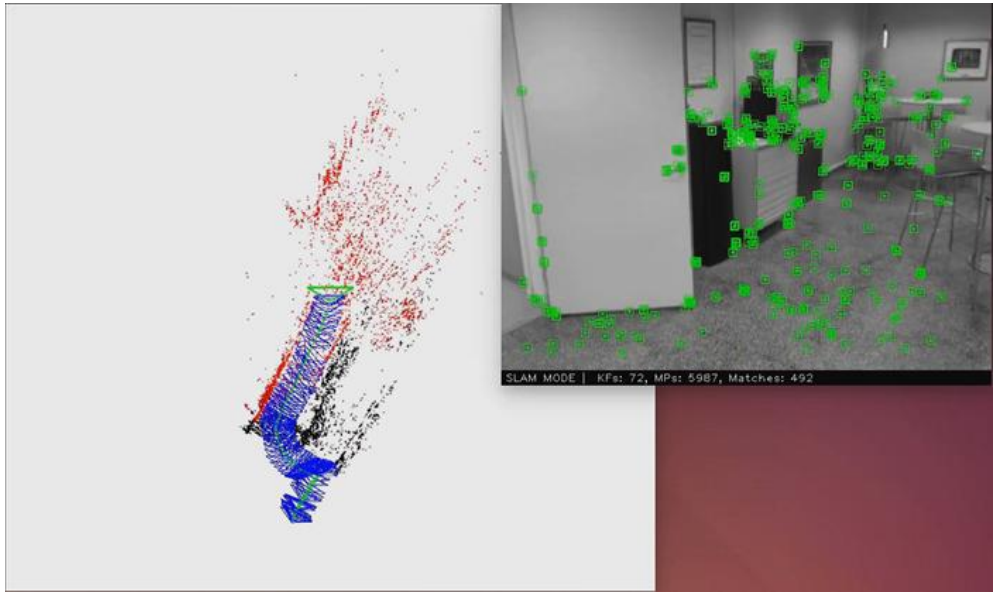
UiO : Department of Technology Systems
University of Oslo

Lecture 13
Visual SLAM and computer vision applications

Trym Vegard Haavardsholm



TEK5030

Today

1. What is Visual SLAM?

2. Short-term, mid-term and long-term tracking
3. Mapping and sensor fusion with factor graphs

4. VSLAM systems

Part I

WHAT IS VISUAL SLAM?

What is SLAM?

What is SLAM?

Simultaneous localisation and mapping

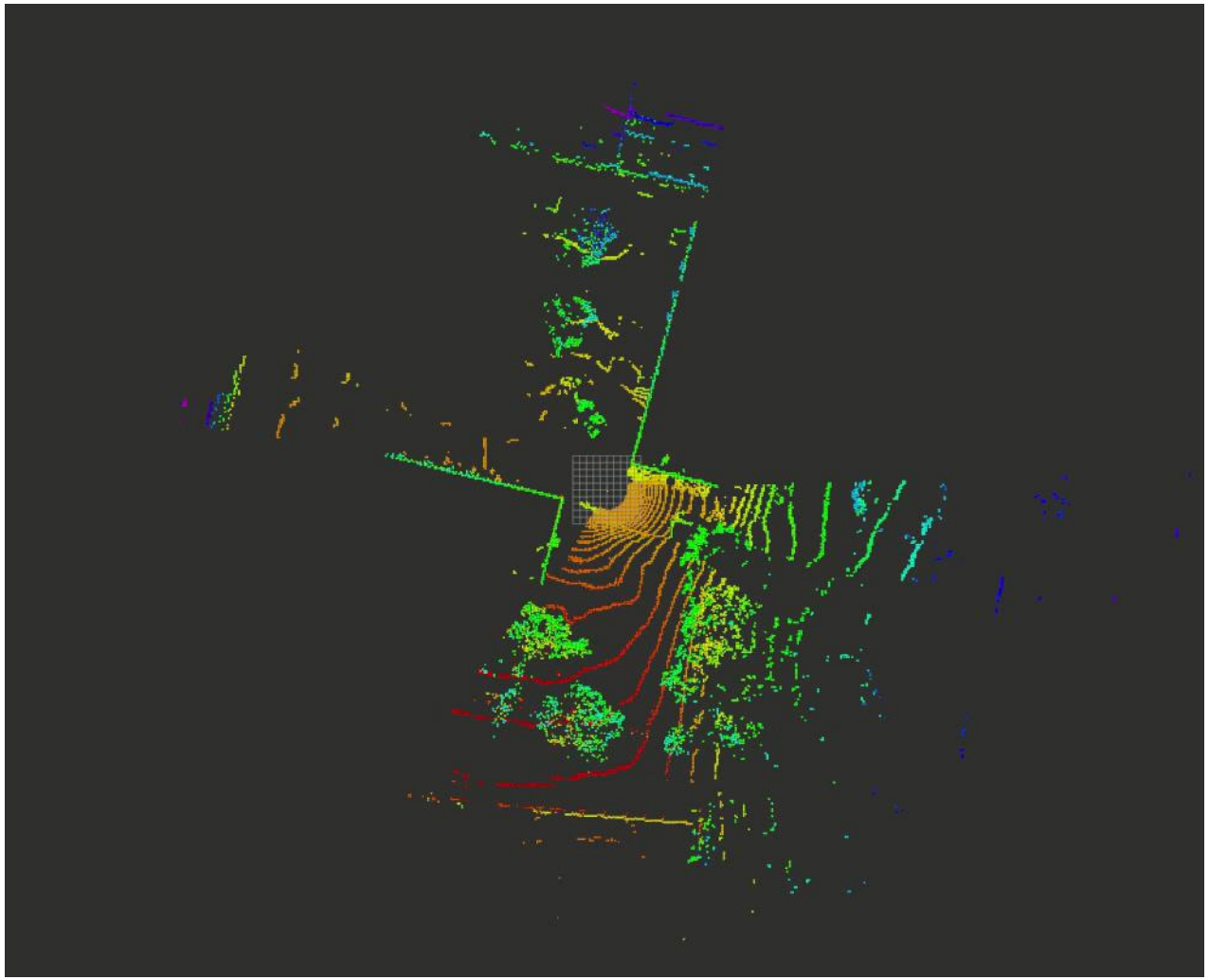
What is SLAM?

Simultaneous localisation and mapping

Simultaneous

- estimation of the state of a robot using on-board sensors

- construction of a map of the environment that the sensors are perceiving



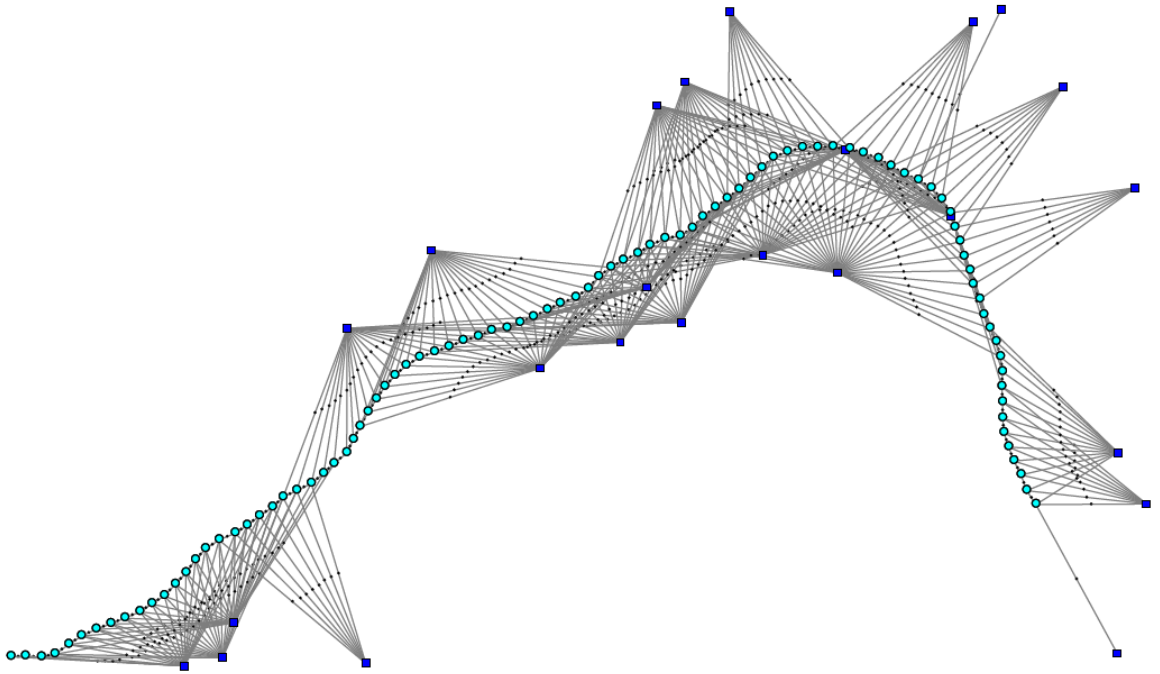
What is SLAM?

Simultaneous localisation and mapping

Simultaneous

- **mapping**:
  Continuously expanding
  and optimising a consistent map
  while exploring the environment

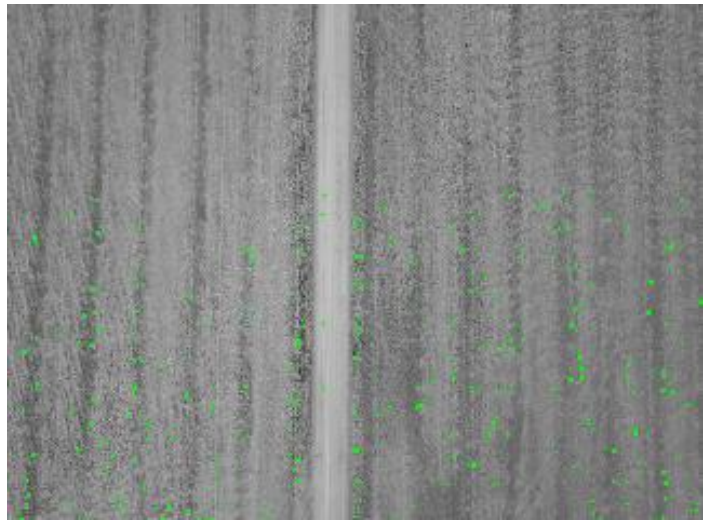
- **localisation**:
  Localisation within the map

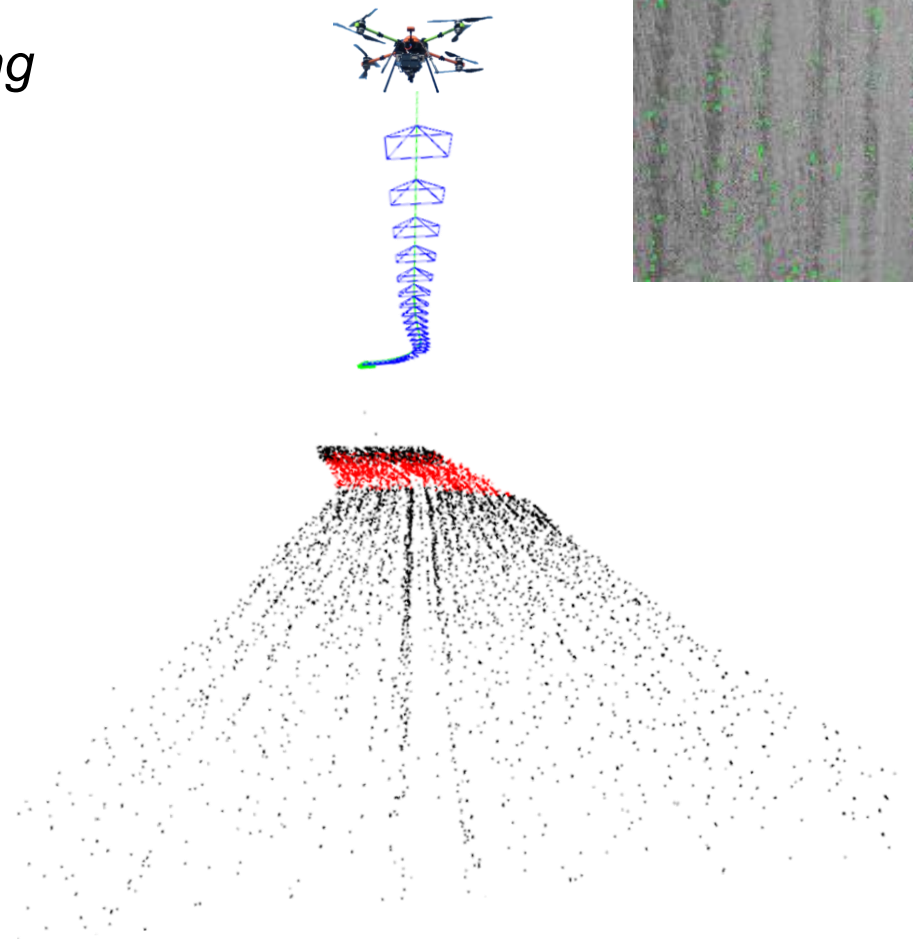


Jing Dong “GTSAM 4.0 Tutorial” License CC BY-NC-SA 3.0

What is Visual SLAM?

Visual simultaneous localisation and mapping





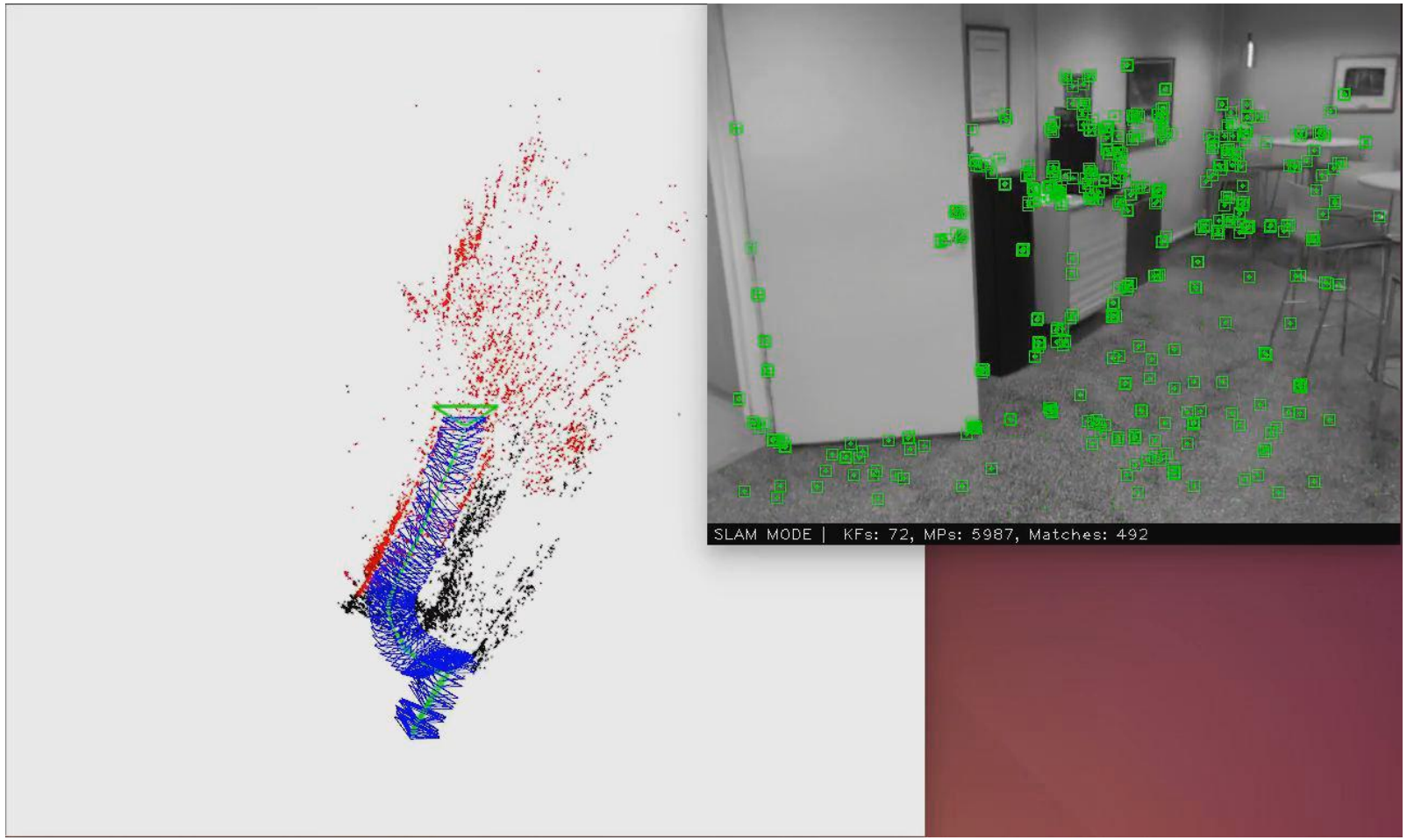
Simultaneous

- **mapping**:
  Continuously expanding
  and optimising a consistent map
  while exploring the environment

- **localisation (tracking)**:
  Localisation within the map
  (tracking the map in image frames)

How can we do visual SLAM?

Visual SLAM example



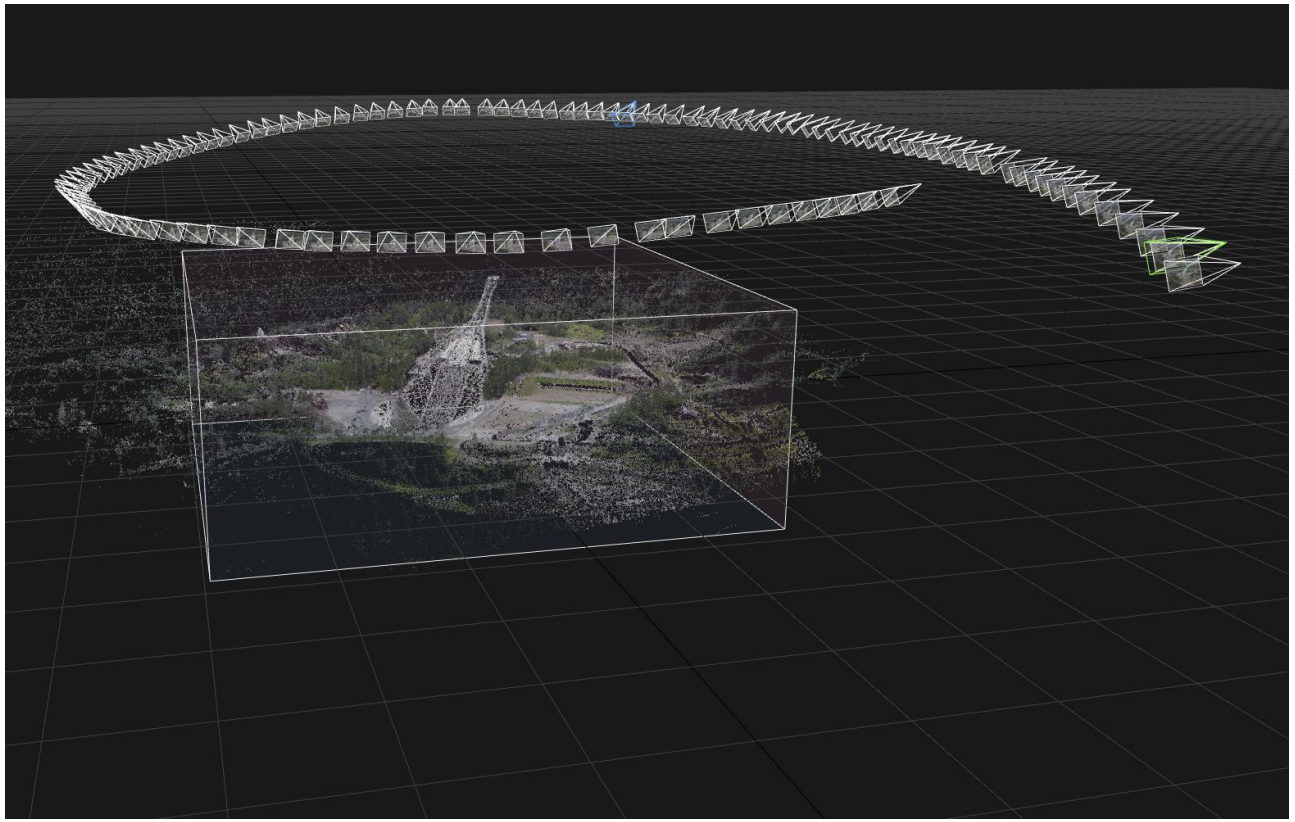
What is the map?

What is the map?

A model of the environment that lets us

- limit the localisation error
  by recognising previously visited areas

- (support other tasks,
  such as obstacle avoidance
  and path planning)



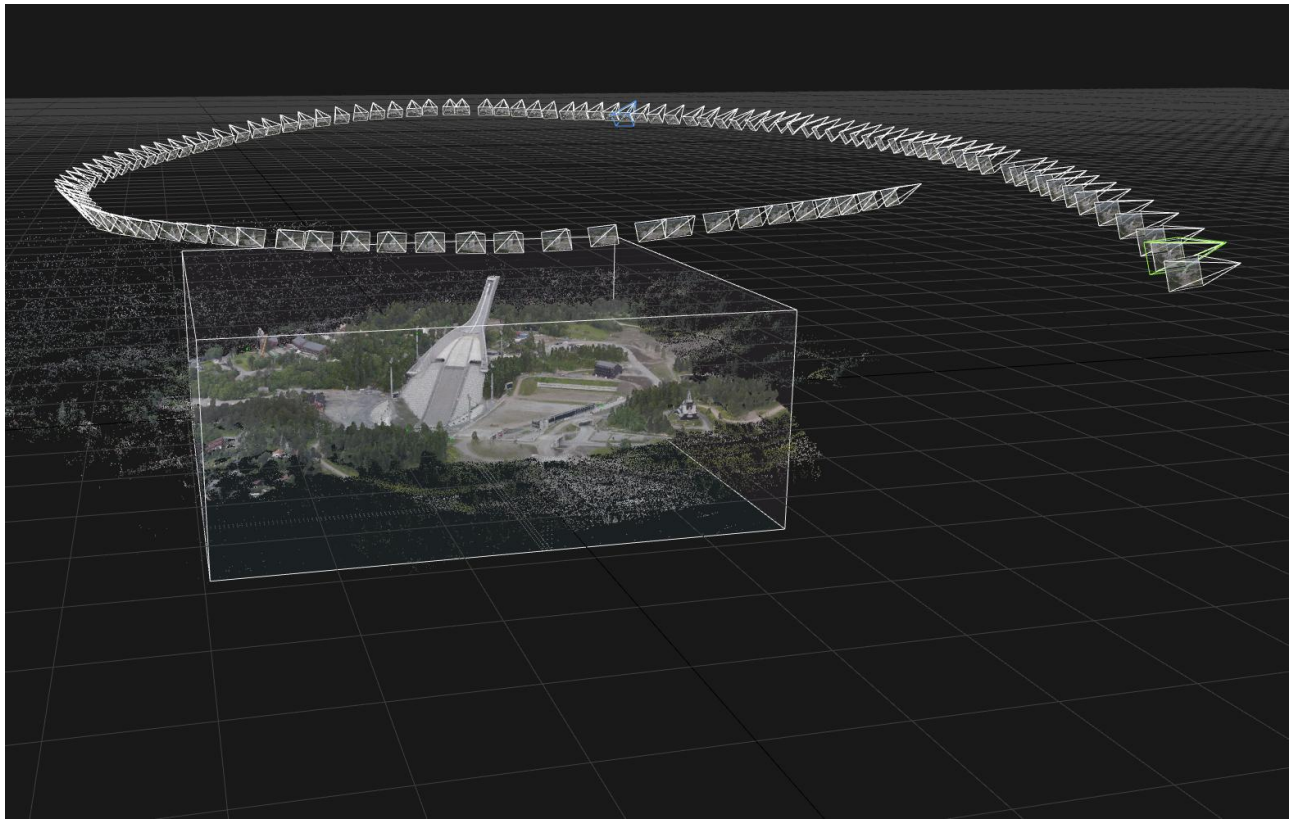
What is the map?

A model of the environment that lets us

- limit the localisation error
  by recognising previously visited areas

- (support other tasks,
  such as obstacle avoidance
  and path planning)

  Maybe best left as auxiliary processing?



Examples of map representations

Feature-based metric maps

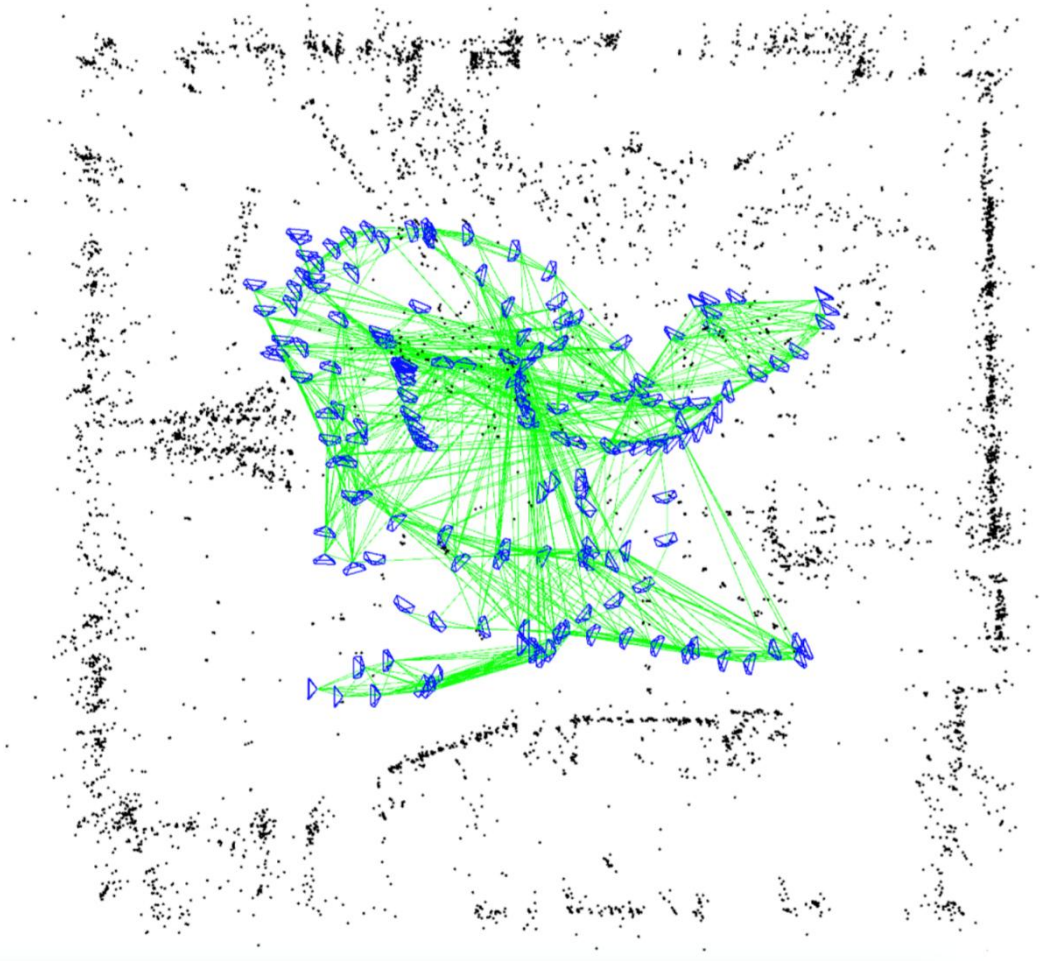


Image: Cadena, C., et al. (2016). Past, Present, and Future of Simultaneous Localization and Mapping: Toward the Robust-Perception Age. *IEEE Transactions on Robotics*, *32*(6), 1309–1332

Representation based on:

- Graphs over *keyframes*
- Kd-trees over *points*

Mur-Artal, R., Montiel, J. M. M., & Tardos, J. D. (2015). ORB-SLAM: A Versatile and Accurate Monocular SLAM System. IEEE Transactions on Robotics, 31(5), 1147–1163. https://doi.org/10.1109/TRO.2015.2463671

Examples of map representations

Dense metric maps

DTAM:
Dense Tracking and Mapping in Real-Time



Image: Cadena, C., et al. (2016). Past, Present, and Future of Simultaneous Localization and Mapping: Toward the Robust-Perception Age. *IEEE Transactions on Robotics*, *32*(6), 1309–1332

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

Examples of map representations

Dense metric maps

DTAM:
Dense Tracking and Mapping in Real-Time

Representation example:



https://voxblox.readthedocs.io/en/latest/

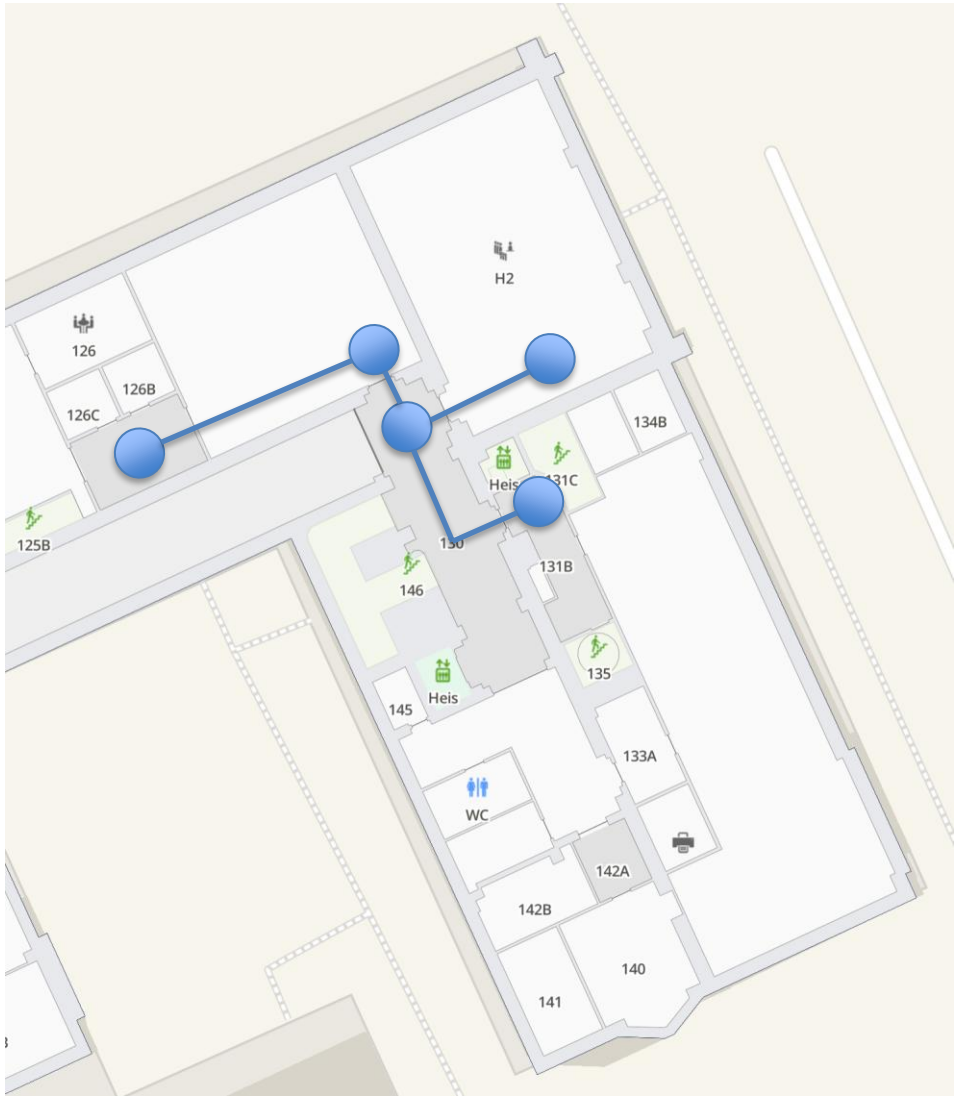


Image: Cadena, C., et al. (2016). Past, Present, and Future of Simultaneous Localization and Mapping: Toward the Robust-Perception Age. *IEEE Transactions on Robotics*, *32*(6), 1309–1332

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

Examples of map representations

Topological maps



Examples of map representations

Topological maps

FABMAP



Image: YouTube: ORI - Oxford Robotics Institute

Cummins, M., & Newman, P. (2008). FAB-MAP: Probabilistic Localization and Mapping in the Space of Appearance. The International Journal of Robotics Research, 27(6), 647–665

Examples of map representations

Topological-metric maps



Examples of map representations

Topological-metric maps

[Visual Teach & Repeat]

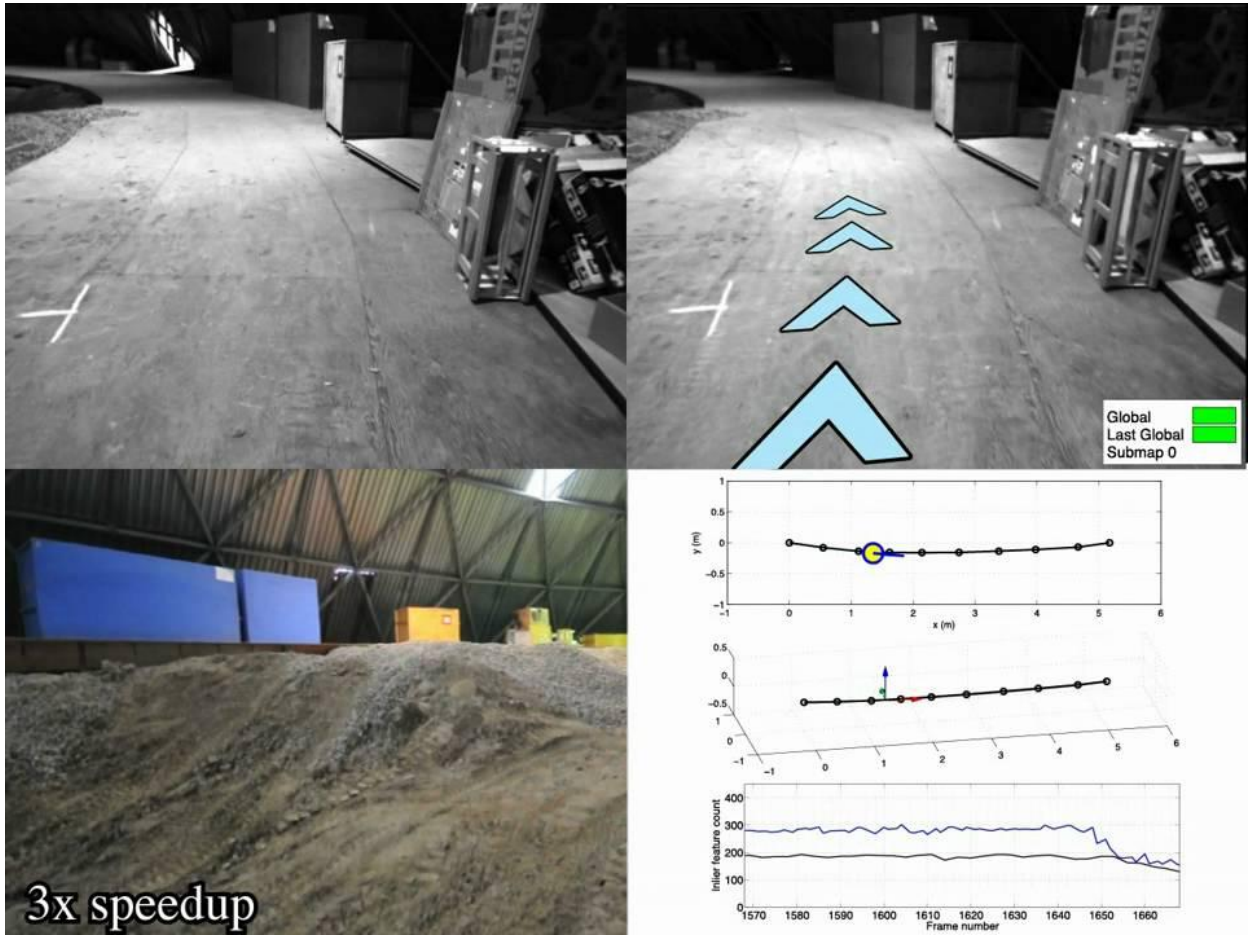
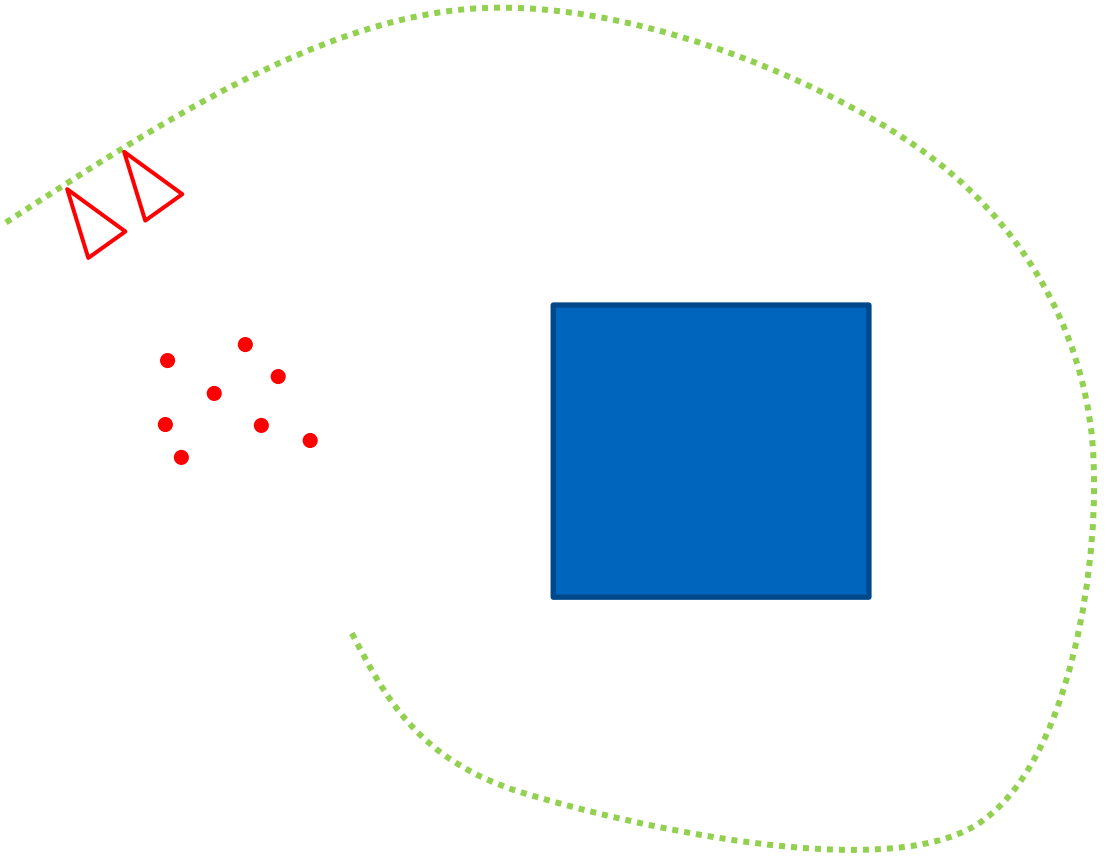


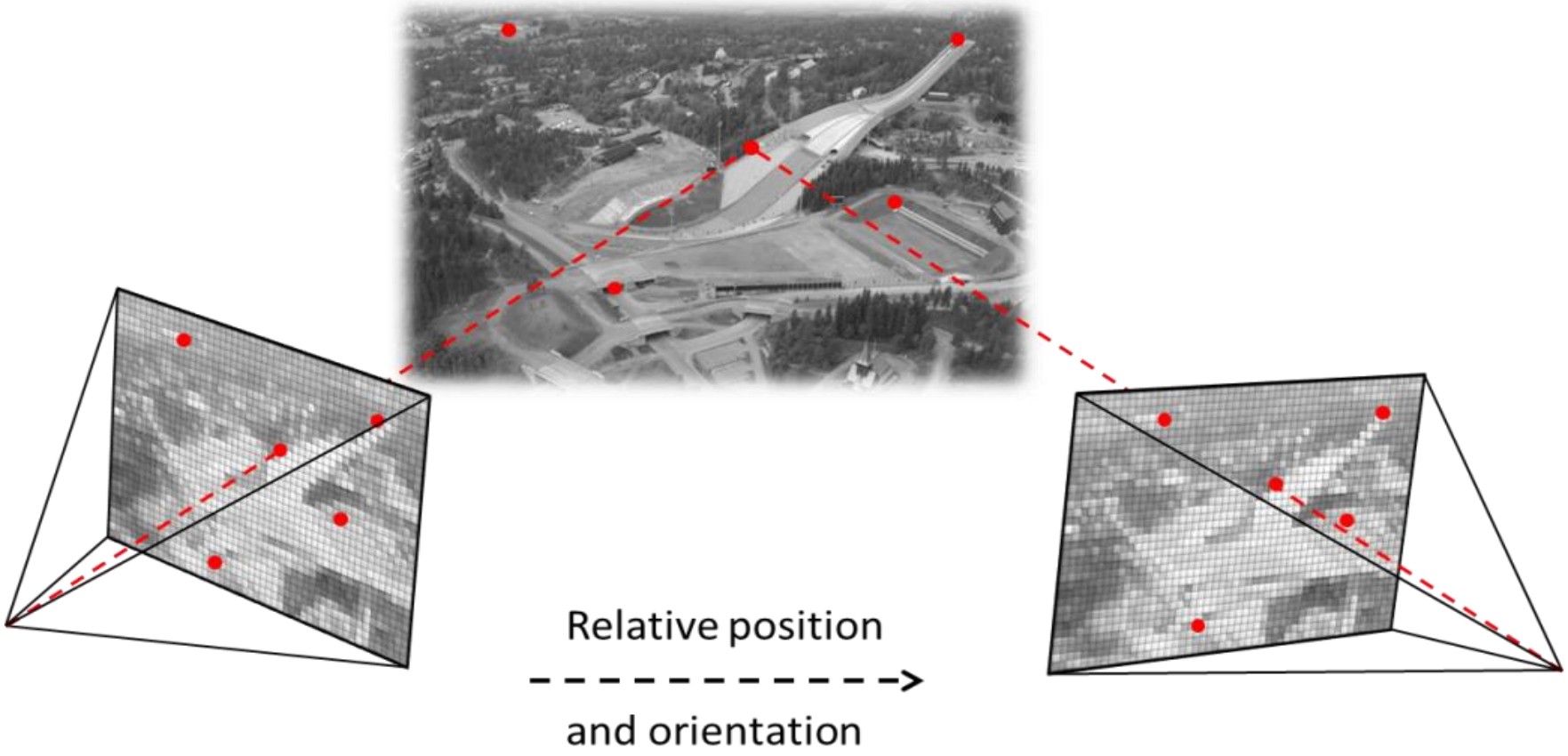
Image: YouTube: utiasASRL

Furgale P T and Barfoot T D. Visual Teach and Repeat for Long-Range Rover Autonomy.
Journal of Field Robotics, special issue on Visual mapping and navigation outdoors, 27(5): 534-560, 2010.

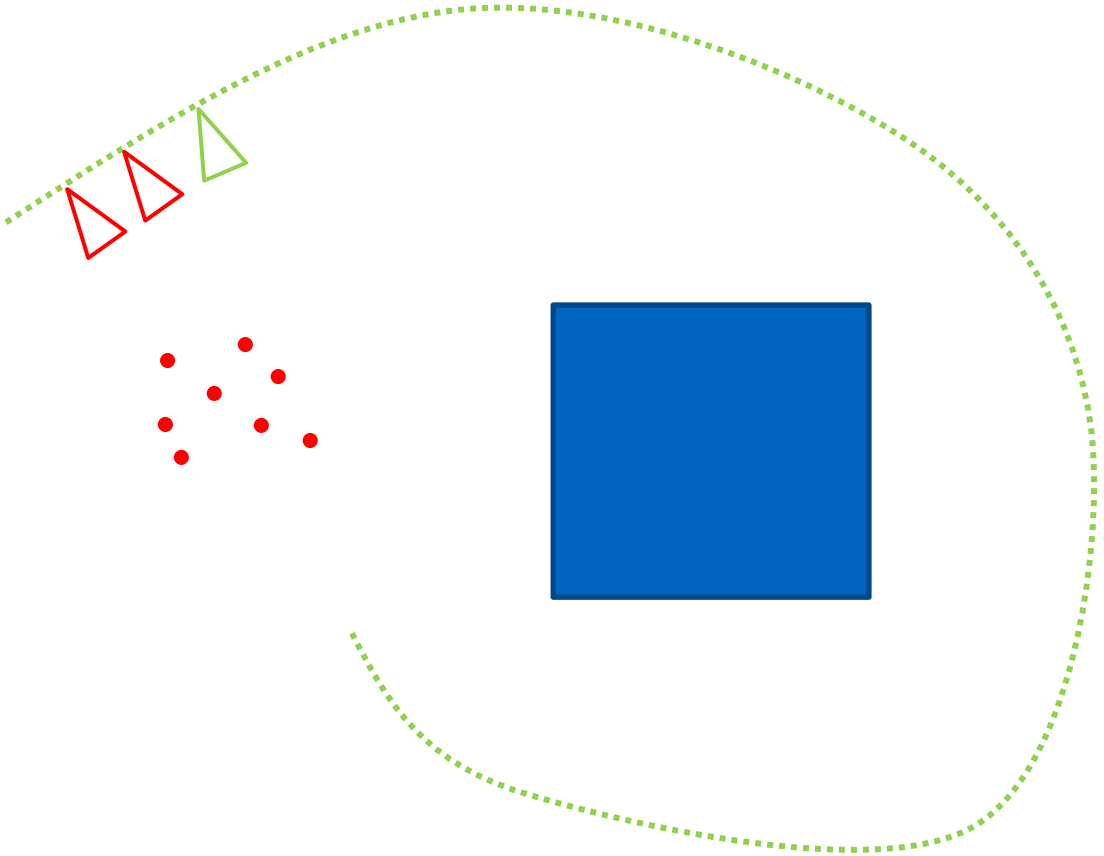
How do we build a map?



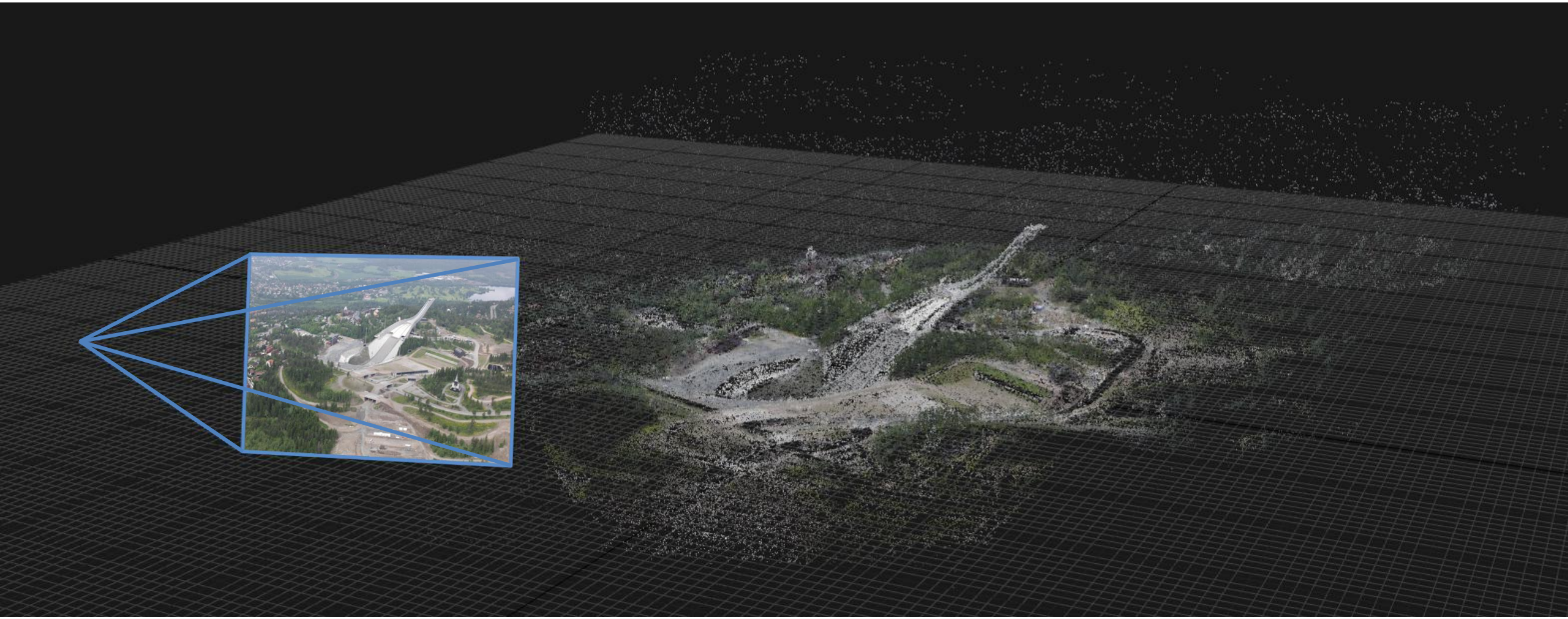
Relative pose and 3D from two views



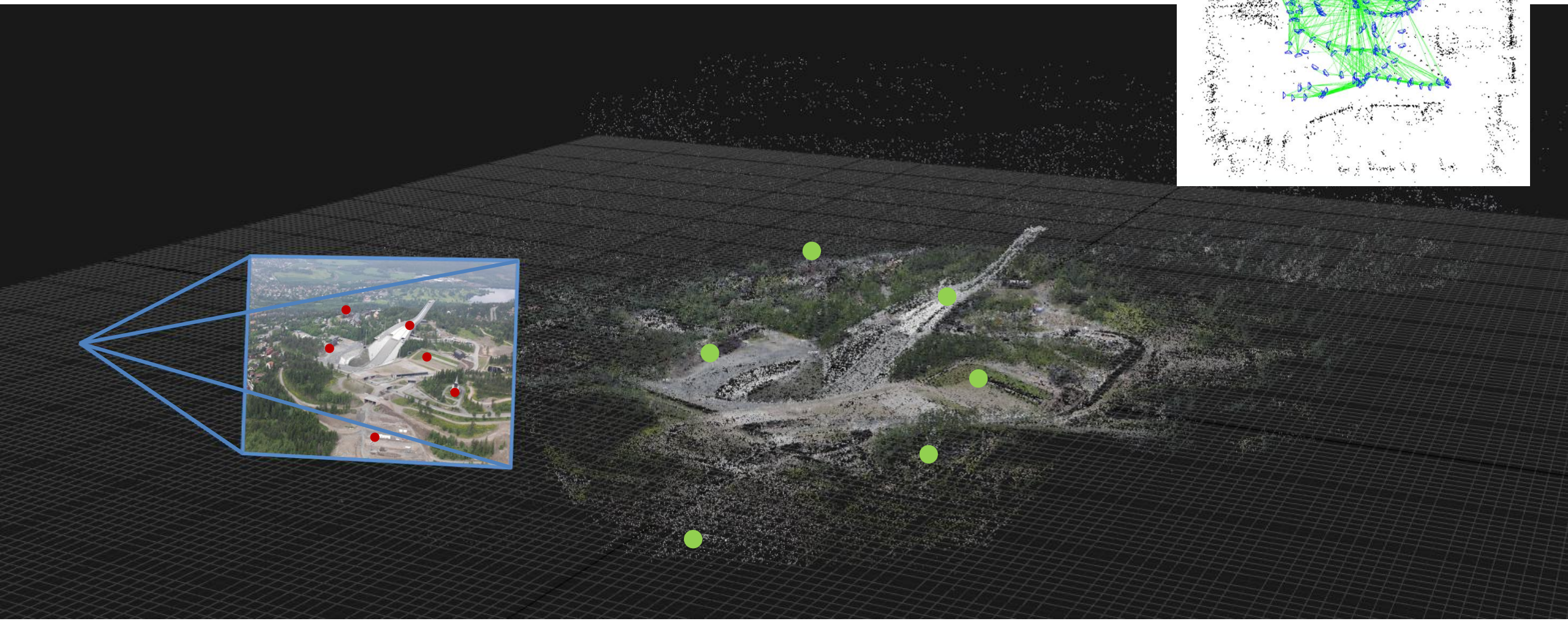
How do we track a map?



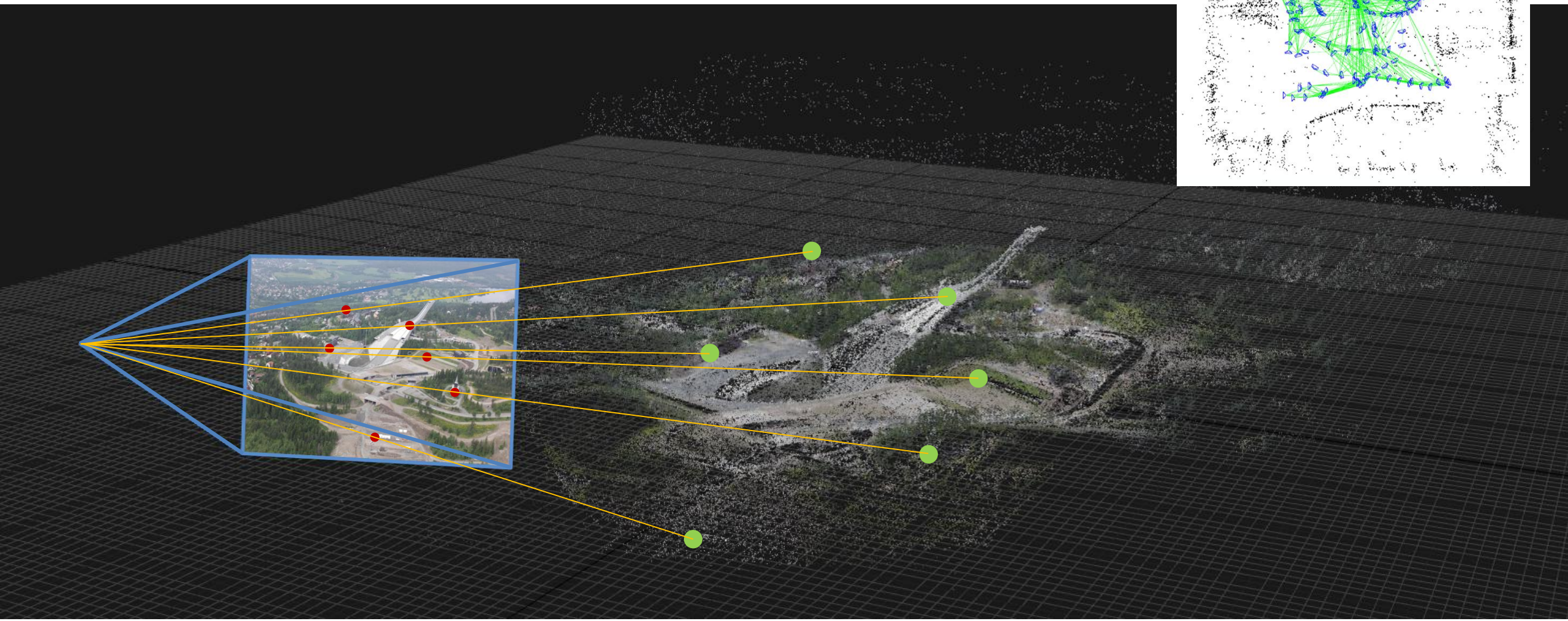
Pose from known 3D map



Pose from point correspondences



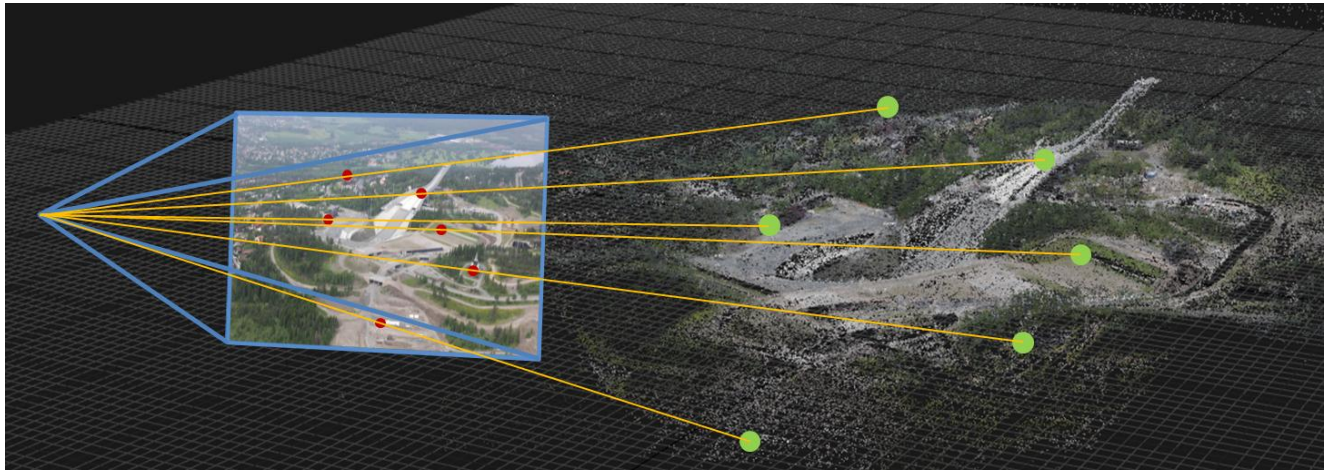
Pose from point correspondences



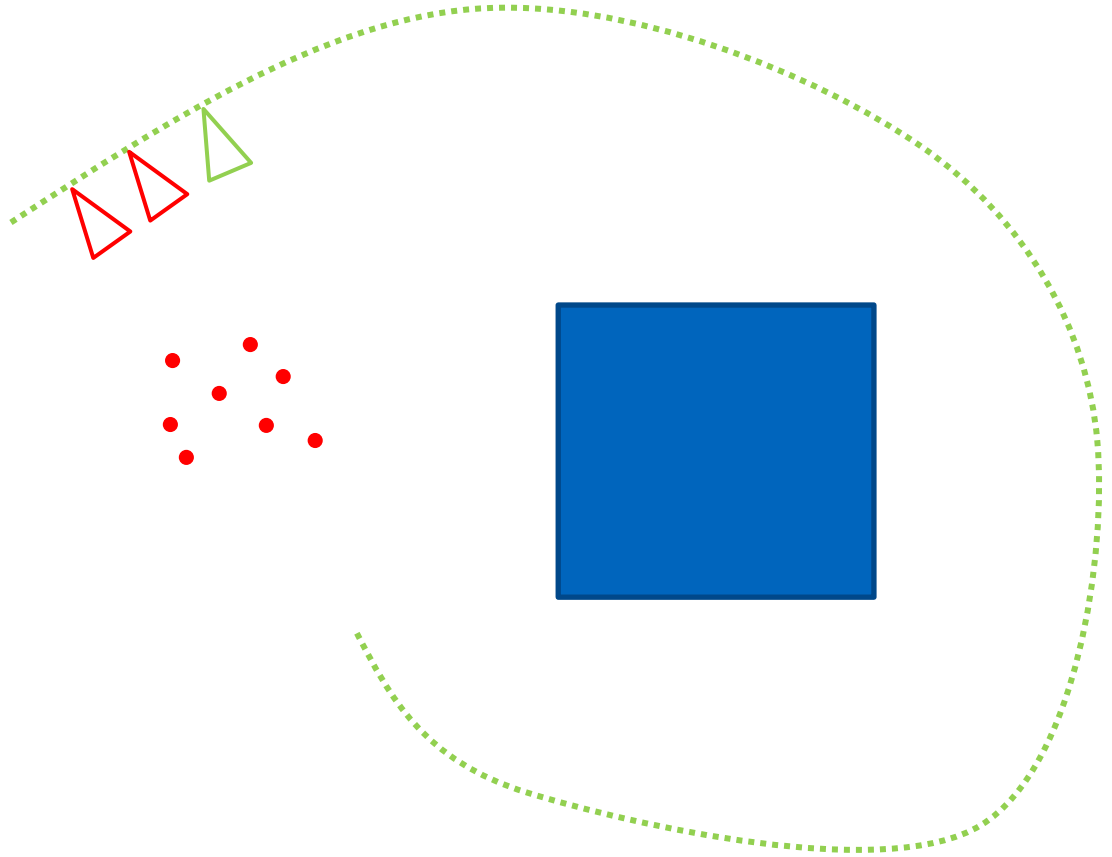
Pose from point correspondences

Minimise *geometric error*

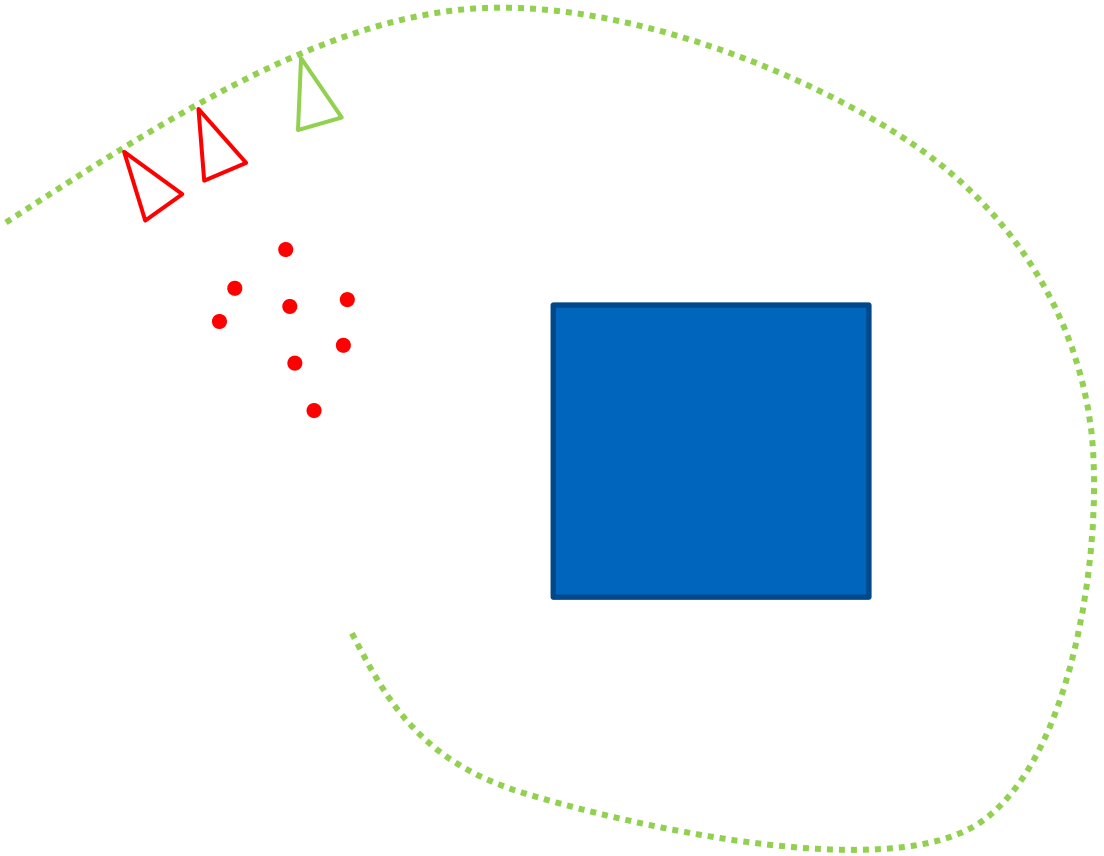
$$\mathbf{T}_{wc}^{*} = \underset{\mathbf{T}_{wc}}{\operatorname{argmin}} \sum_{i} \left\| \pi(\mathbf{T}_{wc}^{-1} \cdot \mathbf{x}_{i}^{w}) - \mathbf{u}_{i} \right\|^{2}$$



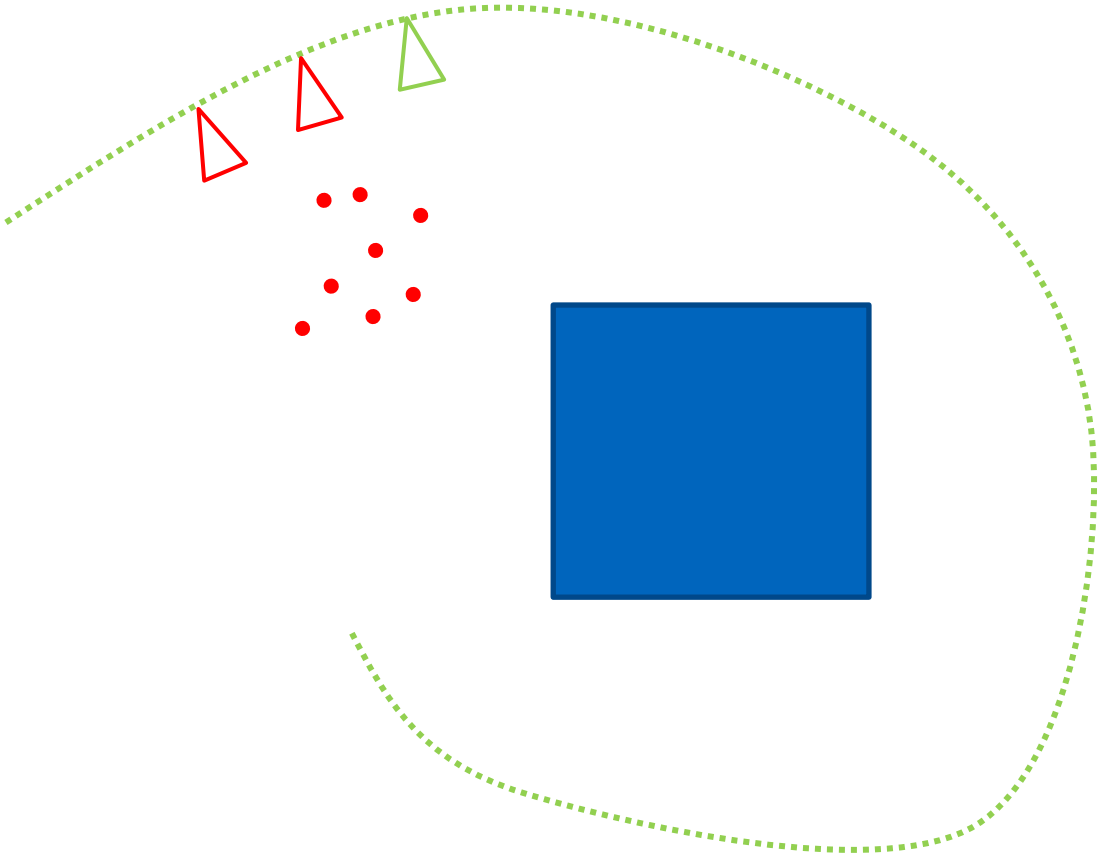
Map initialisation and tracking



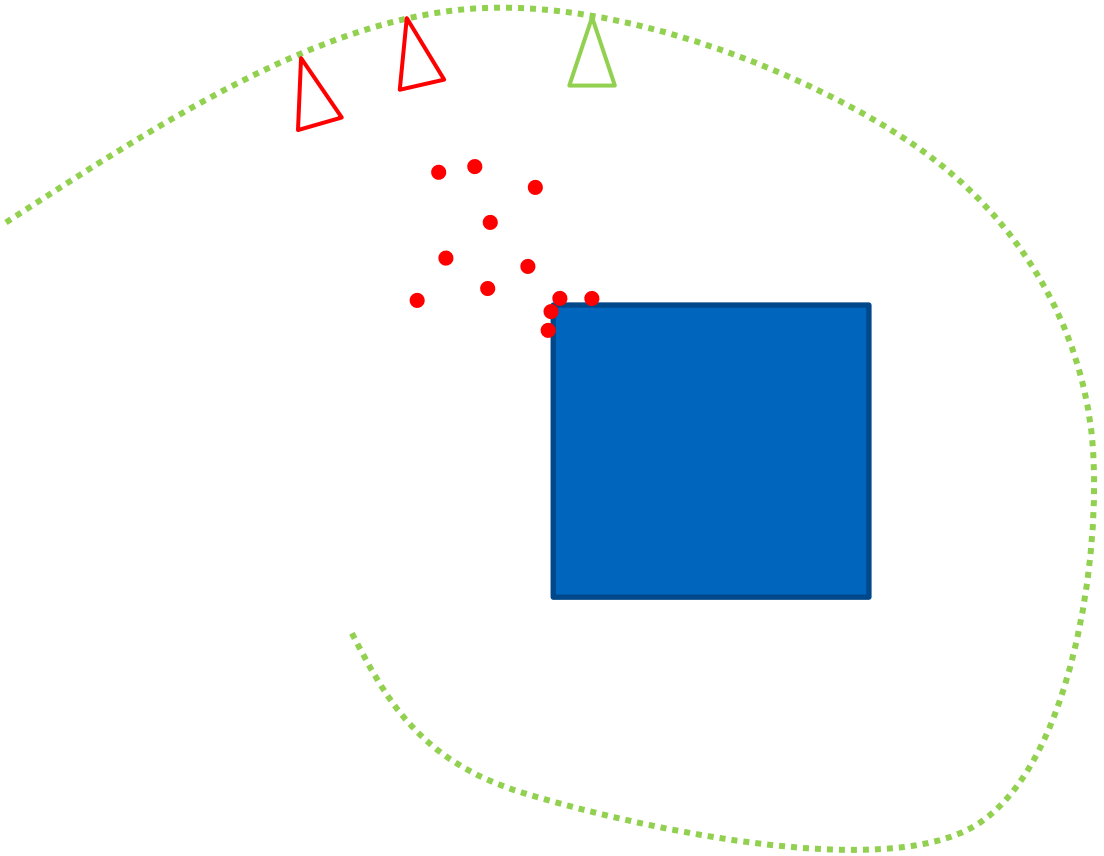
Map reinitialisation and tracking



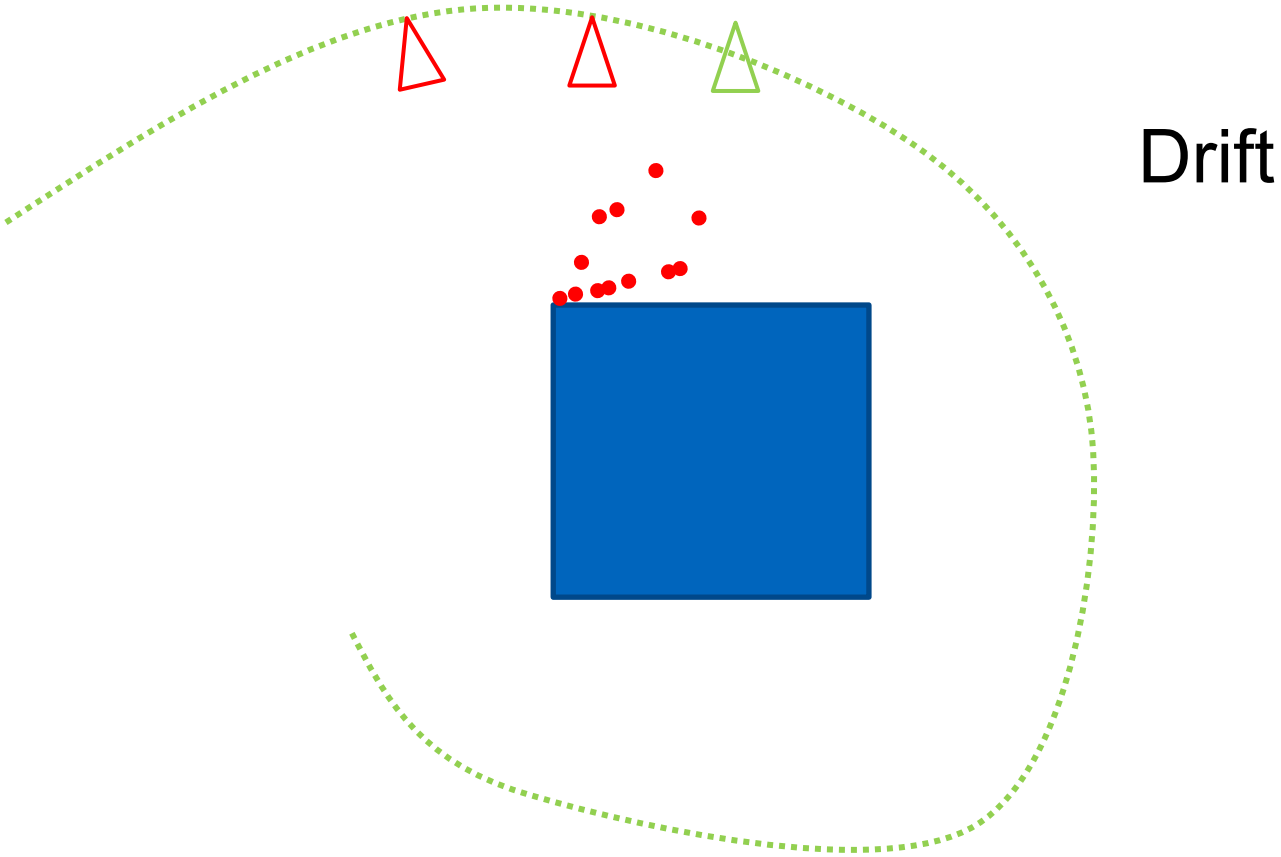
Map reinitialisation and tracking



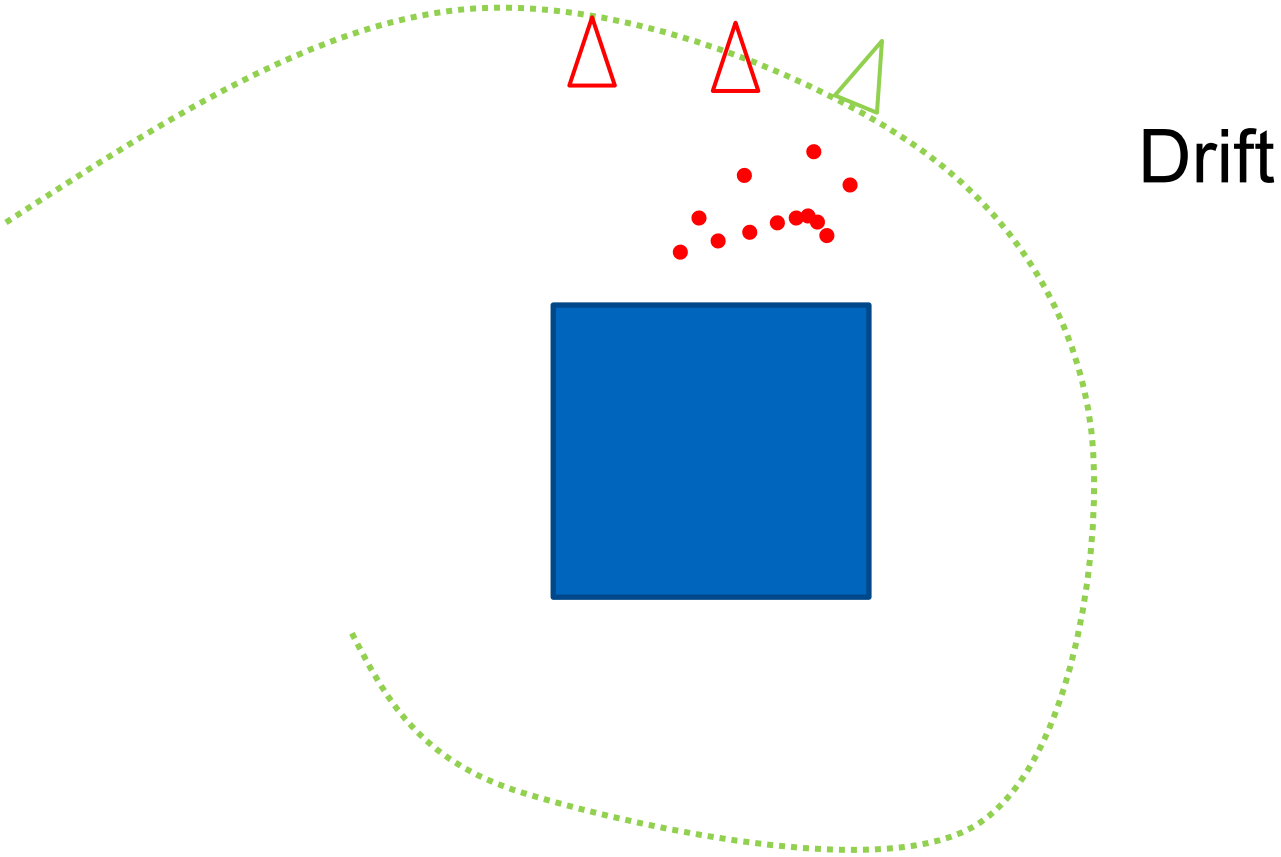
Map reinitialisation and tracking



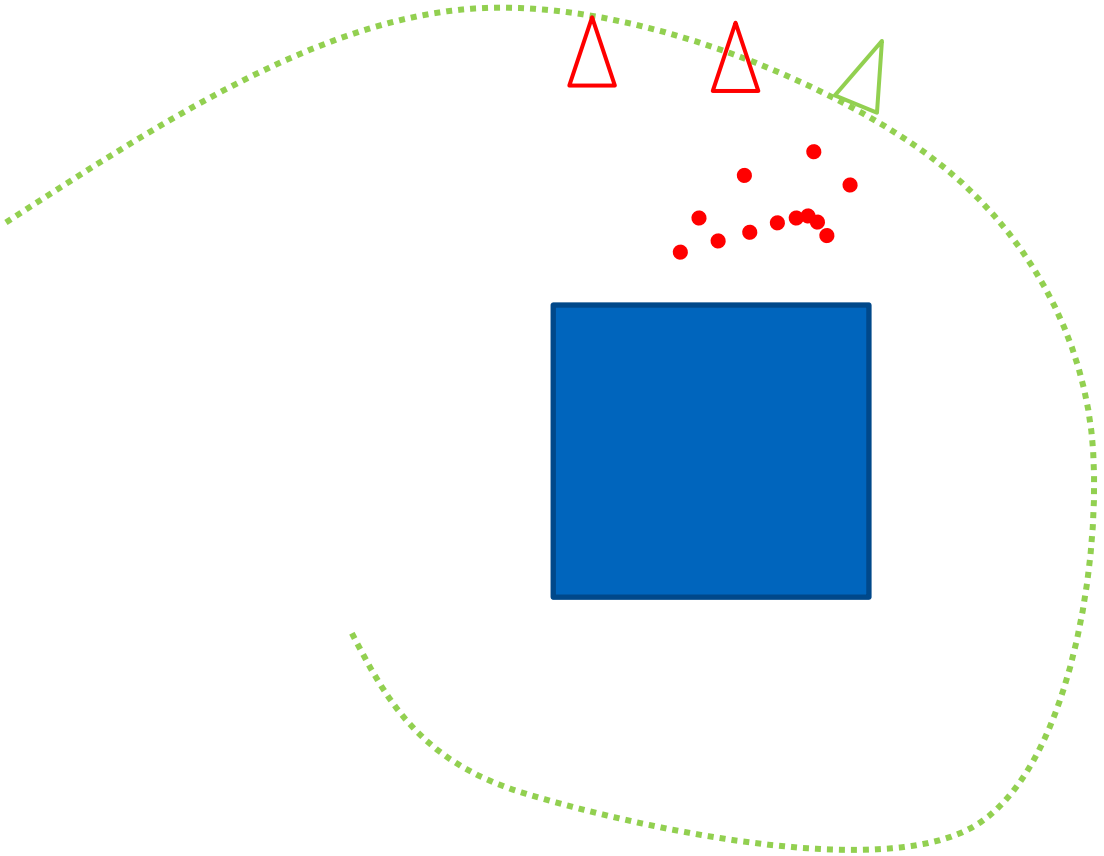
Map reinitialisation and tracking



Map reinitialisation and tracking



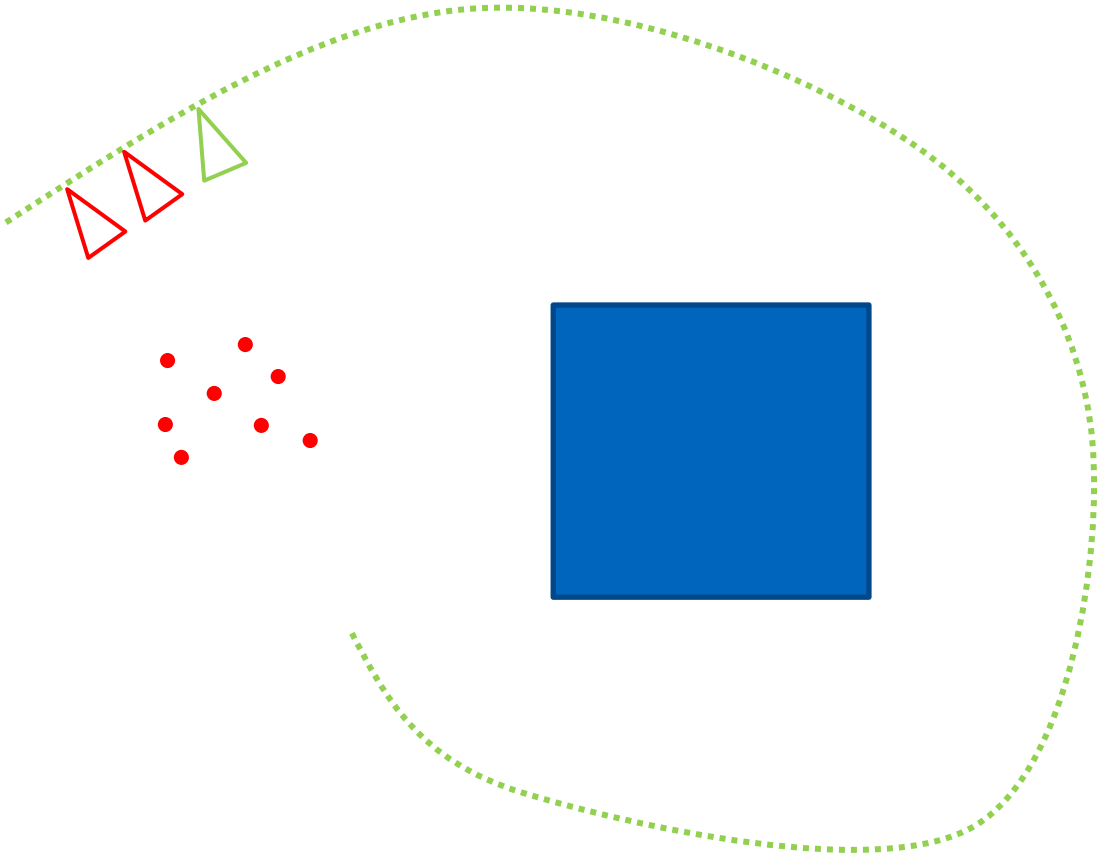
Map reinitialisation and tracking



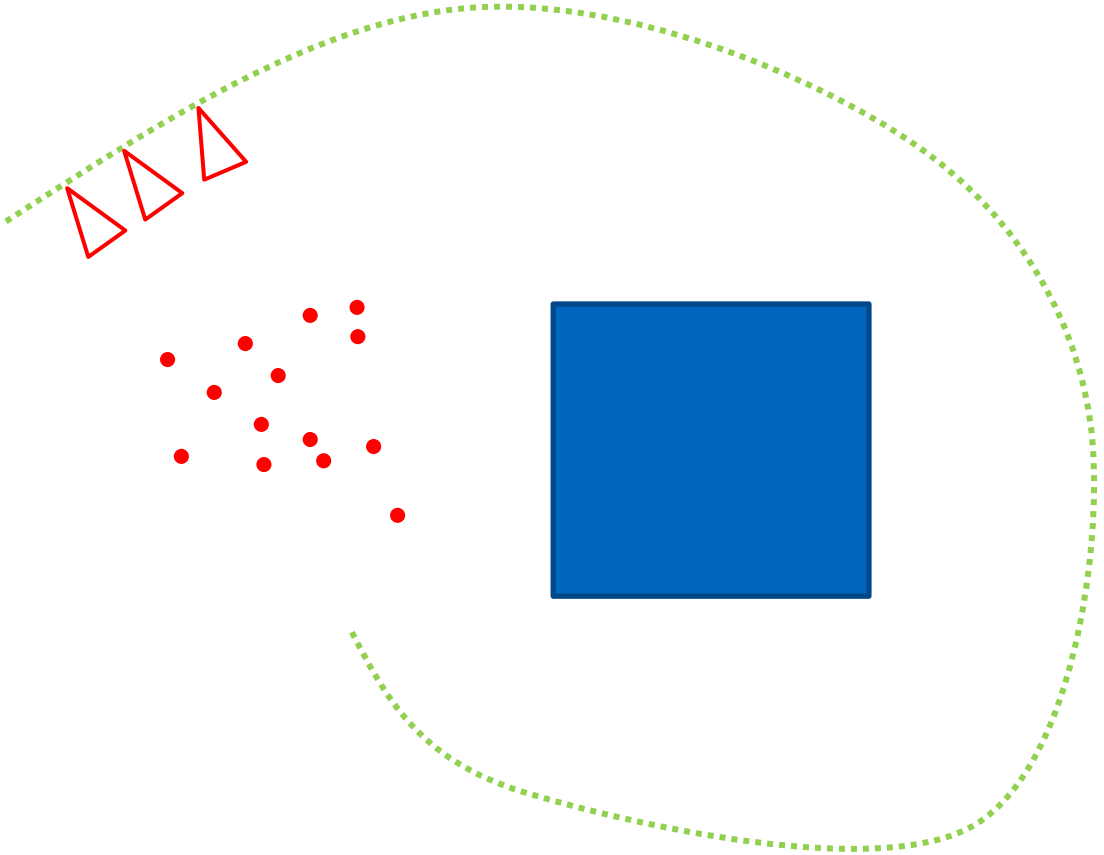
Drift

Very naïve
Visual Odometry (VO)

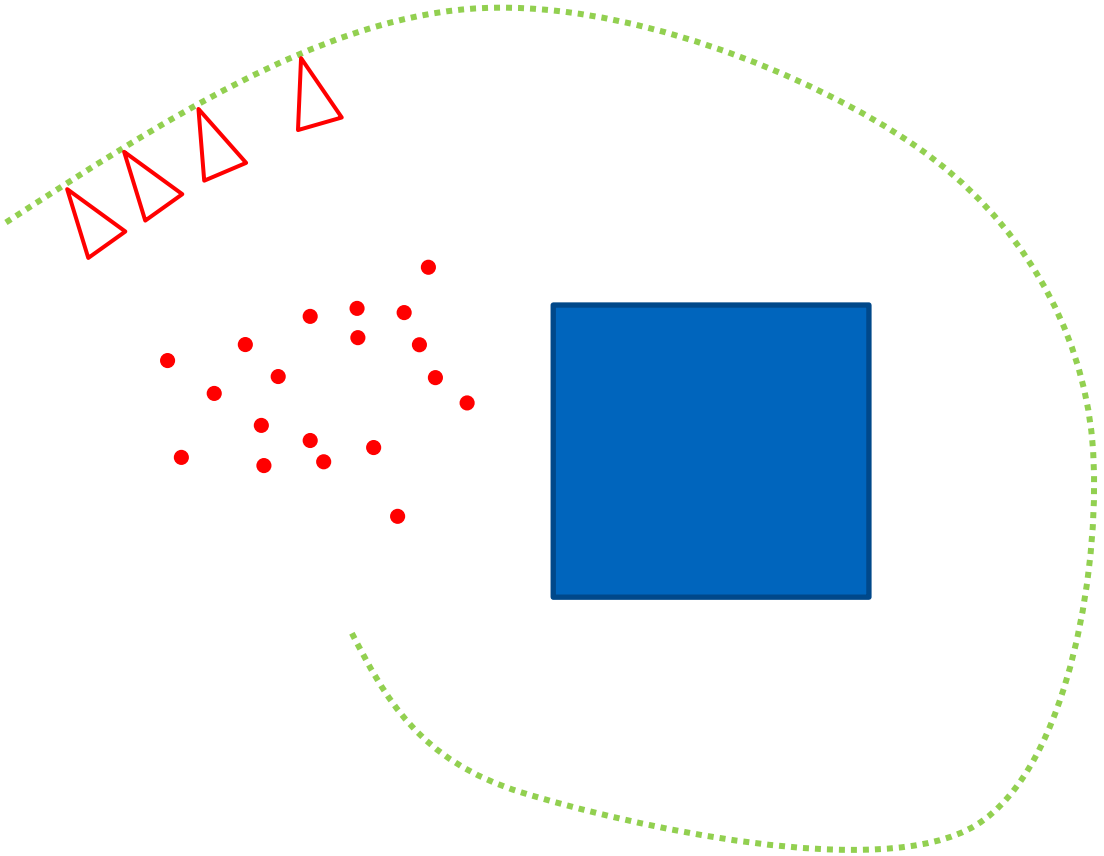
Multi-view mapping



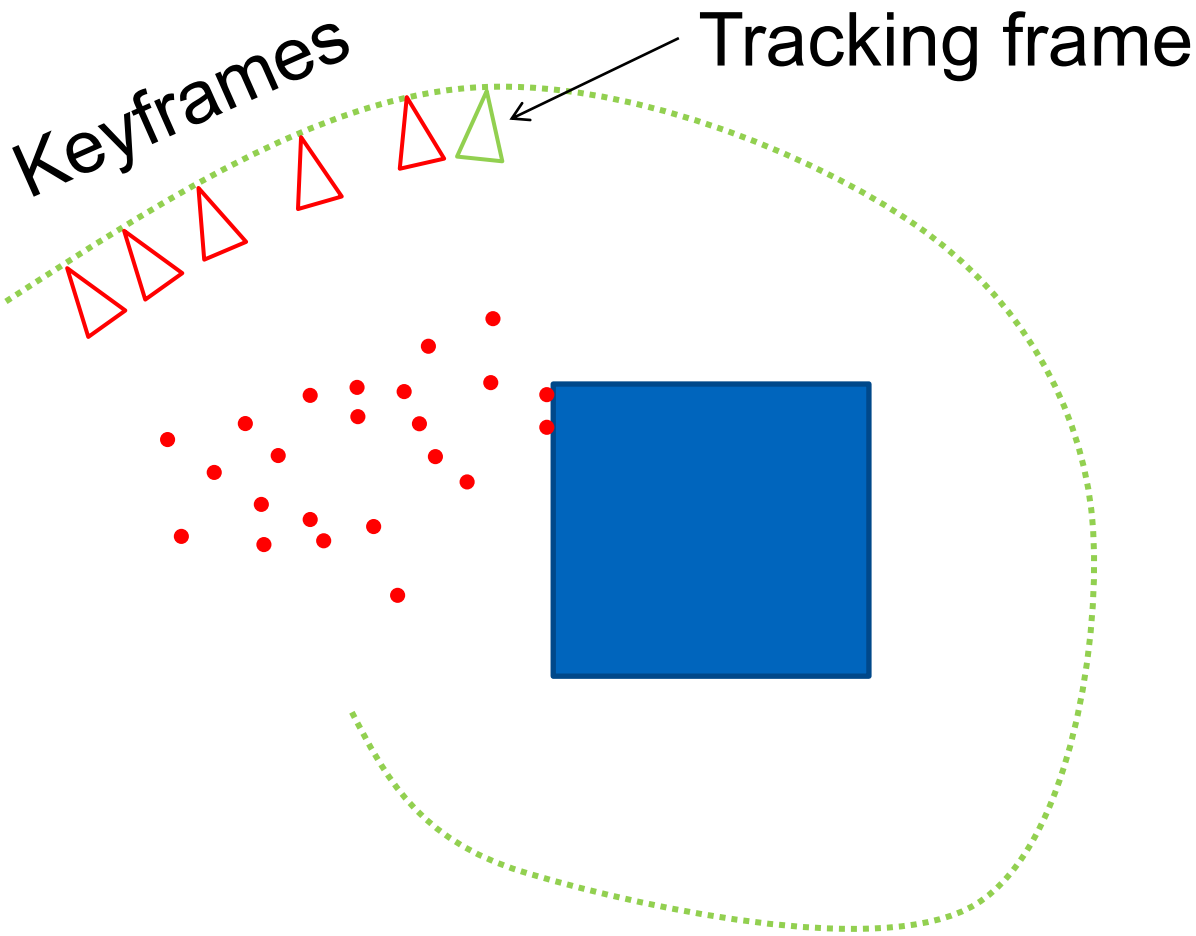
Multi-view mapping



Multi-view mapping



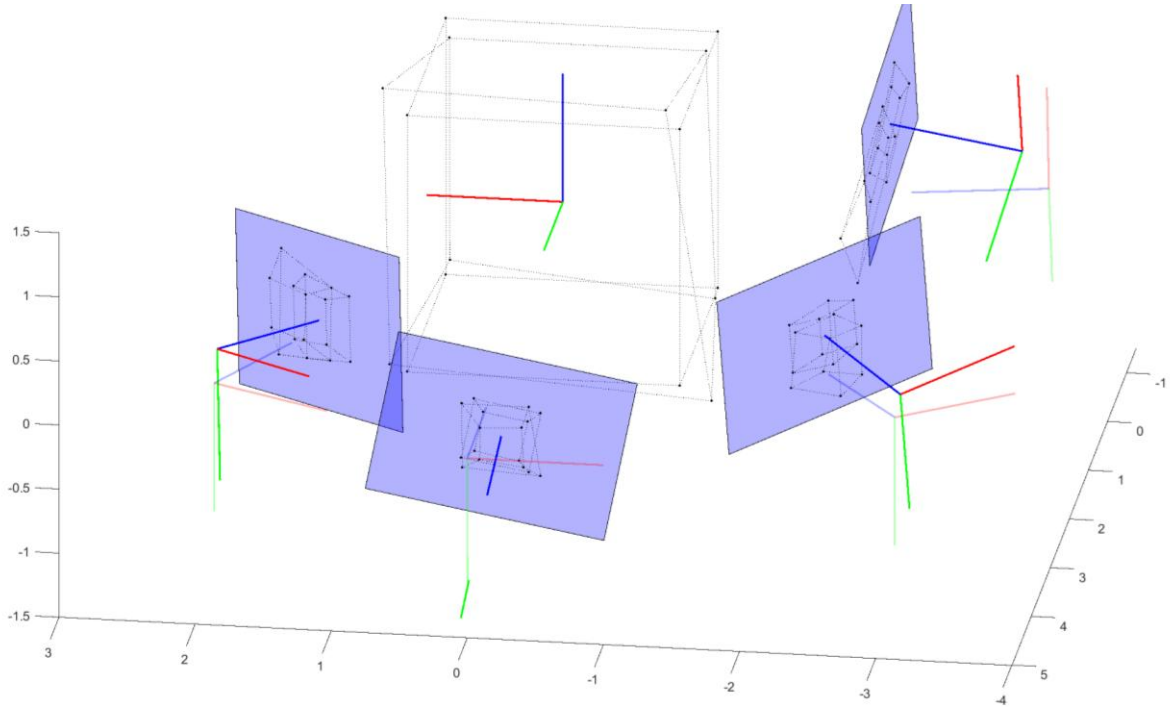
Multi-view mapping



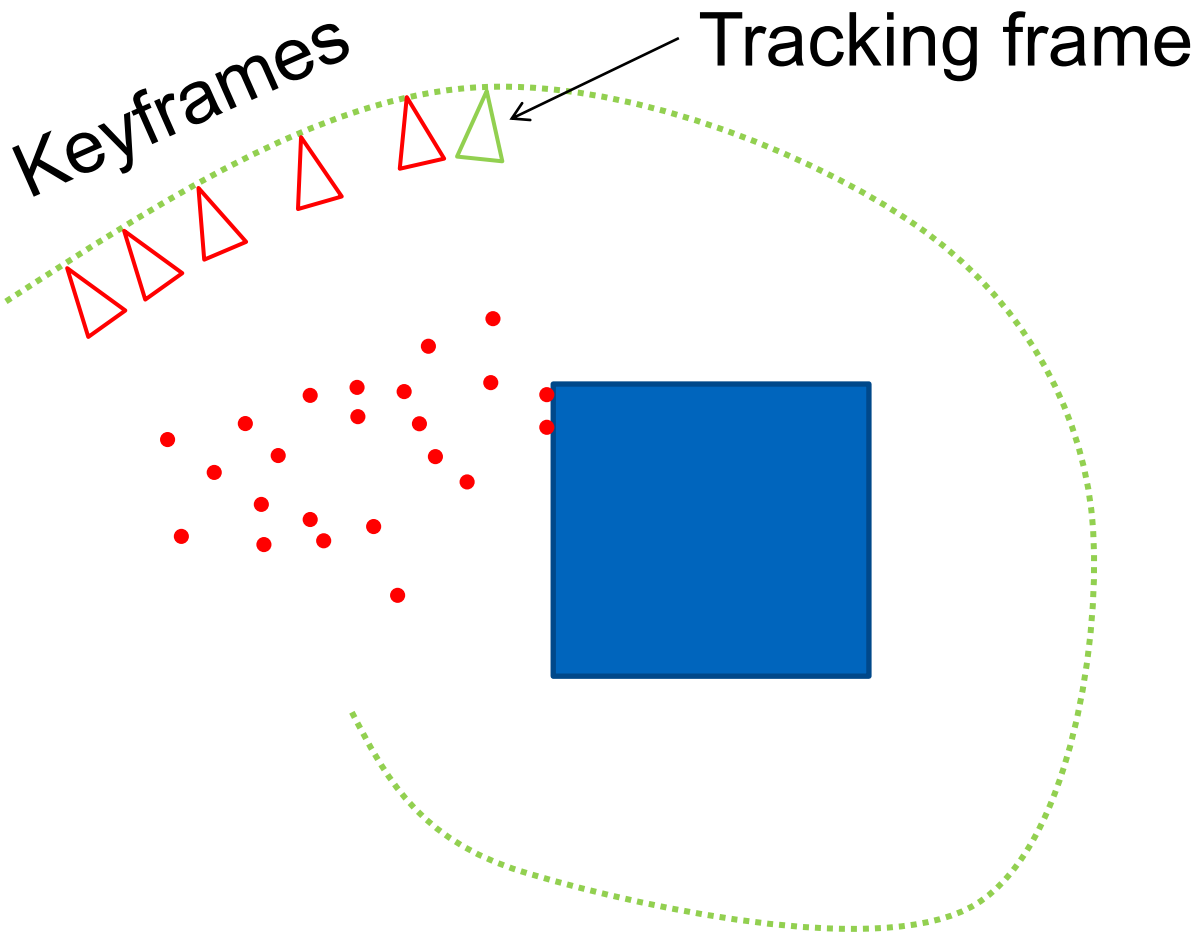
Full bundle adjustment

Minimise **geometric error** over the **camera poses** and **world points**

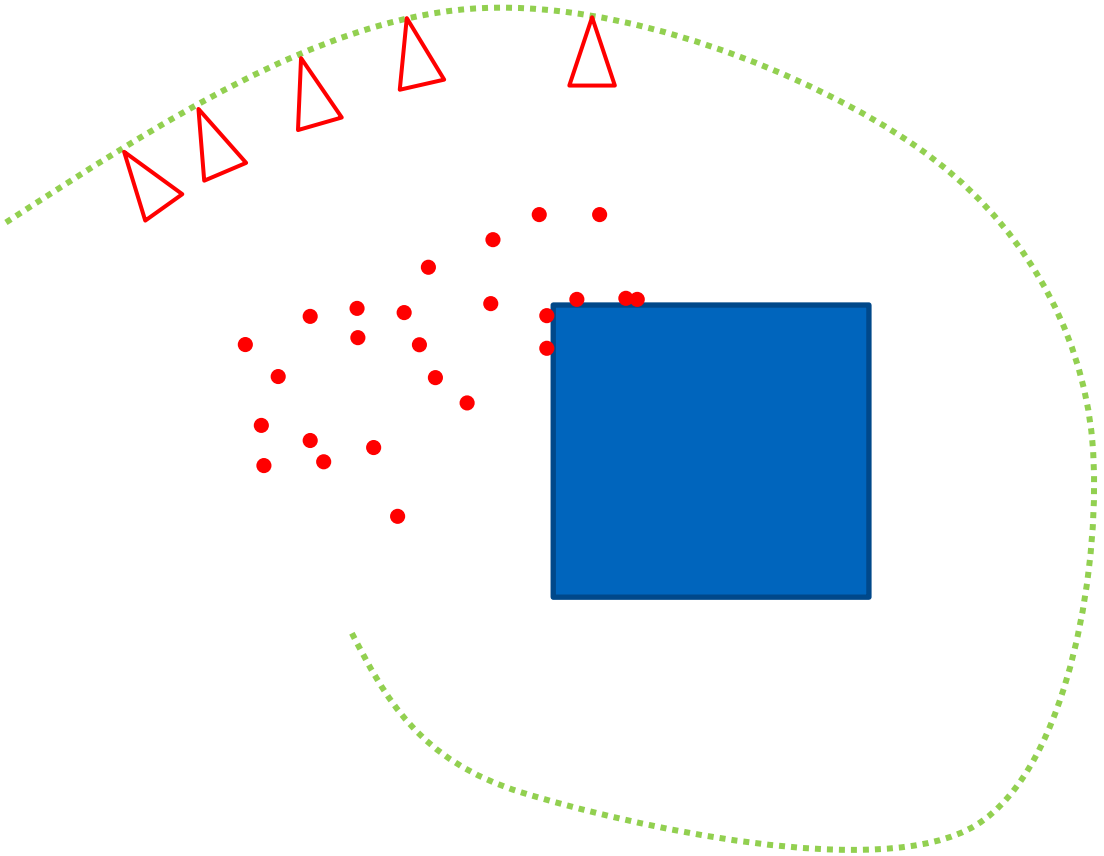
$$\left\{\mathbf{T}_{wc_i}^{*}, \mathbf{x}_j^{w*}\right\} = \underset{\mathbf{T}_{wc_i}, \mathbf{x}_j^w}{\operatorname{argmin}} \sum_i \sum_j \left\| \pi_i(\mathbf{T}_{wc}^{-1} \cdot \mathbf{x}_j^w) - \mathbf{u}_j^i \right\|^2$$



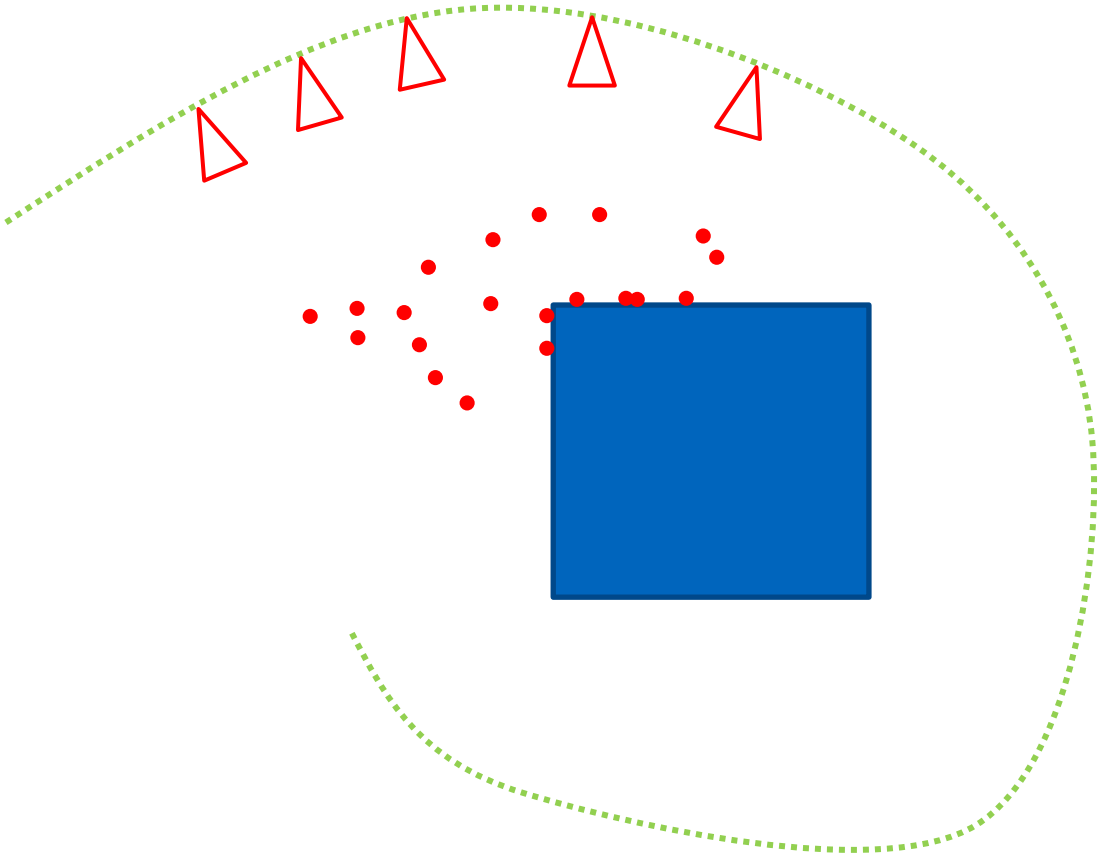
Multi-view mapping



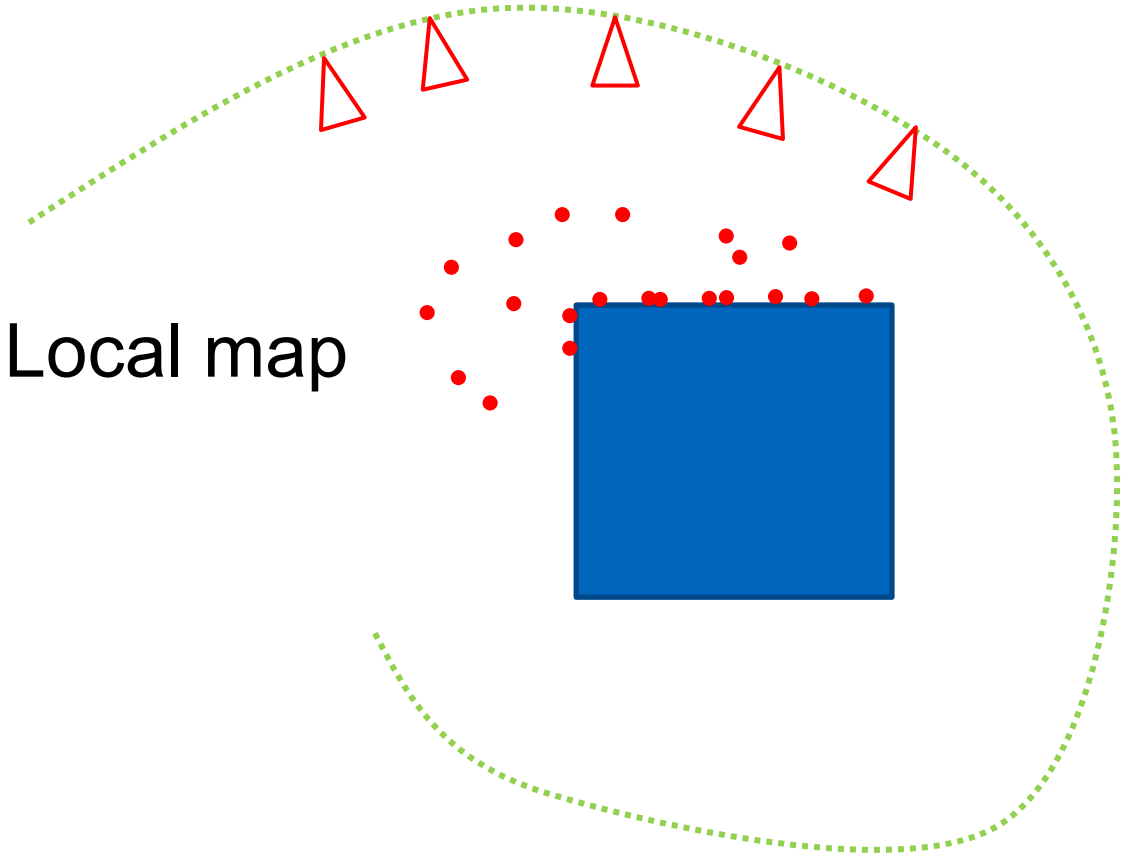
Sliding window mapping



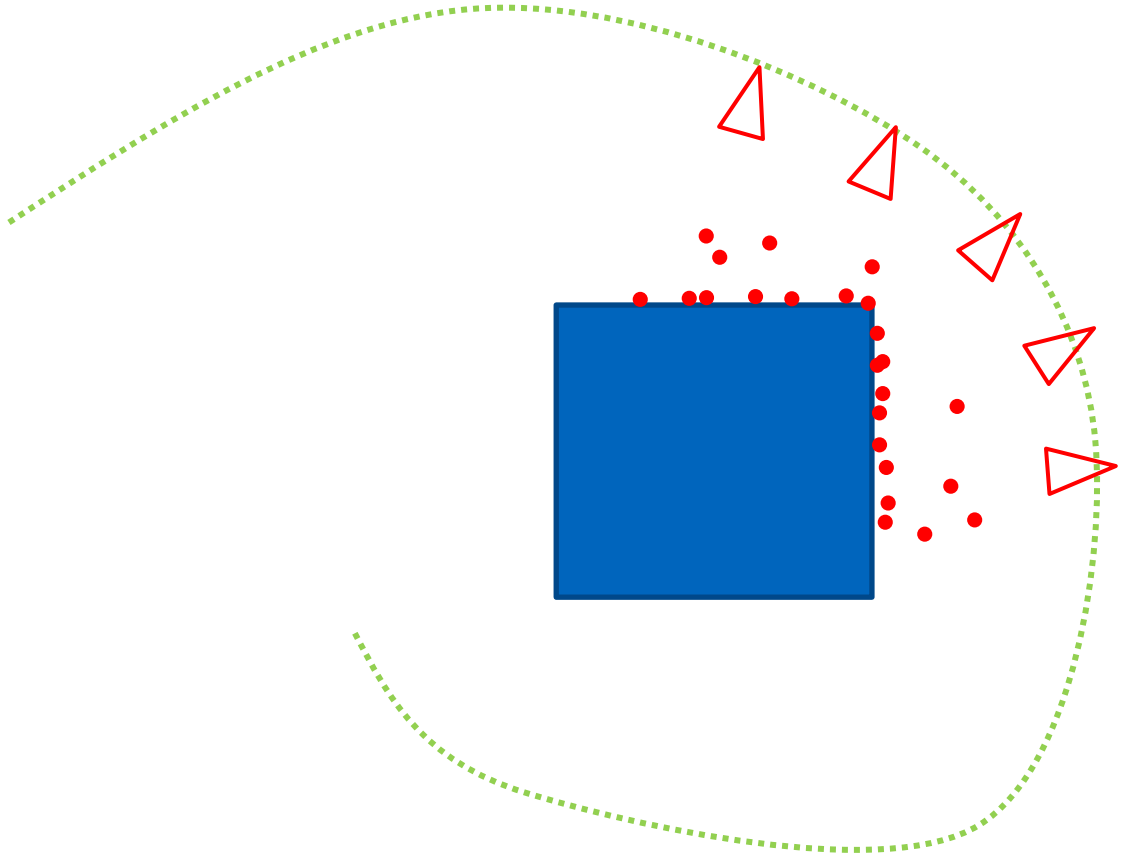
Sliding window mapping



Sliding window mapping

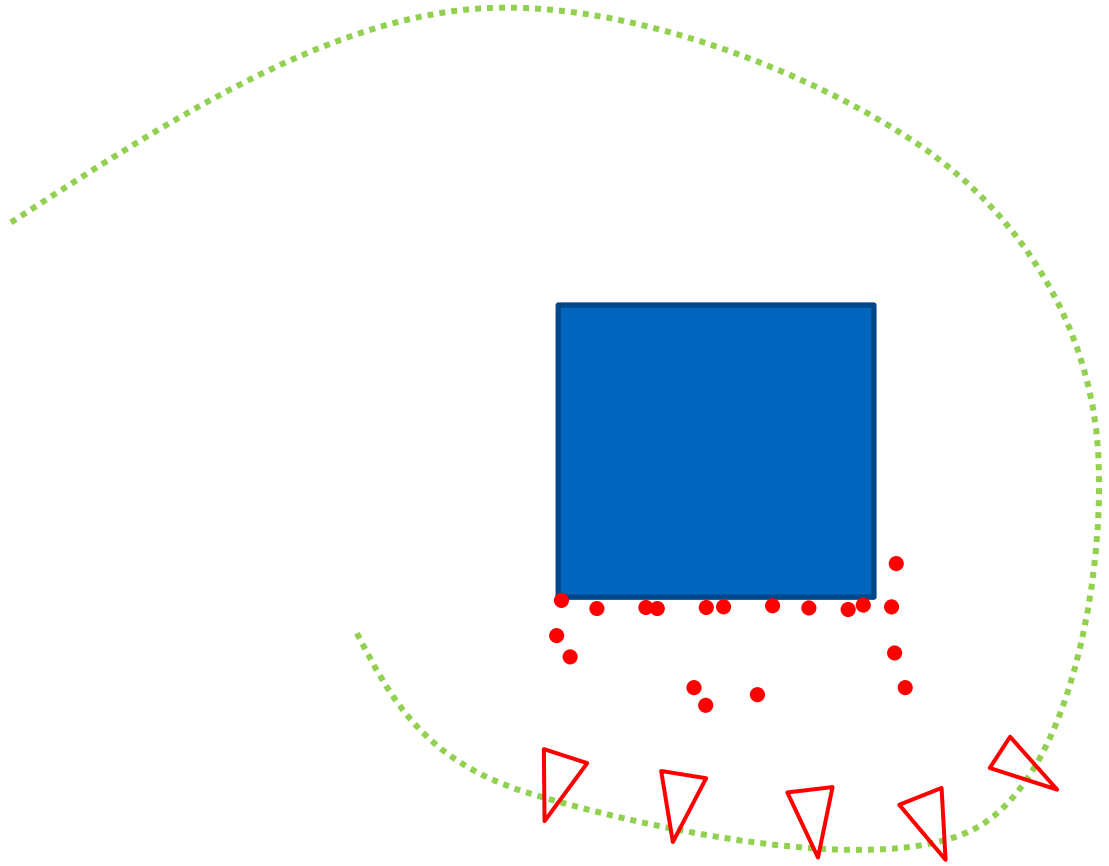


Sliding window mapping



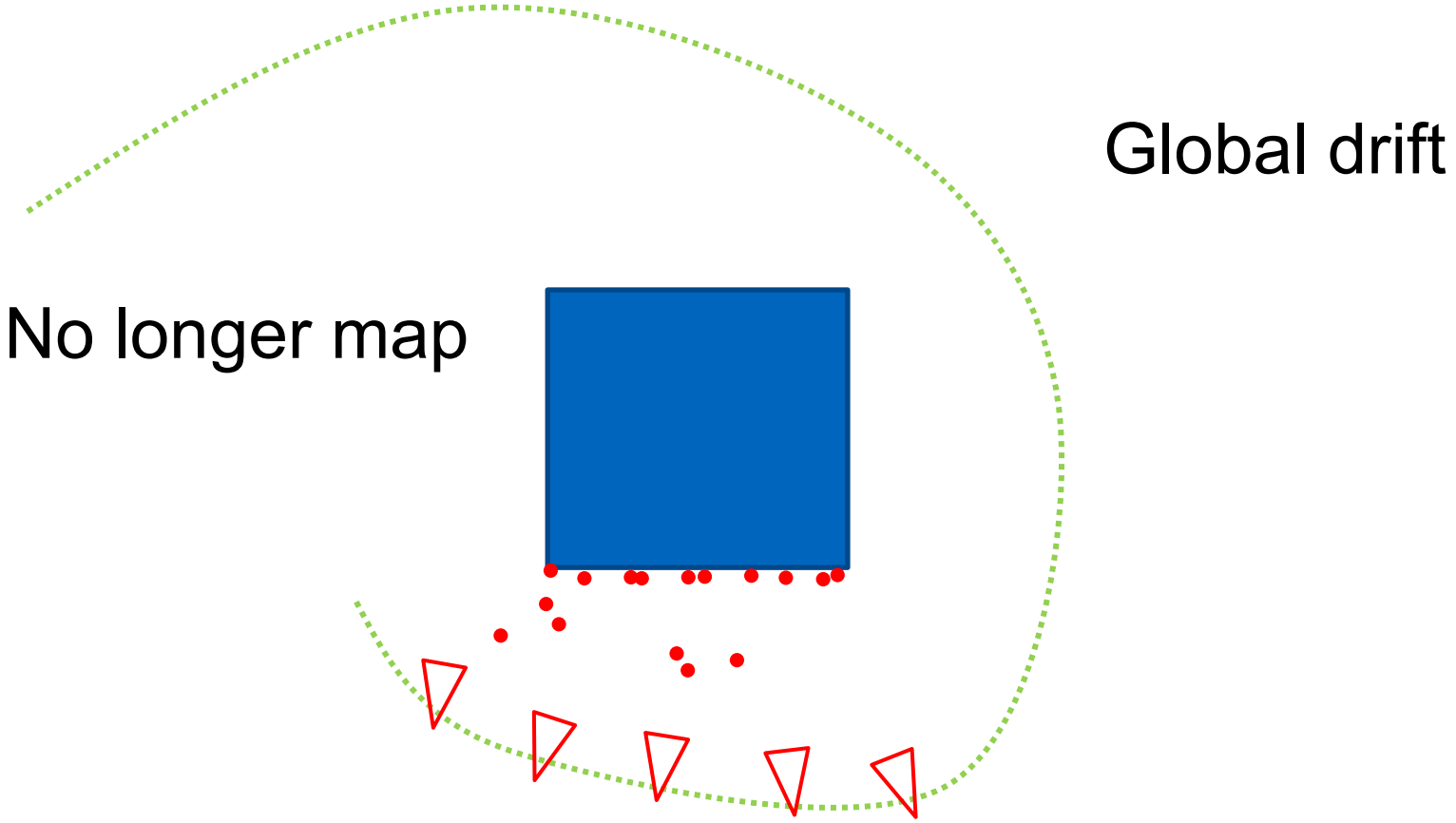
Global drift

Sliding window mapping

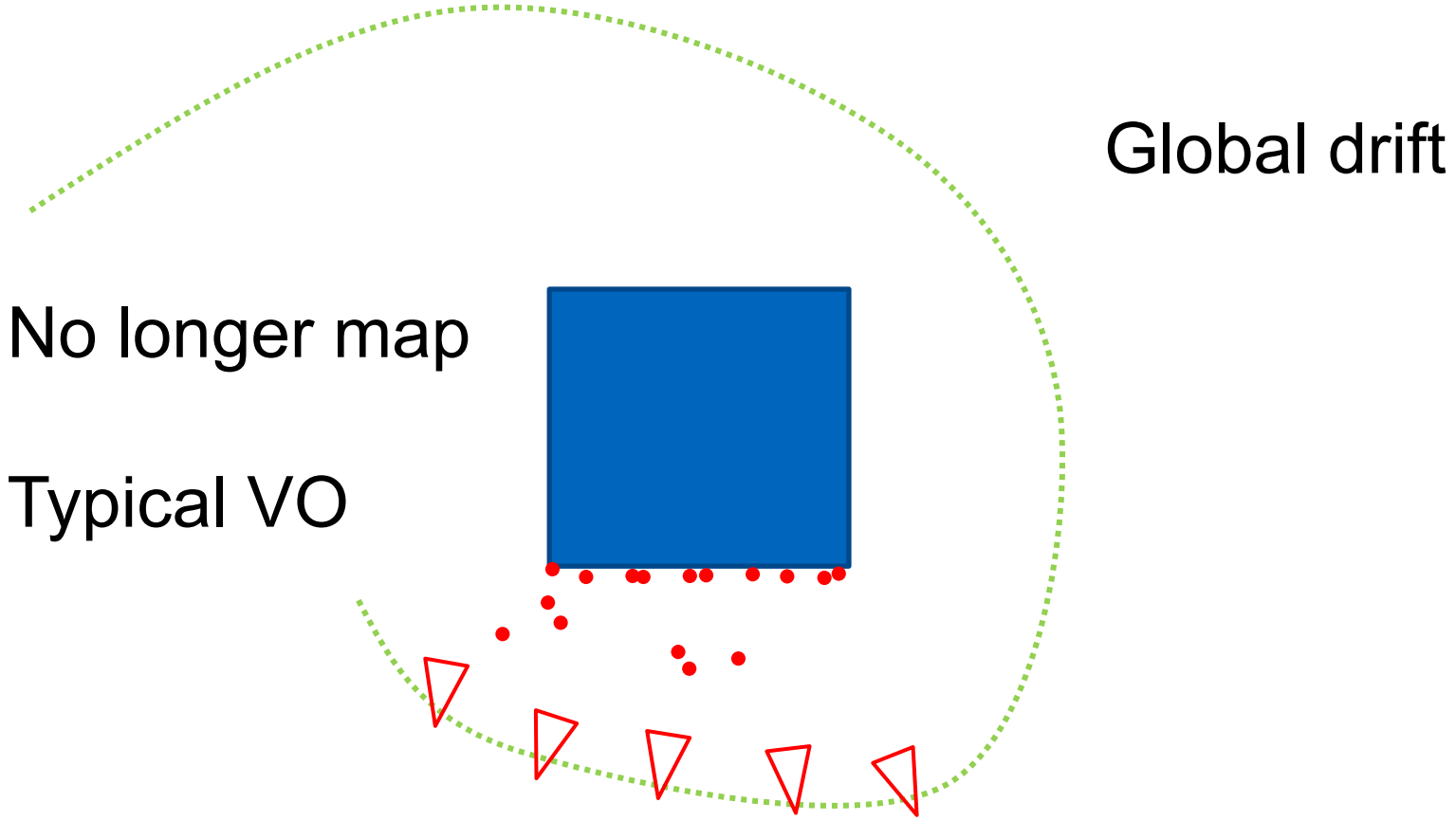


Global drift

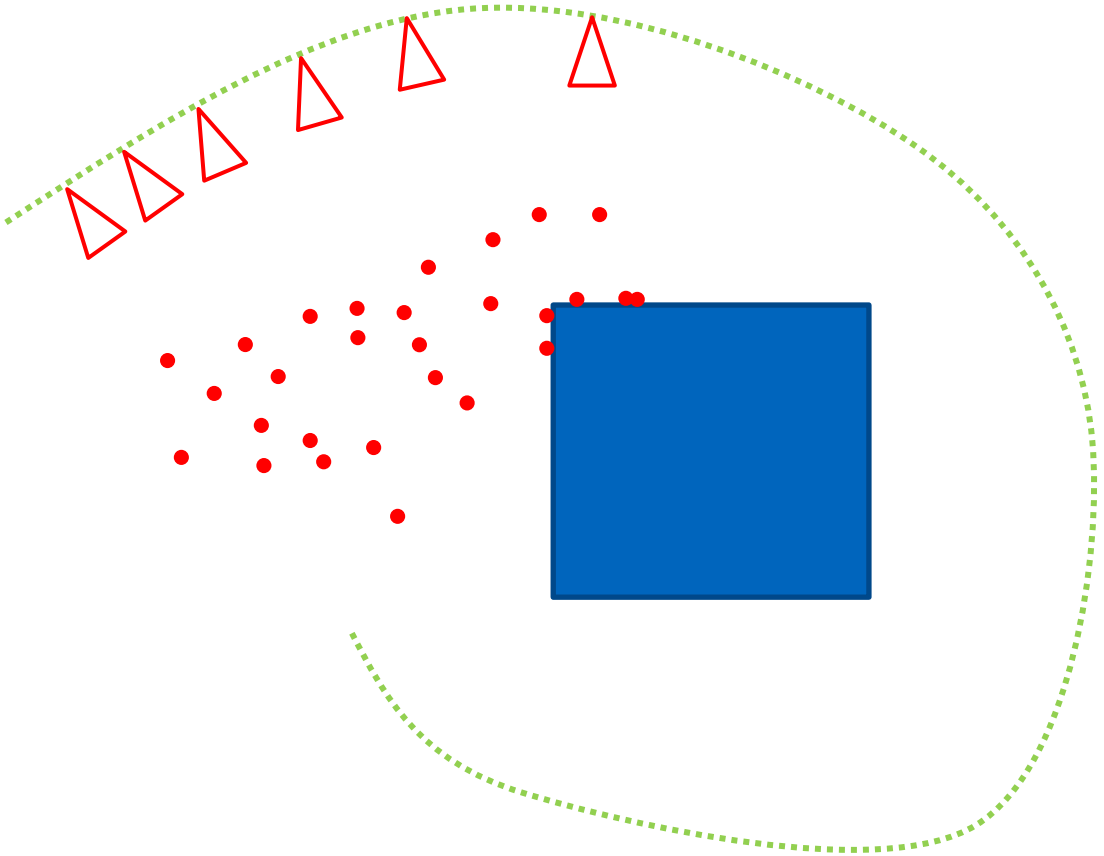
Sliding window mapping



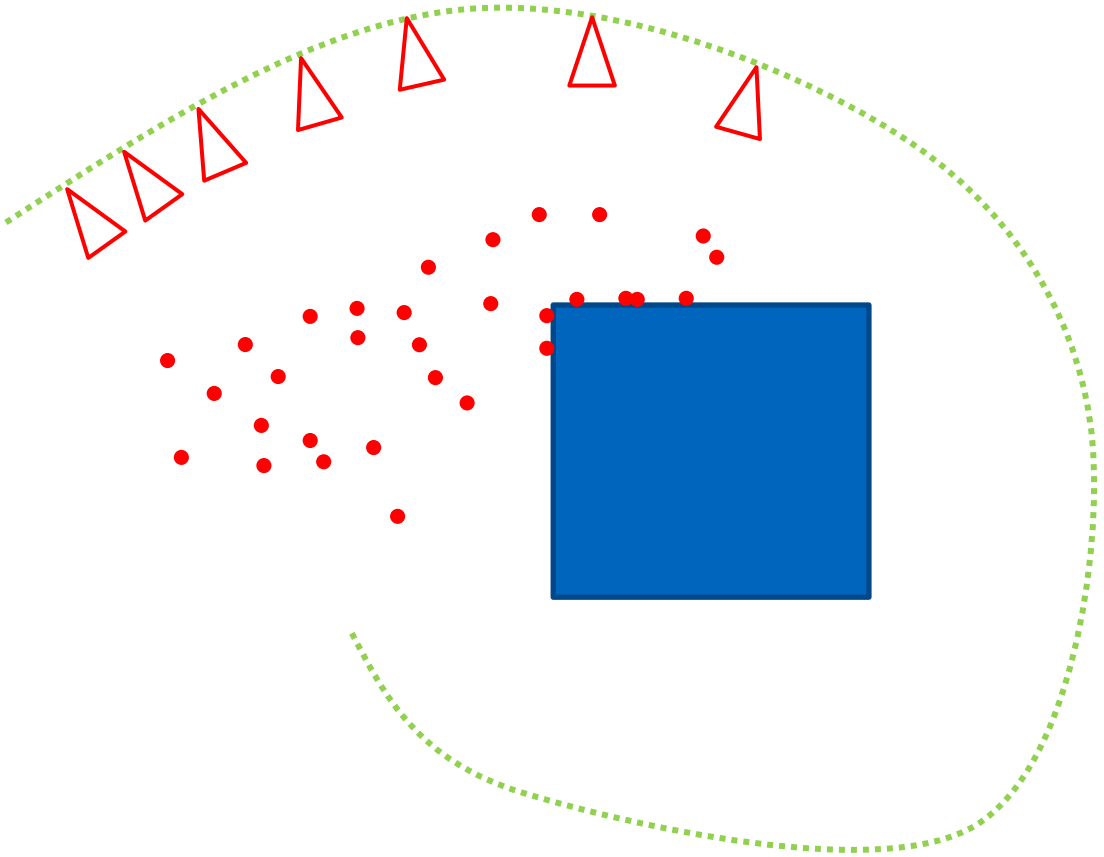
Sliding window mapping



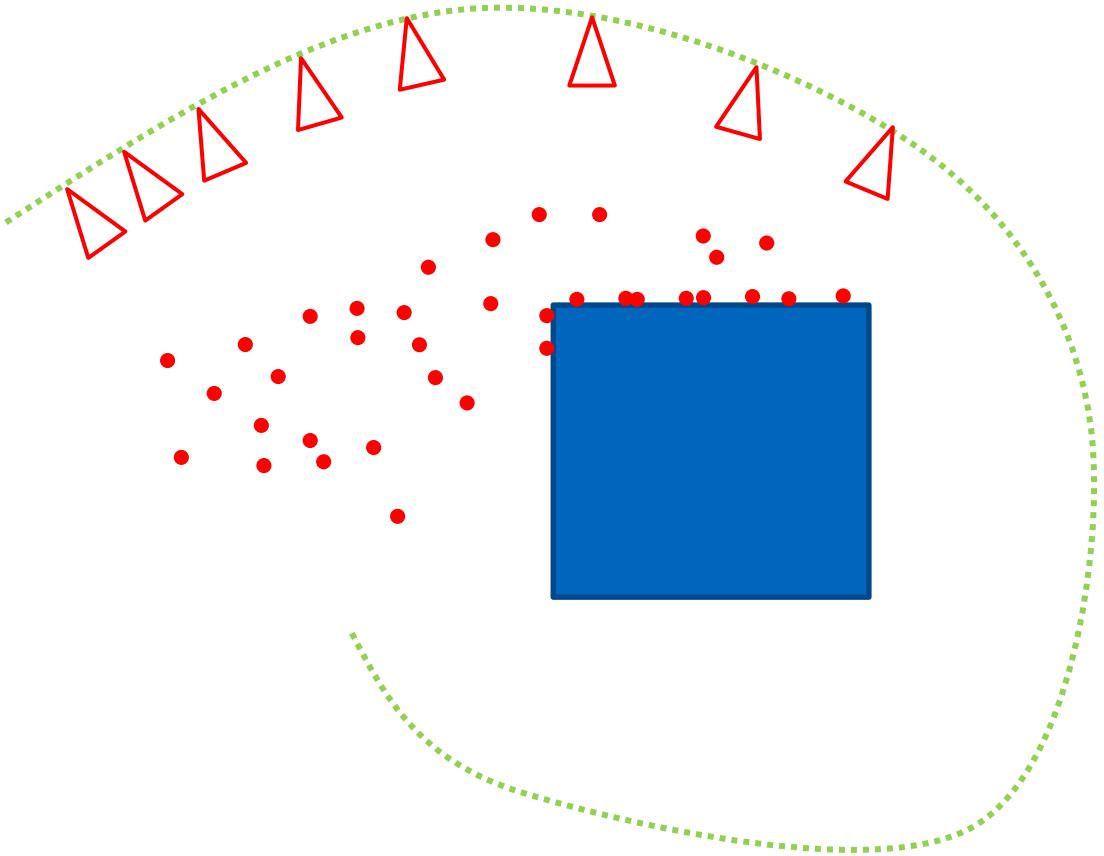
Monocular Visual SLAM



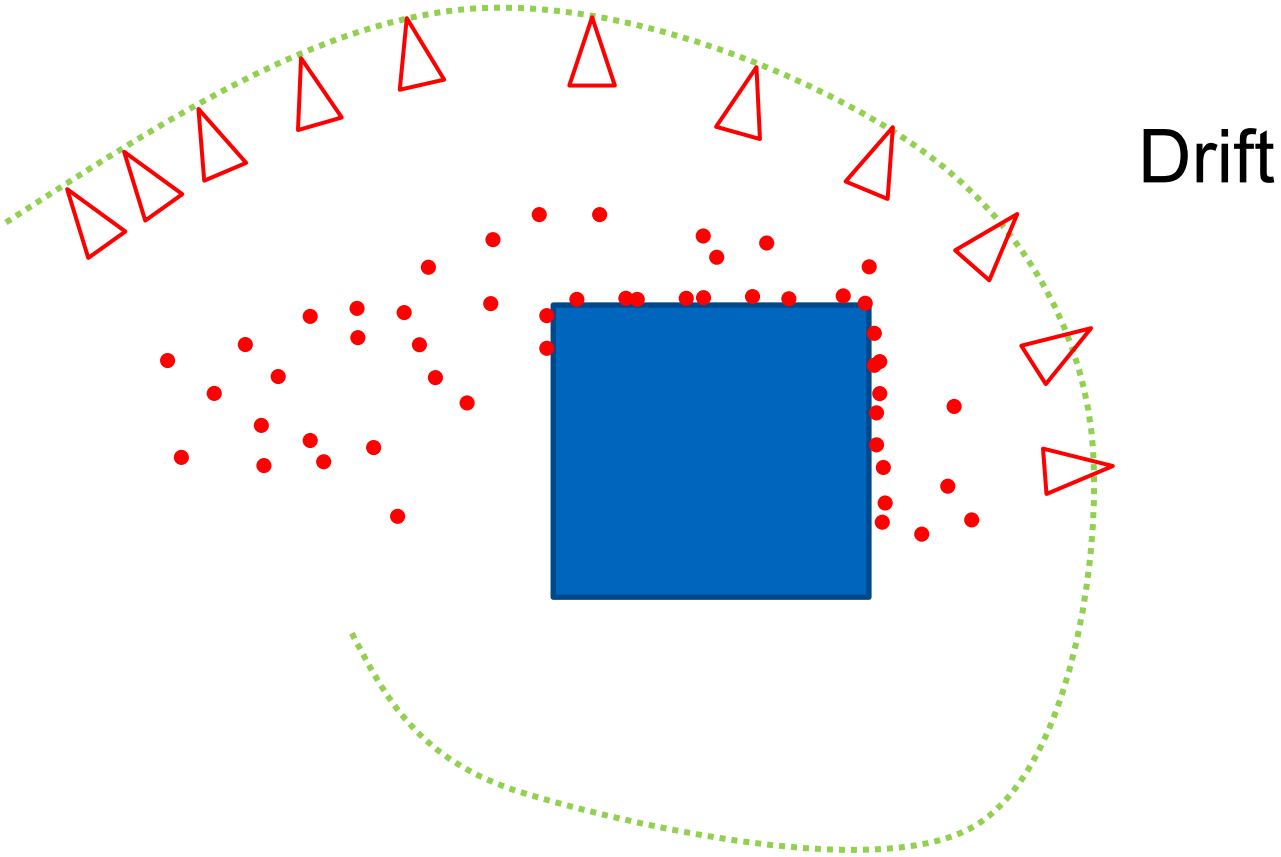
Monocular Visual SLAM



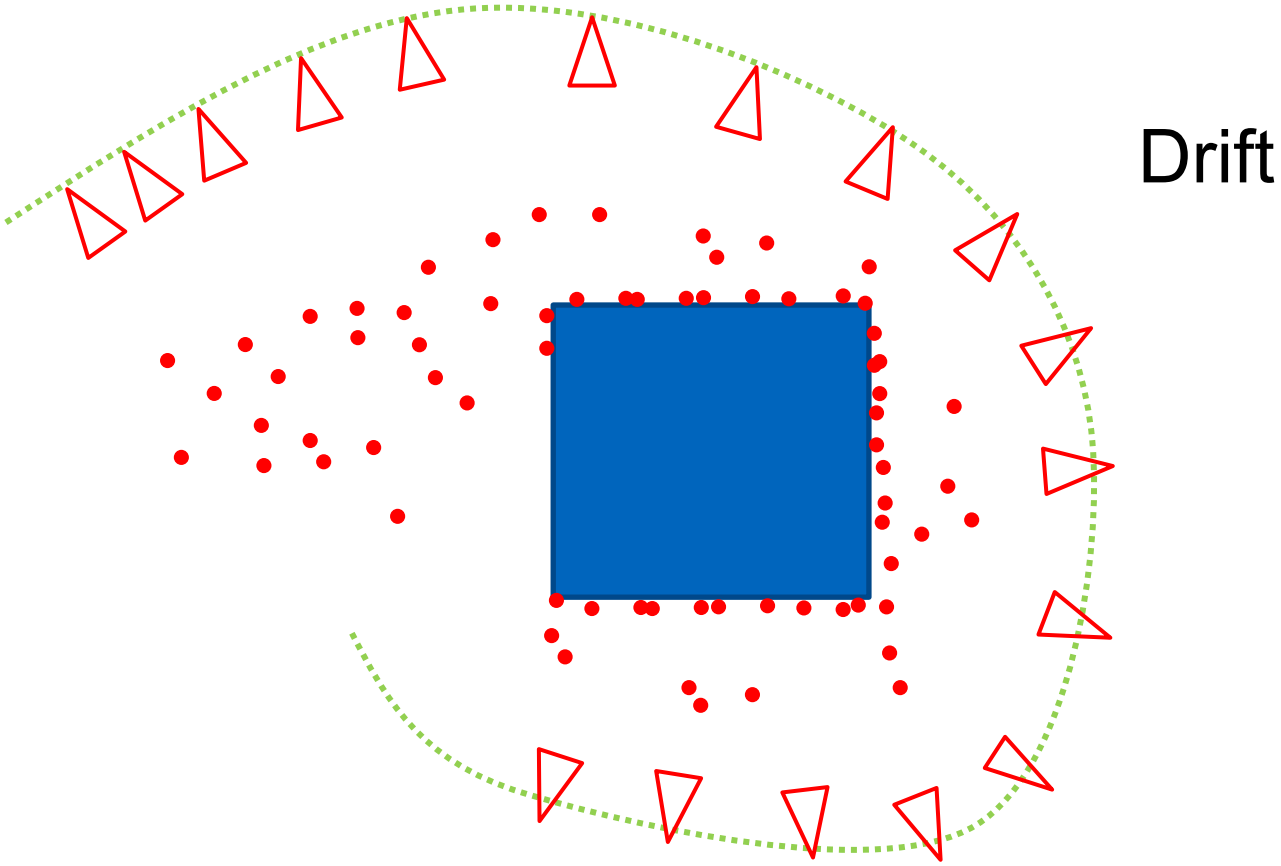
Monocular Visual SLAM



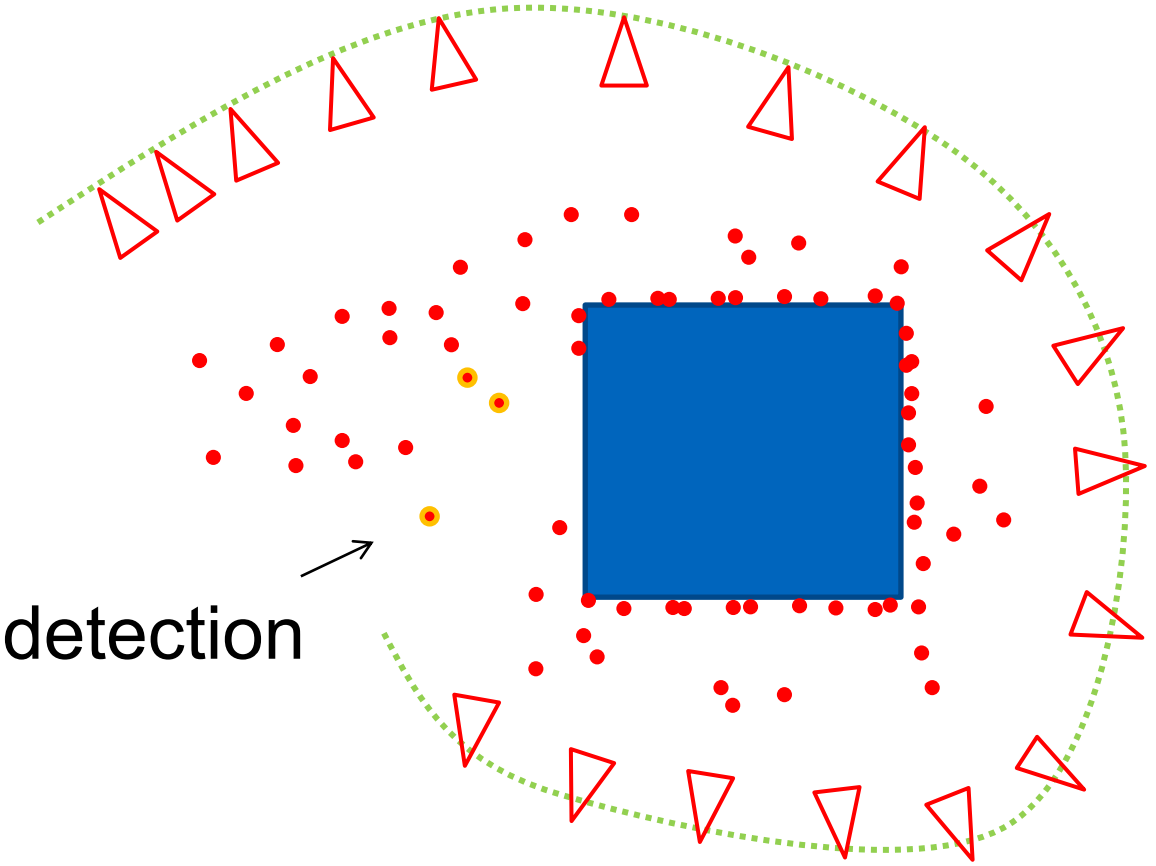
Monocular Visual SLAM



Monocular Visual SLAM



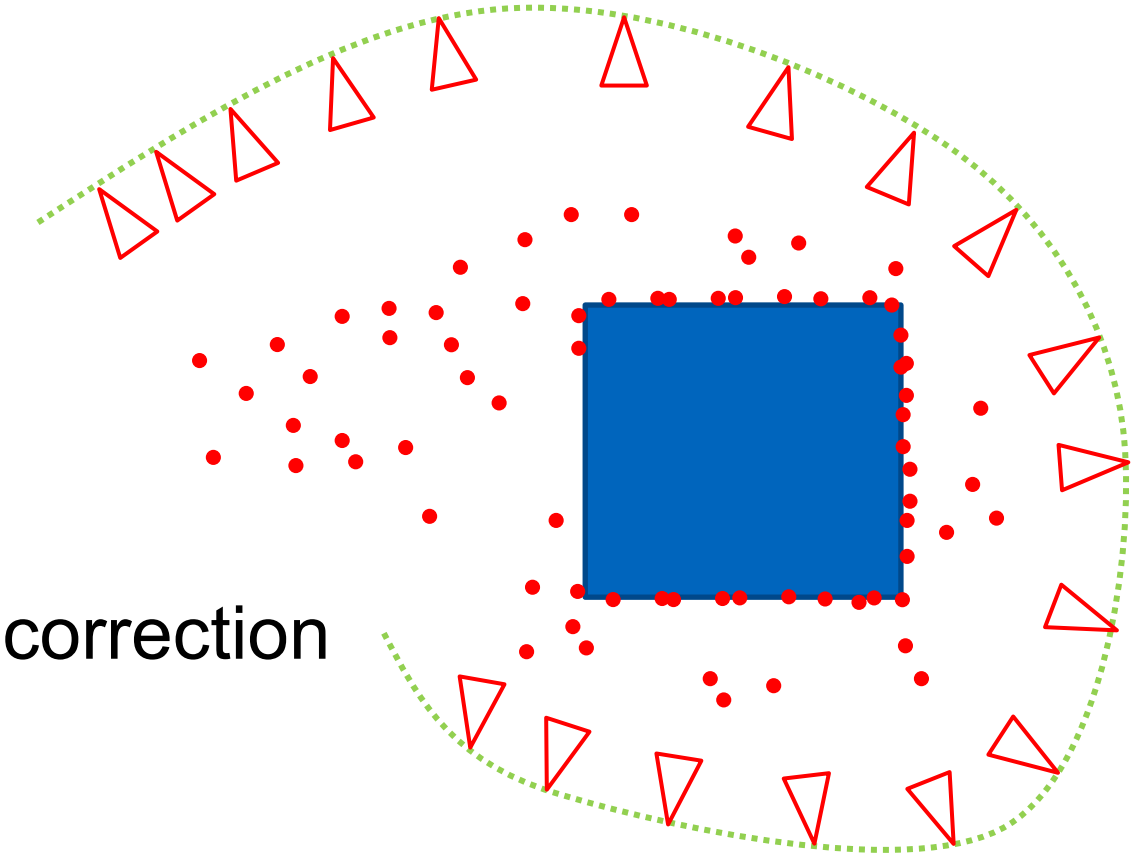
Monocular Visual SLAM



Drift

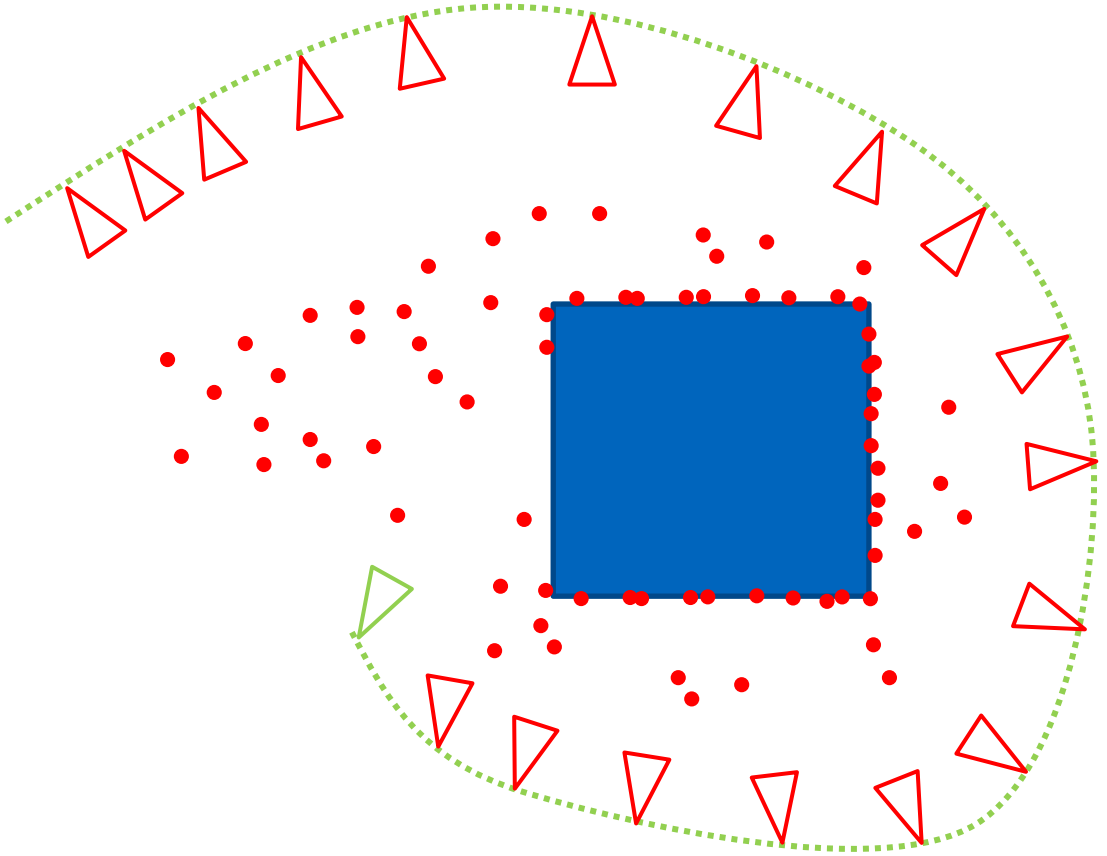
Loop closure detection

Monocular Visual SLAM

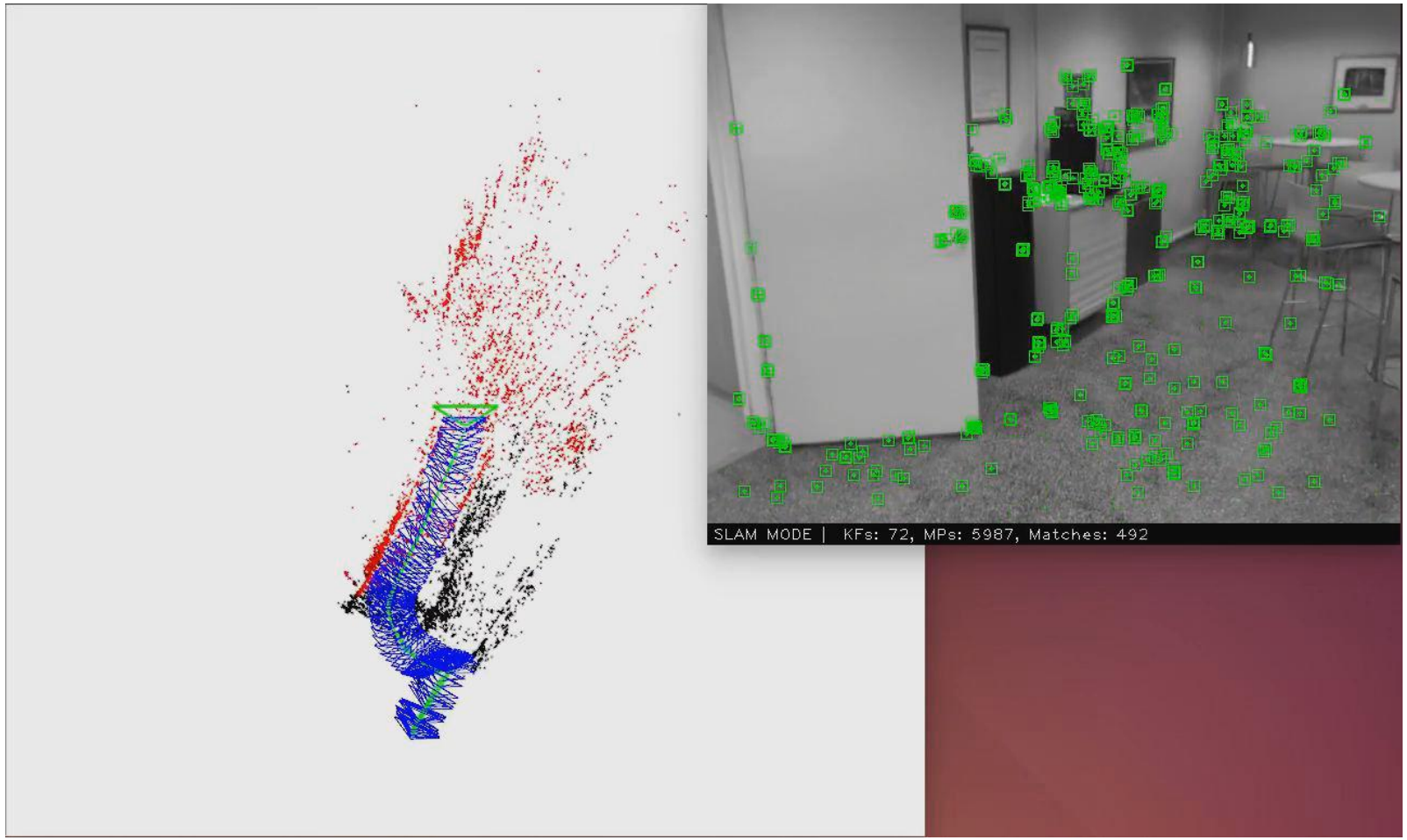


Loop closure correction

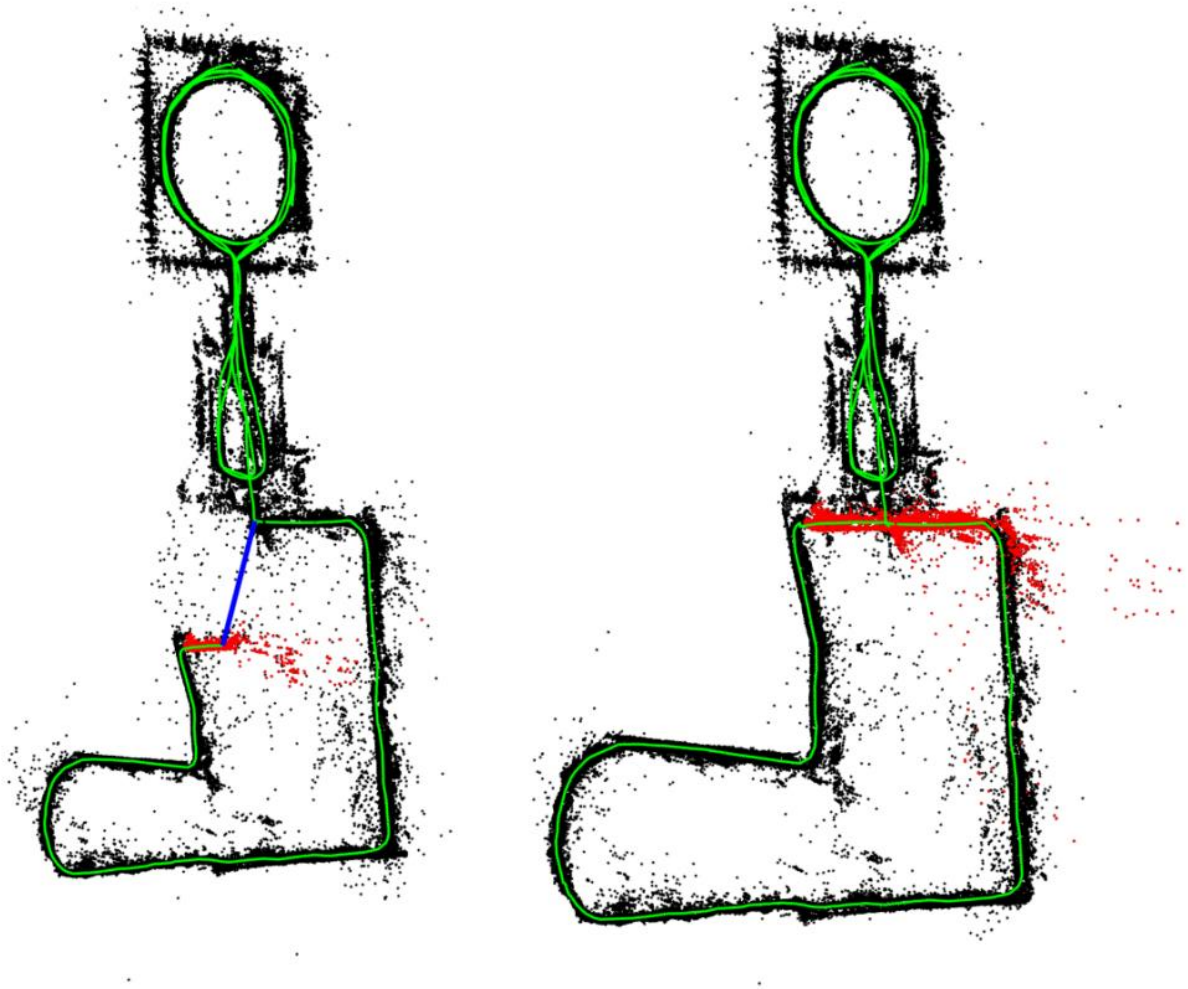
Monocular Visual SLAM



Visual SLAM example

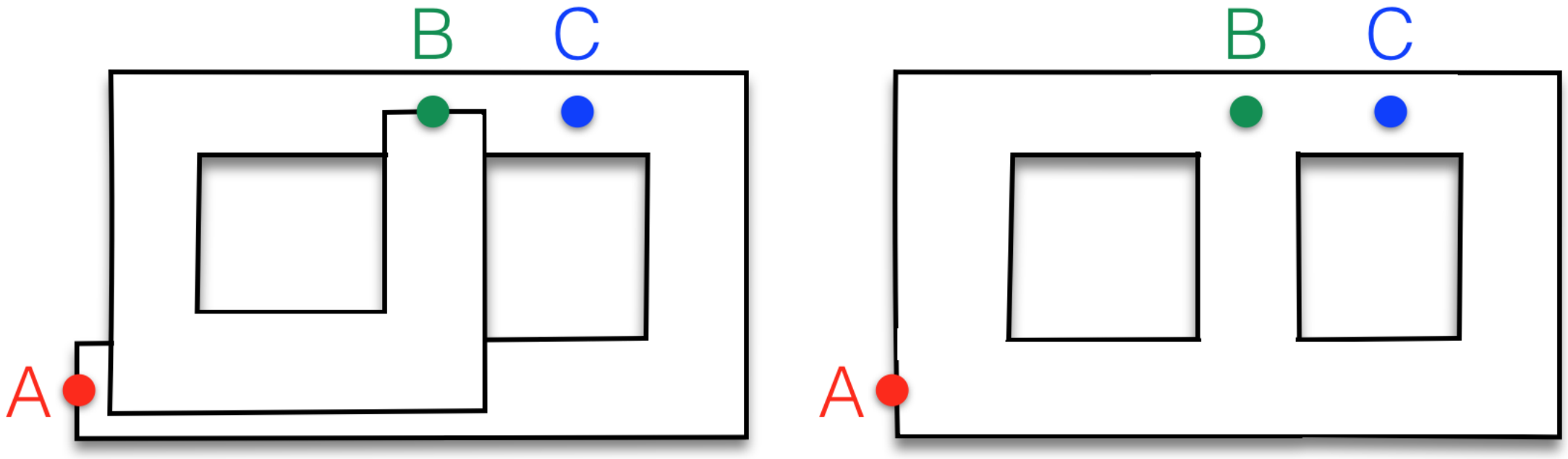


Visual SLAM vs visual odometry



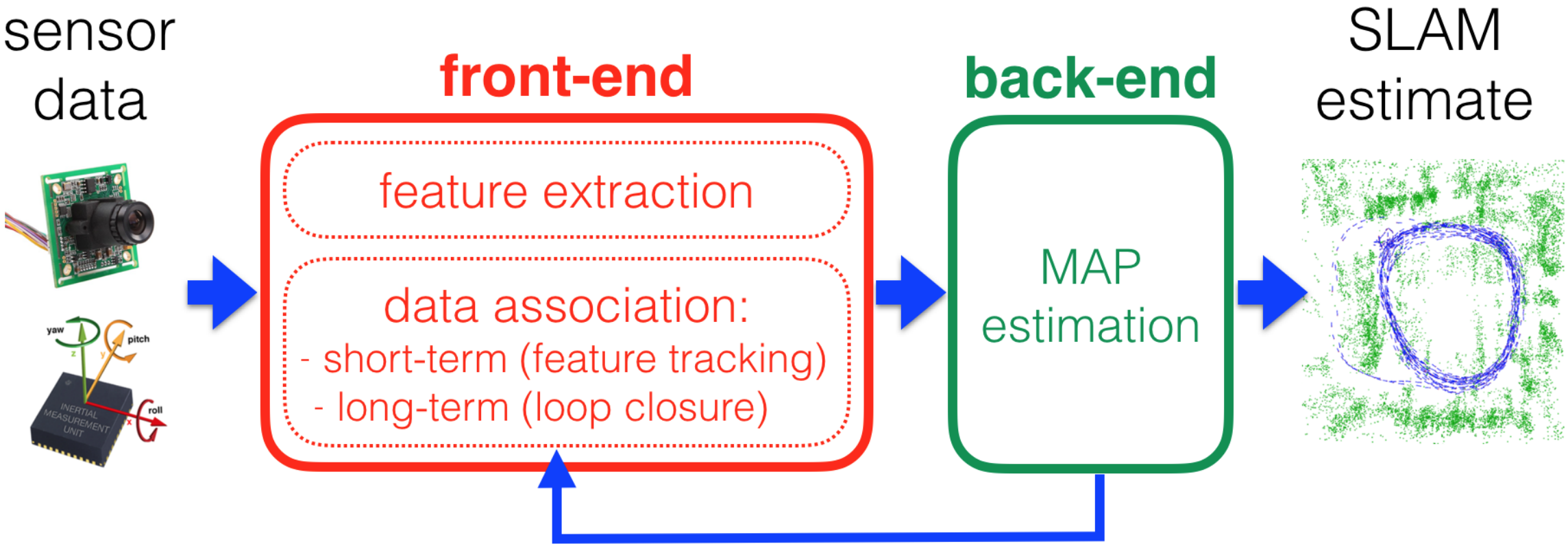
Mur-Artal, R., Montiel, J. M. M., & Tardos, J. D. (2015). ORB-SLAM: A Versatile and Accurate Monocular SLAM System. *IEEE Transactions on Robotics*, *31*(5), 1147–1163

Visual SLAM vs visual odometry



Cadena, C., et al. (2016). Past, Present, and Future of Simultaneous Localization and Mapping: Toward the Robust-Perception Age. *IEEE Transactions on Robotics*, *32*(6), 1309–1332

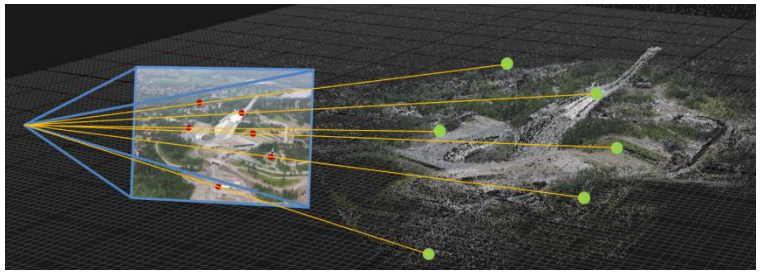
Components of SLAM

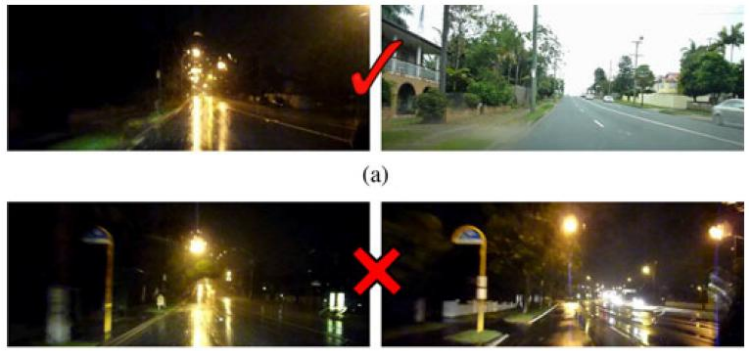


Cadena, C., et al. (2016). Past, Present, and Future of Simultaneous Localization and Mapping: Toward the Robust-Perception Age. *IEEE Transactions on Robotics*, *32*(6), 1309–1332

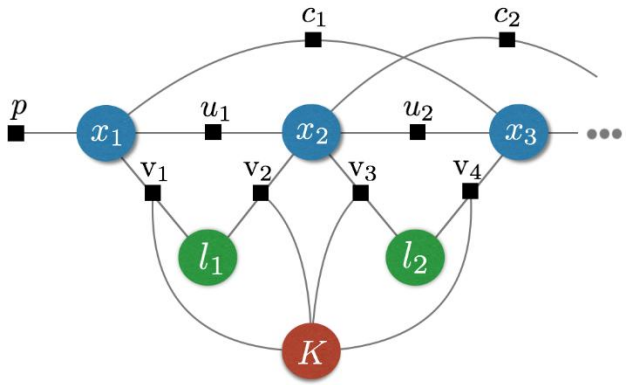
Components of VSLAM

- Short-term tracking
  - Tracking with respect to local keyframes
- Mid-term tracking
  - Loop closure detection in the local map
- Long-term tracking
  - Loop closure detection in the global map
- Mapping
  - Building and optimising the map over keyframes both locally and globally
  - Data fusion



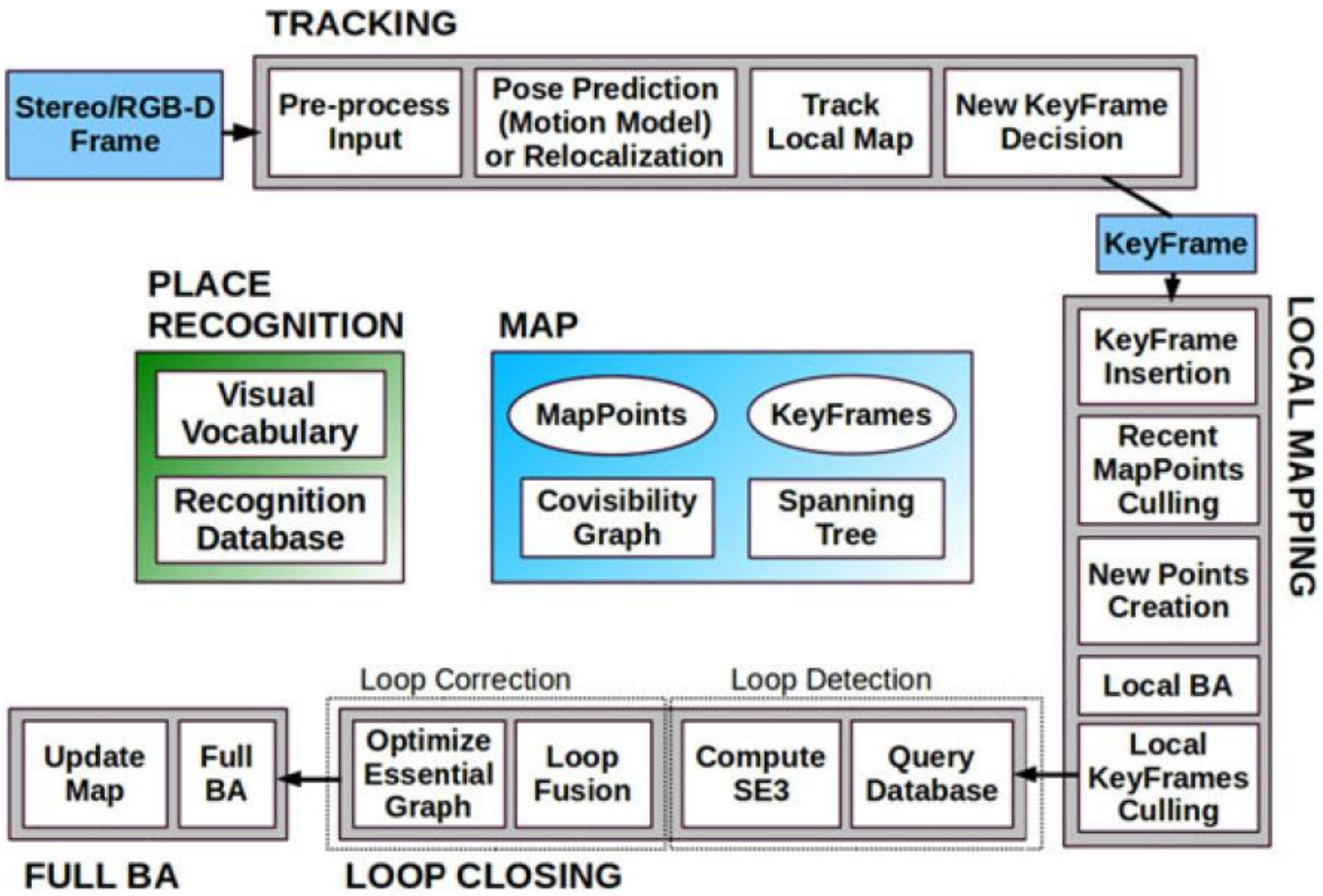


Lowry, S. et al. (2016). Visual Place Recognition: A Survey. IEEE Transactions on Robotics, 32(1), 1–19.



Cadena, C., et al. (2016). Past, Present, and Future of Simultaneous Localization and Mapping: Toward the Robust-Perception Age. *IEEE Transactions on Robotics*, *32*(6), 1309–1332

Example: ORB-SLAM 2



R. Mur-Artal and J. D. Tardos, “ORB-SLAM2: An Open-Source SLAM System for Monocular, Stereo, and RGB-D Cameras,” *IEEE Trans. Robot.*, pp. 1–8, 2017.

Part II

SHORT-TERM, MID-TERM AND LONG-TERM TRACKING

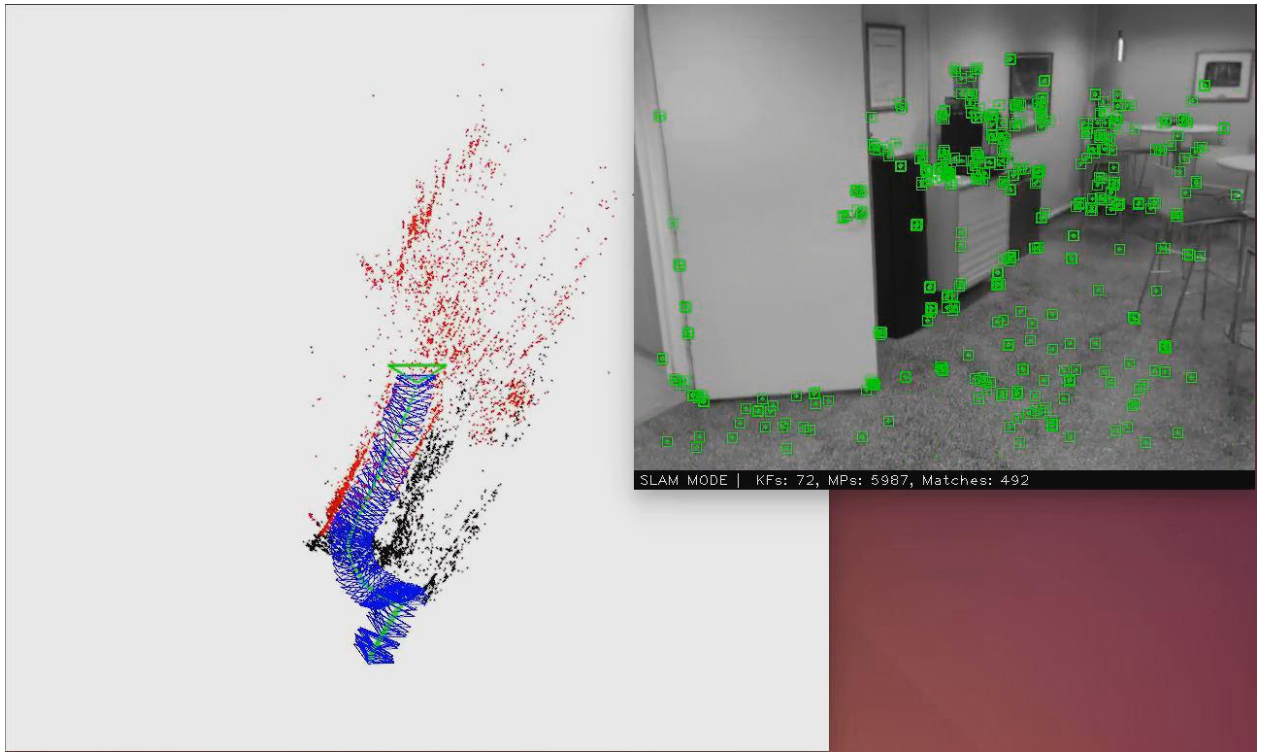
Tracking the map in VSLAM

We track the map for localisation

- Estimate the camera pose relative to the map for each frame

and for building a consistent map

- Detect loop closures



Tracking the map in VSLAM

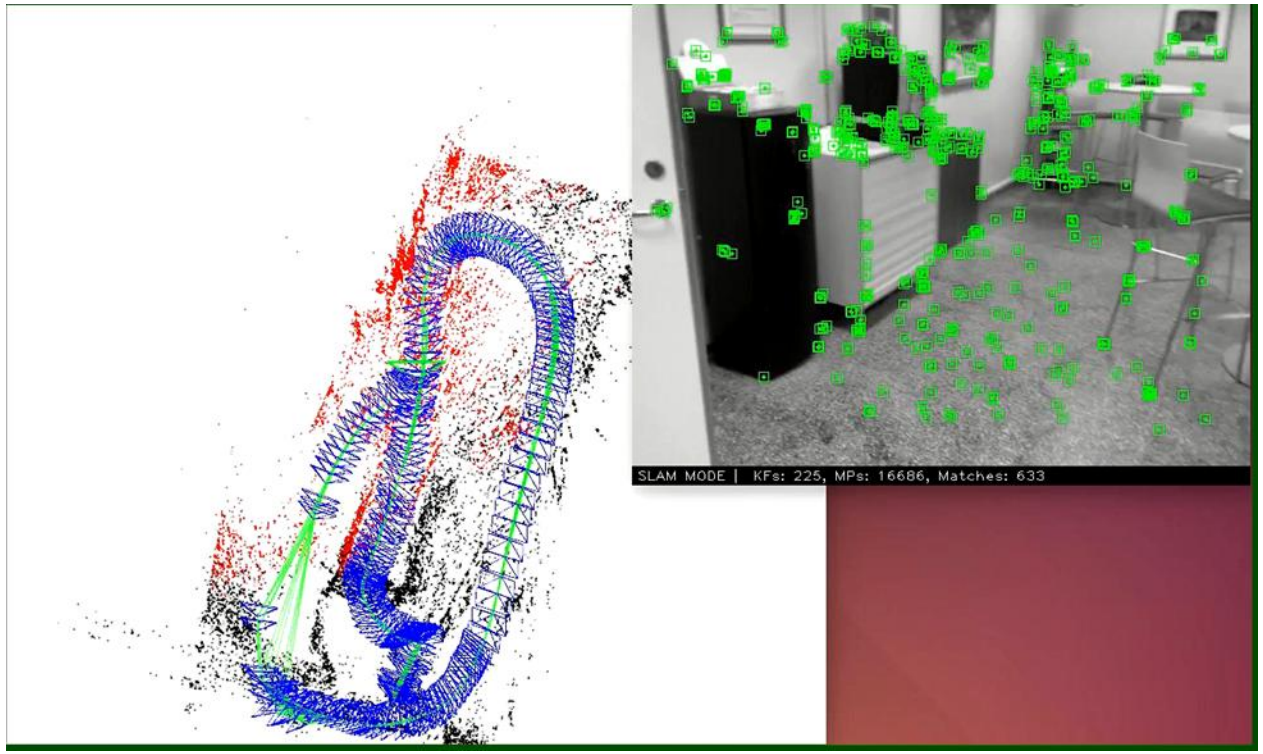
We track the map for localisation

- Estimate the camera pose relative to the map for each frame

and for building a consistent map

- Detect loop closures

These tasks have different *requirements, challenges and opportunities*



Short-term tracking for pose estimation

Requires:

- High tracking rate
- Precise pose estimate

Challenges:

- Fast correspondence search
- Many correspondences

Opportunities:

- A simple motion model often results in a good prediction for the next pose
- Conditions are almost the same, few changes

➢ It is often possible to significantly restrict the search for correspondences



Mid-term tracking for loop closure detection

Requires:

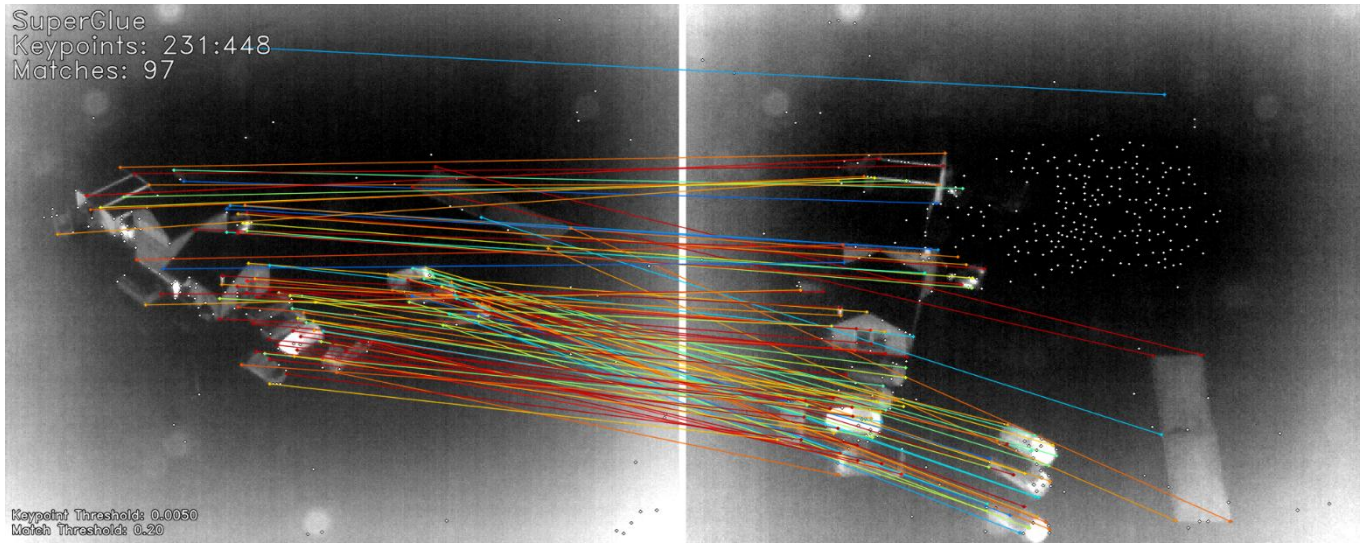
- Tracks across many views after a significant motion
- Relatively high tracking rate (keyframe rate)

Challenges:

- Different viewpoints
- Occlusions
- Several candidate keyframes

Opportunities:

- Do not need to run in frame rate
- We are close to previous keyframes
- ➢ We can restrict our search and exploit longer processing time



https://github.com/magicleap/SuperGluePretrainedNetwork

Sarlin, P. E., Detone, D., Malisiewicz, T., & Rabinovich, A. (2020). SuperGlue: Learning Feature Matching with Graph Neural Networks. Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 4937–4946.

Long-term tracking for loop closure detection

Requires:

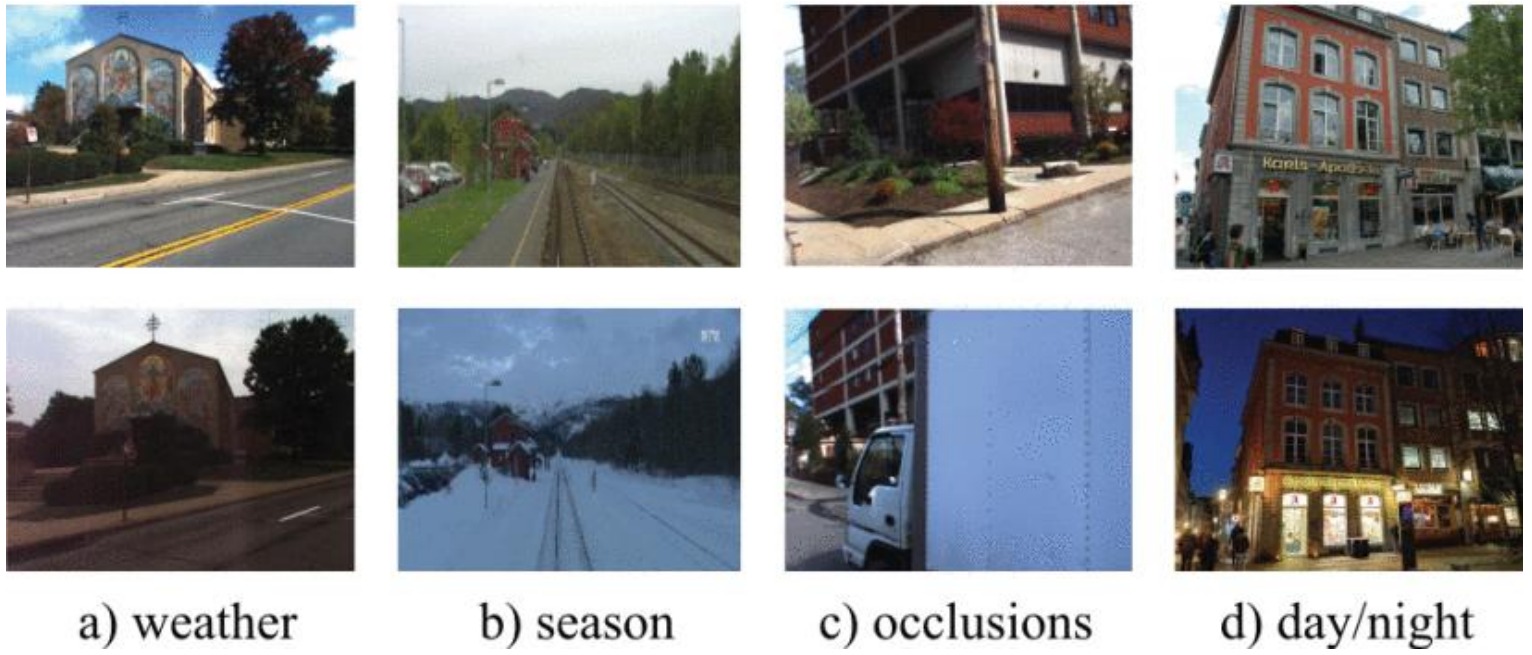
- Tracks across many views after a significant time
- Global search

Additional challenges:

- Changing conditions
- Changing scene
- A very large amount of candidate keyframes

Opportunities:

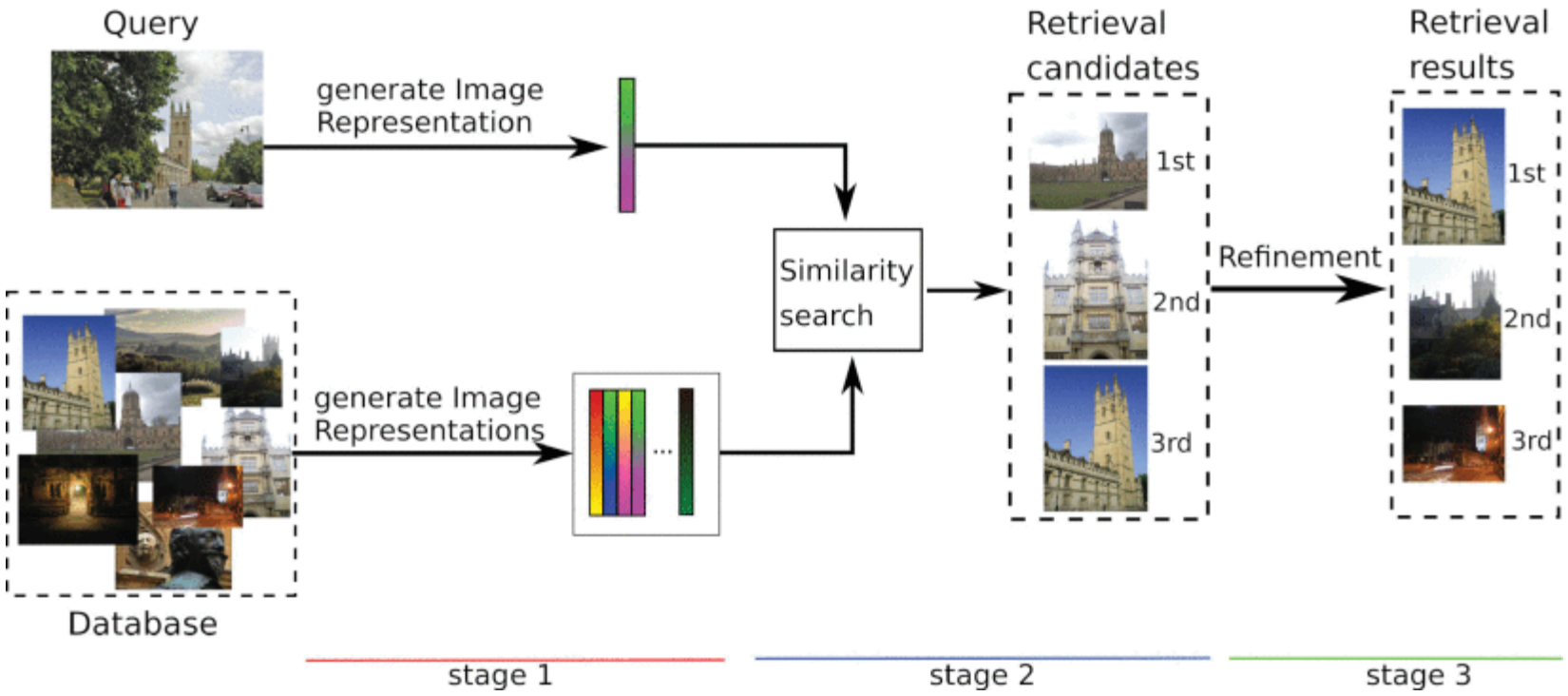
- ➢ We can exploit even longer processing time



a) weather b) season c) occlusions d) day/night

"A Survey on Deep Visual Place Recognition," C. Masone and B. Caputo, IEEE Access, vol. 9, pp. 19516-19547, 2021

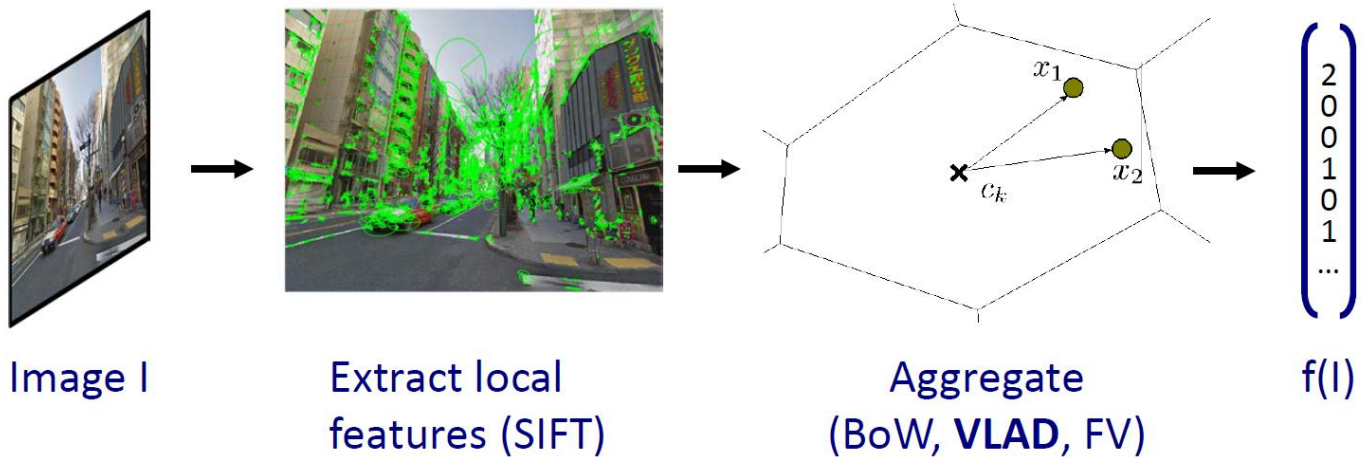
Image retrieval



"A Survey on Deep Visual Place Recognition," C. Masone and B. Caputo, IEEE Access, vol. 9, pp. 19516-19547, 2021

Image retrieval architectures

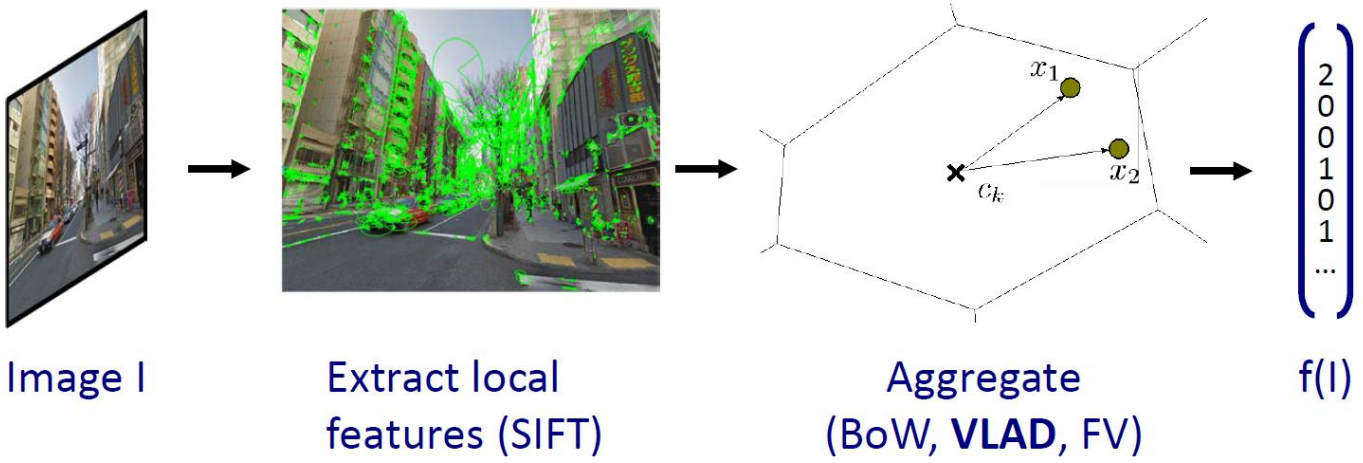
Classical approach



"Cross-weather-time, long term Visual Geo-Localization", R. Kumar, CVPR 2021 tutorial on Cross-view and Cross-modal Visual GeoLocalization

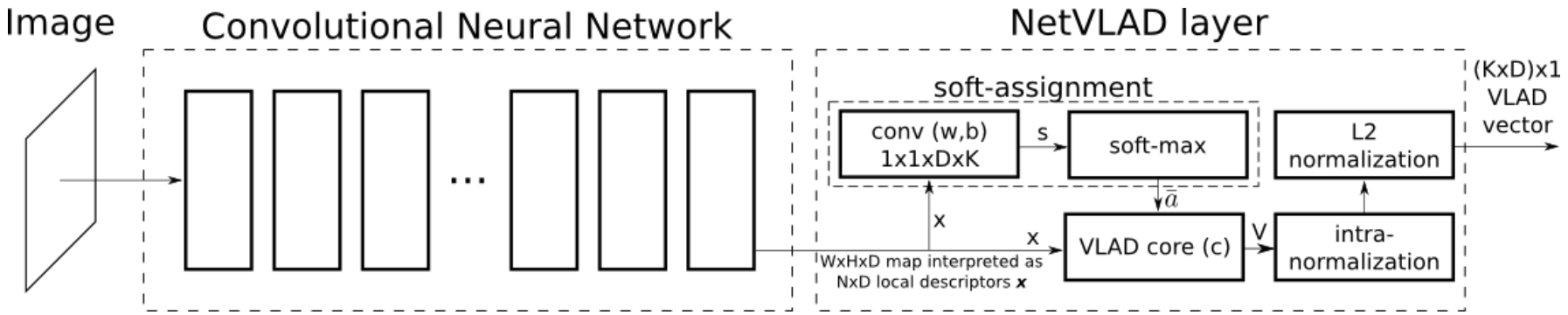
Image retrieval architectures

Classical approach



"Cross-weather-time, long term Visual Geo-Localization", R. Kumar, CVPR 2021 tutorial on Cross-view and Cross-modal Visual GeoLocalization

Trained end-to-end



Arandjelovic, R., Gronat, P., Torii, A., Pajdla, T., & Sivic, J. (2018). NetVLAD: CNN Architecture for Weakly Supervised Place Recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *40*(6), 1437–1451
https://www.di.ens.fr/willow/research/netvlad/

Supplementary material

“Visual Place Recognition: A Survey“,
Lowry, S. et al., IEEE Transactions on Robotics, 32 (1), pp 1–19, 2016
https://ieeexplore.ieee.org/document/7339473

"A Survey on Deep Visual Place Recognition,"
C. Masone and B. Caputo, IEEE Access, vol. 9, pp. 19516-19547, 2021
doi: 10.1109/ACCESS.2021.3054937.

“Cross-weather-time, long term Visual Geo-Localization”
R. Kumar, CVPR 2021 tutorial on Cross-view and Cross-modal Visual GeoLocalization
https://www.sri.com/computer-vision/cvpr-2021-tutorial-on-cross-view-and-cross-modal-visual-geo-localization/

Part III

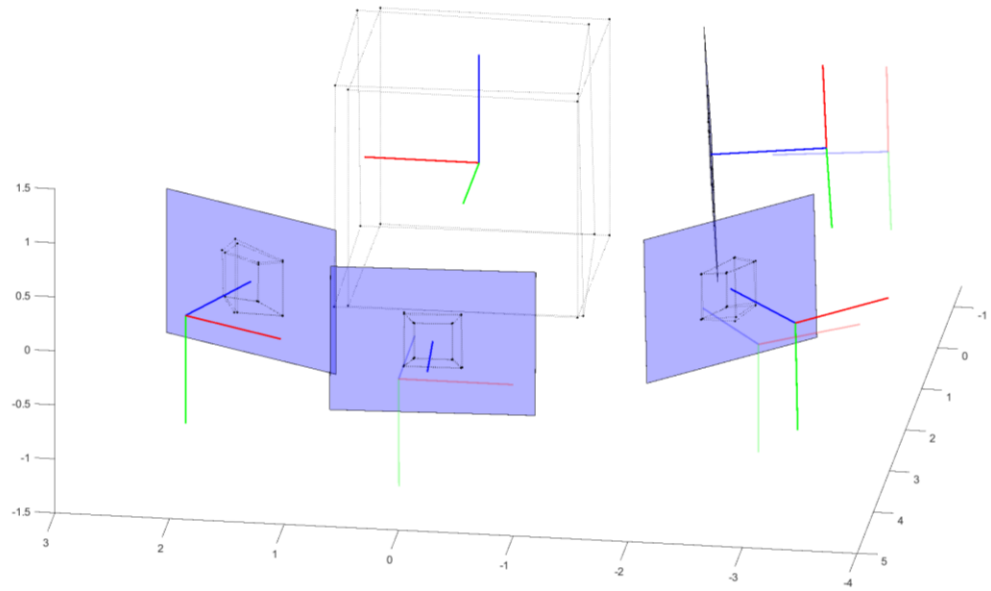
MAPPING WITH FACTOR GRAPHS

Maximum a posteriori inference

Interested in the unknown state variables X, given the measurements Z.

The most often used estimator for X is the MAP estimate:

$$
\begin{aligned}
X^{\text{MAP}} &= \underset{X}{\operatorname{argmax}}\, p(X \mid Z) \\
&= \underset{X}{\operatorname{argmax}}\, \frac{p(Z \mid X)p(X)}{p(Z)} \\
&= \underset{X}{\operatorname{argmax}}\, l(X;Z)p(X) \\
& \qquad l(X;Z) \propto p(Z \mid X)
\end{aligned}
$$



Maximum a posteriori inference

Measurement model:

$$\mathbf{z}_i = h_i(X_i) + \eta, \qquad \eta \sim N(\mathbf{0}, \boldsymbol{\Sigma}_i)$$

Measurement prediction function:

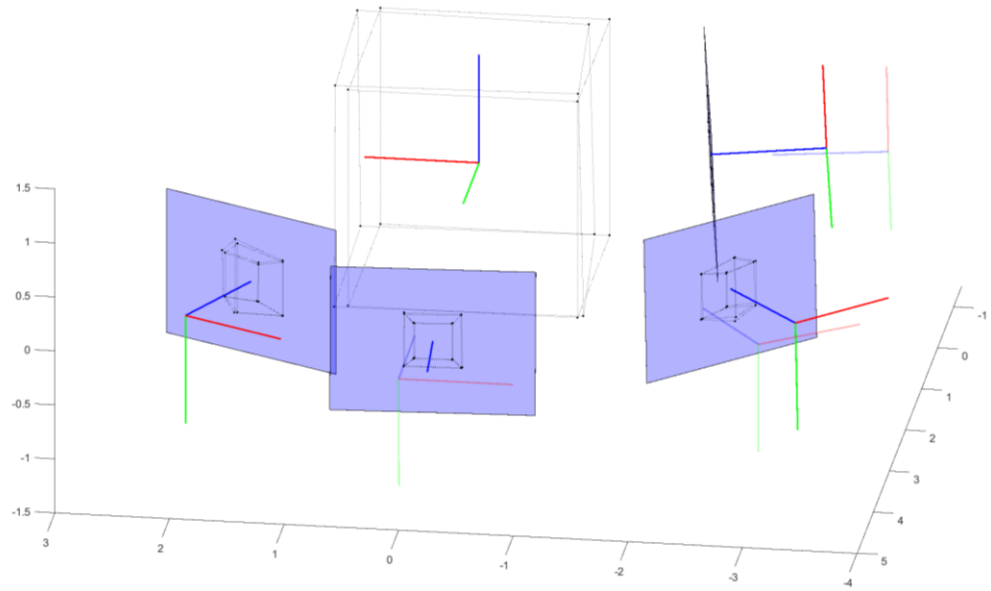
$$\hat{\mathbf{z}}_i = h_i(X_i)$$

Measurement likelihood:

$$p(\mathbf{z}_i \mid X_i) \propto l(X_i; \mathbf{z}_i) = \exp\left(-\frac{1}{2} \left\| h_i(X_i) - \mathbf{z}_i \right\|^2_{\boldsymbol{\Sigma}_i} \right)$$

MAP estimate:

$$X^{\text{MAP}} = \underset{X}{\operatorname{argmin}} \sum_i \left\| h_i(X_i) - \mathbf{z}_i \right\|^2_{\boldsymbol{\Sigma}_i}$$



$$\left\{ \mathbf{T}^*_{wc_i}, \mathbf{x}^{w*}_j \right\} = \underset{\mathbf{T}_{wc_i}, \mathbf{x}^w_j}{\operatorname{argmin}} \sum_i \sum_j \left\| \pi_i(\mathbf{T}^{-1}_{wc_i} \cdot \mathbf{x}^w_j) - \mathbf{u}^i_j \right\|^2$$

Maximum a posteriori inference

Measurement model:

$$\mathbf{z}_i = h_i(X_i) + \eta, \qquad \eta \sim N(\mathbf{0}, \boldsymbol{\Sigma}_i)$$

Measurement prediction function:

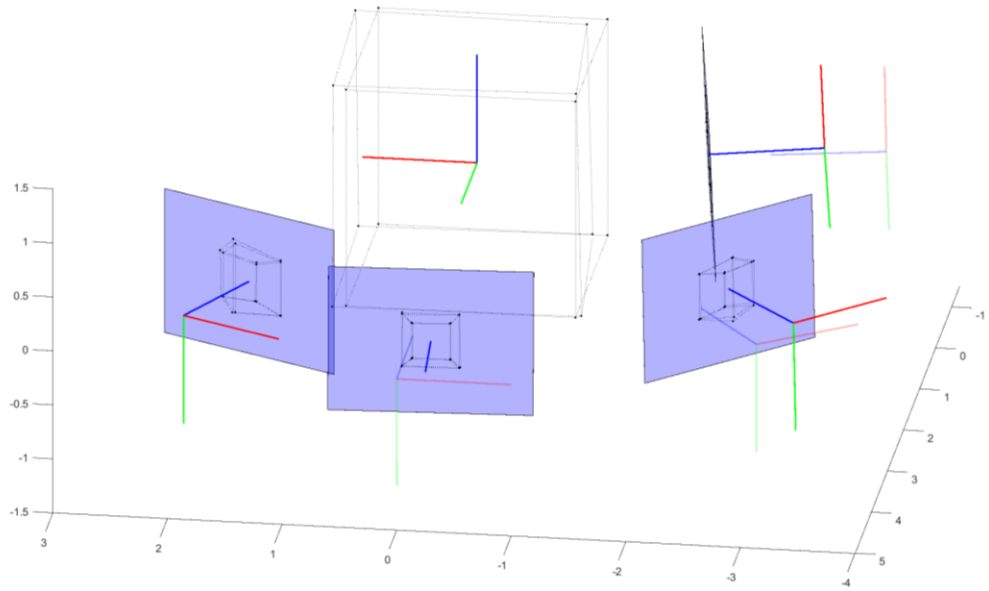
$$\hat{\mathbf{z}}_i = h_i(X_i)$$

Measurement likelihood:

$$p(\mathbf{z}_i \mid X_i) \propto l(X_i; \mathbf{z}_i) = \exp\left(-\frac{1}{2} \left\| h_i(X_i) - \mathbf{z}_i \right\|^2_{\boldsymbol{\Sigma}_i} \right)$$

MAP estimate:

$$X^{\mathrm{MAP}} = \underset{X}{\operatorname{argmin}} \sum_i \left\| h_i(X_i) - \mathbf{z}_i \right\|^2_{\boldsymbol{\Sigma}_i}$$



Applying the MAP framework

This results in the linearised weighted least squares problem

$$\begin{aligned} \underline{\boldsymbol{\tau}}^* &= \underset{\underline{\boldsymbol{\tau}}}{\arg\min} \sum_{i=1}^{k} \sum_{j=1}^{n} \left\| \mathbf{P}_{ij} \boldsymbol{\xi}_i + \mathbf{S}_{ij} \delta \mathbf{x}_j - \mathbf{b}_{ij} \right\|^2 \\ &= \underset{\underline{\boldsymbol{\tau}}}{\arg\min} \left\| \mathbf{A} \underline{\boldsymbol{\tau}} - \mathbf{b} \right\|^2, \end{aligned}$$

where

$$\begin{aligned} \mathbf{P}_{ij} &= \boldsymbol{\Sigma}_{n\,ij}^{-1/2} \mathbf{J}_{\mathbf{T}_{wc_i}}^{h_{ij}} \\ \mathbf{S}_{ij} &= \boldsymbol{\Sigma}_{n\,ij}^{-1/2} \mathbf{J}_{\mathbf{x}_j^w}^{h_{ij}} \\ \mathbf{b}_{ij} &= \boldsymbol{\Sigma}_{n\,ij}^{-1/2} (\mathbf{x}_{n\,j}^i - h_{ij}(\mathbf{T}_{wc_i}, \mathbf{x}_j^w)), \end{aligned}$$

$$\mathbf{A} = \begin{bmatrix} \mathbf{P}_{11} & & & \mathbf{S}_{11} & & \\ \vdots & & & & \ddots & \\ \mathbf{P}_{1n} & & & & & \mathbf{S}_{1n} \\ & \ddots & & & \vdots & \\ & & \mathbf{P}_{k1} & \mathbf{S}_{k1} & & \\ & & \vdots & & \ddots & \\ & & \mathbf{P}_{kn} & & & \mathbf{S}_{kn} \end{bmatrix} \quad \underline{\boldsymbol{\tau}} = \begin{bmatrix} \boldsymbol{\xi}_1 \\ \vdots \\ \boldsymbol{\xi}_k \\ \delta \mathbf{x}_1 \\ \vdots \\ \delta \mathbf{x}_n \end{bmatrix} \quad \mathbf{b} = \begin{bmatrix} \mathbf{b}_{11} \\ \vdots \\ \mathbf{b}_{1n} \\ \vdots \\ \mathbf{b}_{k1} \\ \vdots \\ \mathbf{b}_{kn} \end{bmatrix}.$$

Maximum a posteriori inference and factor graphs

Measurement model:

$$\mathbf{z}_i = h_i(X_i) + \eta, \qquad \eta \sim N(\mathbf{0}, \boldsymbol{\Sigma}_i)$$

Measurement prediction function:

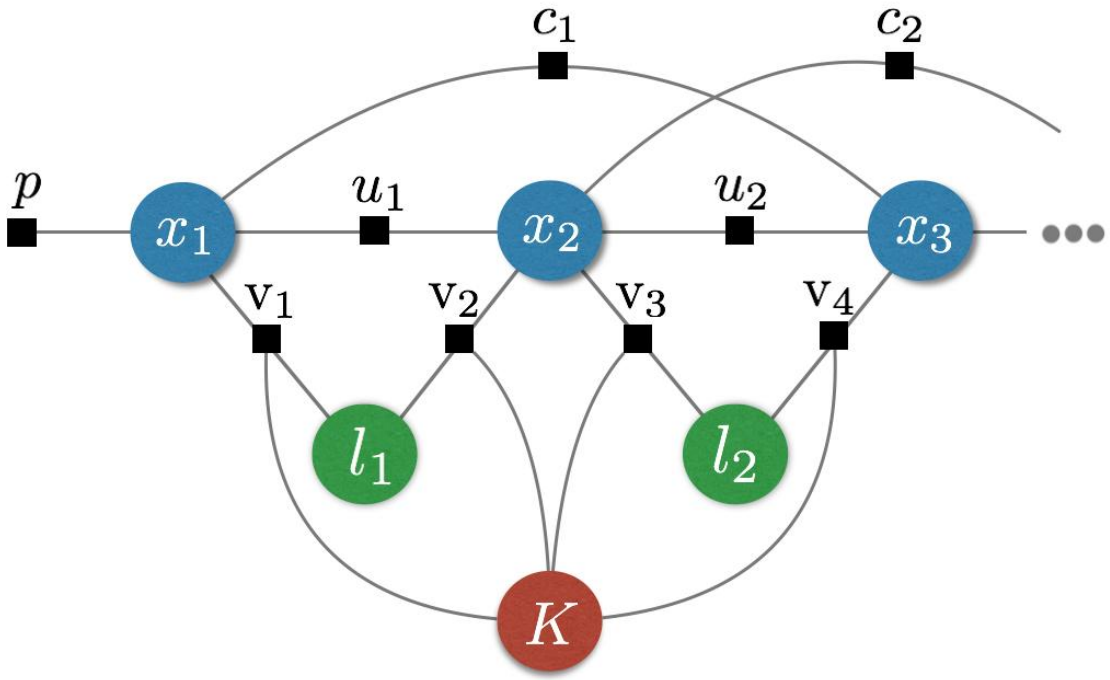
$$\hat{\mathbf{z}}_i = h_i(X_i)$$

Measurement likelihood:

$$p(\mathbf{z}_i \mid X_i) \propto l(X_i; \mathbf{z}_i) = \exp\left(-\frac{1}{2} \left\| h_i(X_i) - \mathbf{z}_i \right\|^2_{\boldsymbol{\Sigma}_i} \right)$$

MAP estimate:

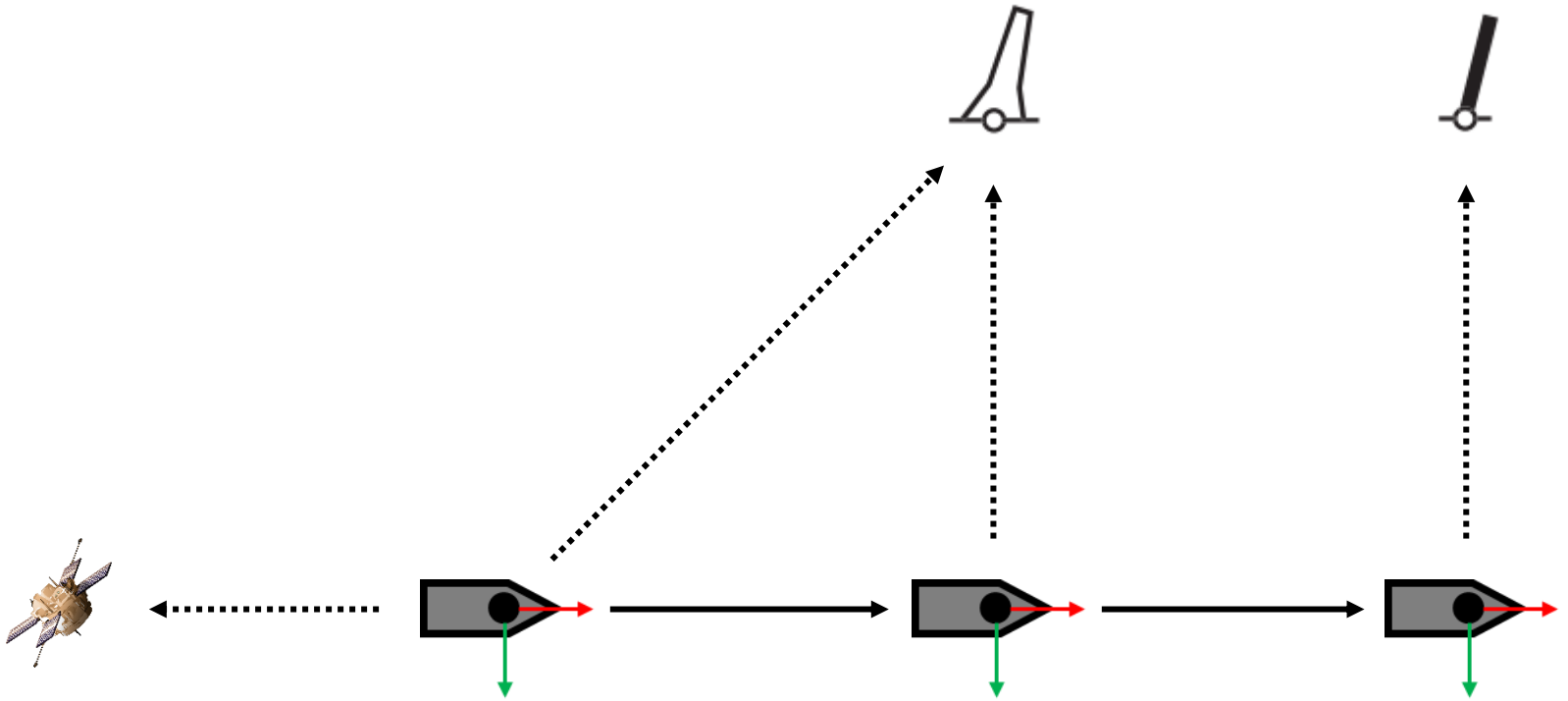
$$X^{\mathrm{MAP}} = \underset{X}{\operatorname{argmin}} \sum_i \left\| h_i(X_i) - \mathbf{z}_i \right\|^2_{\boldsymbol{\Sigma}_i}$$



Cadena, C., et al. (2016). Past, Present, and Future of Simultaneous Localization and Mapping: Toward the Robust-Perception Age. *IEEE Transactions on Robotics*, *32*(6), 1309–1332

Maximum a posteriori inference and factor graphs

Simple SLAM example



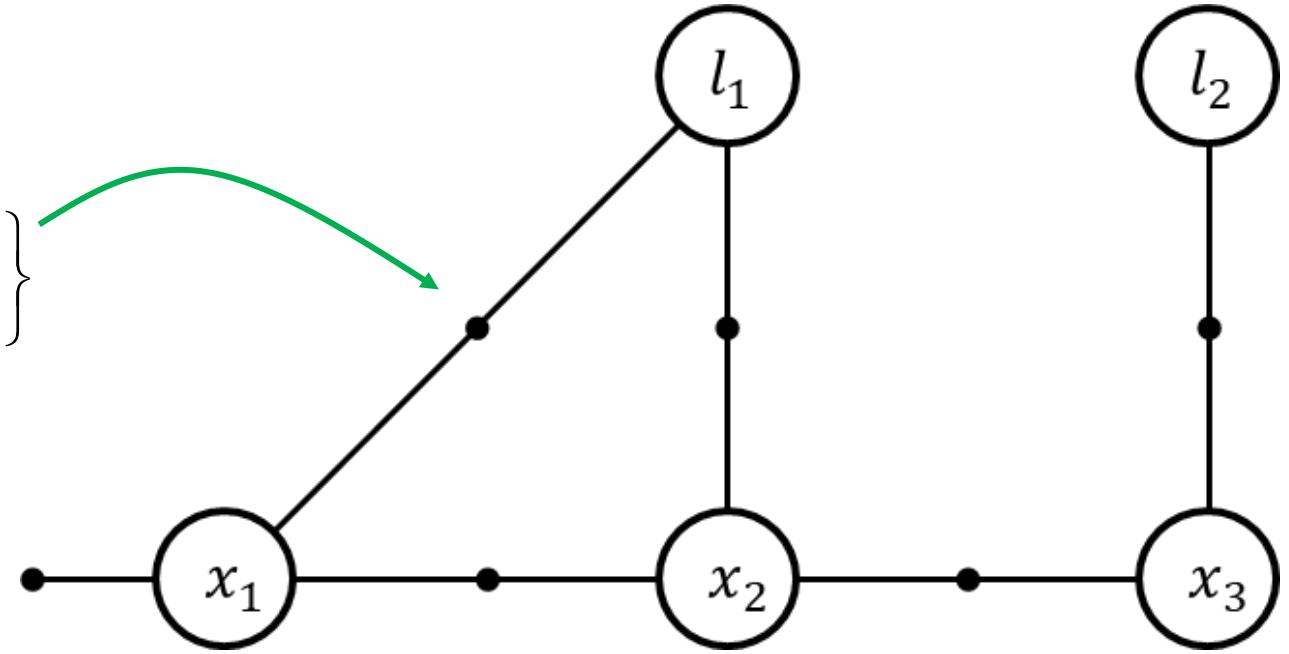
https://github.com/tussedrotten/simple-factorgraph-example

Maximum a posteriori inference and factor graphs

Simple SLAM example

$$\begin{bmatrix} r \\ \alpha \end{bmatrix} = \rho(\mathbf{x}^r) = \begin{bmatrix} \sqrt{x^2 + y^2} \\ \arctan\left(\frac{y}{x}\right) \end{bmatrix}$$

$$\phi_i(X_i) \propto \exp\left\{-\frac{1}{2}\|h_i(X_i) - z_i\|^2_{\Sigma_i}\right\}$$

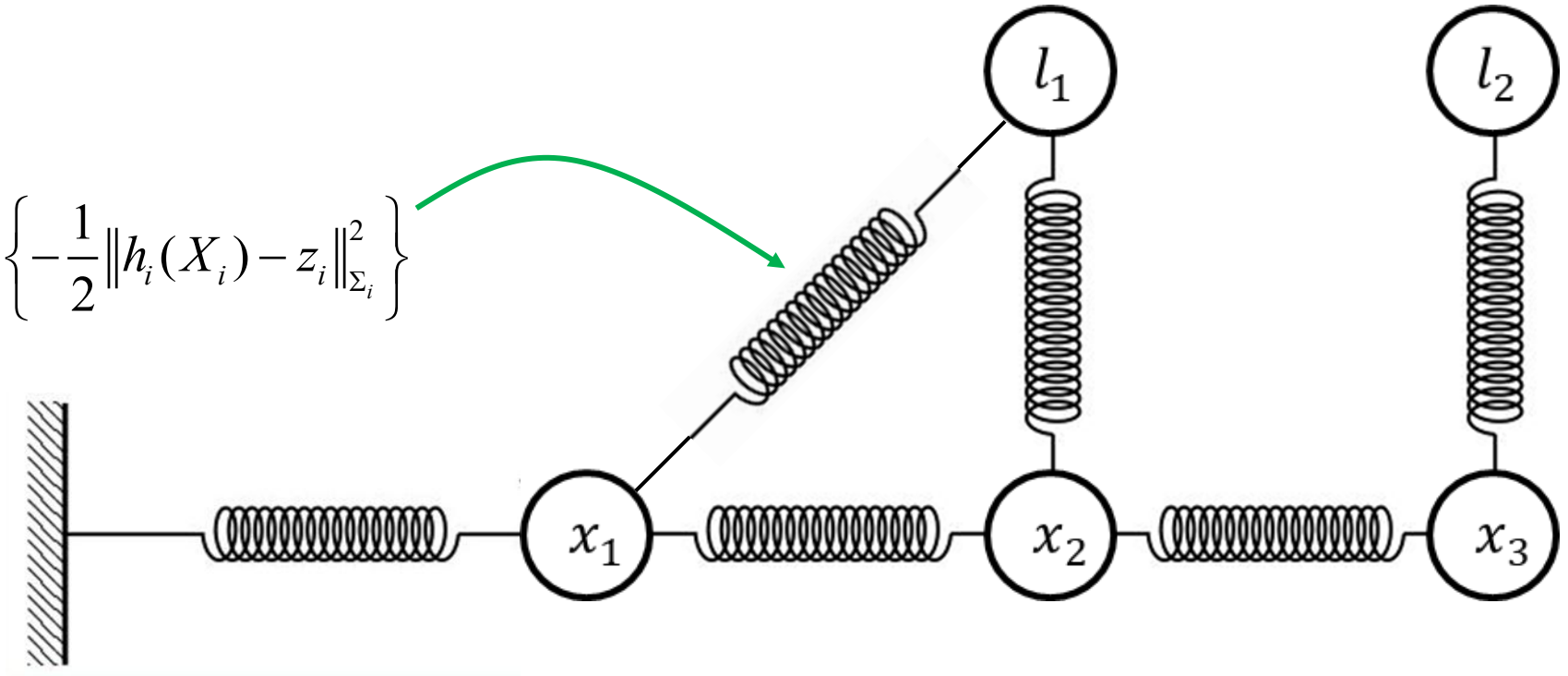


https://github.com/tussedrotten/simple-factorgraph-example

Maximum a posteriori inference and factor graphs

Simple SLAM example

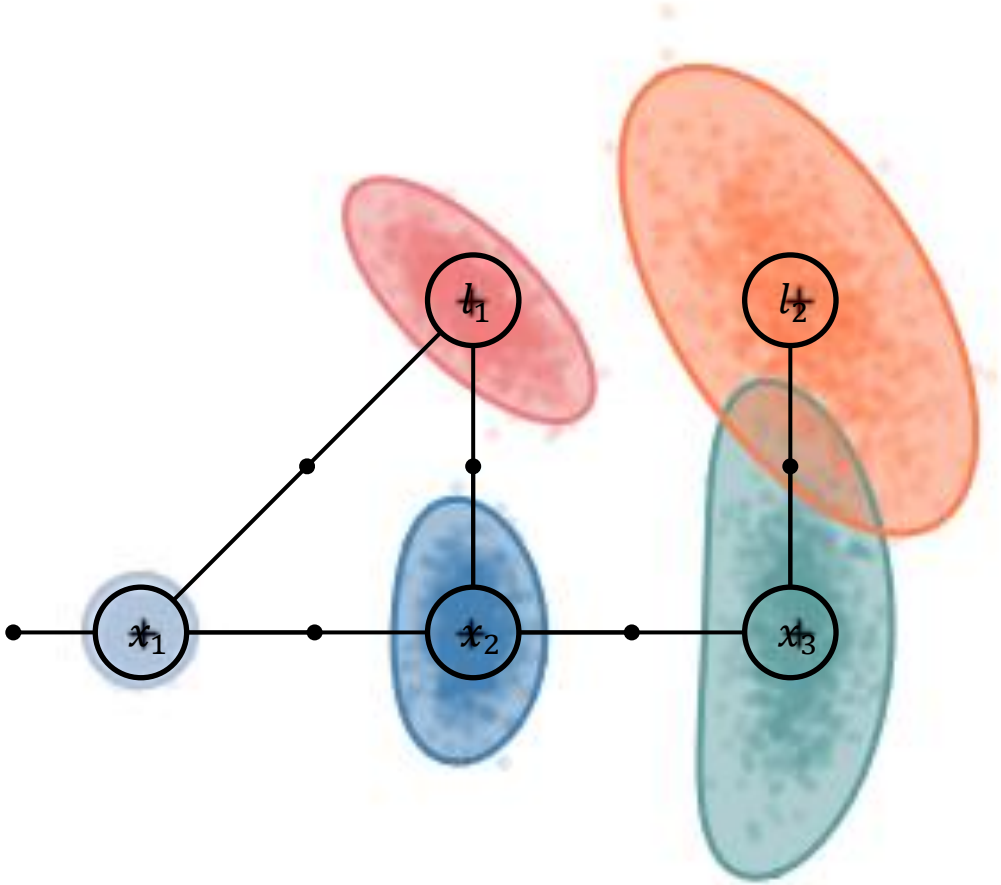
$$\phi_i(X_i) \propto \exp\left\{-\frac{1}{2}\|h_i(X_i) - z_i\|^2_{\Sigma_i}\right\}$$



https://github.com/tussedrotten/simple-factorgraph-example

Maximum a posteriori inference and factor graphs

Simple SLAM example



https://github.com/tussedrotten/simple-factorgraph-example

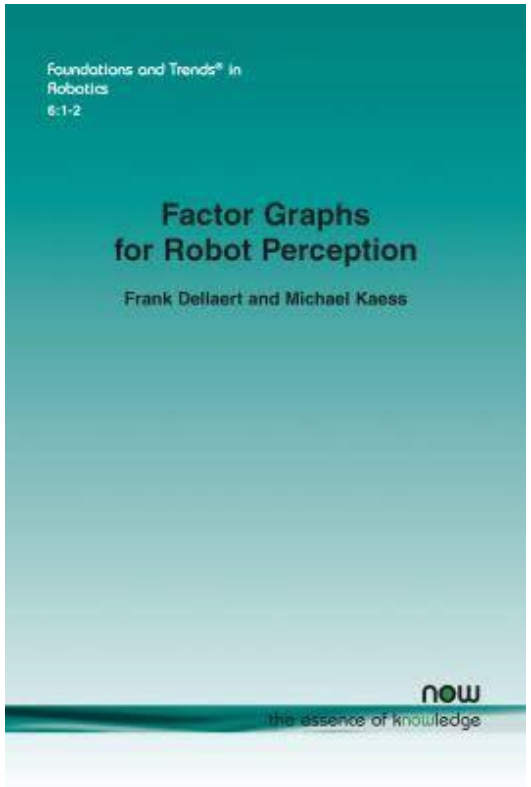
Supplementary material

Georgia Tech Smoothing and Mapping library

- https://gtsam.org/
- https://github.com/borglab/gtsam

Tutorial: https://gtsam.org/tutorials/intro.html

Factor Graphs for Robot Perception
by Frank Dellaert and Michael Kaess
https://www.cc.gatech.edu/~dellaert/pubs/Dellaert17fnt.pdf



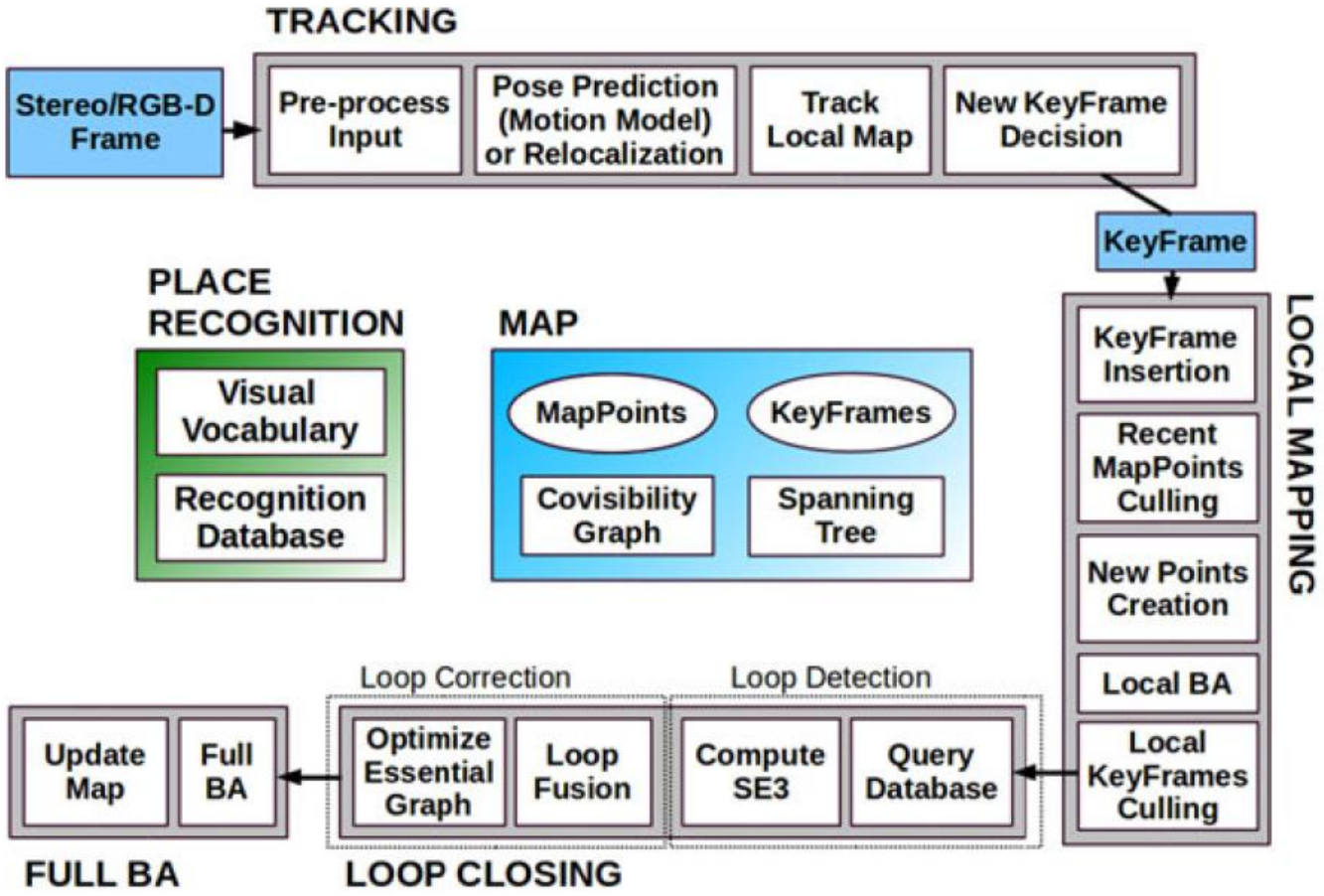
Part IV

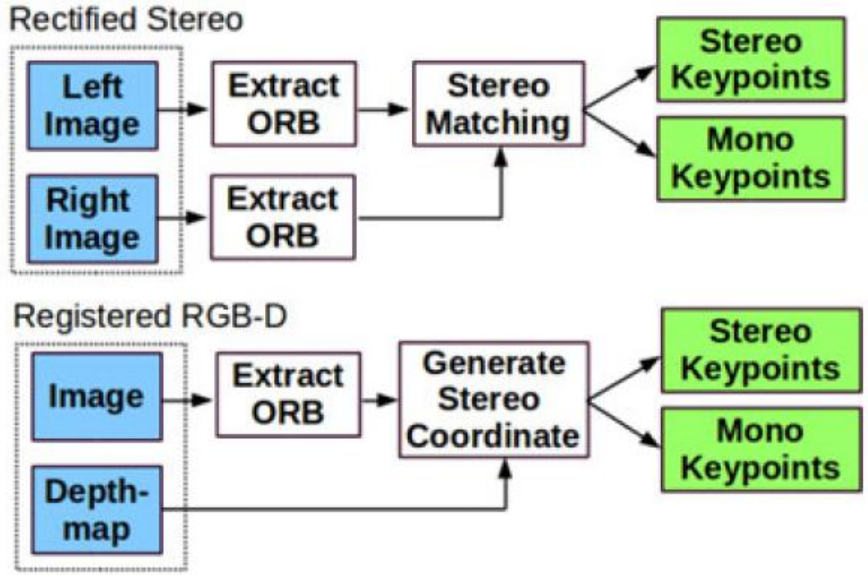
VSLAM SYSTEMS

	SLAM or VO	Pixels used	Data association	Estimation	Relocali-zation	Loop closing	Multi Maps	Mono	Stereo	Mono IMU	Stereo IMU	Fisheye	Accuracy	Robustness	Open source
Mono-SLAM [13], [14]	SLAM	Shi Tomasi	Correlation	EKF	-	-	-	✓	-	-	-	-	Fair	Fair	[15][1]
PTAM [16]–[18]	SLAM	FAST	Pyramid SSD	BA	Thumbnail	-	-	✓	-	-	-	-	Very Good	Fair	[19]
LSD-SLAM [20], [21]	SLAM	Edgelets	Direct	PG	-	FABMAP PG	-	✓	✓	-	-	-	Good	Fair	[22]
SVO [23], [24]	VO	FAST+ Hi.grad.	Direct	Local BA	-	-	-	✓	✓	-	-	✓	Very Good	Very Good	[25][2]
ORB-SLAM2 [2], [3]	SLAM	ORB	Descriptor	Local BA	DBoW2	DBoW2 PG+BA	-	✓	✓	-	-	-	Exc.	Very Good	[26]
DSO [27]–[29]	VO	High grad.	Direct	Local BA	-	-	-	✓	✓	-	-	✓	Fair	Very Good	[30]
DSM [31]	SLAM	High grad.	Direct	Local BA	-	-	-	✓	-	-	-	-	Very Good	Very Good	[32]
MSCKF [33]–[36]	VO	Shi Tomasi	Cross correlation	EKF	-	-	-	✓	-	✓	✓	-	Fair	Very Good	[37][3]
OKVIS [38], [39]	VO	BRISK	Descriptor	Local BA	-	-	-	-	-	✓	✓	✓	Good	Very Good	[40]
ROVIO [41], [42]	VO	Shi Tomasi	Direct	EKF	-	-	-	-	-	✓	✓	✓	Good	Very Good	[43]
ORBSLAM-VI [4]	SLAM	ORB	Descriptor	Local BA	DBoW2	DBoW2 PG+BA	-	✓	-	✓	-	-	Very Good	Very Good	-
VINS-Fusion [7], [44]	VO	Shi Tomasi	KLT	Local BA	DBoW2	DBoW2 PG	✓	-	✓	✓	✓	✓	Good	Exc.	[45]
VI-DSO [46]	VO	High grad.	Direct	Local BA	-	-	-	-	-	✓	-	-	Very Good	Exc.	-
BASALT [47]	VO	FAST	KLT (LSSD)	Local BA	-	ORB BA	-	-	-	-	✓	✓	Very Good	Exc.	[48]
Kimera [8]	VO	Shi Tomasi	KLT	Local BA	-	DBoW2 PG	-	-	-	-	✓	-	Good	Exc.	[49]
ORB-SLAM3 (ours)	SLAM	ORB	Descriptor	Local BA	DBoW2	DBoW2 PG+BA	✓	✓	✓	✓	✓	✓	Exc.	Exc.	[5]

C. Campos, R. Elvira, J. J. G. Rodríguez, J. M. M. Montiel and J. D. Tardós, "ORB-SLAM3: An Accurate Open-Source Library for Visual, Visual–Inertial, and Multimap SLAM," in IEEE Transactions on Robotics, doi: 10.1109/TRO.2021.3075644.

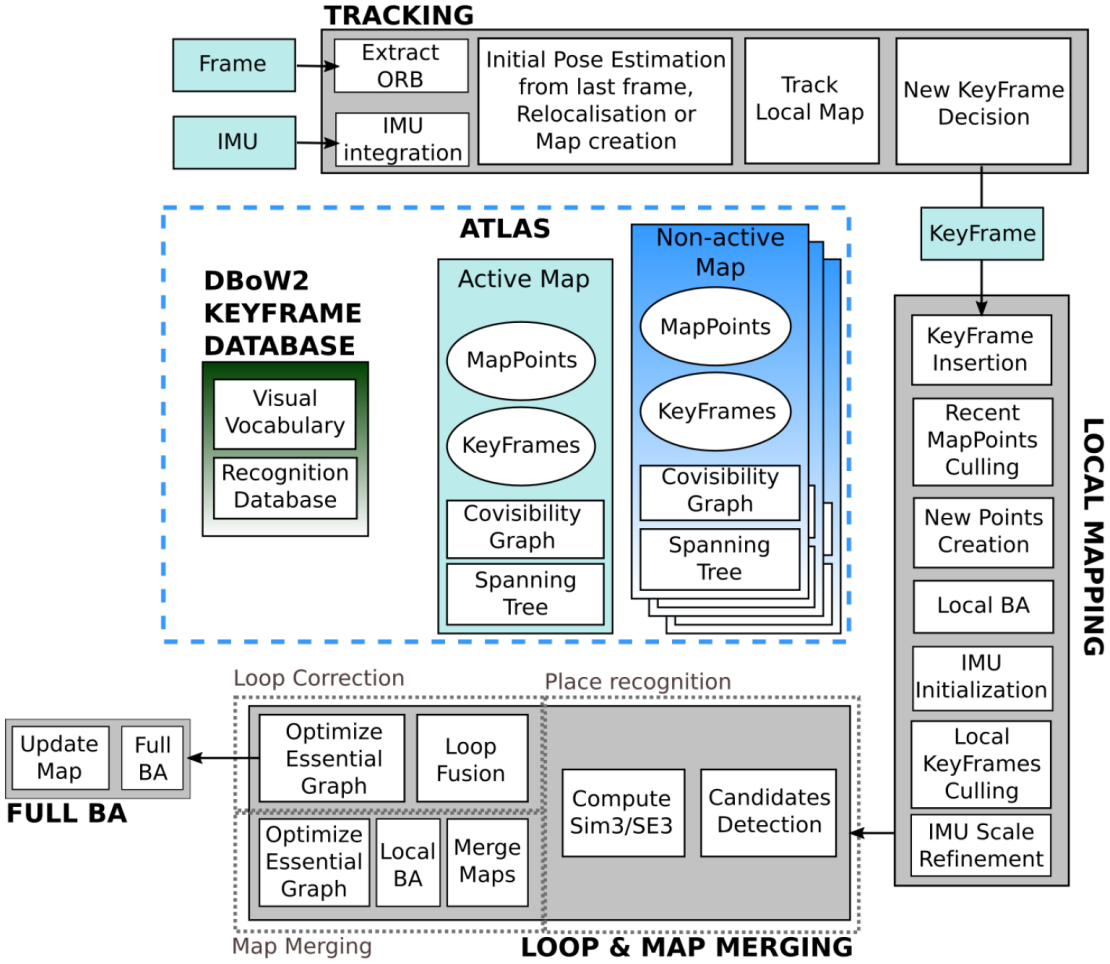
ORB-SLAM 2 system overview





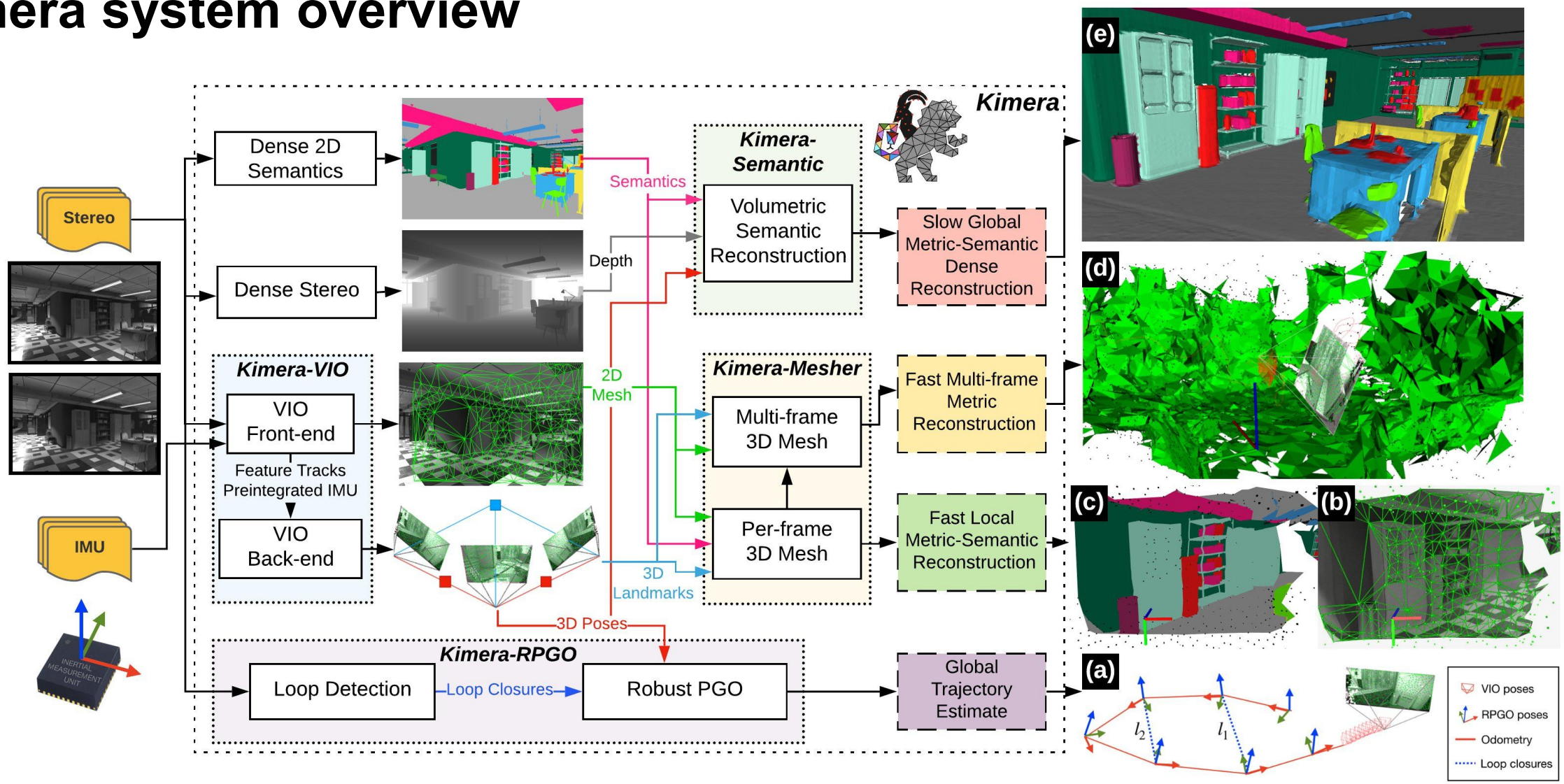
R. Mur-Artal and J. D. Tardos, "ORB-SLAM2: An Open-Source SLAM System for Monocular, Stereo, and RGB-D Cameras," IEEE Trans. Robot., pp. 1–8, 2017.

ORB-SLAM 3 system overview



C. Campos, R. Elvira, J. J. G. Rodríguez, J. M. M. Montiel and J. D. Tardós, "ORB-SLAM3: An Accurate Open-Source Library for Visual, Visual–Inertial, and Multimap SLAM," in IEEE Transactions on Robotics, doi: 10.1109/TRO.2021.3075644.

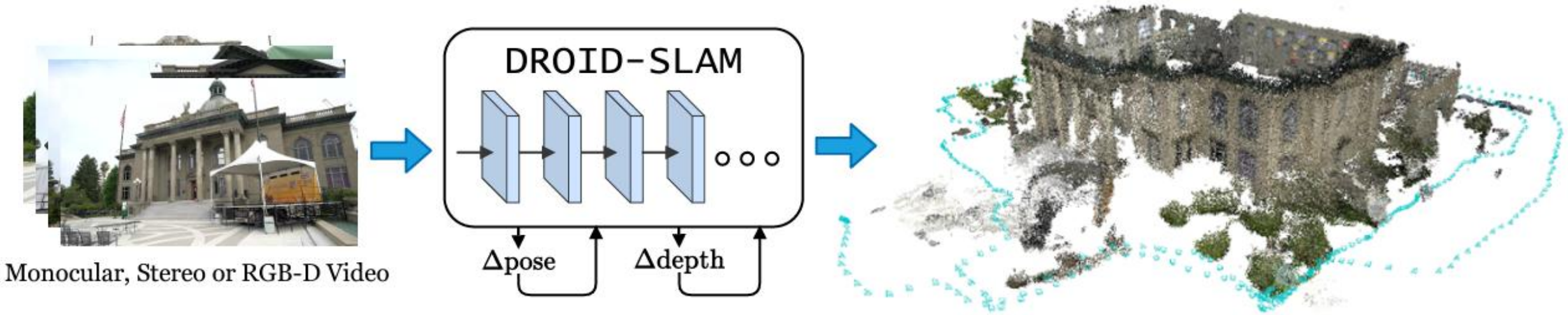
Kimera system overview



https://github.com/MIT-SPARK/Kimera

DROID-SLAM

Code: https://github.com/princeton-vl/DROID-SLAM

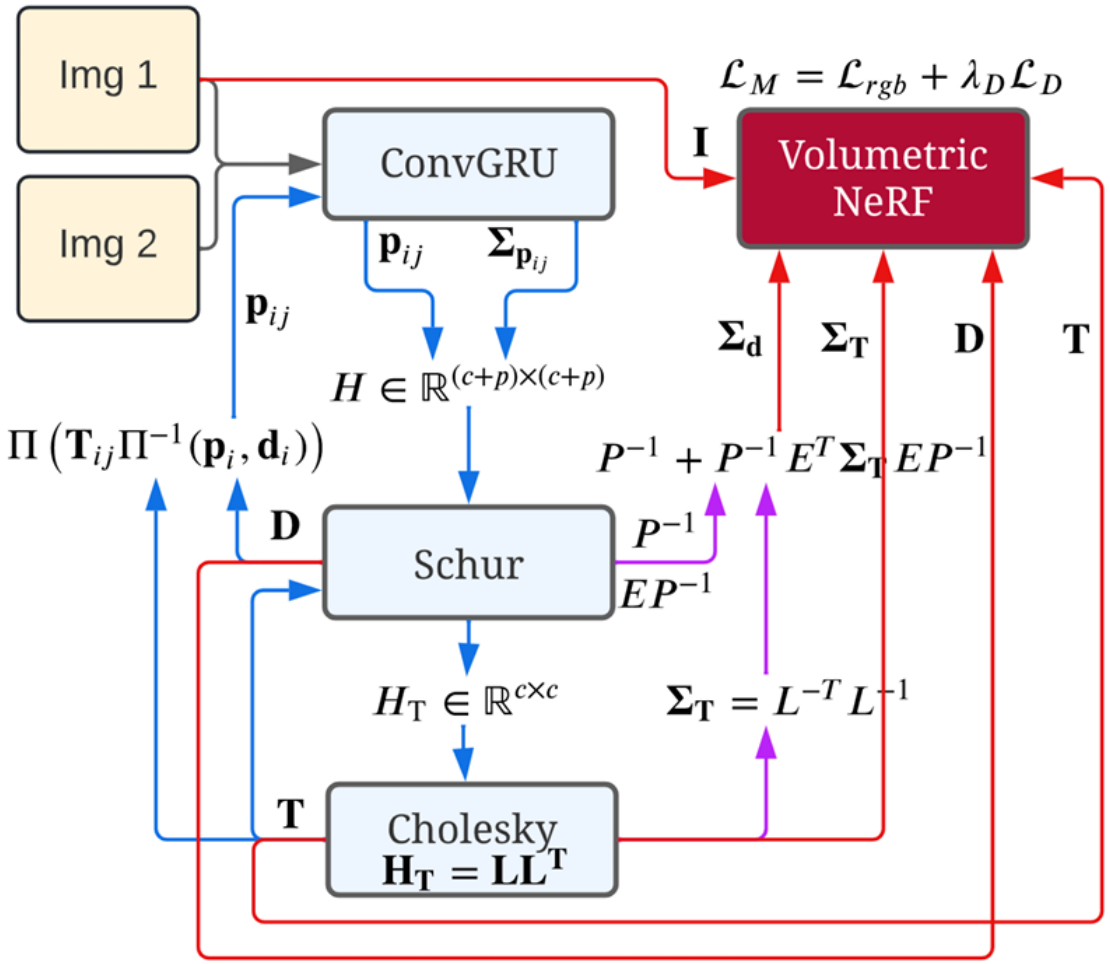


- DROID: Differentiable Recurrent Optimization-Inspired Design
- Indirect and dense!
- Learned update operator
- Local VO over fixed number of keyframes + global SLAM over all keyframes

Teed. Zachary, Deng, Jia. (2021). DROID-SLAM: Deep Visual SLAM for Monocular, Stereo, and RGB-D Cameras
Arxiv. https://arxiv.org/abs/2108.10869

NERF-SLAM

Code: https://github.com/ToniRV/NeRF-SLAM

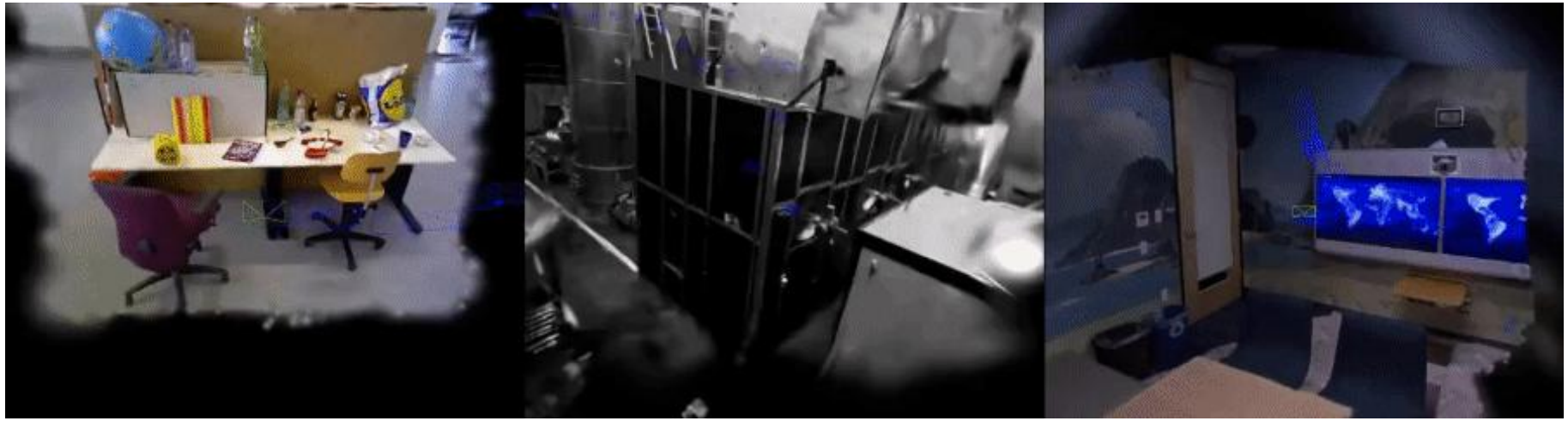


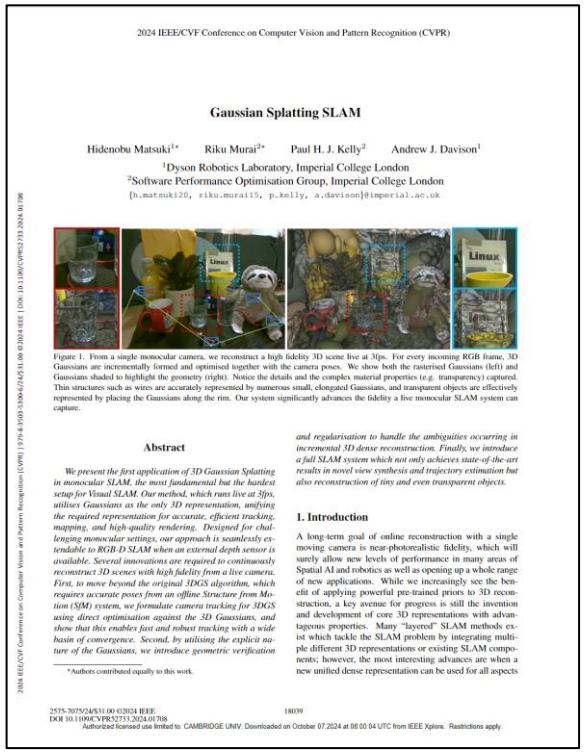


Rosinol, Antoni, Leonard, John, Carlone, Luca (2022). NeRF-SLAM: Real-Time Dense Monocular SLAM with Neural Radiance Fields
Arxiv. https://arxiv.org/abs/2210.13641

Gaussian Splatting SLAM

Code: https://github.com/muskie82/MonoGS





2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)

Gaussian Splatting SLAM

Hidenobu Matsuki[1]* Riku Murai[2]* Paul H. J. Kelly[2] Andrew J. Davison[1]

[1]Dyson Robotics Laboratory, Imperial College London
[2]Software Performance Optimisation Group, Imperial College London
{h.matsuki20, riku.murai15, p.kelly, a.davison}@imperial.ac.uk

Figure 1. From a single monocular camera, we reconstruct a high fidelity 3D scene live at 3fps. For every incoming RGB frame, 3D Gaussians are incrementally formed and optimised together with the camera poses. We show both the rasterised Gaussians (left) and Gaussians shaded to highlight the geometry (right). Notice the details and the complex material properties (e.g. transparency) captured. Thin structures such as wires are accurately represented by numerous small, elongated Gaussians, and transparent objects are effectively represented by placing the Gaussians along the rim. Our system significantly advances the fidelity a live monocular SLAM system can capture.

Abstract

We present the first application of 3D Gaussian Splatting in monocular SLAM, the most fundamental but the hardest setup for Visual SLAM. Our method, which runs live at 3fps, utilises Gaussians as the only 3D representation, unifying the required representation for accurate, efficient tracking, mapping, and high-quality rendering. Designed for challenging monocular settings, our approach is seamlessly extendable to RGB-D SLAM when an external depth sensor is available. Several innovations are required to continuously reconstruct 3D scenes with high fidelity from a live camera. First, to move beyond the original 3DGS algorithm, which requires accurate poses from an offline Structure from Motion (SfM) system, we formulate camera tracking for 3DGS using direct optimisation against the 3D Gaussians, and show that this enables fast and robust tracking with a wide basin of convergence. Second, by utilising the explicit nature of the Gaussians, we introduce geometric verification and regularisation to handle the ambiguities occurring in incremental 3D dense reconstruction. Finally, we introduce a full SLAM system which not only achieves state-of-the-art results in novel view synthesis and trajectory estimation but also reconstruction of tiny and even transparent objects.

1. Introduction

A long-term goal of online reconstruction with a single moving camera is near-photorealistic fidelity, which will surely allow new levels of performance in many areas of Spatial AI and robotics as well as opening up a whole range of new applications. While we increasingly see the benefit of applying powerful pre-trained priors to 3D reconstruction, a key avenue for progress is still the invention and development of core 3D representations with advantageous properties. Many "layered" SLAM methods exist which tackle the SLAM problem by integrating multiple different 3D representations or existing SLAM components; however, the most interesting advances are when a new unified dense representation can be used for all aspects

*Authors contributed equally to this work.

2575-7075/24/$31.00 ©2024 IEEE
DOI 10.1109/CVPR52733.2024.01708

Authorized licensed use limited to: CAMBRIDGE UNIV. Downloaded on October 07,2024 at 08:00:04 UTC from IEEE Xplore. Restrictions apply.

2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) | 979-8-3503-5300-6/24/$31.00 ©2024 IEEE | DOI: 10.1109/CVPR52733.2024.01708

H. Matsuki, R. Murai, P. H. J. Kelly, and A. J. Davison, 'Gaussian Splatting SLAM',
in *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Seattle, WA, USA: IEEE, Jun. 2024, pp. 18039–18048. doi: 10.1109/CVPR52733.2024.01708.