Robot Dynamics

Rotary Wing UAS: Case Study

Weixuan Zhang

151-0851-00 V

Marco Hutter, Roland Siegwart and Thomas Stastny

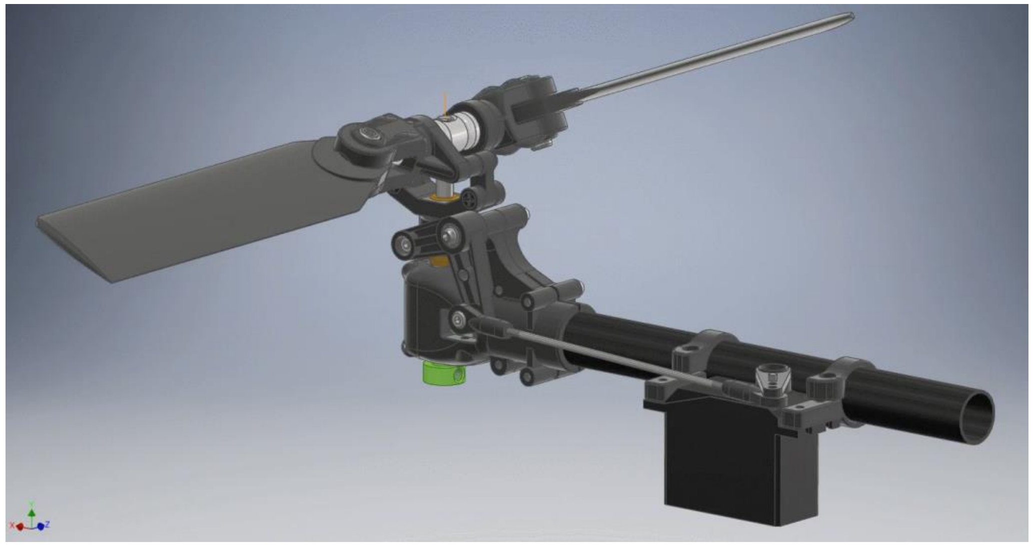
Contents | Rotary Wing UAS

Multirotor Configurations

Multirotor Configurations

- Hexacopter/Octocopter
  - Similar to quadcopters
  - Mechanical redundancy
  - More involved control allocation

- Variable pitch rotors
  - More responsive
  - Propellers can reverse thrust
  - Mechanically more complex than fixed-pitch multirotor.



Multirotor Configurations

- Tail-sitter
  - Vertical take off and landing
  - Can cover wide area
  - Difficult transition control

- Tiltrotor
  - Direct control of forces and moments
  - Complex control allocation

ETH zürich

Multirotor Applications

Multirotor Applications | Inspection

- Industrial inspection (e.g. pipe thickness)
- Bridge inspection
- Tree cavity inspection

Multirotor Applications | Art

- Light art
- Footage and cinematographic applications

Multirotor Applications | Delivery

- Lightweight objects transportation
- Medicine delivery in remote areas

Multirotor Applications | Surveillance

- Military application and border surveillance
- After disaster inspection and damage assessment
- Search and rescue operations

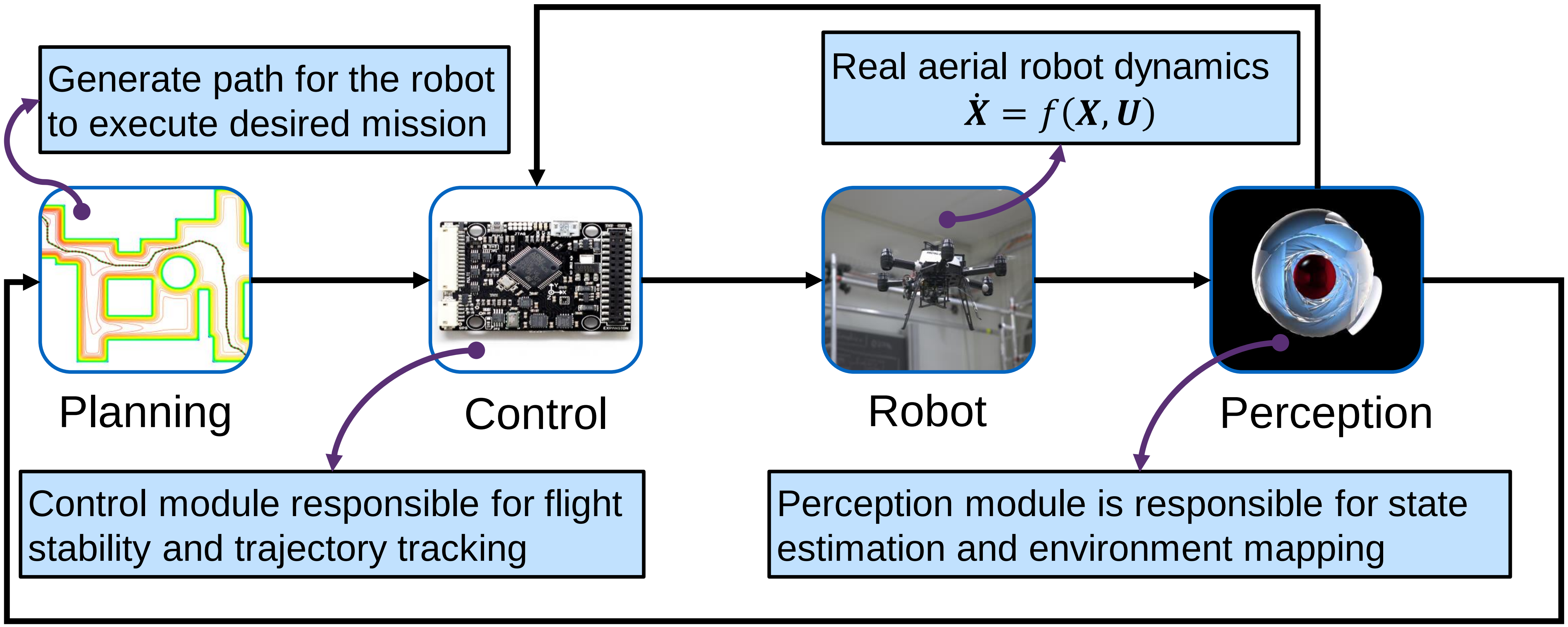
ETH zürich

Micro Aerial Vehicle Autonomy

Micro Aerial Vehicle Autonomy

Autonomous robot is an intelligent machine able to perform tasks without explicit human intervention

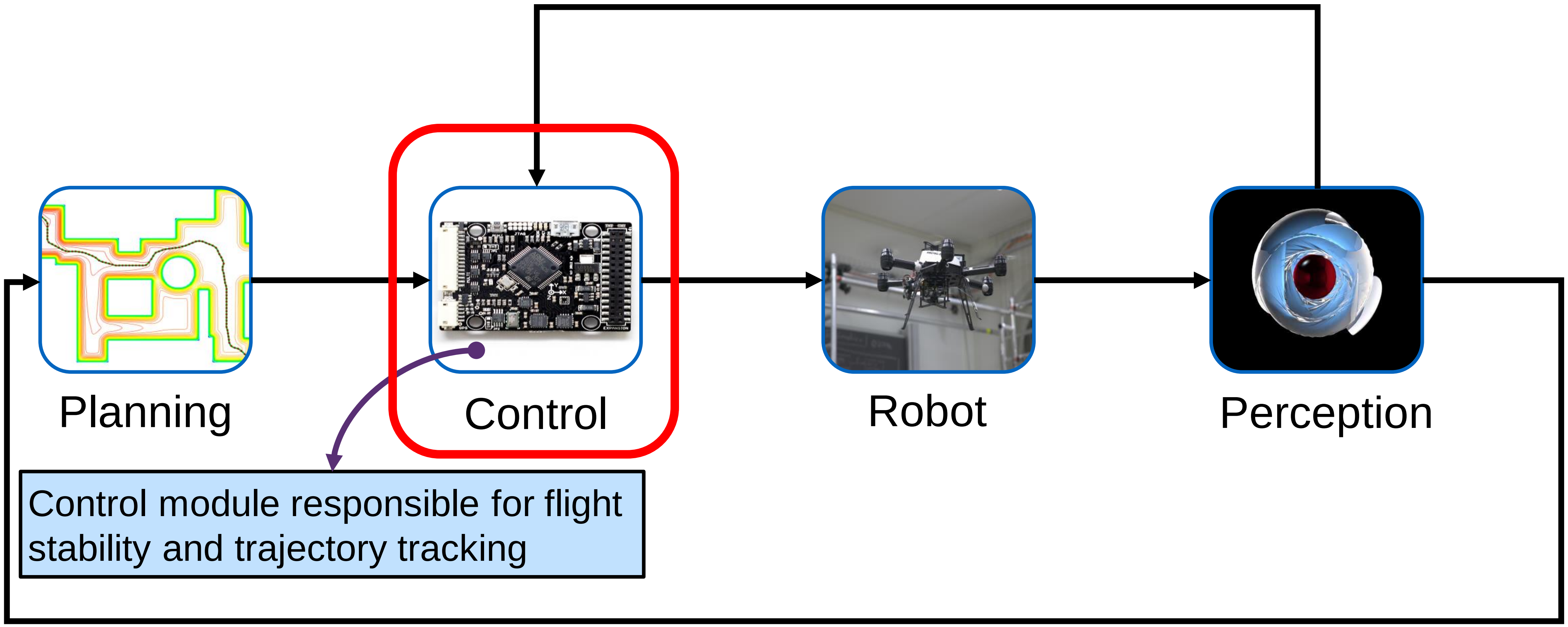
- Basic components to achieve autonomy:



Micro Aerial Vehicle Autonomy | Control

Autonomous robot is an intelligent machine able to perform tasks without explicit human intervention

- Basic components to achieve autonomy:



ETH zürich

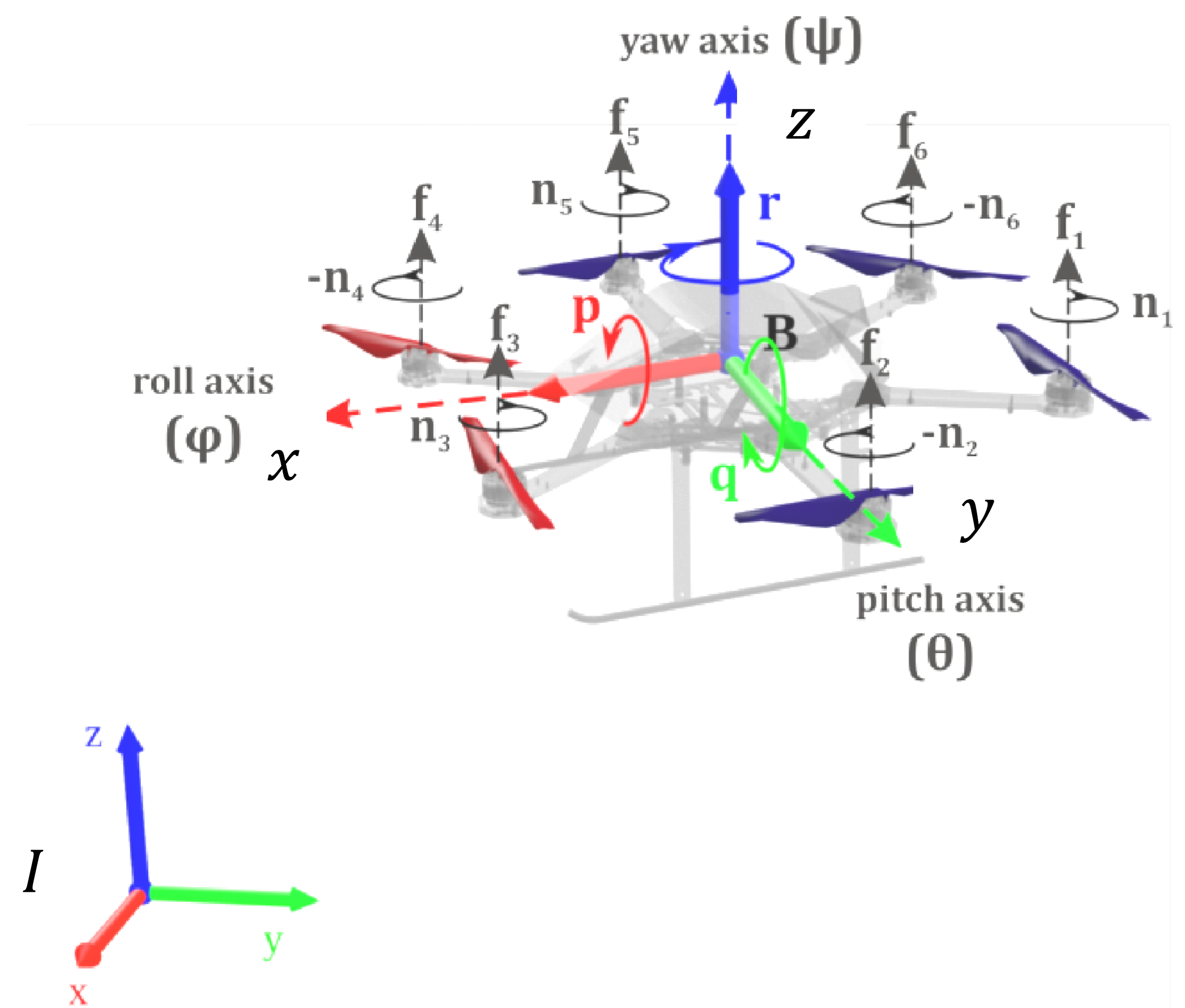
Control for Micro Aerial Vehicles

Control for MAVs | Model recap

- Vehicle dynamics in inertial frame ***W***
  - Translational dynamics
    - $$\begin{cases} \dot{\boldsymbol{p}} = \boldsymbol{R}_{IB}\ {}_{B}\boldsymbol{v} \\ {}_{B}\dot{\boldsymbol{v}} = -\boldsymbol{\omega} \times {}_{B}\boldsymbol{v} + \begin{pmatrix} 0 \\ 0 \\ \frac{U_1}{m} \end{pmatrix} + \boldsymbol{R}_{IB}^{T}\ \boldsymbol{g} \end{cases}$$

Usually negligible small

  - Rotational dynamics
    - $$\begin{cases} \dot{\boldsymbol{R}}_{IB} = \boldsymbol{R}_{IB}\widehat{\boldsymbol{\omega}} \\ \dot{\boldsymbol{\omega}} = \boldsymbol{J}^{-1}\left(-\boldsymbol{\omega} \times \boldsymbol{J}\boldsymbol{\omega} + \begin{pmatrix} U_2 \\ U_3 \\ U_4 \end{pmatrix}\right) \end{cases}$$



Control for MAVs | Model recap

- Vehicle dynamics
  - Translational dynamics
    - $$\begin{cases} \dot{\boldsymbol{p}} = \boldsymbol{R}_{IB}\ {}_B\boldsymbol{v} \\ {}_B\dot{\boldsymbol{v}} = -\boldsymbol{\omega} \times {}_B\boldsymbol{v} + \begin{pmatrix} 0 \\ 0 \\ \frac{U_1}{m} \end{pmatrix} + \boldsymbol{R}_{IB}^T\ \boldsymbol{g} \end{cases}$$
  - Rotational dynamics
    - $$\begin{cases} \dot{\boldsymbol{R}}_{IB} = \boldsymbol{R}_{IB}\widehat{\boldsymbol{\omega}} \\ \dot{\boldsymbol{\omega}} = \boldsymbol{J}^{-1}\left(-\boldsymbol{\omega} \times \boldsymbol{J}\boldsymbol{\omega} + \begin{pmatrix} U_2 \\ U_3 \\ U_4 \end{pmatrix}\right) \end{cases}$$

$\boldsymbol{p}$: vehicle position in inertial frame

$\boldsymbol{v}$: vehicle velocity in inertial frame

m: vehicle mass

$\boldsymbol{g}$: the gravitational acceleration in inertial frame

$\boldsymbol{R}_{IB} \in SO(3)$: is the rotation matrix from body frame to inertial frame

$\boldsymbol{\omega}$: vehicle angular velocity

$\boldsymbol{J}$: vehicle inertia tensor

$U_1 \ldots U_4$: virtual control inputs

Control for MAVs | Model recap

- Virtual control input (Control allocation)

$$\begin{pmatrix} U_1 \\ U_2 \\ U_3 \\ U_4 \end{pmatrix} = \boldsymbol{A} \begin{pmatrix} \omega_1^2 \\ \vdots \\ \omega_{n_r}^2 \end{pmatrix}$$

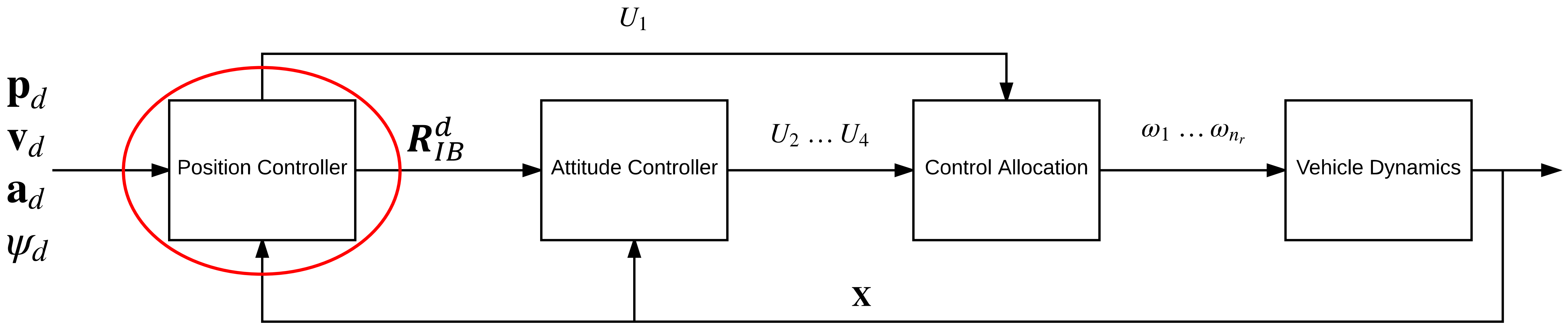
A: allocation matrix dependent on rotors geometric configuration

ω_i: rotational velocity of $i - th$ rotor

n_r: number of rotors

Control for MAVs | Geometric Control on SE(3)

- Control block diagram



*Lee, Taeyoung, Melvin Leoky, and N. Harris McClamroch. "Geometric tracking control of a quadrotor UAV on SE (3)." *49th IEEE conference on decision and control (CDC)*. IEEE, 2010.

Control for MAVs | Geometric Control on SE(3)

- Trajectory Tracking Controller

$$\boldsymbol{e_p} = \boldsymbol{p} - \boldsymbol{p_d}, \qquad \boldsymbol{e_v} = \boldsymbol{v} - \boldsymbol{v_d}$$

PD controller

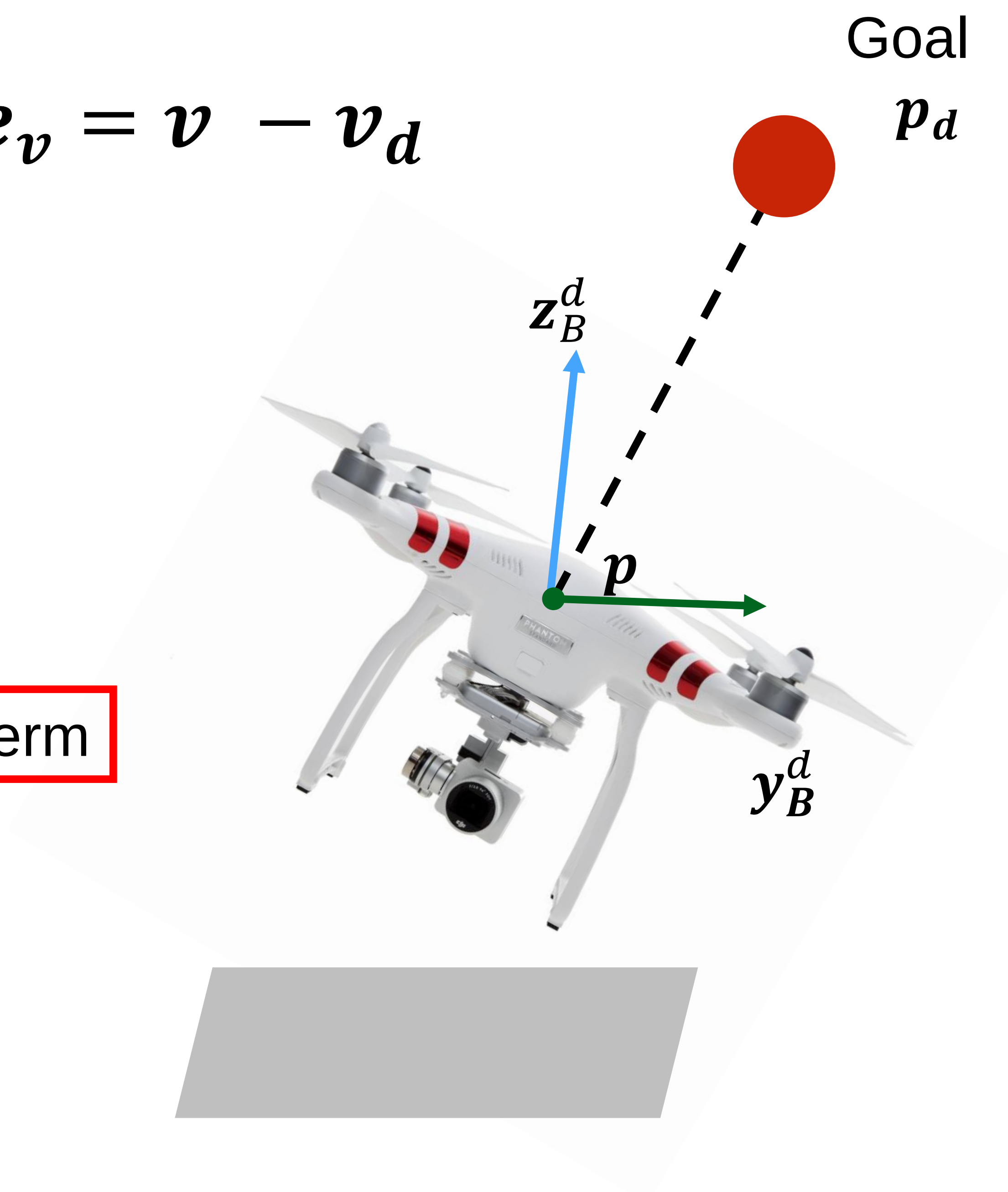
feed-forward term

$$\boldsymbol{z}_B^d = \frac{-K_p \boldsymbol{e_p} - K_v \boldsymbol{e_v} - m(\boldsymbol{g} - \boldsymbol{a_d})}{\left\| -K_p \boldsymbol{e_p} - K_v \boldsymbol{e_v} - m(\boldsymbol{g} - \boldsymbol{a_d}) \right\|}$$

$${}_I\boldsymbol{x}_{temp}^d = (\cos\psi_d \quad \sin\psi_d \quad 0)^T$$

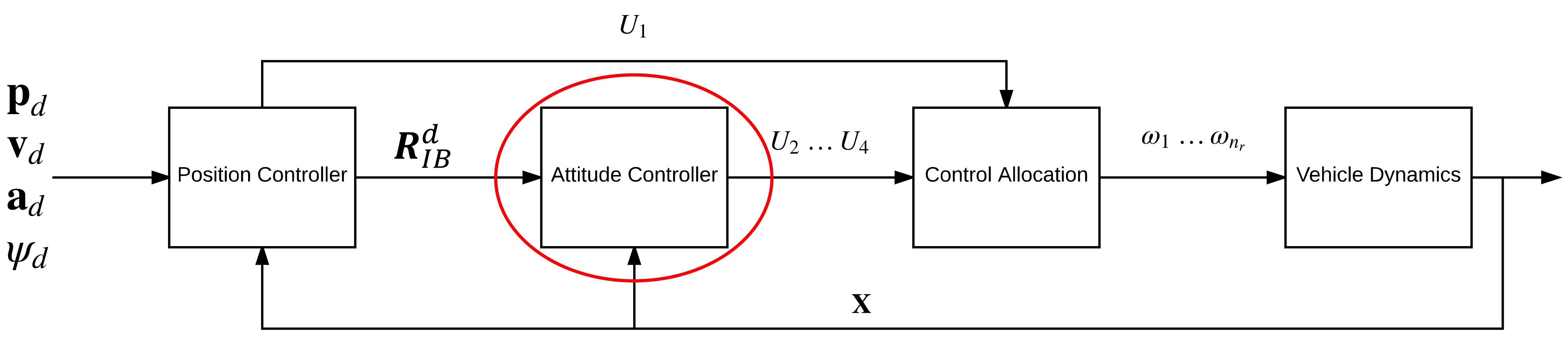
$${}_I\boldsymbol{y}_B^d = \frac{{}_I\boldsymbol{z}_B^d \times {}_I\boldsymbol{x}_{temp}^d}{\left\| {}_I\boldsymbol{z}_B^d \times {}_I\boldsymbol{x}_{temp}^d \right\|}$$

From this construct $\boldsymbol{R}_{IB}^d = ({}_I\boldsymbol{y}_B^d \times {}_I\boldsymbol{z}_B^d \quad {}_I\boldsymbol{y}_B^d \quad {}_I\boldsymbol{z}_B^d)$



Control for MAVs | Geometric Control* on SE(3)

- Control block diagram



Control for MAVs | Geometric Control on SE(3)

- Attitude control
  - Attitude dynamics are evolving on a nonlinear manifold SO(3)
  - Euler angles will always have a singularity
  - PD attitude controller is not globally stable, instability occurs at large angular error

Control for MAVs | Geometric Control on SE(3)

- Attitude control

$$\begin{pmatrix} U_2 \\ U_3 \\ U_4 \end{pmatrix} = -\boldsymbol{K_R e_R} - \boldsymbol{K_\omega e_\omega}$$

v-map: the inverse of the hat map.
Skew-symmetric matrix to vector

$$\boldsymbol{e_R} = \frac{1}{2}\left(\left(\boldsymbol{R}_{IB}^{d}\right)^{T}\boldsymbol{R}_{IB} - \boldsymbol{R}_{IB}^{T}\boldsymbol{R}_{IB}^{d}\right)^{\vee}$$

$$\boldsymbol{e_\omega} = \boldsymbol{\omega} - \boldsymbol{R}_{IB}^{T}\boldsymbol{R}_{IB}^{d}\boldsymbol{\omega_d}$$

Control for MAVs | Geometric Control on SE(3)

- Trajectory Tracking Controller
  - Control input

Project thrust vector along current body z-axis

$$U_1 = \left(-K_p \boldsymbol{e_p} - K_v \boldsymbol{e_v} - m(\boldsymbol{g} - \boldsymbol{a_d})\right) \cdot \boldsymbol{z_B}$$

$$\begin{pmatrix} U_2 \\ U_3 \\ U_4 \end{pmatrix} = -K_R \boldsymbol{e_R} - K_\omega \boldsymbol{e_\omega} + \boldsymbol{\omega} \times \boldsymbol{J\omega}$$

This controller is “almost” globally exponentially stable.
This means it can stabilize the vehicle from any initial attitude, except for a set of critical points.

Control for MAVs | Model recap

- Virtual control input (Control allocation)

$$\begin{pmatrix} U_1 \\ U_2 \\ U_3 \\ U_4 \end{pmatrix} = \boldsymbol{A} \begin{pmatrix} \omega_1^2 \\ \vdots \\ \omega_{n_r}^2 \end{pmatrix}$$

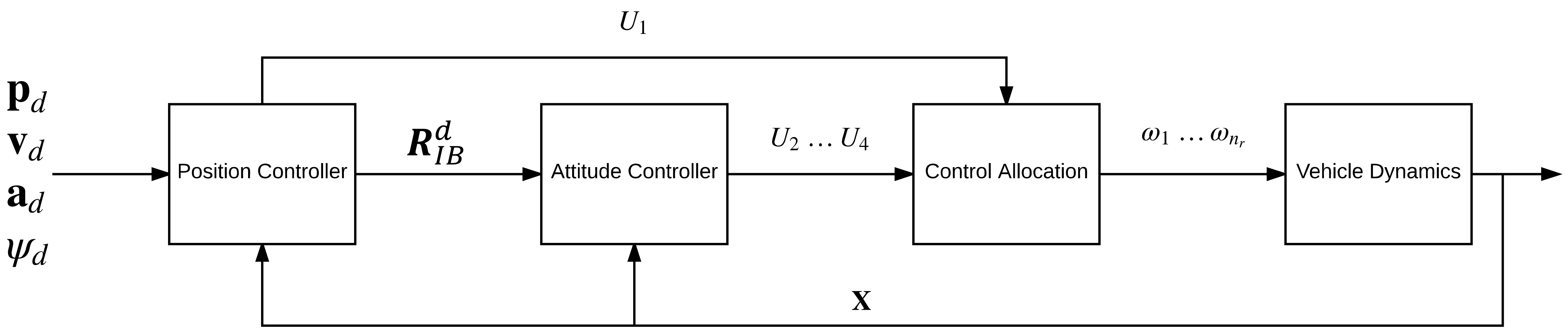
A: allocation matrix dependent on rotors geometric configuration

ω_i: rotational velocity of $i - th$ rotor

n_r: number of rotors

Control for MAVs | Geometric Control on SE(3)

- Control block diagram

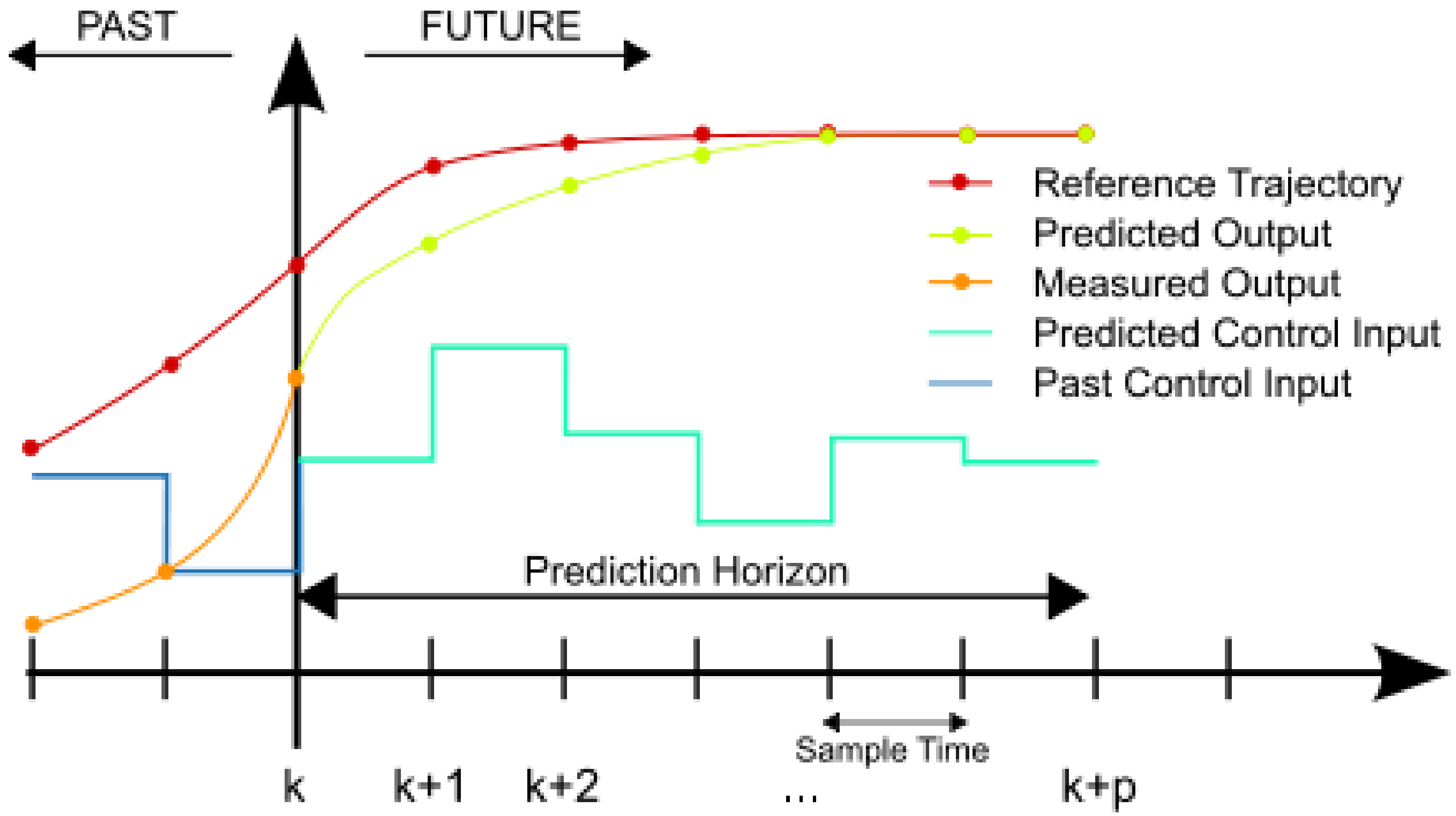


Control for MAVs | Model Predictive Control (MPC)

- MPC for MAV Trajectory Tracking
  - A simplified model of the system is used to predict the vehicle behavior as a function of the control input
  - Every sampling time, an optimization problem is solved to find the "**best**" control input that respects system physical **constraints**
  - In the simple case of no constraints and linearized system, the problem can be solved offline, leading to LQR controller
  - It could be computationally demanding to solve the optimization problem online every time step.

Control for MAVs | Model Predictive Control (MPC)

- MPC for MAV Trajectory Tracking



Control for MAVs | Model Predictive Control (MPC)

- MPC for MAV Trajectory Tracking
  - The Optimal Control Problem we solve every step is given by

$$\min_{\boldsymbol{X},\boldsymbol{U}} \int_{t=0}^{T} \left\| \boldsymbol{x}(t) - \boldsymbol{x}_{ref} \right\|_{\boldsymbol{Q}_{\boldsymbol{x}}} + \left\| \boldsymbol{u}(t) - \boldsymbol{u}_{ref} \right\|_{\boldsymbol{R}_{\boldsymbol{u}}} dt + \left\| \boldsymbol{x}(T) \right\|_{\boldsymbol{P}}$$

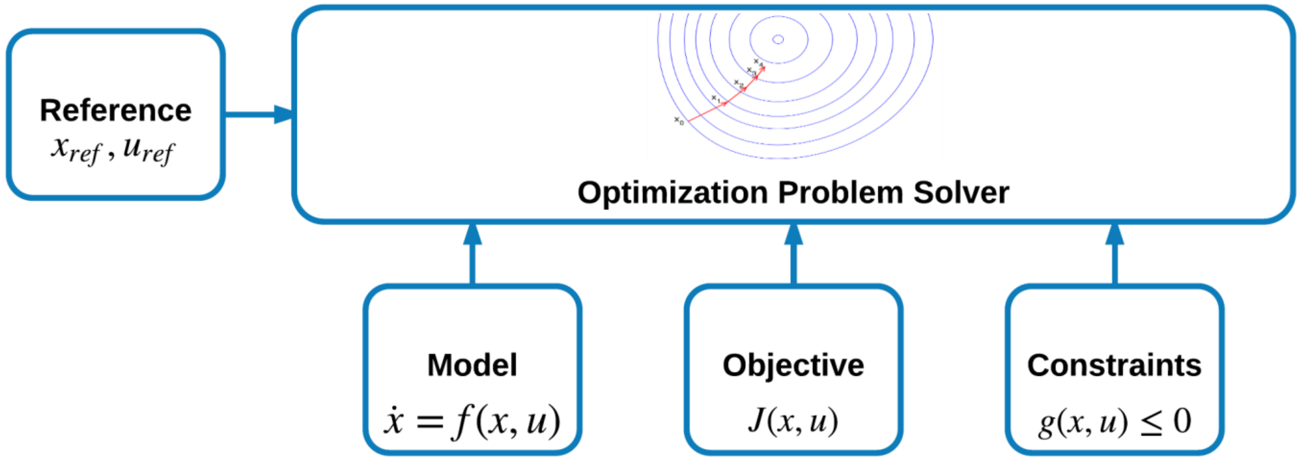
$$subject\ to\text{:} \begin{cases} \dot{\boldsymbol{x}} = \boldsymbol{f}(\boldsymbol{x}, \boldsymbol{u}) \\ \boldsymbol{u}(t) \in \mathbb{U} \\ \boldsymbol{x}(t) \in \mathbb{X} \\ x(0) = x(t_0) \end{cases}$$

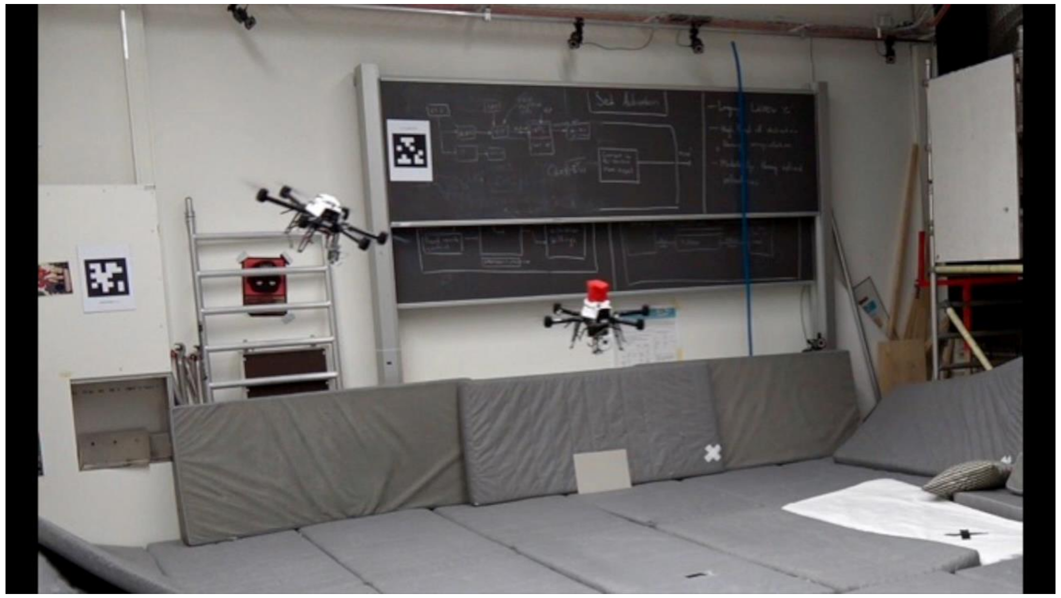
$\boldsymbol{Q}_{\boldsymbol{x}}$: the penalty on state error
$\boldsymbol{R}_{\boldsymbol{u}}$: the penalty on the control input
$\boldsymbol{P}$: terminal state penalty

Efficient solvers can solve this OCP problem in less than 2 ms

Control for MAVs | Model Predictive Control (MPC)

- MPC for MAV Trajectory Tracking
  - system dynamics are taken into account
  - can handle constraints
  - computationally efficient
  - ensures performance and robustness





Control for MAVs | Machine Learning Based

- Fully autonomous airshow

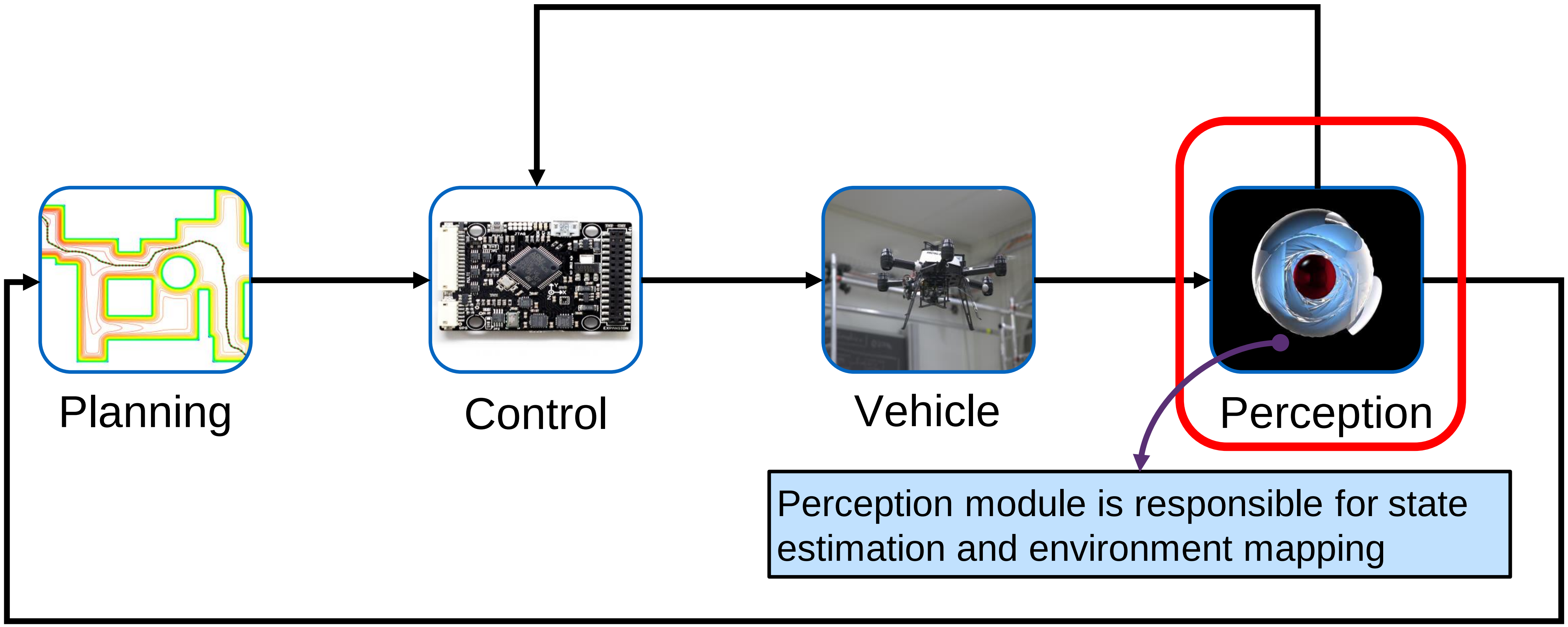
Control for MAVs | Machine Learning Based

- Control using reinforcement learning

Micro Aerial Vehicle Autonomy | Perception

Autonomous robot is an intelligent machine able to perform tasks without explicit human intervention

- Basic components to achieve autonomy:



ETH zürich

Robot Dynamics

Rotary Wing UAS: Case Study

Victor Reijgwart

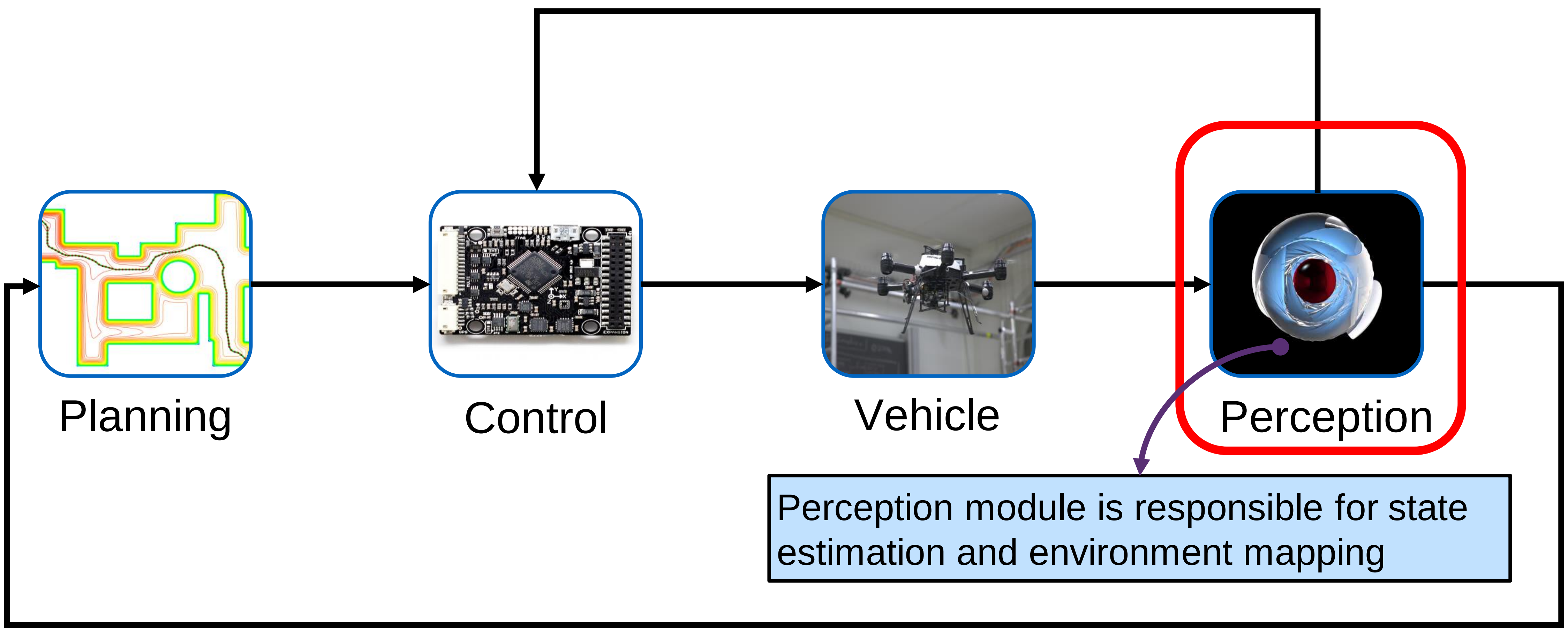
151-0851-00 V

Marco Hutter, Roland Siegwart and Thomas Stastny

Micro Aerial Vehicle Autonomy | Perception

Autonomous robot is an intelligent machine able to perform tasks without explicit human intervention

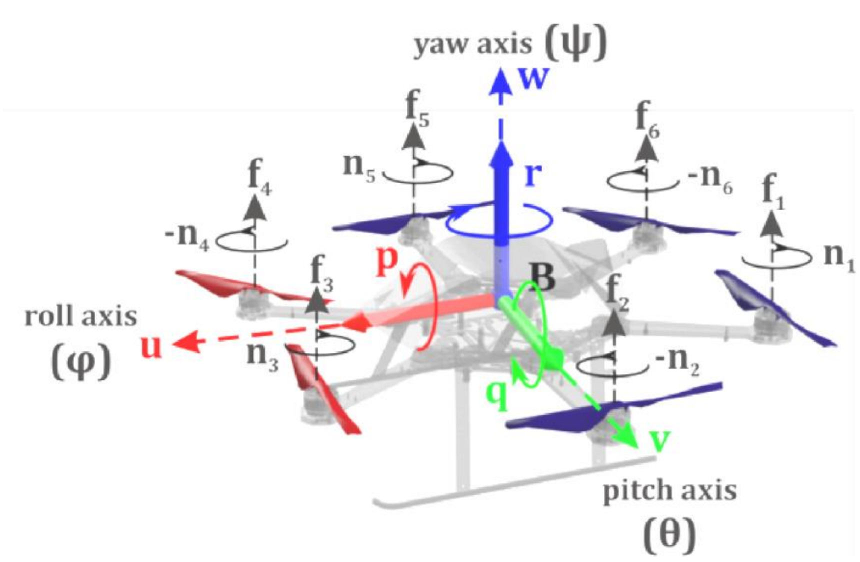
- Basic components to achieve autonomy:



Perception for Micro Aerial Vehicles

Perception for MAVs

$x, y, z, \psi, \theta, \varphi, \ldots ?$

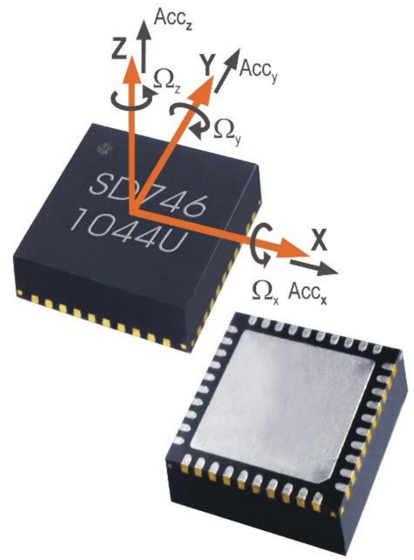


State Estimation
(relative, 50-200 hz)

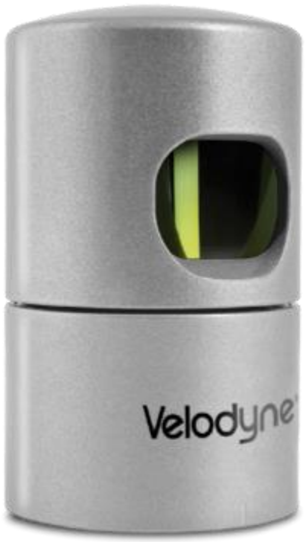
Mapping

Localization
(absolute, few hz)

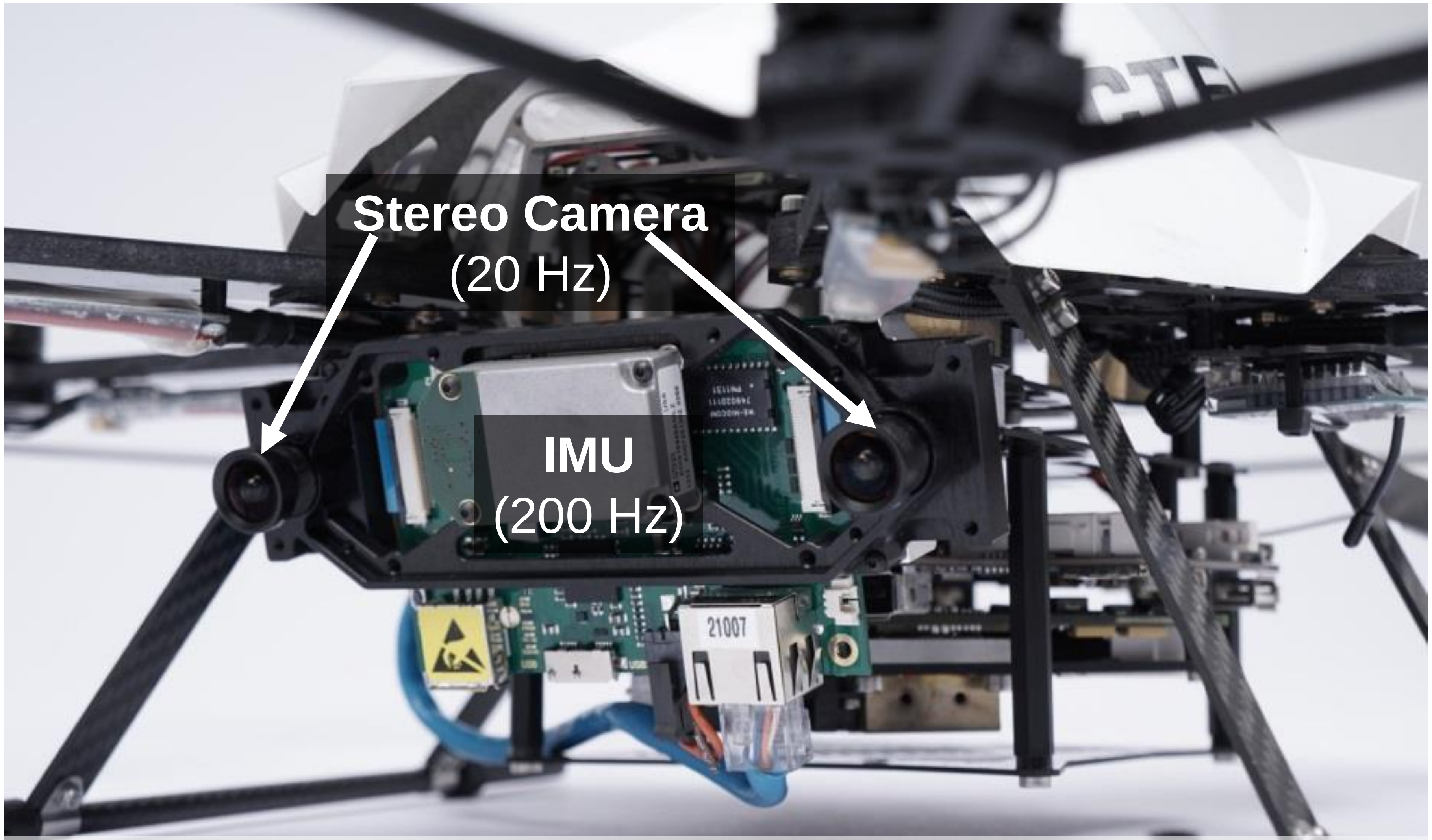
Perception for MAVs | Sensors



- Typical navigation sensors:
  - Inertial sensors (IMU)
    - Accelerometers
    - Gyroscope
  - Camera (mono, stereo, depth)
  - LiDAR
  - GPS
  - Magnetometer (compass)
  - Barometers (pressure sensor)



Perception for MAVs | Sensors



Nikolic, Janosch, et al. "A synchronized visual-inertial sensor system with FPGA pre-processing for accurate real-time SLAM." *Robotics and Automation (ICRA), 2014 IEEE International Conference on*. IEEE, 2014.

Perception for MAVs

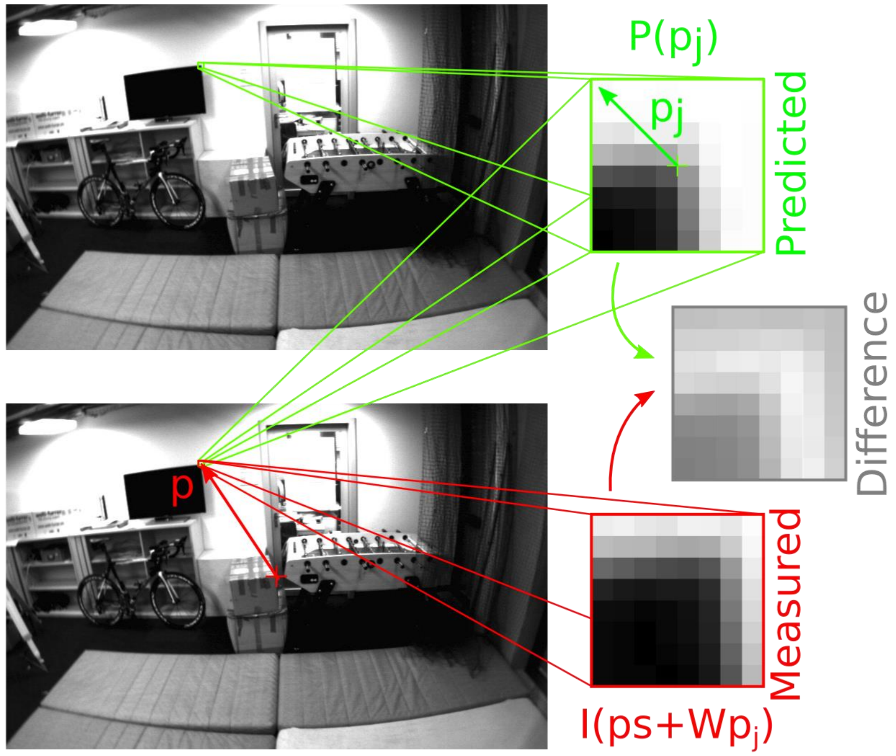
- Perception onboard of MAVs is challenging:
  - Limited computation power
  - Limited payload
  - Limited sensor quality
  - Fast dynamics
  - Unlike ground robots, MAVs operate in 3D space

Perception for MAVs | State Estimation

- In a perfect world: Twice integrate accelerations and rotational velocities from IMU
  - (doesn’t work – IMU has non-zero mean noise = BIAS!)
- Idea: Use sensor fusion to constrain/compensate!
- Common:
  - GPS + IMU (Accelerometer + Gyroscope)
  - **Visual-Inertial**: Camera + IMU (Accelerometer + Gyroscope)

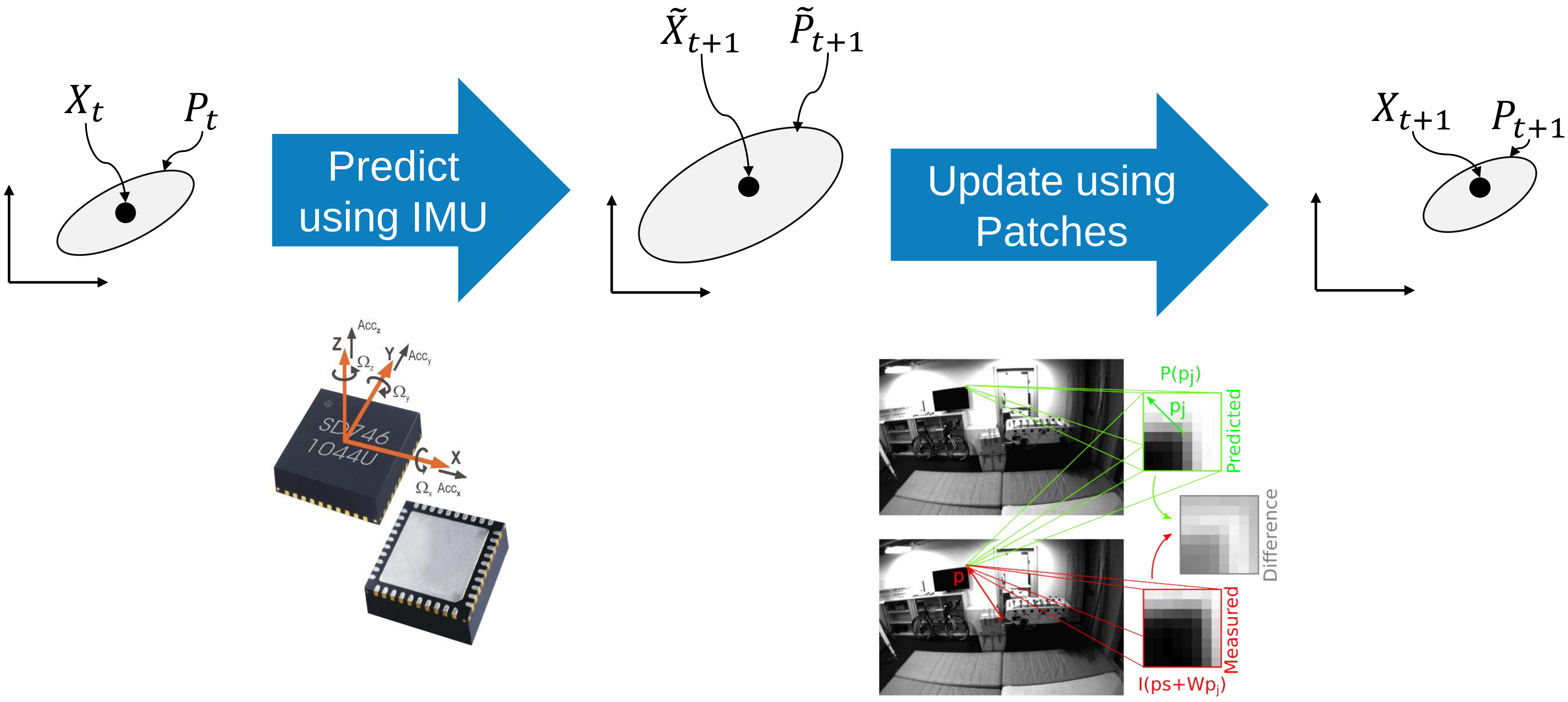
Perception for MAVs | Visual-Inertial I

- Kalman Filter based approach:
  - Predict using IMU
  - Update using Pixels intensity error
  - Robust against aggressive motion and illumination change
  - Computationally efficient
  - Large drift over time



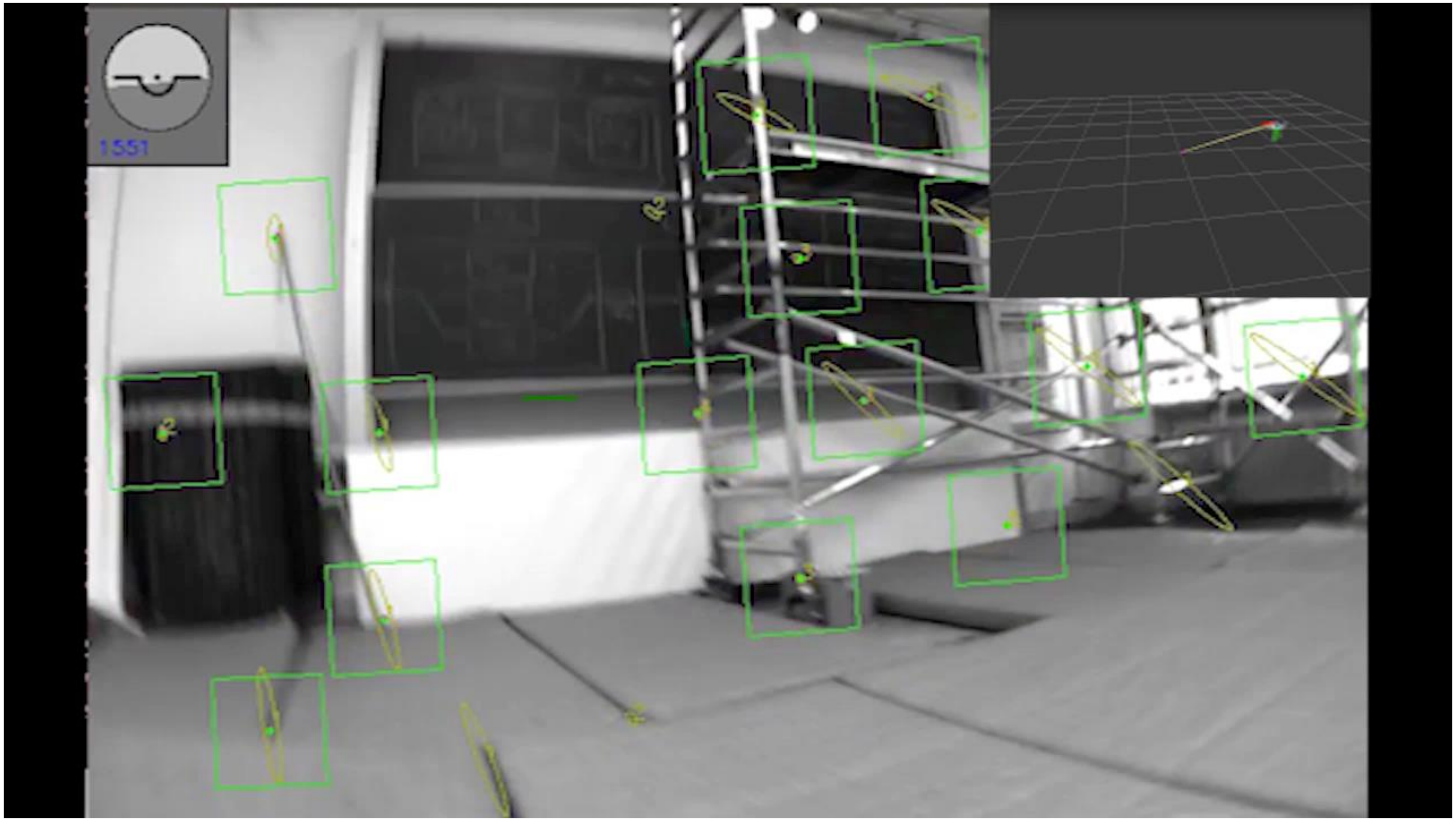
Bloesch, Michael, et al. “Iterated Extended Kalman Filter Based Visual-Inertial Odometry Using Direct Photometric Feedback.” The International Journal of Robotics Research 36, no. 10 (September 2017): 1053–72. https://doi.org/10.1177/0278364917728574.

Perception for MAVs | Visual-Inertial I



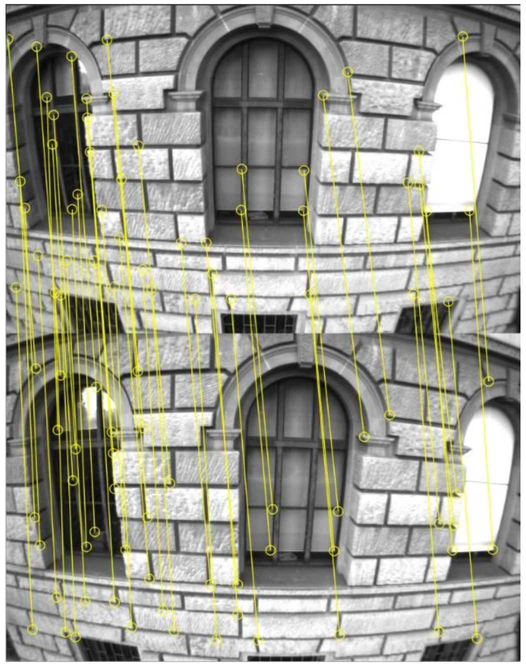
Bloesch, Michael, et al. “Iterated Extended Kalman Filter Based Visual-Inertial Odometry Using Direct Photometric Feedback.” The International Journal of Robotics Research 36, no. 10 (September 2017): 1053–72. https://doi.org/10.1177/0278364917728574.

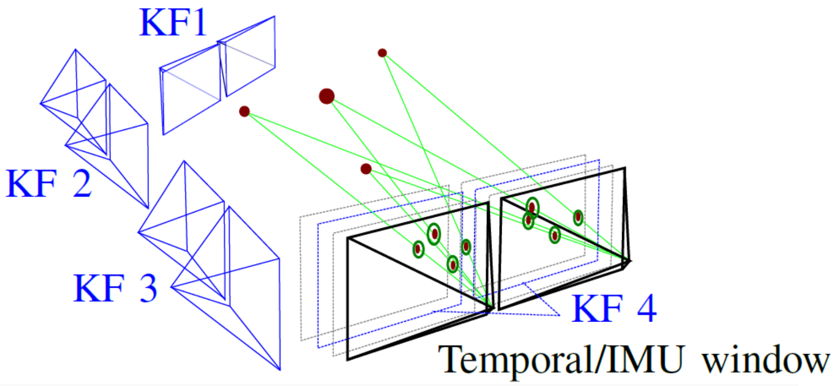
Perception for MAVs | Visual-Inertial I



Perception for MAVs | Visual-Inertial II

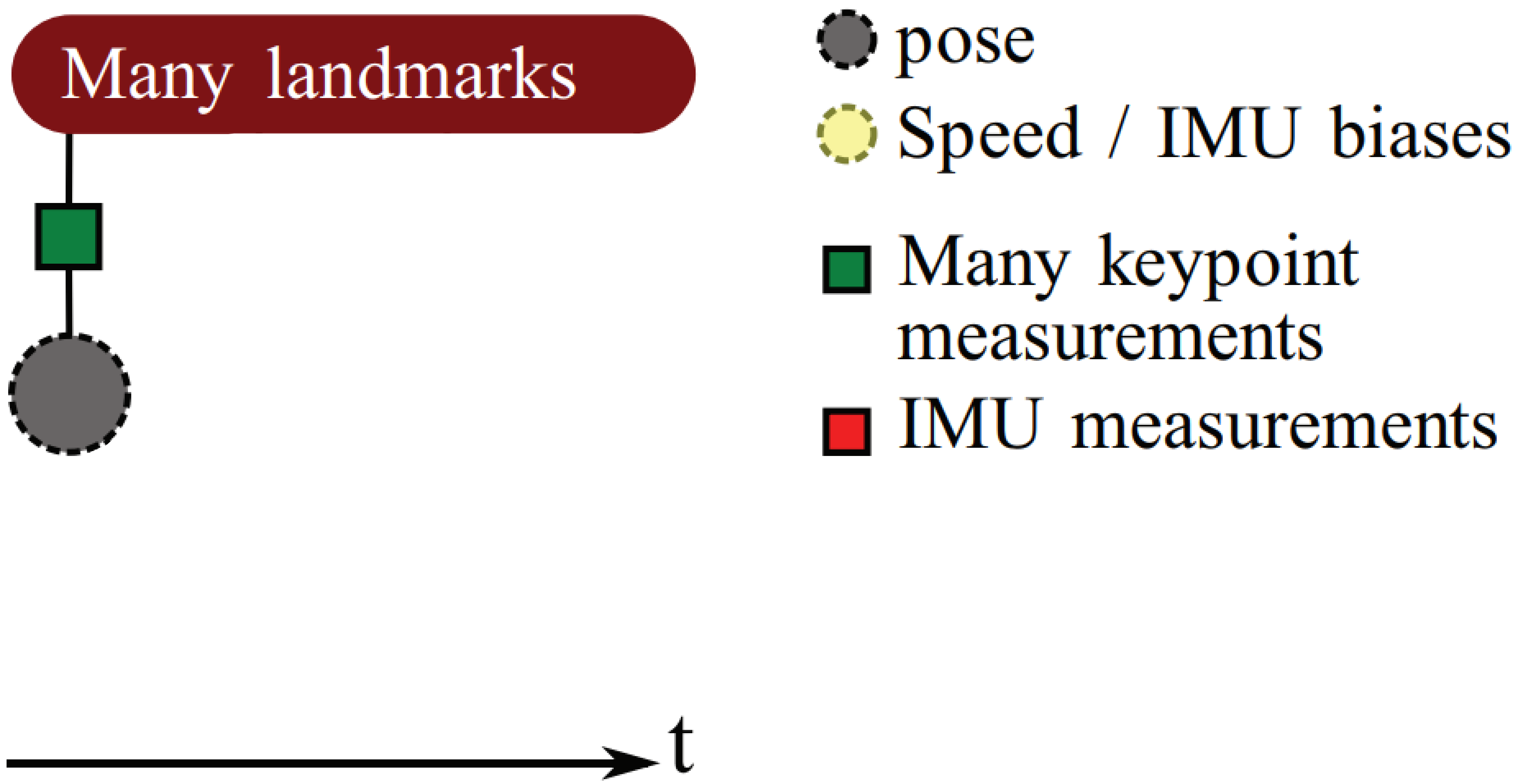
- Keyframe/Graph based approach
  - Extract features from frames, triangulate and match them
  - Cannot keep track of everything solution: use keyframes
  - Optimization based method
  - Small drift over time
  - Computationally expensive





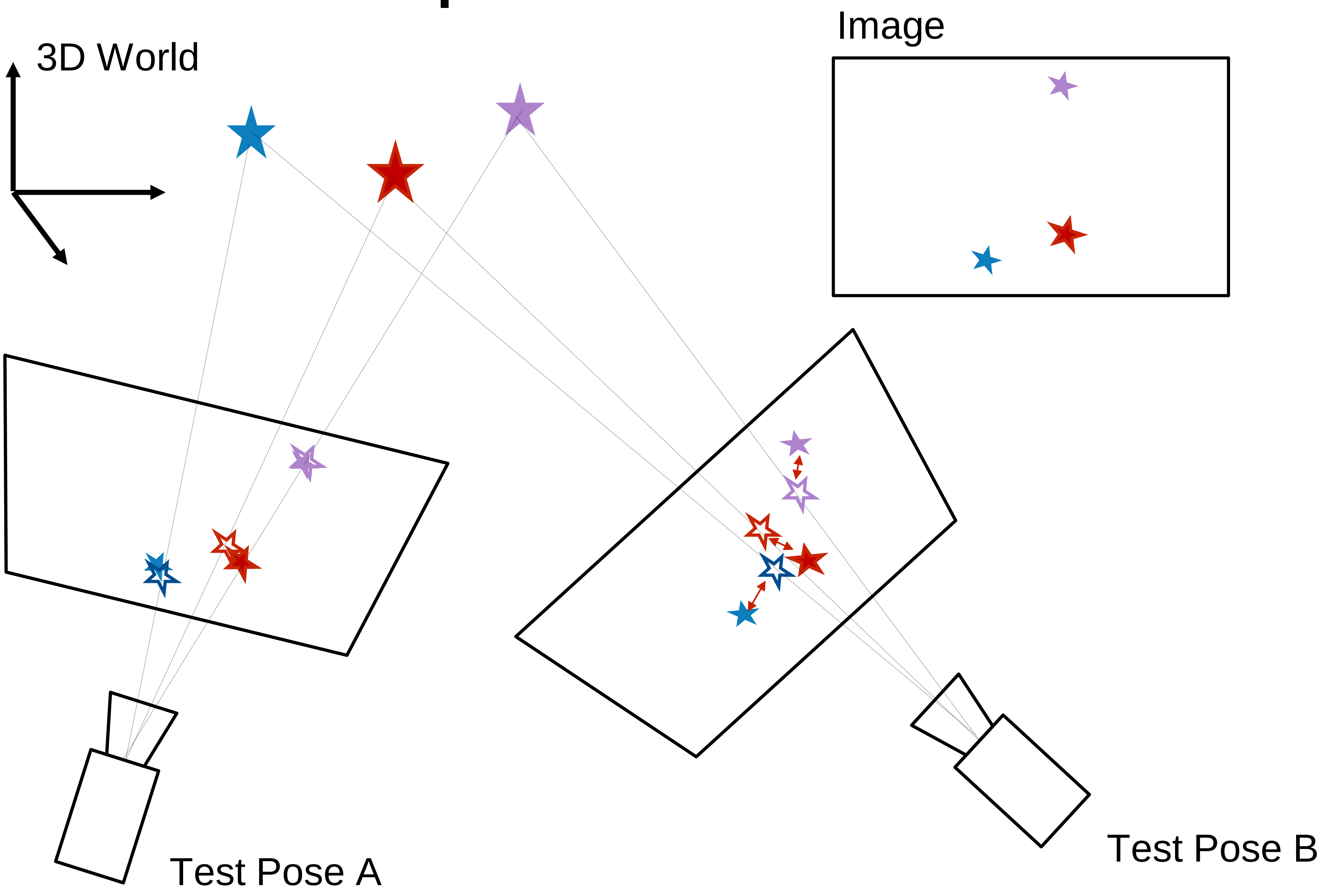
Leutenegger, Stefan, Simon Lynen, Michael Bosse, Roland Siegwart, and Paul Furgale. “Keyframe-Based Visual–Inertial Odometry Using Nonlinear Optimization.” The International Journal of Robotics Research 34, no. 3 (March 2015): 314–34. https://doi.org/10.1177/0278364914554813.

Perception for MAVs | Visual-Inertial II

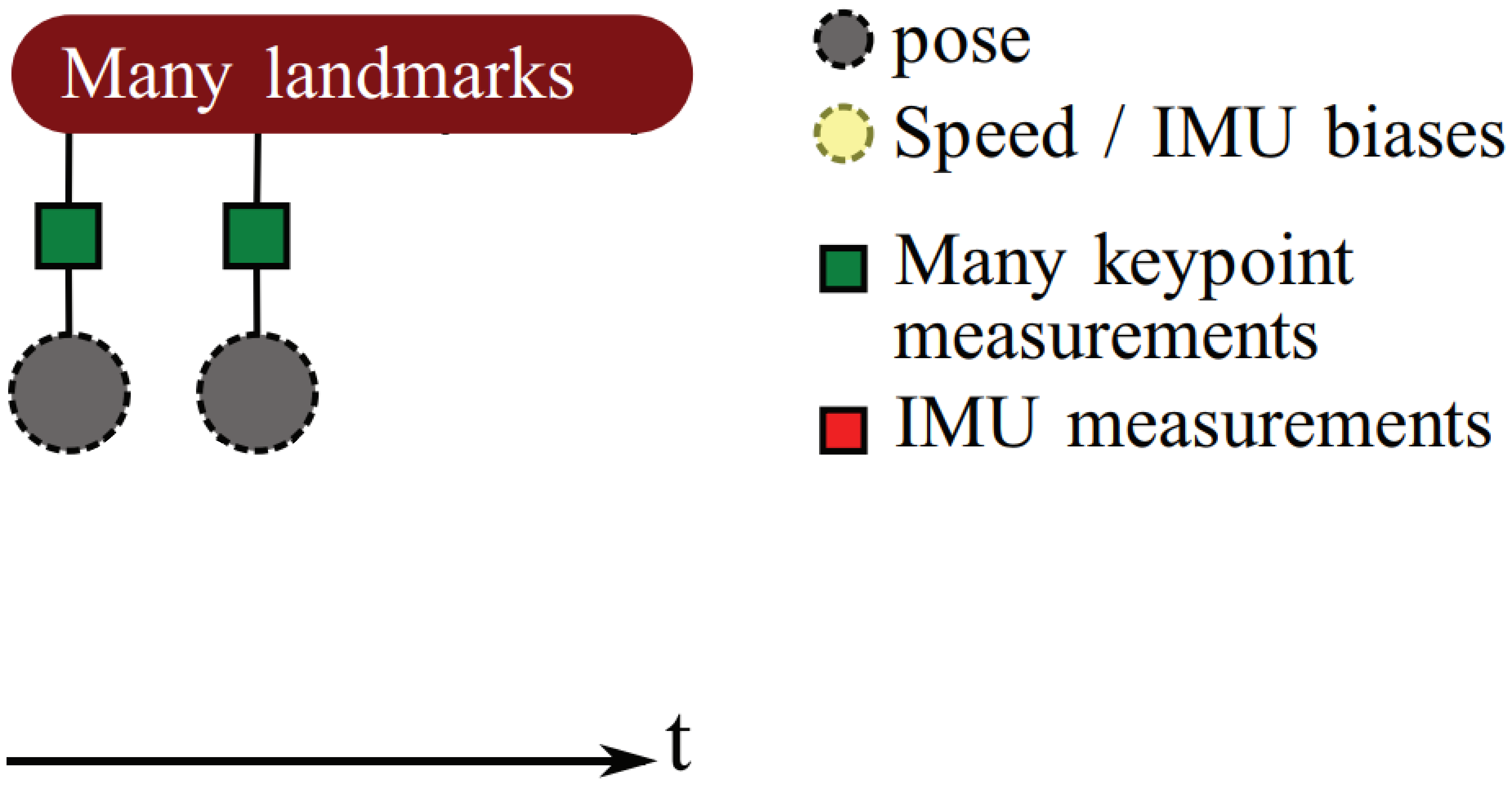


Leutenegger, Stefan, Simon Lynen, Michael Bosse, Roland Siegwart, and Paul Furgale. “Keyframe-Based Visual–Inertial Odometry Using Nonlinear Optimization.” The International Journal of Robotics Research 34, no. 3 (March 2015): 314–34. https://doi.org/10.1177/0278364914554813.

Perception for MAVs | Visual-Inertial II

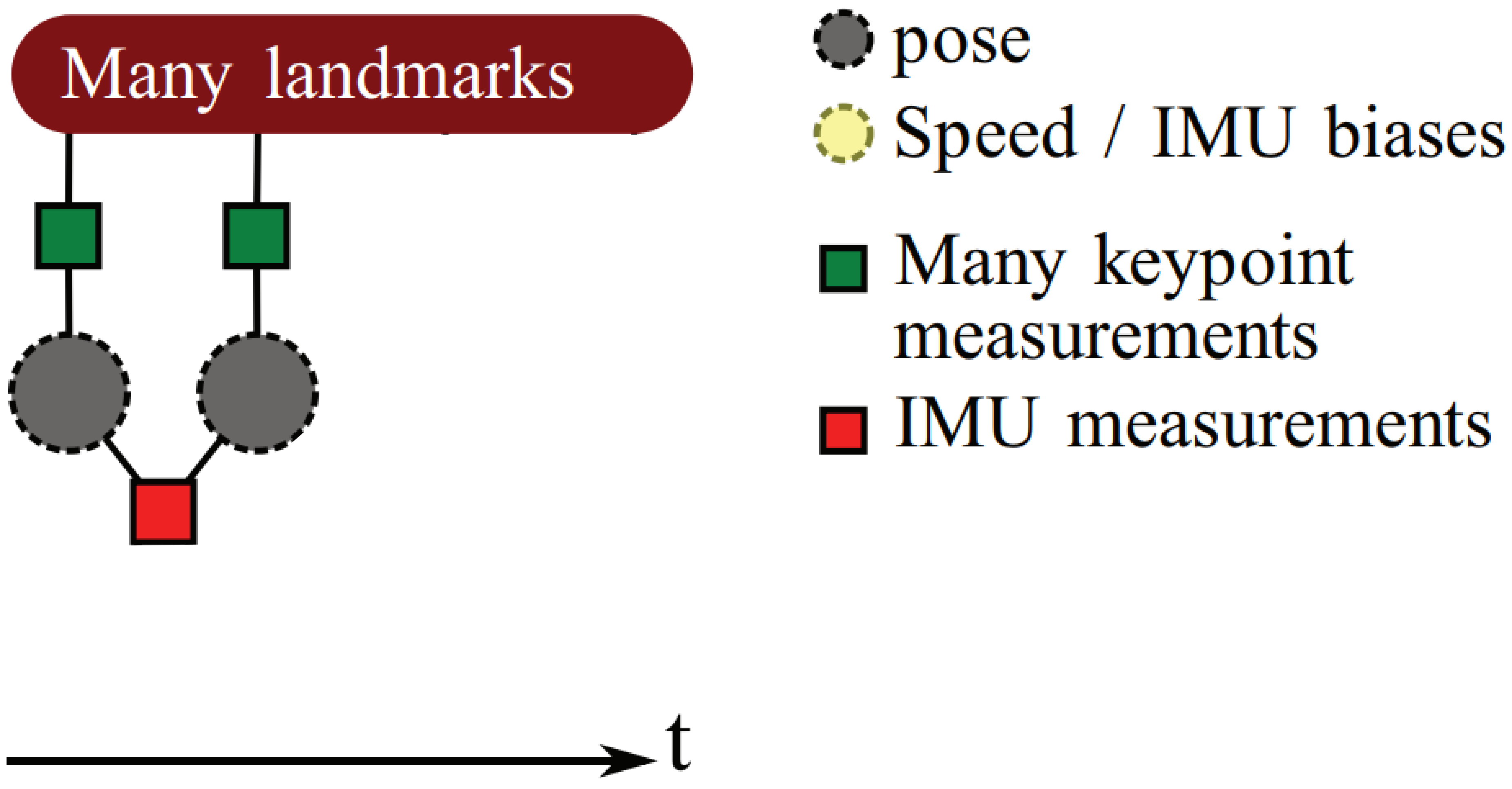


Perception for MAVs | Visual-Inertial II



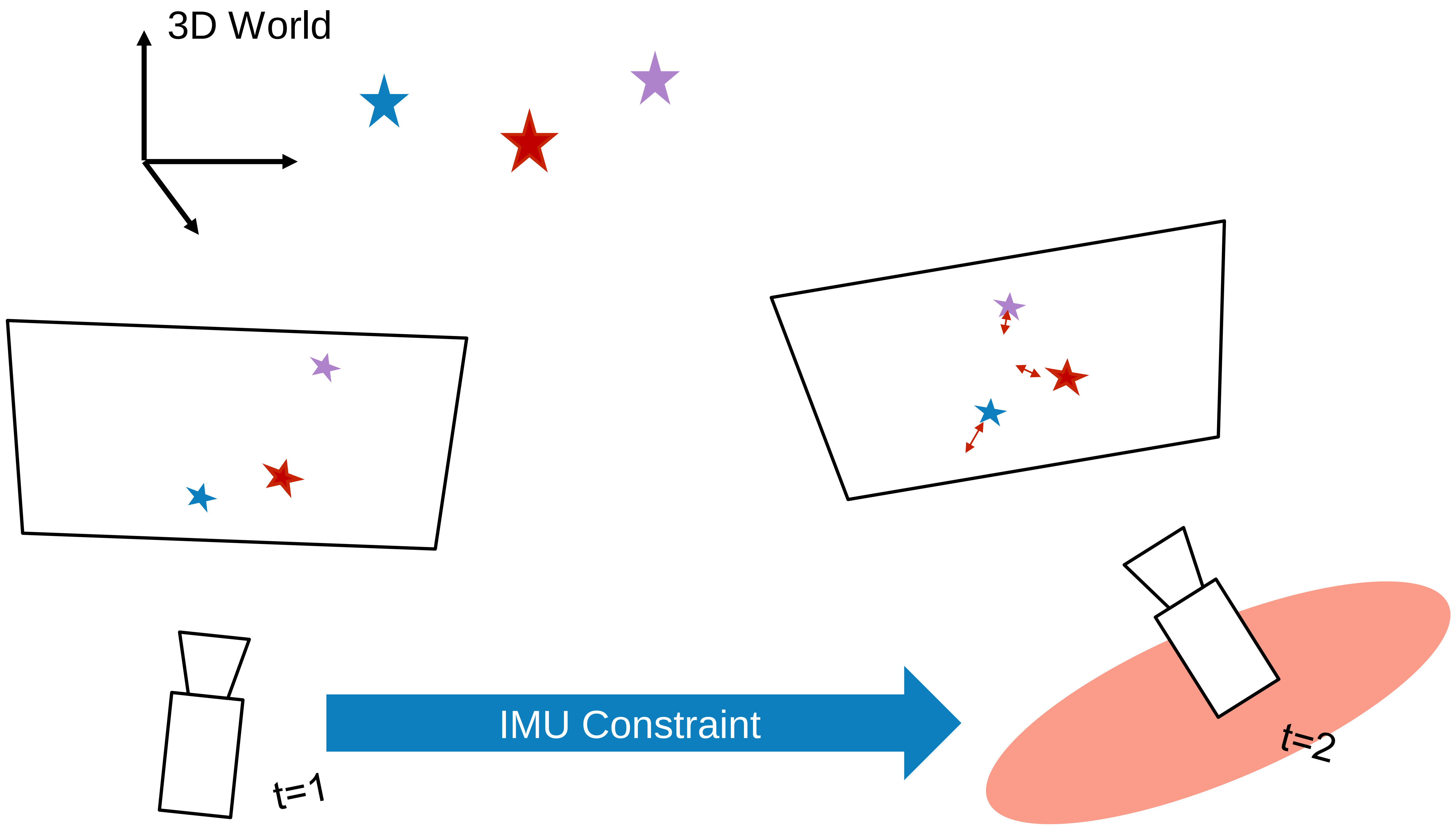
Leutenegger, Stefan, Simon Lynen, Michael Bosse, Roland Siegwart, and Paul Furgale. "Keyframe-Based Visual–Inertial Odometry Using Nonlinear Optimization." The International Journal of Robotics Research 34, no. 3 (March 2015): 314–34. https://doi.org/10.1177/0278364914554813.

Perception for MAVs | Visual-Inertial II

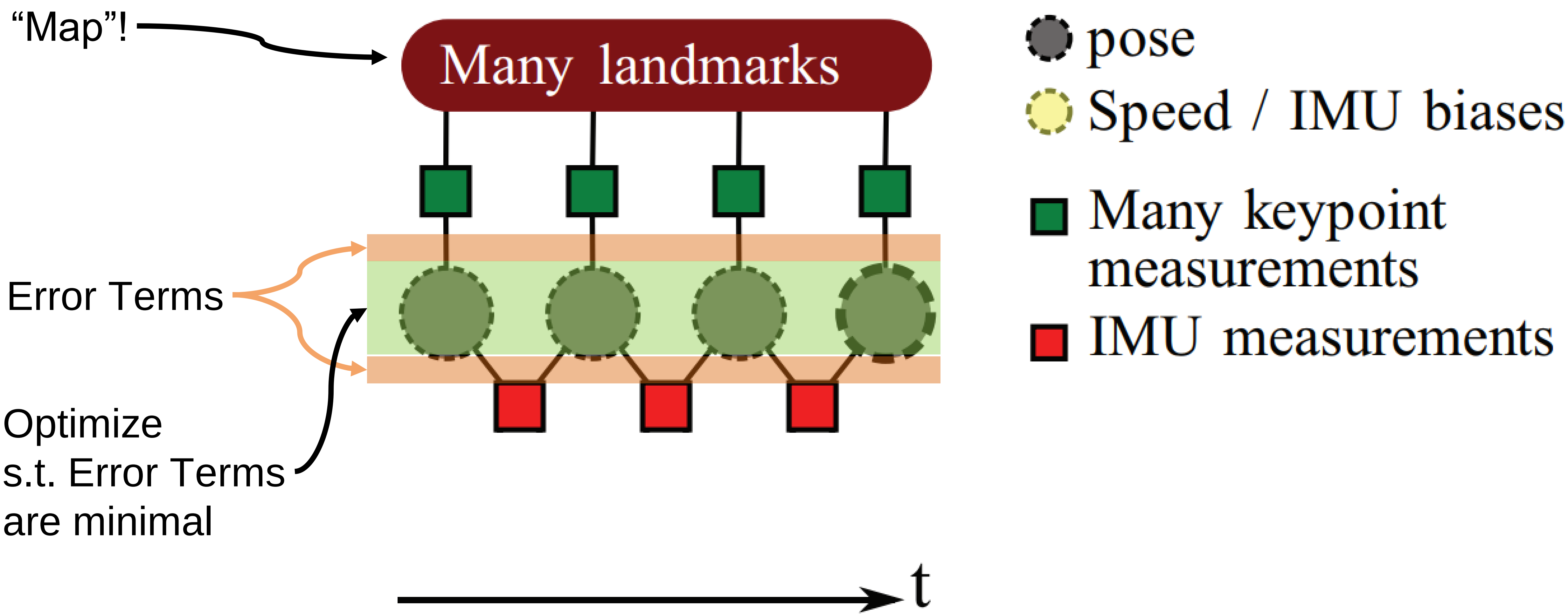


Leutenegger, Stefan, Simon Lynen, Michael Bosse, Roland Siegwart, and Paul Furgale. “Keyframe-Based Visual–Inertial Odometry Using Nonlinear Optimization.” The International Journal of Robotics Research 34, no. 3 (March 2015): 314–34. https://doi.org/10.1177/0278364914554813.

Perception for MAVs | Visual-Inertial II

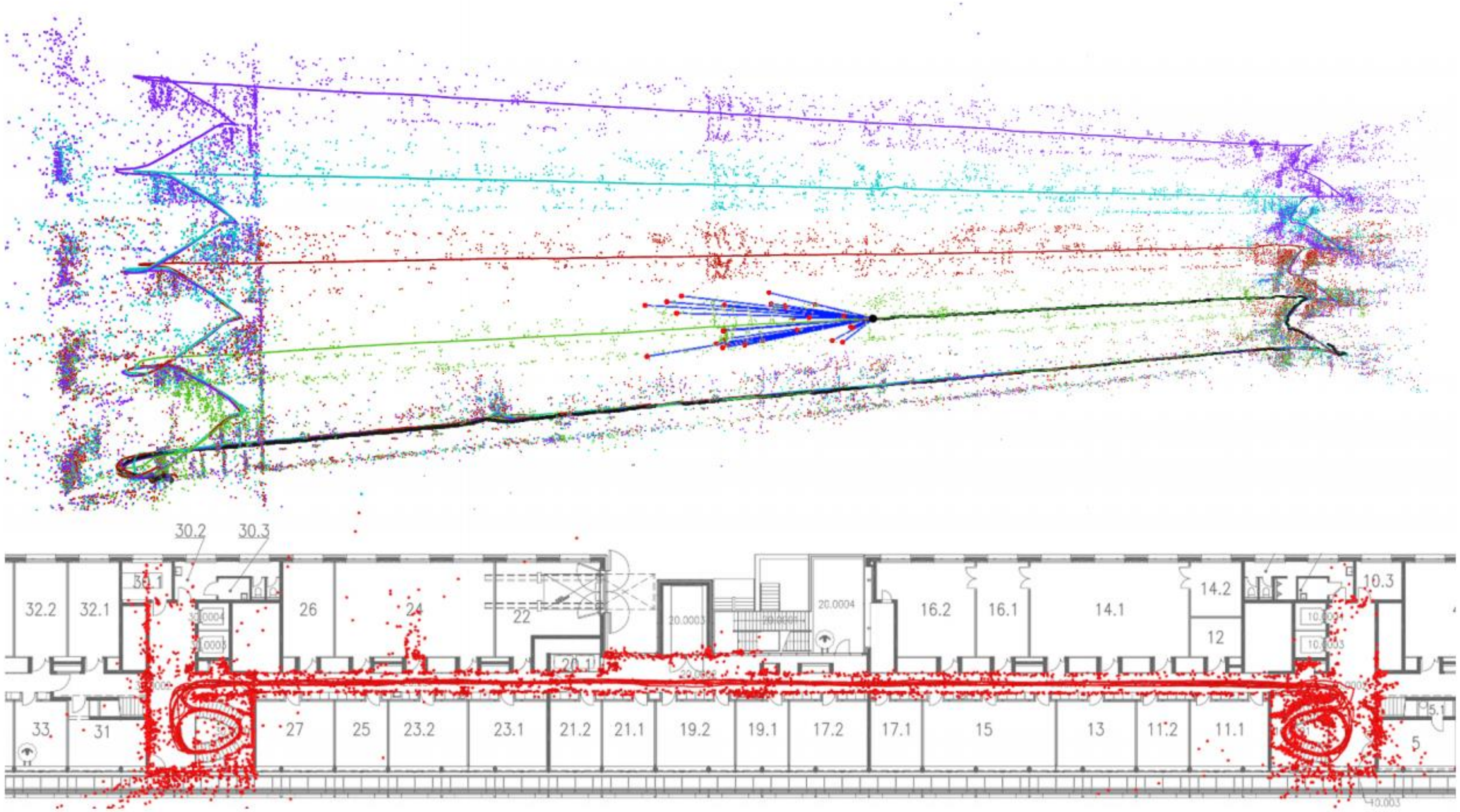


Perception for MAVs | Visual-Inertial II



Leutenegger, Stefan, Simon Lynen, Michael Bosse, Roland Siegwart, and Paul Furgale. “Keyframe-Based Visual–Inertial Odometry Using Nonlinear Optimization.” The International Journal of Robotics Research 34, no. 3 (March 2015): 314–34. https://doi.org/10.1177/0278364914554813.

Perception for MAVs | Sparse Feature Map

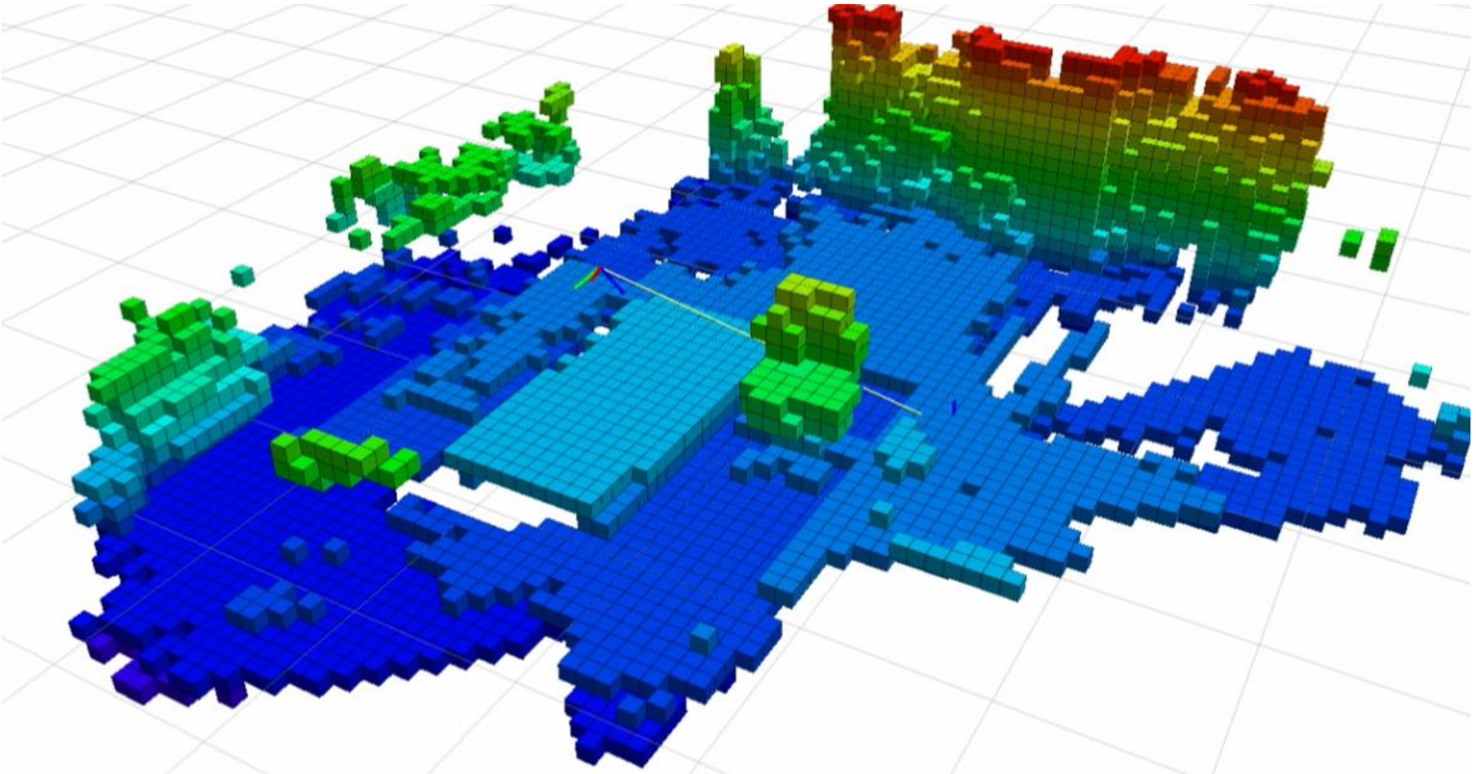


Schneider, Thomas, Marcin Dymczyk, Marius Fehr, Kevin Egger, Simon Lynen, Igor Gilitschenski, and Roland Siegwart. “Maplab: An Open Framework for Research in Visual-Inertial Mapping and Localization.” IEEE Robotics and Automation Letters 3, no. 3 (July 2018): 1418–25. https://doi.org/10.1109/LRA.2018.2800113.

Perception for MAVs | Mapping

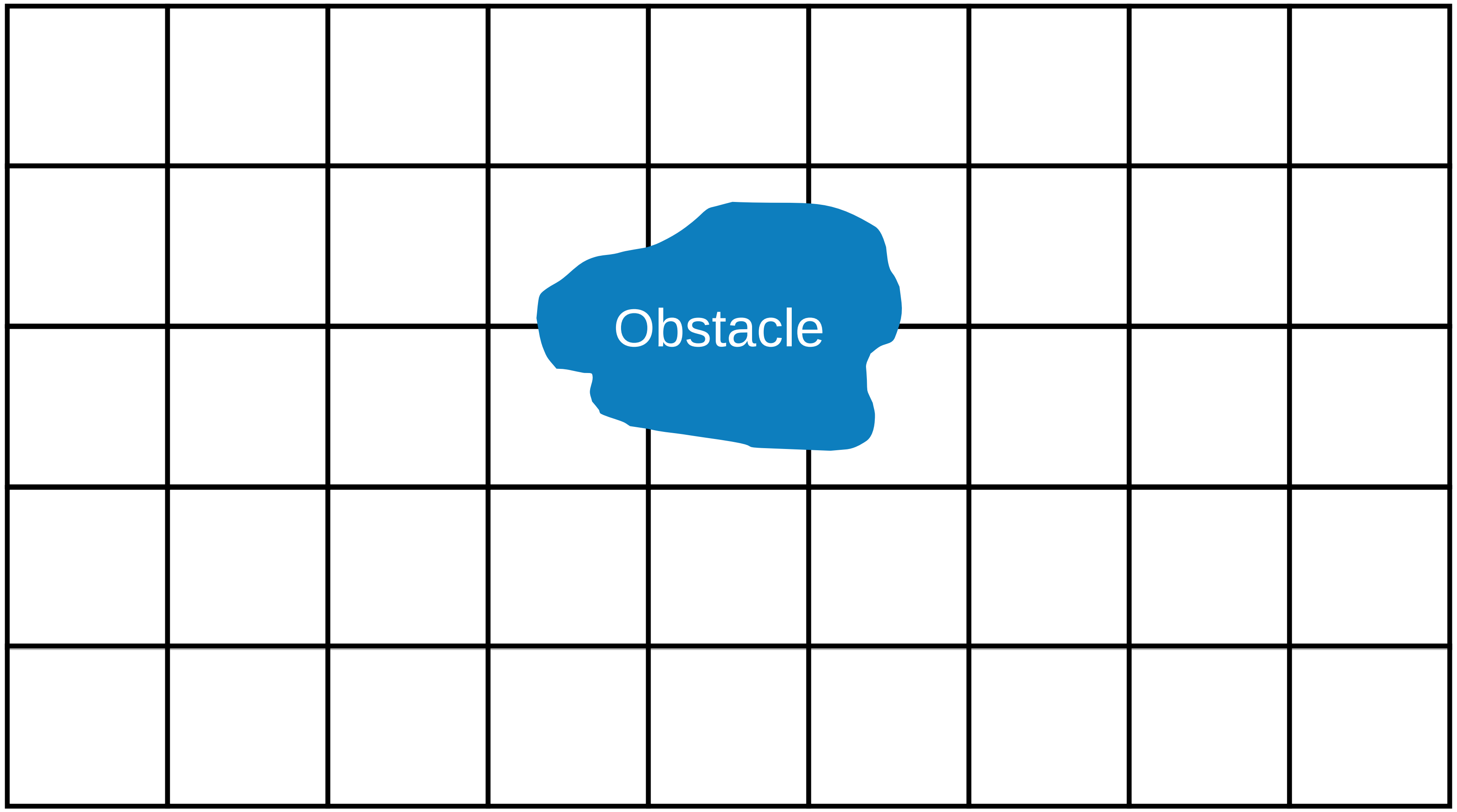
- Problem: Acquire model of physical environment by a moving robot

- Representation
  - Triangulated Features
  - Topological
  - **Volumetric**
  - Semantic (Humans)



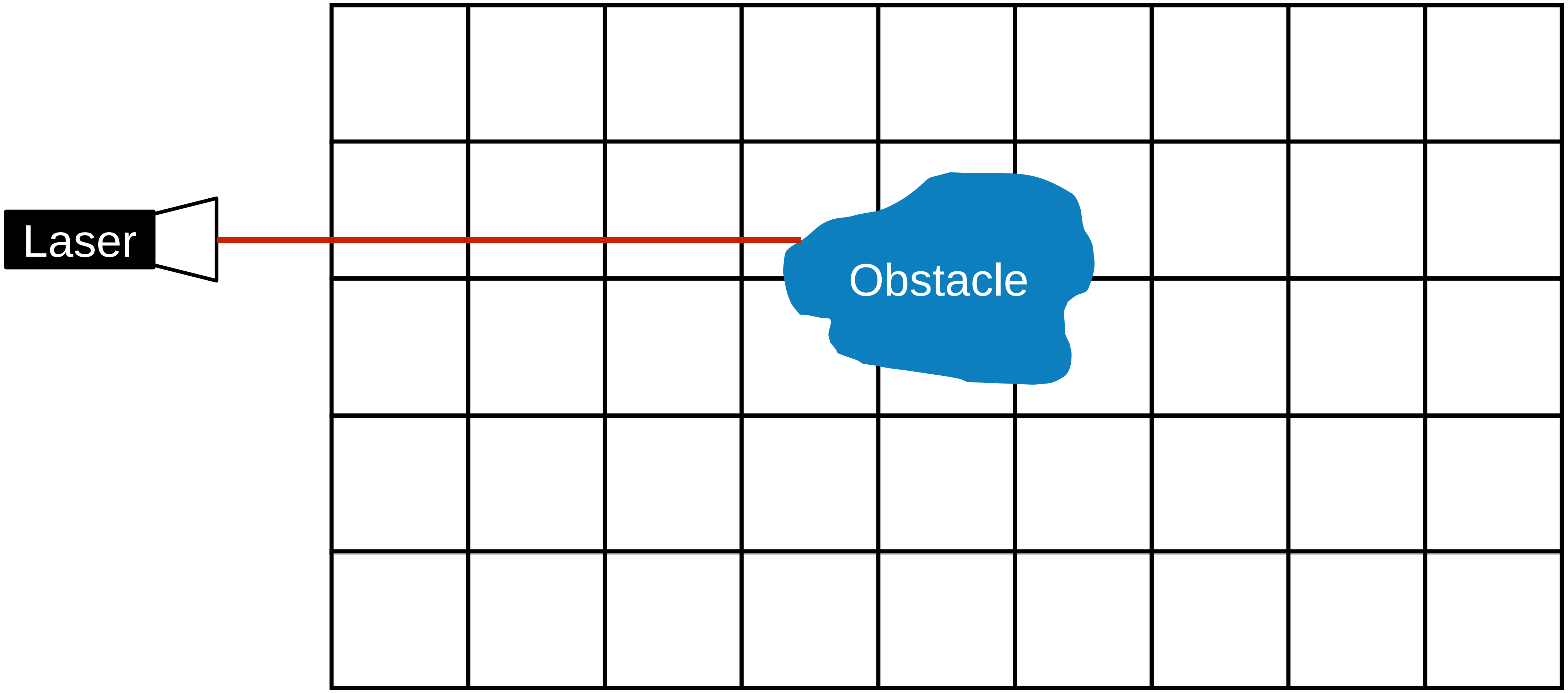
Perception for MAVs | Mapping

- Simplest Occupancy Grid (black = occupied space, white = unknown, green = free)



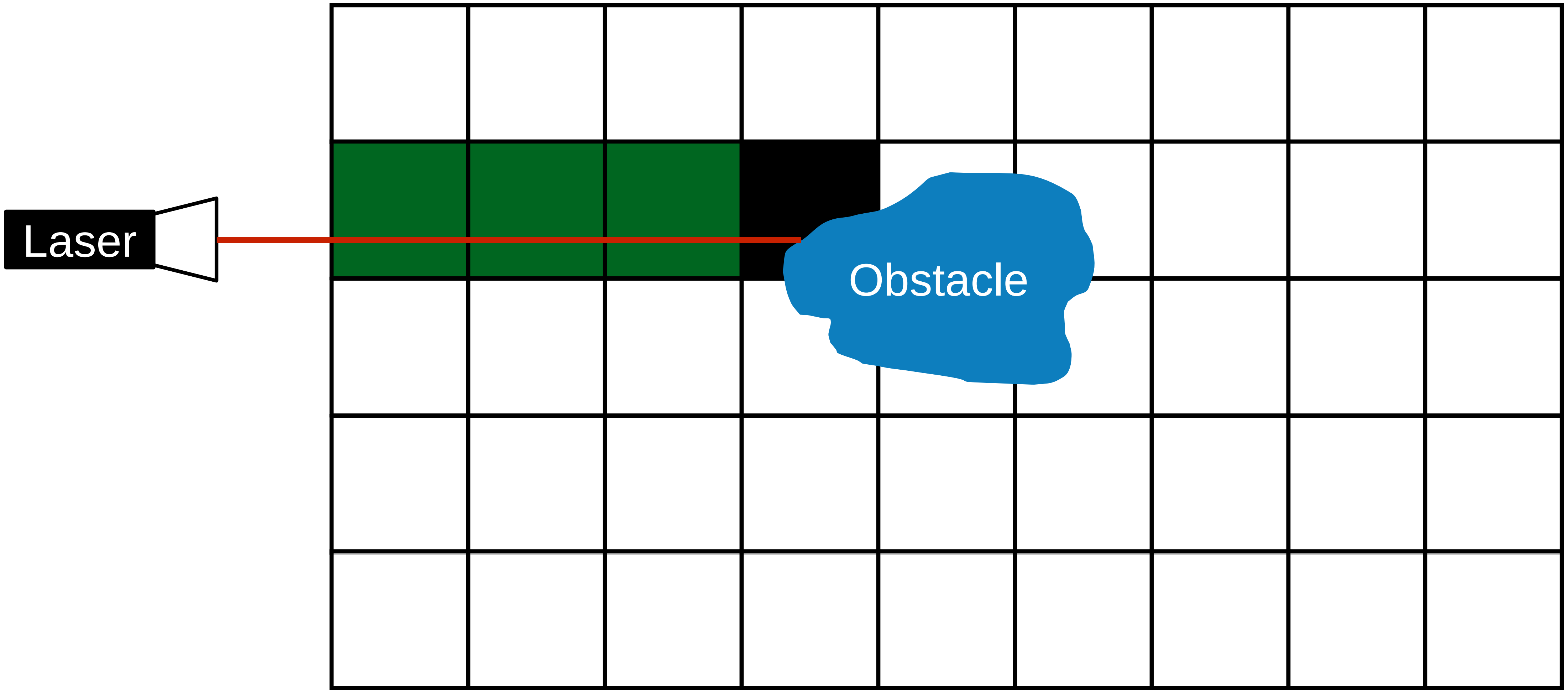
Perception for MAVs | Mapping

- Simplest Occupancy Grid (black = occupied space, white = unknown, green = free)



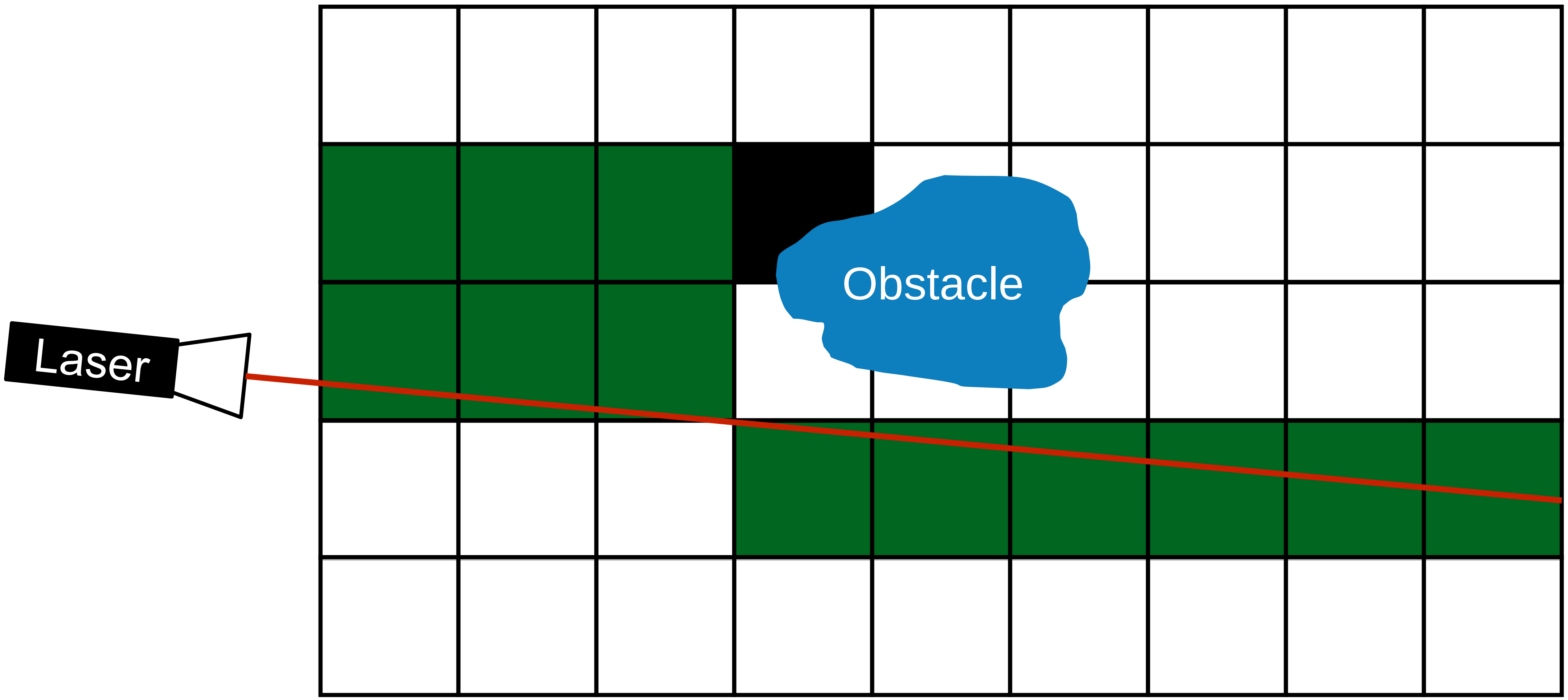
Perception for MAVs | Mapping

- Simplest Occupancy Grid (black = occupied space, white = unknown, green = free)



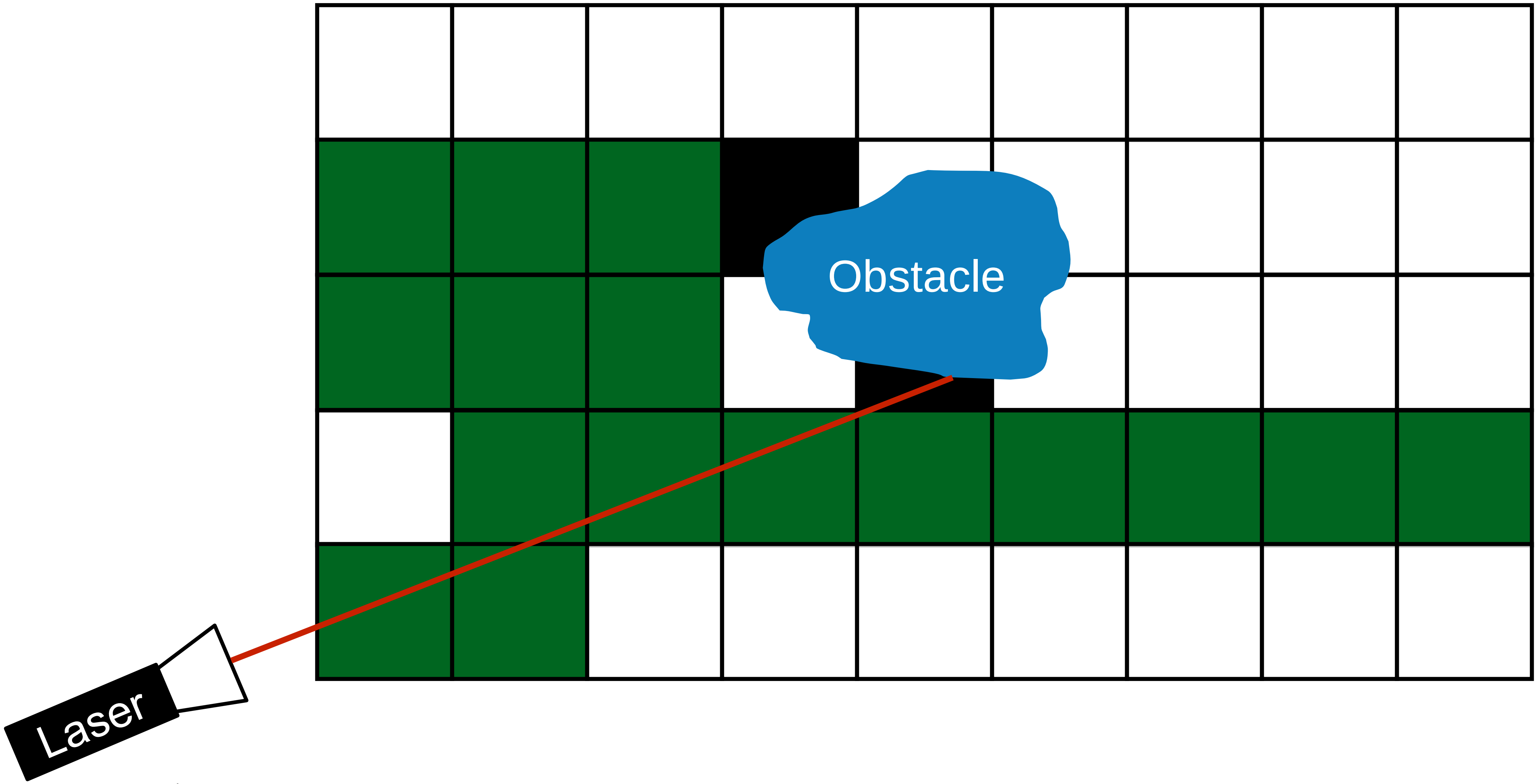
Perception for MAVs | Mapping

- Simplest Occupancy Grid (black = occupied space, white = unknown, green = free)



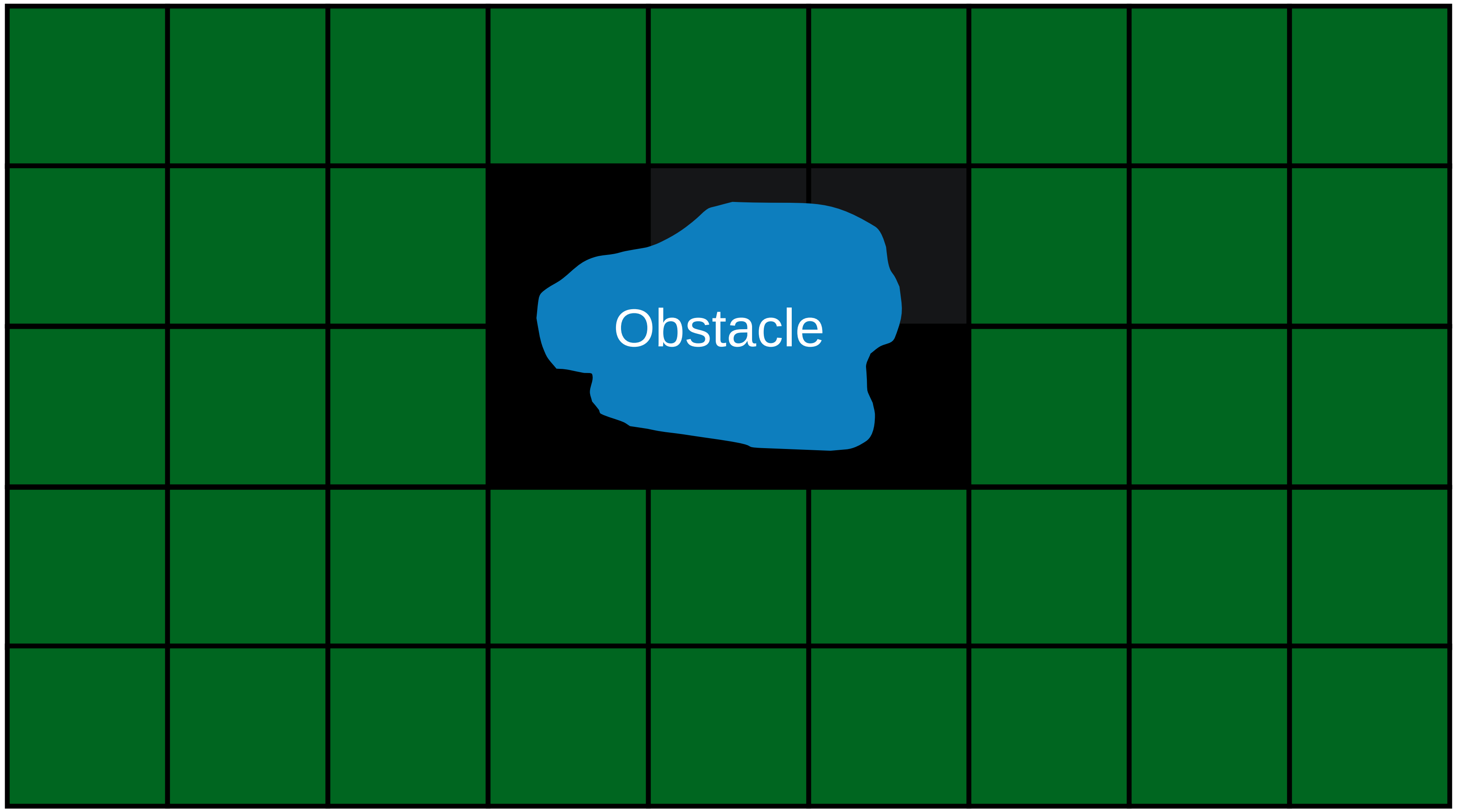
Perception for MAVs | Mapping

- Simplest Occupancy Grid (black = occupied space, white = unknown, green = free)



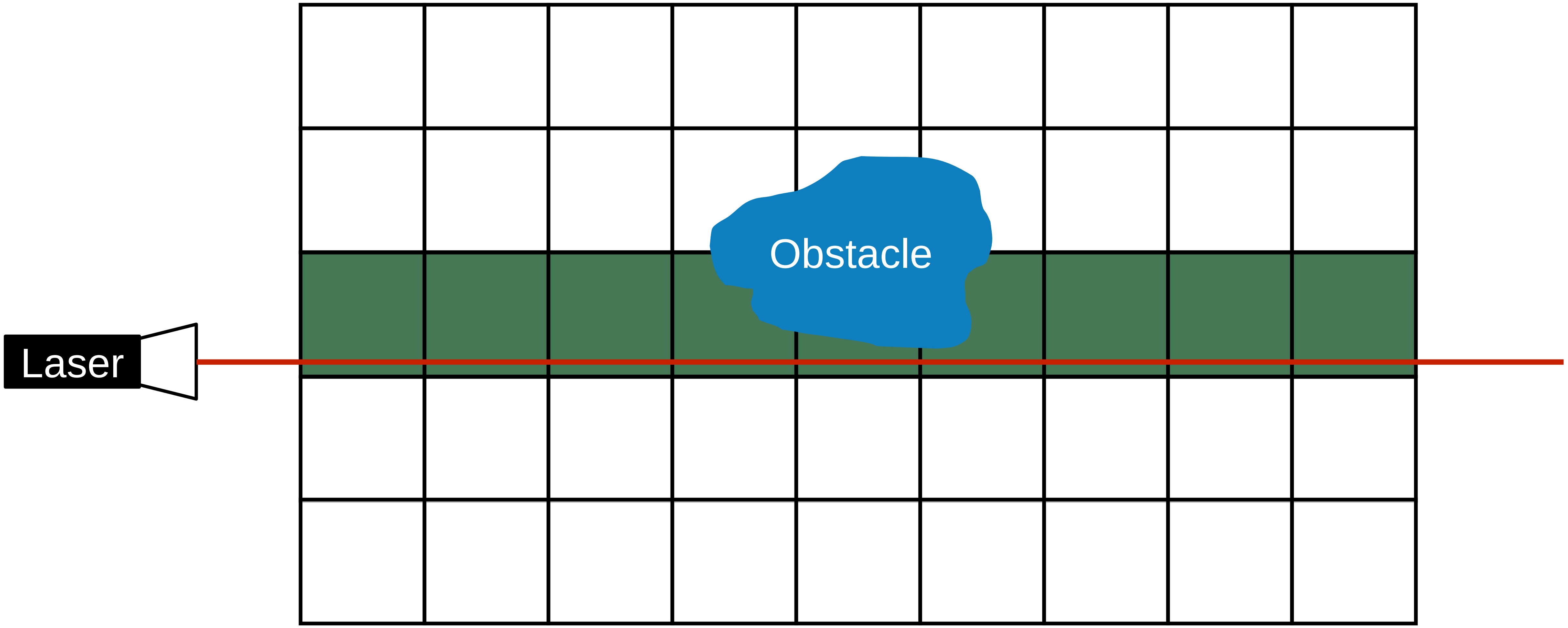
Perception for MAVs | Mapping

- Simplest Occupancy Grid (black = occupied space, white = unknown, green = free)



Perception for MAVs | Mapping

- Simplest Occupancy Grid (black = occupied space, white = unknown, green = free)

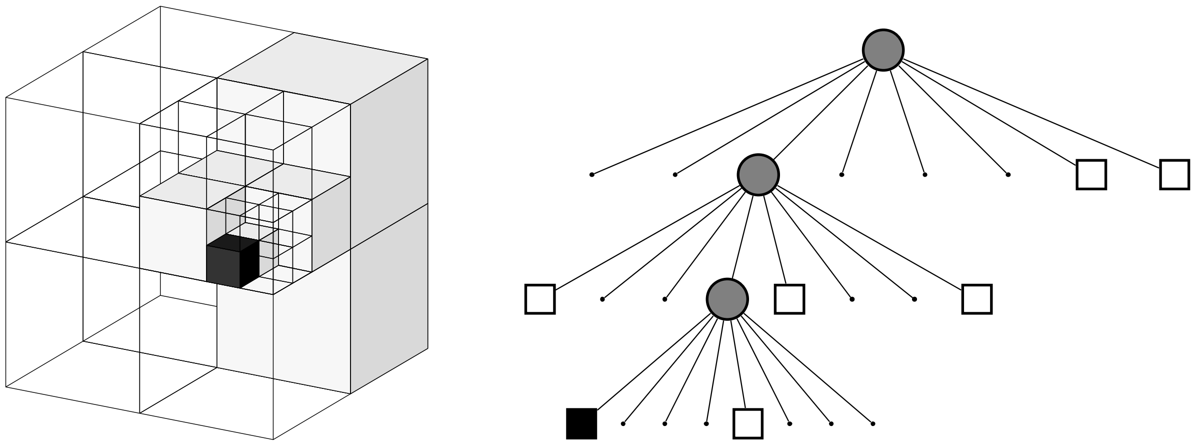


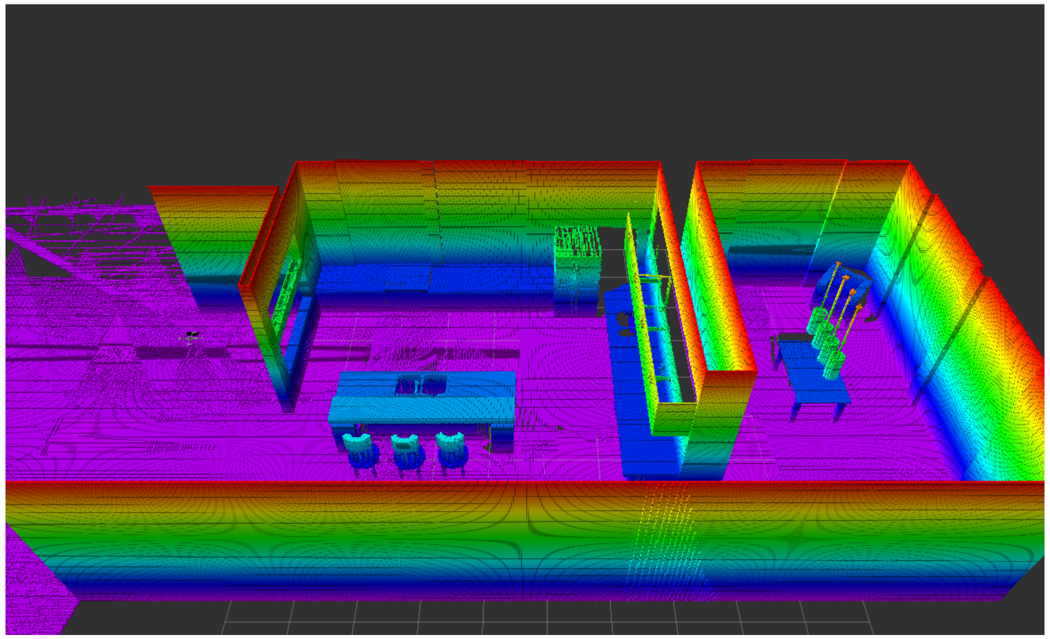
Perception for MAVs | Mapping

- Issues with naive occupancy grid
  - Fixed resolution – either very coarse or very memory inefficient
  - Sensor noise not accounted for – not probabilistic
  - No gradient for planning
    - If we plan through an obstacle, which way do we need to adjust our plan?

Perception for MAVs | Mapping

- OctoMap (An Efficient Probabilistic 3D Mapping Framework):
  - Grid can be subdivided (octree)
  - Uses probabilistic occupancy estimation
  - Able to model ***Free***, ***Occupied*** and ***Unknown*** space
  - Memory efficient for large scale environment
  - Assumes known robot pose

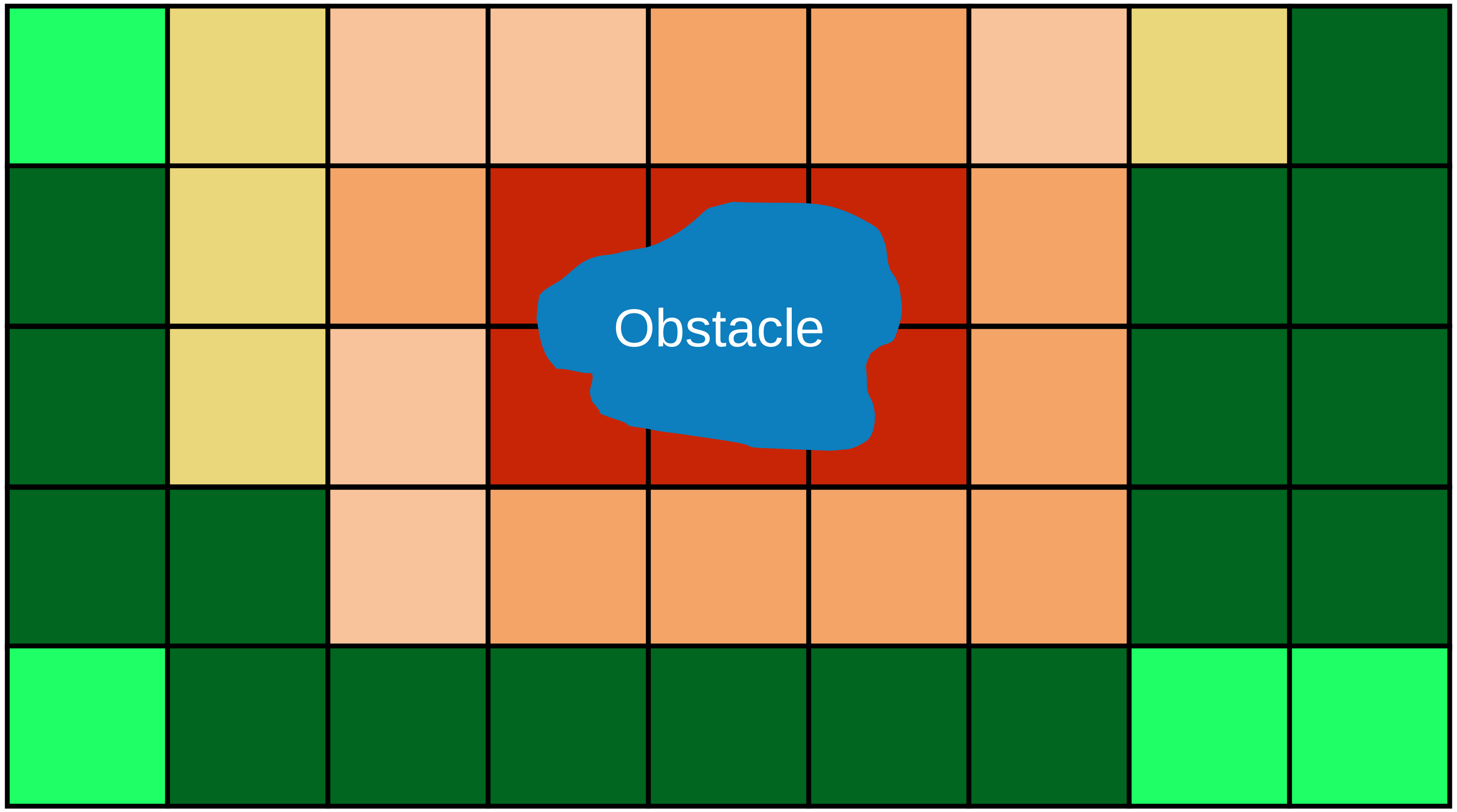




Hornung, Armin, Kai M. Wurm, Maren Bennewitz, Cyrill Stachniss, and Wolfram Burgard. "OctoMap: An Efficient Probabilistic 3D Mapping Framework Based on Octrees." Autonomous Robots 34, no. 3 (April 2013): 189–206. https://doi.org/10.1007/s10514-012-9321-0.

Perception for MAVs | Mapping

- What if we include distances? E.g. Center of cell to obstacle? (greener = further away from obstacle)



Perception for MAVs | Mapping

- Voxblox
  - Volumetric mapping library
  - Runs realtime on board
  - Constructs occupancy grid and distance fields (ESDF/TSDF)
  - See paper for how they are constructed

Oleynikova, Helen, Zachary Taylor, Marius Fehr, Roland Siegwart, and Juan Nieto. “Voxblox: Incremental 3D Euclidean Signed Distance Fields for on-Board MAV Planning,” 1366–73. IEEE, 2017. https://doi.org/10.1109/IROS.2017.8202315.

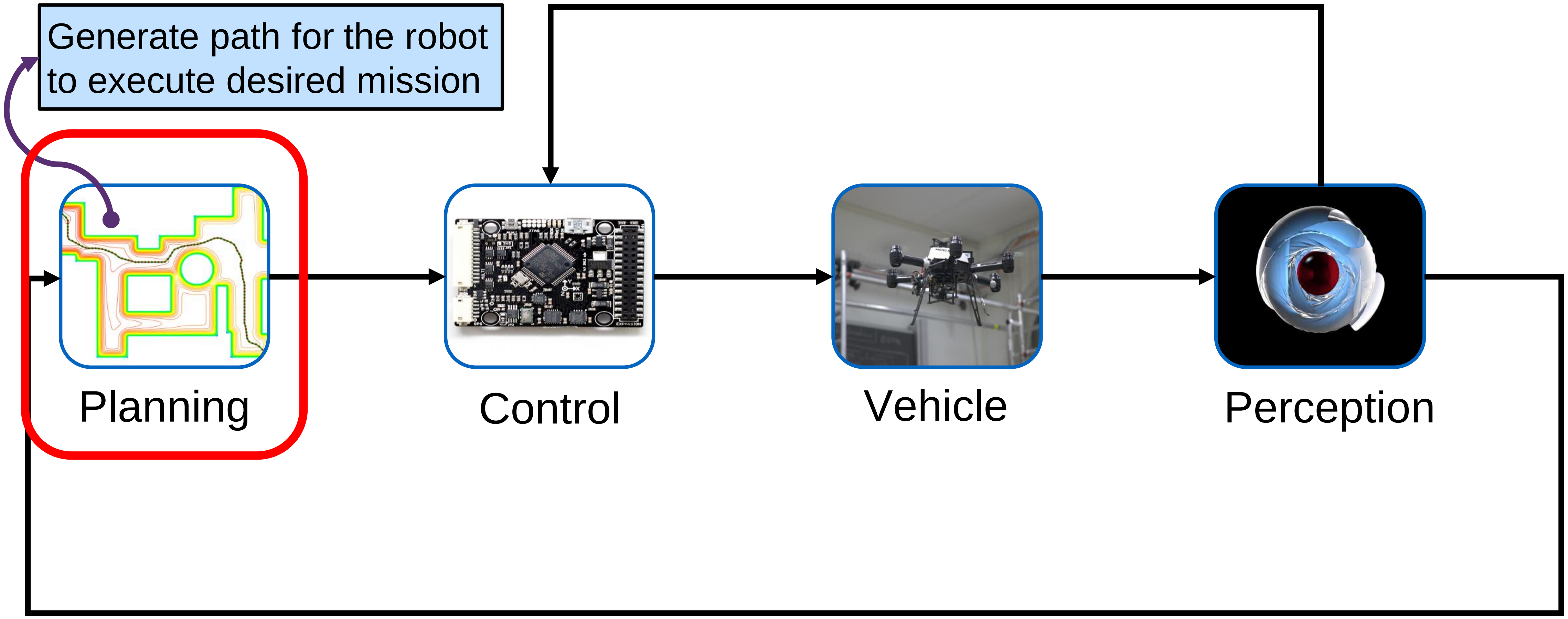
Perception for MAVs | Mapping

Oleynikova, Helen, Zachary Taylor, Marius Fehr, Roland Siegwart, and Juan Nieto. "Voxblox: Incremental 3D Euclidean Signed Distance Fields for on-Board MAV Planning," 1366–73. IEEE, 2017. https://doi.org/10.1109/IROS.2017.8202315.

Micro Aerial Vehicle Autonomy | Planning

Autonomous robot is an intelligent machine able to perform tasks without explicit human intervention

- Basic components to achieve autonomy:



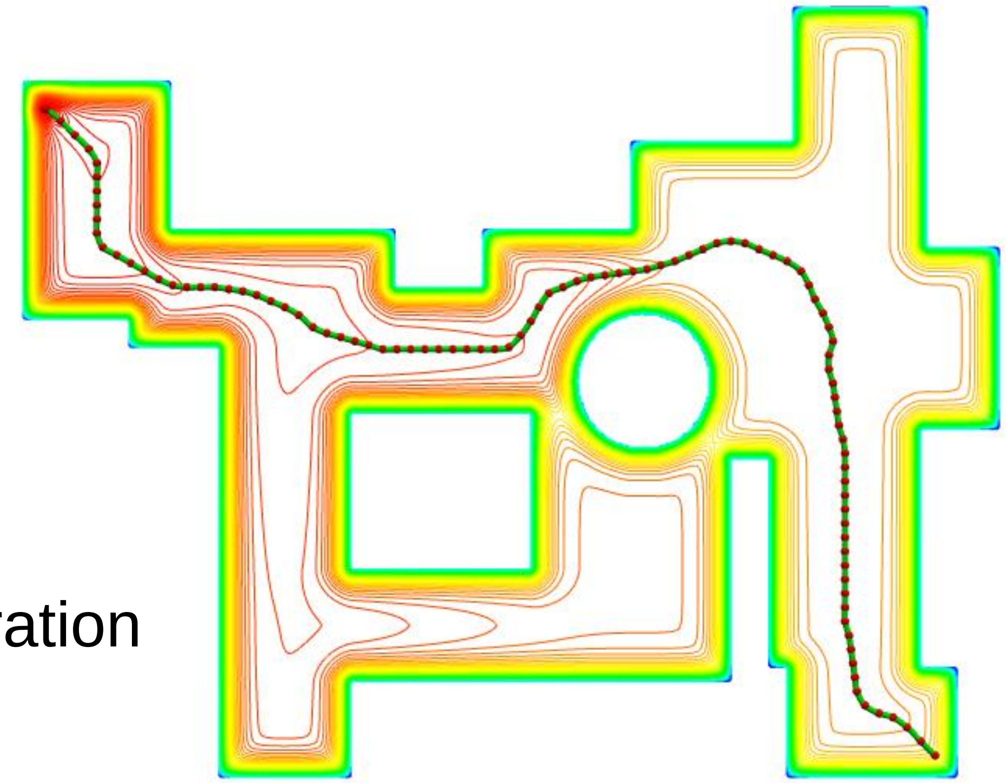
Motion Planning for Micro Aerial Vehicles

Motion Planning for MAVs | Introduction

What is motion planning?

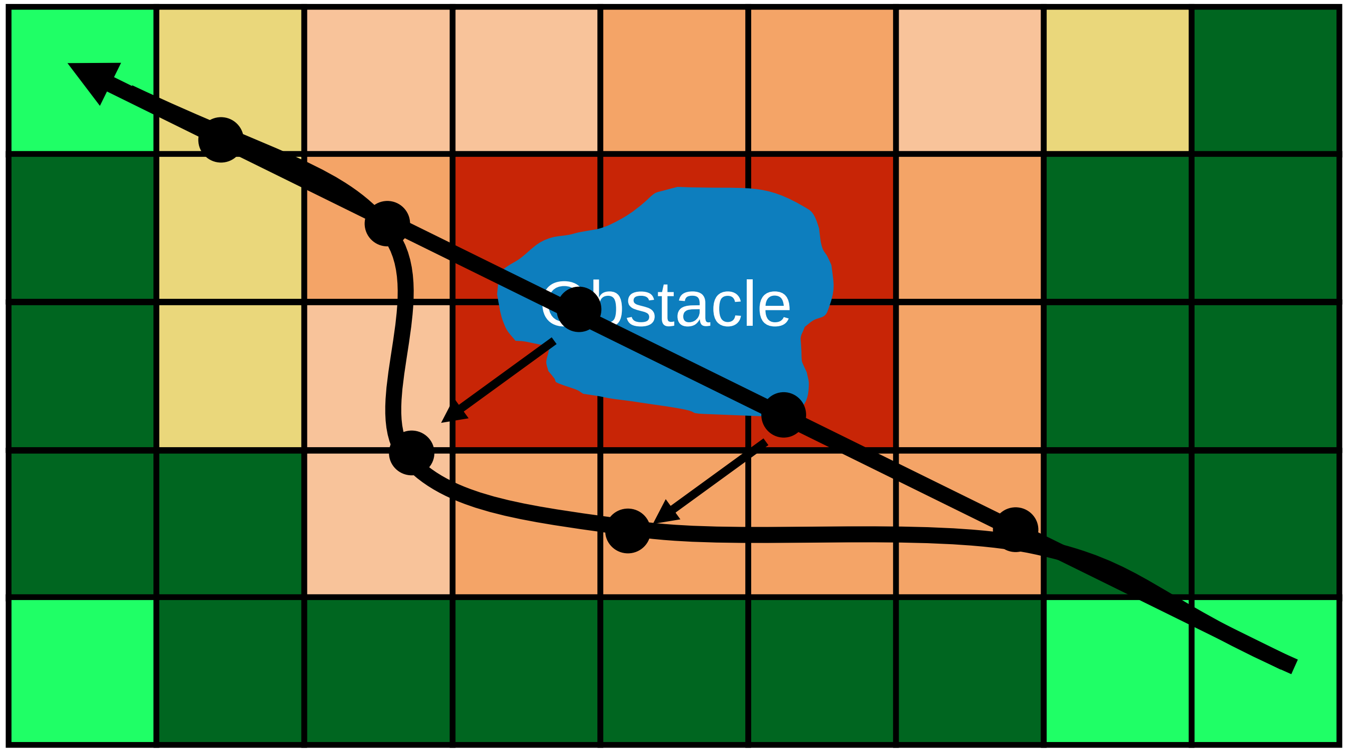
Compute a continuous sequence of **collision-free** robot configuration connecting **initial and goal configurations**

- Motion Planning algorithm input:
  - Initial and goal configurations
  - Environment model
  - Robot kinematics and geometry
- Motion Planning algorithm output:
  - Collision-free path from initial to goal configuration



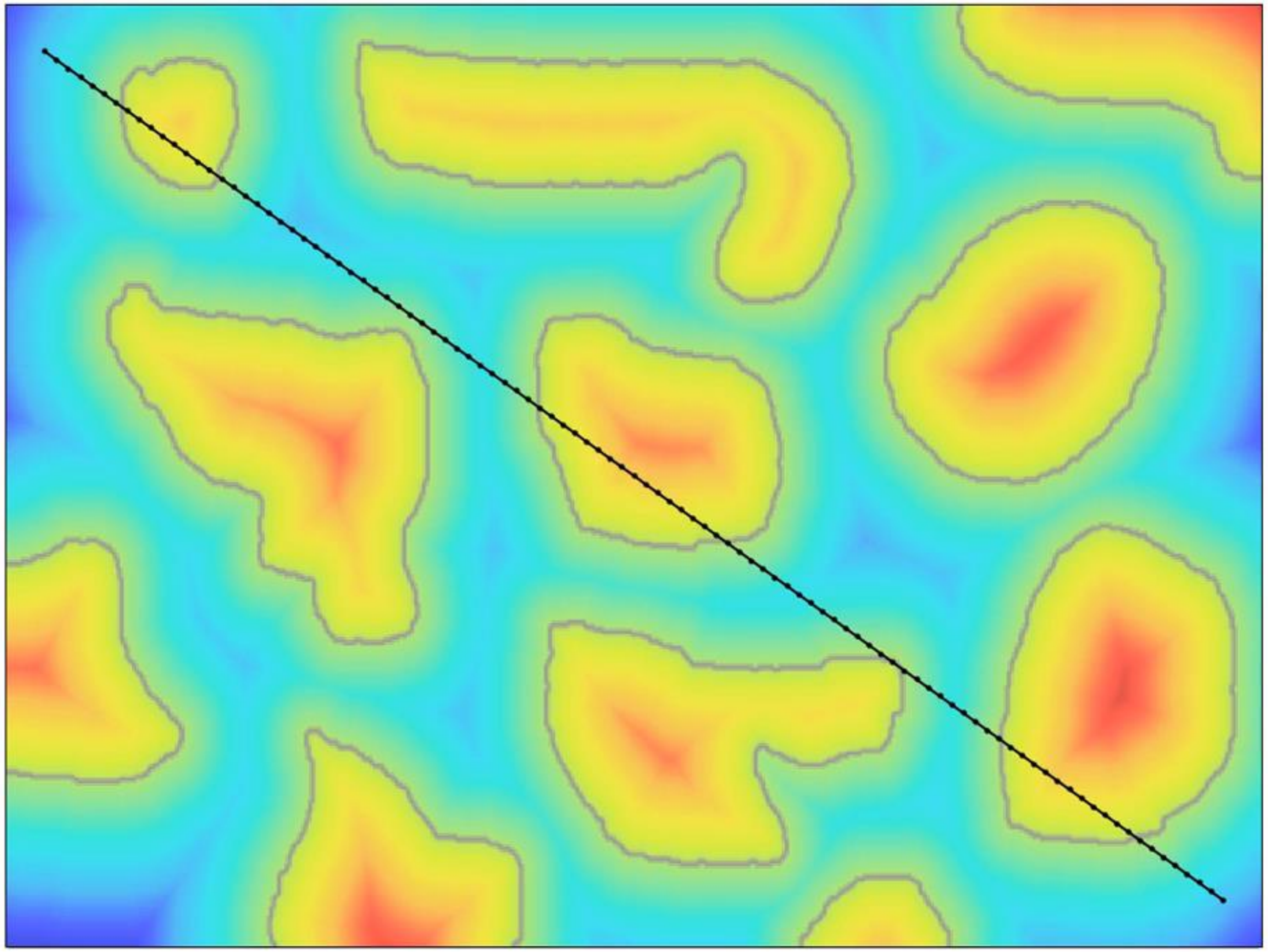
Motion Planning for MAVs | Example CHOMP

- Plan direct path
- Move control points away along gradient if collision
- Keep smoothness!
- Iterate



Algorithm: Matthew Zucker, et. al. “CHOMP: Covariant Hamiltonian Optimization for Motion Planning“
Journal Article, Carnegie Mellon University, International Journal of Robotics Research, May, 2013

Motion Planning for MAVs | Example CHOMP



Visualization: Chris Dellin, https://www.youtube.com/watch?v=41UYgGHGSSo
Algorithm: Matthew Zucker, et. al. “CHOMP: Covariant Hamiltonian Optimization for Motion Planning“
Journal Article, Carnegie Mellon University, International Journal of Robotics Research, May, 2013

Motion Planning for MAVs | Collision Avoidance