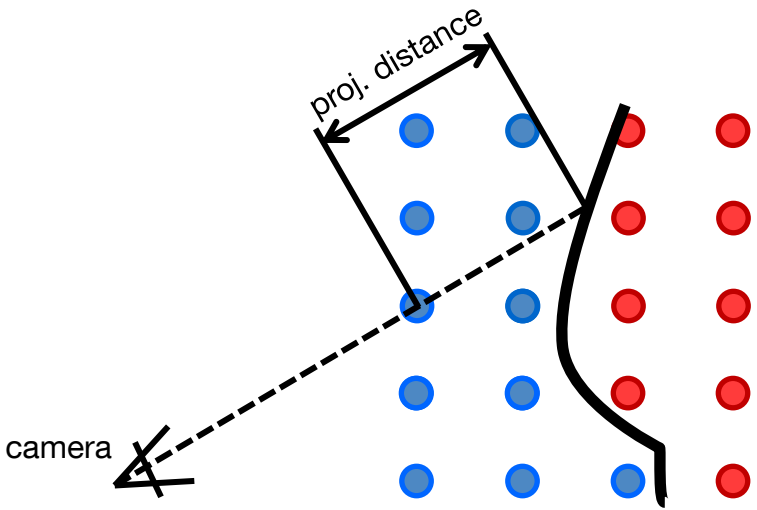



Negative distance to surface (= outside)

Positive distance to surface (= inside)

# Signed Distance Functions

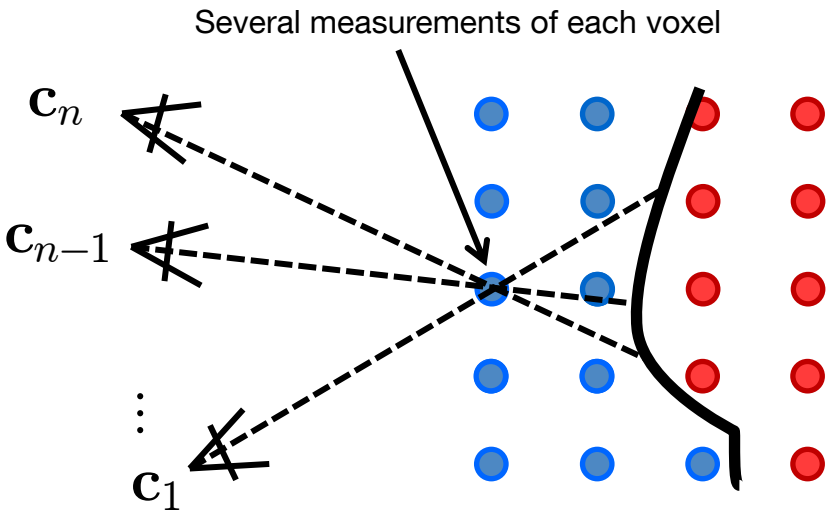
- Compute SDF from a depth image
- Measure distance of each voxel to the observed surface




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

# Signed Distance Functions

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



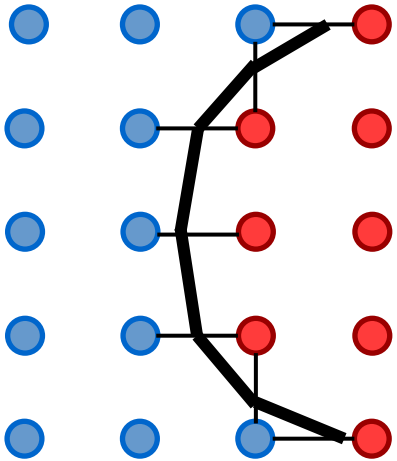

$$D \leftarrow \frac{WD + wd}{W + w}$$

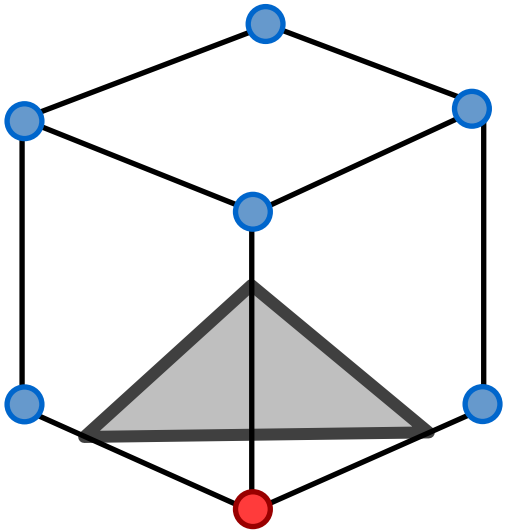
$$C \leftarrow \frac{WC + wc}{W + w}$$

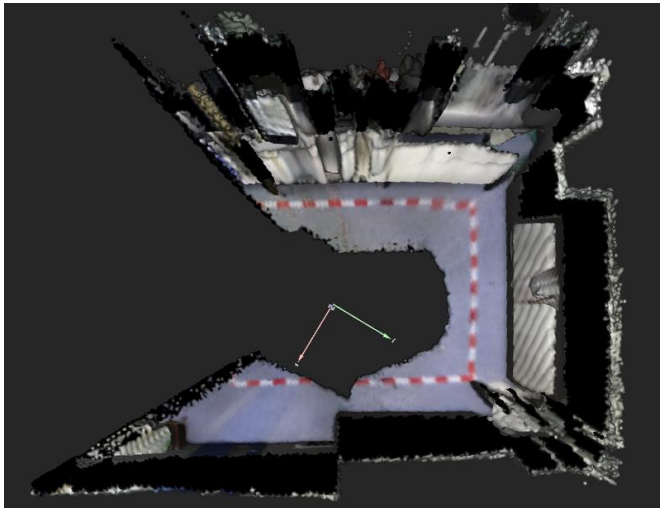
$$W \leftarrow W + w$$

# Mesh Extraction

- **Marching cubes:** Find zero-crossings in the signed distance function by interpolation

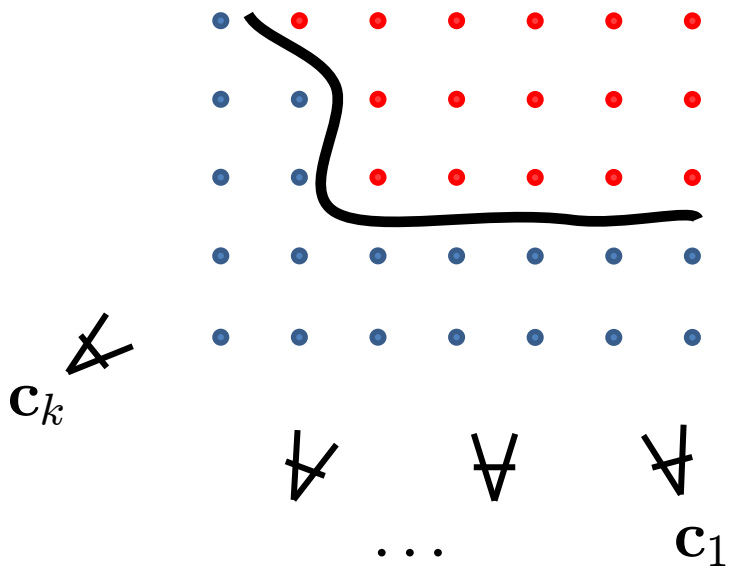






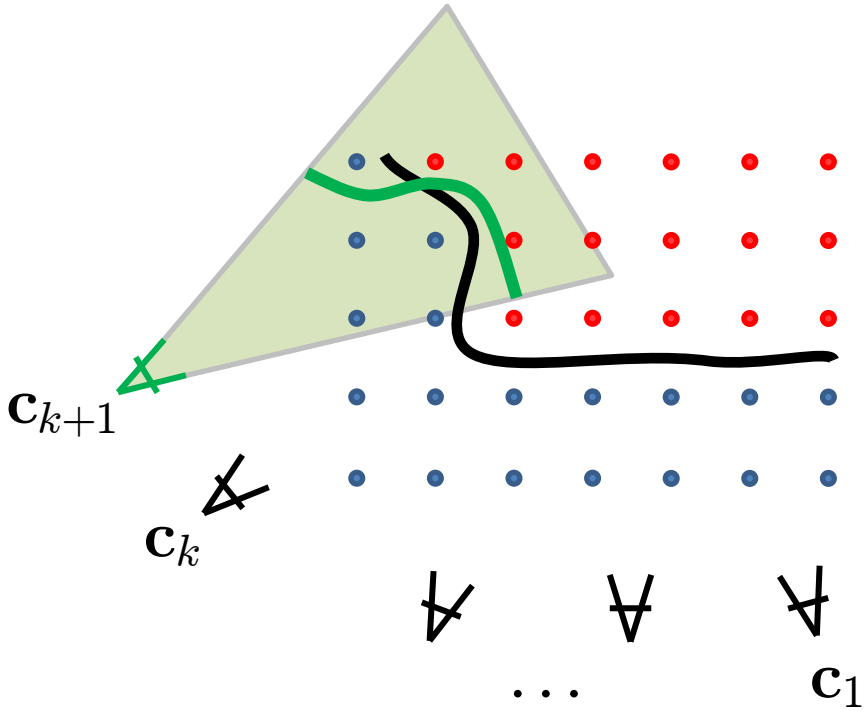
# Estimating the Camera Pose

- SDF built from the first k frames



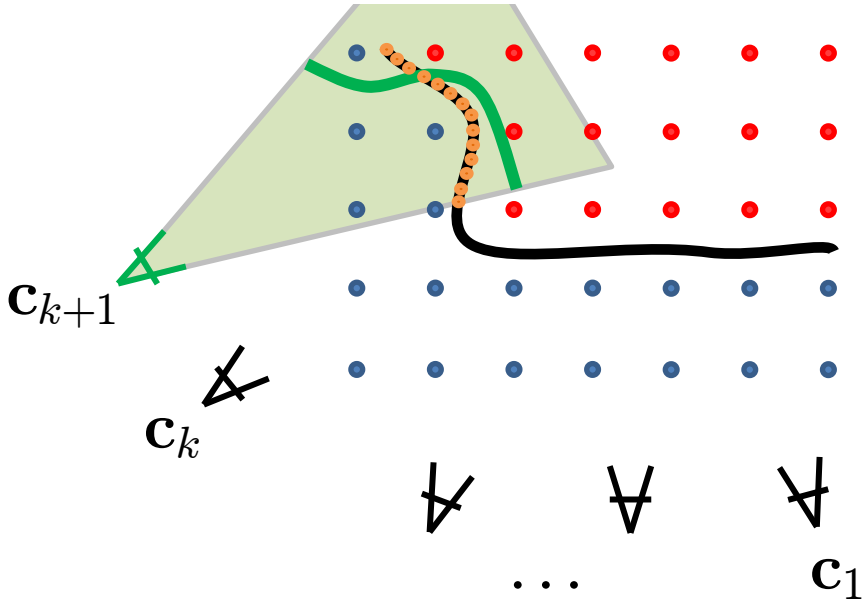
# Estimating the Camera Pose

- We seek the next camera pose $\mathbf{c}_{k+1}$



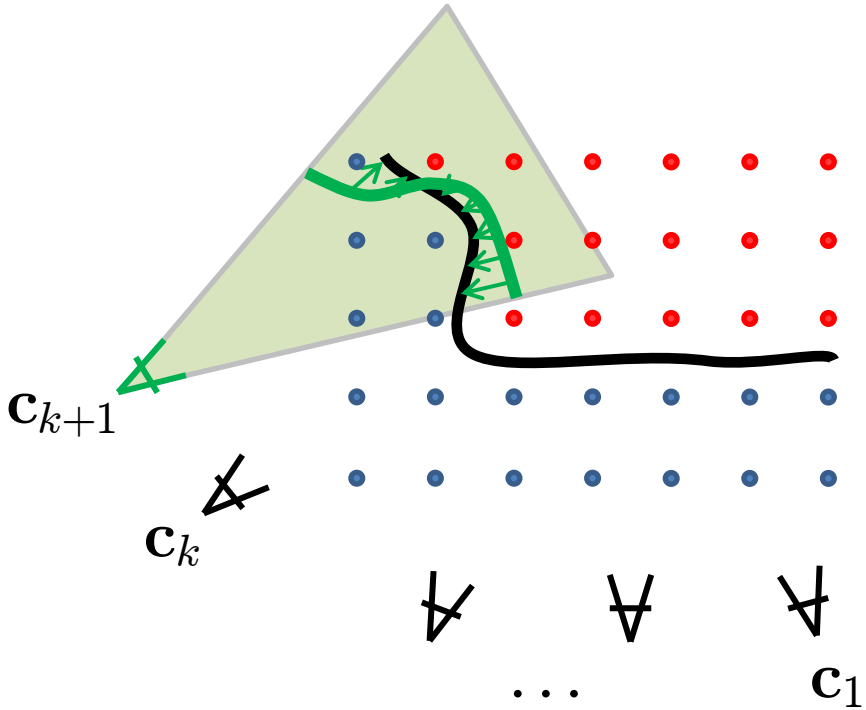
# Estimating the Camera Pose

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



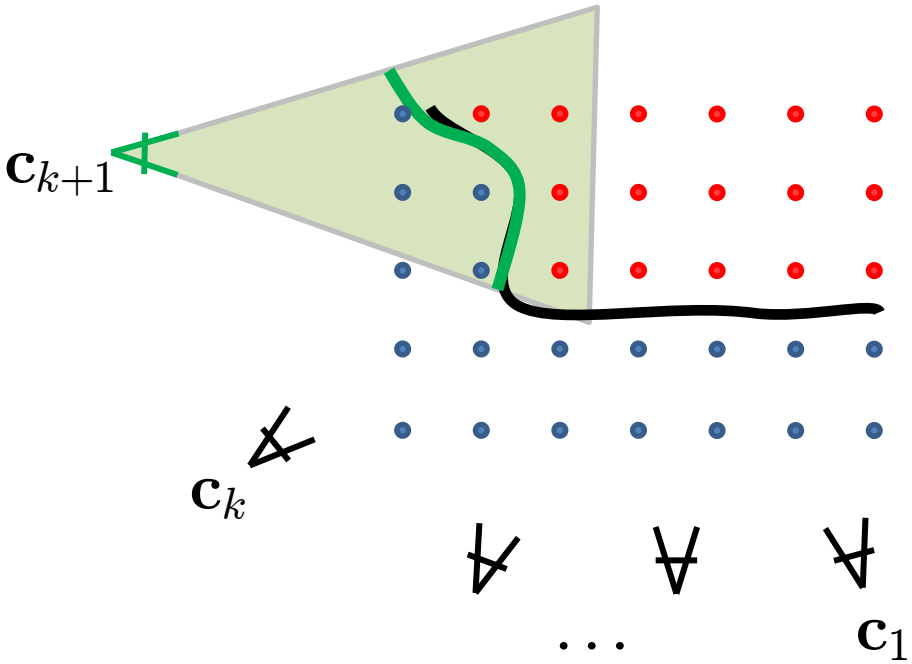
# Estimating the Camera Pose

- **Our approach:** Use SDF directly during minimization



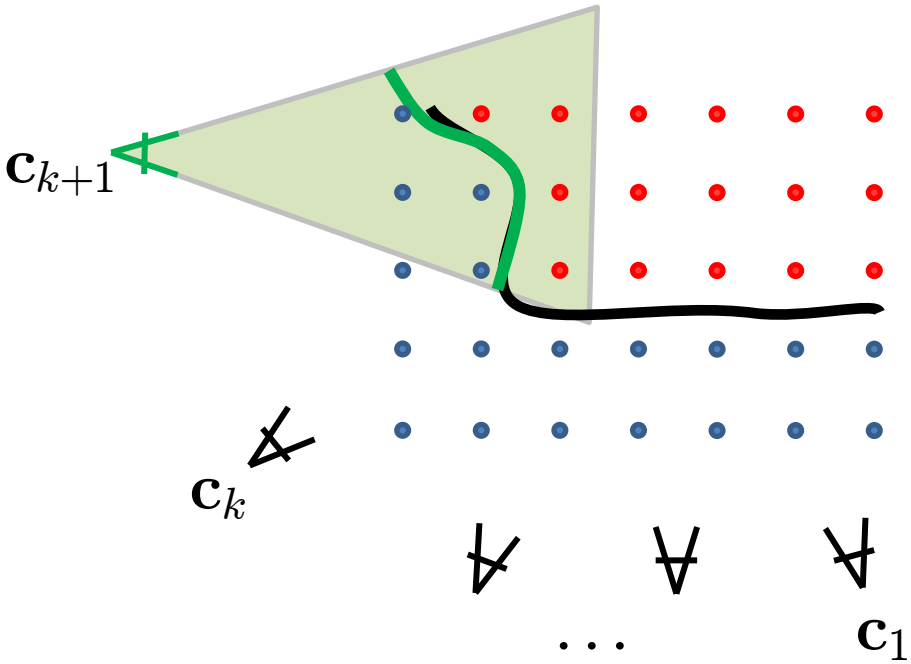
# Estimating the Camera Pose

- **Our approach:** Use SDF directly during minimization



# Estimating the Camera Pose

- **Our approach:** Use SDF directly during minimization

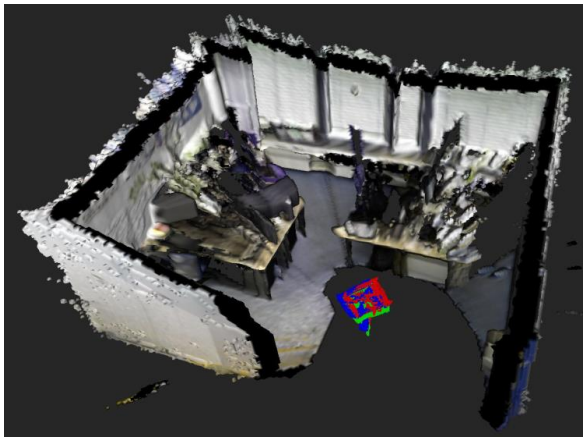
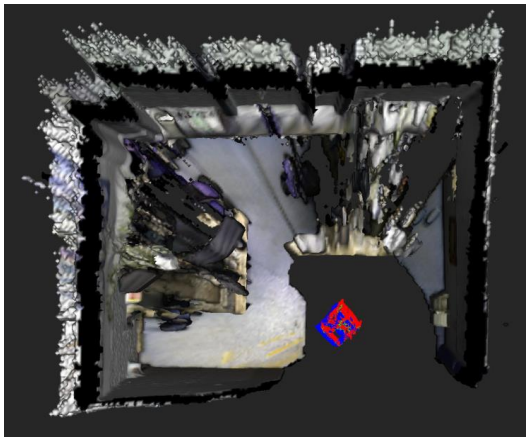



$$\arg\min_{\mathbf{R},\mathbf{t}} \sum_{ij} D(\mathbf{R}\mathbf{x}_{ij} + \mathbf{t})^2$$

# Mapping with a Quadrotor

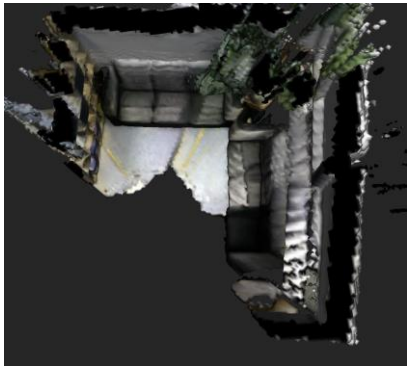
- Equip quadrotor with depth camera
- Estimate quadrotor pose in real-time (offboard, needs GPU)
- Position control
- Follow trajectory
- Extract dense 3D model from SDF

# Results on 3D Mapping

Dense Tracking and Mapping with a Quadrocopter (J. Sturm, E. Bylow, F. Kahl, D. Cremers), In Unmanned Aerial Vehicle in Geomatics (UAV-g), 2013.

# More Results


Dense Tracking and Mapping with a Quadrocopter (J. Sturm, E. Bylow, F. Kahl, D. Cremers), In Unmanned Aerial Vehicle in Geomatics (UAV-g), 2013.

# 3D Person Scanning and Printing

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

CopyMe3D: Scanning and Printing Persons in 3D (J. Sturm, E. Bylow, F. Kahl, D. Cremers), In German Conference on Pattern Recognition (GCPR), 2013.
http://youtu.be/9B2IdD2jHGw

Download demo version
from http://fablitec.com

# Lessons Learned

- Depth cameras
- Signed distance functions
- Pose tracking on SDFs
- Real-time position control of a quadrotor
- 3D reconstruction