Robotics

Lecture 7: Simultaneous Localisation and Mapping (SLAM)

See course website
http://www.doc.ic.ac.uk/~ajd/Robotics/ for up to date information.

Andrew Davison and Stefan Leutenegger
Department of Computing
Imperial College London
- Need repeatable spin and measurement.
- Recognising orientation too will be computationally costly without invariant descriptors.
Simultaneous Localisation and Mapping

- One of the big successes of probabilistic robotics.

  A body with quantitative sensors moves through a previously unknown, static environment, mapping it and calculating its egomotion.

- When do we need SLAM?
  - When a robot must be truly autonomous (no human input).
  - When nothing is known in advance about the environment.
  - When we can’t place beacons (or even use GPS like indoors or underwater).
  - And when the robot actually needs to know where it is.

- In SLAM we build a map incrementally, and localise with respect to that map as it grows and is gradually refined.
Features for SLAM

- Most SLAM algorithms make maps of natural scene features.
- Laser/sonar: line segments, 3D planes, corners, etc.
- Vision: salient point features, lines, textured surfaces.

- Features should be distinctive; recognisable from different viewpoints (data association).
SLAM seems like a chicken and egg problem — but we can make progress if we assume the robot is the only thing that moves.

Main assumption: the world is static.

We extend probabilistic estimation (from just the robot state as in MCL) to the features of the map as well. In SLAM we store and update a joint distribution over the states of both the robot and the mapped world.

New features are gradually discovered as the robot explores so the dimension of this joint estimation problem will grow.
Simultaneous Localisation and Mapping

(a) Robot start (zero uncertainty); first measurement of feature A.
Simultaneous Localisation and Mapping

(b) Robot drives forwards (uncertainty grows).
Simultaneous Localisation and Mapping

(c) Robot first observes B and C: they inherit its uncertainty.
Simultaneous Localisation and Mapping

(d) Robot drives back towards start (uncertainty grows more)
Simultaneous Localisation and Mapping

(e) Robot re-measures A; loop closure! Uncertainty shrinks.
(f) Robot re-measures B; note that uncertainty of C also shrinks.
Simultaneous Localisation and Mapping

- First Order Uncertainty Propagation

\[
\hat{x} = \begin{pmatrix}
\hat{x}_v \\
\hat{y}_1 \\
\hat{y}_2 \\
\vdots
\end{pmatrix}, \quad P = \begin{bmatrix}
P_{xx} & P_{xy_1} & P_{xy_2} & \cdots \\
P_{y_1x} & P_{y_1y_1} & P_{y_1y_2} & \cdots \\
P_{y_2x} & P_{y_2y_1} & P_{y_2y_2} & \cdots \\
\vdots & \vdots & \vdots & \ddots
\end{bmatrix}
\]

- \( x_v \) is robot state, e.g. (\( x, y, \theta \)) in 2D; \( y_i \) is feature state, e.g. (\( X, Y \)) in 2D.

- PDF over robot and map parameters is modelled as a single multi-variate Gaussian and we can use the Extended Kalman Filter.

- PDF represented with state vector and covariance matrix.
SLAM Using Active Vision

- Stereo active vision; 3-wheel robot base.
- Automatic fixated active mapping and measurement of arbitrary scene features.
- Sparse mapping.
SLAM Using Active Stereo Vision
SLAM with Ring of Sonars

Newman, Leonard, Neira and Tardós, ICRA 2002
SLAM with a Single Camera – the Principle

• **Frontend**: keypoints are detected in successive images, and associated with a 3D point in the world.

• **Backend**: the pose of the camera(s) and 3D points that best explain these keypoint measurements are estimated.
SLAM with a Single Camera

Davison, ICCV 2003; Davison, Molton, Reid, Stasse, PAMI 2007.
Purely metric probabilistic SLAM is limited to small domains due to:

- Poor computational scaling of probabilistic filters.
- Growth in uncertainty at large distances from map origin makes representation of uncertainty inaccurate.
- Data Association (matching features) gets hard at high uncertainty.
Large Scale Localisation and Mapping

Practical modern solutions to large scale mapping follow a *metric/topological* approach which approximates full metric SLAM. They need the following elements:

- Local metric mapping to estimate trajectory and make local maps.
- Place recognition, to perform ‘loop closure’ or relocalise the robot when lost.
- Map optimisation/relaxation to optimise a map when loops are closed.
Global Topological: ‘Loop Closure Detection’

Pure Topological SLAM

- Graph-based representation.
- Segmentation of the environment into linked distinct places.
- Adapted to symbolic planning and navigation.

Figure: Topological representation
Pure Topological SLAM

- Map defined as a graph of connected locations.
- Edges model relationships between locations (e.g. traversability, similarity).
Indoor Topological Map
Adding Metric Information to the Graph Edges

- The edges between linked nodes are annotated with relative motion information; could be from local mapping or purely incremental information like odometry or visual odometry.
- Apply **pose graph optimisation (relaxation)** algorithm, which computes the set of node positions which is maximally probable given both the metric and topological constraints.
- Pose graph optimisation only has an effect when there are loops in the graph.

[Diagram showing loop-closure detection and pose graph optimisation]
The position and orientation of node $j$ is obtained as the mean of the positions obtained from nodes $i$ and $k$ (i.e., by composing their positions with the corresponding relative displacements to node $j$).

The size of the nodes is proportional to uncertainty.
Map Relaxation: Good Odometry, One Loop Closure
Simple Large-Scale SLAM: RATSLAM

http://www.youtube.com/watch?v=-0XSUi69Yvs

• Very simple ‘visual odometry’ gives rough trajectory.
• Simple visual place recognition provides many loop closures.
• Map relaxation/optimisation to build global map.
ORB-SLAM: Tracking and Mapping Recognizable Features.
https://www.youtube.com/watch?v=8DISRms02YQ

- Very accurate ‘visual odometry’ trajectory.
- Visual place recognition based on 2D image features with binary descriptors for very fast matching.
- Pose graph/map optimisation for global consistency.
More Information about SLAM

If you want to find out more about SLAM there is plenty of good information and open source software available online; e.g.:

- Visual/Monocular SLAM: SceneLib2, PTAM, ScaViSLAM
- Pose Graph Optimisation: g2o, Ceres
- Many others: www.openslam.org