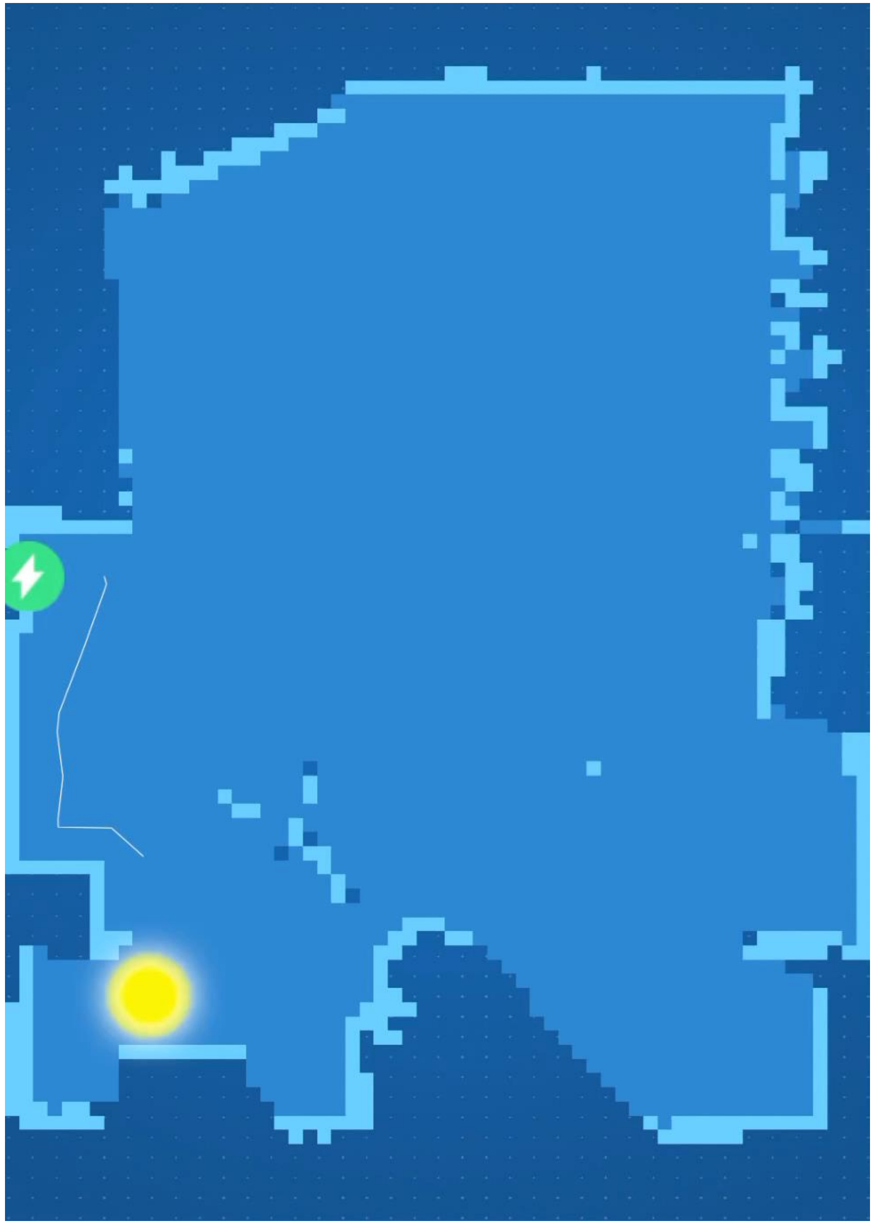
SLAM

Antonio BUEMI
System Research and Application



Agenda

life.augmented

Definition

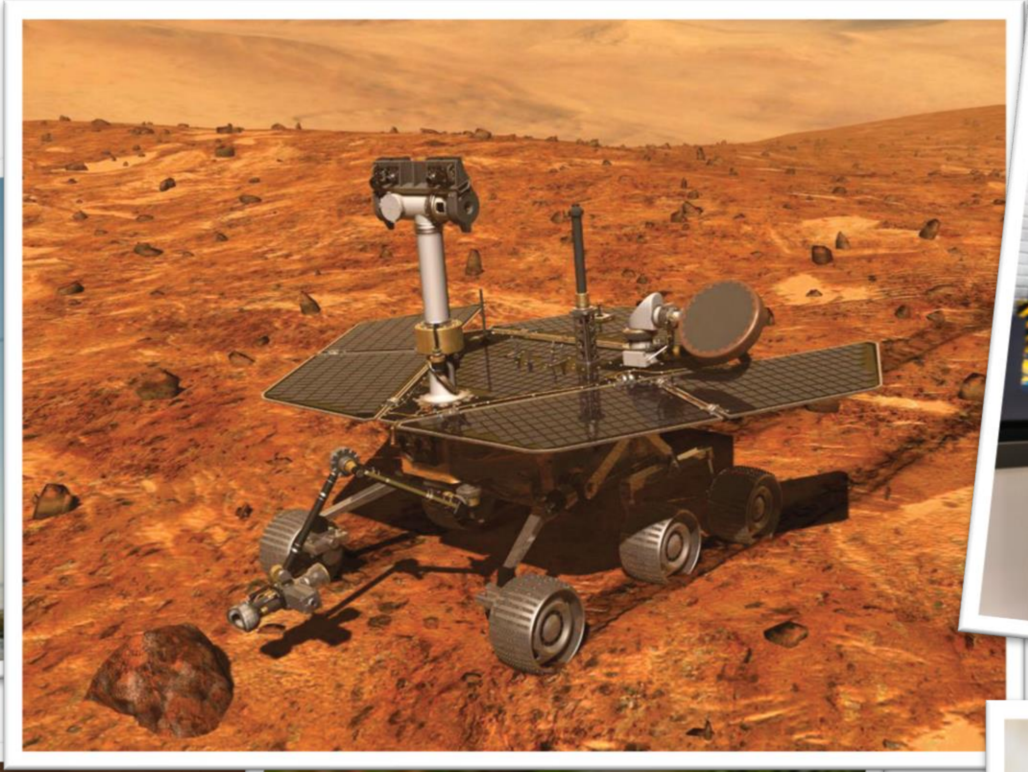
S **L** **A** **M**

Simultaneous

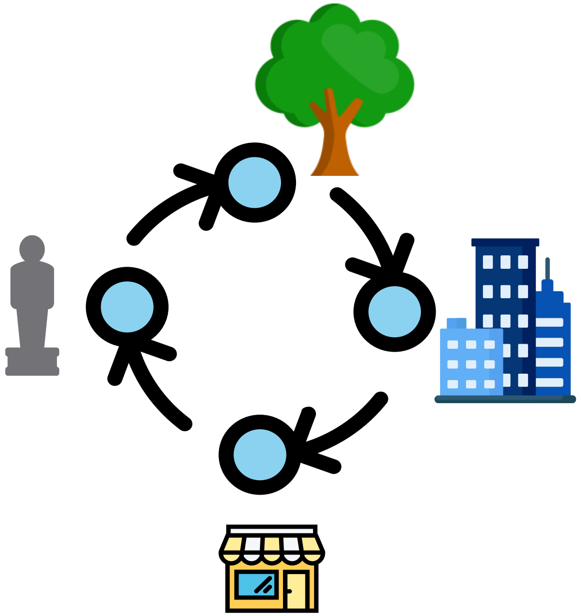
Localization
Find "your" position in an (unknown) environment

And

Mapping
Build and update the map of the environment



SLAM building blocks



Noise adds errors to the pose estimate and map.

The **Loop Detection (also called Place Recognition)** is the ability of the system to recognize the case in which one is passing in a place already visited.

The **Loop Closure** is the process of correcting the map using that information and the fact that the landmarks are supposed to be stationary

A mechanism **to adjust the pose** estimation is required.

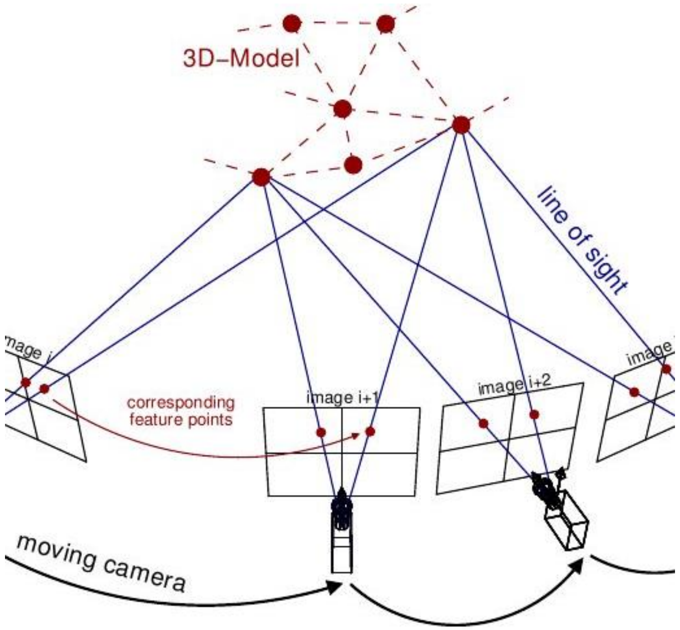
Adjusting the camera pose also means **correcting the map**, so a strategy to do this is required, too.

SLAM siblings

SfM

Struct from Motion

Reconstructs 3D struct from a set of 2D images

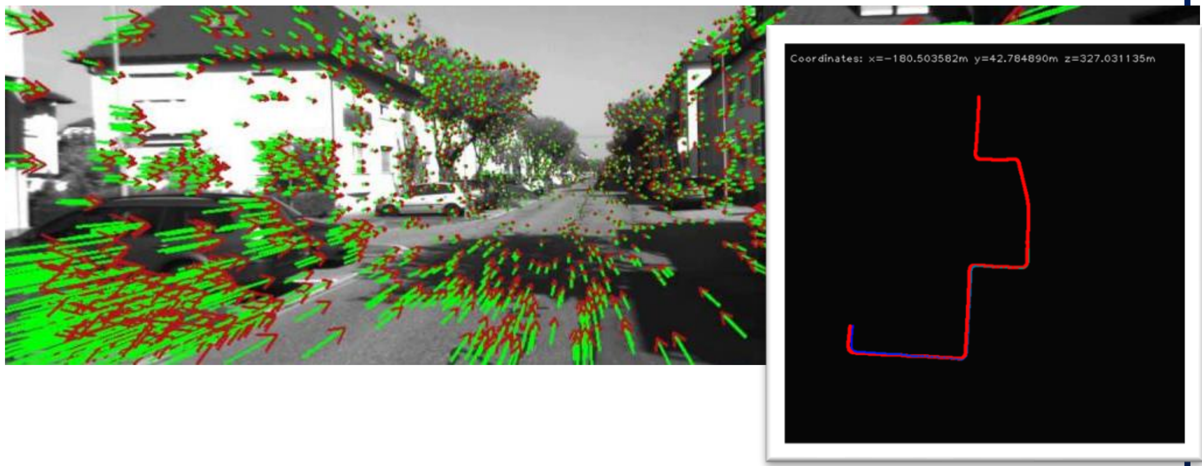


- NO odometry
- NO computational constraints
- NO real-time

VO

Visual Odometry

Estimate odometry computing differences in camera poses between two frame taken one after another

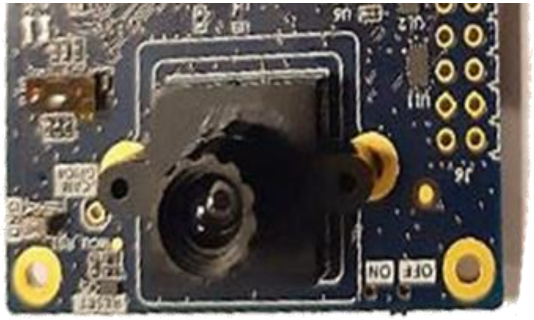


- NO Loop Closure
- NO global mapping
- Efficiency

life.augmented

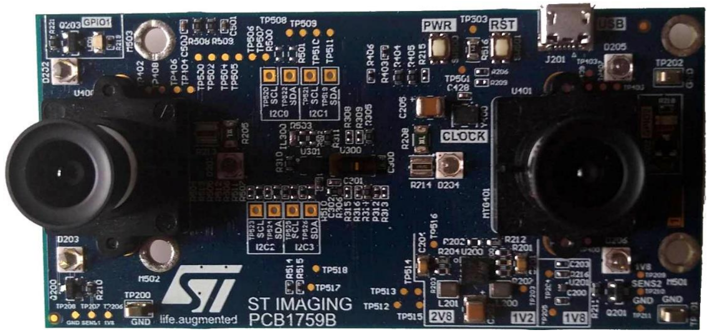
SLAM tools: how to get depth information

Monocular camera



Single image
- comparing frames taken at different camera positions

Stereo camera



Left and right images
- Correspondence between frame pairs + triangulation

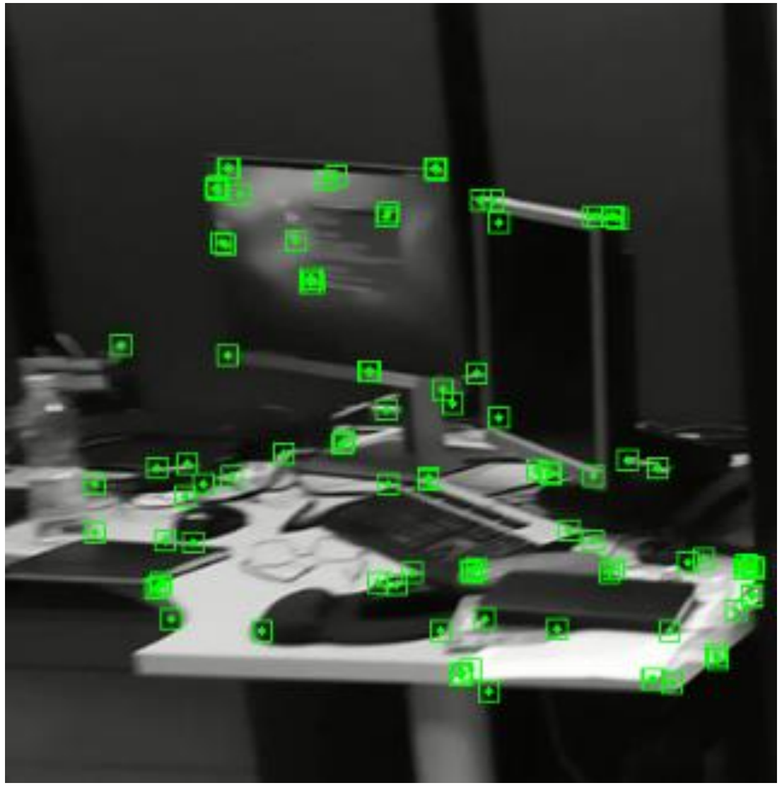
Depth camera

Depth information+RGB
- time of flight
- the pattern density of the Infra Red light emitted by the camera.

Lidar
Light **D**etection **a**nd **R**anging

Depth information
- Laser

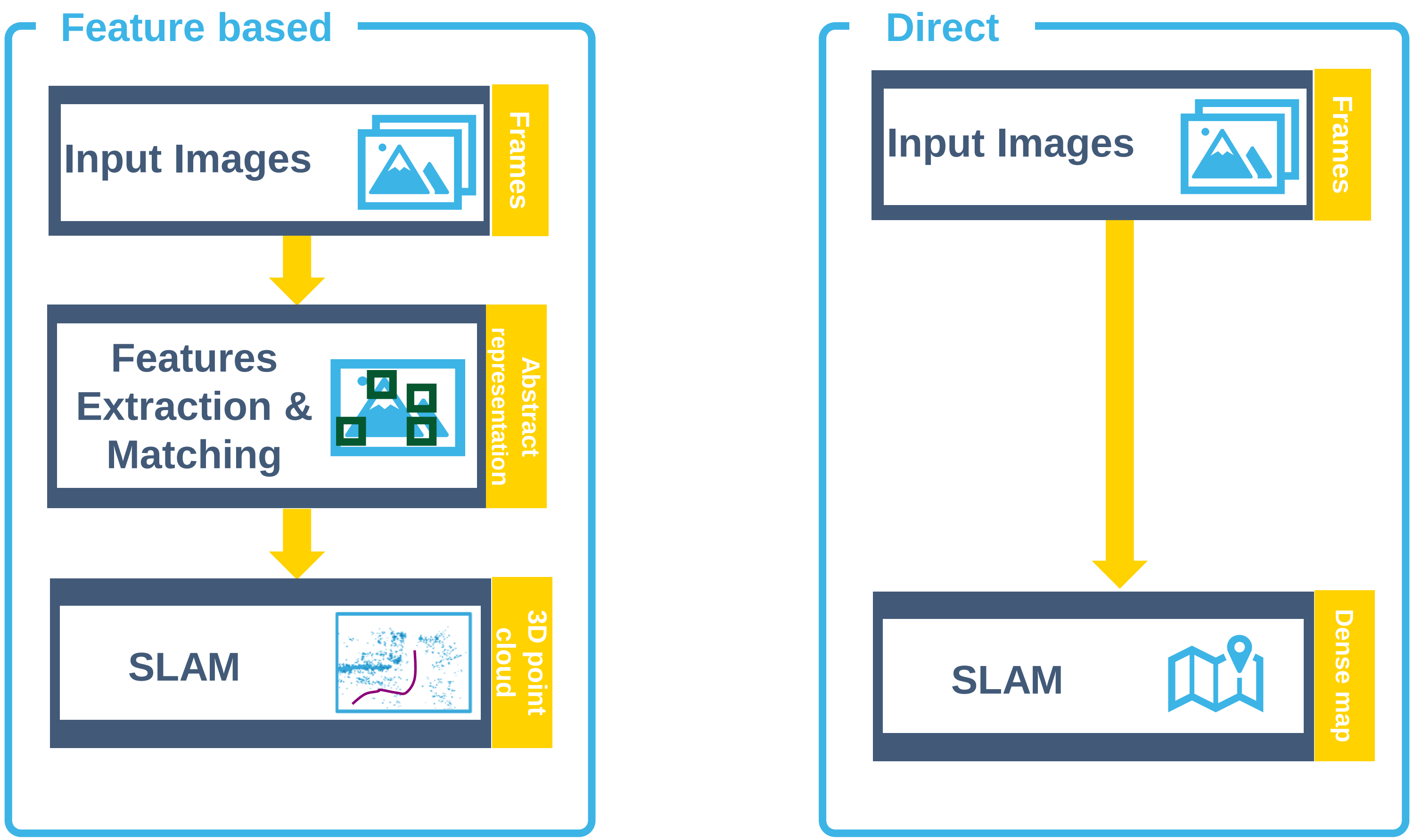
SLAM tools: descriptors



Keypoints detectors and descriptors

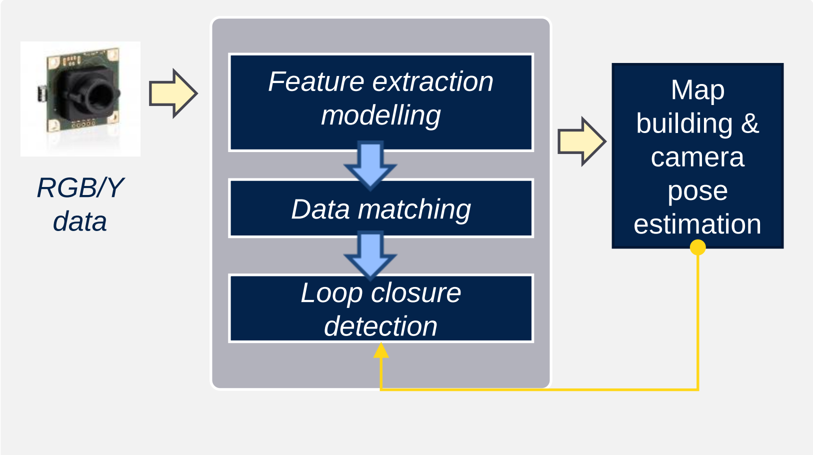
- Speeded Up Robust Features (SURF)
- Scale-invariant Feature Transform (SIFT)
- Oriented FAST and Rotated BRIEF (ORB)
- …

Feature based & Direct methods



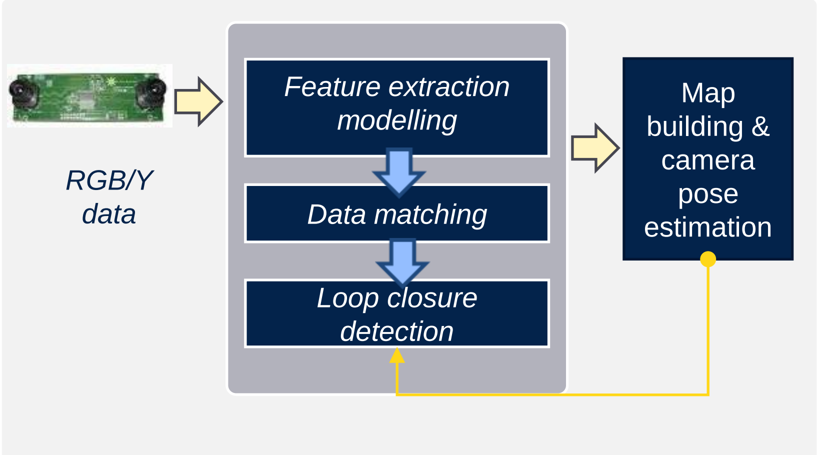
SLAM: not only visual

Monocular



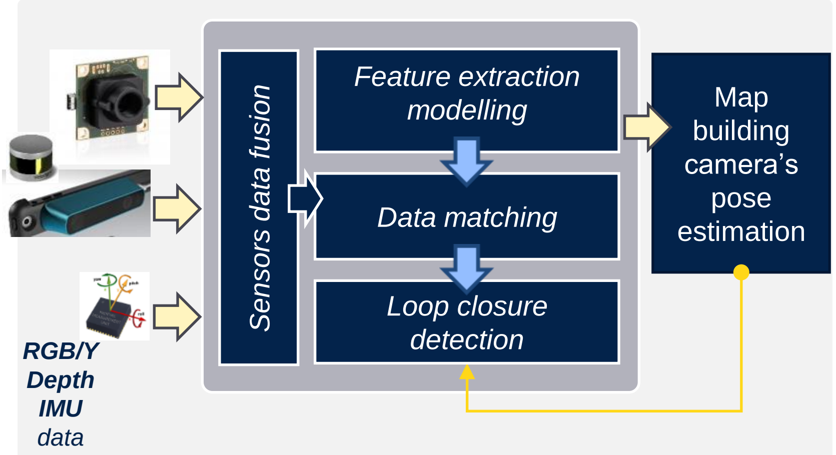
- Cheap HW
- No depth infos without camera motion
- Absolute scale information lack

Stereo



- Depth infos also with static cameras
- Computational overhead

Multimodal

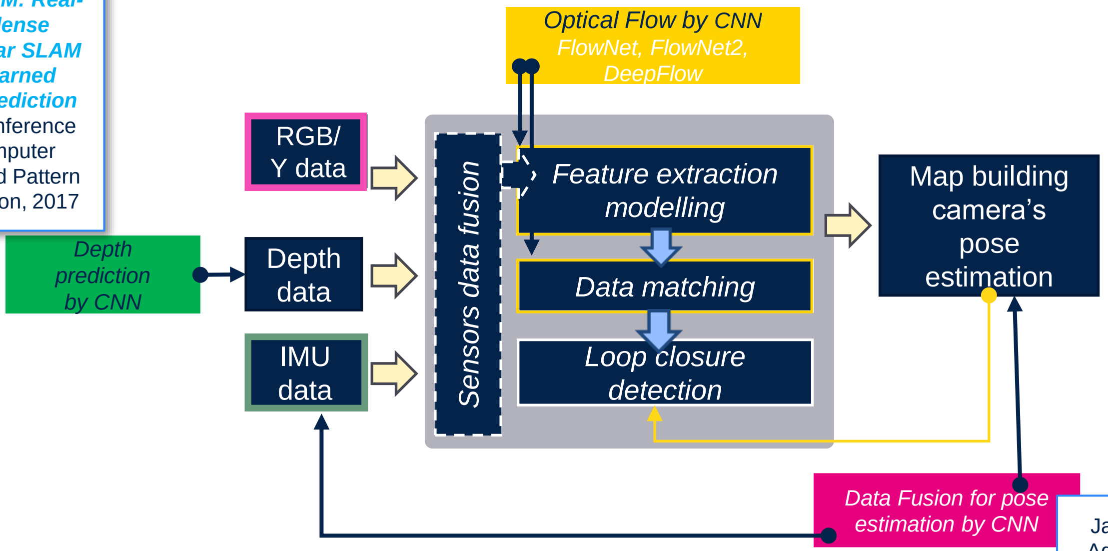


- High accuracy
- Sensors data fusion required

life.augmented

SLAM: where can we use Neural Networks?

K. Tateno, F. Tombari, I. Laina, N. Navab
CNN-SLAM: Real-time dense monocular SLAM with learned depth prediction
IEEE Conference on Computer Vision and Pattern Recognition, 2017



Patrick van der Smagt, Daniel Cremers, Thomas Brox
FlowNet: Learning Optical Flow with Convolutional Networks
IEEE International Conference on Computer Vision (ICCV) 2015

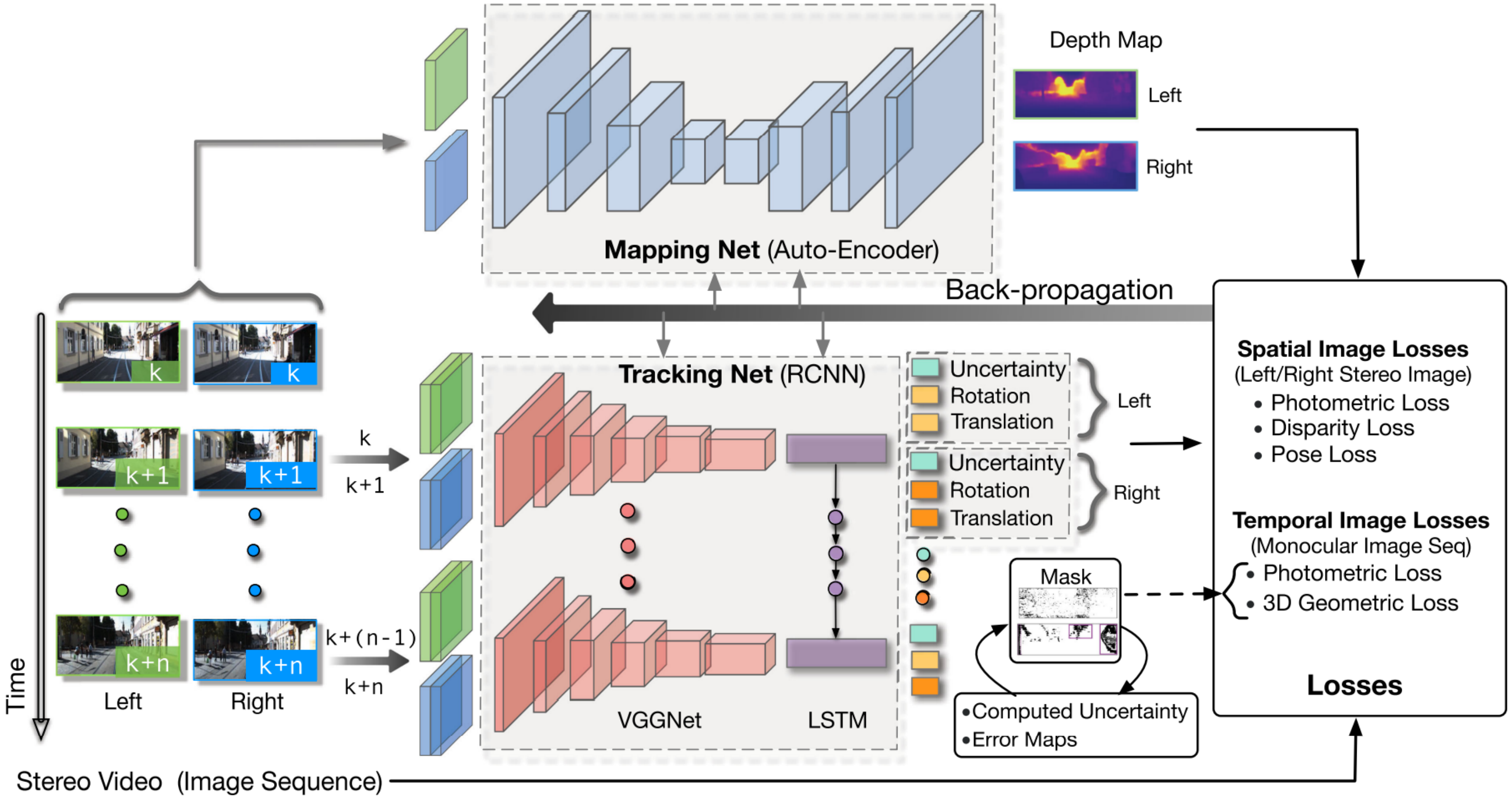
Eddy Ilg, Nikolaus Mayer, Tonmoy Saikia, Margret Keuper, Alexey Dosovitskiy, Thomas Brox
FlowNet 2.0: Evolution of Optical Flow Estimation with Deep Networks
IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2017

Philippe Weinzaepfel, Jerome Revaud, Zaid Harchaoui, Cordelia Schmid
DeepFlow: Large displacement optical flow with deep matching
IEEE Intenational Conference on Computer Vision (ICCV), 2013

Jason R. Rambach, Aditya Tewari, Alain Pagani, Didier Stricker
Learning to Fuse: A Deep Learning Approach to Visual-Inertial Camera Pose Estimation
IEEE International Symposium on Mixed and Augmented Reality (ISMAR), 2016

life.augmented

Deep SLAM: training



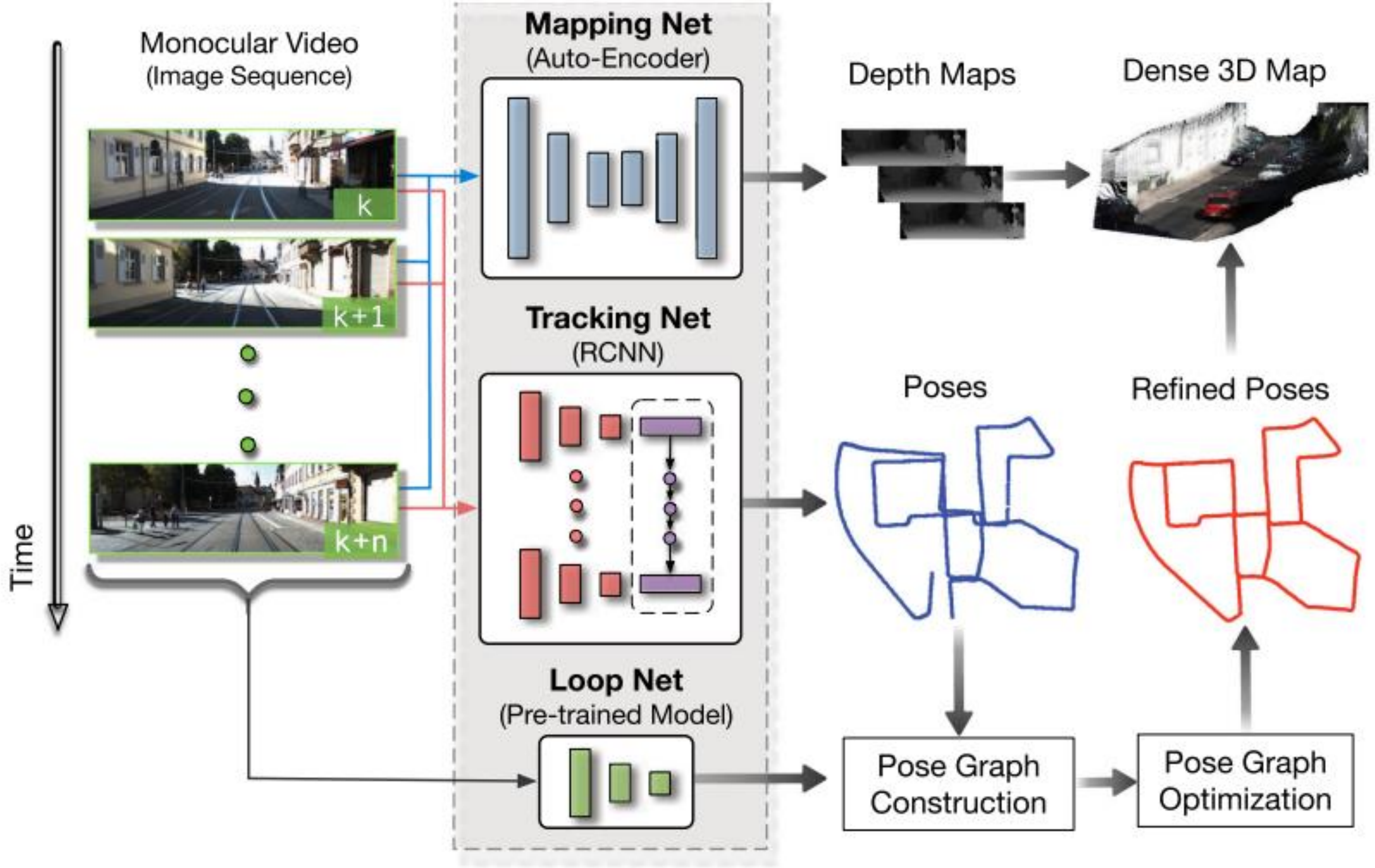
The **stereo** images used in the training phase allow to recover the scale information of the environment.

Note that this limits the generalization: the scenes used in the testing **must be similar** to ones used in training.

Ruihao Li, Sen Wang
DeepSLAM: A Robust Monocular SLAM System With Unsupervised Deep Learning
IEEE Transactions on Industrial Electronics, Vol. 68, No. 4, April 2021

life.augmented

Deep SLAM: testing



The testing framework takes **monocular** color images as input and produces depth maps, poses and point clouds as outputs by using Mapping-Net, Tracking-Net, and Loop-Net.

Ruihao Li, Sen Wang
DeepSLAM: A Robust Monocular SLAM System With Unsupervised Deep Learning
IEEE Transactions on Industrial Electronics, Vol. 68, No. 4, April 2021

ORB-SLAM

ORB-SLAM2

ORB-SLAM2 is a feature-based monocular SLAM system

real time on CPU

small & large scale

indoor & outdoor environment

loop closure

relocalization

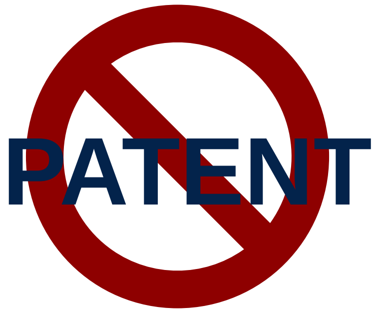
sparse 3D map

Main features

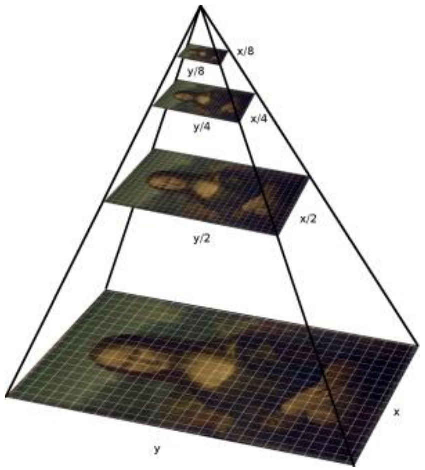
Raúl Mur-Artal and Juan D. Tardós
ORB-SLAM2: an Open-Source SLAM System for Monocular, Stereo and RGB-D Cameras
IEEE Transactions on Robotics, vol. 33, no. 5, pp. 1255-1262, 2017

ORB: oriented FAST and rotated BRIEF

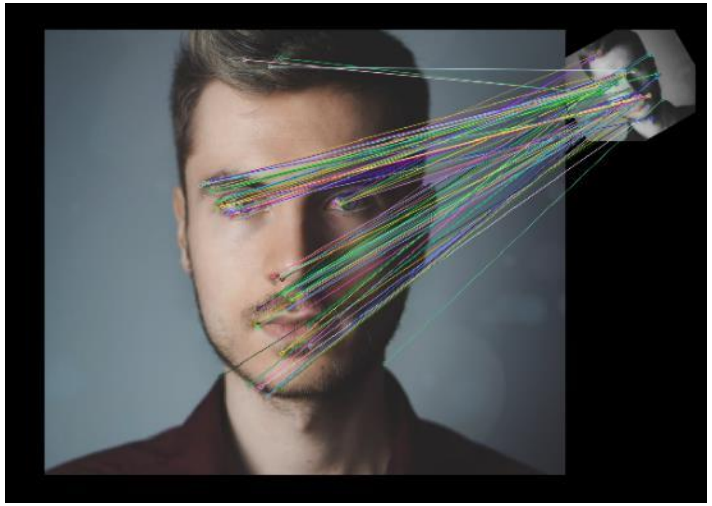
- ORB is a fusion of FAST keypoints detector and BRIEF descriptors
- It is a free alternative to SIFT and SURF and it overcomes them in computation cost and performance



free



partial scale invariant



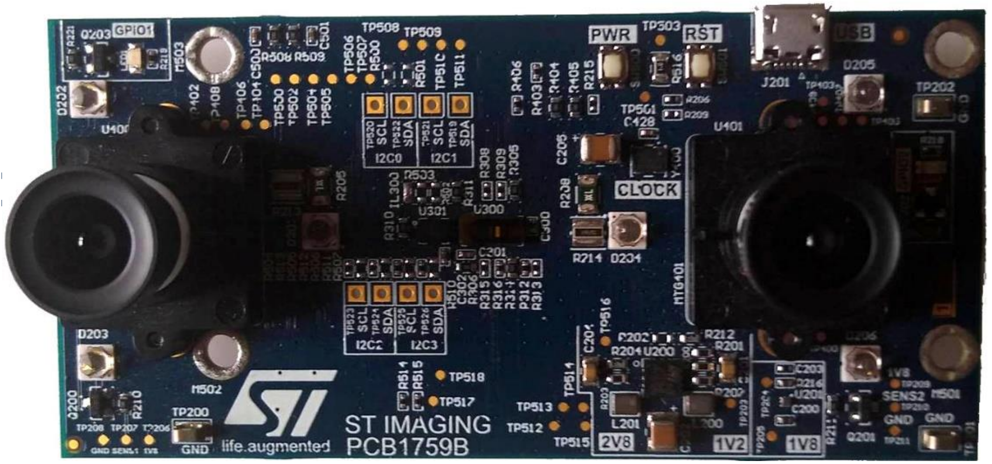
rotation invariant

STEREO-CAM board

- STEREO-CAM (mezzanine board) is a board designed in ST
- It contains:
  - No microcontroller
  - 96connector to be connected to any 96board

  - 2x VD56G3 OLGA80 (FoX NIR imaging sensor with HW Optical Flow accelerator)
  - 2x LSM6DSRXTR (3D accelerometer, 3D gyroscope)
  - 1x Ewok MZ Module V2

- Two 96boards are used (for drivers availability):
  - DragonBoard 410C
    - Qualcomm Snapdragon, 4 cores @ 1.4GHz
  - Avenger96 (STM32MP157)

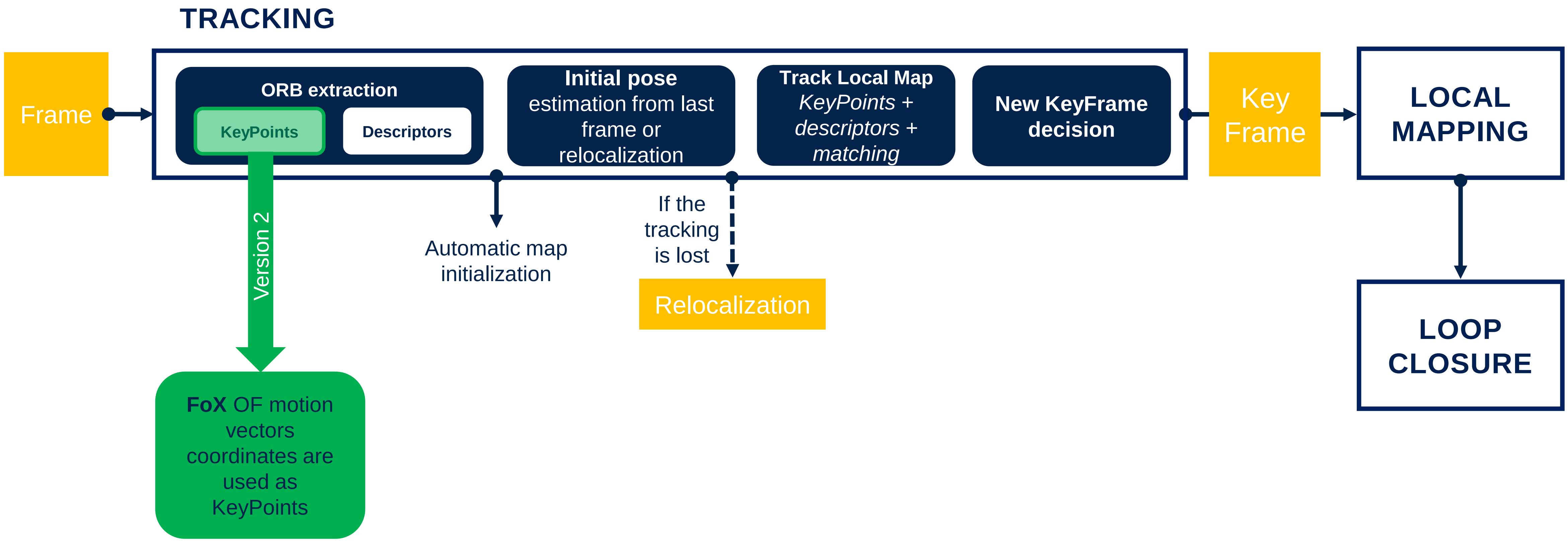


life.augmented

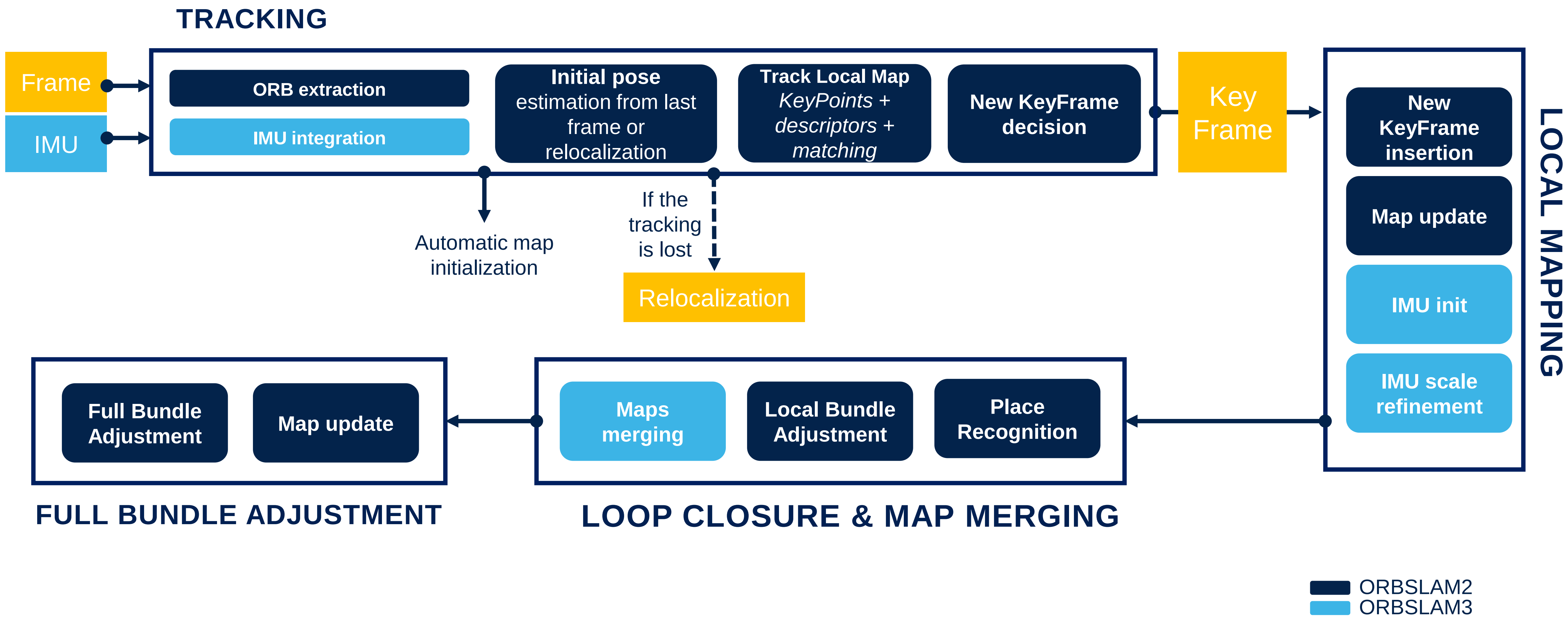
From Version 1 to Version 2

Version 1 ORB-SLAM2 uses Oriented FAST+Rotated BRIEF.

Version 2 exploit the Optical Flow (Keypoints+MVs) retrieved by FoX

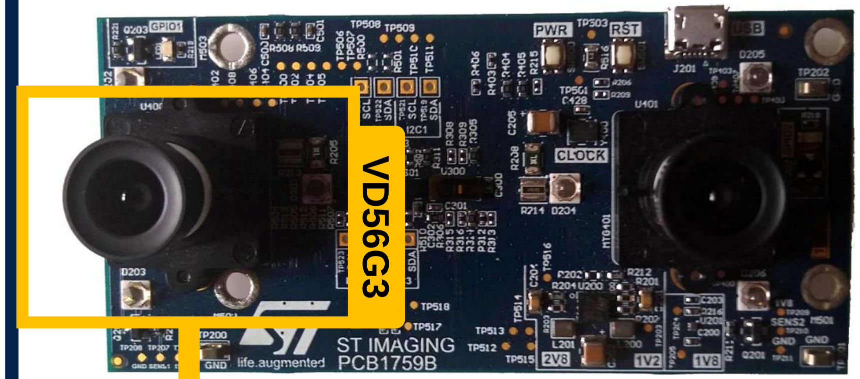


ORBSLAM2 and ORBSLAM3



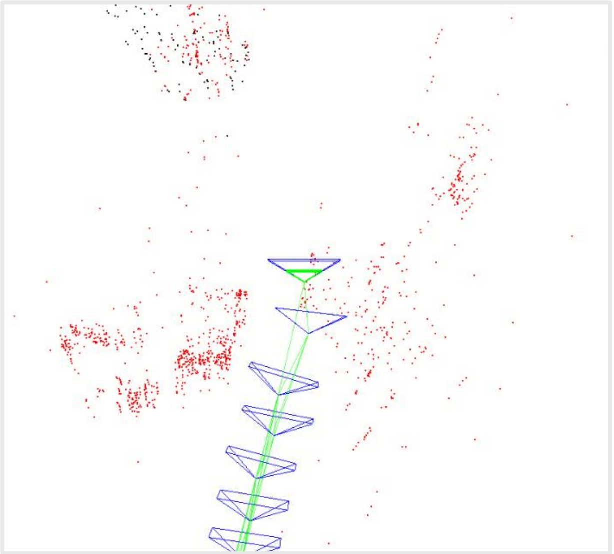
ORB-SLAM2 (standard) IN/OUT

STEREO-CAM



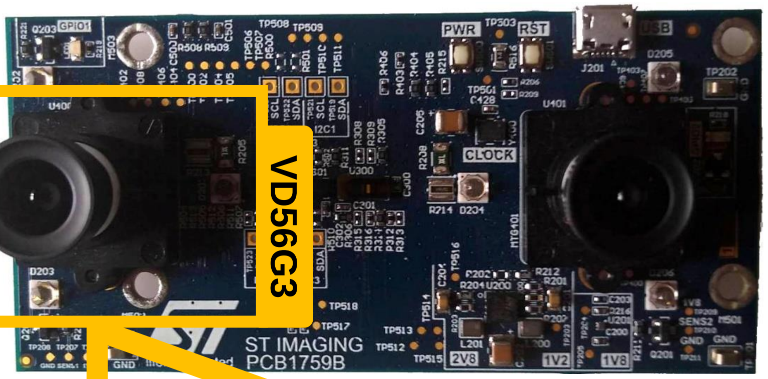
Near-InfraRed (NIR) sequence

ORB-SLAM



ORB-SLAM2 (OF-based) IN/OUT

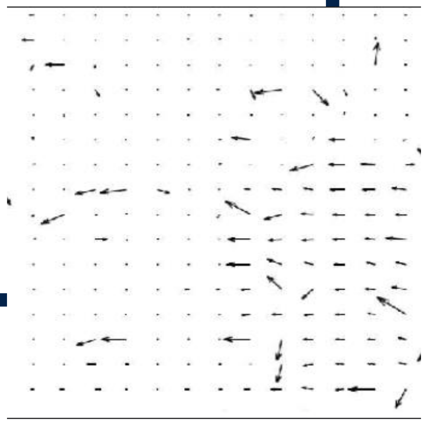
STEREO-CAM



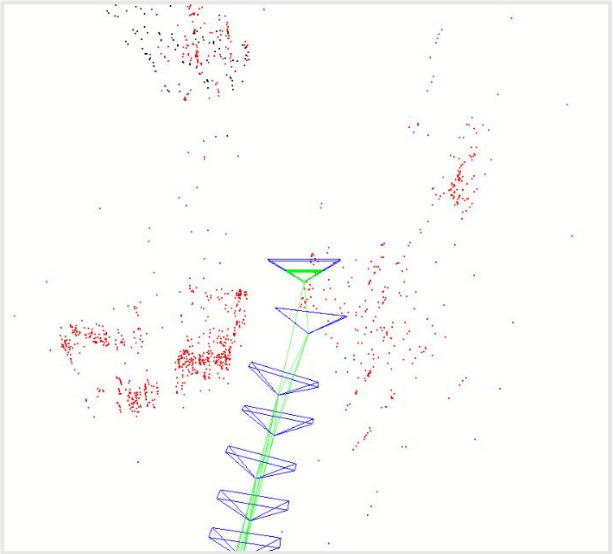
VD56G3

Near-InfraRed (NIR) sequence

Motion Vectors (MVs)



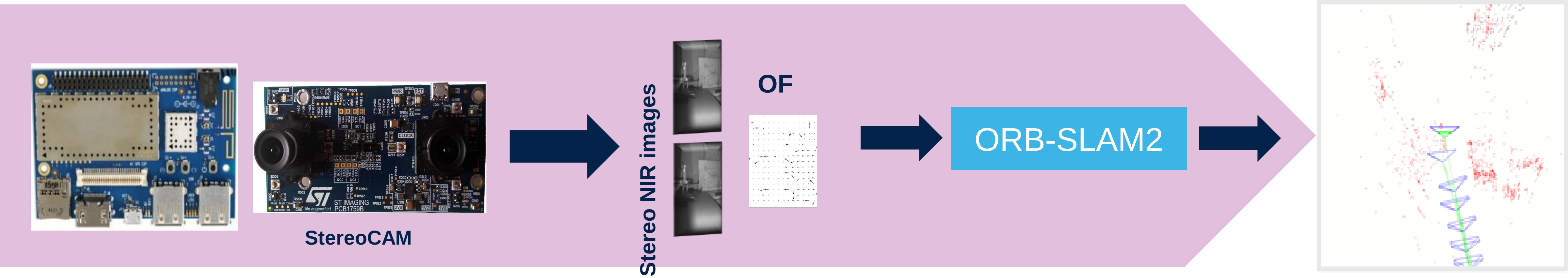
ORB-SLAM



life.augmented

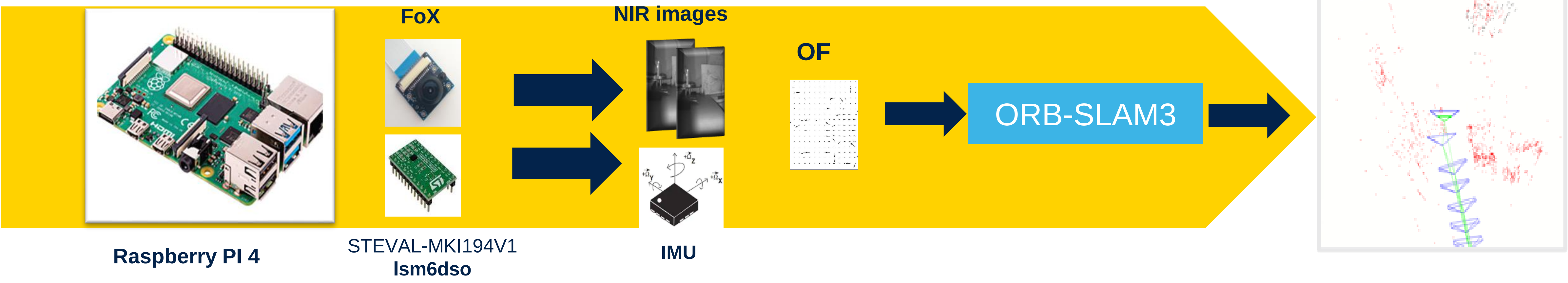
Platforms 1/2

ORB-SLAM2+DragonBoard



Platforms 2/2

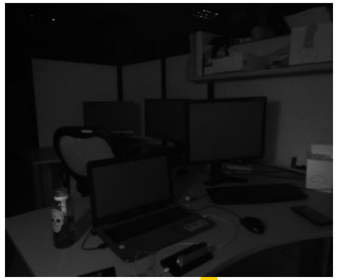
ORB-SLAM3+Raspberry



Results

iToF

image



depth map

resolution
402x336

StereoCAM+DragonBoard 410C

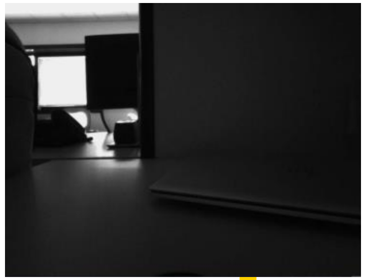
ORB-SLAM2

3D points cloud+trajectory

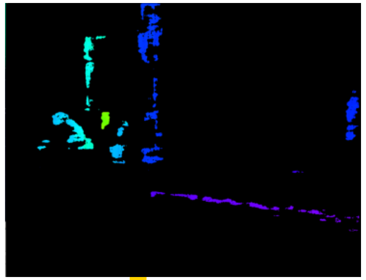
6.55 fps

Eys3D

image



depth map



resolution
1104x848
↓2
552x424

StereoCAM+DragonBoard 410C

ORB-SLAM2

3D points cloud+trajectory

3.78 fps

life.augmented

Conclusion

life.augmented

Is SLAM solved?

Short answer: *It depends on the application*

Context definition aspects

Robots

Motion limits
e.g. max speed, dynamics

Sensors features
e.g. resolution, sampling rate

Computational resources
e.g. CPU features, RAM

Environment

2D or 3D map?

Presence of landmarks
(natural or artificial)

Amount of dynamic elements

Performances

Target accuracy

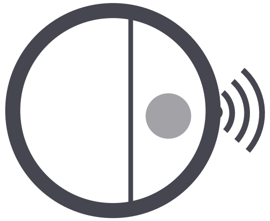
Success rate

Estimation latency

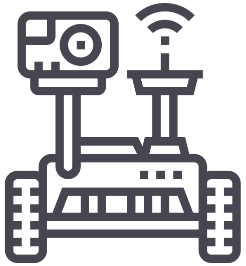
Maximum size of the map

Maximum operation time

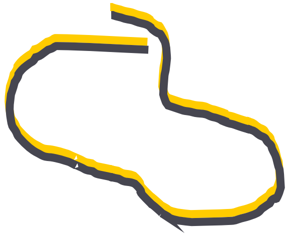
Examples of solved cases



2D mapping of indoor environment with robot equipped with wheels and laser scanner



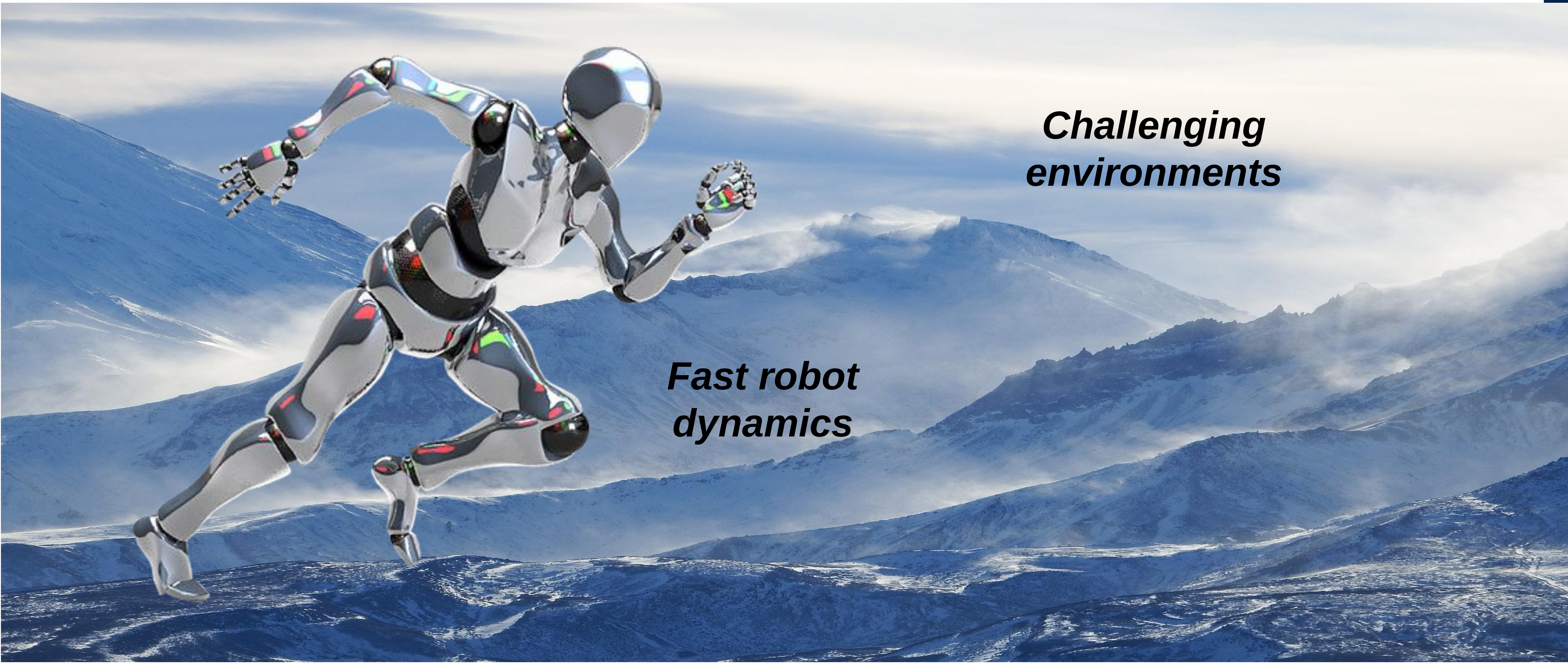
Vision-based SLAM using slowly moving robots



Visual-Inertial Odometry

life.augmented

Some open problem



Future SLAM key requirements

Robust performances

Reliablility
Low failure rate for an extendend time in a broad range of environments

Automomy
Fail-safe mechanism
Self-tuning capabilities

Flexibility
System parameters adaptivity to the scenario

High-level understanding

Geometry
beyond basic geometry reconstruction to understand the environment

Semantic
Additional Artificial Intelligence modules

Resource Awareness

Smart system
Monitoring of the available sensing and computational resource to adjust accordingly the computational load

Task Driven Perception

Selectivity
Filter irrelevant sensor data, in order to support the task

Scalability
Adaptive map representation whose complexity may vary depending on the task

Thank you

life.augmented