SLAM Definition

SLAM stands for Simultaneous Localization and Mapping. It is a technique that allows robots to build a map of their environment while simultaneously determining their own position within that map. SLAM is used in a variety of applications, including autonomous vehicles, drones, and robots.

There are two main types of SLAM:

Odometry-based SLAM: This type of SLAM uses the robot's odometry to estimate its position and orientation. Odometry is the measurement of the robot's movement based on its wheel encoders or inertial measurement unit (IMU).

Feature-based SLAM: This type of SLAM uses features in the environment to build a map. Features can be anything that is unique and can be easily identified, such as corners, edges, or blobs.

SLAM is a challenging problem, but it is essential for robots to be able to operate in unknown environments. There are a number of challenges that SLAM algorithms must address, including:

Sensor noise: The sensors used by robots are not perfect, and they introduce noise into the measurements. This noise can make it difficult to accurately estimate the robot's position and map the environment.

Motion blur: When the robot is moving quickly, the images captured by its sensors may be blurred. This can make it difficult to identify features in the environment.

Outliers: Sometimes, the sensors may capture measurements that are not accurate. These outliers can corrupt the map and make it inaccurate.

Despite these challenges, SLAM is a powerful technique that allows robots to operate in a wide variety of environments. As sensors and algorithms continue to improve, SLAM will become even more reliable and accurate.

How SLAM Works:

Broadly speaking, there are two types of technology components used to achieve SLAM. The first type is sensor signal processing, including the front-end processing, which is largely dependent on the sensors used. The second type is pose-graph optimization, including the back-end processing, which is sensor-agnostic.

Visual SLAM

As the name suggests, visual SLAM (or vSLAM) uses images acquired from cameras and other image sensors. Visual SLAM can use simple cameras (wide angle, fish-eye, and spherical cameras), compound eye cameras (stereo and multi cameras), and RGB-D cameras (depth and ToF cameras).

Visual SLAM can be implemented at low cost with relatively inexpensive cameras. In addition, since cameras provide a large volume of information, they can be used to detect landmarks (previously measured positions). Landmark detection can also be combined with graph-based optimization, achieving flexibility in SLAM implementation.

Monocular SLAM is when vSLAM uses a single camera as the only sensor, which makes it challenging to define depth. This can be solved by either detecting AR markers, checkerboards, or other known objects in the image for localization or by fusing the camera information with another sensor such as inertial measurement units (IMUs), which can measure physical quantities such as velocity and orientation. Technology related to vSLAM includes structure from motion (SfM), visual odometry, and bundle adjustment.

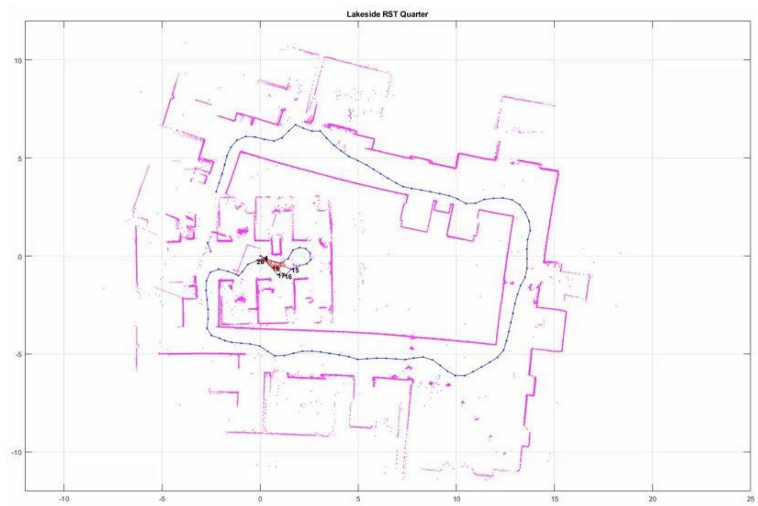
Visual SLAM algorithms can be broadly classified into two categories. Sparse methods match feature points of images and use algorithms such as PTAM and ORB-SLAM. Dense methods use the overall brightness of images and use algorithms such as DTAM, LSD-SLAM, DSO, and SVO.

Common Challenges with SLAM

Although SLAM is used for some practical applications, several technical challenges prevent more general-purpose adoption. Each has a countermeasure that can help overcome the obstacle.

The most common challenge is localization errors accumulate, causing substantial deviation from actual values

SLAM estimates sequential movement, which include some margin of error. The error accumulates over time, causing substantial deviation from actual values. It can also cause map data to collapse or distort, making subsequent searches difficult. Let’s take an example of driving around a square-shaped passage. As the error accumulates, robot’s starting and ending point no longer match up. This is called a loop closure problem. Pose estimation errors like these are unavoidable. It is important to detect loop closures and determine how to correct or cancel out the accumulated error.



In robotics, EKF SLAM is a class of algorithms which uses the extended Kalman filter (EKF) for SLAM. Typically, EKF SLAM algorithms are feature based, and use the maximum likelihood algorithm for data association. In the 1990s and 2000s, EKF SLAM had been the de facto method for SLAM, until the introduction of FastSLAM.[25]

Associated with the EKF is the gaussian noise assumption, which significantly impairs EKF SLAM's ability to deal with uncertainty. With greater amount of uncertainty in the posterior, the linearization in the EKF fails.[26]

GraphSLAM

In robotics, GraphSLAM is a SLAM algorithm which uses sparse information matrices produced by generating a factor graph of observation interdependencies (two observations are related if they contain data about the same landmark).[26]

Resources:

1.Wikipedia: https://en.wikipedia.org/wiki/Simultaneous_localization_and_mapping

2.Matlab: https://www.mathworks.com/discovery/slam.html#:~:text=SLAM%20(simultaneous%20localization%20and%20mapping)%20is%20a%20method%20used%20for,to%20map%20out%20unknown%20environments.