Imperial College London

SMART ROBOTICS LAB

18/01/2020

Lecture 1:
Introduction, Problem Formulation, and Examples

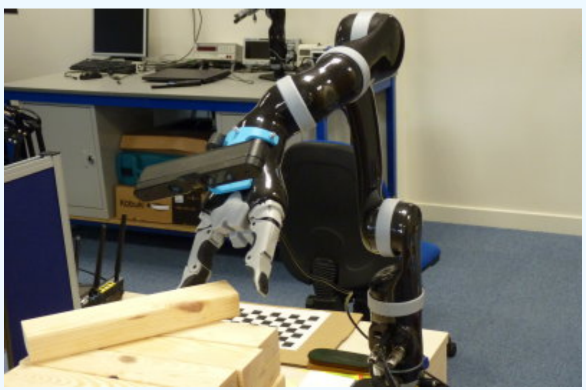
70003 Advanced Robotics

Dr Stefan Leutenegger

Teaching Assistants: Dr Marija Popovic,
Sotiris Papatheodorou, Binbin Xu, and Nils Funk

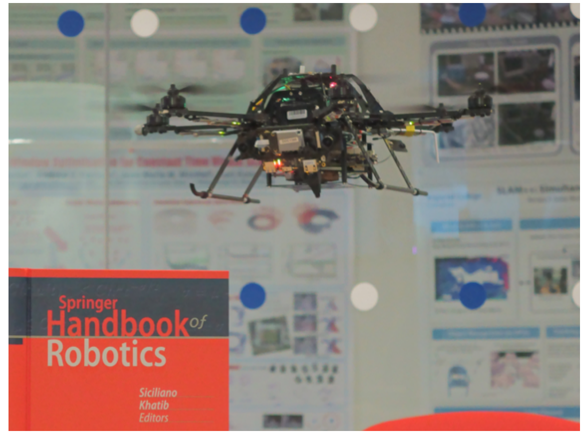
Applications

Domestic robotics
(picture: Dyson 360 eye)



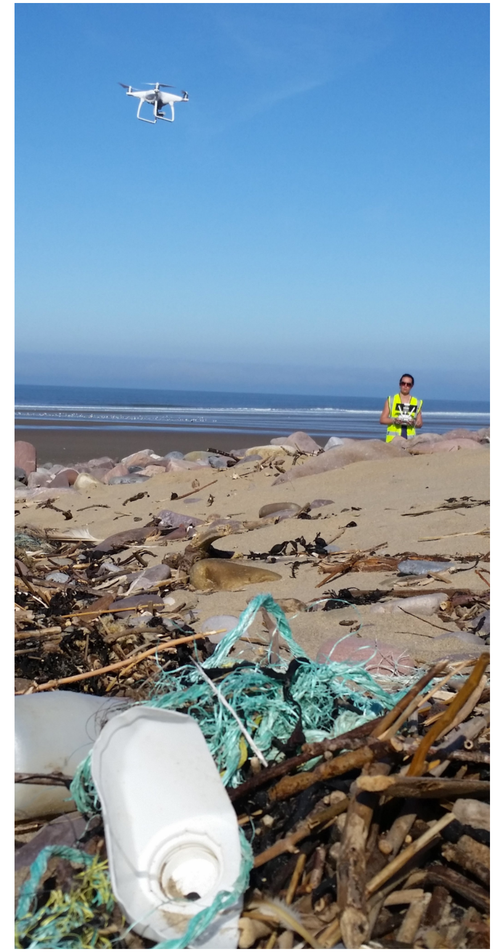
Advanced manipulation

dyson Robotics Lab — S. Leutenegger & A. Davison



Drone inspection and construction

Self-driving cars
(picture: Waymo self-driving car)



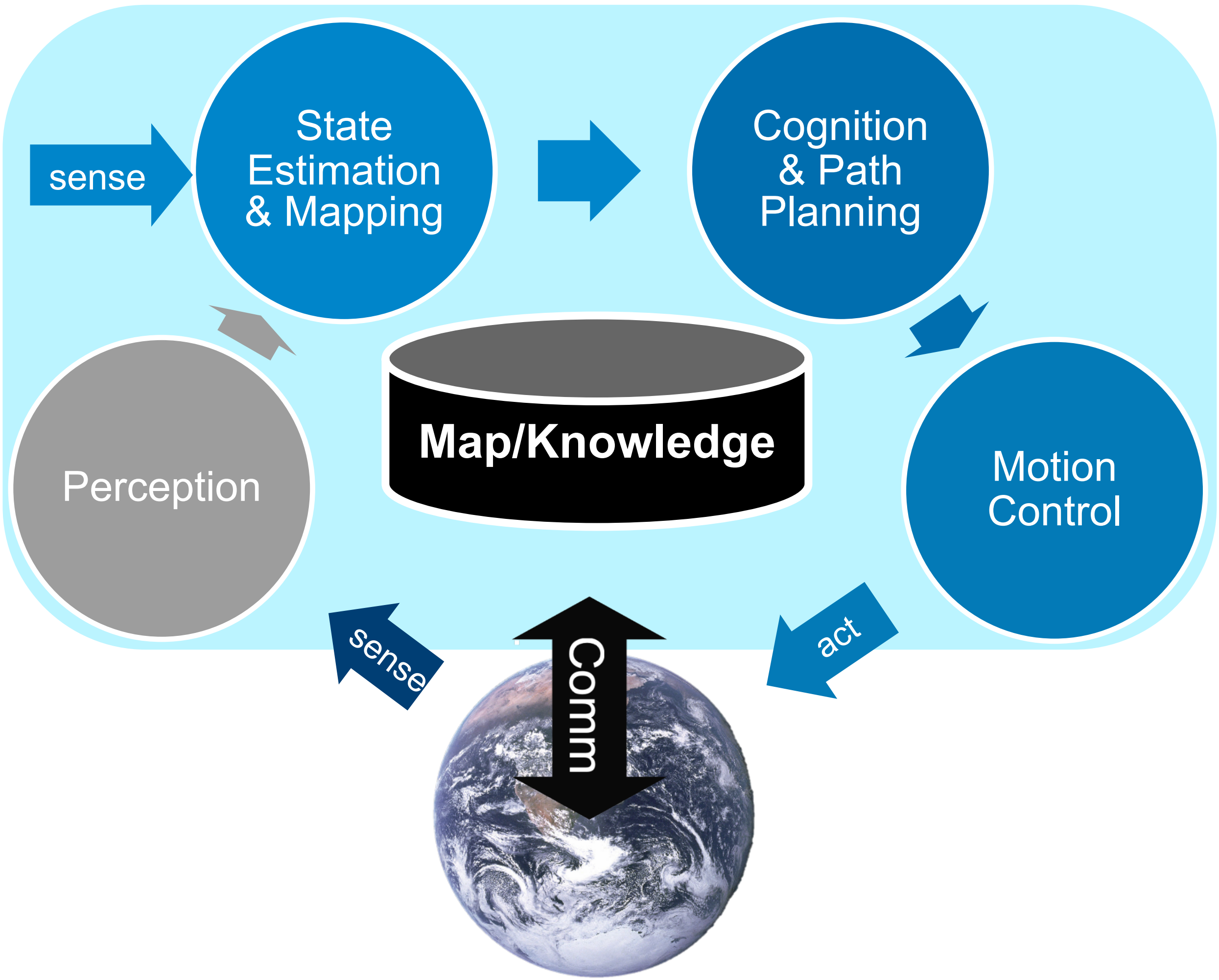
Environmental monitoring

SMART ROBOTICS LAB

Mobile Robotics Questions

- Where am I?
- What does the world look like?
- Where am I going?
- How do I get there?

Mobile Robotics Challenges



Requirements on algorithms:
- Robust
- Efficient
- Accurate

Visual-Inertial SLAM & Model-Predictive Control

Joint work with Dimos Tzoumanikas, Wenbin Li, and Marius Grimm

Robotics at The Department of Computing

- **Smart Robotics Lab**
  Dr Stefan Leutenegger
  http://wp.doc.ic.ac.uk/sleutene/

- **Dyson Robotics Lab**
  Prof. Andrew Davison
  http://www.imperial.ac.uk/dyson-robotics-lab

- **Robot Learning Lab**
  Dr Ed Johns
  http://www.imperial.ac.uk/robot-learning/

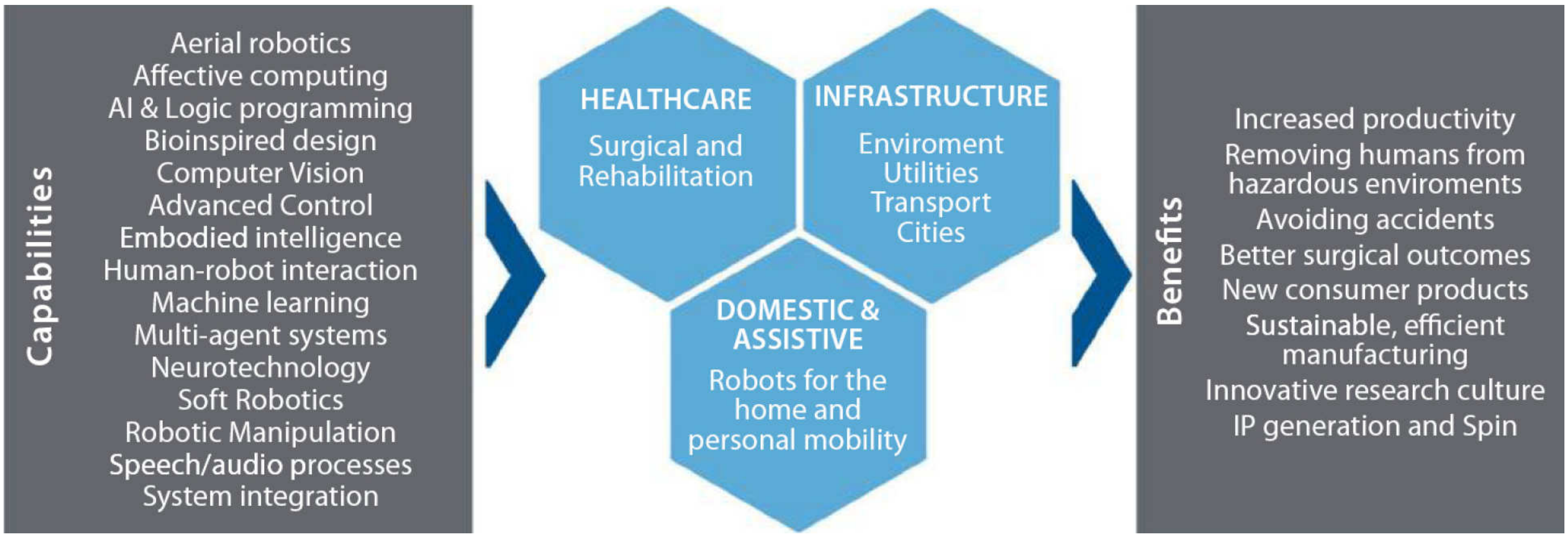
- **Robots and AI**
  Dr Antoine Cully
  http://www.imperial.ac.uk/adaptive-intelligent-robotics

Robotics Forum at Imperial College

A college-wide network, see www.imperial.ac.uk/robotics

44 PIs | 100+ post-docs & research fellows | 200+ PhD and EngD students

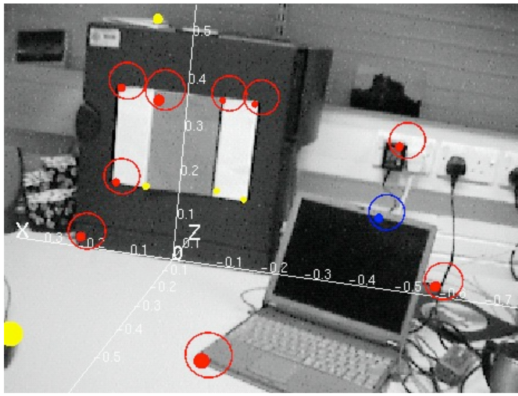
Adaptive and Intelligent Robotics Lab | Aerial Robotics Lab | Biomechanics Lab | Biomechatronics Lab | Brain and Behaviour Lab | Centre for Neurotechnology | Centre for Process Systems Engineering | Circuits and Systems | Cognitive Robotics | Computer vision and learning lab | Control and Power | Department of Aeronautics | Design Psychology | Dyson Robotics Lab | Dyson School of Design Engineering | Geometric Modelling and Manufacturing | The Hamlyn Centre | HARMS Lab | Human Insights Group | Human Robotics Group | Intelligent Behaviour Understanding Group | Intelligent Digital Systems (iDSL) | KKH Research Group | Kornyshev's Theoretical Chemical Physics Group | Krapp Lab | Mechatronics in Medicine | Morphological Computation and Learning Laboratory | Neuromechanics Lab | Personal Robotics Lab | Photonics Group | REDS Lab | Robot Intelligence Lab | Robot Learning Lab | Robot Vision Group | Sensory Neuroengineering | SiMMS - Simulation and Modelling in Medicine and Surgery | **Smart Robotics Lab** | Speech and Audio Processing Group | Statistical Machine Learning | Transport Systems and Logistics Laboratory | Two Dimensional | Materials and Wearable Bioelectronics | Verification of Autonomous Systems Group



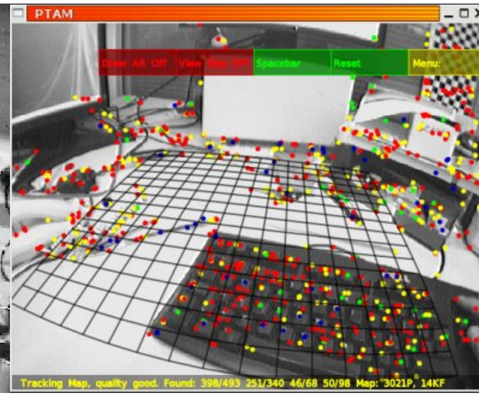
You can sign up to the robotics mailing list (seminar announcements etc.)

History of Visual Localisation and Mapping

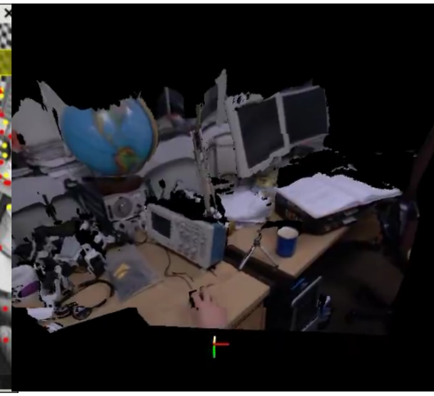
SLAM: Simultaneous Localisation And Mapping



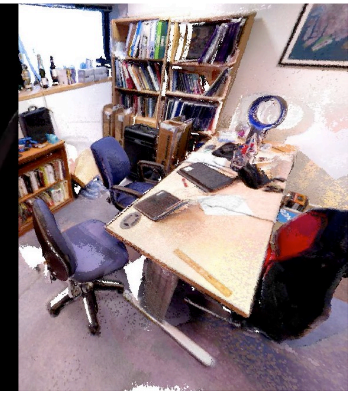
MonoSLAM
2003 [1]



PTAM
2007 [2]



DTAM
2011 [3]



ElasticFusion
2015 [4]

[1] Davison, A. J., et al.: MonoSLAM: Real-time single camera SLAM. In: Pattern Analysis and Machine Intelligence, IEEE Transactions on 29.6 (2007): 1052-1067.

[2] Klein, G. & Murray, D.: Parallel tracking and mapping for small AR workspaces. In: Proc Intl. Symposium on Mixed and Augmented Reality (ISMAR 2007), Nara (November 2007)

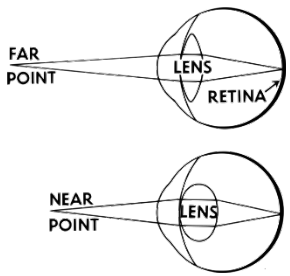
[3] Newcombe, R. A., Lovegrove, S. J., & Davison, A. J. (2011, November). DTAM: Dense tracking and mapping in real-time. In Computer Vision (ICCV), 2011 IEEE International Conference on (pp. 2320-2327). IEEE

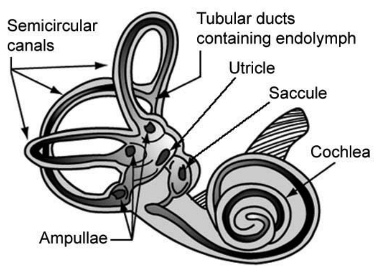
[4] Whelan, T., Leutenegger, S., Salas-Moreno, R. F., Glocker, B., & Davison, A. J. ElasticFusion: Dense SLAM Without A Pose Graph.

Why Multi-Sensor Fusion?

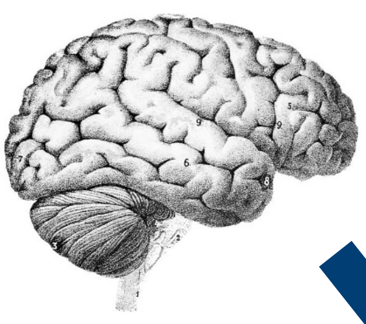
How humans do it

Sensing





Processing



How?
Who knows…

How robots do it

Bumblebee
Stereo Camera

ADIS166448
Inertial Measurement Unit (IMU)

Using maths and code

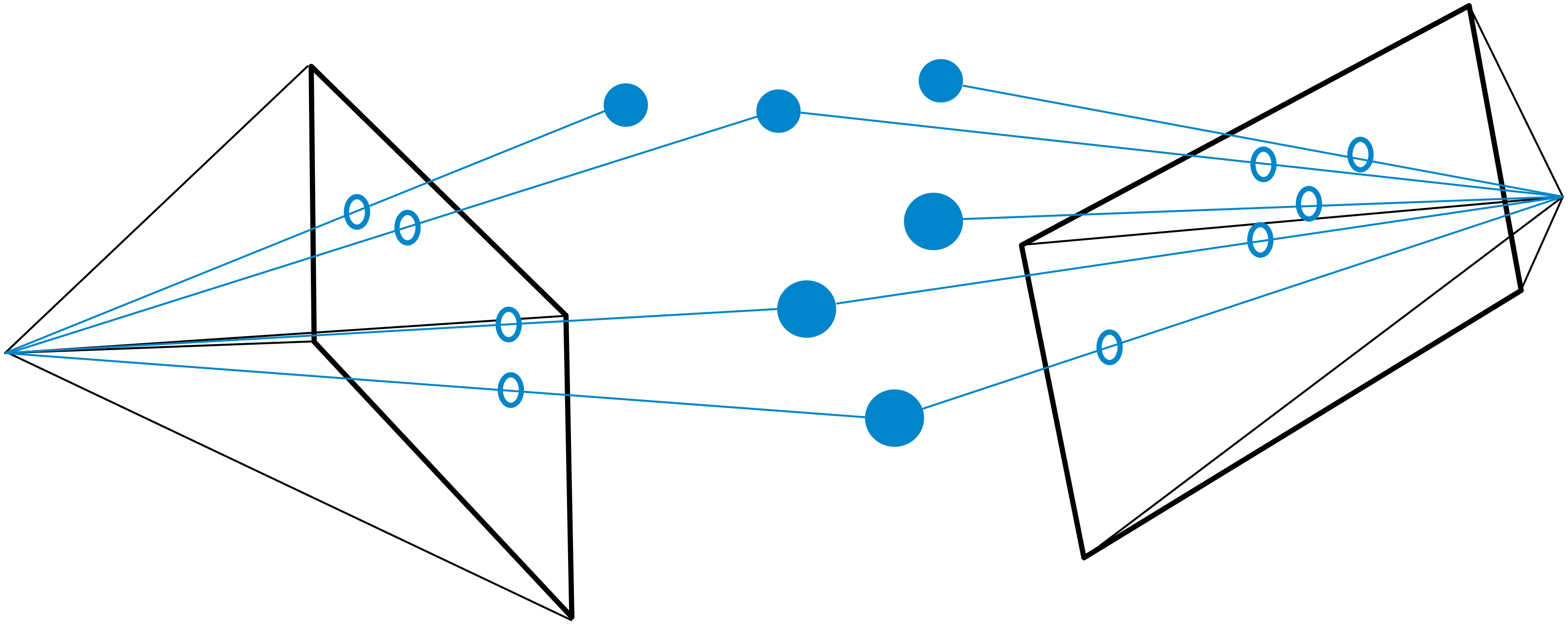
$$\mathbf{b}^*_{\lambda_1} = \underbrace{\mathbf{b}_{\lambda_1,0} - \mathbf{H}_{\lambda_1\mu}\mathbf{H}^{-1}_{\mu\mu}\mathbf{b}_{\mu,0}}_{\mathbf{b}^*_{\lambda_1,0}} - \mathbf{H}^*_{\lambda_1\lambda_1}\Delta\boldsymbol{\chi}_{\lambda_1}.$$

```
bool MarginalizationError::marginalizeOut(
    const std::vector<uint64_t>& parameterBlockIds,
    const std::vector<bool> & keepParameterBlocks)
{
  if (parameterBlockIds.size() == 0){
    return false;
  }

  // copy so we can manipulate
  std::vector<uint64_t> parameterBlockIdsCopy = parameterBlockIds;
```

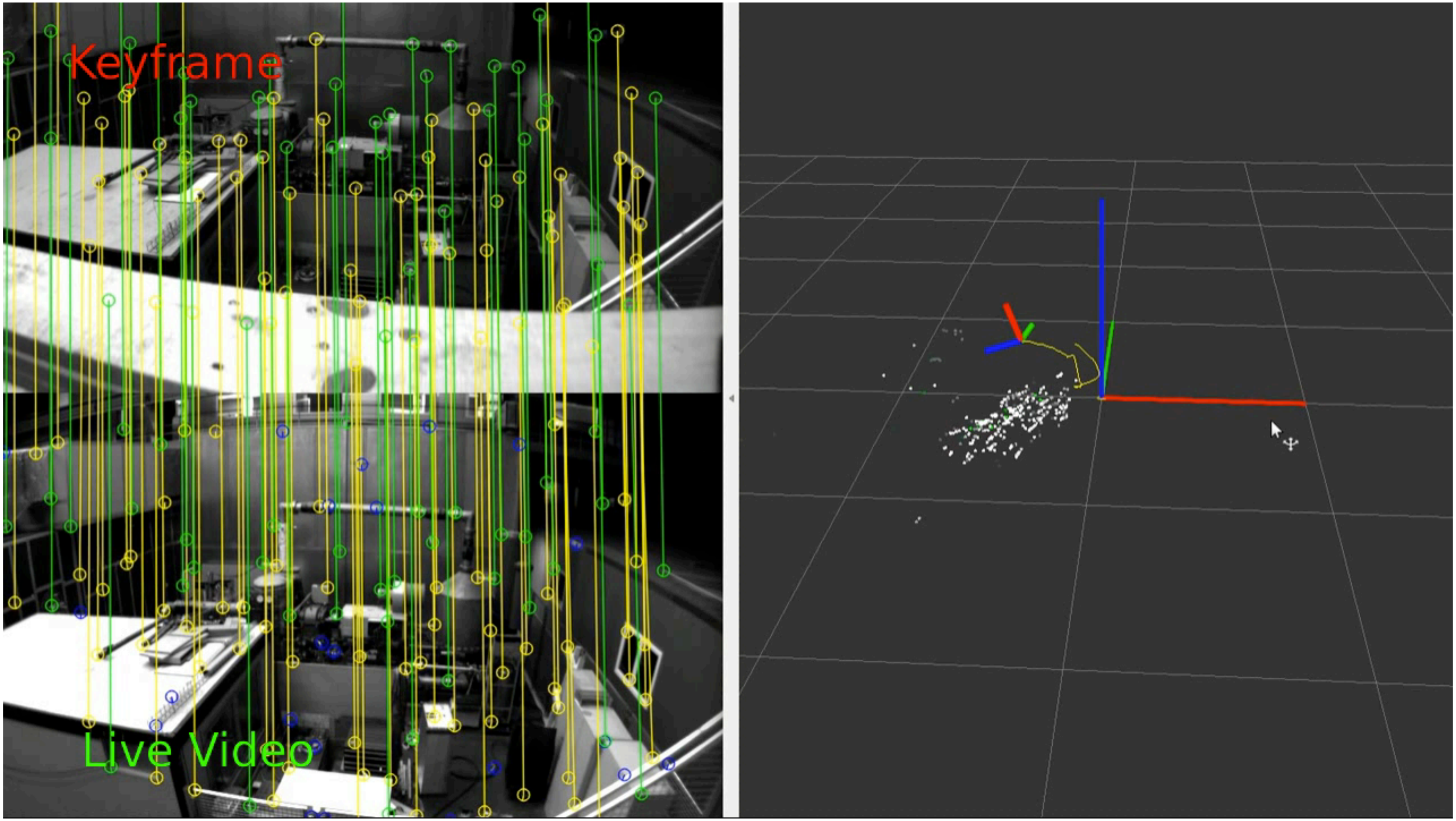
The Visual SLAM Problem

SLAM: Simultaneous Localisation and Mapping



OKVIS: Open Keyframe-based Visual-Inertial SLAM

Joint work with P. Furgale, M. Bosse, S. Lynen, M. Chli, V. Rabaud, K. Konolige, and R. Siegwart



https://www.youtube.com/watch?v=TbKEPA2_-m4

Learning Outcomes

- explain the **software** components of a typical mobile robot, as well as their interactions with **hardware** (sensors, motors),
- explain the components of multi-sensor Simultaneous Localisation and Mapping (**SLAM**) systems,
- describe the **kinematics** and **dynamics** of wheeled and flying robots in maths,
- describe **multi-sensor estimators** with sparse and dense map representations in maths,
- describe different **feedback-control** approaches for robots in maths and explain the differences,
- **implement** basic estimators as well as feedback controllers to run in real-time using modern C++.

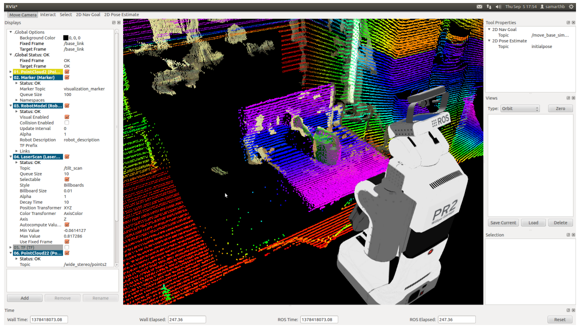
Schedule: Lectures

What	Date	Topic
Lecture 1	**18/01/21**	**Introduction, Problem Formulation, and Examples**
Lecture 2	25/01/21	Representations and Sensors
Lecture 3	01/02/21	Kinematics and Temporal Models
Lecture 4	08/02/21	The Extended Kalman Filter
Lecture 5	15/02/21	Feedback Control
Lecture 6	22/02/21	Nonlinear Least Squares
Lecture 7	01/03/21	Vision-Based Simultaneous Localisation and Mapping
Lecture 8	08/03/21	Revision, Q&A

Practical: Autonomous AR Drone 2 Operation

Goals:

- Perform a simple delivery task with Parrot's AR Drone 2, see www.parrot.com/uk/drones/parrot-ardrone-20-elite-édition
- Implement state estimation and control
- Using the Robot Operating System (ROS) for communication and introspection, see www.ros.org
- Write generic **and** efficient C++ code, exploiting Inheritance, templates, and the Eigen library.
- **This year: simulation-based (ROS Gazebo), but with seamless transferability to the real-world**
- For assessments: submit git repo commit hash through CATe



Please form teams of 3-4 students by this week's practical session (see CATe: group formation)

System Setup for Practicals

We recommend you set everything up on your own machine as follows.

Ubuntu 18.04 or 20.04 (recommended and tested by us)

- Install system dependencies:

```
sudo apt-get install libsdl1.2-dev libsdl2-dev
```

- Install ROS 1:

```
sudo sh -c 'echo "deb http://packages.ros.org/ros/ubuntu $(lsb_release -
sc) main" > /etc/apt/sources.list.d/ros-latest.list'
```

```
sudo apt-key adv --keyserver 'hkp://keyserver.ubuntu.com:80' --recv-key
C1CF6E31E6BADE8868B172B4F42ED6FBAB17C654
```

```
sudo apt-get update
```

- Then, on Ubuntu 20.04:

```
sudo apt-get install ros-noetic-desktop-full
```

- Or on Ubuntu 18.04:

```
sudo apt-get install ros-melodic-desktop-full
```

Other Systems

- Make sure to have SDL 1.2 and 2 installed.
- Regarding ROS 1, please see http://wiki.ros.org/noetic/Installation

AR Drones: The Amazin' Challenge

AR Drones: The Amazin' Challenge

Schedule: Practicals

What	Date	Topic
Practical 1	21/01/21	Get the drone off the ground
Practical 2	28/01/21	Vision-only pose estimation
Practical 2	04/02/21	Vision-only pose estimation
Practical 3	11/02/21	Visual-inertial state estimation
Practical 3	18/02/21	Visual-inertial state estimation
Practical 4	25/02/21	Feedback Control
Practical 5	04/03/21	Amazin' Challenge (preparation)
Practical 5	11/03/21	The Amazin' Challenge

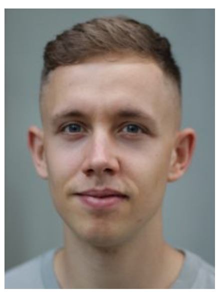
The Team

Lecturer:

Dr Stefan Leutenegger
Room E360, ACEX
Tel: +44 (0)20 7594 7123
Email: s.leutenegger@imperial.ac.uk
Web: http://wp.doc.ic.ac.uk/sleutene

Teaching Assistants:

Dr Masha
Popovic



Nils
Funk

Sotiris
Papatheodorou

Binbin
Xu

Course Material

Course Material: CATe

- Schedule
- Slides
- Practical task sheets

Literature (note: very extensive):

[5] Probabilistic Robotics (2006),
by Sebastian Thrun, Wolfram Burgard, and Dieter Fox.
See also http://www.probabilistic-robotics.org/

[6] Estimation with Applications to Tracking and Navigation: Theory Algorithms and Software (2001),
by Yaakov Bar-Shalom, X. Rong Li, and Thiagalingam Kirubarajan.

[7] Optimal State Estimation: Kalman, H Infinity, and Nonlinear Approaches (2006),
by Dan Simon.

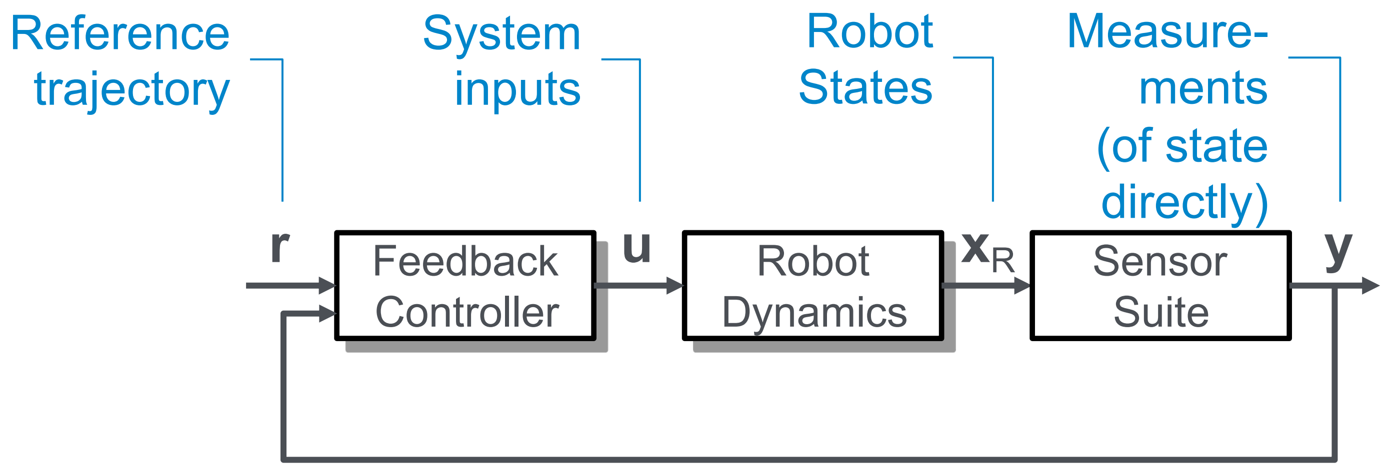
[8] Szeliski, Richard. Computer vision: algorithms and applications. Springer Science & Business Media, 2010.

Assessment

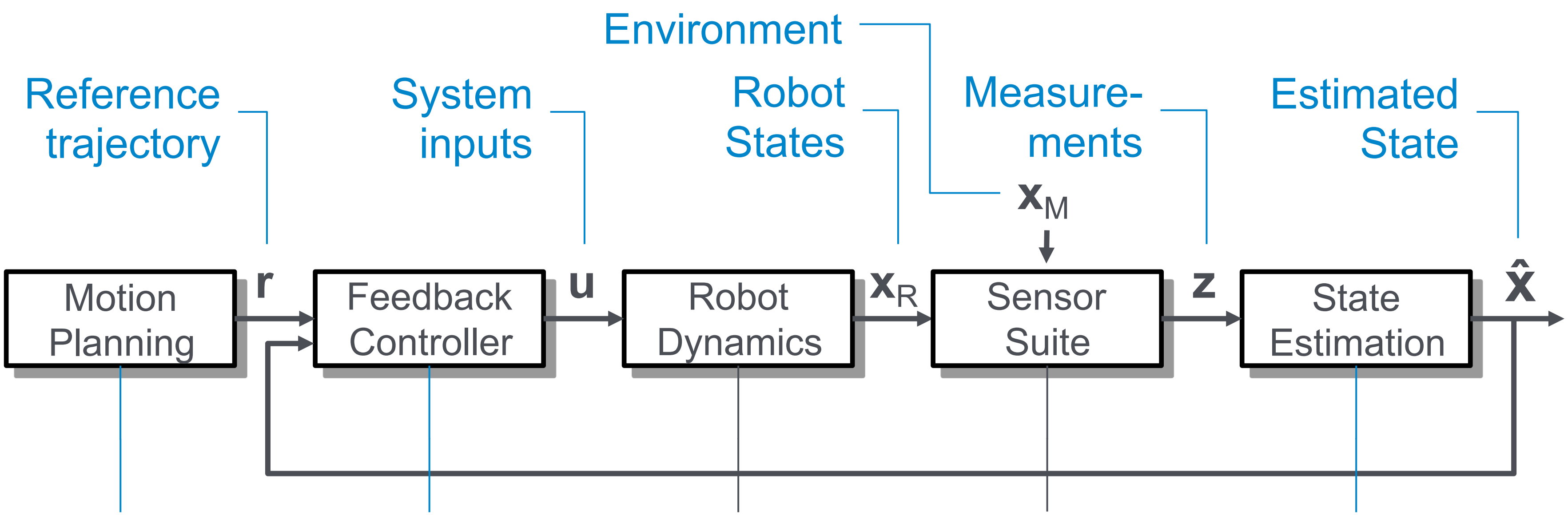
- 3 out of 5 practicals (details see CATe)
- 120 minutes written exam: answer 3 questions

Any questions about the course?

A Relatively Simple Setting



A More Realistic Setting



What trajectory that is safe do we want the robot to follow?

What commands should we send to the robot so it follows the reference trajectory?

How does the robot react to inputs given initial states?

How are measure-ments are generated given the robot states?

What is the best explan-ation for the robot states given the measure-ments?

To Be Estimated: Robot States and Environment

Robot state

- Typical components:
- Position
- Orientation
- Velocity
- Sensor-internal states
- …

Characteristics:

- Typically *time-varying*

Map – the environment

- Popular representations:
- 3D points (sparse)
- 3D pointcloud/surfels (dense)
- Occupancy grid / voxels
- Triangulated mesh
- …

Characteristics:

- Typically, a large part is assumed to be *static*

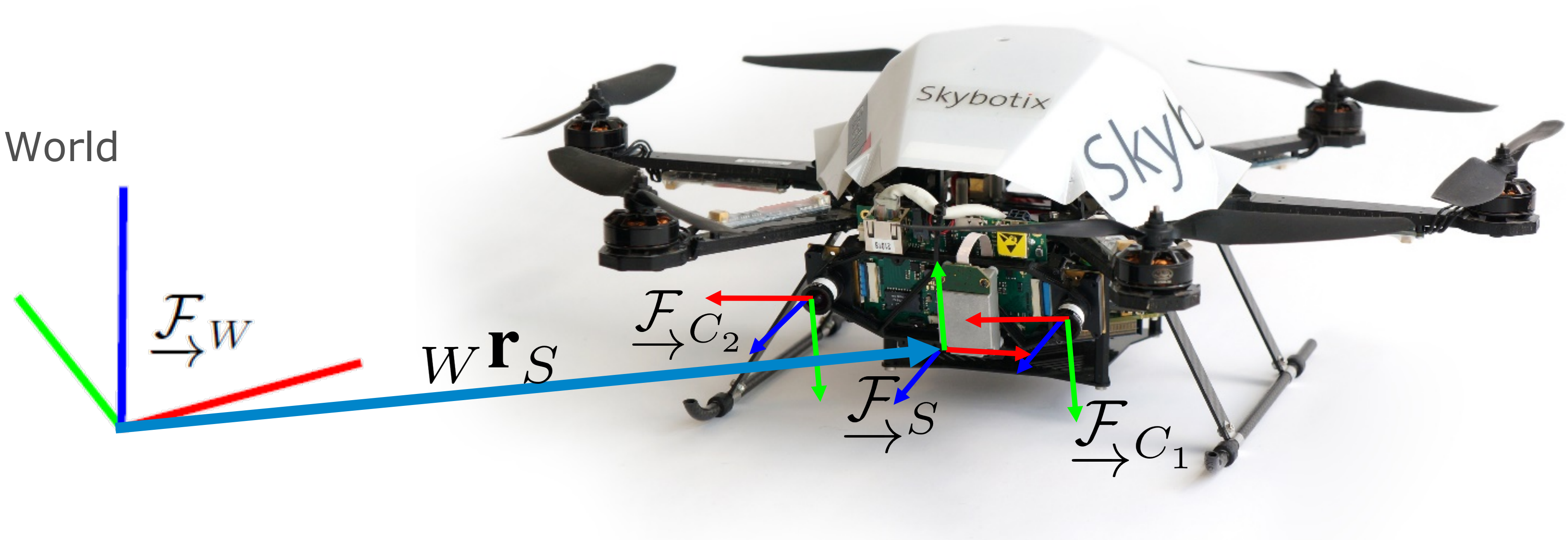
Robot State

An example state vector:

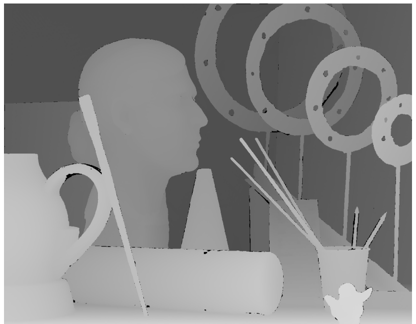
$$\mathbf{x}_{\mathrm{R}}^T = \left[{}_W\mathbf{r}_S^T, \mathbf{q}_{WS}^T, {}_W\mathbf{v}^T, {}_S\mathbf{r}_{C_1}^T, \mathbf{q}_{SC_1}^T, {}_S\mathbf{r}_{C_2}^T, \mathbf{q}_{SC_2}^T, \mathbf{b}_g^T, \mathbf{b}_a^T\right]$$

Position and orientation of frame S origin w.r.t. in W	Velocity of frame S origin expressed in W	Poses of cameras 1 and 2 relative to frame S	IMU gyro and accelerometer biases

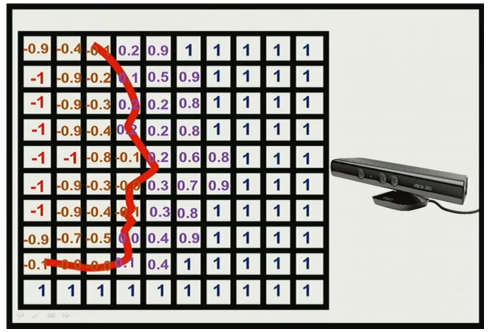
Physical robot states | **Sensor extrinsics (here: camera)** | **Sensor intrinsics (here: IMU)**



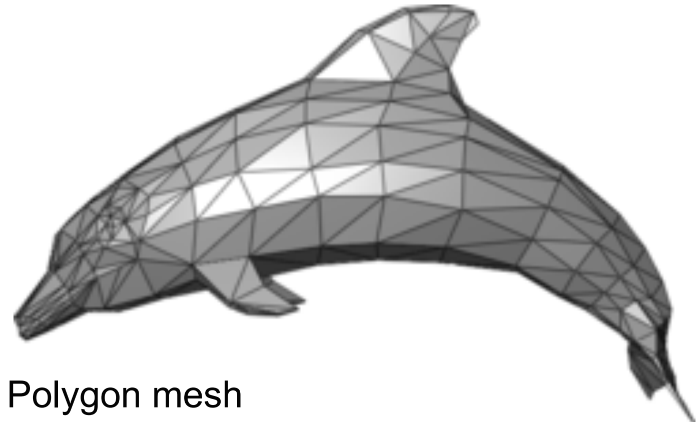
Map Representations



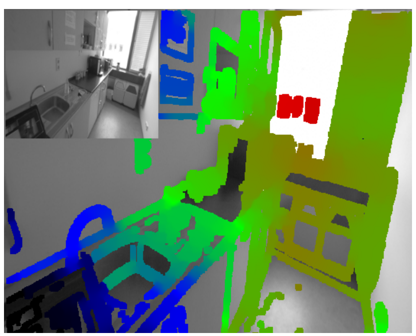
Depth maps
[vision.middlebury.edu]



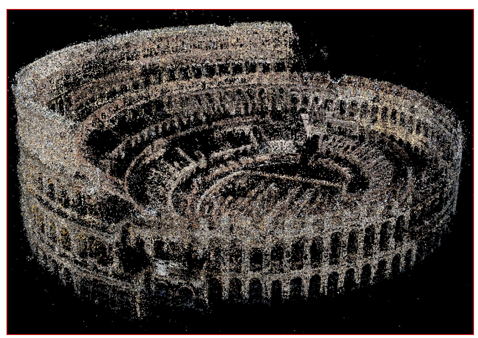
Truncated Signed Distance
Function [pointclouds.org]



Polygon mesh
[en.wikipedia.org/wiki/Polygon_mesh]



Semi-dense depth maps
[vision.in.tum.de]



Point clouds (here: sparse)
[grail.cs.washington.edu]

Surfel maps
[wp.doc.ic.ac.uk/thefutureofslam]

Probabilistic Estimation

We use Bayesian Statistics:

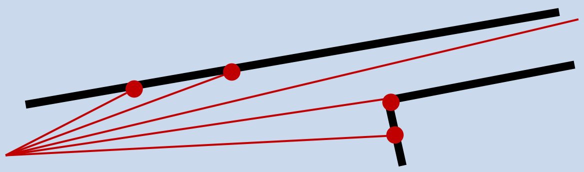
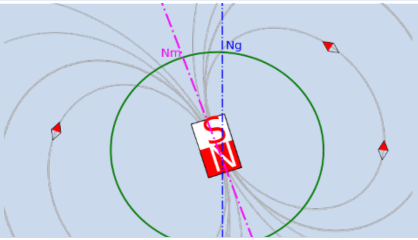
Measurements $\mathbf{z}$ are modeled as samples from a **distribution** $p(\mathbf{z}|\mathbf{x})$ given the variables $\mathbf{x}$ (here: robot states plus possibly the map).

- **Maximum Likelihood (ML) Estimation:**
  What values for variables best explain all the measurements?

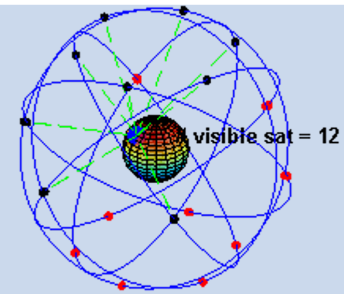
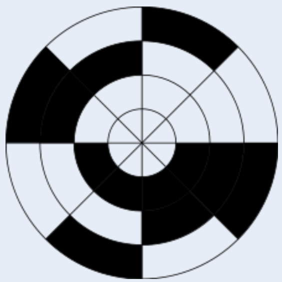
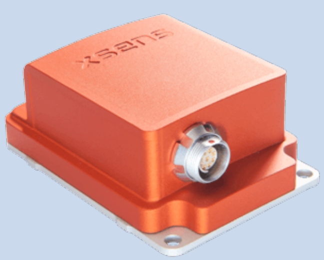
- **Maximum a Posteriori (MAP) Estimation:**
  What values for variables best explain all the measurements and a prior belief $p(\mathbf{x})$ about those variables?

How to compute these values? To be discussed in later lectures.

Typical Sensors – Exteroceptive

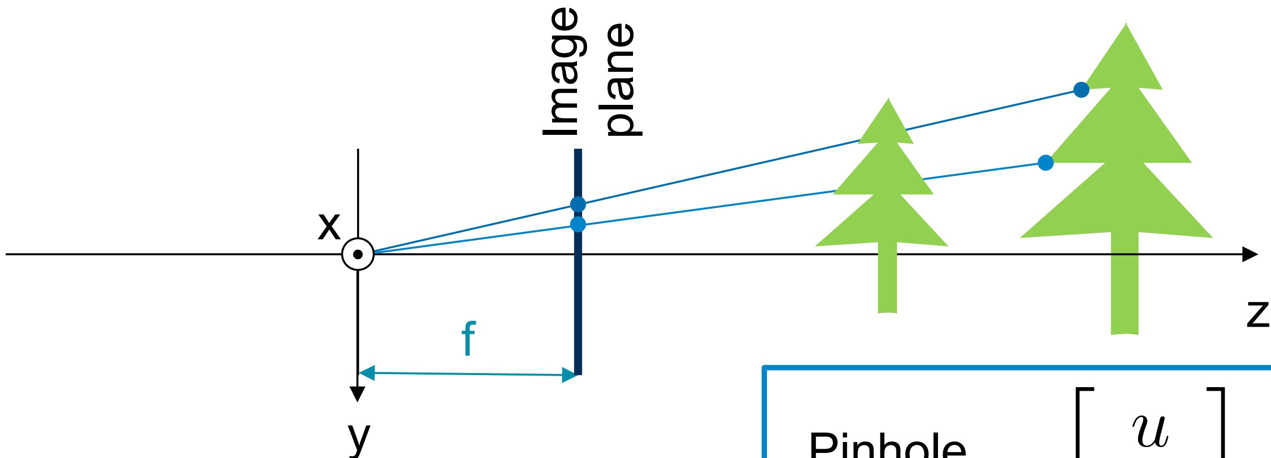
Sensor	Measurement	
Laser Scanner	3D points	
Camera	(Colour) image (RGB-D: with depth!)	
Magnetometer	3D magnetic field	
Pressure sensor	Air pressure (altitude / airspeed)	

Typical Sensors – Proprioceptive

Sensor	Measurement	
GPS	pseudo-ranges (position)	
Encoders	Joint / wheel angles	
Inertial Measurement Unit (IMU)	Rotation rates and accelerations (with caution: orientation)	

The Camera

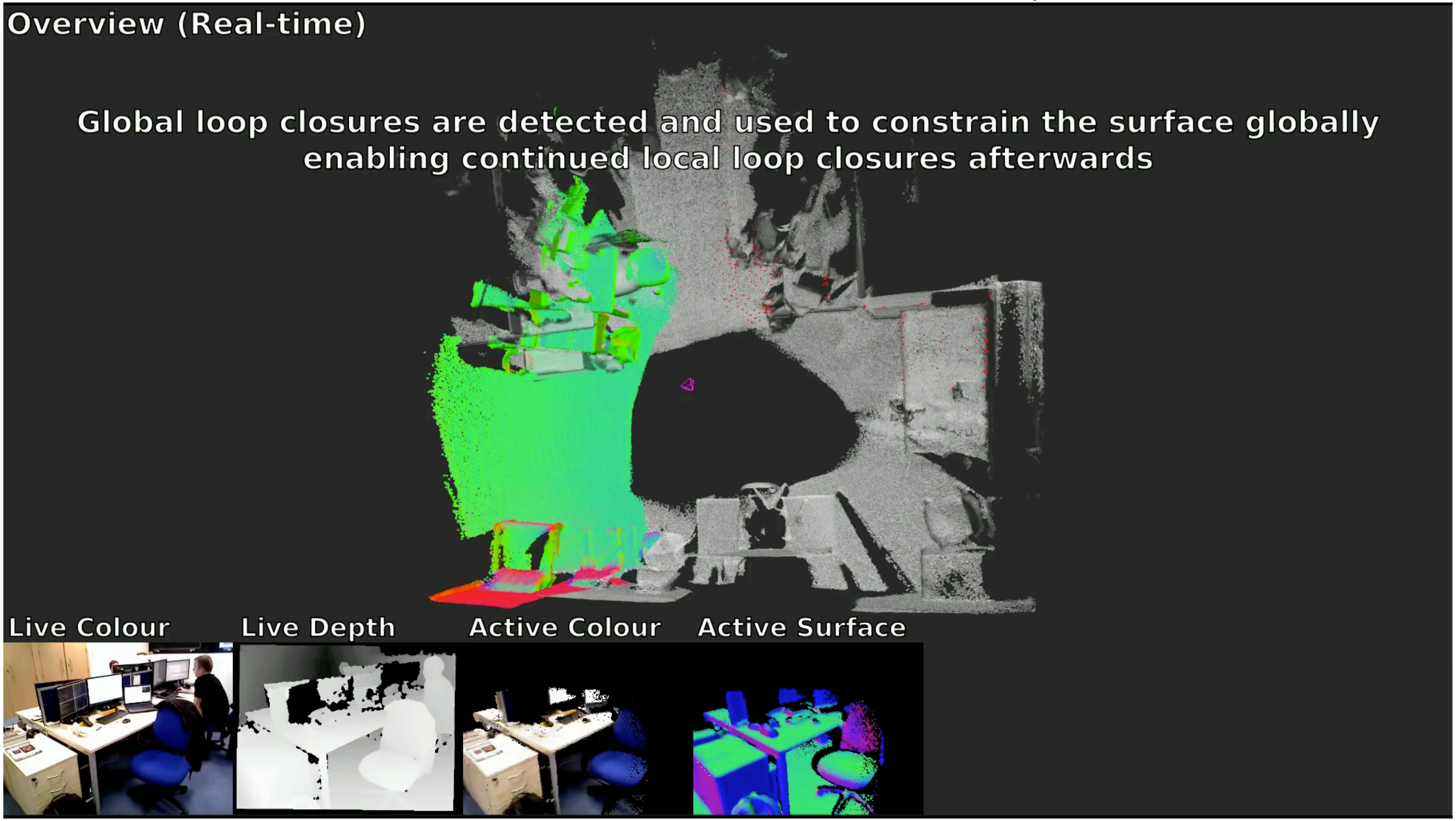
Simplified pinhole model



Pinhole projection: $$\begin{bmatrix} u \\ v \end{bmatrix} = \frac{f}{z} \begin{bmatrix} x \\ y \end{bmatrix}$$

ElasticFusion [RSS'15, IJRR16]

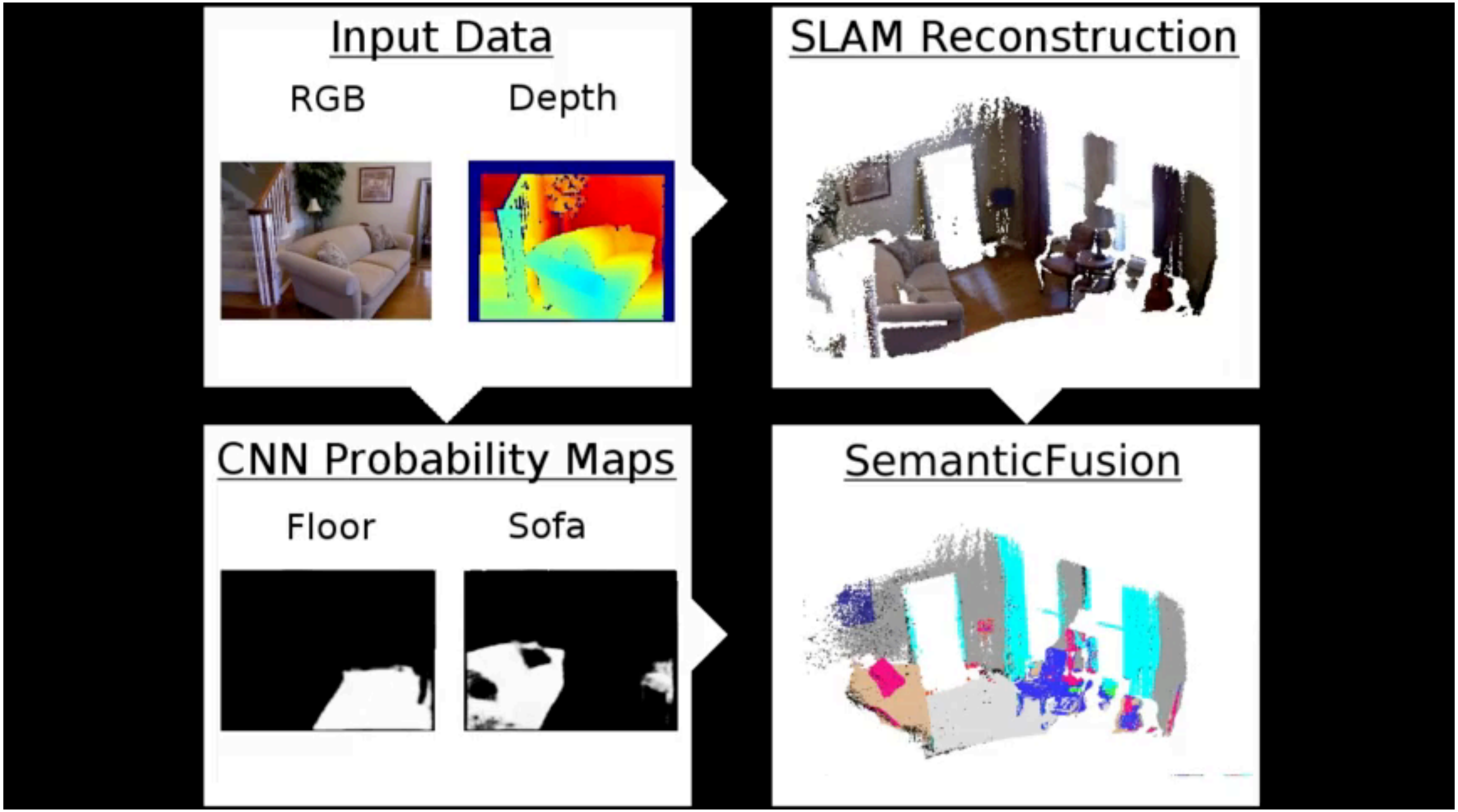
Joint work with Tom Whelan, Renato Salas-Moreno, Ben Glocker, and Andy Davison



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

Fusing Semantics From CNN [ICRA'17]

Joint work with John McCormac, Ankur Handa, and Andrew Davison



https://www.youtube.com/watch?v=cGuoyNY54kU

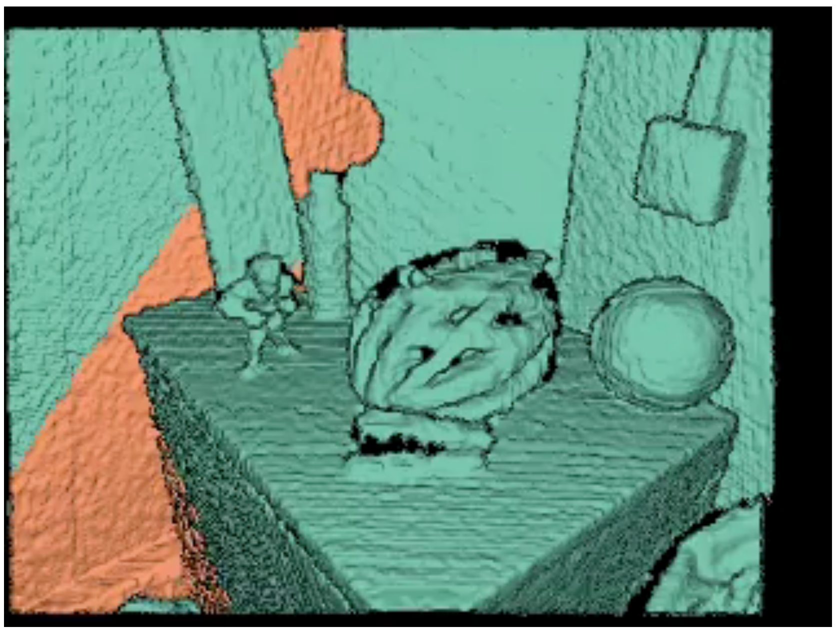
Synthetic Data: Decrease Reality Gap [BMVC'18]

Ongoing work with Wenbin Li, John McCormac, Ronnie Clark, et al.

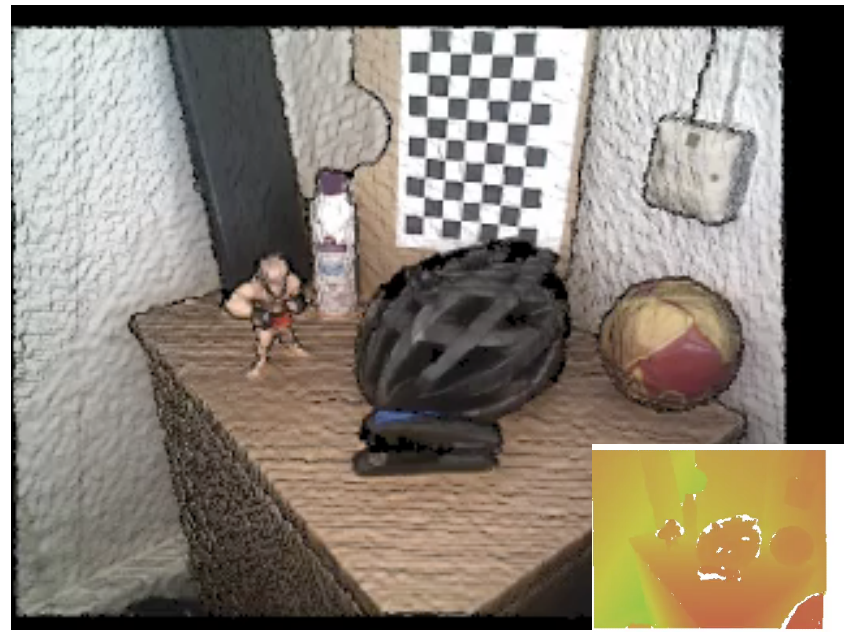
Live Reconstruction [3DV'19]

Emanuele Vespa, Nils Funk, Paul Kelly, Stefan Leutenegger

3D reconstruction



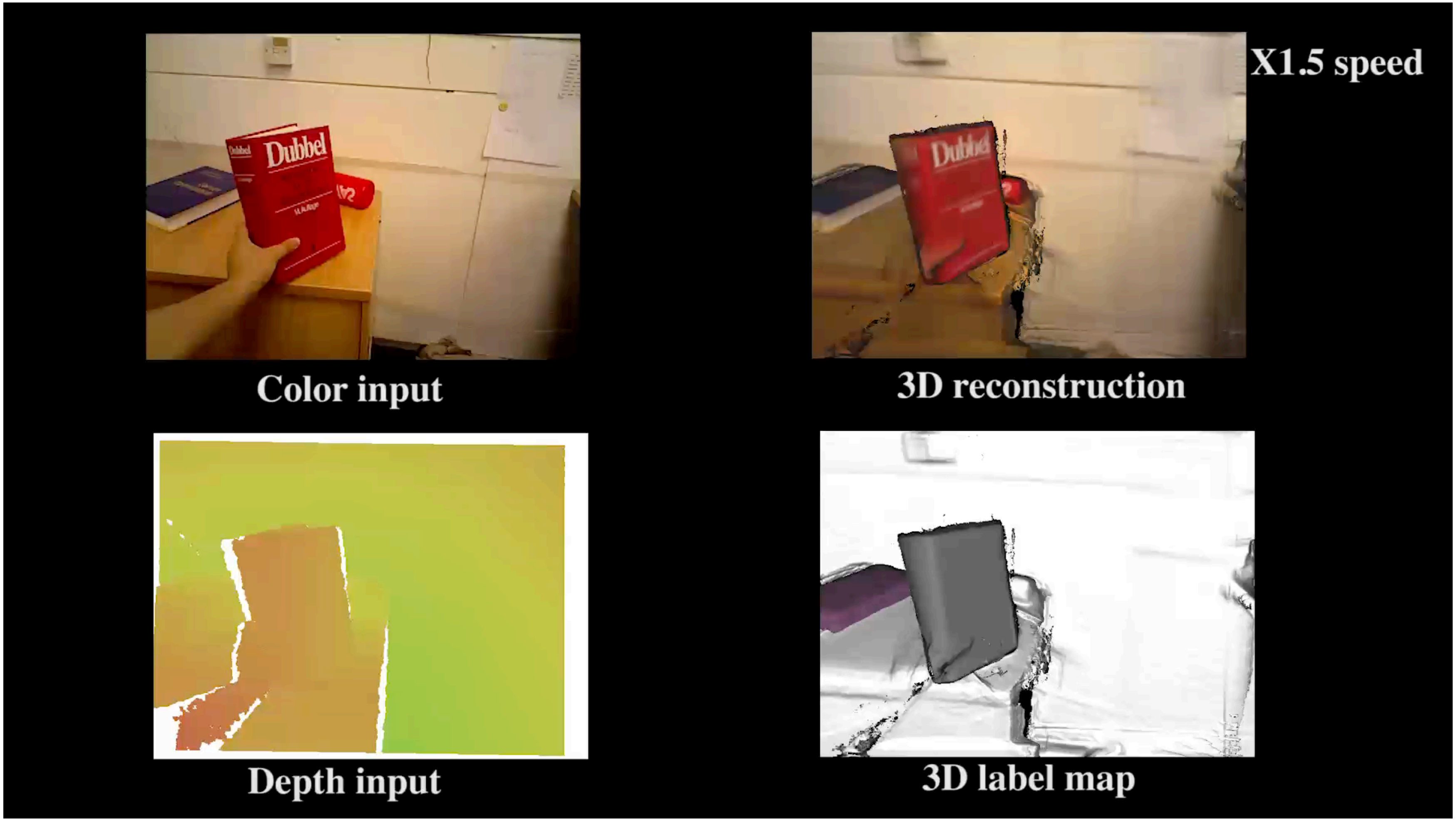
3D RGB reconstruction



Raw depth

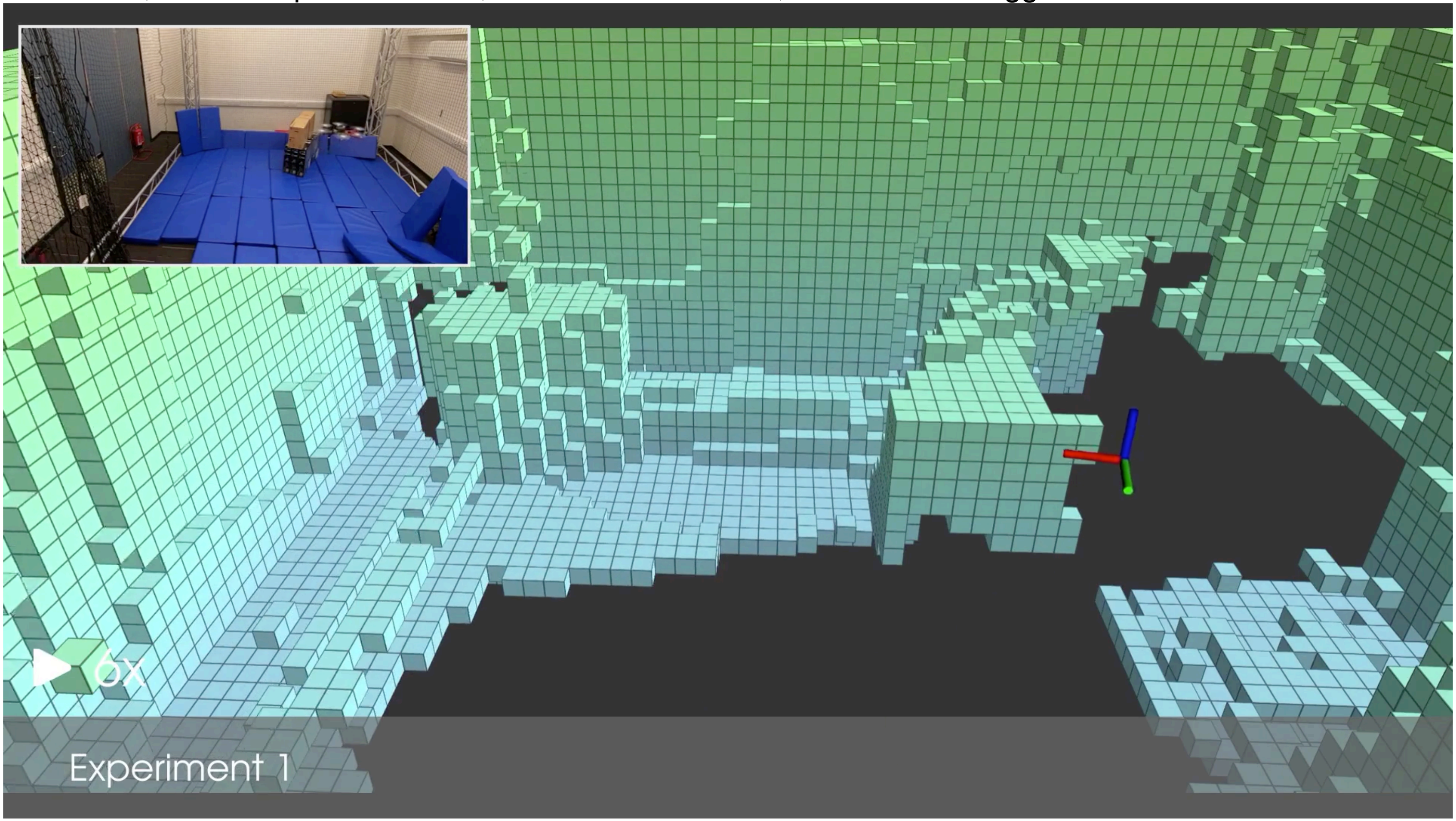
MID-Fusion [under review]

Joint work with Binbin Xu, Dimos Tzoumanikas, Michael Bloesch, and Andrew Davison



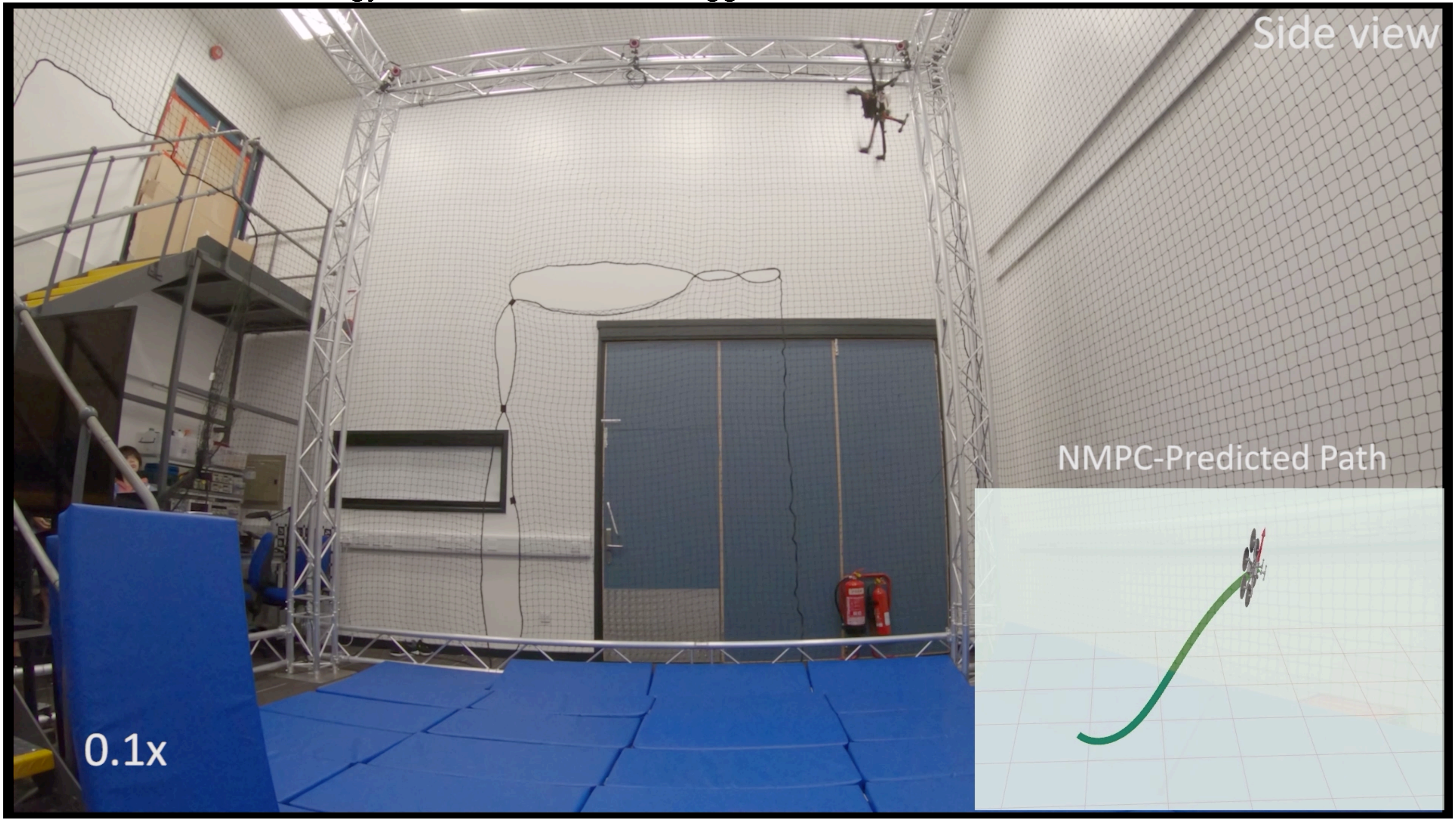
Drone Exploration [Under Review]

Anna Dai*, Sotiris Papatheodorou*, Dimos Tzoumanikas, Stefan Leutenegger



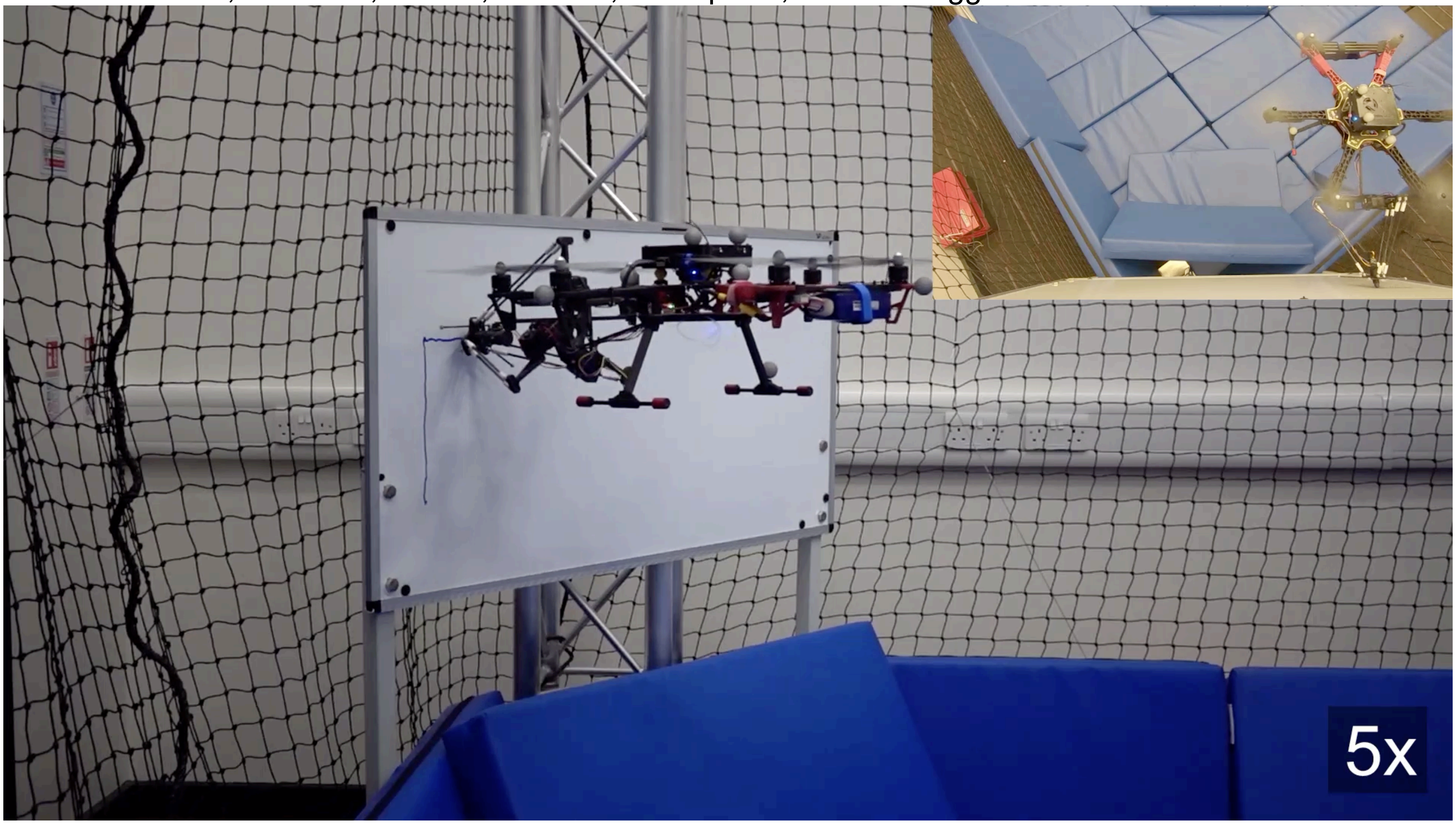
Aggressive Flight / Failure Recovery [Under Review]

Dimos Tzoumanikas, Qingyue Yan, Stefan Leutenegger



Towards Mobile Manipulation [RSS'20]

D. Tzoumanikas, F. Graule, Q. Yan, D. Shah, M. Popovic, S. Leutenegger



Imperial College London

SMART ROBOTICS LAB

Any Questions?

See you on **Thursday at 1pm for the practical** (see CECLAT Teams link).
In preparation:

- Please form teams of 3-4 students (see CATe group formation)
- Feel free to already set up your system (see slide 15)