



Računarski fakultet

Review on the topic of

Simultaneous localization and mapping (SLAM)

Course: Robotics

Advisor:

prof. Miloš Jovanović

Student:

Vanja Kovinić

Belgrade, 2025.

Contents

| | | |
|----------|---|----------|
| 1 | Introduction to SLAM | 2 |
| 1.1 | Definition and Problem Statement | 2 |
| 1.2 | Historical Context and Importance in Robotics | 4 |

1 Introduction to SLAM

1.1 Definition and Problem Statement

Simultaneous Localization and Mapping (SLAM) represents one of the fundamental challenges in robotics. At its core, SLAM addresses a deceptively simple question:

How can a mobile robot build a map of an unknown environment while simultaneously determining its own position within that map?

This seemingly straightforward task embodies a classic "chicken-and-egg" problem - to create an accurate map, the robot needs to know its precise location, but to determine its location, it needs an accurate map.

The SLAM problem can be formally defined as the process of constructing or updating a map of an unknown environment while keeping track of an agent's location within it. Mathematically, SLAM estimates both the trajectory of the robot ($X_1 = x_1, x_2, \dots, x_t$) and the map of the environment (M) given the observations (Z_1) and control inputs (U_1):

$$p(X_1, M \mid Z_1, U_1)$$

This joint probability distribution encapsulates the inherent uncertainty in both the robot's position and the environmental map, recognizing that both must be estimated simultaneously.

The technical challenges of SLAM extend beyond this core problem. Real-world implementations must contend with:

1. **Sensor limitations:** All sensors provide imperfect measurements with noise, limited range, and potential occlusions.
2. **Data association:** Determining whether sensor observations correspond to previously observed landmarks or represent new features.
3. **Loop closure:** Recognizing when the robot has returned to a previously visited location, allowing for correction of accumulated errors.
4. **Computational efficiency:** Balancing accuracy with real-time performance requirements, especially on platforms with limited computing resources.
5. **Environmental dynamics:** Managing changes in the environment, such as moving objects or changing conditions.

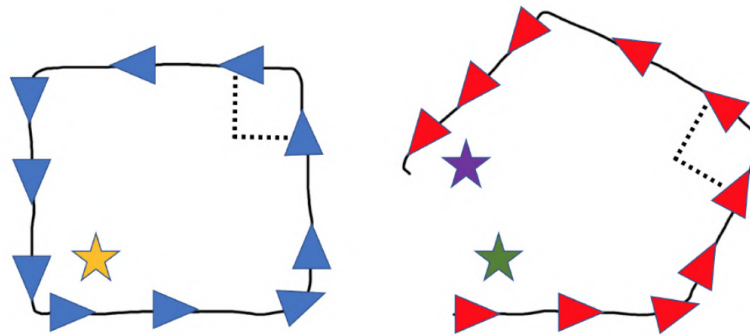


Figure 1: The diagram shows loop closure in robot mapping. When a feature is recognized again (left), a constraint is added to correct the robot's trajectory. If the feature is not recognized (right), errors in the estimated path remain uncorrected. [1]

A successful SLAM system must integrate sensor data over time while accounting for uncertainties and constraints to produce a consistent representation of both the robot's path and the surrounding environment.

1.2 Historical Context and Importance in Robotics

The development of SLAM techniques represents a crucial milestone in the evolution of autonomous robotics. Before SLAM, robots relied on pre-built maps or simple reactive behaviors, severely limiting their autonomy and usefulness in unknown environments. The conceptual foundations of SLAM emerged in the 1980s through seminal work by researchers such as Hugh Durrant-Whyte and John J. Leonard [2], who recognized the statistical correlations between landmarks in a map. However, the term "SLAM" itself was not coined until the early 1990s.

The historical progression of SLAM research can be divided into several key phases:

1. **Early probabilistic approaches (1990s):** The initial formulation of SLAM using Extended Kalman Filters (EKF) provided the mathematical framework for handling uncertainty in both robot motion and sensing [3]. These approaches were limited by computational complexity that scaled quadratically with the number of landmarks.
2. **Particle filter methods (early 2000s):** FastSLAM [4] and other particle filter approaches offered improved performance for nonlinear systems and non-Gaussian noise, enabling more robust implementations.
3. **Graph-based optimization (mid-2000s):** The formulation of SLAM as a sparse graph optimization problem led to more efficient solutions that could handle larger environments and longer trajectories.
4. **Visual SLAM systems (2007-2015):** The incorporation of camera data enabled systems like MonoSLAM [5], PTAM, and ORB-SLAM [6], which could operate using only visual information without specialized ranging sensors.

5. **Learning-based approaches (2015-present):** The integration of deep learning methods has pushed the boundaries of SLAM performance, particularly in feature extraction, depth estimation, and semantic understanding.

The introduction of SLAM algorithms effectively gave robots a spatial intelligence - the ability to interpret, map, and navigate previously unseen environments without human guidance. Some of the most notable applications of SLAM include:

- **Autonomous vehicles:** Tesla's FSD system and Waymo's autonomous taxis implement sophisticated SLAM variants that fuse visual data with other sensor modalities. Their ability to navigate complex urban environments relies not just on GPS (which fails in tunnels or urban canyons) but on real-time environmental mapping at centimeter-level precision.
- **Disaster response:** After the Fukushima nuclear disaster, robots equipped with RGBD-SLAM systems entered radiation-contaminated zones inaccessible to humans, creating 3D maps that enabled remote assessment of structural damage without endangering human lives [7].
- **Domestic robotics:** The evolution of cleaning robots from random bouncers like early Roombas to the systematic path-planners seen in Roborock S7 models illustrates how SLAM transformed consumer robotics from novelties into genuinely useful tools that can efficiently cover complex home layouts.
- **Mixed reality:** Apple's ARKit and Microsoft's HoloLens use visual-inertial SLAM to maintain stable positioning of virtual objects in the physical world, enabling applications from surgical training to architectural visualization that rely on precise spatial registration.

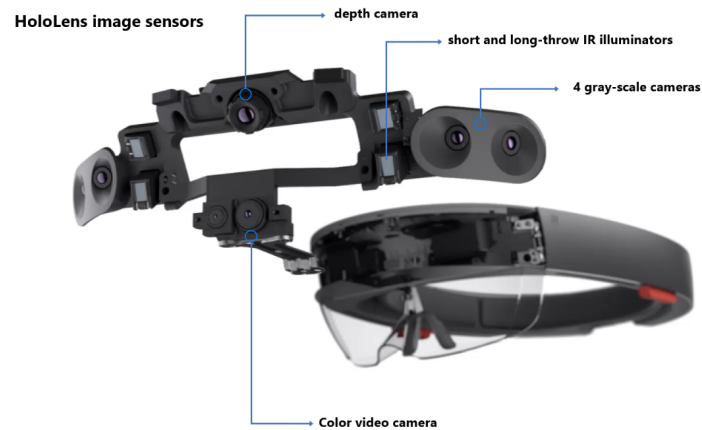


Figure 2: Microsoft HoloLens uses SLAM by fusing data from cameras, depth sensors, and IMUs to simultaneously track the headset’s position and map the environment. This enables accurate placement of virtual objects in the real world for stable and immersive augmented reality experiences.

References

- [1] Samer B. Nashed. “A Brief Survey of Loop Closure Detection: A Case for Rethinking Evaluation of Intelligent Systems”. In: (2020). URL: https://ml-retrospectives.github.io/neurips2020/camera_ready/21.pdf.
- [2] J.J. Leonard and H.F. Durrant-Whyte. “Simultaneous map building and localization for an autonomous mobile robot”. In: (1991), 1442–1447 vol.3. DOI: 10.1109/IROS.1991.174711.
- [3] Randall Smith, Matthew Self, and Peter Cheeseman. “Estimating Uncertain Spatial Relationships in Robotics”. In: (2013). arXiv: 1304.3111 [cs.AI]. URL: <https://arxiv.org/abs/1304.3111>.
- [4] Michael Montemerlo et al. “FastSLAM: A Factored Solution to the Simultaneous Localization and Mapping Problem”. In: (2002). URL: <https://cdn.aaai.org/AAAI/2002/AAAI02-089.pdf>.

- [5] Andrew J. Davison et al. “MonoSLAM: Real-Time Single Camera SLAM”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 29.6 (2007), pp. 1052–1067. DOI: 10.1109/TPAMI.2007.1049.
- [6] Raul Mur-Artal, J. M. M. Montiel, and Juan D. Tardos. “ORB-SLAM: A Versatile and Accurate Monocular SLAM System”. In: *IEEE Transactions on Robotics* 31.5 (Oct. 2015), pp. 1147–1163. ISSN: 1941-0468. DOI: 10.1109/tro.2015.2463671. URL: <http://dx.doi.org/10.1109/TR0.2015.2463671>.
- [7] Keiji Nagatani et al. “Emergency response to the nuclear accident at the Fukushima Daiichi Nuclear Power Plants using mobile rescue robots”. In: *Journal of Field Robotics* 30 (Feb. 2013), pp. 44–63. DOI: 10.1002/rob.21439.