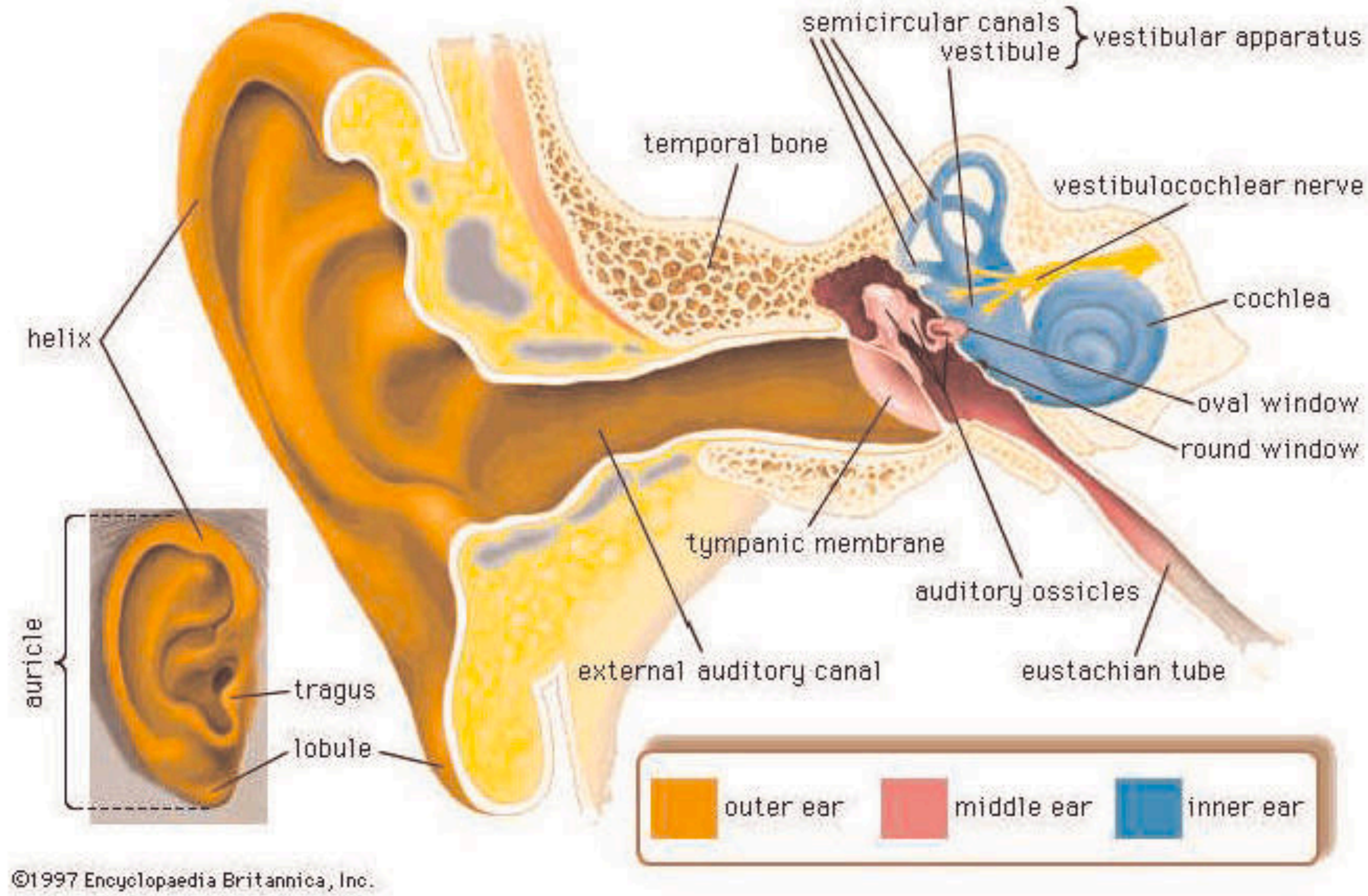
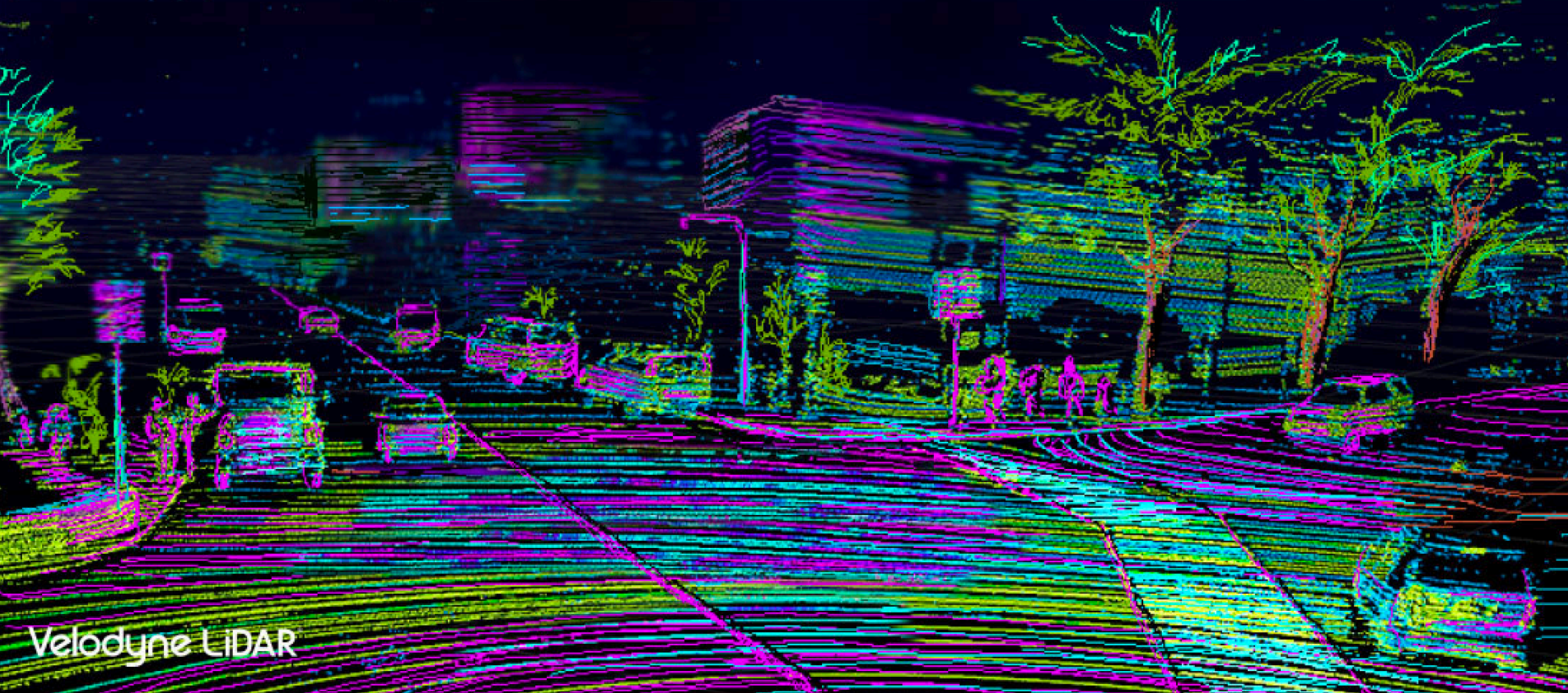


Fig. 2. Human ear (taken with permission from Encyclopaedia Britannica 2001).

16.485: VNAV - Visual Navigation for Autonomous Vehicles

Luca Carlone

Lecture 25-26: Advanced Topics - Beyond Cameras



16.485: VNAV - Visual Navigation for Autonomous Vehicles

Luca Carlone

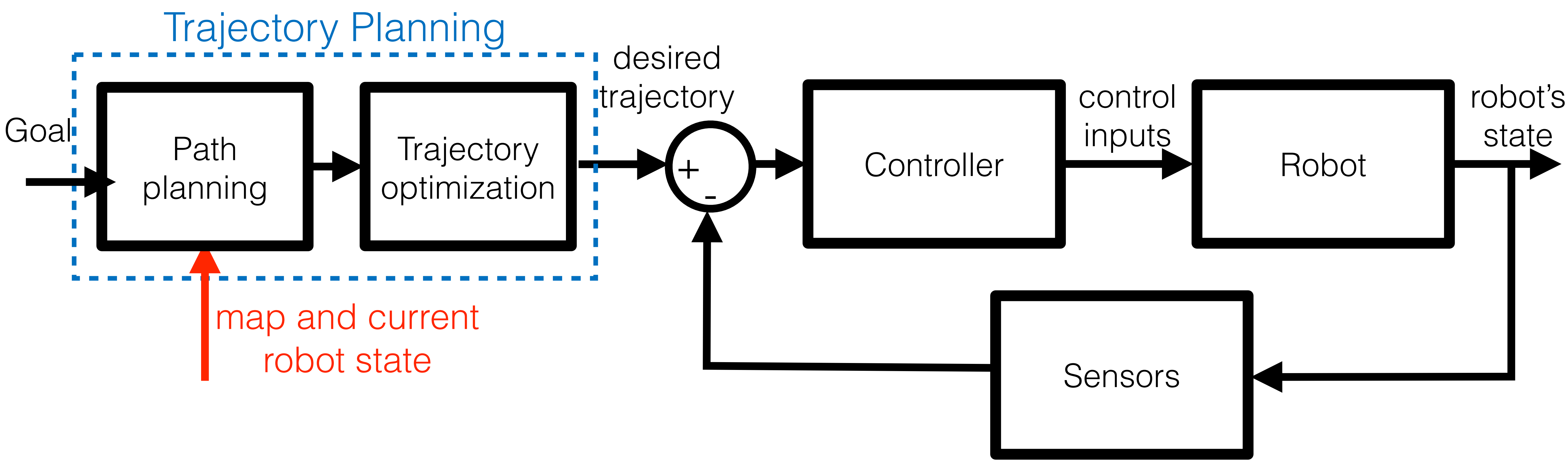
Lecture 25-26: Advanced Topics - Beyond Cameras

Next Steps

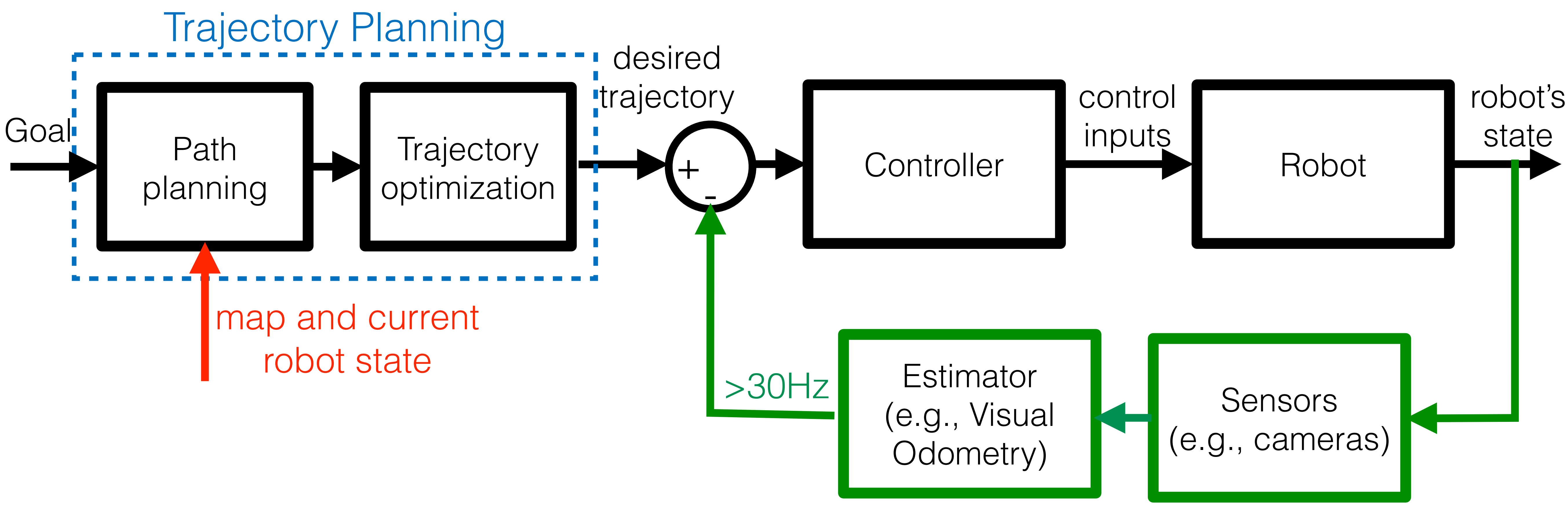
DATE	LECTURE	FINAL PROJECT STAGE
8-Nov	**Advanced topic:** Beyond cameras	
10-Nov	**Advanced topic:** 3D reconstruction	Project discussion
12-Nov	**Advanced topic:** Overview of open problems in robot perception and SLAM	
15-Nov	**Advanced topic:** Robust estimation	
17-Nov	**Advanced topic:** Robust estimation	Team check-in (on demand)
19-Nov	**Advanced topic:** Graph Neural Networks	
22-Nov	**Advanced topic:** Graph Neural Networks	
24-Nov	**Advanced topic:** Graph Neural Networks	Team check-in
	THANKSGIVING	
29-Nov	**Guest speaker:** Autonomous drones (Skydio)	
1-Dec	**Guest speaker:** ML uncertainty and verification	Team check-in
3-Dec	**Final presentations (Survey & System)**	
6-Dec	**Final presentations (System & Research)**	
8-Dec	**Final presentations (System & Research)**	

November 10th: final project open house and decisions

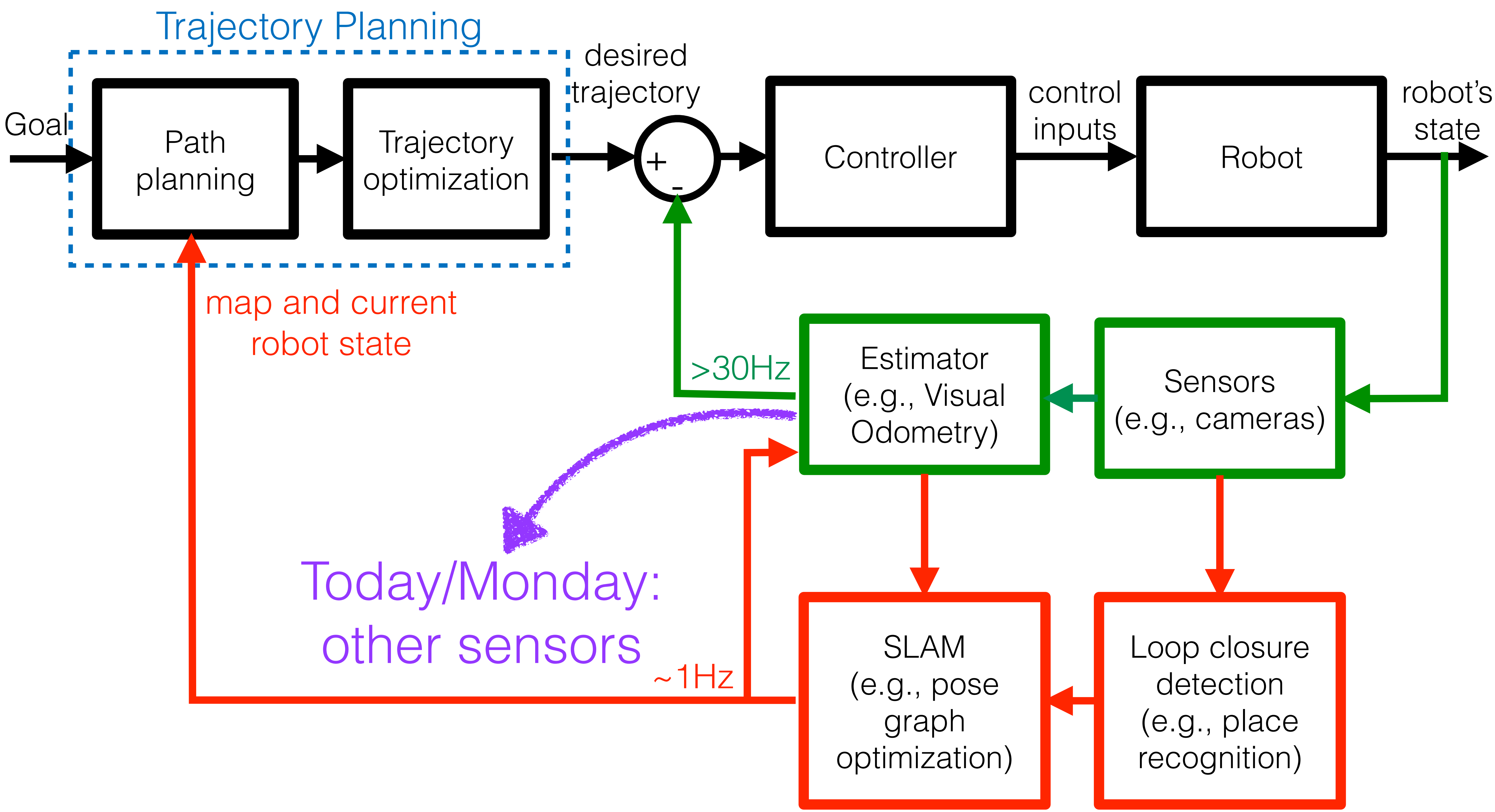
Big Picture


Trajectory Planning
Goal
Path planning
Trajectory optimization
desired trajectory
+
-
Controller
control inputs
Robot
robot's state
Sensors
map and current robot state

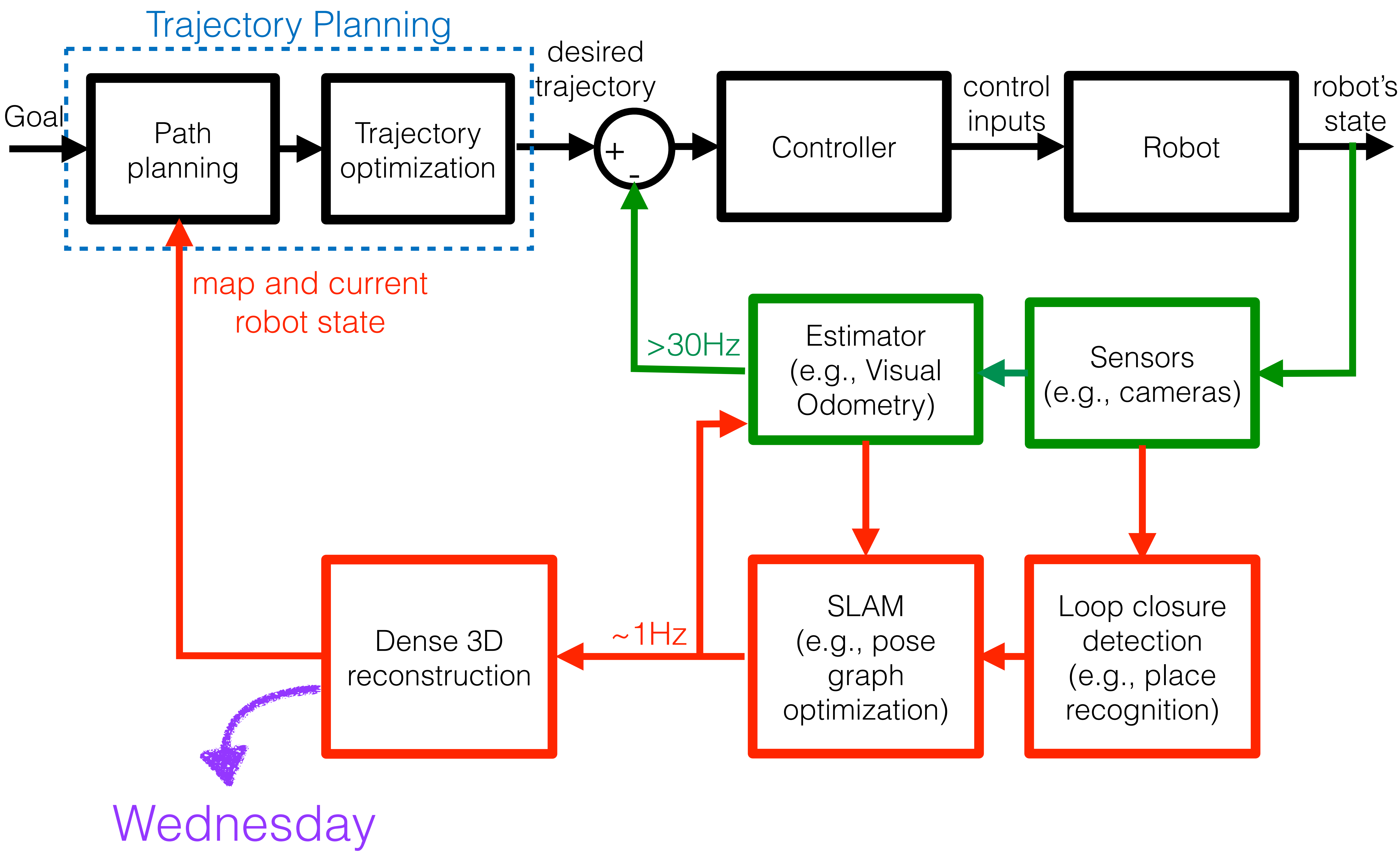
Big Picture


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)

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)
Today/Monday: other sensors
~1Hz
SLAM (e.g., pose graph optimization)
Loop closure detection (e.g., place recognition)

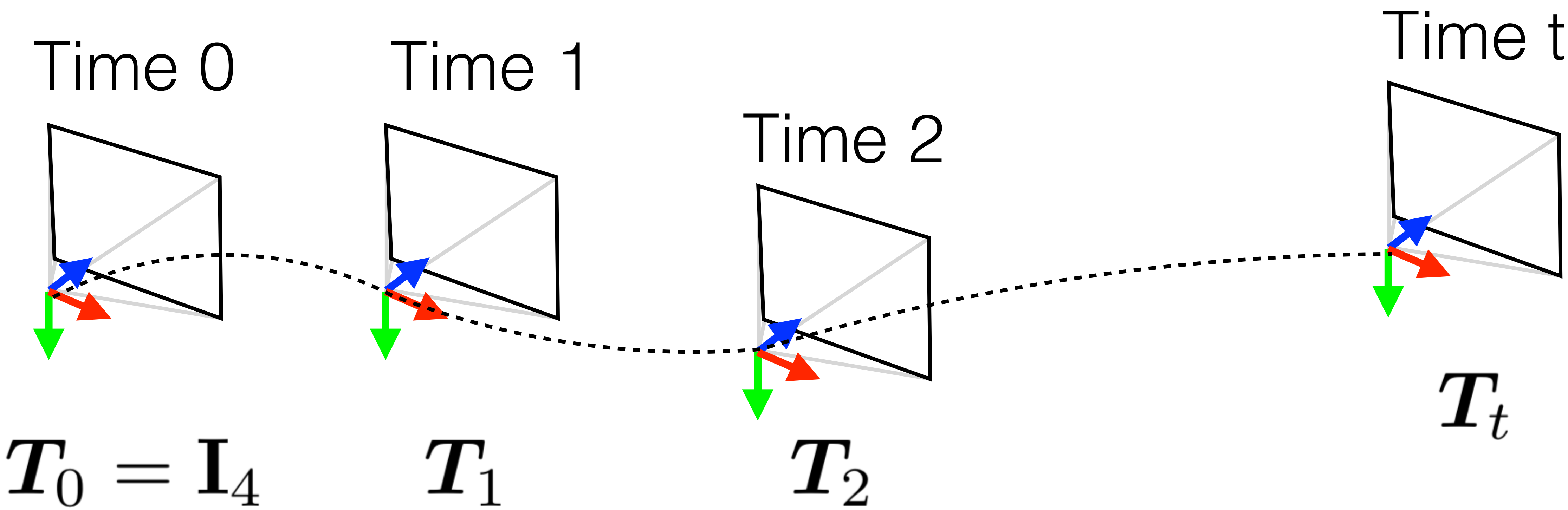
Big Picture



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)
Wednesday

Previously on VNAV: Visual Odometry

Visual odometry (VO): motion estimation estimation based on cameras (monocular, stereo, RGB-D, …)

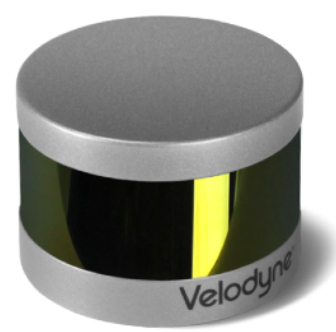


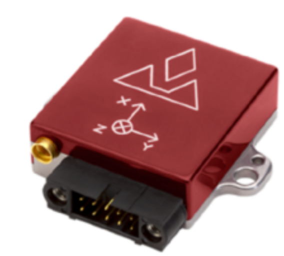
- feature matching tends to fail during fast motion
- feature-less frames
- mono VO only estimates motion up to scale

Beyond Cameras

How to get scale and improve robustness?
add more sensors!

- wheel odometry
- GPS
- Lidar
- Inertial Measurement Unit (IMU)
- Event Cameras






830g	160g	4g	3g
8 W	2.5 W	0.3W	~1 W

VIO: **V**isual-**I**nertial **O**dometry

- (a.k.a. visual-inertial navigation, VIN)

On-Manifold Preintegration for Real-Time Visual-Inertial Odometry

Christian Forster, Luca Carlone, Frank Dellaert, Davide Scaramuzza

***Abstract*—Current approaches for visual-inertial odometry (VIO) are able to attain highly accurate state estimation via nonlinear optimization. However, real-time optimization quickly becomes infeasible as the trajectory grows over time; this problem is further emphasized by the fact that inertial measurements come at high rate, hence leading to fast growth of the number of variables in the optimization. In this paper, we address this issue by preintegrating inertial measurements between selected keyframes into single relative motion constraints. Our first contribution is a *preintegration theory* that properly addresses the manifold structure of the rotation group. We formally discuss the generative measurement model as well as the nature of the rotation noise and derive the expression for the *maximum a posteriori* state estimator. Our theoretical development enables the computation of all necessary Jacobians for the optimization and a-posteriori bias correction in analytic form. The second contribution is to show that the preintegrated IMU model can be seamlessly integrated into a visual-inertial pipeline under the unifying framework of factor graphs. This enables the application of incremental-smoothing algorithms and the use of a *structureless* model for visual measurements, which avoids optimizing over the 3D points, further accelerating the computation. We perform an extensive evaluation of our monocular VIO pipeline on real and simulated datasets. The results confirm that our modelling effort leads to accurate state estimation in real-time, outperforming state-of-the-art approaches.**

of monocular vision and gravity observable [1] and provides robust and accurate inter-frame motion estimates. Applications of VIO range from autonomous navigation in GPS-denied environments, to 3D reconstruction, and augmented reality.

The existing literature on VIO imposes a trade-off between accuracy and computational efficiency (a detailed review is given in Section II). On the one hand, filtering approaches enable fast inference, but their accuracy is deteriorated by the accumulation of linearization errors. On the other hand, full smoothing approaches, based on nonlinear optimization, are accurate, but computationally demanding. Fixed-lag smoothing offers a compromise between accuracy for efficiency; however, it is not clear how to set the length of the estimation window so to guarantee a given level of performance.

In this work we show that it is possible to overcome this trade-off. We design a VIO system that enables fast incremental smoothing and computes the optimal *maximum a posteriori* (MAP) estimate in real time. An overview of our approach is given in Section IV.

The first step towards this goal is the development of a novel preintegration theory. The use of *preintegrated IMU measurements* was first proposed in [2] and consists of combining

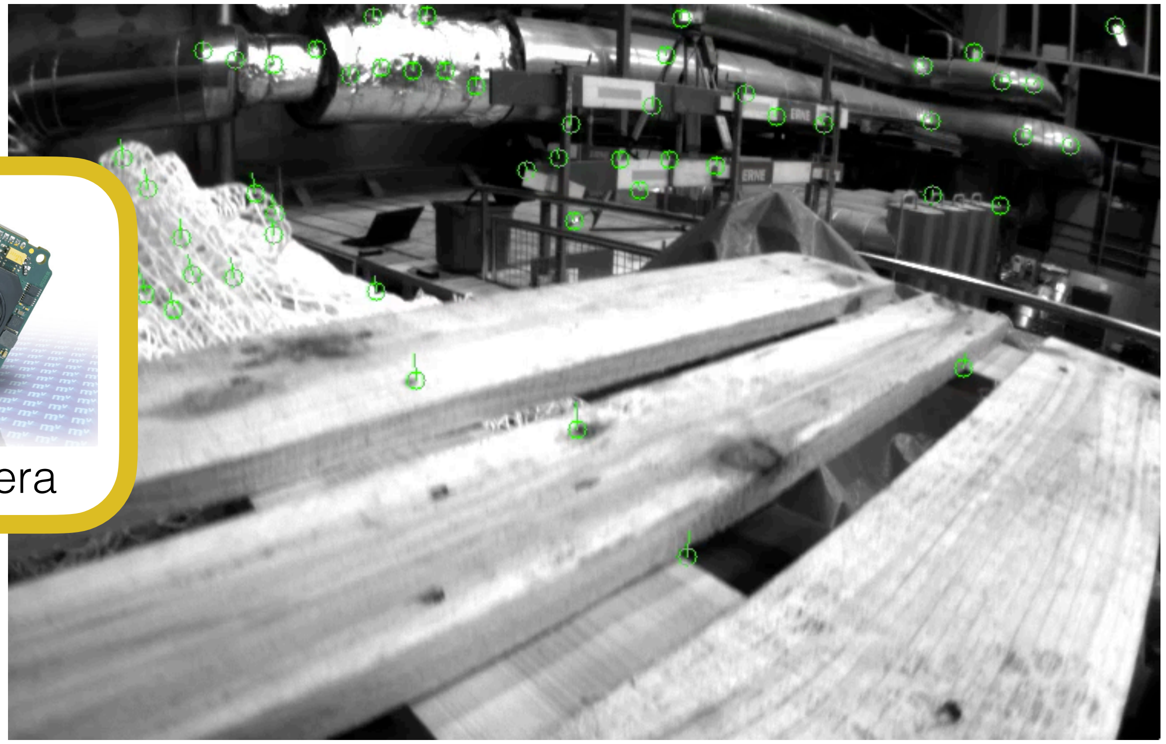
More resources:

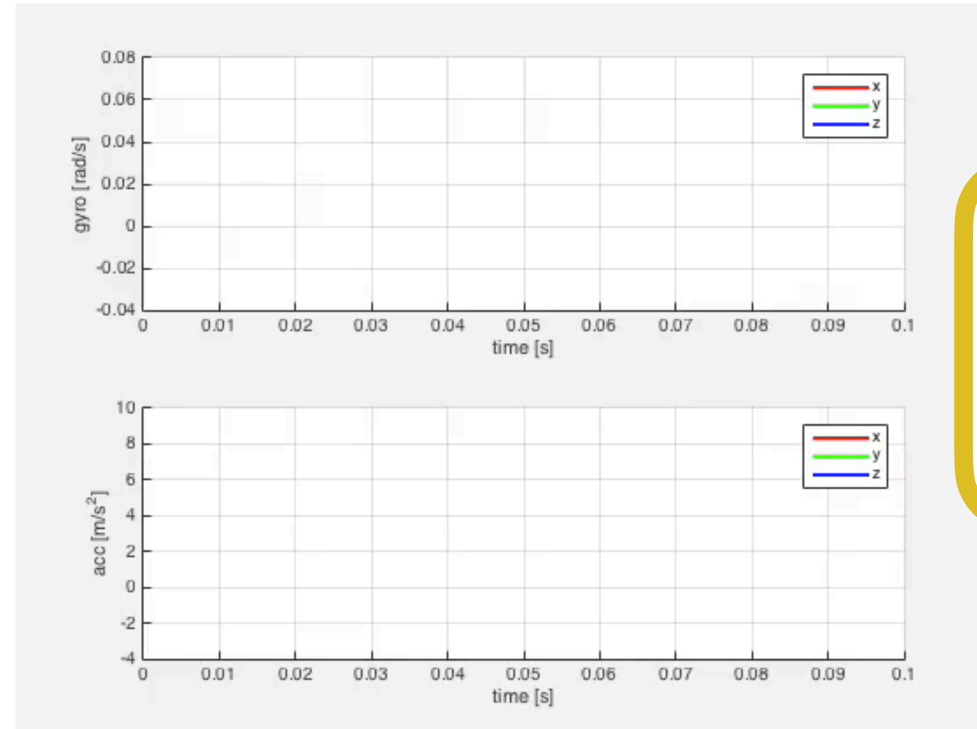
Visual-Inertial Navigation: Challenges and Applications

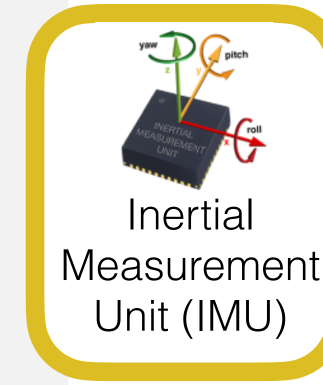
IROS 2019 Full-day Workshop: November 8, 2019, Macau, China

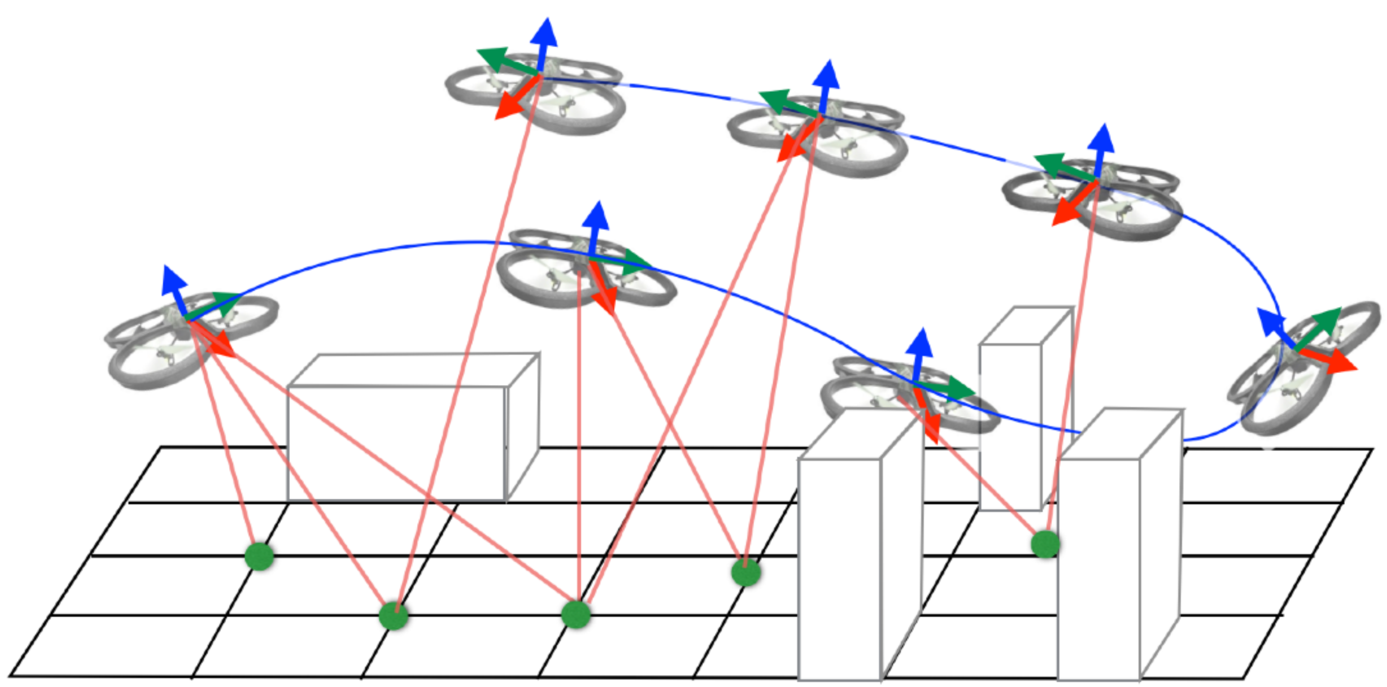
https://udel.edu/~ghuang/iros19-vins-workshop/

Visual-Inertial Navigation (VIN)

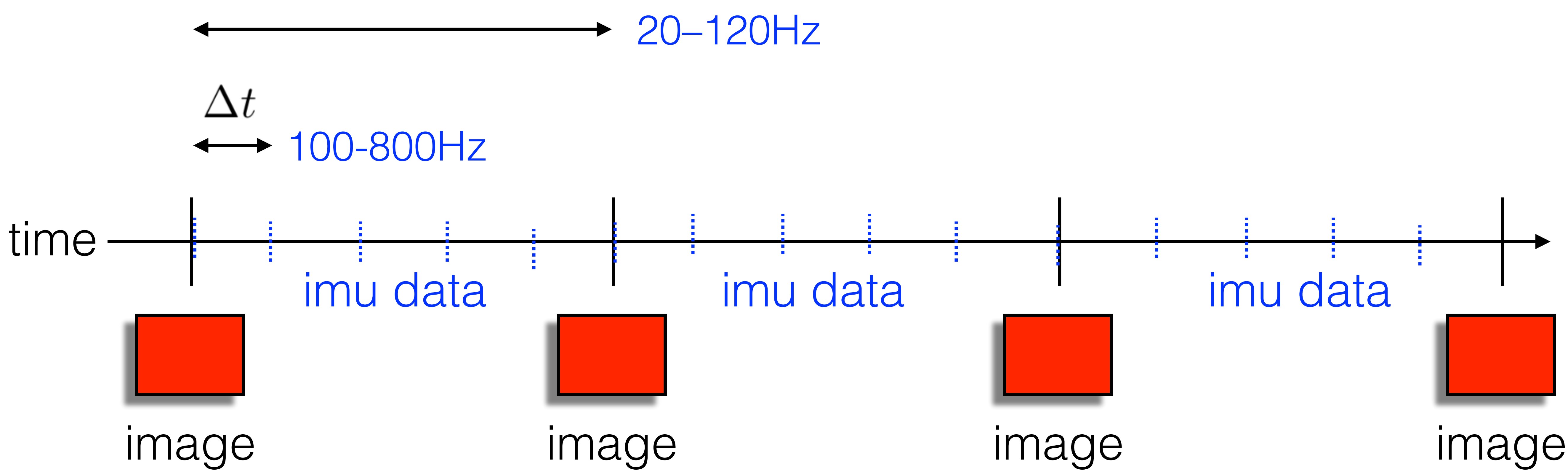


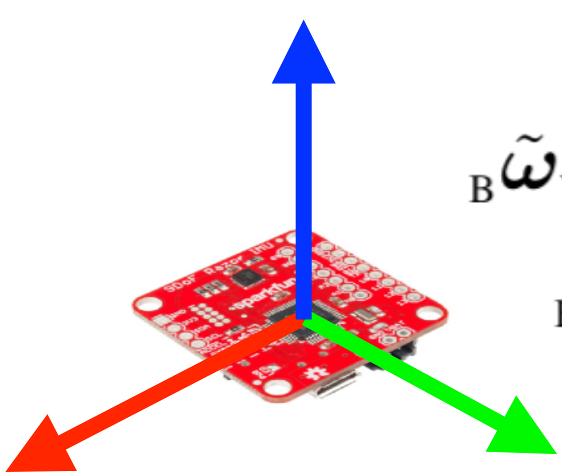




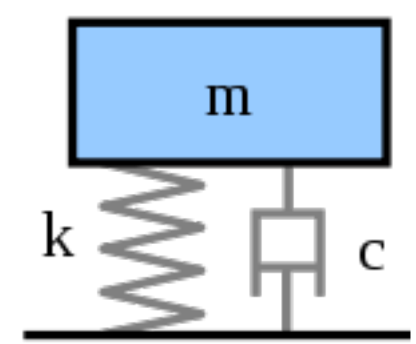


Visual-Inertial Odometry

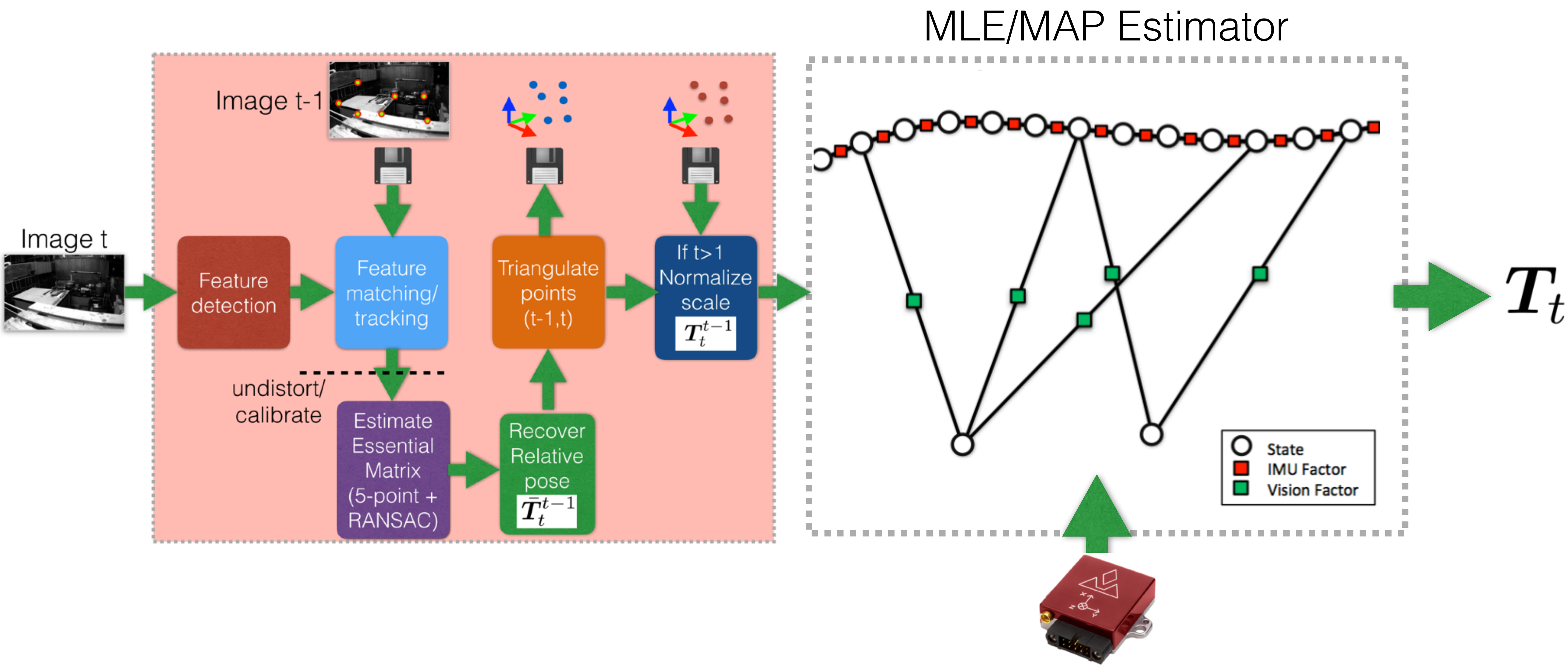




$$
\begin{aligned}
{}_{\mathrm{B}}\tilde{\boldsymbol{\omega}}_{\mathrm{WB}}(t) &= {}_{\mathrm{B}}\boldsymbol{\omega}_{\mathrm{WB}}(t) + \mathbf{b}^{g}(t) + \boldsymbol{\eta}^{g}(t) \\
{}_{\mathrm{B}}\tilde{\mathbf{a}}(t) &= \mathrm{R}_{\mathrm{WB}}^{\mathsf{T}}(t)\left({}_{\mathrm{W}}\mathbf{a}(t) - {}_{\mathrm{W}}\mathbf{g}\right) + \mathbf{b}^{a}(t) + \boldsymbol{\eta}^{a}(t),
\end{aligned}
$$



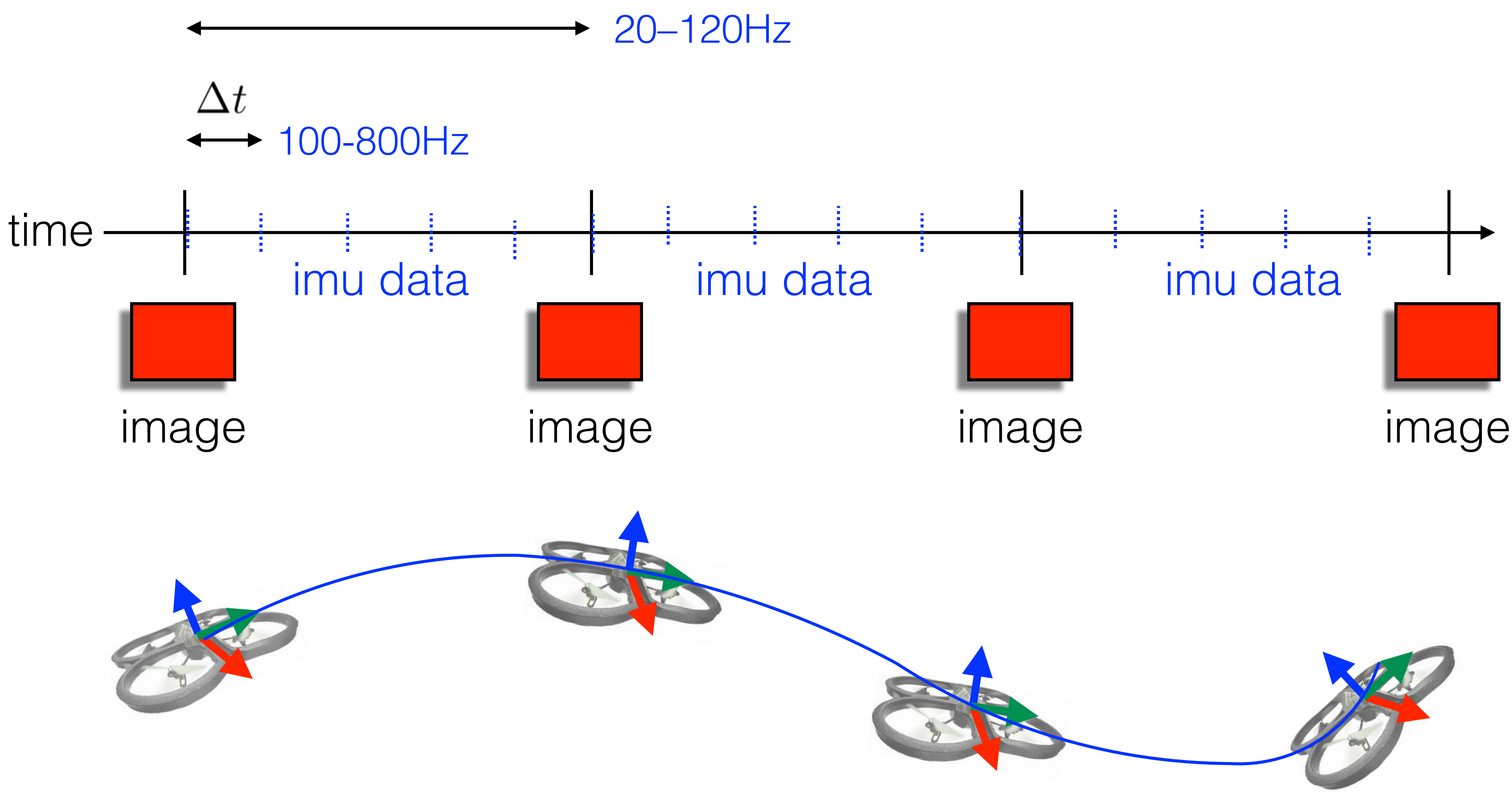
Visual-Inertial Odometry



$$\min_{\substack{\boldsymbol{T}_i, i=1,\dots,N_C \\ \boldsymbol{p}_k, k=1,\dots,N}} \overbrace{\sum_{k=1}^{N} \sum_{i\in\mathcal{C}_k} \|\boldsymbol{x}_{k,i} - \pi(\boldsymbol{T}_i, \boldsymbol{p}_k)\|^2}^{\text{Camera factors}} + \overbrace{\sum_{i=1,\dots,N_C-1} \|r_{\mathrm{imu}}(\boldsymbol{T}_i, \boldsymbol{T}_{i+1}, \boldsymbol{v}_i, \boldsymbol{v}_{i+1}, \boldsymbol{b}_i, \boldsymbol{b}_{i+1})\|^2}^{\text{Imu factors}}$$

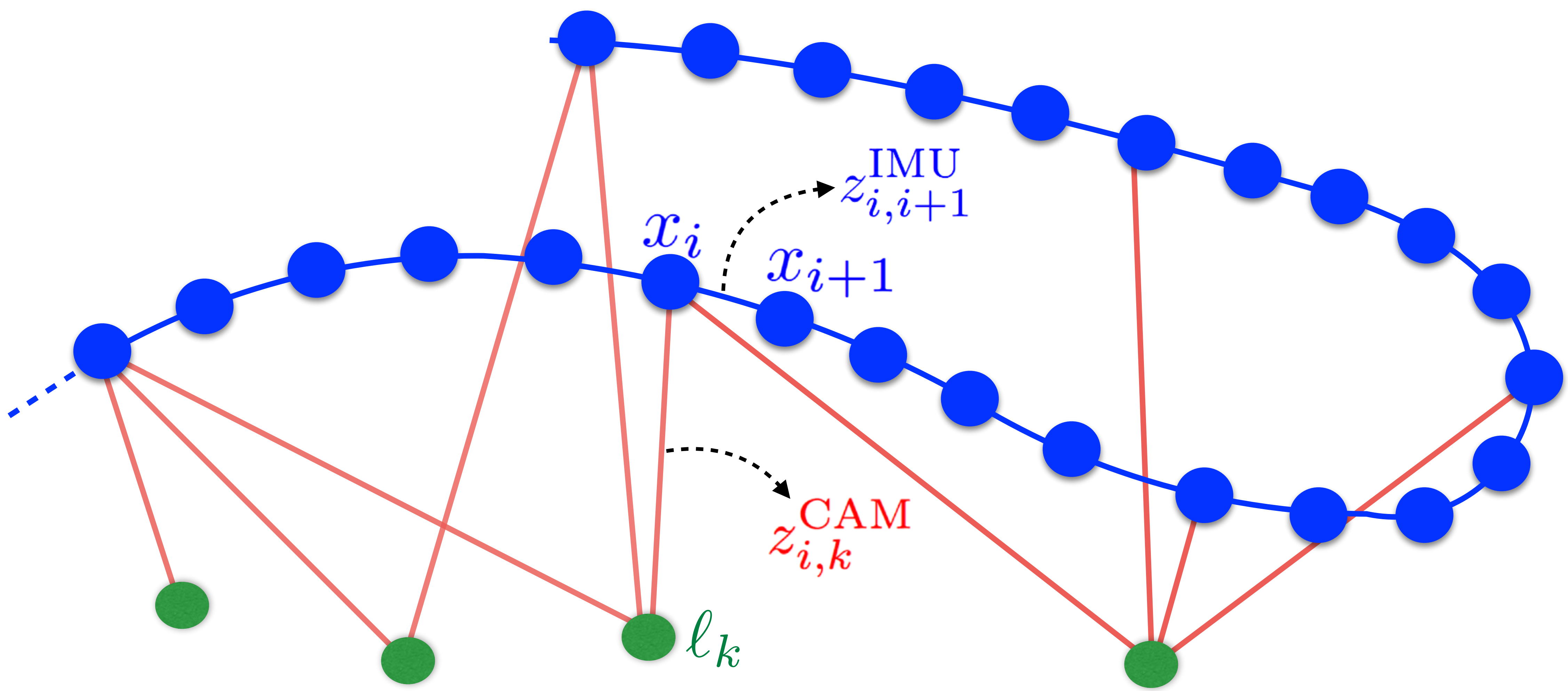
Need to include velocities and IMU biases in the state …

Visual-Inertial Odometry



- **Fixed-lag smoother**: estimate a fixed window of recent states from time k-T, k-T+1, .. k (sliding window)

MAP Estimation



Challenges:

- IMU measurements arrive at high-rate (~200Hz) ⇨ **IMU preintegration**
- camera observes hundreds of landmarks per frame ⇨ **structureless vision factors**
- need to solve optimization problem quickly

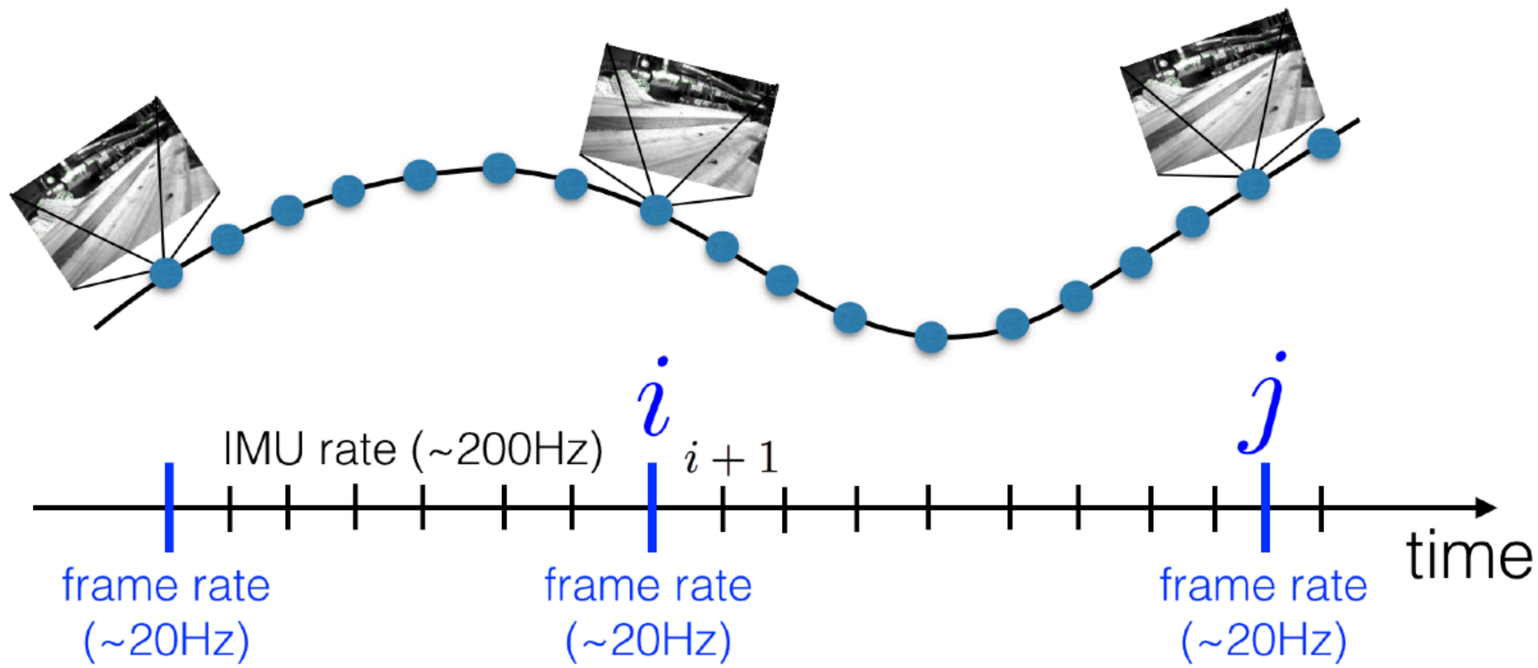
IMU Preintegration

Key idea: integrate IMU measurements between frames

many measurements & states

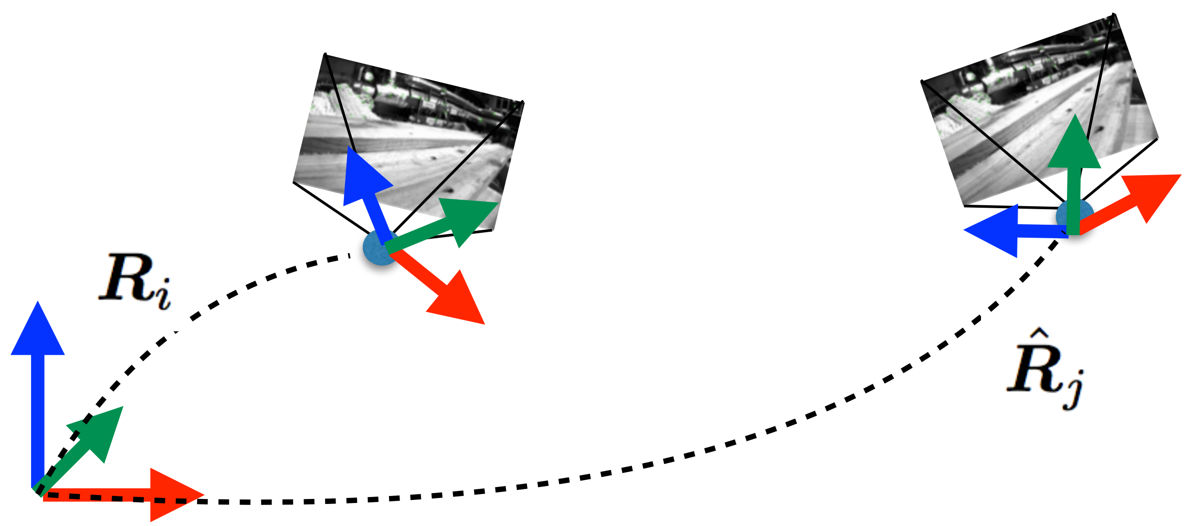
$$
\begin{aligned}
z^{\mathrm{IMU}}_{i,i+1} &= f(x_i, x_{i+1}) + \epsilon \\
z^{\mathrm{IMU}}_{i+1,i+2} &= f(x_{i+1}, x_{i+2}) + \epsilon \\
&\vdots \\
z^{\mathrm{IMU}}_{j-1,j} &= f(x_{j-1}, x_j) + \epsilon
\end{aligned}
$$

original model



IMU Preintegration

Standard integration

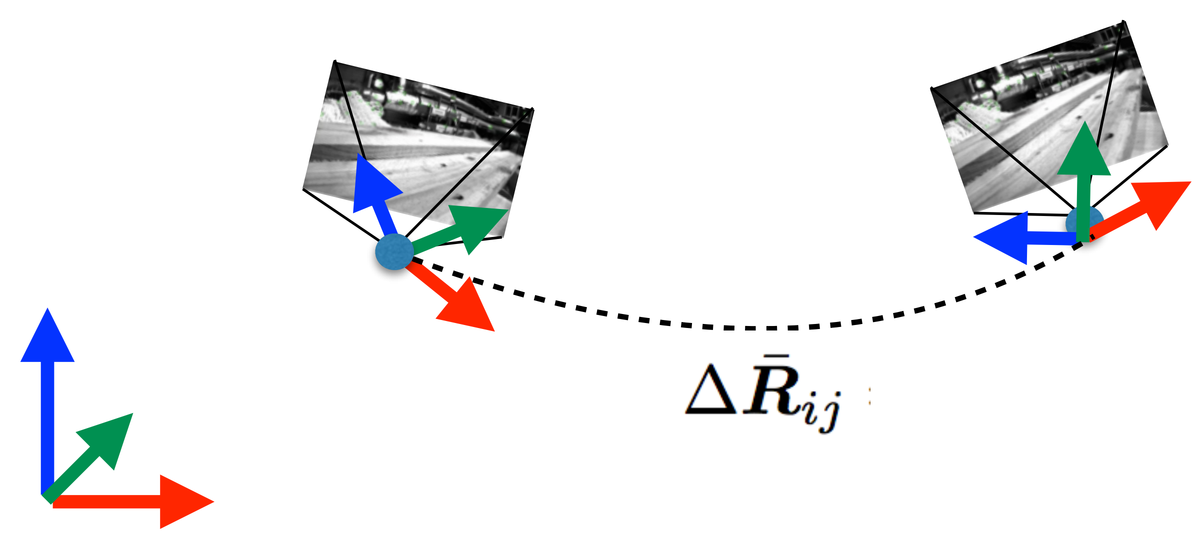


$$\hat{\boldsymbol{R}}_j = \boldsymbol{R}_i \cdot \mathrm{Exp}\left(w_{i,i+1}\delta t\right) \cdots \mathrm{Exp}\left(w_{j-1,j}\delta t\right)$$

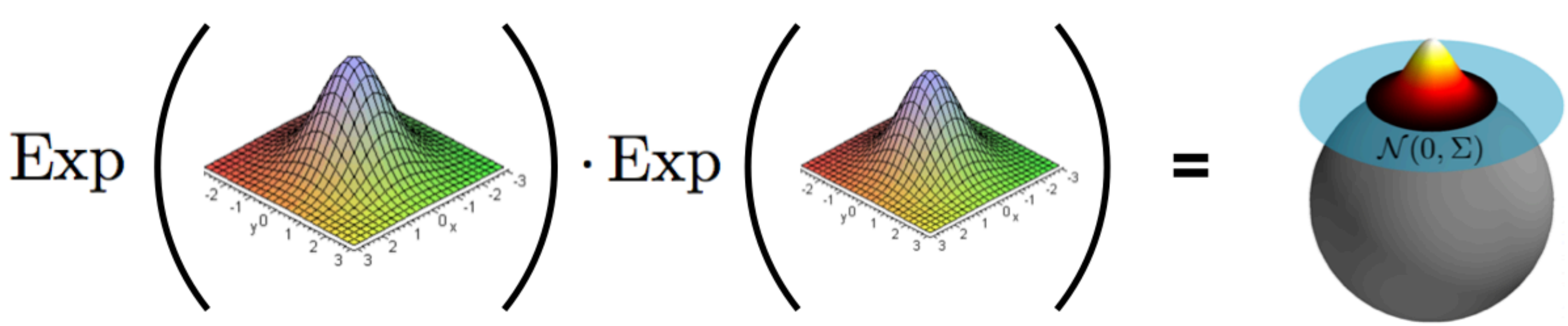
initial rotation

rotation rate measurements

Preintegration



$$\Delta\bar{\boldsymbol{R}}_{ij} = \mathrm{Exp}\left(w_{i,i+1}\delta t\right) \cdots \mathrm{Exp}\left(w_{j-1,j}\delta t\right)$$



Carlone, Kira, Beall, Indelman, Dellaert, Eliminating conditionally independent sets in factor graphs: a unifying perspective based on smart factors, ICRA'14.
Forster, Carlone, Dellaert, Scaramuzza, *IMU Preintegration on Manifold for Efficient Visual-Inertial Maximum-a-Posteriori Estimation*, RSS'15 (best paper finalist)

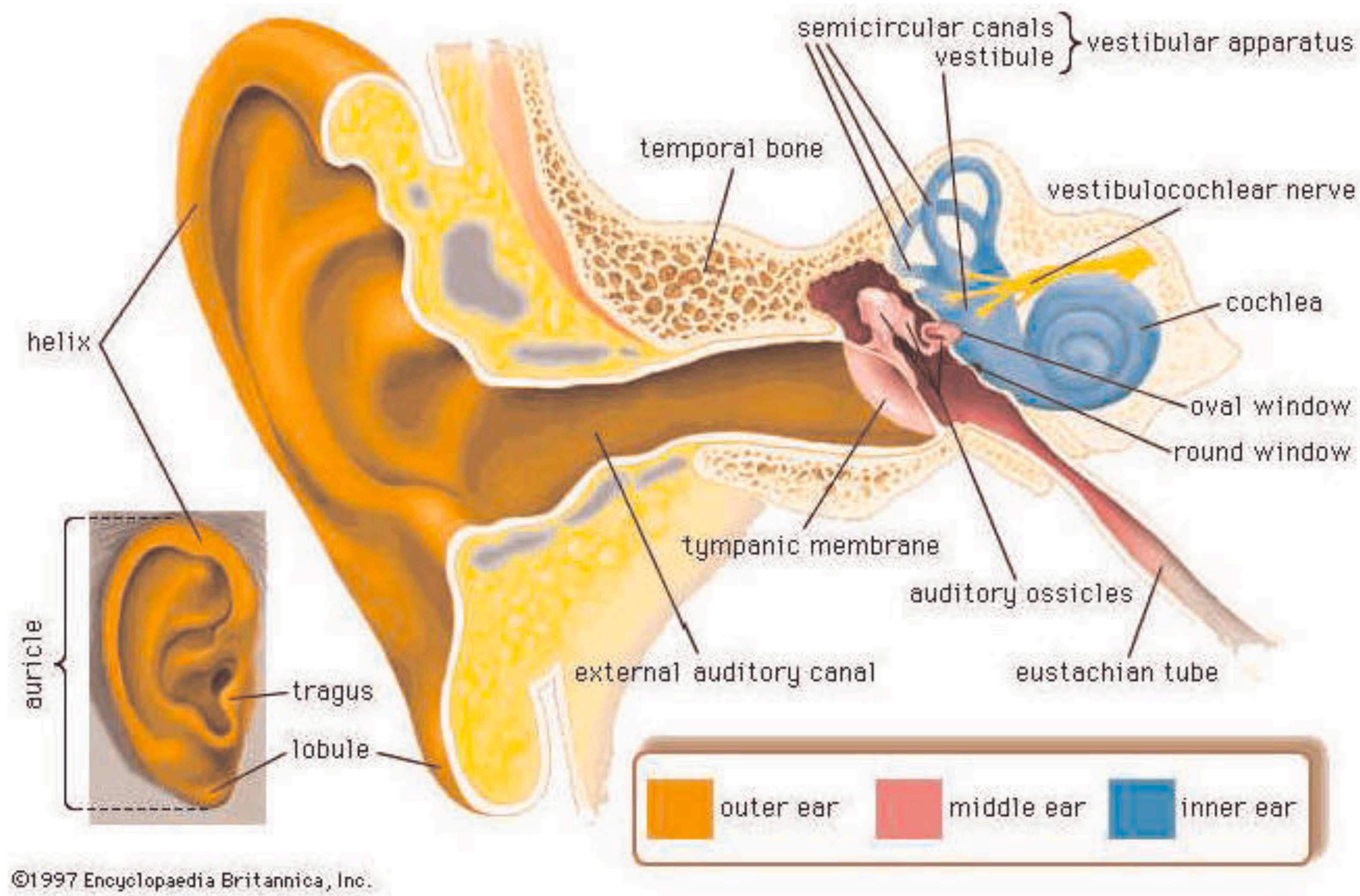


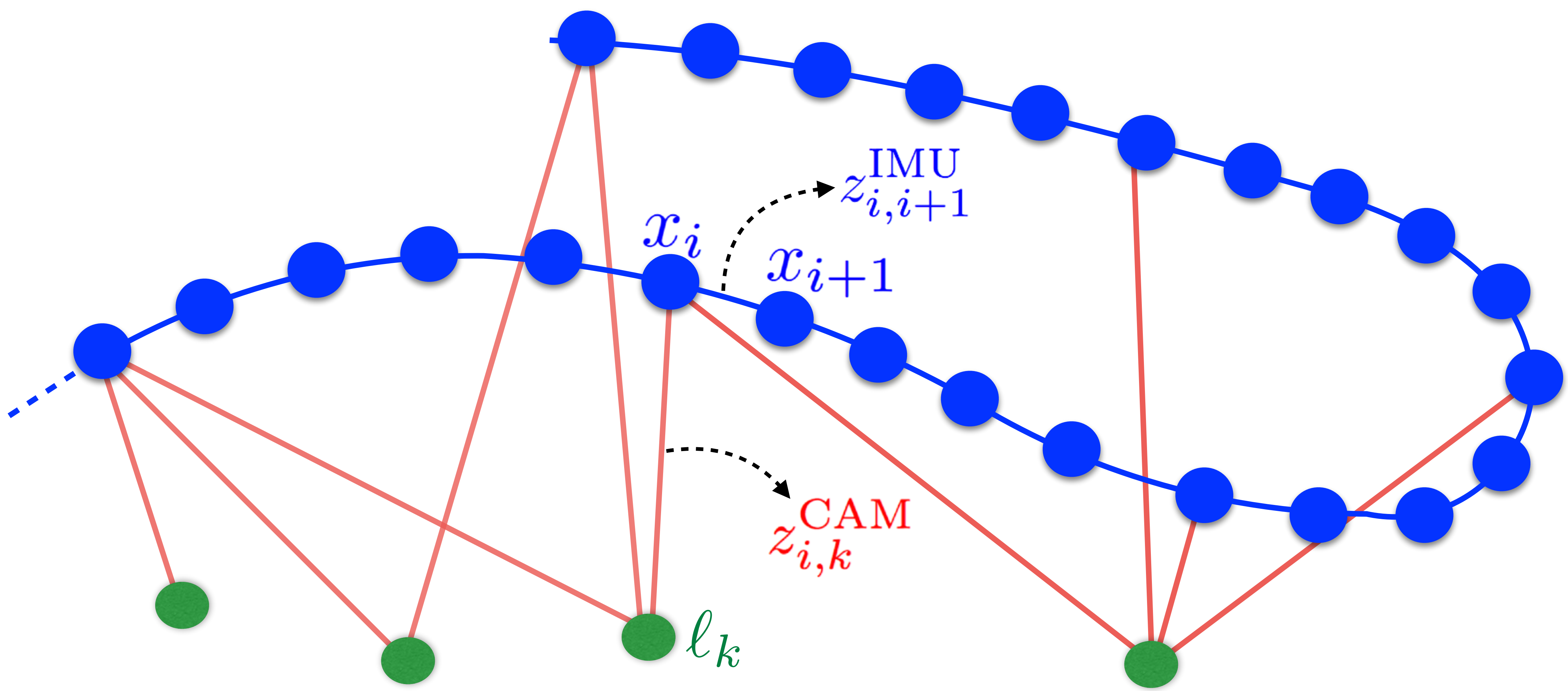
Fig. 2. Human ear (taken with permission from Encyclopaedia Britannica 2001).

16.485: VNAV - Visual Navigation for Autonomous Vehicles

Luca Carlone

Lecture 25-26: Advanced Topics - Beyond Cameras

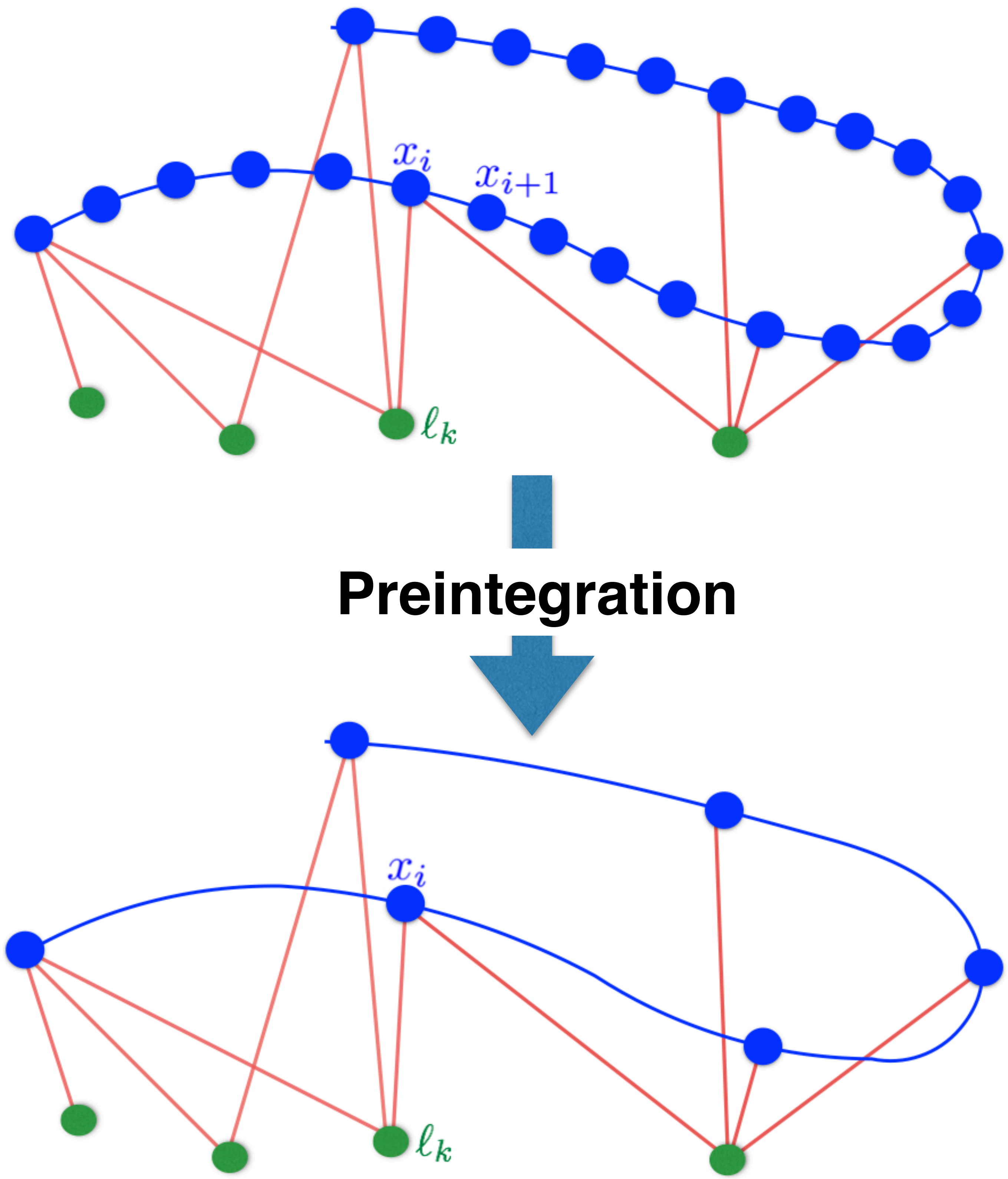
MAP Estimation



Challenges:

- IMU measurements arrive at high-rate (~200Hz) ⇨ **IMU preintegration**
- camera observes hundreds of landmarks per frame ⇨ **structureless vision factors**
- need to solve optimization problem quickly

IMU Preintegration



After 10 seconds, original problem has ~10^4 states

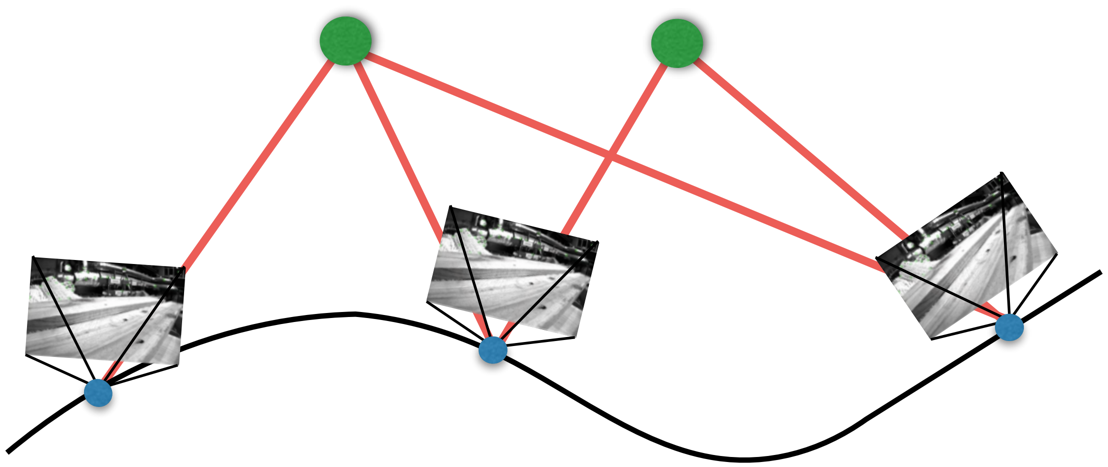
After 10 seconds, preintegrated problem has ~10^2 states

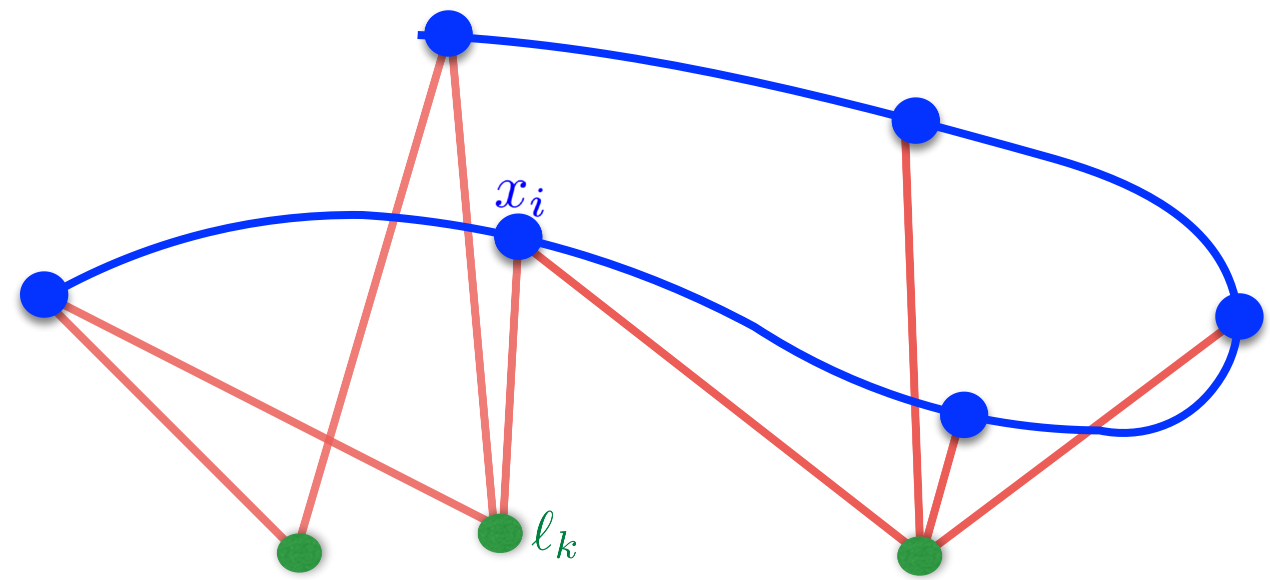
On-Manifold Preintegration for Real-Time Visual-Inertial Odometry

Christian Forster, Luca Carlone, Frank Dellaert, Davide Scaramuzza

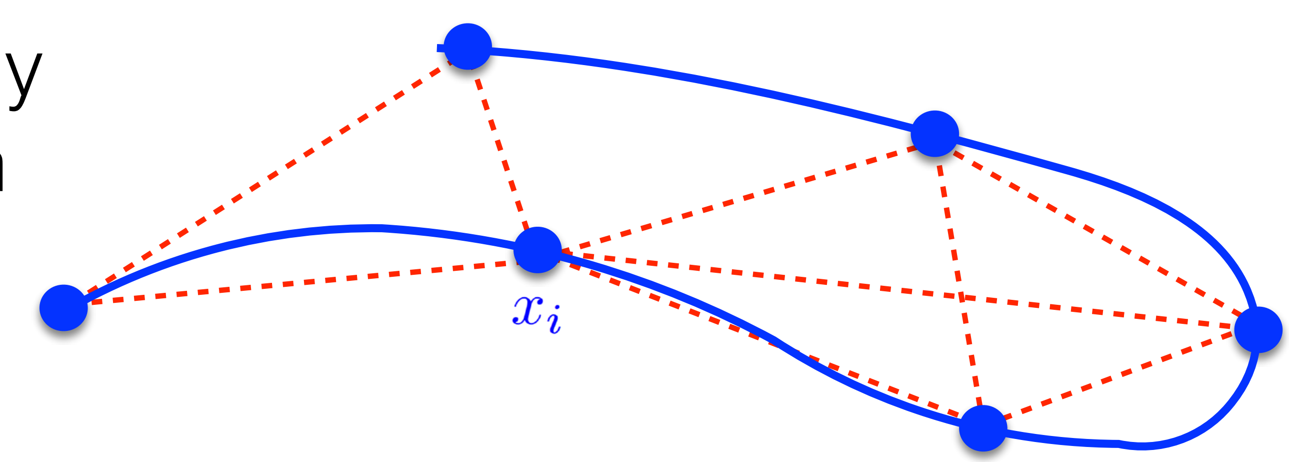
Structureless Vision Model

Marginalization
of 3D landmarks





Schur complement



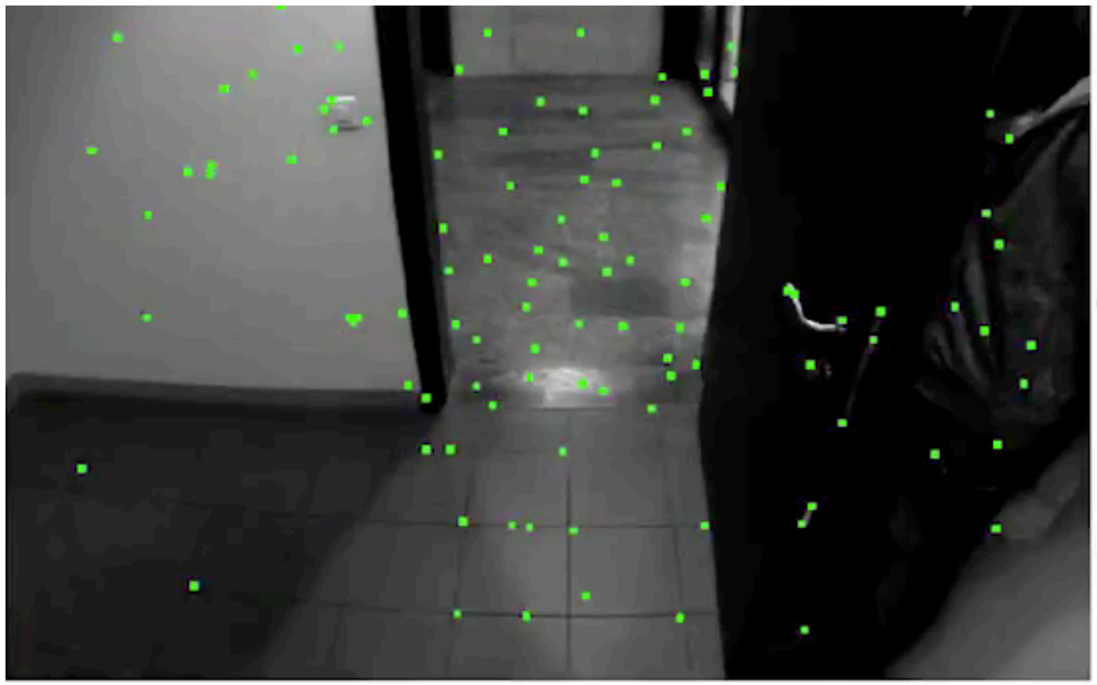
Schur complement trick:

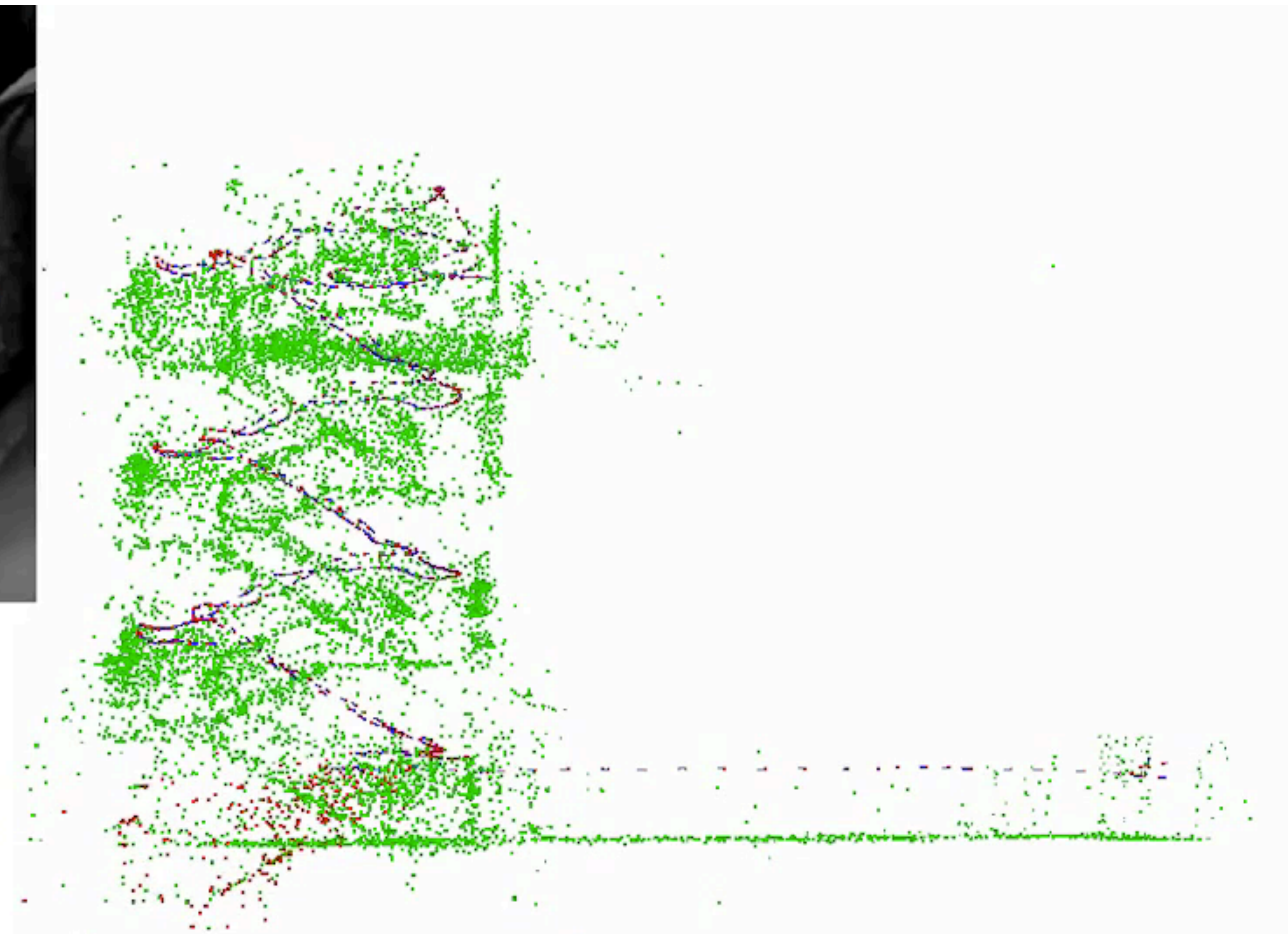
- solve for each landmark separately
- substitute back in the optimization

Further reduction of the number of variables in the optimization!

Carlone, Alcantarilla, Chiu, Zsolt, Dellaert, *Mining structure fragments for smart bundle adjustment*, BMVC'14.

Visual-Inertial Odometry





Hand-held sensor

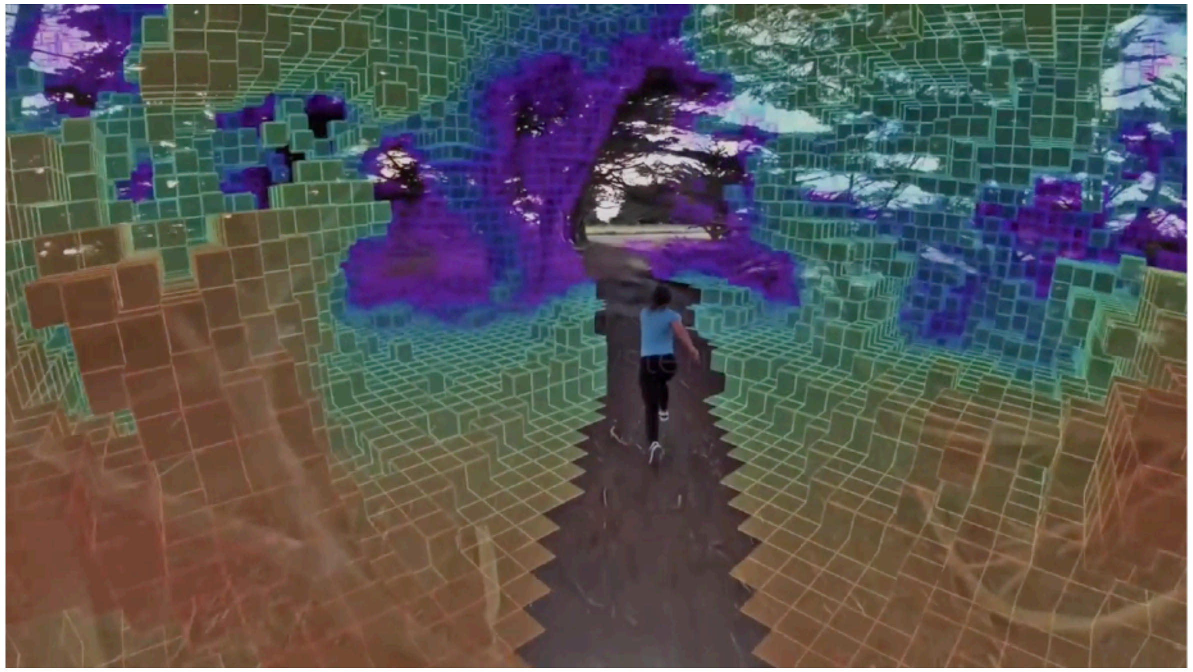
Implemented in GTSAM (ImuFactor)

Others: OpenVINS, VINS-mono, ORB-SLAM3, ROVIO, ..

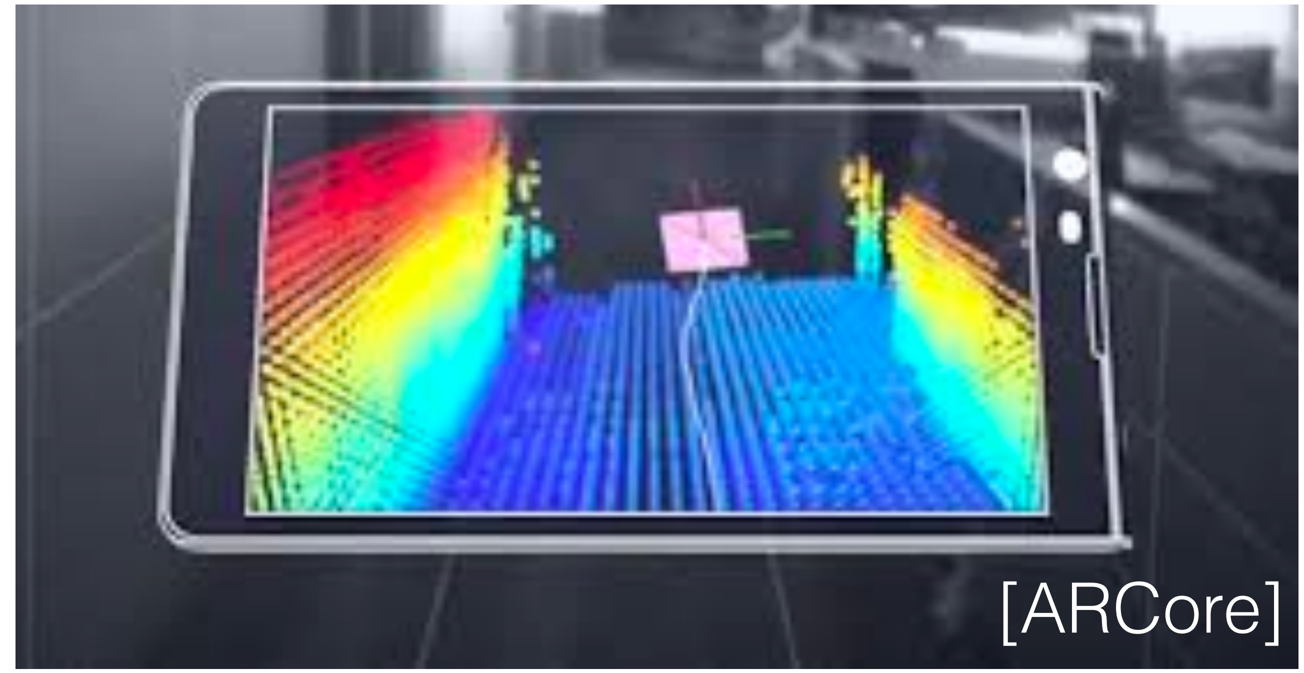
[Forster, Carlone, Dellaert, Scaramuzza. On-manifold preintegration for real-time visual-inertial odometry. *TRO* 2017]

Engineered Solutions / Applications

Skydio R1 drone

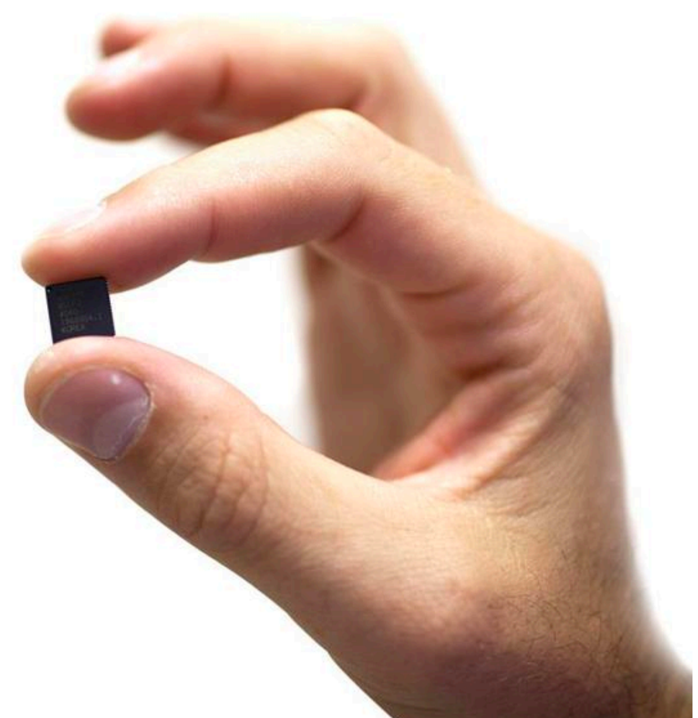


Google Tango



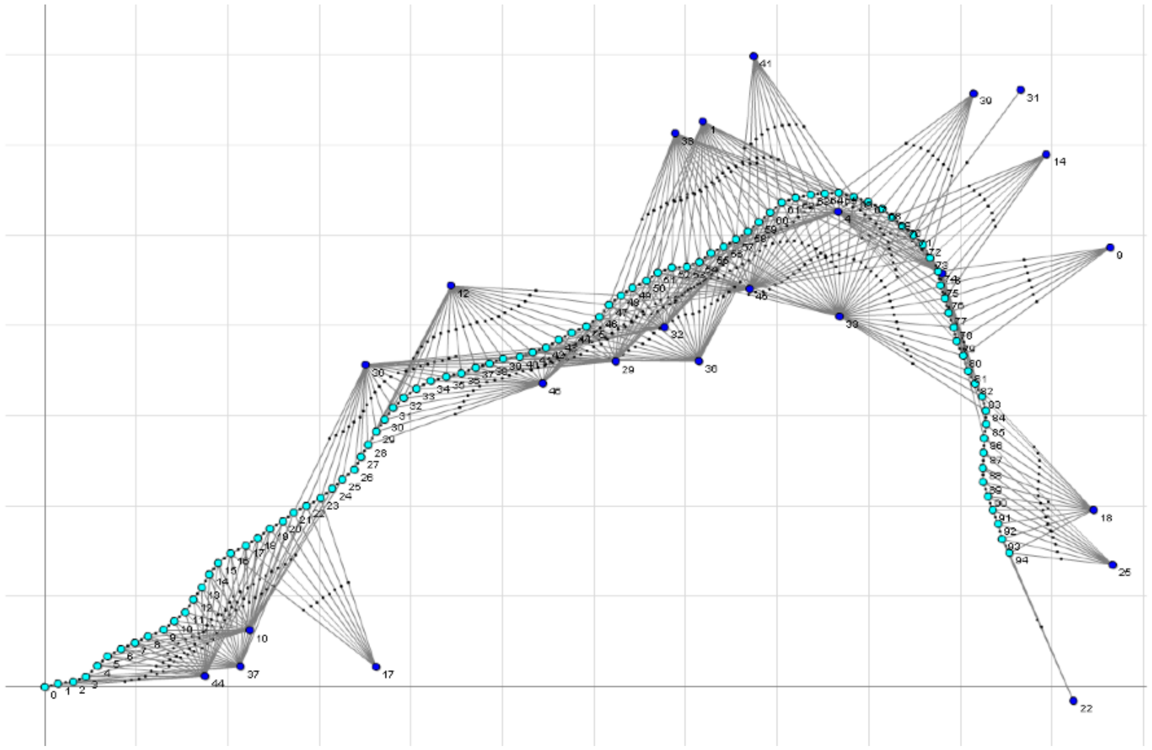
Oculus Rift Goggles

Pokemon Go



Navion Chip
2017

SLAM



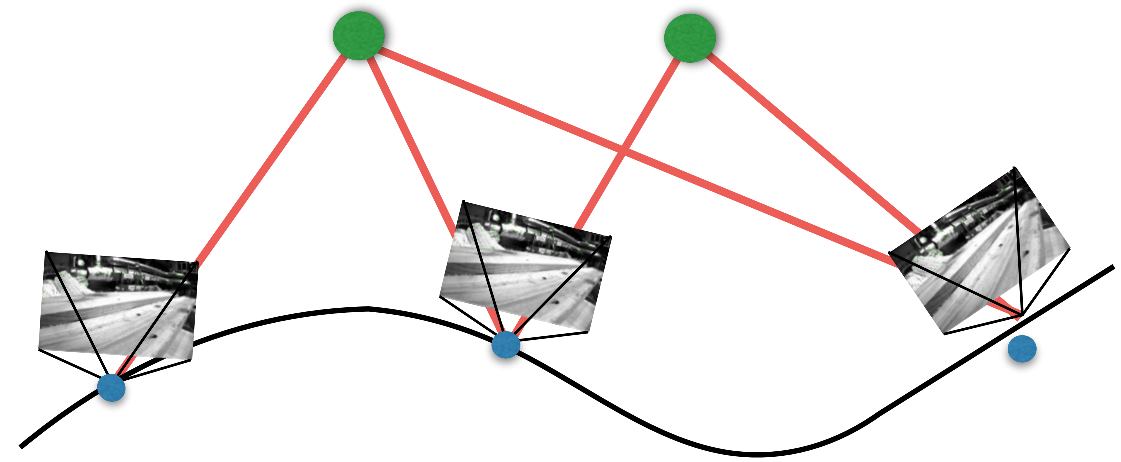
All sensors

Optimize entire history (full smoothing)

Latency: >200ms

Loop closures: error remains bounded

VIN



Camera & IMU

Typically optimize over receding horizon

Latency: <50ms

No loop closures: error accumulates over time

Observability

Cases	Number of Solutions
Rotation around 2 or more axes Varying Acceleration $n_i = 5,\ N \geq 2$; $n_i \geq 6,\ \forall N$	Unique Solution
Rotation around a single axis Varying Acceleration $n_i = 5,\ N \geq 2$; $n_i \geq 6,\ \forall N$	Two Solutions
Rotation around 1 or more axes Varying Acceleration $n_i = 4,\ N \geq 2$	Two Solutions
Rotation around 2 or more axes Constant and non null Acceleration $n_i = 4, 5,\ N \geq 2$; $n_i \geq 6,\ \forall N$	Two Solutions
Rotation around a single axis Constant Acceleration	Infinite Solutions
No rotation $\forall n_i,\ \forall N$	Infinite Solutions
Null Acceleration $\forall n_i,\ \forall N$	Infinite Solutions
Any Motion $n_i \leq 3,\ \forall N$; $n_i = 4, 5,\ N = 1$	Infinite Solutions

N: # points

n_i = #images in which point "i" is observed

Agostino Martinelli. Closed-form solution of visual-inertial structure from motion. International Journal of Computer Vision, Springer Verlag, 2013. <hal-00905881>

Visual-Inertial Odometry

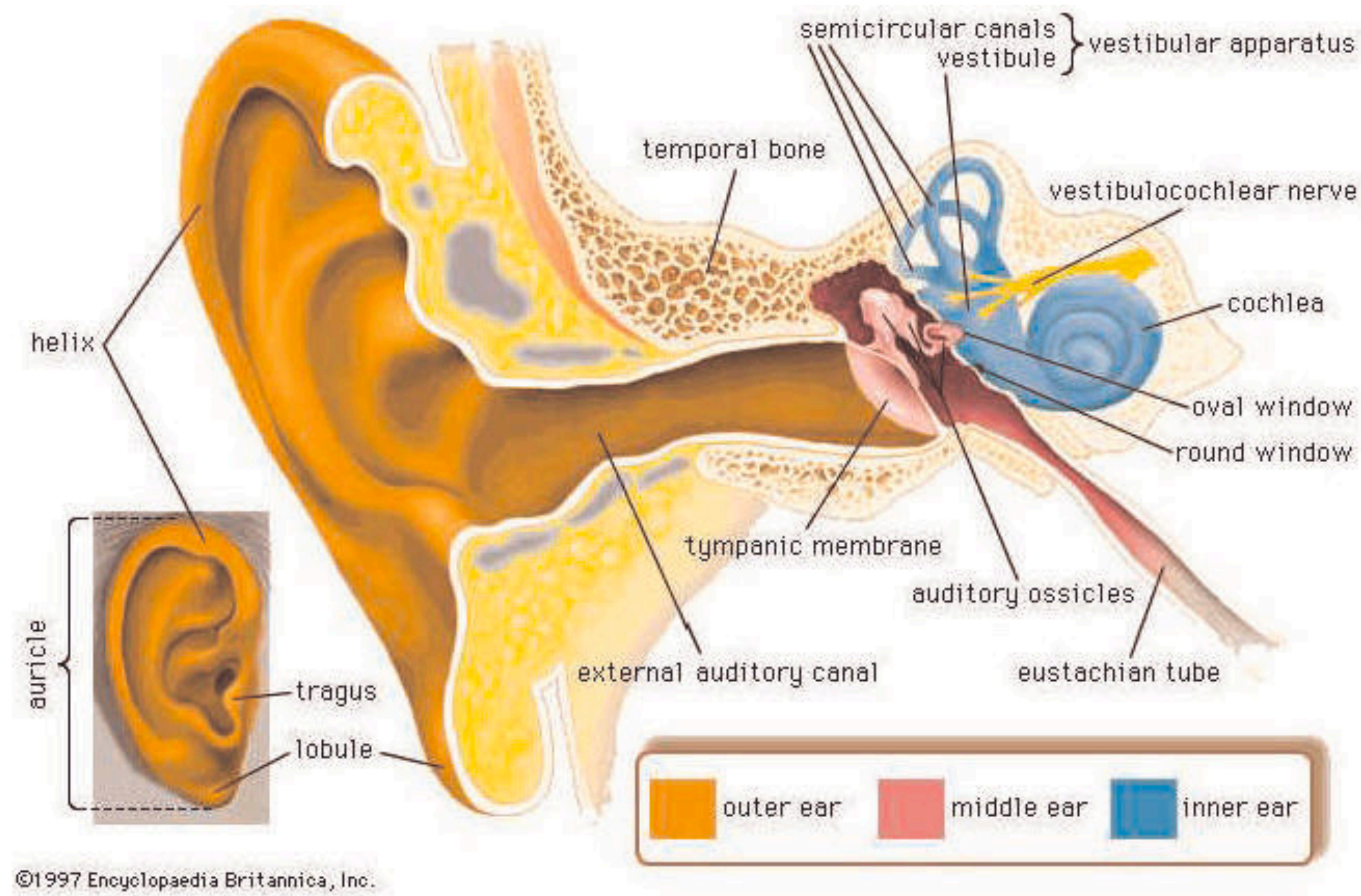
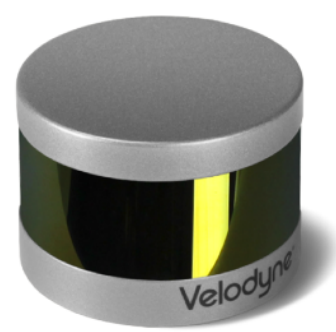


Fig. 2. Human ear (taken with permission from Encyclopaedia Britannica 2001).

- the **semicircular canals** measure rotational movements
- and the **otoliths** measure linear accelerations

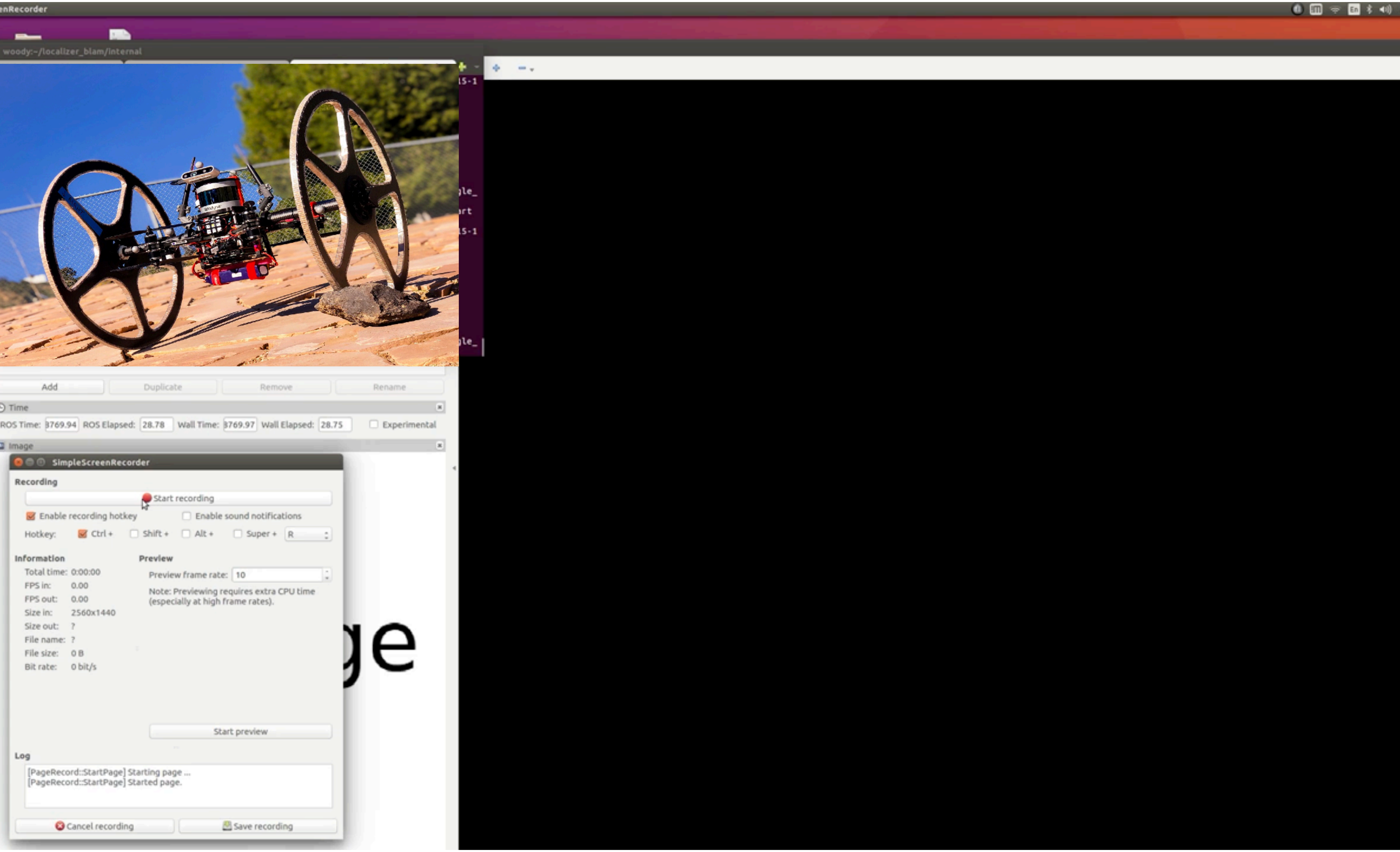
Beyond Cameras

- wheel odometry
- GPS
- Lidar
- Inertial Measurement Unit (IMU)
- Event Cameras



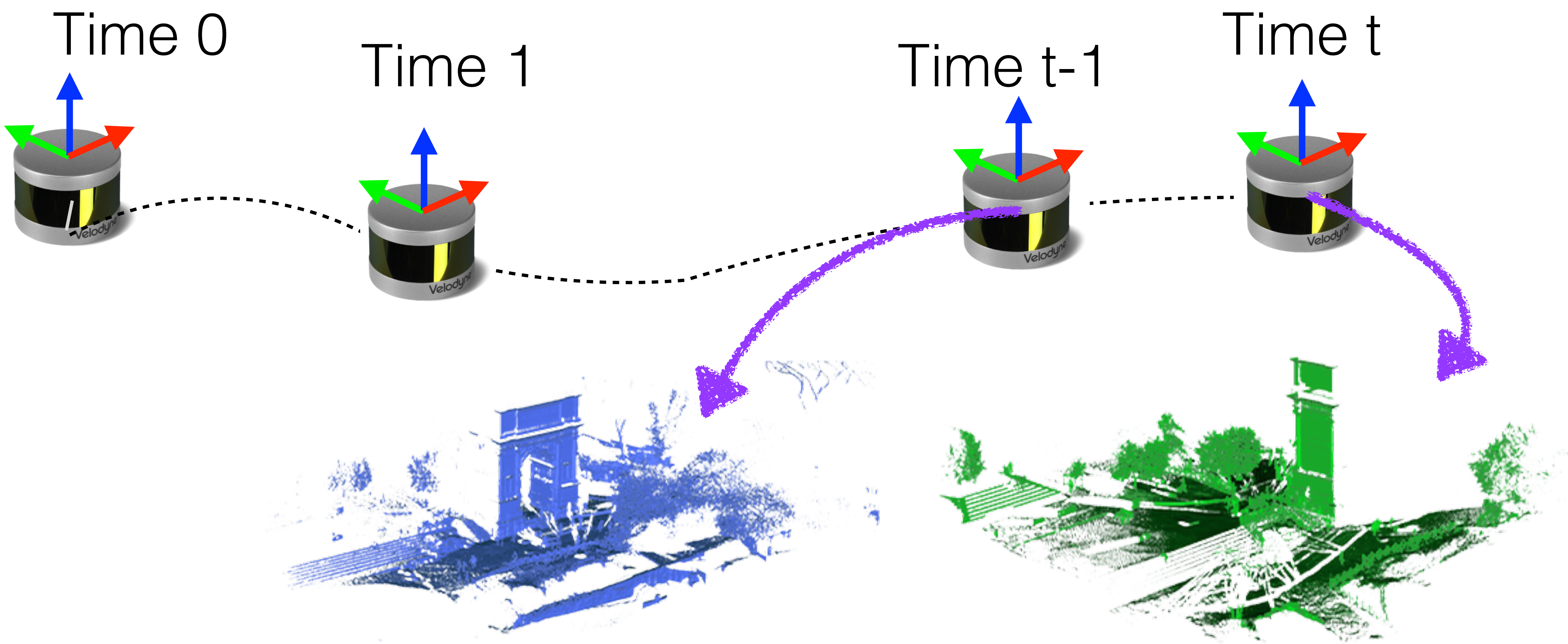
830g	160g	4g	3g
8 W	2.5 W	0.3W	~1 W

Lidar Odometry & Lidar SLAM

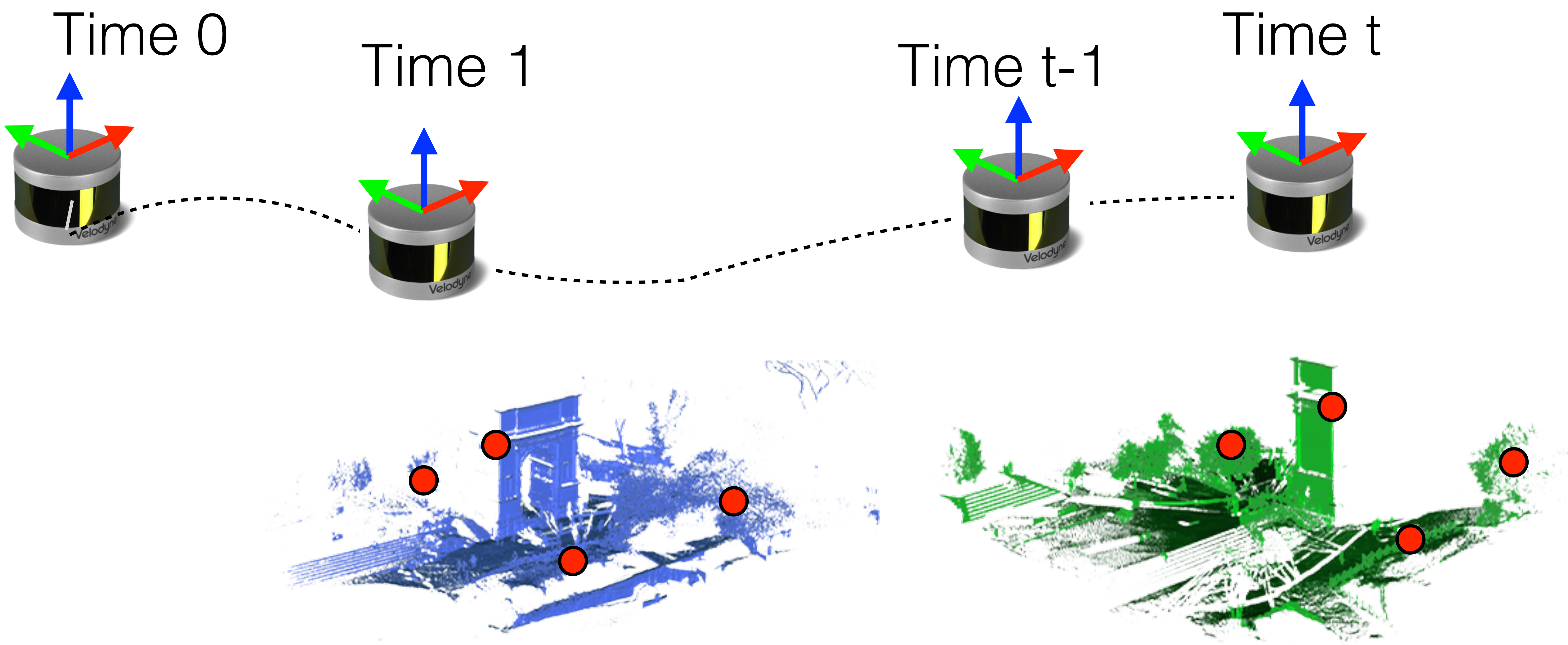


DARPA Subterranean Challenge, in collaboration with JPL

Feature-based Lidar Odometry



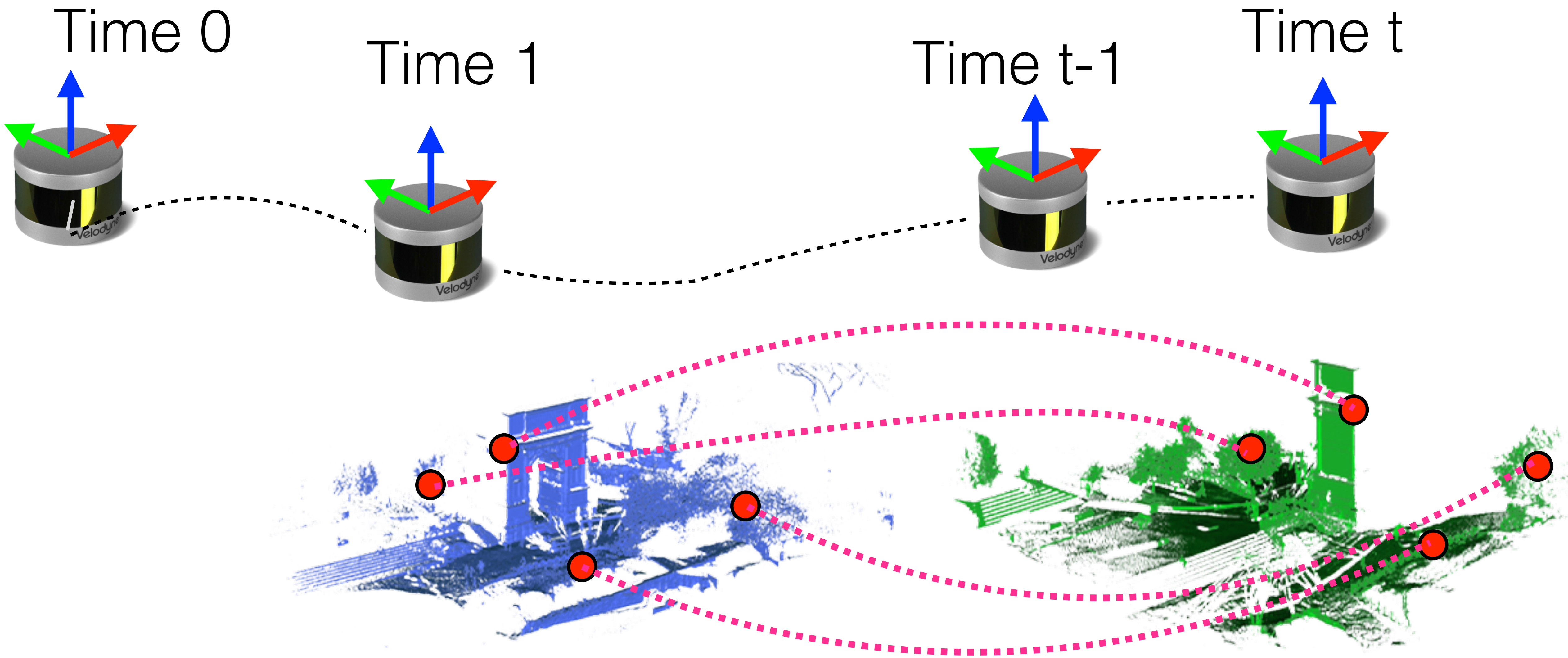
Feature-based Lidar Odometry



Registration: compute relative pose between scans:

- extract features & descriptors
- use descriptors for matching
- compute relative pose

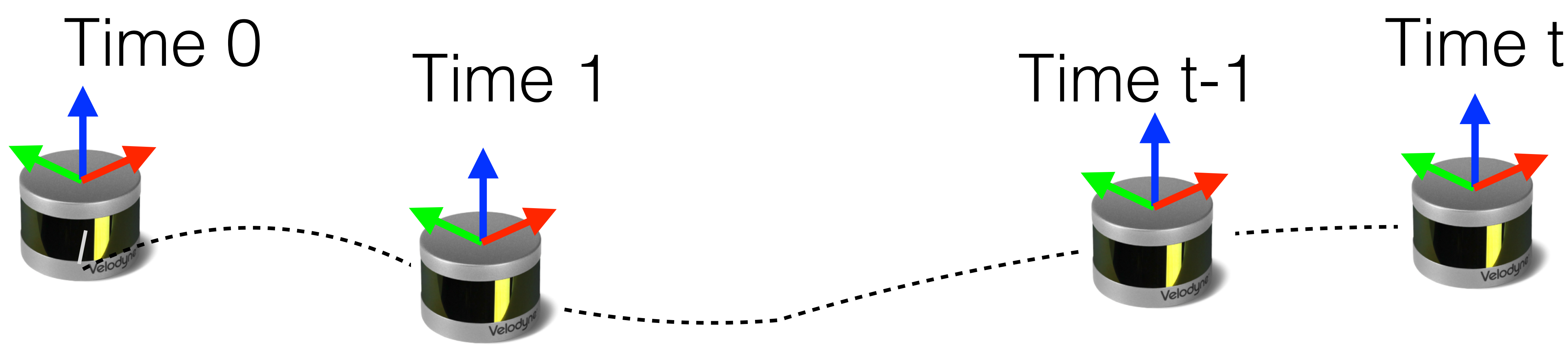
Feature-based Lidar Odometry

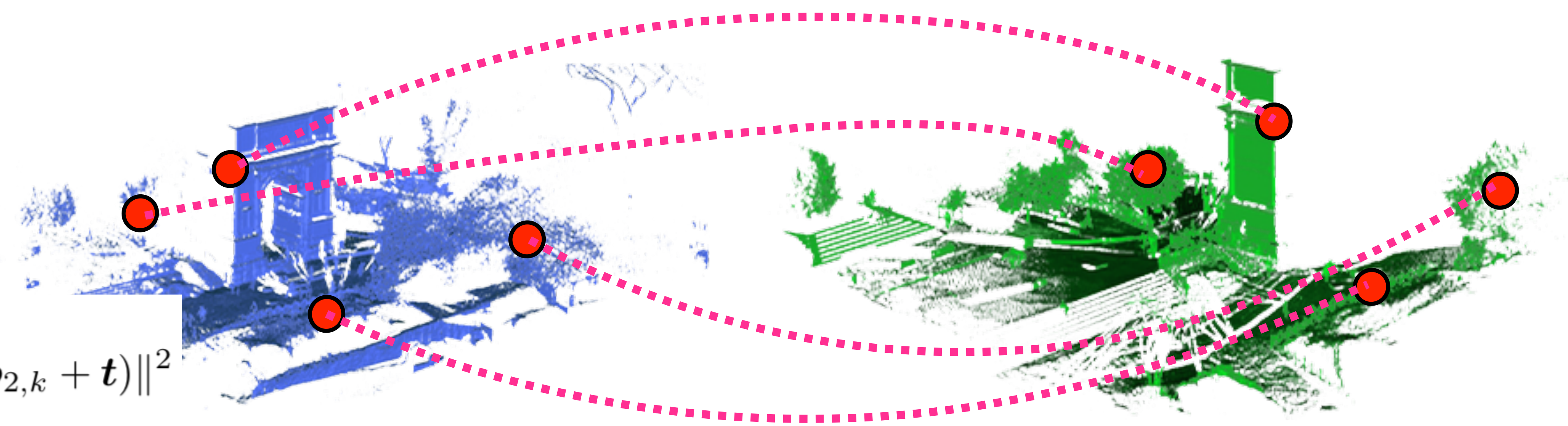


Registration: compute relative pose between scans:

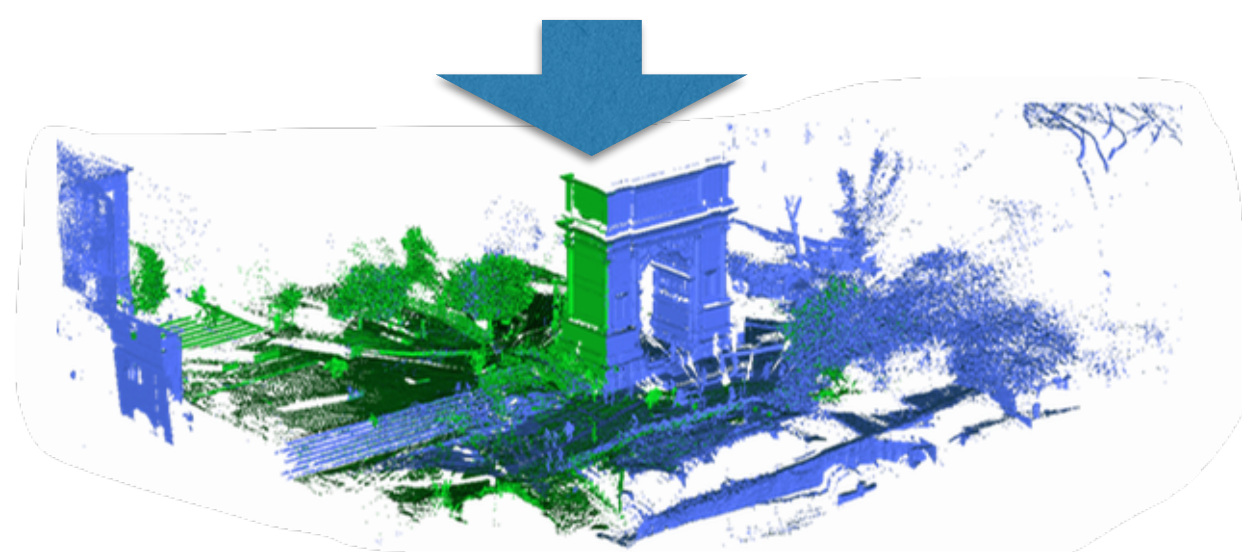
- extract features & descriptors
- use descriptors for matching
- compute relative pose

Feature-based Lidar Odometry





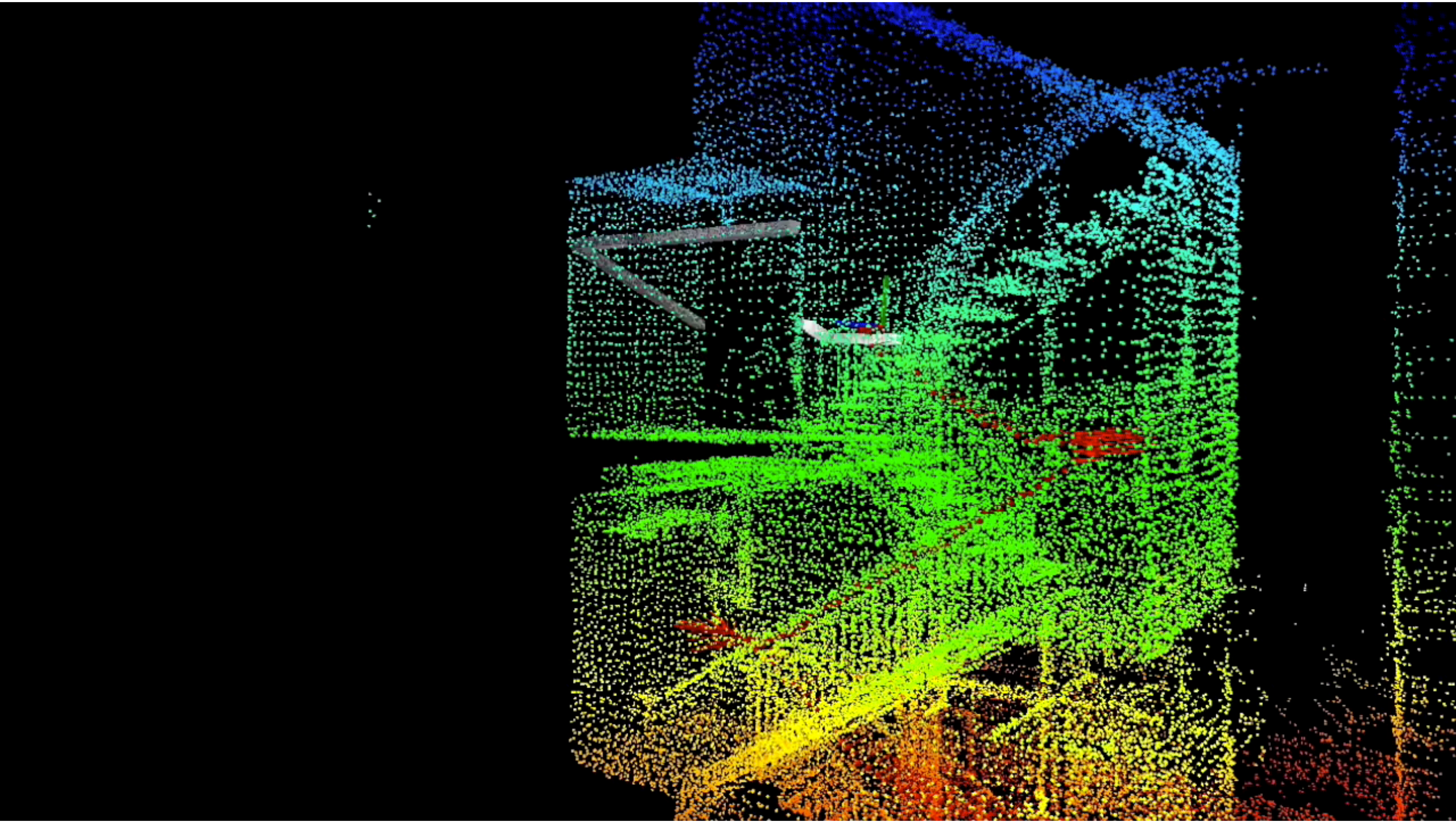
$$\min_{\substack{\boldsymbol{R}\in \mathrm{SO}(3)\\ \boldsymbol{t}\in\mathbb{R}^3}} \sum_{k=1}^{N} \|\boldsymbol{p}_{1,k} - (\boldsymbol{R}\boldsymbol{p}_{2,k} + \boldsymbol{t})\|^2$$



Registration: compute relative pose between scans:

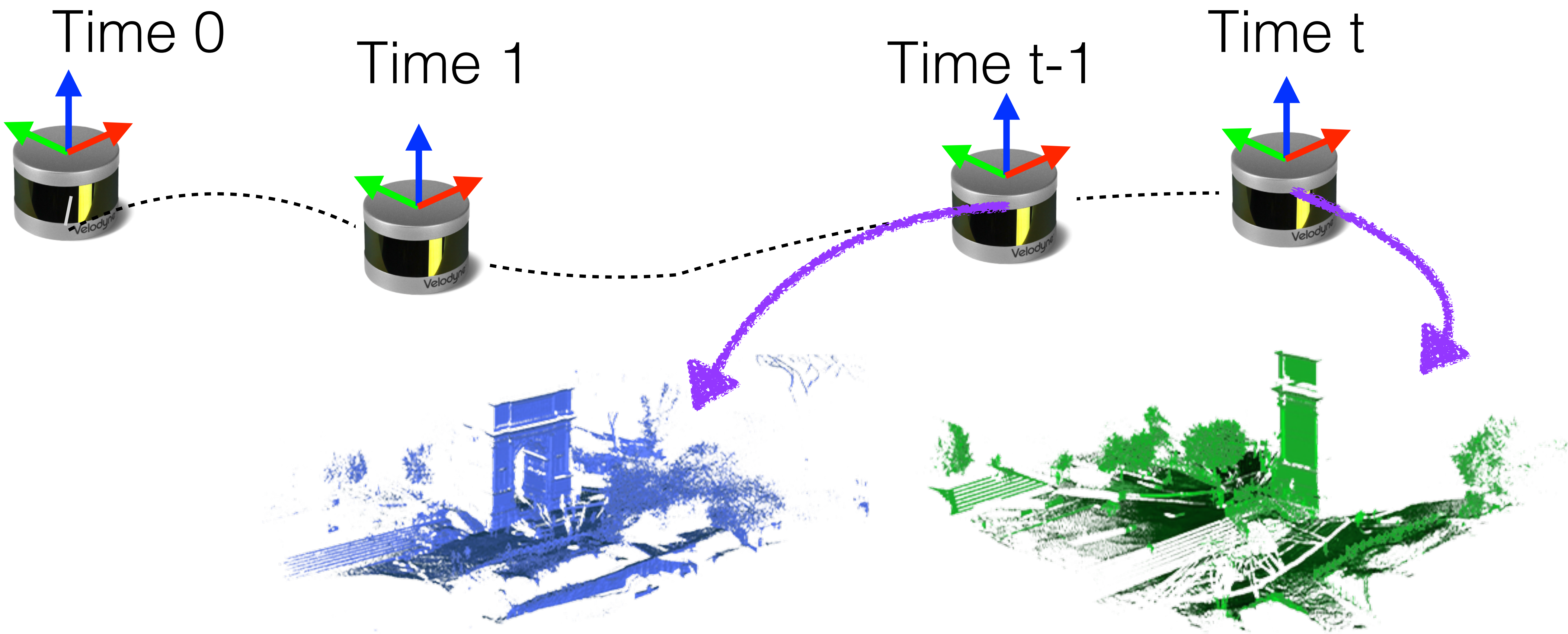
- extract features & descriptors
- use descriptors for matching
- compute relative pose

Feature-based Lidar Odometry



[Zhang and Singh: LOAM: Lidar Odometry and Mapping in Real-time, 2014]

Dense Lidar Odometry



Iterative Closest Point (ICP)

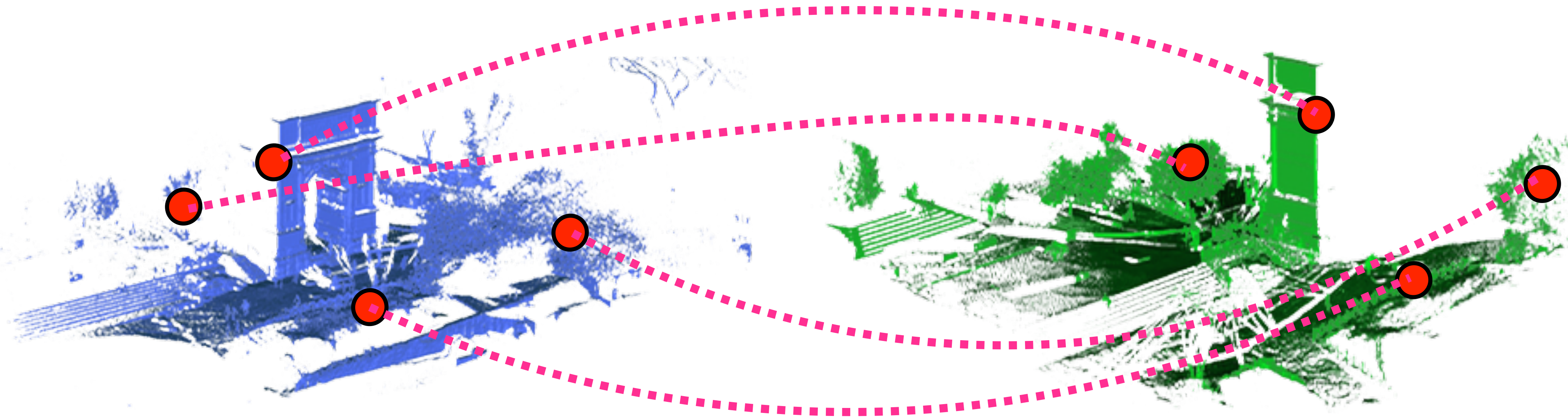
- Alternative to feature-based approaches
- Simultaneous Pose and Correspondences

Iterative Closest Point (ICP)

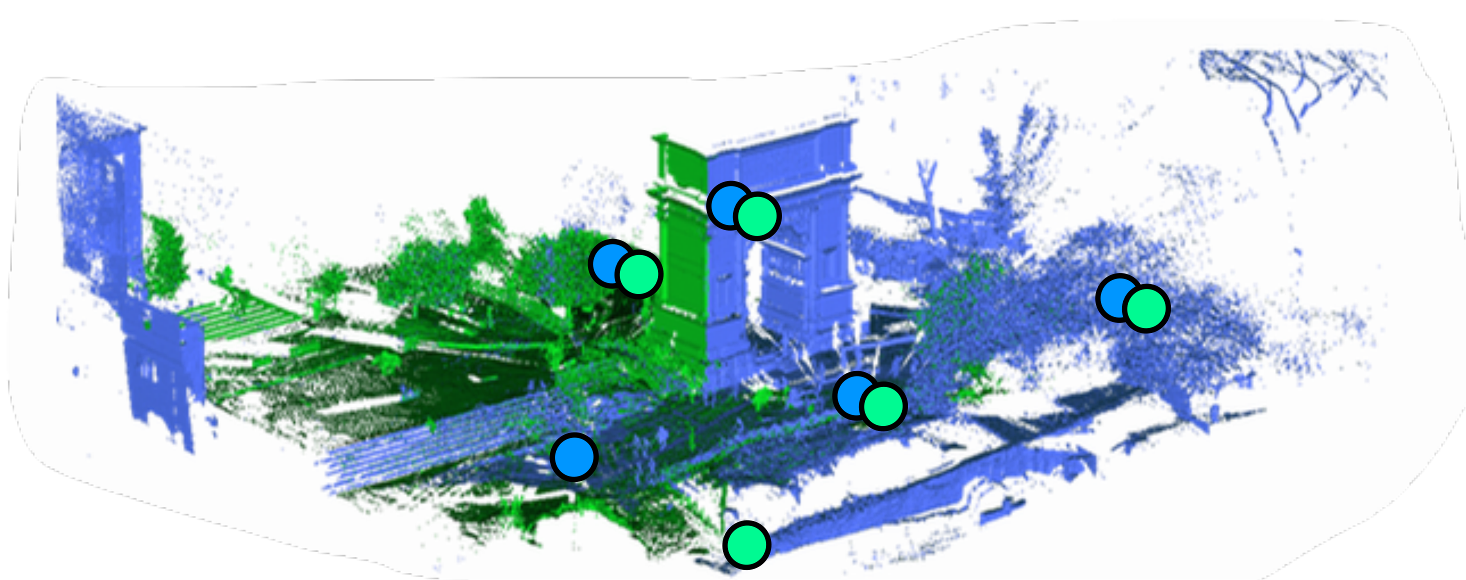
- **Observations:**

$$\min_{\substack{\boldsymbol{R}\in\mathrm{SO}(3)\\ \boldsymbol{t}\in\mathbb{R}^3}} \sum_{k=1}^{N} \|\boldsymbol{p}_{1,k} - (\boldsymbol{R}\boldsymbol{p}_{2,k} + \boldsymbol{t})\|^2$$

1. Easy to compute alignment given ground-truth correspondences



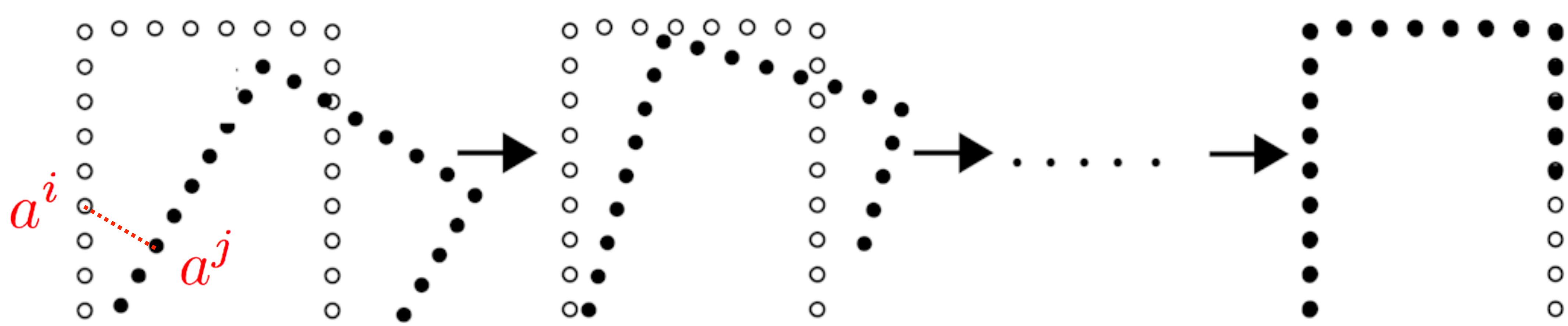
2. Easy to compute correspondences given ground-truth alignment



Iterative Closest Point (ICP)

ICP algorithm: given initial guess, perform the following:

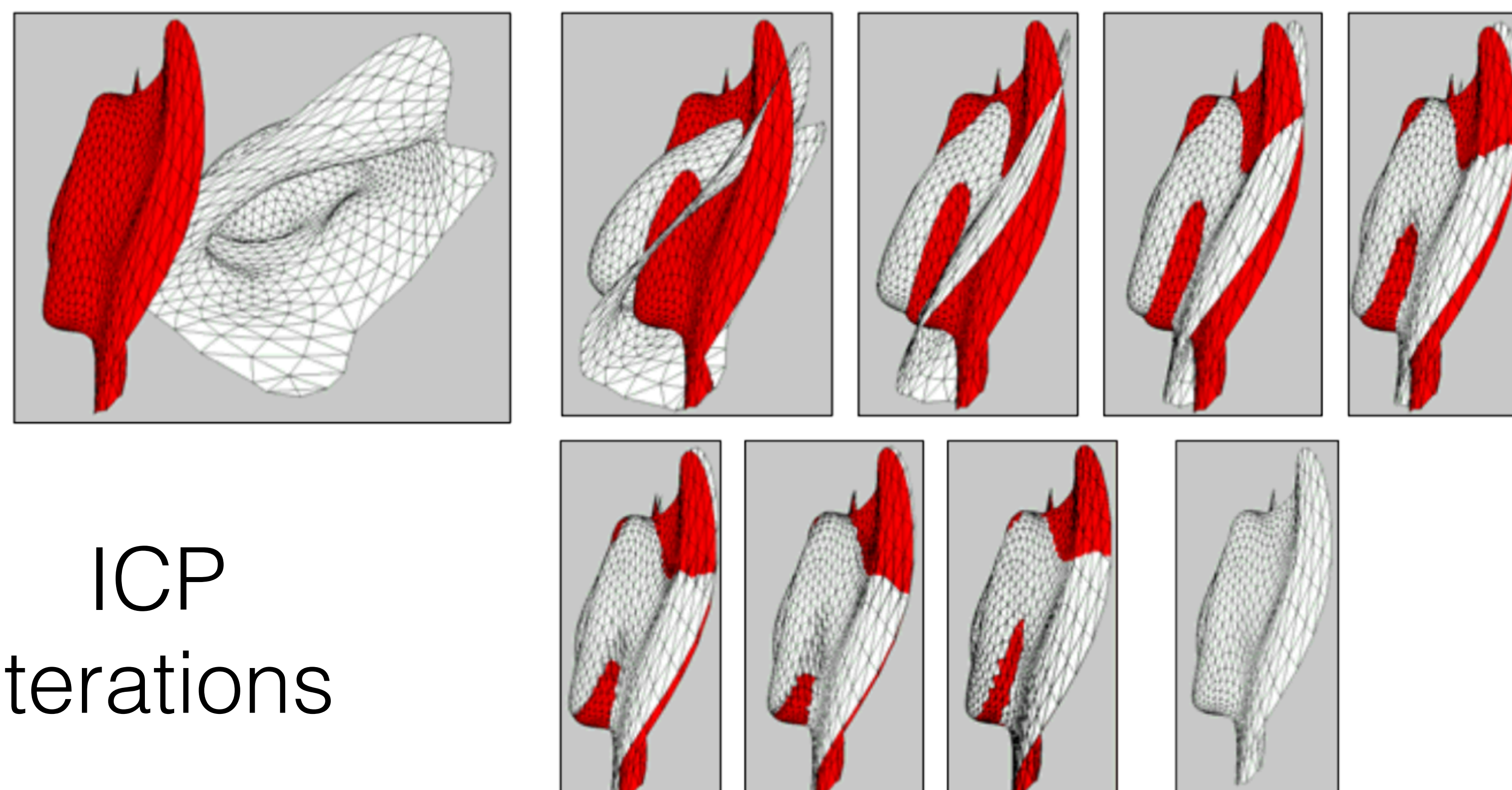
1. **Establish correspondences**: associate to each point in Cloud 1 the closest point in Cloud 2
2. **Compute relative pose given correspondences** (e.g., using Horn's or Arun's method)
3. **Transform point cloud and repeat** (stop when alignment does not improve or after max iter.)



Iterative Closest Point (ICP)

ICP algorithm: given initial guess, perform the following:

1. **Establish correspondences**: associate to each point in Cloud 1 the closest point in Cloud 2
2. **Compute relative pose given correspondences** (e.g., using Horn's or Arun's method)
3. **Transform point cloud and repeat** (stop when alignment does not improve or after max iter.)



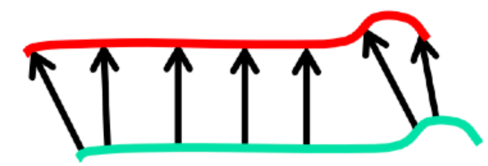
ICP Iterations

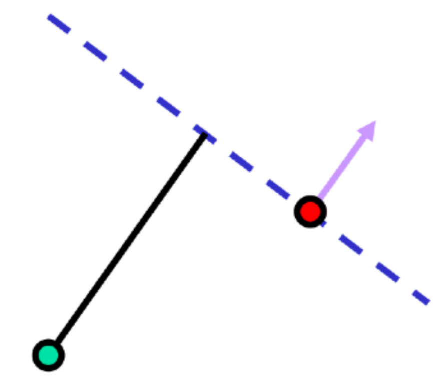
[courtesy: http://www.cs.technion.ac.il/~cs236329/tutorials/ICP.pdf]

Iterative Closest Point (ICP): Issues and Extensions

Extensions

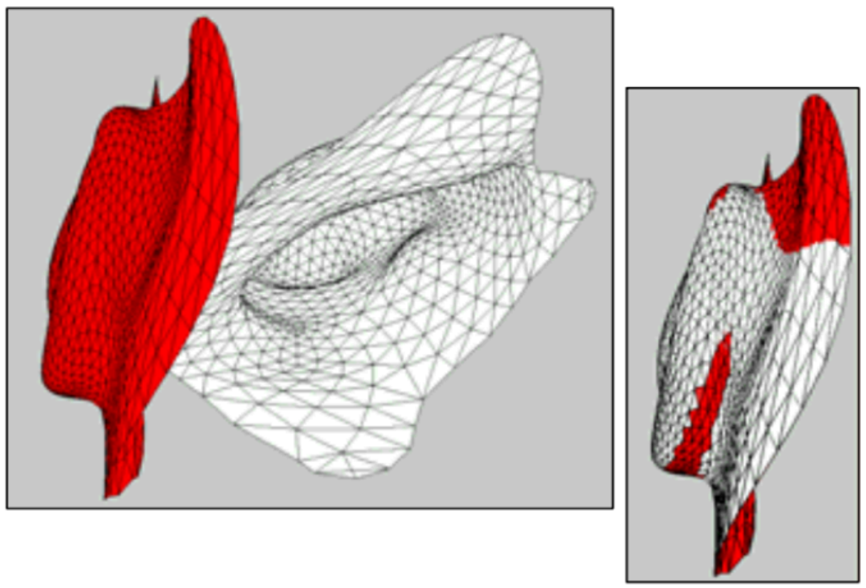
- Kd-tree spatial subdivision
- Different error metrics (e.g., point to plane)
- Reject outliers



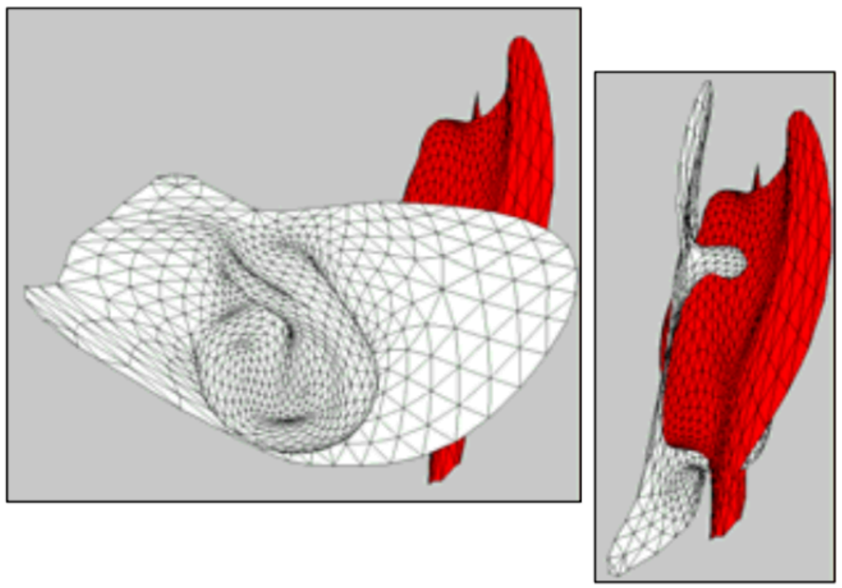


Local convergence

Initial guess 1

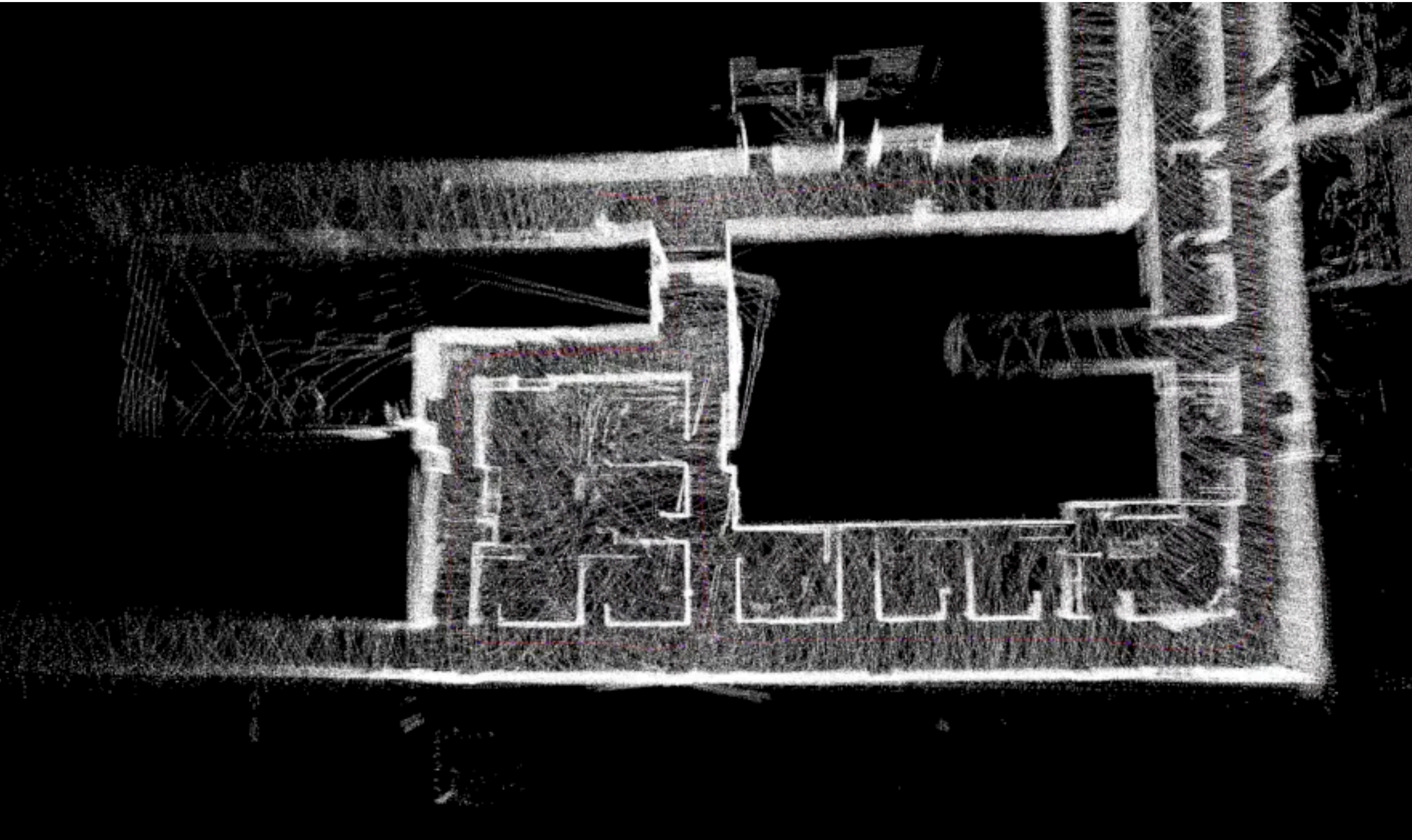


Initial guess 2



[courtesy: http://www.cs.technion.ac.il/~cs236329/tutorials/ICP.pdf]

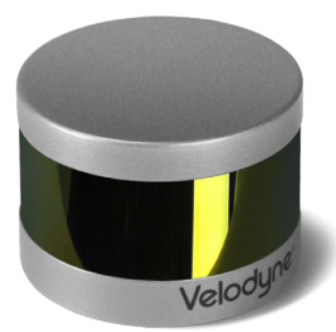
ICP-based SLAM: Failure Mode



DARPA Subterranean Challenge, in collaboration with JPL

Beyond Cameras

- wheel odometry
- GPS
- Lidar
- Inertial Measurement Unit (IMU)
- Event Cameras





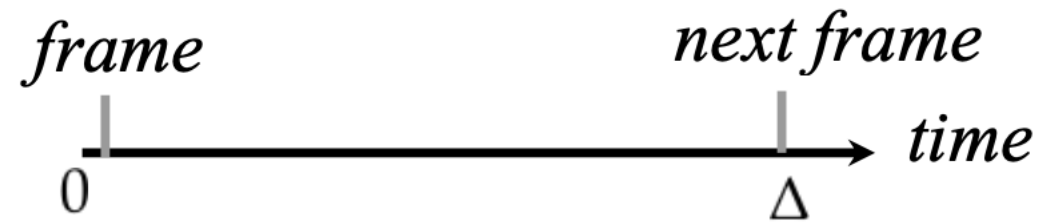
830g	160g	4g	3g
8 W	2.5 W	0.3W	~1 W

Event-based Vision: A Survey

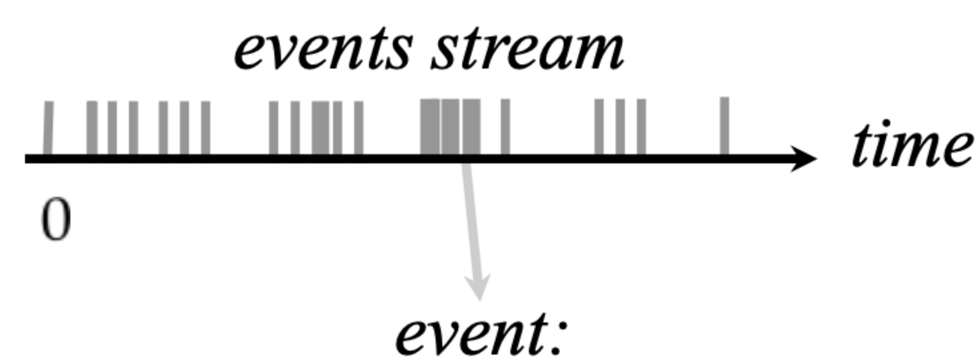
Guillermo Gallego, Tobi Delbrück, Garrick Orchard, Chiara Bartolozzi, Brian Taba, Andrea Censi, Stefan Leutenegger, Andrew J. Davison, Jörg Conradt, Kostas Daniilidis, Davide Scaramuzza

Event-based Cameras

- Speed of robot is constrained by speed at which it can sense (and compute)
- Common cameras: 20-120fps



- event-based cameras (e.g., Dynamic Vision Sensor, DVS)
  - Temporal resolution: 1 μs
  - High dynamic range: 120 dB
  - Low power: 20 mW
  - Cost: 2,500 EUR

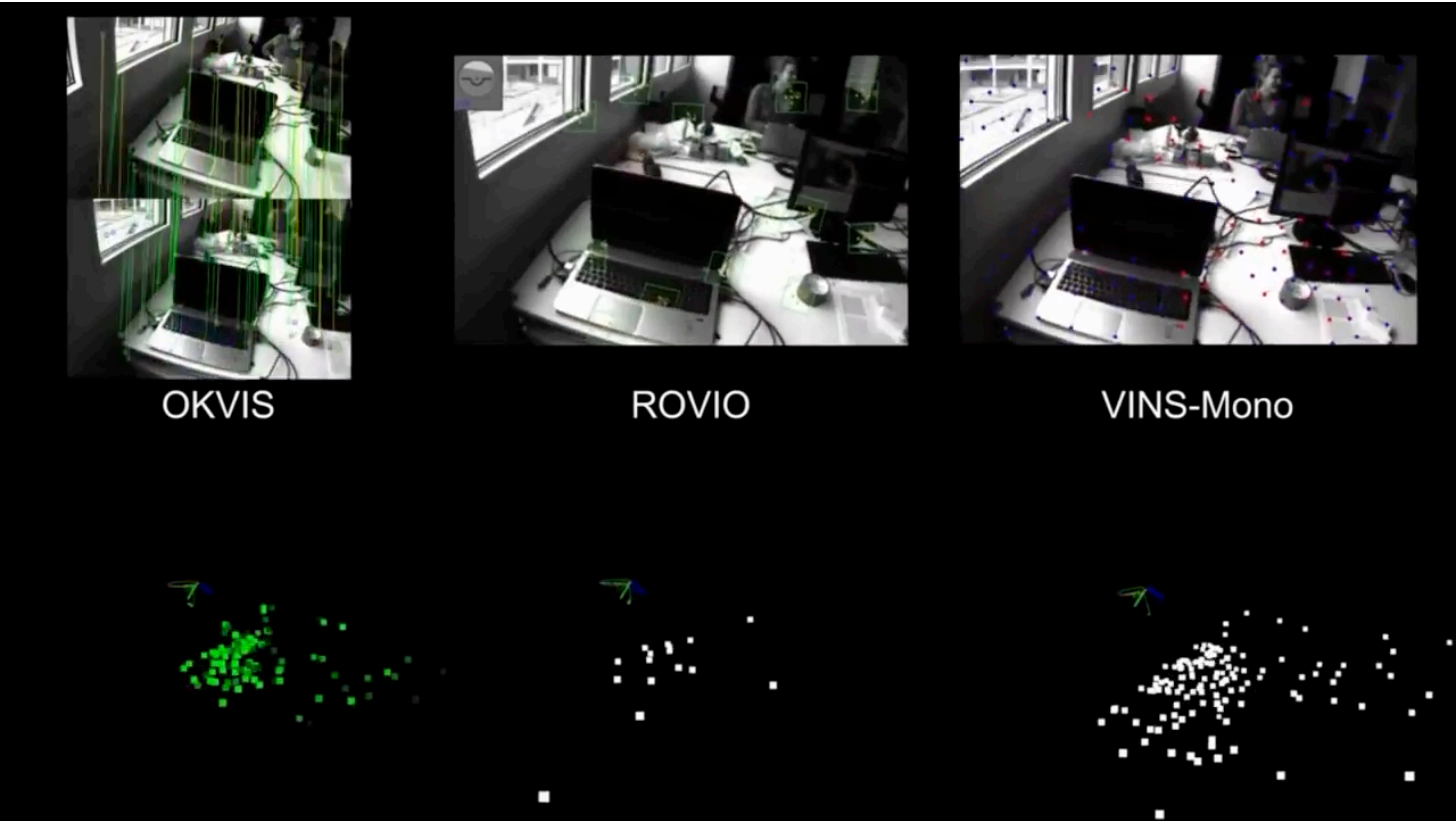


$$\left\langle t, \langle x, y \rangle, sign\left(\frac{d}{dt}\log(I_t(x,y))\right)\right\rangle$$

Event-based Cameras

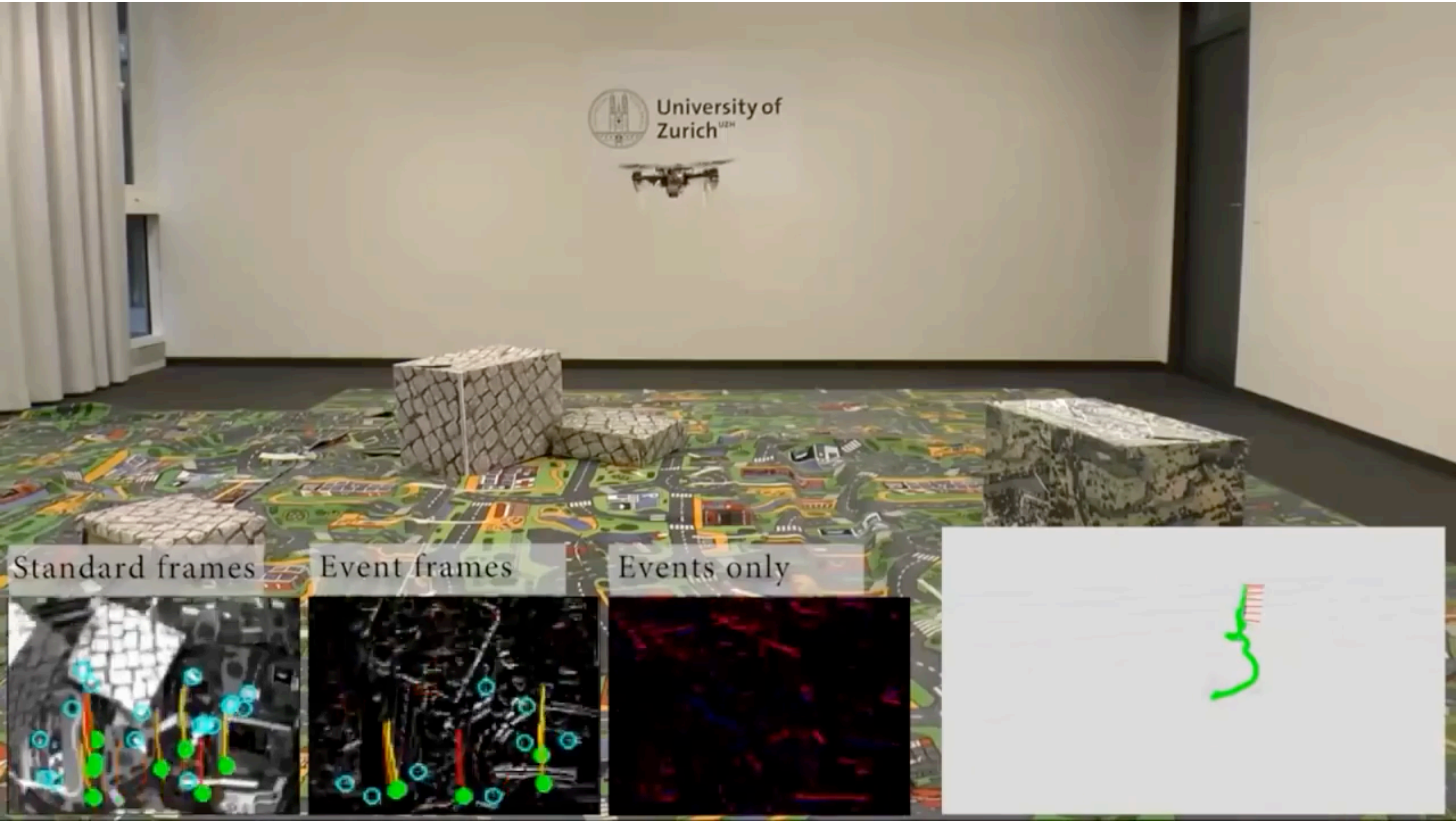
Scaramuzza et al. 2014: Neuromorphic camera

Event-based Cameras for SLAM



Antoni Rosinol Vidal, Henri Rebecq, Timo Horstschaefer, Davide Scaramuzza Ultimate SLAM? Combining Events, Images, and IMU for Robust Visual SLAM in HDR and High Speed Scenarios R-AL 2018.

Event-based Cameras for SLAM



Antoni Rosinol Vidal, Henri Rebecq, Timo Horstschaefer, Davide Scaramuzza Ultimate SLAM? Combining Events, Images, and IMU for Robust Visual SLAM in HDR and High Speed Scenarios R-AL 2018.