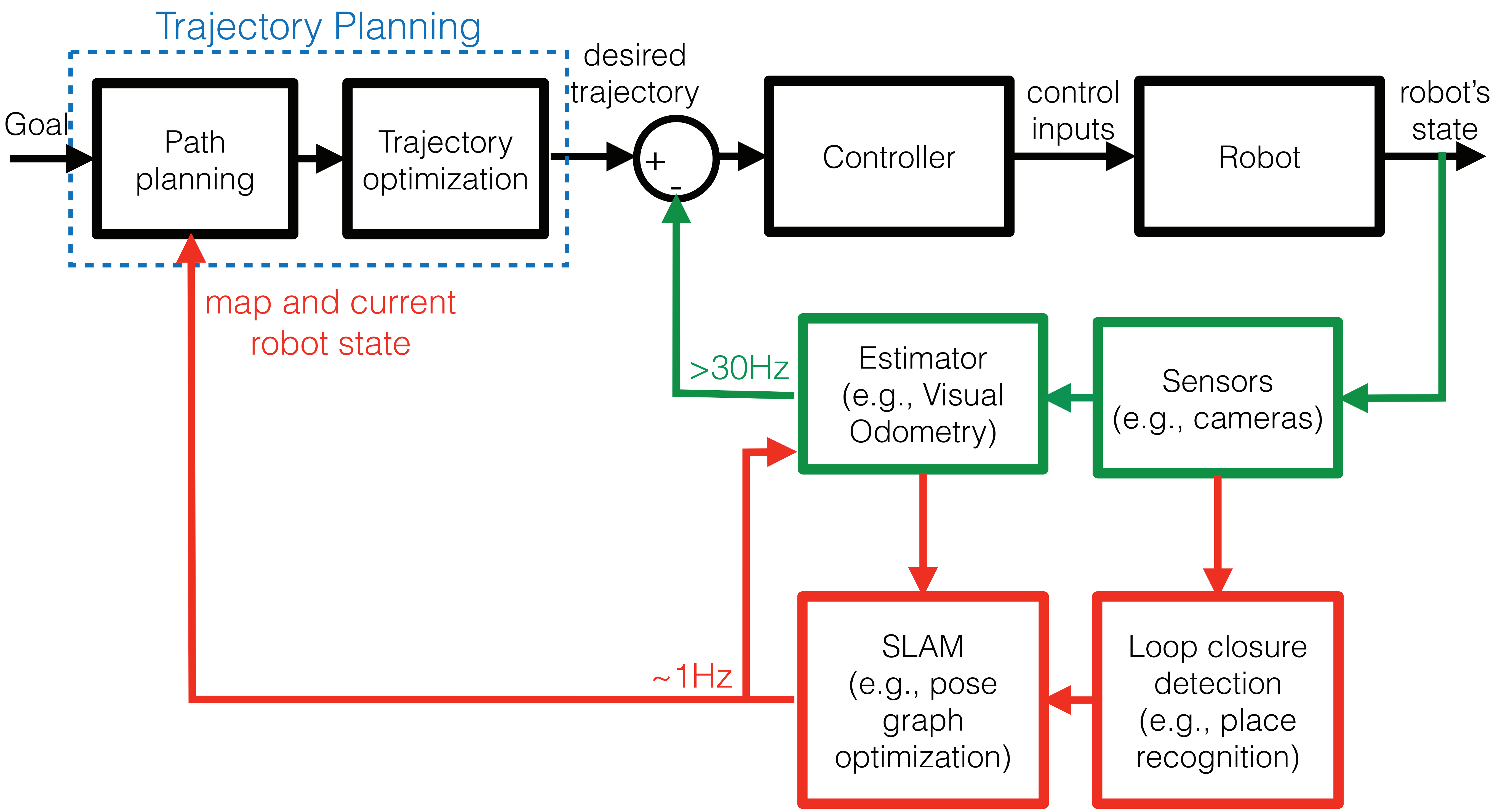
16.485: VNAV - Visual Navigation for Autonomous Vehicles

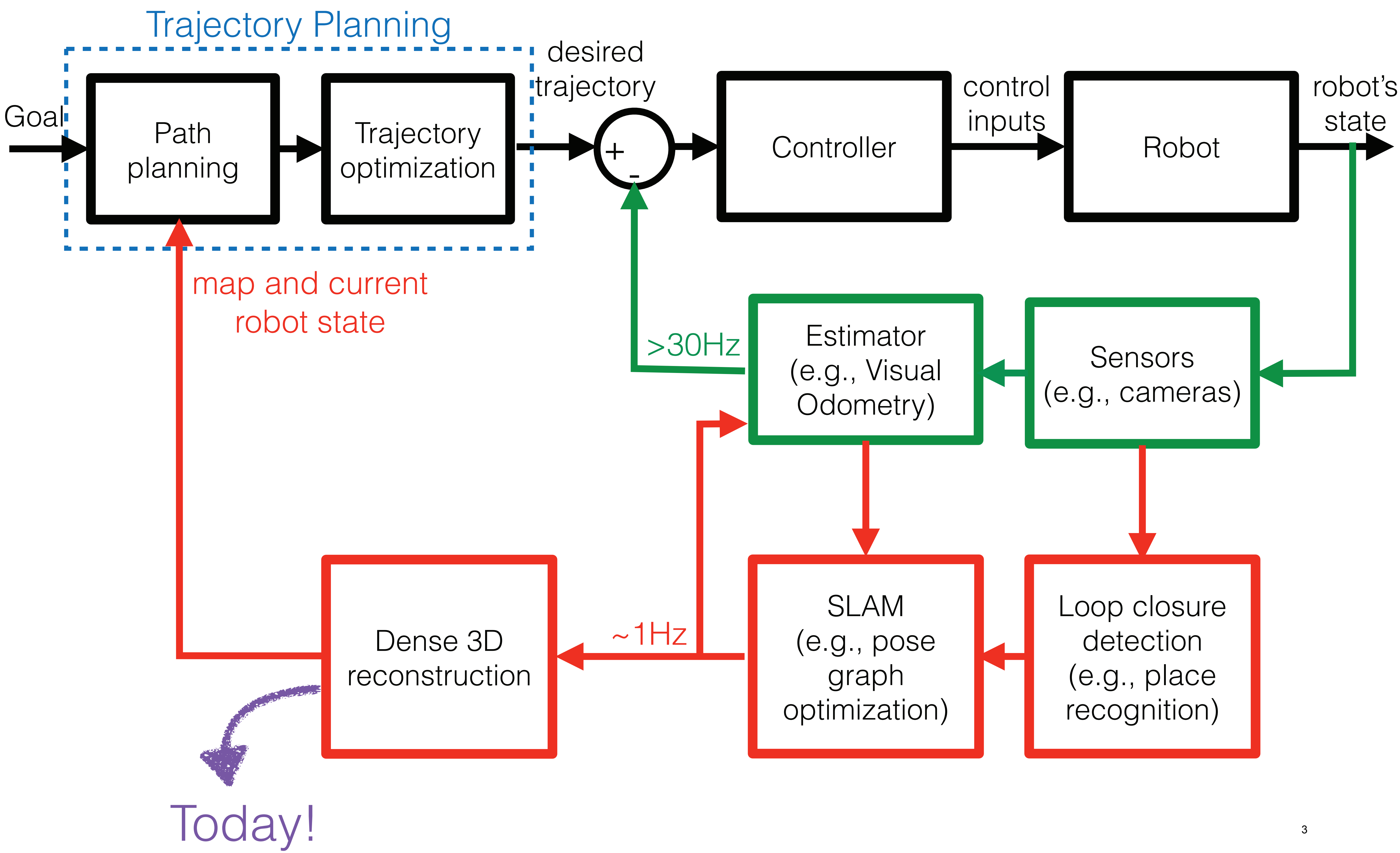
Luca Carlone

Lecture 25: Advanced topics: Dense 3D Reconstruction

Big Picture


Trajectory Planning
Goal
Path planning
Trajectory optimization
desired trajectory
+
-
Controller
control inputs
Robot
robot's state
map and current robot state
>30Hz
Estimator (e.g., Visual Odometry)
Sensors (e.g., cameras)
~1Hz
SLAM (e.g., pose graph optimization)
Loop closure detection (e.g., place recognition)

Big Picture


Trajectory Planning
Goal
Path planning
Trajectory optimization
desired trajectory
+
-
Controller
control inputs
Robot
robot's state
map and current robot state
>30Hz
Estimator (e.g., Visual Odometry)
Sensors (e.g., cameras)
Dense 3D reconstruction
~1Hz
SLAM (e.g., pose graph optimization)
Loop closure detection (e.g., place recognition)
Today!

Today

- Dense Reconstruction
  - 3D representations
  - (Some) Multi-view Stereo
  - Depth fusion
- Final thoughts

Multi-View Stereo: A Tutorial

2015

Yasutaka Furukawa
Washington University in St. Louis
furukawa@wustl.edu

Carlos Hernández
Google Inc.
carloshernandez@google.com

ElasticFusion: Dense SLAM Without A Pose Graph

Thomas Whelan*, Stefan Leutenegger*, Renato F. Salas-Moreno†, Ben Glocker† and Andrew J. Davison*
*Dyson Robotics Laboratory at Imperial College, Department of Computing, Imperial College London, UK
†Department of Computing, Imperial College London, UK
{t.whelan,s.leutenegger,r.salas-moreno10,b.glocker,a.davison}@imperial.ac.uk

2016

KinectFusion: Real-Time Dense Surface Mapping and Tracking*

Richard A. Newcombe, Imperial College London
Shahram Izadi, Microsoft Research
Otmar Hilliges, Microsoft Research
David Molyneaux, Microsoft Research, Lancaster University
David Kim, Microsoft Research, Newcastle University
Andrew J. Davison, Imperial College London
Pushmeet Kohli, Microsoft Research
Jamie Shotton, Microsoft Research
Steve Hodges, Microsoft Research
Andrew Fitzgibbon, Microsoft Research

2011

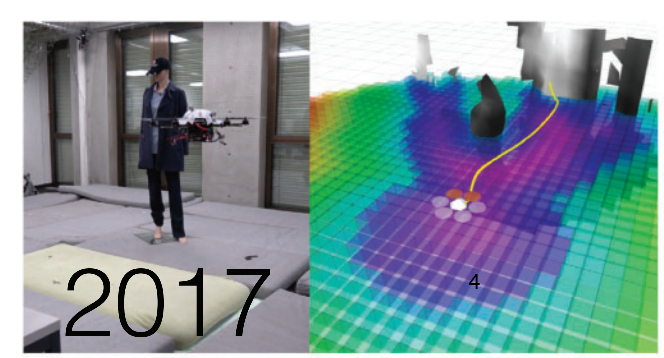
Figure 1: Example output from our system, generated in real-time with a handheld Kinect depth camera and no other sensing infrastructure. Normal maps (colour) and Phong-shaded renderings (greyscale) from our dense reconstruction system are shown. On the left for comparison is an example of the live, incomplete, and noisy data from the Kinect sensor (used as input to our system).

Voxblox: Incremental 3D Euclidean Signed Distance Fields for On-Board MAV Planning

Helen Oleynikova, Zachary Taylor, Marius Fehr, Roland Siegwart, and Juan Nieto
Autonomous Systems Lab, ETH Zürich

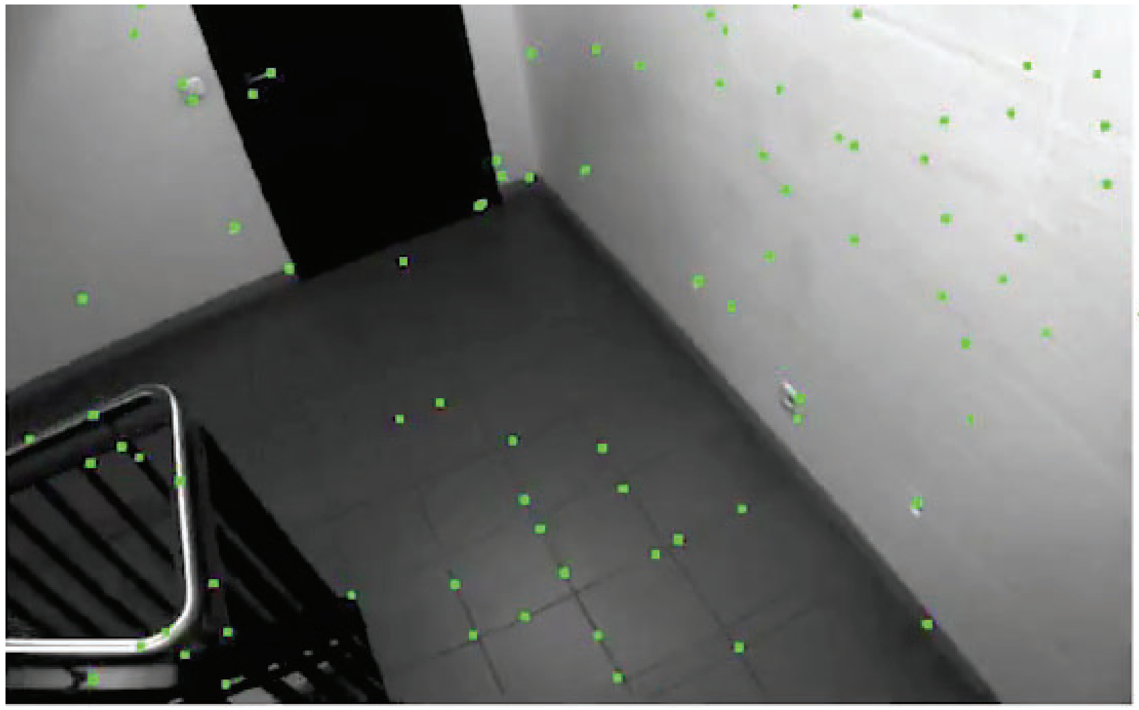
***Abstract*—Micro Aerial Vehicles (MAVs) that operate in unstructured, unexplored environments require fast and flexible local planning, which can replan when new parts of the map are explored. Trajectory optimization methods fulfill these needs, but require obstacle distance information, which can be given by Euclidean Signed Distance Fields (ESDFs).**

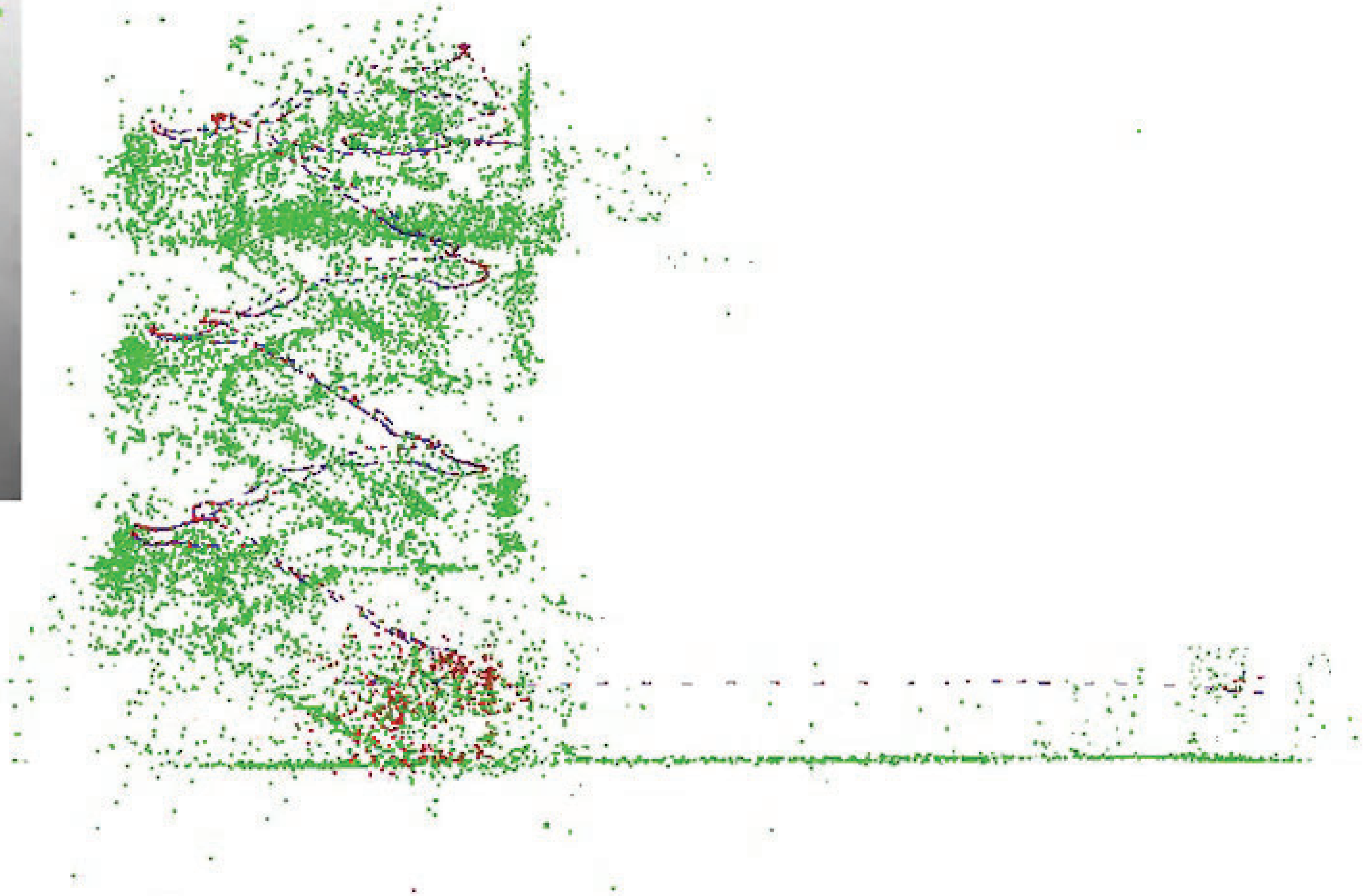
We propose a method to incrementally build ESDFs from Truncated Signed Distance Fields (TSDFs), a common implicit surface representation used in computer graphics and vision. TSDFs are fast to build and smooth out sensor noise over many observations, and are designed to produce surface meshes. Meshes allow human operators to get a better assessment of the robot's environment, and set high-level mission goals.



2017

Point Clouds

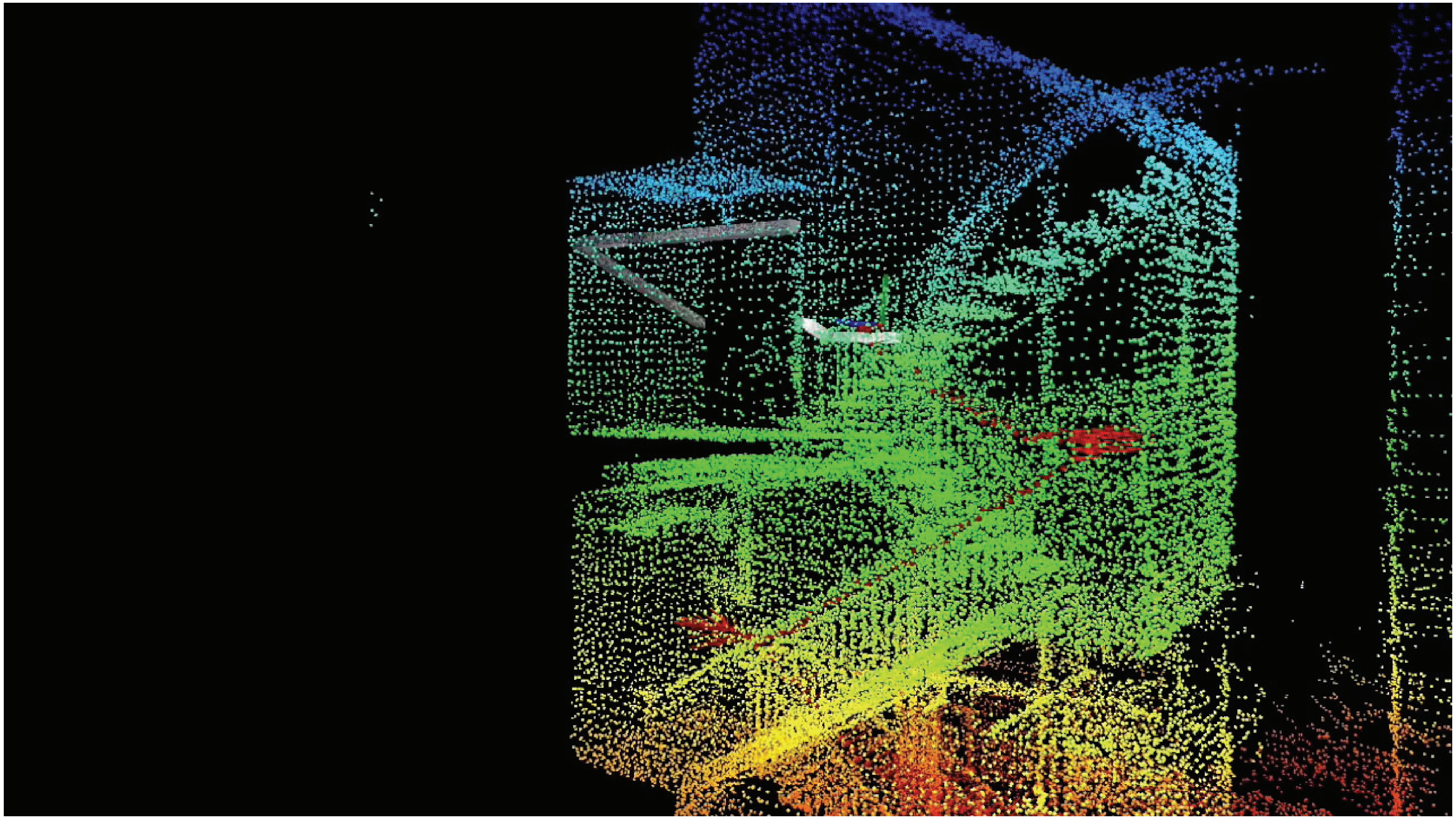




Point Clouds	✗	✓/✗ No, if Dense	X	✓/✗ No, if Sparse

5

Point Clouds

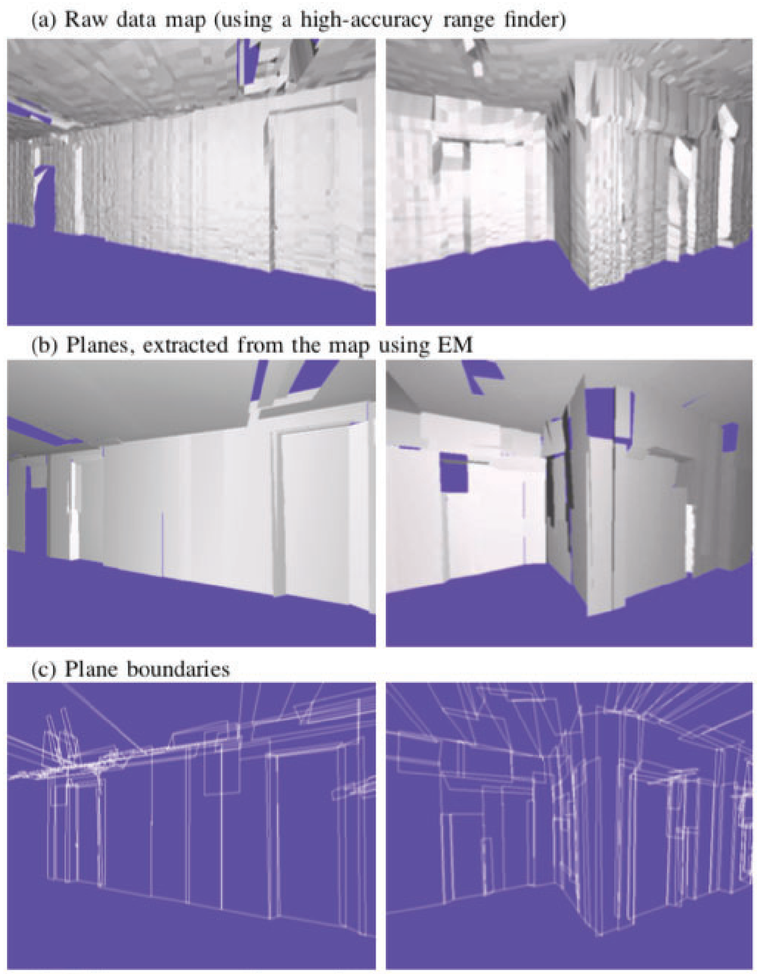


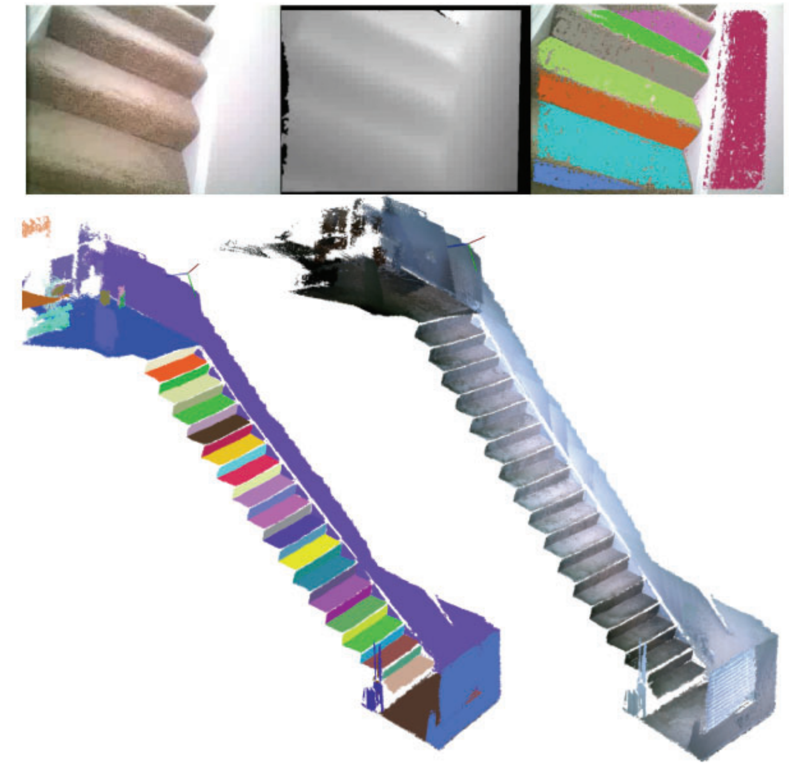
Map representation	3D Topology?	Lightweight?	Filters Noise/ Outliers?	Semantics?	Generality
Point Clouds	✗	✓/✗ No, if Dense	X	✓/✗ No, if Sparse	✓

Geometric Primitives

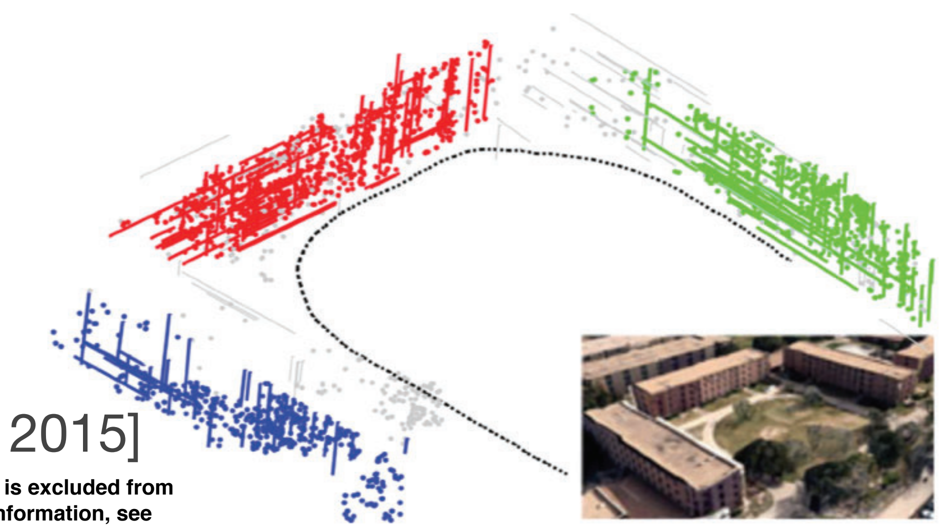
Point, lines, planes

[Thrun et al. 2004]





[Kaess 2015]

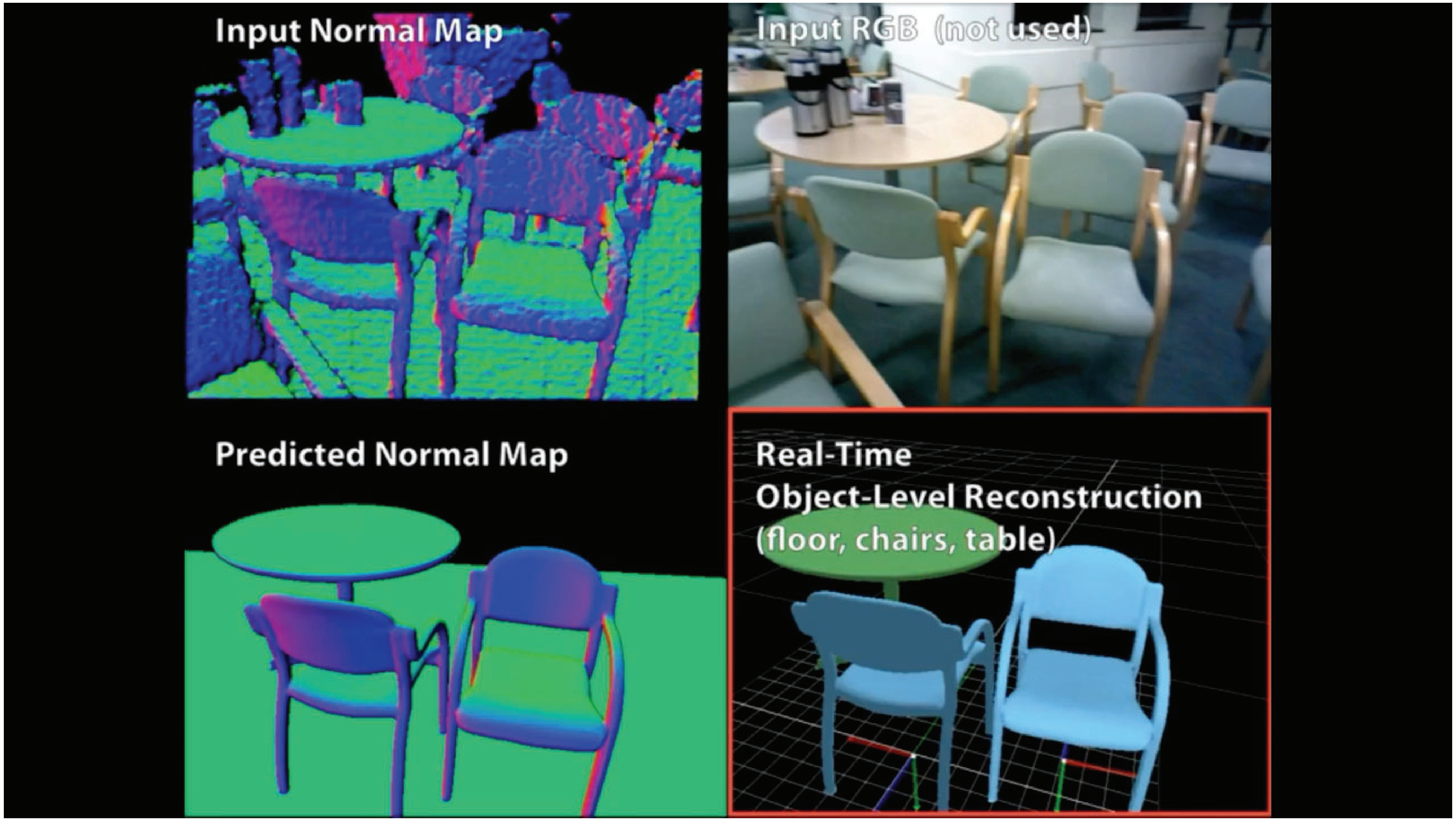


[Lu et al. 2015]

Map representation	3D Topology?	Lightweight?	Filters Noise/ Outliers?	Semantics?	Generality
Point Clouds	✗	✓/✗ No, if Dense	X	✓/✗ No, if Sparse	✓
Geometric primitives	✗	✓	✓	✓/✗ No, if Sparse	✗

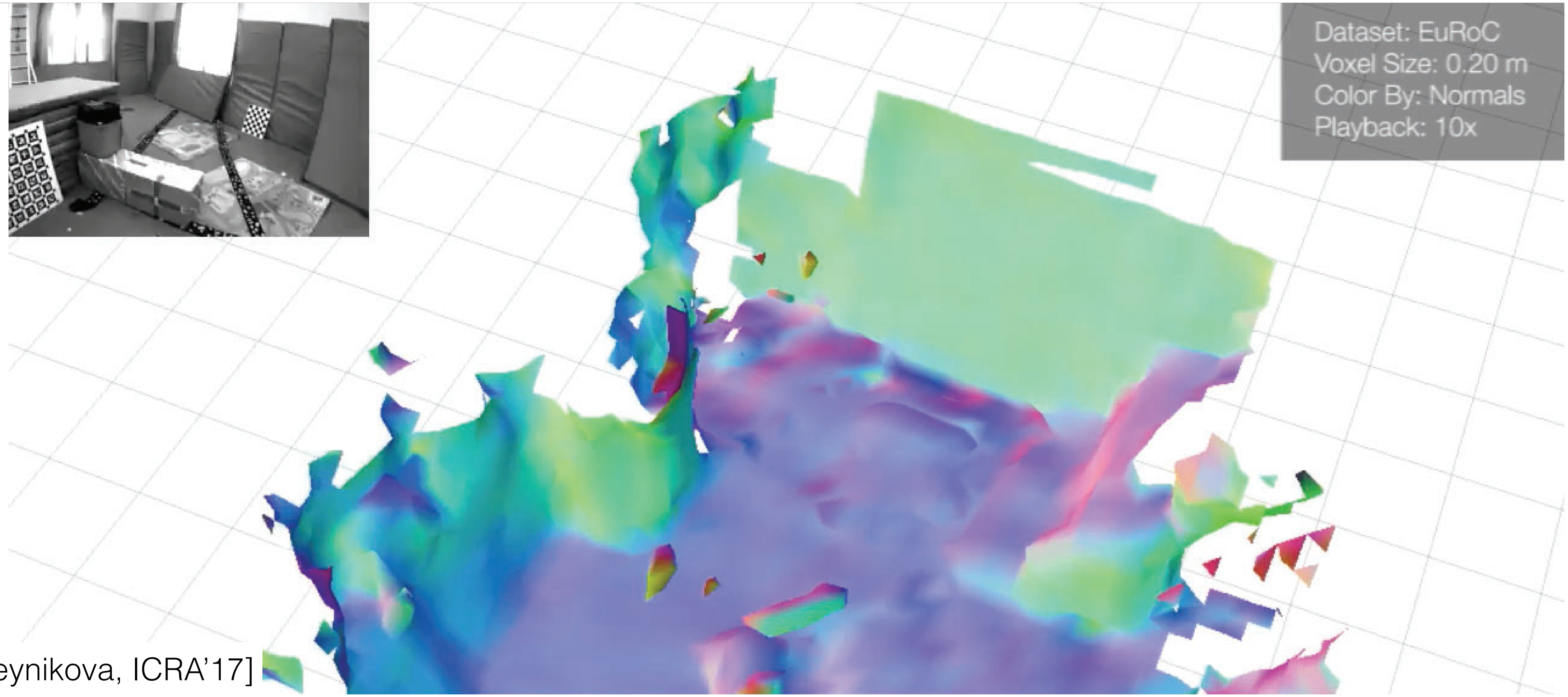
Object-based Maps

[Salas-Moreno et al, 2014]



Map representation	3D Topology?	Lightweight?	Filters Noise/ Outliers?	Semantics?	Generality
Point Clouds	✗	✓/✗ No, if Dense	X	✓/✗ No, if Sparse	✓
primitives & **objects**	✗	✓	✓	✓/✗ No, if Sparse	✗

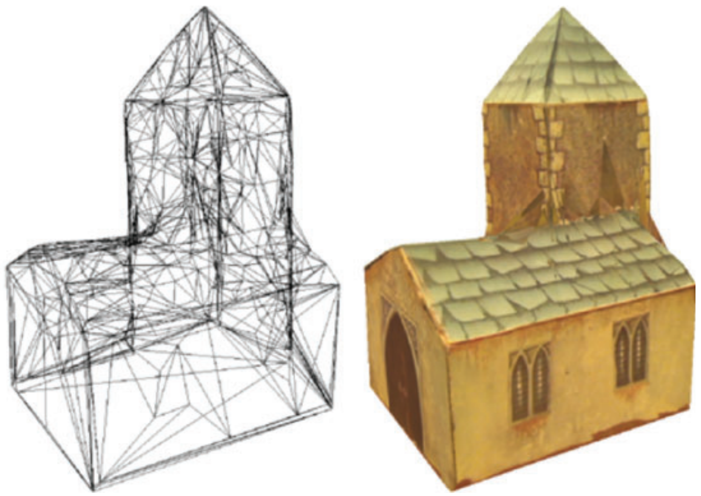
Volumetric Methods: Voxels/Octrees

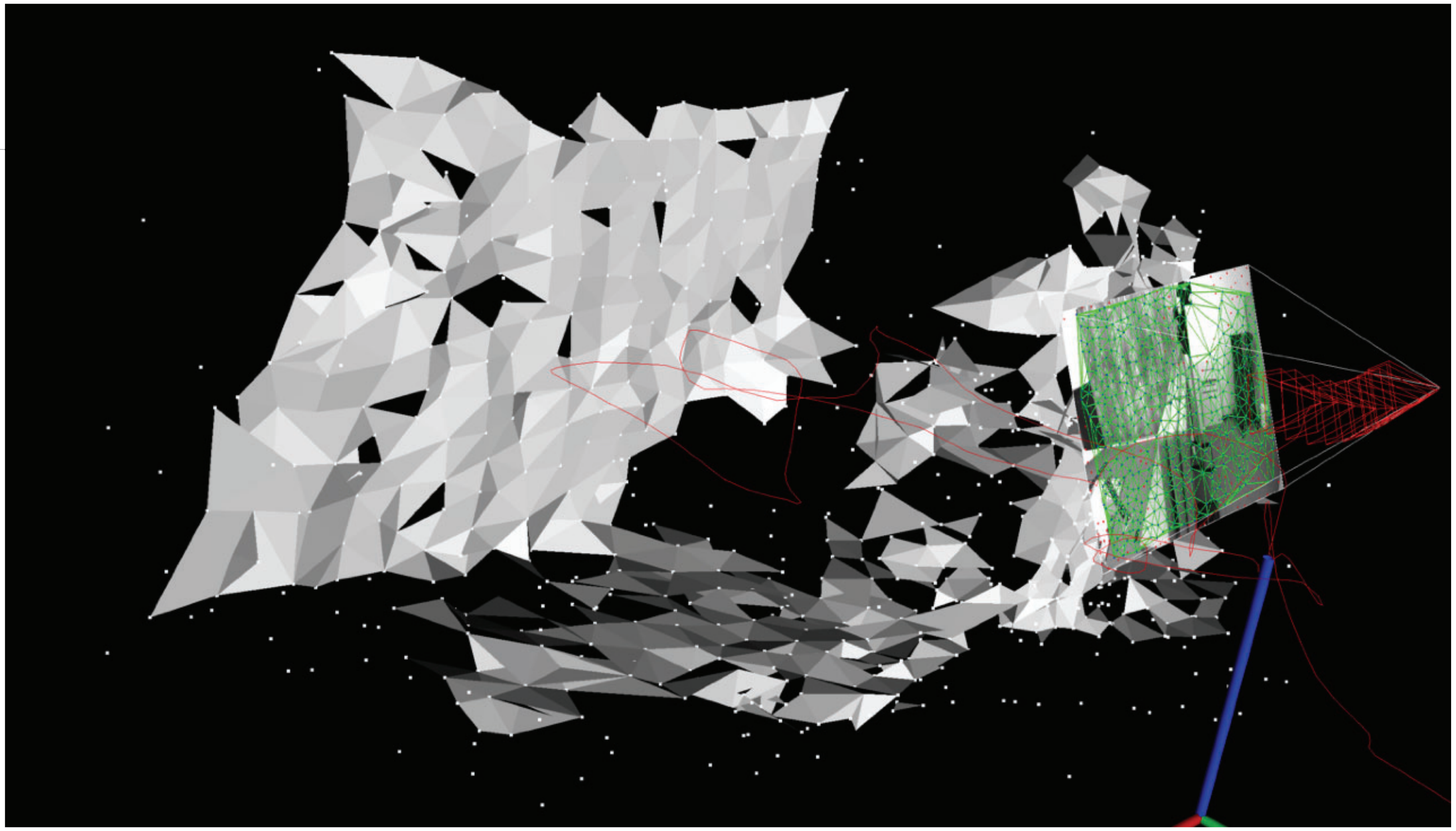


[Oleynikova, ICRA'17]

Map representation	3D Topology?	Lightweight?	Filters Noise/Outliers?	Semantics?	Generality
Point Clouds	✗	✓/✗ No, if Dense	X	✓/✗ No, if Sparse	✓
primitives & objects	✗	✓	✓	✓/✗ No, if Sparse	✗
Voxels	✓	✓/✗ No, if small voxel	✓	✓/✗ No, if large	✓

Meshes





Map representation	3D Topology ?	Lightweight?	Filters Noise/ Outliers?	Semantics?	Generalit y
Point Clouds	✗	✓/✗ No, if Dense	X	✓/✗ No, if Sparse	✓
primitives & objects	✗	✓	✓	✓/✗ No, if Sparse	✗
Voxels	✓	✓/✗ No, if small voxel	✓	✓/✗ No, if large voxel	✓
3D Mesh	✓	✓	✗	✓	✓

10

Today

- Dense Reconstruction
  - 3D representations
  - (Some) Multi-view Stereo
  - Depth fusion
- Final thoughts

Multi-View Stereo: A Tutorial

2015

Yasutaka Furukawa
Washington University in St. Louis
furukawa@wustl.edu

Carlos Hernández
Google Inc.
carloshernandez@google.com

ElasticFusion: Dense SLAM Without A Pose Graph

Thomas Whelan*, Stefan Leutenegger*, Renato F. Salas-Moreno†, Ben Glocker† and Andrew J. Davison*
*Dyson Robotics Laboratory at Imperial College, Department of Computing, Imperial College London, UK
†Department of Computing, Imperial College London, UK
{t.whelan,s.leutenegger,r.salas-moreno10,b.glocker,a.davison}@imperial.ac.uk

2016

KinectFusion: Real-Time Dense Surface Mapping and Tracking*

Richard A. Newcombe, Imperial College London
Shahram Izadi, Microsoft Research
Otmar Hilliges, Microsoft Research
David Molyneaux, Microsoft Research, Lancaster University
David Kim, Microsoft Research, Newcastle University
Andrew J. Davison, Imperial College London
Pushmeet Kohli, Microsoft Research
Jamie Shotton, Microsoft Research
Steve Hodges, Microsoft Research
Andrew Fitzgibbon, Microsoft Research

2011

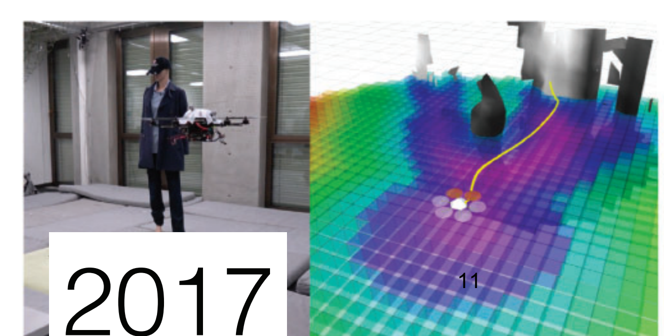
Figure 1: Example output from our system, generated in real-time with a handheld Kinect depth camera and no other sensing infrastructure. Normal maps (colour) and Phong-shaded renderings (greyscale) from our dense reconstruction system are shown. On the left for comparison is an example of the live, incomplete, and noisy data from the Kinect sensor (used as input to our system).

Voxblox: Incremental 3D Euclidean Signed Distance Fields for On-Board MAV Planning

Helen Oleynikova, Zachary Taylor, Marius Fehr, Roland Siegwart, and Juan Nieto
Autonomous Systems Lab, ETH Zürich

Abstract—Micro Aerial Vehicles (MAVs) that operate in unstructured, unexplored environments require fast and flexible local planning, which can replan when new parts of the map are explored. Trajectory optimization methods fulfill these needs, but require obstacle distance information, which can be given by Euclidean Signed Distance Fields (ESDFs).

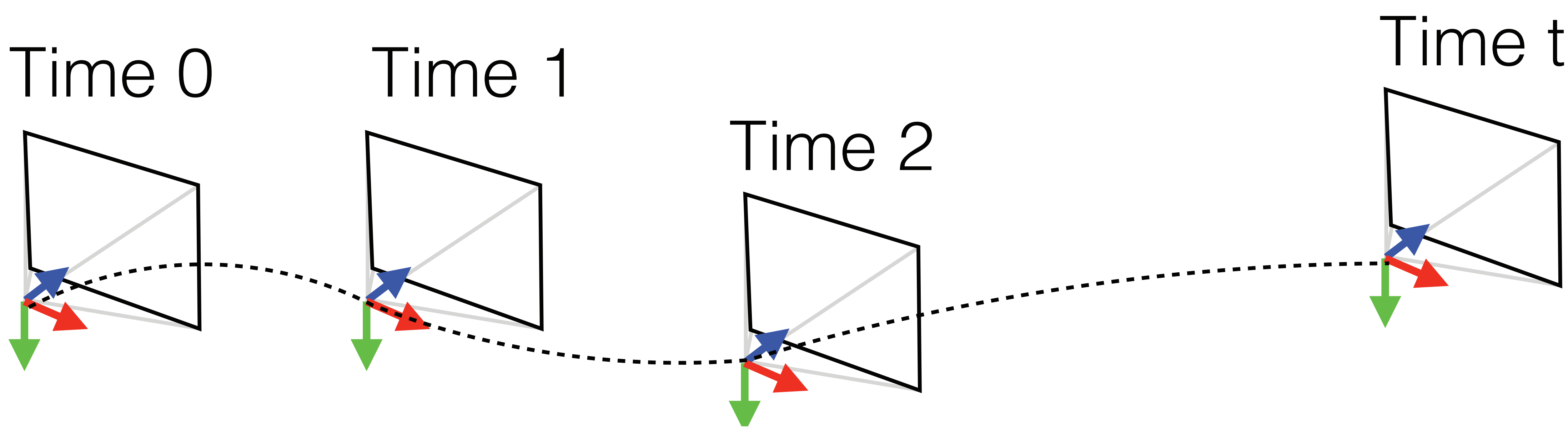
We propose a method to incrementally build ESDFs from Truncated Signed Distance Fields (TSDFs), a common implicit surface representation used in computer graphics and vision. TSDFs are fast to build and smooth out sensor noise over many observations, and are designed to produce surface meshes. Meshes allow human operators to get a better assessment of the robot's environment, and set high-level mission goals.



2017

Multi-view Stereo

From previous lectures: we know how to use SLAM to get a good estimate of the poses of the cameras

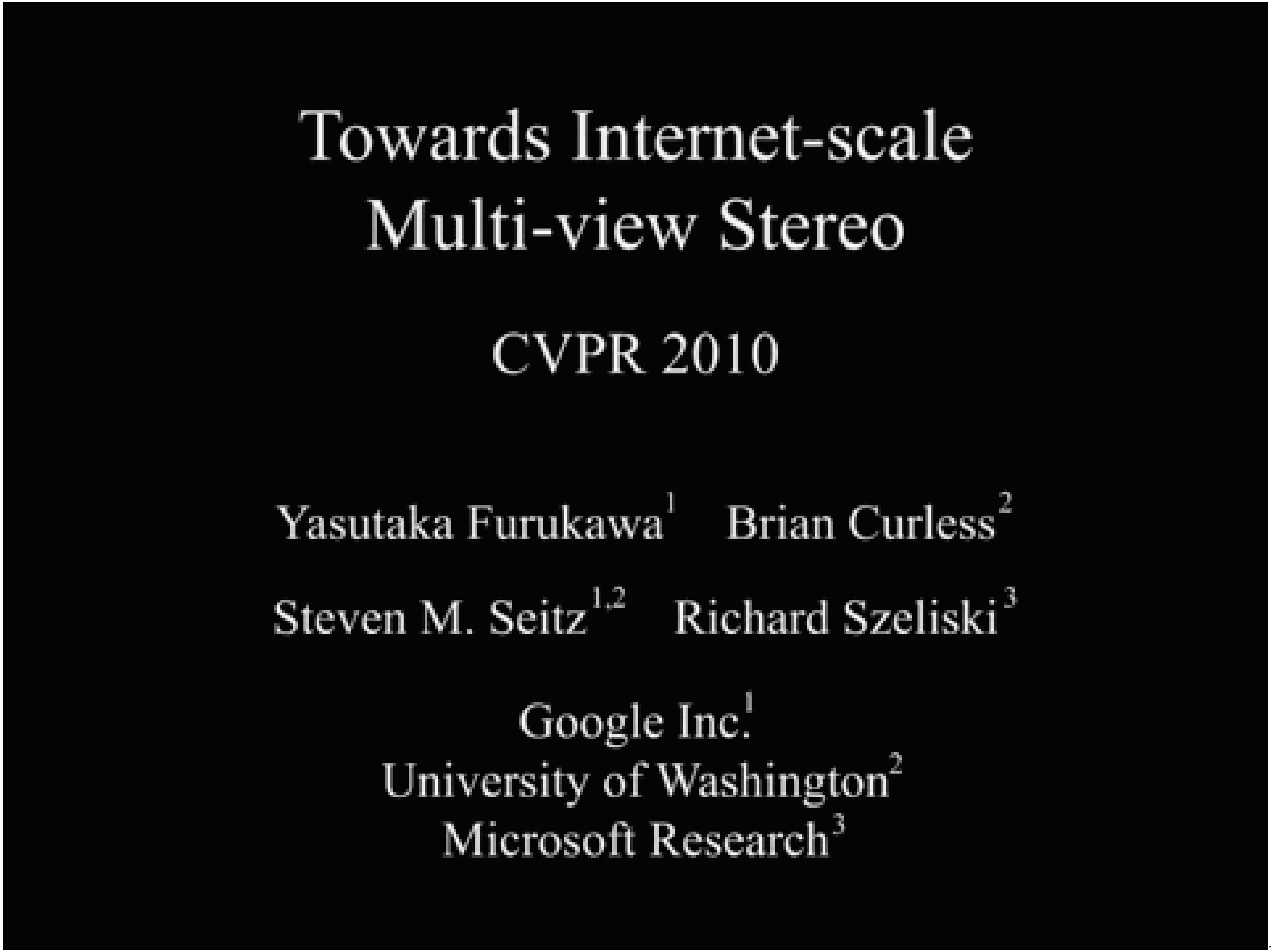


Stereo

Multi-view stereo

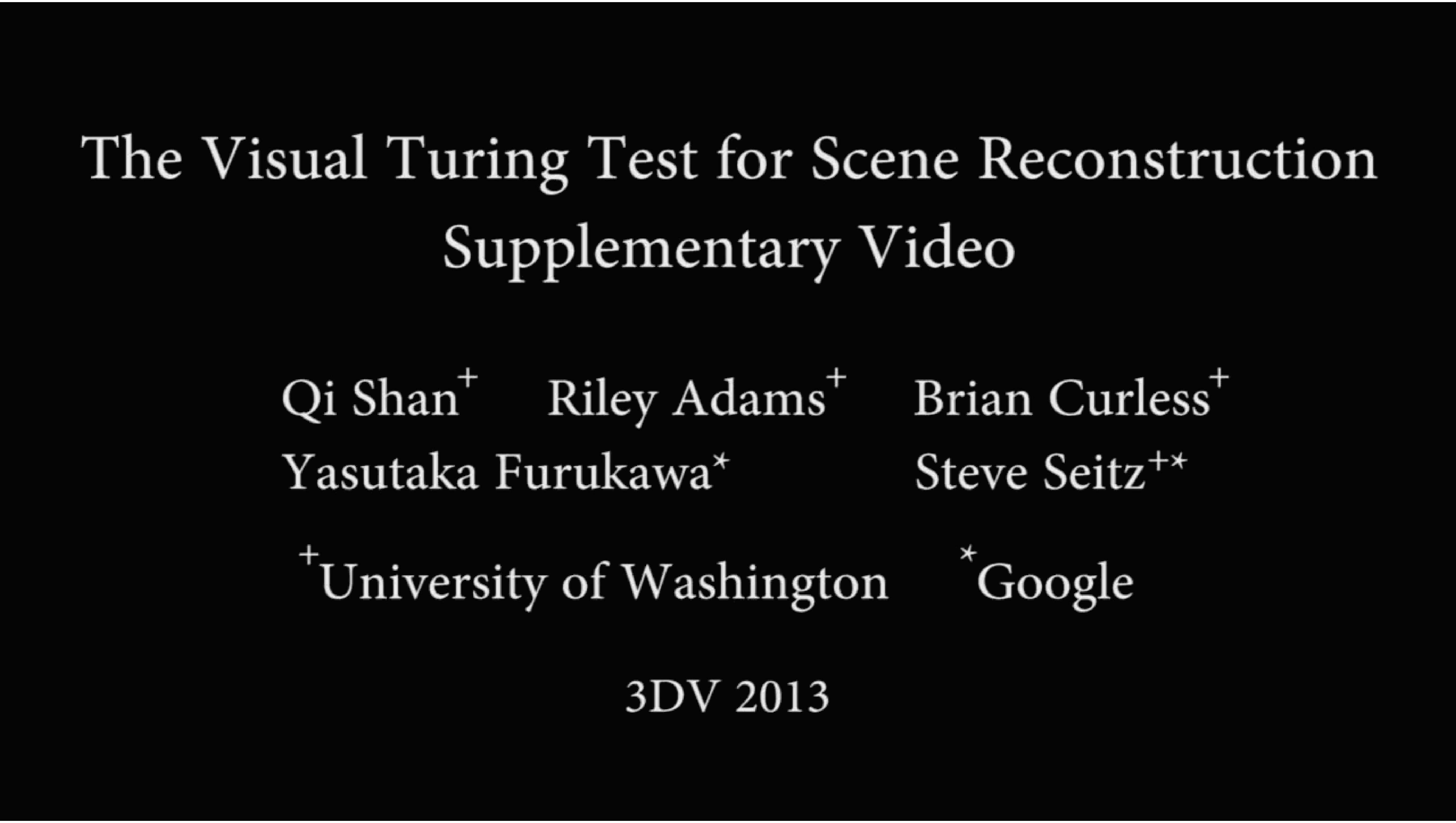
[courtesy of N. Snavely]

Multi-view Stereo



13

Multi-view Stereo



14

Multi-view Stereo

Patch-based methods:

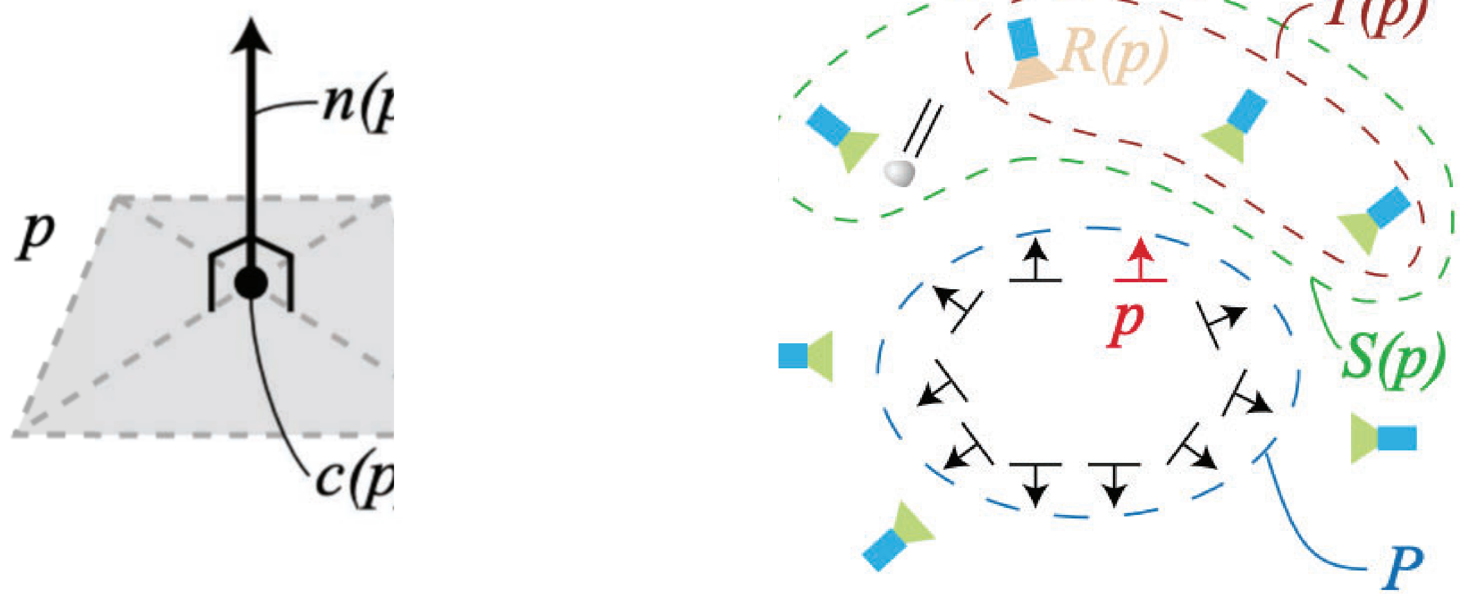


Figure 2. Definition of a patch (left) and of the images associated with it (right). See text for the details.

Estimate normal and center of patch to maximize **photometric consistency**:

Projection
To camera

$$C_{ij}(p) = \rho(I_i(\Omega(\pi_i(p))), I_j(\Omega(\pi_j(p))))$$

Matching Score

Image Intensity

Rectangular Patch

3D point

Example of matching score:

$$1 - \sum_{x,y} |W_1(x,y) - W_2(x,y)|^2$$

[Furukawa and Ponce, "Accurate, Dense, and Robust Multi-View Stereopsis", 2007]

Multi-view Stereo

Enforcing regularity: Markov Random Fields
Find depth k_p of point "p" such that point is photo-consistent and depth changes smoothly..

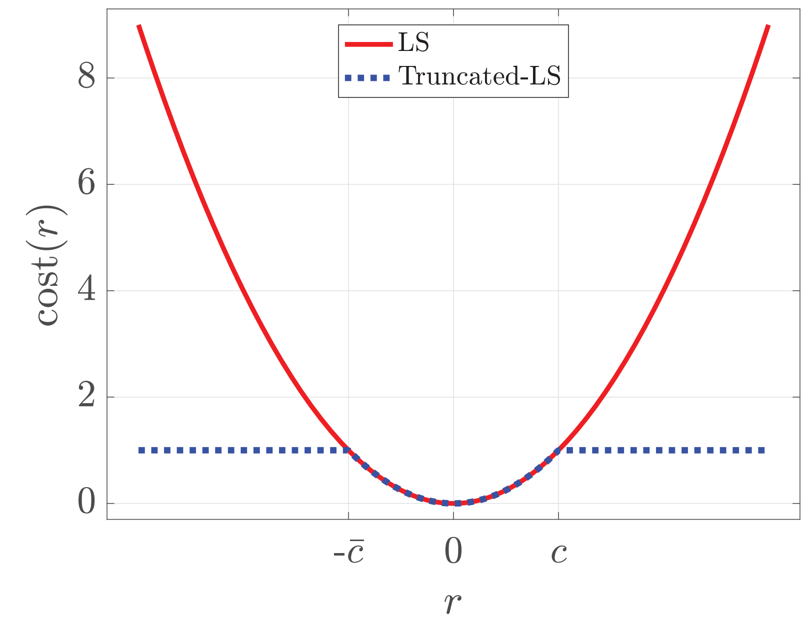
$$E(\{k_p\}) = \sum_p \Phi(k_p) + \sum_{(p,q)\in\mathcal{N}} \Psi(k_p, k_q)$$

Unary potentials
(similar to previous slides)

$$\Phi(k_p = d) = \min(\tau_u, 1 - \mathcal{C}(p, d))$$

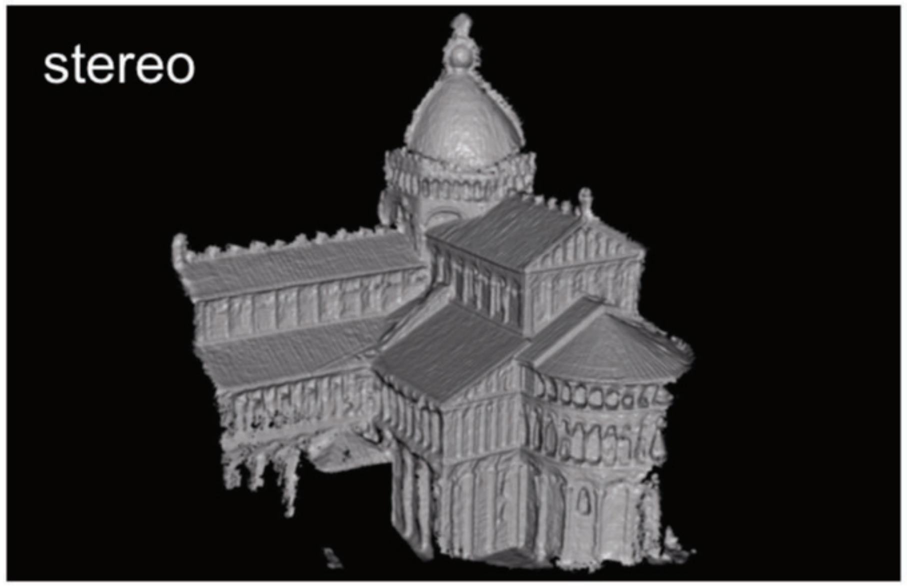
Pairwise potentials

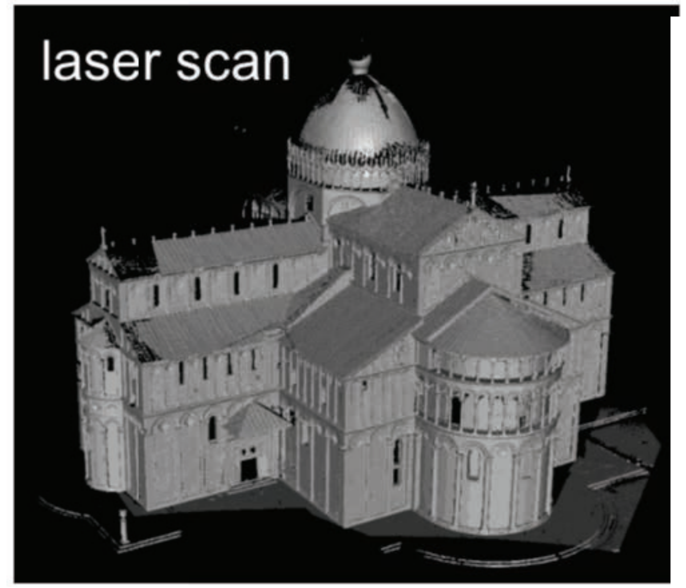
$$\Psi(k_p = d_1, k_q = d_2) = \min(\tau_p, |d_1 - d_2|)$$

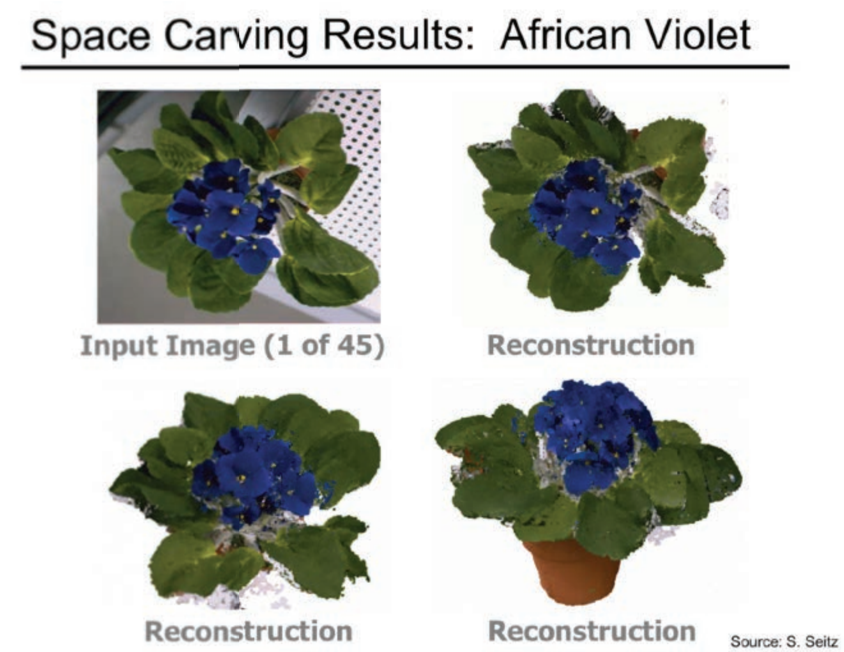


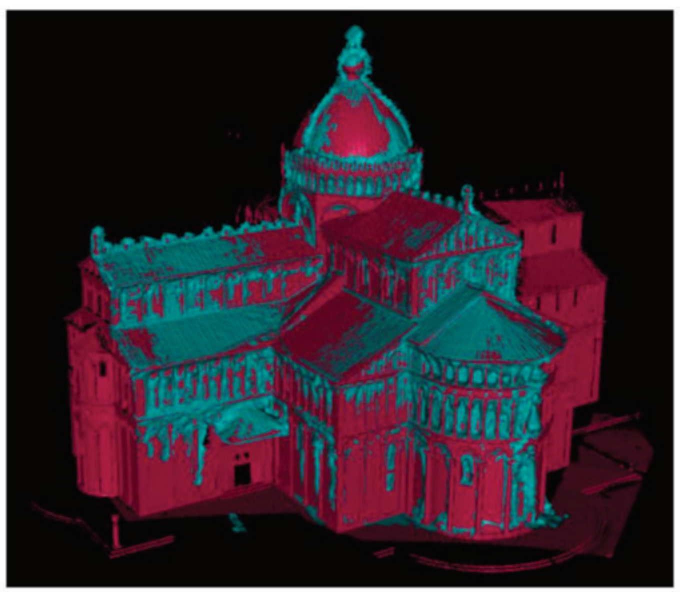
Depth is typically discretized before solving..

How Accurate is Multi-view Stereo?

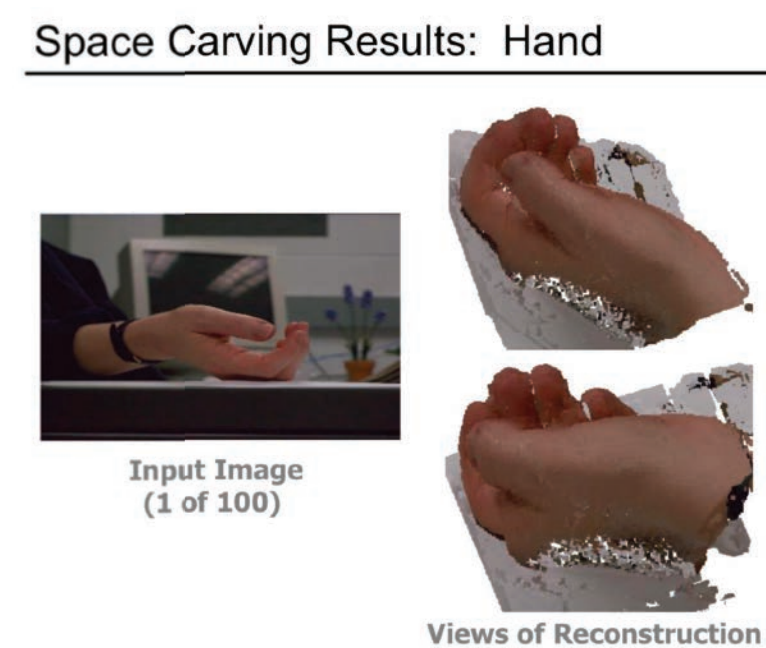








Comparison: 90% of points within 0.128 m of laser scan (building height 51m)



M. Goesele, N. Snavely, B. Curless, H. Hoppe, S. Seitz, Multi-View Stereo for Community Photo Collections, ICCV 2007

Many methods: volumetric stereo, space carving, Shape from silhouettes, carved visual hull

Today

- Dense Reconstruction
  - 3D representations
  - (Some) Multi-view Stereo
  - Depth fusion
- Final thoughts

Multi-View Stereo: A Tutorial

2015

Yasutaka Furukawa
Washington University in St. Louis
furukawa@wustl.edu

Carlos Hernández
Google Inc.
carloshernandez@google.com

KinectFusion: Real-Time Dense Surface Mapping and Tracking*

Richard A. Newcombe, Imperial College London
Shahram Izadi, Microsoft Research
Otmar Hilliges, Microsoft Research
David Molyneaux, Microsoft Research, Lancaster University
David Kim, Microsoft Research, Newcastle University
Andrew J. Davison, Imperial College London
Pushmeet Kohli, Microsoft Research
Jamie Shotton, Microsoft Research
Steve Hodges, Microsoft Research
Andrew Fitzgibbon, Microsoft Research

2011

Figure 1: Example output from our system, generated in real-time with a handheld Kinect depth camera and no other sensing infrastructure. Normal maps (colour) and Phong-shaded renderings (greyscale) from our dense reconstruction system are shown. On the left for comparison is an example of the live, incomplete, and noisy data from the Kinect sensor (used as input to our system).

ElasticFusion: Dense SLAM Without A Pose Graph

Thomas Whelan*, Stefan Leutenegger*, Renato F. Salas-Moreno[†], Ben Glocker[†] and Andrew J. Davison*
*Dyson Robotics Laboratory at Imperial College, Department of Computing, Imperial College London, UK
[†]Department of Computing, Imperial College London, UK
{t.whelan,s.leutenegger,r.salas-moreno10,b.glocker,a.davison}@imperial.ac.uk

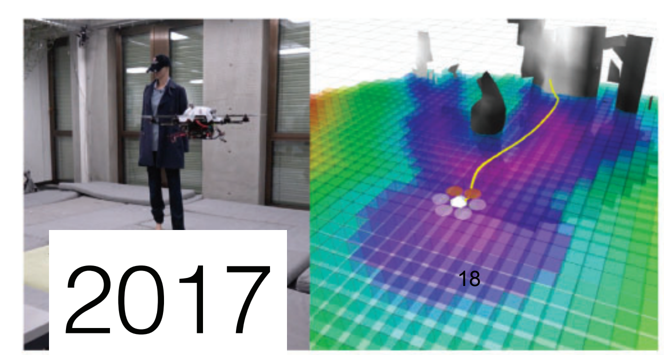
2016

Voxblox: Incremental 3D Euclidean Signed Distance Fields for On-Board MAV Planning

Helen Oleynikova, Zachary Taylor, Marius Fehr, Roland Siegwart, and Juan Nieto
Autonomous Systems Lab, ETH Zürich

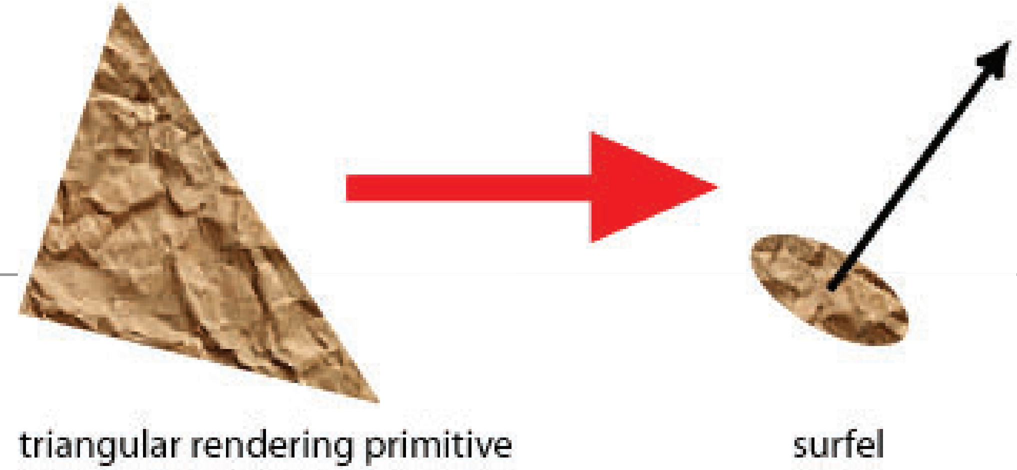
***Abstract*—Micro Aerial Vehicles (MAVs) that operate in unstructured, unexplored environments require fast and flexible local planning, which can replan when new parts of the map are explored. Trajectory optimization methods fulfill these needs, but require obstacle distance information, which can be given by Euclidean Signed Distance Fields (ESDFs).**

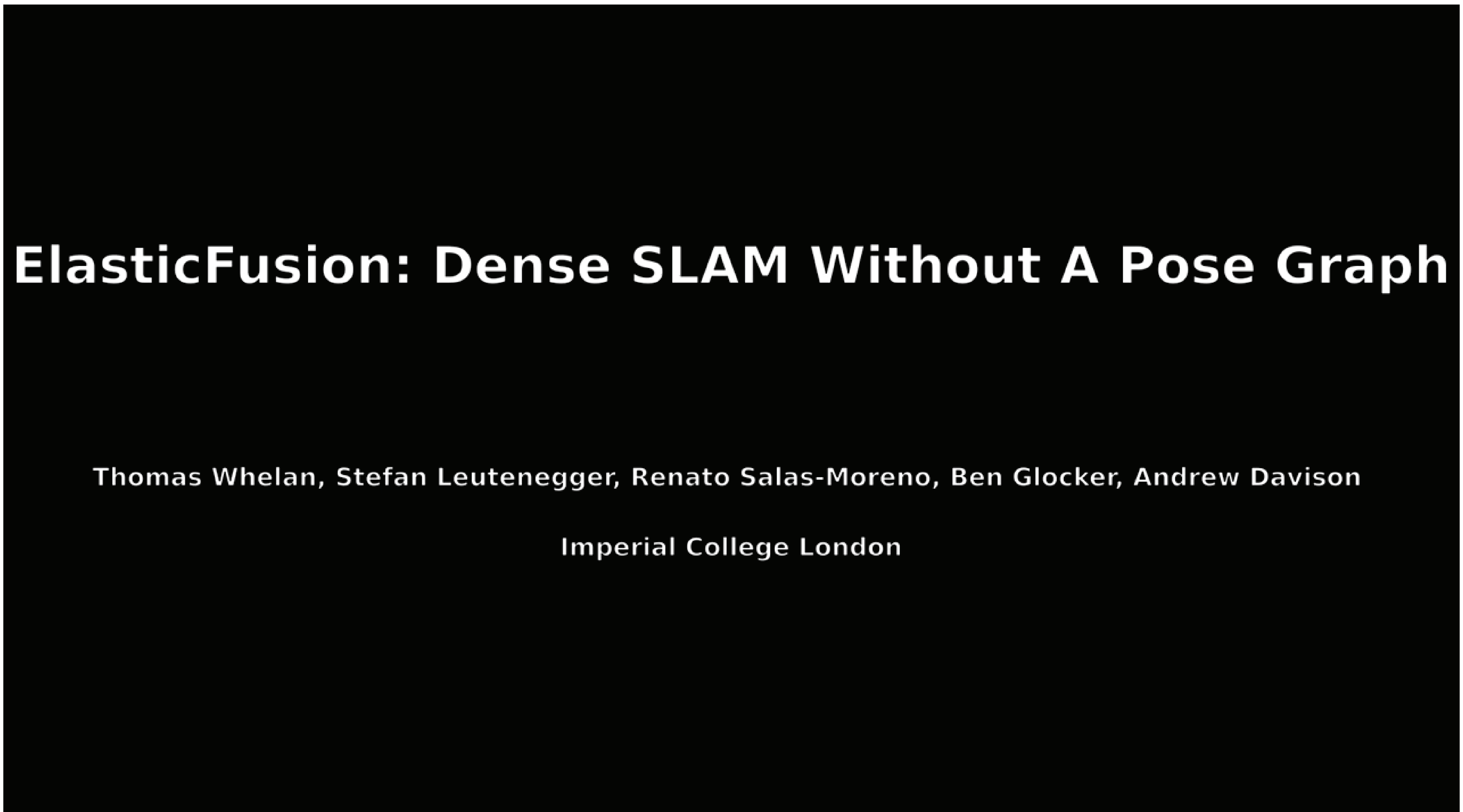
We propose a method to incrementally build ESDFs from Truncated Signed Distance Fields (TSDFs), a common implicit surface representation used in computer graphics and vision. TSDFs are fast to build and smooth out sensor noise over many observations, and are designed to produce surface meshes. Meshes allow human operators to get a better assessment of the robot's environment, and set high-level mission goals.



2017

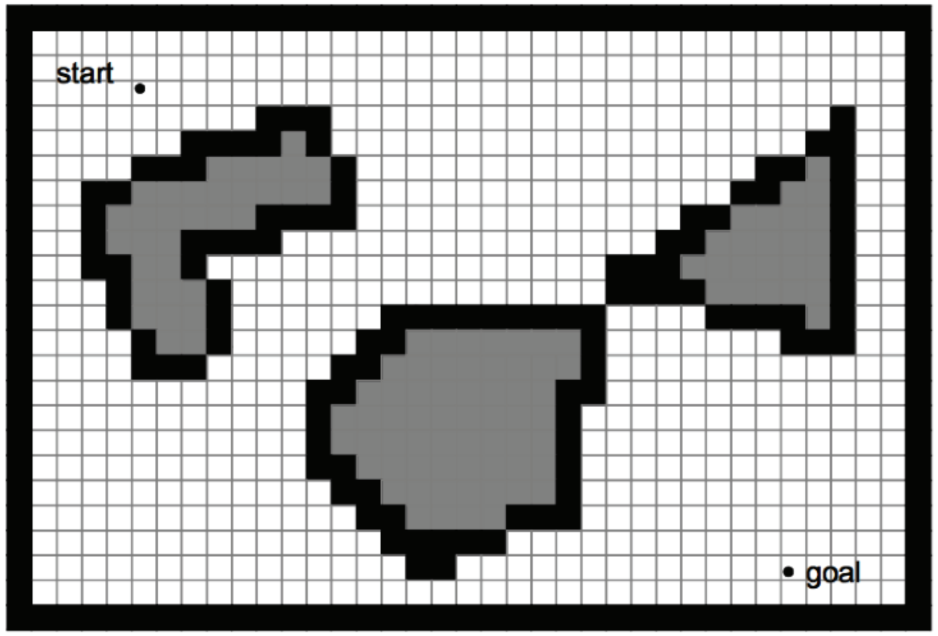
Surfels

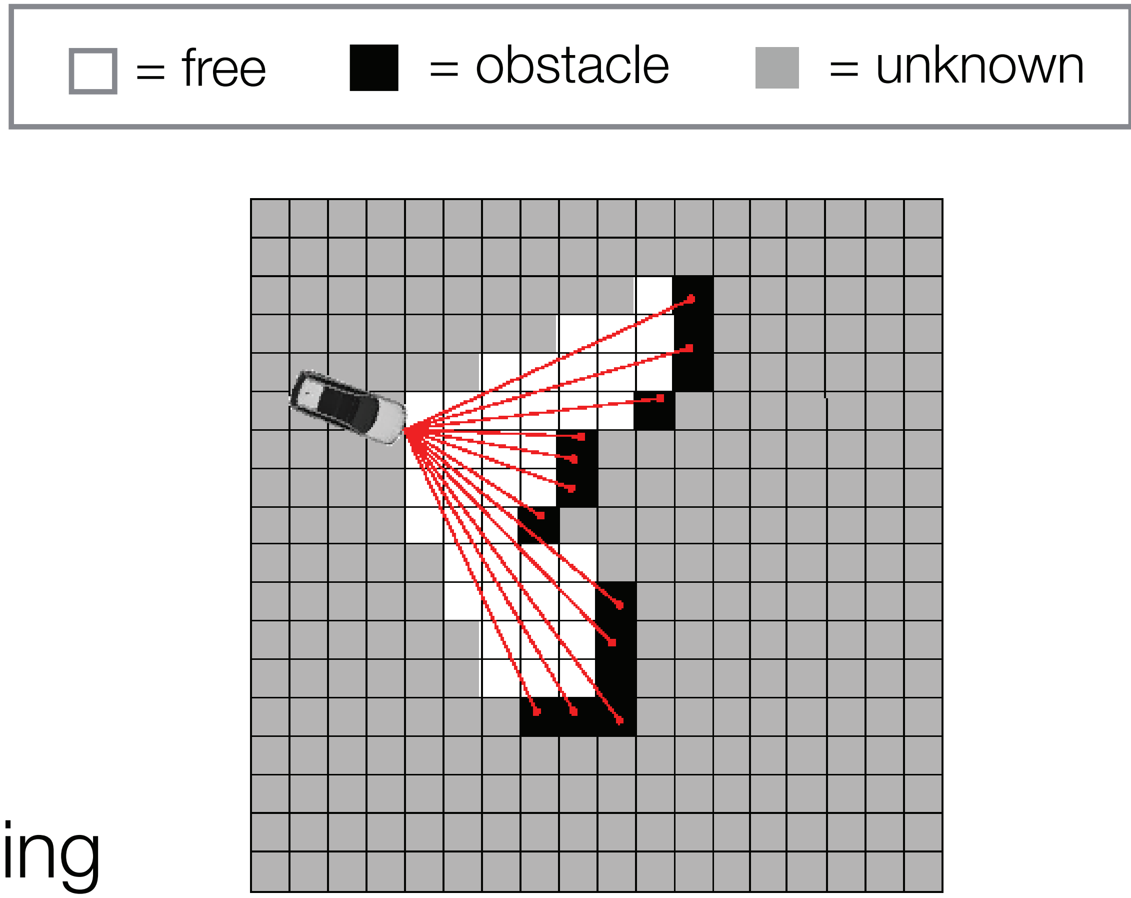




note: still based on RGB-D (contrarily to multi-view stereo)

A Gentle Start: 2D Occupancy Grid Maps





- discretize the environment into cells
- Each cell holds real number [0,1], representing the probability of the cell being occupied

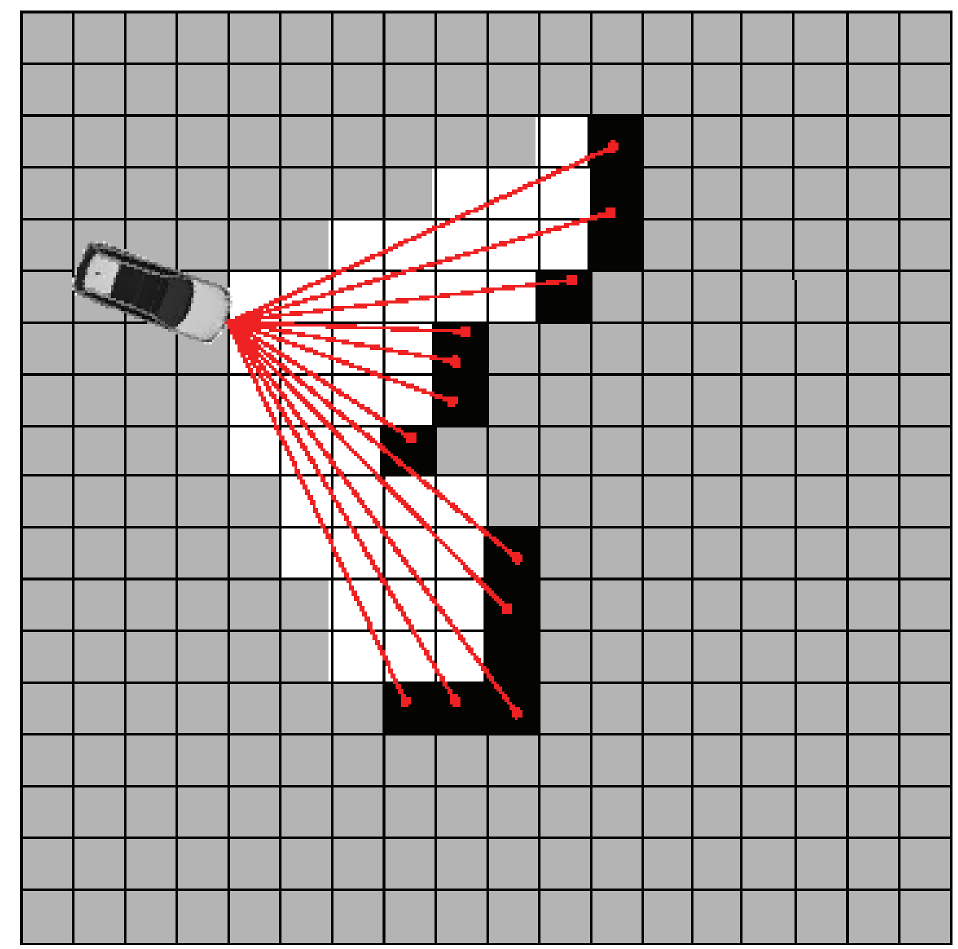
Map posterior $p(\underbrace{m}_{\text{Unknown Map}} \mid \underbrace{z_{1:t}, x_{1:t}}_{\text{Known sensor Depth and robot poses}})$

A Gentle Start: 2D Occupancy Grid Maps

$$p(m \mid z_{1:t}, x_{1:t}) \Rightarrow p(\mathbf{m}_i \mid z_{1:t}, x_{1:t})$$

Probability of cell being occupied

Binary value (free/occupied)



Bayes rule (omitting "x" for simplicity):

$$p(m_i \mid z_{1:t+1}) = \frac{p(z_{t+1} \mid m_i)\, p(m_i \mid z_{1:t})}{p(m_i)}$$

Uninformative Prior

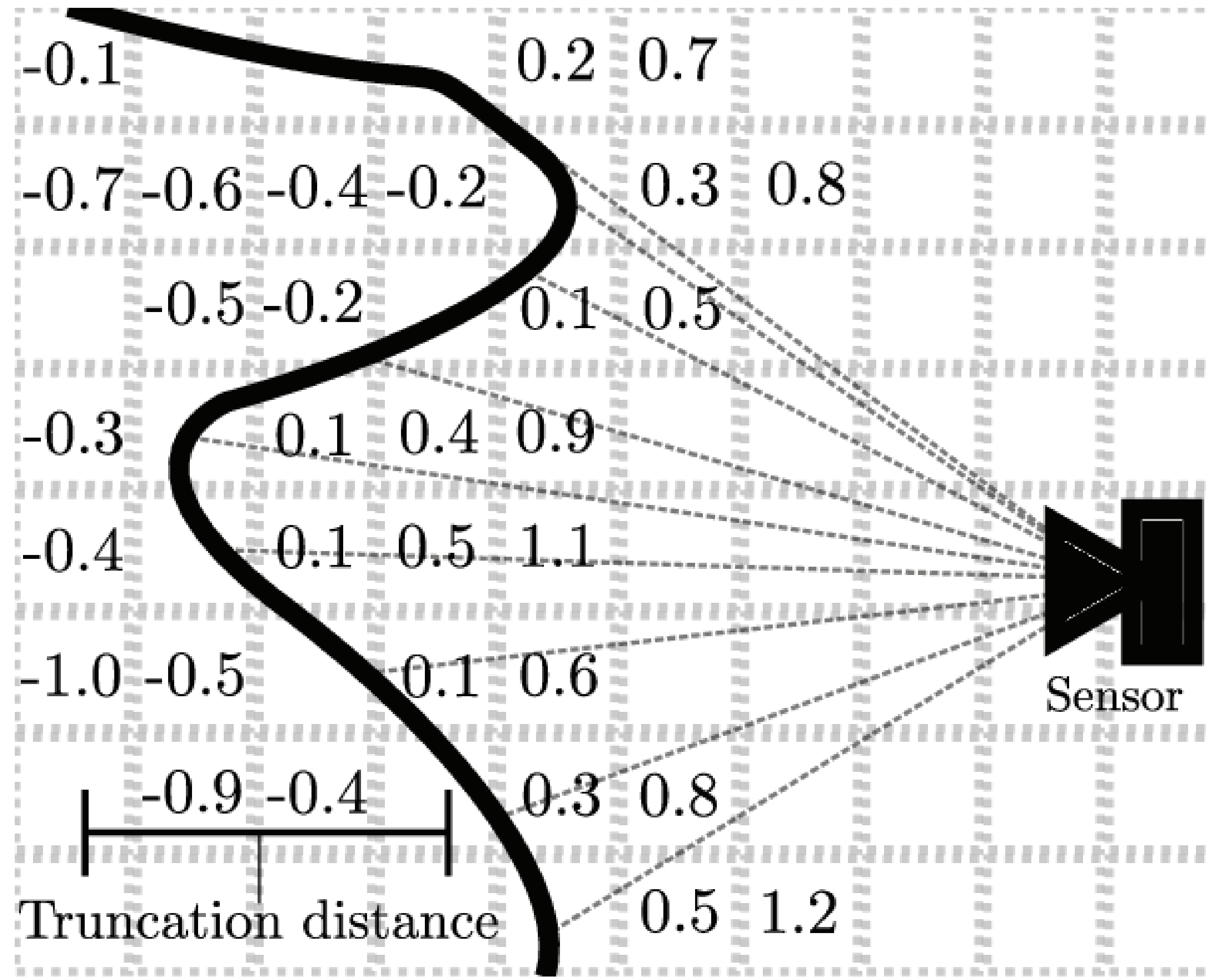
Log-odd representation is typically used to avoid numerical instabilities

$$\frac{p(\mathbf{m}_i \mid z_{1:t}, x_{1:t})}{1 - p(\mathbf{m}_i \mid z_{1:t}, x_{1:t})} \Rightarrow l_{t,i} = \log \frac{p(\mathbf{m}_i \mid z_{1:t}, x_{1:t})}{1 - p(\mathbf{m}_i \mid z_{1:t}, x_{1:t})}$$

Truncated Signed Distance Function (SDF)

- Store distance to nearest obstacle (with sign)
- Only update around obstacle itself

  (implicit surface model)



Update rule:

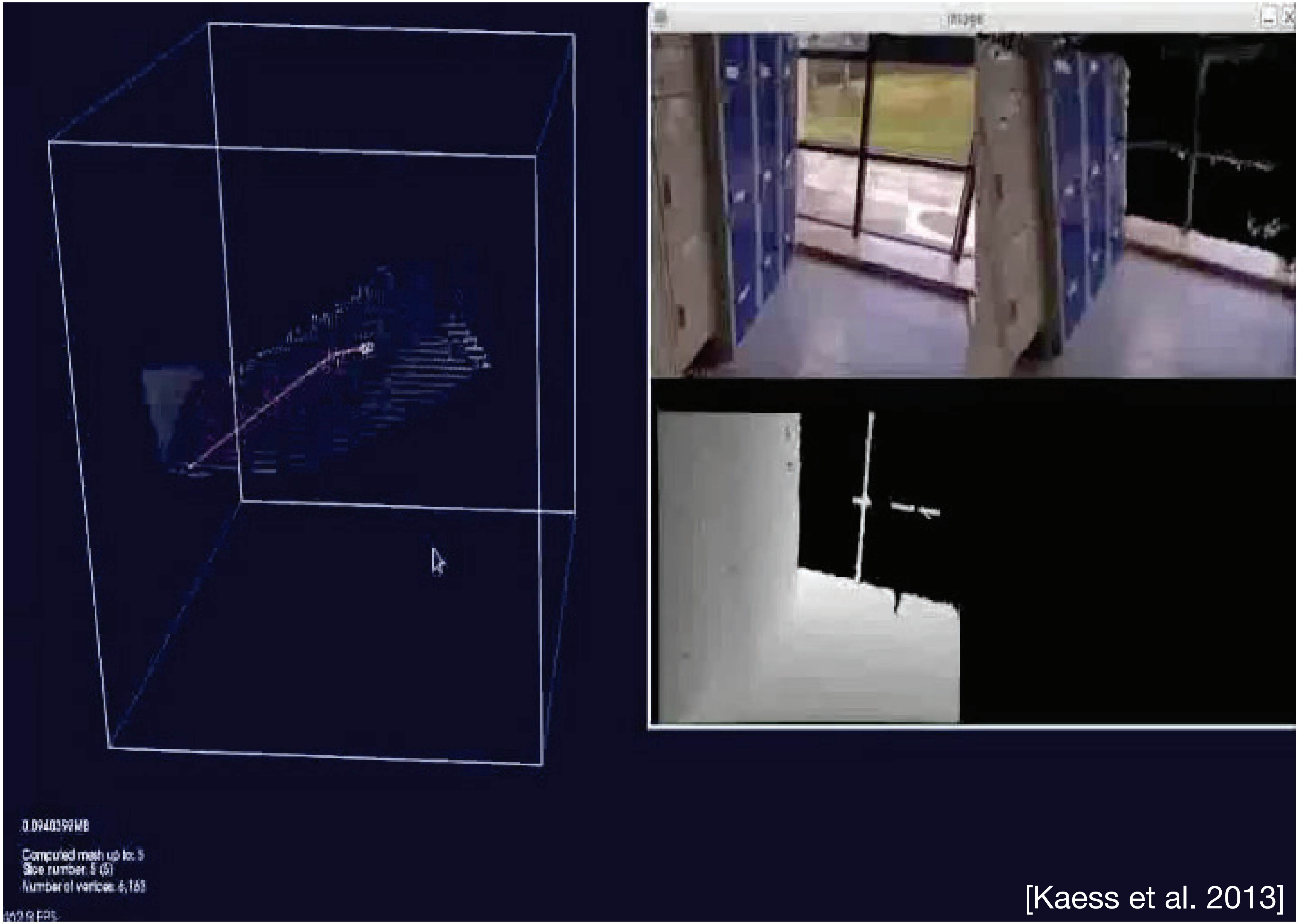
$$
\begin{aligned}
d(\mathbf{x},\mathbf{p},\mathbf{s}) &= \|\mathbf{p}-\mathbf{x}\| \operatorname{sign}\left((\mathbf{p}-\mathbf{x}) \bullet (\mathbf{p}-\mathbf{s})\right) && (1)\\
w_{\text{const}}(\mathbf{x},\mathbf{p}) &= 1 && (2)\\
D_{i+1}(\mathbf{x},\mathbf{p}) &= \frac{W_i(\mathbf{x})D_i(\mathbf{x}) + w(\mathbf{x},\mathbf{p})d(\mathbf{x},\mathbf{p})}{W_i(\mathbf{x}) + w(\mathbf{x},\mathbf{p})} && (3)\\
W_{i+1}(\mathbf{x},\mathbf{p}) &= \min\left(W_i(\mathbf{x}) + w(\mathbf{x},\mathbf{p}), W_{\max}\right) && (4)
\end{aligned}
$$

[Curless and Levoy, “A Volumetric Method for Building Complex Models from Range Images”, 2007]

Kinect Fusion (2011)

GPU, memory …

Kintinuous (2013)



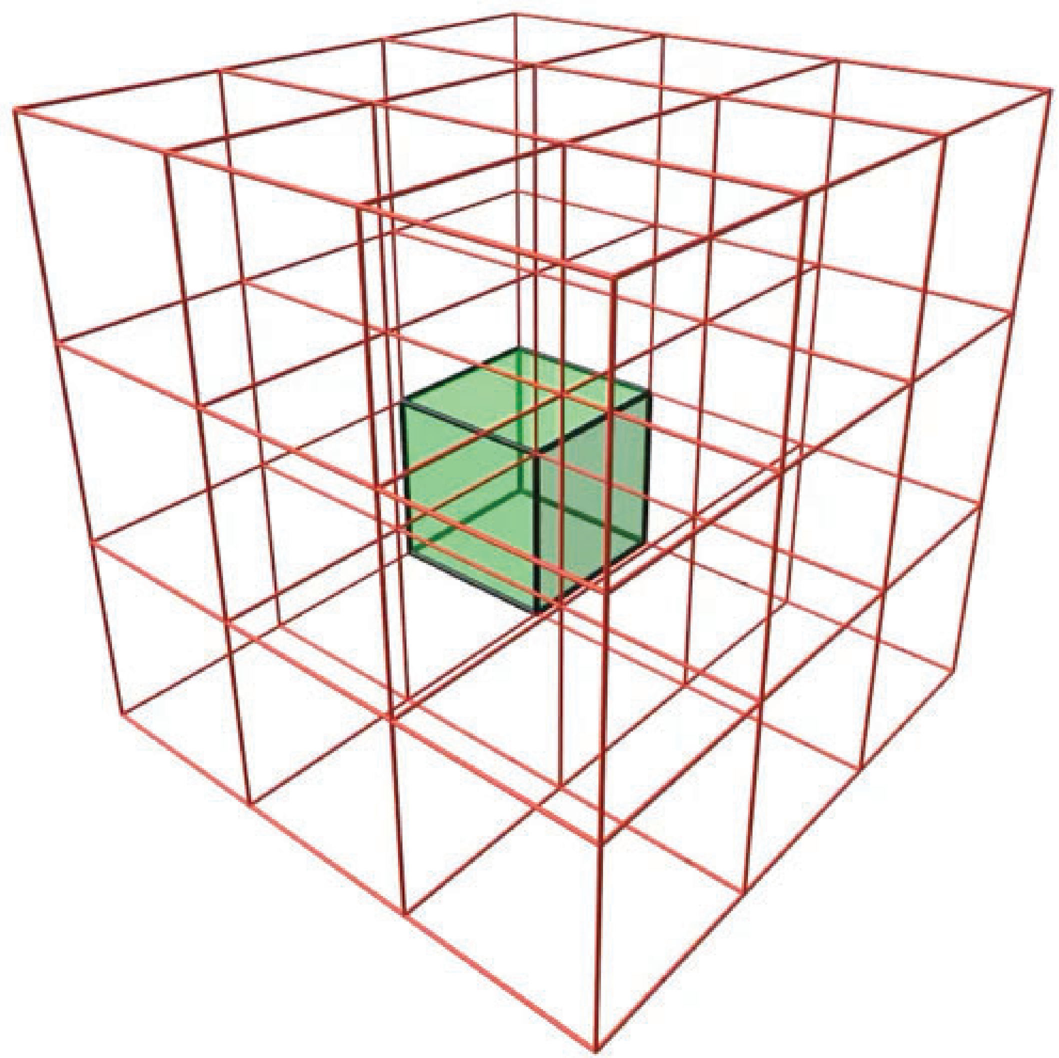
GPU, bounded memory …

VoxBlox (2017)

CPU, memory

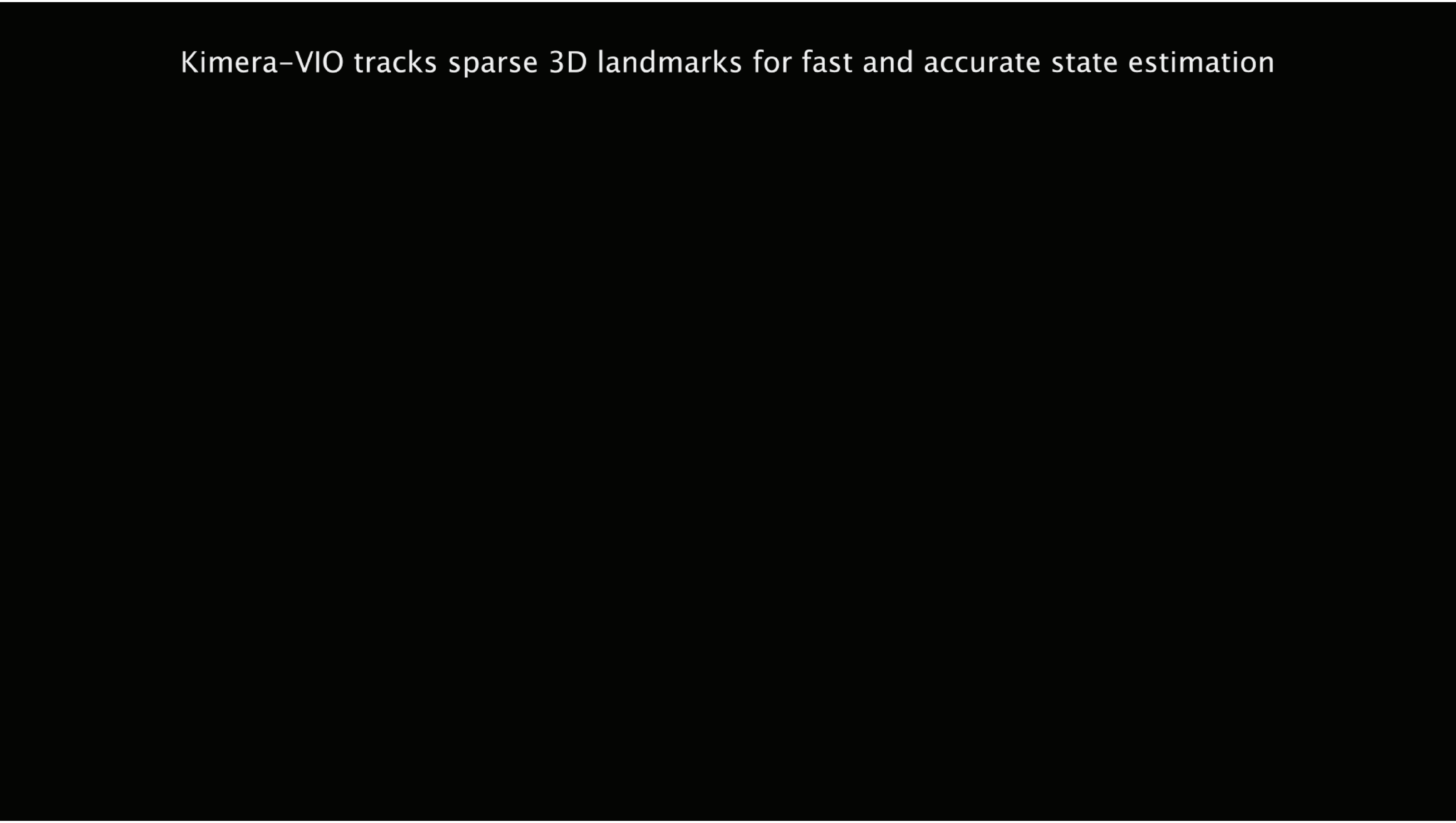
From Voxels to Meshes

Marching cubes



https://www.youtube.com/watch?v=B_xk71YopsA

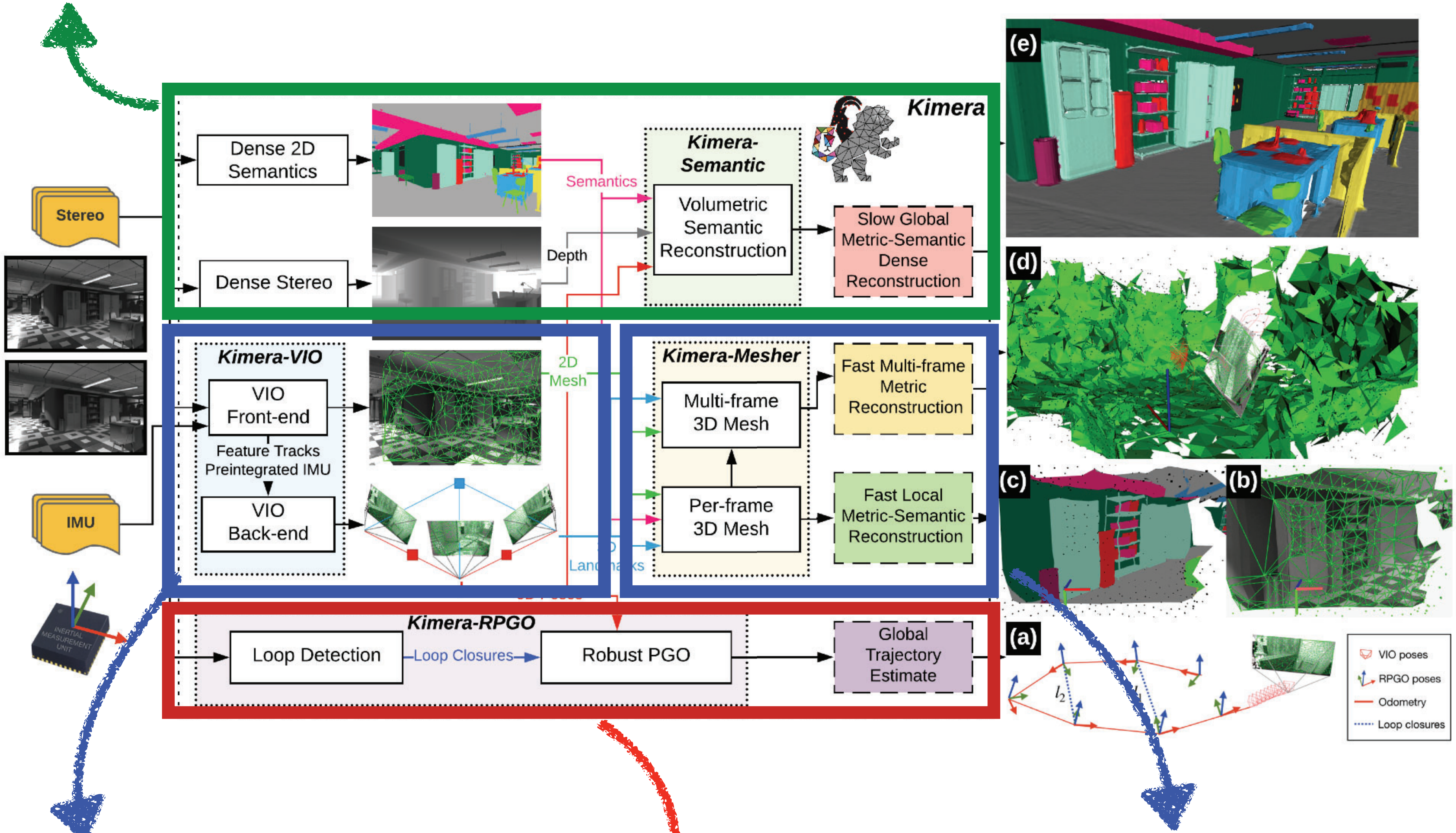
Kimera (2020)



Rosinol, Abate, Chang, Carlone. Kimera: an open-source library for real-time metric-semantic localization and mapping. ICRA, 2020.

Metric-semantic 3D Reconstruction

Kimera-Semantics



Kimera-VIO

Kimera-Mesher

Kimera-RPGO

Today

- Dense Reconstruction
  - 3D representations
  - (Some) Multi-view Stereo
  - Depth fusion
- Final thoughts

KinectFusion: Real-Time Dense Surface Mapping and Tracking*

Richard A. Newcombe, Imperial College London; Shahram Izadi, Microsoft Research; Otmar Hilliges, Microsoft Research; David Molyneaux, Microsoft Research, Lancaster University; David Kim, Microsoft Research, Newcastle University; Andrew J. Davison, Imperial College London; Pushmeet Kohli, Microsoft Research; Jamie Shotton, Microsoft Research; Steve Hodges, Microsoft Research; Andrew Fitzgibbon, Microsoft Research

2011

Figure 1: Example output from our system, generated in real-time with a handheld Kinect depth camera and no other sensing infrastructure. Normal maps (colour) and Phong-shaded renderings (greyscale) from our dense reconstruction system are shown. On the left for comparison is an example of the live, incomplete, and noisy data from the Kinect sensor (used as input to our system).

Multi-View Stereo: A Tutorial

2015

Yasutaka Furukawa
Washington University in St. Louis
furukawa@wustl.edu

Carlos Hernández
Google Inc.
carloshernandez@google.com

ElasticFusion: Dense SLAM Without A Pose Graph

Thomas Whelan*, Stefan Leutenegger*, Renato F. Salas-Moreno†, Ben Glocker† and Andrew J. Davison*
*Dyson Robotics Laboratory at Imperial College, Department of Computing, Imperial College London, UK
†Department of Computing, Imperial College London, UK
{t.whelan,s.leutenegger,r.salas-moreno10,b.glocker,a.davison}@imperial.ac.uk

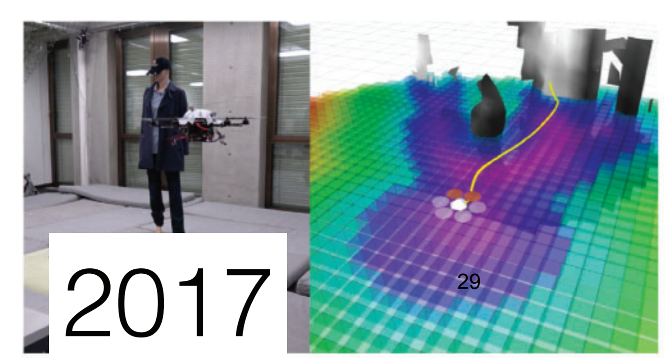
2016

Voxblox: Incremental 3D Euclidean Signed Distance Fields for On-Board MAV Planning

Helen Oleynikova, Zachary Taylor, Marius Fehr, Roland Siegwart, and Juan Nieto
Autonomous Systems Lab, ETH Zürich

Abstract—Micro Aerial Vehicles (MAVs) that operate in unstructured, unexplored environments require fast and flexible local planning, which can replan when new parts of the map are explored. Trajectory optimization methods fulfill these needs, but require obstacle distance information, which can be given by Euclidean Signed Distance Fields (ESDFs).

We propose a method to incrementally build ESDFs from Truncated Signed Distance Fields (TSDFs), a common implicit surface representation used in computer graphics and vision. TSDFs are fast to build and smooth out sensor noise over many observations, and are designed to produce surface meshes. Meshes allow human operators to get a better assessment of the robot's environment, and set high-level mission goals.

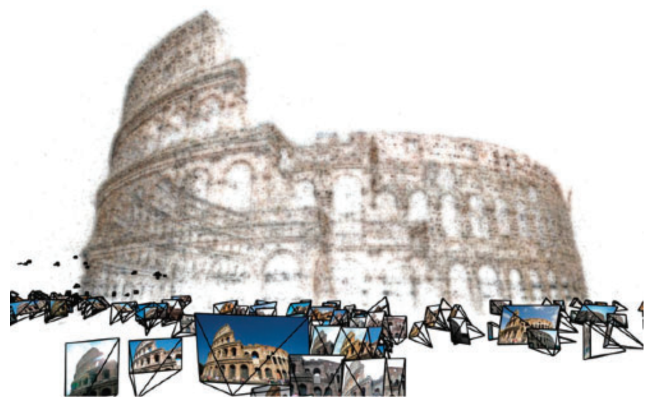


2017

Robot Perception or Computer Vision?

Computer vision

.. "a day on a cluster with 500 compute cores"



Robotics

50-100ms latency, embedded, incremental

No longer a dichotomy for many vision applications!

Robot Perception or Computer Vision?

Unordered
Vs
Sequential