7

Mobile robot Localization

7.1) Introduction

⇒ Mobile robot Localization is the problem of determining the pose of a robot relative to a given map of the environment.

Position estimation   Position tracking

⇒ Problem of mobile robot localization: The pose can usually not be sensed directly.

↓

{The pose has therefore to be inferred from data}

⇒ Key difficulty: Single sensor measurement is usually insufficient to determine the pose.

↳ Instead, the robot has to integrate data over time to determine its pose.

⇒ Localization assumes that an accurate map is available.

↳ The space of map representations used in todays research is huge.

=> Taxonomy of Localization Problems

Classification

=> Not every localization problem is equally hard.

=> Localization problems are characterized by the type of knowledge that is available initially and at run-time.

(a) Position tracking

-> Position tracking assumes that the initial robot pose is known.

-> Method of position tracking assumes the pose error is small.

-> The pose uncertainity is often approximated by a unimodal distribution (e.g Gaussian)

-> The position tracking problem is a local problem, since the uncertainity is local and confined to region near the robot's true pose.

(b) Global localization

-> Here the initial pose of the robot is unknown.

-> Global localization is more difficult than position tracking; in fact it subsumes the position tracking problem.

Position tracking

(c) Kidnapped robot problem

→ Variant of the global localization problem, but one that is even more difficult.

→ The Kidnapped robot problem is more difficult than the global localization problem, in that the robot might believe it knows where it is while it does not.

→ In global localization problem, robots known that it doesn't known where it is.

⇒ Second dimension that has a substantial impact on the difficults of localization is the environment.

(a) Static environment

→ Environment where only variable quantity (state) is the robot's pose.

↳ All other objects in the environments remains at the same location forever

(b) Dynamic Environment

→ Dynamic environments possess objects other than the robot whose location or configuration changes over time.

Example: People, daylight (for robot equipped with cameras), movable furniture, doors.

Scanned by CamScanner

→ A third dimension that characterizes different localization problems pertains to the fact whether or not the localization algorithm controls the motion of the robot.

ⓐ <u>Passive localization</u>

→ In passive approaches, the localization module only observes the robot operating.

↳ The robot's motion is not aimed at facilitating localization.

ⓑ <u>Active localization</u>

→ Algorithms control the robot so as to minimize the localization error

↳ and/or the costs arising from moving in poorly localized robot into a hazardous place.

⇒ Active approaches to localization typically yield better localization results than passive ones.

⇒ A fourth dimension of the localization problem is related to the number of robots involved.

ⓐ <u>Single-robot localization</u>

→ Single robot localization offers the convenience that all data is collected at a single robot platform.

(D) Multi-robot localization

⇒ At first glance, each robot could localize itself individually, hence the multi-robot localization problem can be solved through single robot localization.

→ But, if robots are able to detect each other, however, there is the opportunity to do better.

⇒ The issue of multi-robot localization raises interesting, non-trivial issues on the representation of beliefs and the nature of the communication between them.

⇒ These four dimensions capture the four most important characteristic of the mobile robot localization problem.

7.3> Markov Localization

⇒ Probabilistic localization algorithms are variants of the Bayes filter.

⇒ The straightforward application of Bayes filters to the localization problem is called Markov localization.

Algorithm Markov_Localization$(bel(x_{t-1}), u_t, z_t, m)$:

for all x_t do

$$\overline{bel}(x_t) = \int p(x_t \mid u_t, x_{t-1}, m)\, bel(x_{t-1})\, dx$$

$$bel(x_t) = \eta\, p(z_t \mid x_t, m)\, \overline{bel}(x_t)$$

endfor

return $bel(x_t)$

⇒ Just like the Bayes filter, Markov localization transforms a probabilistic belief at time $t-1$ into a belief at time t.

⇒ The initial belief, $bel(x_0)$, reflects the initial knowledge of the robot's pose.

↳ It is set differently depending on the type of localization problem:

ⓐ Position tracking

⇒ If initial pose is known, $bel(x_0)$ is initialized by a point mass distribution.

↳ Let $\overline{x_0}$ denote the known initial pose.

$$bel(x_0) = \begin{cases} 1 & \text{if } x_0 = \bar{x}_0 \\ 0 & \text{otherwise} \end{cases}$$

⇒ ~~Fo~~ In practice the initial pose is often just known in approximation

↳ The belief $bel(x_0)$ is then usually initialized by a narrow Gaussian distribution centered around $\bar{x}_0$.

$$\boxed{bel(x_0) = \det(2\pi\Sigma)^{-\frac{1}{2}} \exp\left\{-\frac{1}{2}(x_0-\bar{x}_0)^T \Sigma^{-1}(x_0-\bar{x}_0)\right\}}$$

$$\sim N(x_0 : \bar{x}_0, \Sigma)$$

where, Σ is the covariance of the initial pose uncertainty.

ⓑ <u>Global localization</u>

⇒ If the initial pose is unknown, $bel(x_0)$ is initialized by a uniform distribution over the space of all legal poses in the map.

$$bel(x_0) = \frac{1}{|X|}$$

where, $|X|$ stands for the volume (Lebesgue measure) of the space of all poses within the map.

ⓒ <u>Other</u>

⇒ Partial knowledge of the robot's position can usually easily be transformed into an appropriate initial distribution.

7.5 EKF Localization

- The extended Kalman filter localization algorithm, or EKF is a special case of Markov localization.
- EKF localization represent beliefs $bel(x_t)$ by their first and second moments.
  - mean μ_t
  - Covariance Σ_t
- Our EKF algorithm assumes that the map is represented by a collection of features.
- Thus at any point of time t, the robot gets to observe a vector of ranges and bearings to nearby features: $Z_t = \{Z_t^1, Z_t^2, \dots\}$
- The identity of a feature is expressed by set of correspondence variables, denoted c_t^i, one for each feature vector Z_t^i
  - Correspondence variable are assumed to be known.

7.5.2 The EKF localization Algorithm

<u>Input</u>: Gaussian estimate of the robot Pose — μ_{t-1} (mean), Σ_{t-1} (covariance)

control u_t

map m

set of features $Z_t = \{Z_t^1, Z_t^2 \dots\}$

Correspondence variable $c_t = \{c_t^1, c_t^2 \dots\}$

<u>Output</u>: μ_t, Σ_t

Algorithm EKF_localization_known_correspondences$(\mu_{t-1}, \Sigma_{t-1}, u_t, z_t, c_t, m)$

$$\bar{\mu}_t = \mu_{t-1} + \begin{pmatrix} -\frac{v_t}{\omega_t} \sin \mu_{t-1,\theta} + \frac{v_t}{\omega_t} \sin(\mu_{t-1,\theta} + \omega_t \Delta t) \\ \frac{v_t}{\omega_t} \cos \mu_{t-1,\theta} - \frac{v_t}{\omega_t} \cos(\mu_{t-1,\theta} + \omega_t \Delta t) \\ \omega_t \Delta t \end{pmatrix}$$

$$G_t = \begin{pmatrix} 1 & 0 & \frac{v_t}{\omega_t} \cos \mu_{t-1,\theta} - \frac{v_t}{\omega_t} \cos(\mu_{t-1,\theta} + \omega_t \Delta t) \\ 0 & 1 & \frac{v_t}{\omega_t} \sin \mu_{t-1,\theta} - \frac{v_t}{\omega_t} \sin(\mu_{t-1,\theta} + \omega_t \Delta t) \\ 0 & 0 & 1 \end{pmatrix}$$

$$\bar{\Sigma}_t = G_t \Sigma_{t-1} G_t^T + R_t$$

$$Q_t = \begin{pmatrix} \sigma_r & 0 & 0 \\ 0 & \sigma_\phi & 0 \\ 0 & 0 & \sigma_s \end{pmatrix}$$

for all observed features $z_t^i = (r_t^i \ \phi_t^i \ s_t^i)^T$ do

$j = c_t^i$

$$\delta = \begin{pmatrix} \delta_x \\ \delta_y \end{pmatrix} = \begin{pmatrix} m_{j,x} - \bar{\mu}_{t,x} \\ m_{j,y} - \bar{\mu}_{t,y} \end{pmatrix}$$

$$q = \delta^T \delta$$

$$\hat{z}_t^i = \begin{pmatrix} \sqrt{q} \\ \operatorname{atan2}(\delta_y, \delta_x) - \bar{\mu}_{t,\theta} \\ m_{j,s} \end{pmatrix}$$

$$H_t^i = \frac{1}{q}\begin{pmatrix} \sqrt{q}\,\delta_x & -\sqrt{q}\,\delta_y & 0 \\ \delta_y & \delta_x & -1 \\ 0 & 0 & 0 \end{pmatrix}$$

$$K_t^i = \bar{\Sigma}_t H_t^{i\,T}\left(H_t^i \bar{\Sigma}_t H_t^{i\,T} + Q_t\right)^{-1}$$

endfor

$$\mu = \bar{\mu}_t + \sum_i K_t^i (z_t^i - \hat{z}_t^i)$$

$$\Sigma = \left(I - \sum_i K_t^i H_t^i\right)\bar{\Sigma}_t$$

return μ_t, Σ_t

7.6> Estimating Correspondences

7.6.1> EKF Localization with Unknown Correspondences

⇒ The most simple of all is known as maximum likelihood correspondence, in which our first determines the most likely value of the correspondence variable, and then takes this value for granted.

⇒ To reduce the danger of asserting a false data association, there exist essentially two techniques:

① Select landmarks that are sufficiently unique and sufficiently far apart from each other that confusing them with each other is unlikely.

② Make sure that the robot's pose uncertainty remains small.

1 Algorithm EKF_localization $(\mu_{t-1}, \Sigma_{t-1}, u_t, z_t, m)$:

2 $$\bar{\mu}_t = \mu_{t-1} + \begin{pmatrix} -\frac{v_t}{\omega_t}\sin\mu_{t-1,\theta} + \frac{v_t}{\omega_t}\sin(\mu_{t-1,\theta} + \omega_t\Delta t) \\ \frac{v_t}{\omega_t}\cos\mu_{t-1,\theta} - \frac{v_t}{\omega_t}\cos(\mu_{t-1,\theta} + \omega_t\Delta t) \\ \omega_t\Delta t \end{pmatrix}$$

3 $$G_t = \begin{pmatrix} 1 & 0 & \frac{v_t}{\omega_t}\cos\mu_{t-1,\theta} - \frac{v_t}{\omega_t}\cos(\mu_{t-1,\theta} + \omega_t\Delta t) \\ 0 & 1 & \frac{v_t}{\omega_t}\sin\mu_{t-1,\theta} - \frac{v_t}{\omega_t}\sin(\mu_{t-1,\theta} + \omega_t\Delta t) \\ 0 & 0 & 1 \end{pmatrix}$$

4 $$\bar{\Sigma}_t = G_t \Sigma_{t-1} G_t^T + R_t$$

5 $$Q_t = \begin{pmatrix} \sigma_r & 0 & 0 \\ 0 & \sigma_\varphi & 0 \\ 0 & 0 & \sigma_s \end{pmatrix}$$

6 for all landmarks K in the map m do

7 $$\delta_K = \begin{pmatrix} \delta_{K,x} \\ \delta_{K,y} \end{pmatrix} = \begin{pmatrix} m_{K,x} - \bar{\mu}_{t,x} \\ m_{K,y} - \bar{\mu}_{t,y} \end{pmatrix}$$

8 $$q_K = \delta_K^T \delta_K$$

9 $$\hat{z}_t^K = \begin{pmatrix} \sqrt{q_K} \\ \operatorname{atan2}(\delta_{K,y}, \delta_{K,x}) - \bar{\mu}_{t,\theta} \\ m_{K,s} \end{pmatrix}$$

10 $$H_t^K = \frac{1}{q_K}\begin{pmatrix} \sqrt{q_K}\,\delta_{K,x} & -\sqrt{q_K}\,\delta_{K,y} & 0 \\ \delta_{K,y} & \delta_{K,x} & -1 \\ 0 & 0 & 0 \end{pmatrix}$$

11 $$\Psi_K = H_t^K \bar{\Sigma}_t [H_t^K]^T + Q_t$$

12 endfor

Scanned by CamScanner

for all observed features $z_t^i = (r_t^i \ \phi_t^i \ s_t^i)^T$ do

$$j(i) = \operatorname{argmin} (z_t^i - \hat{z}_t^k)^T \Psi_k^{-1} (z_t^i - \hat{z}_t^k)$$

$$K_t^i = \bar{\Sigma}_t [H_t^{j(i)}]^T \Psi_{j(i)}^{-1}$$

endfor

$$\mu_t = \bar{\mu}_t + \sum_i K_t^i (z_t^i - \hat{z}_t^{j(i)})$$

$$\Sigma = (I - \sum_i K_t^i H_t^{j(i)}) \bar{\Sigma}_t$$

return μ_t, Σ_t

⇒ The Standard approach is to only accept landmarks for which the Mahalanobis distance or the associated probability passes a threshold test.

7.7) Multi-Hypothesis tracking

⇒ There exist a number of extension of the basic EKF to accommodate situation where the correct data association cannot be determined with sufficient reliability.

⇒ A classical technique that overcomes the difficulties in data association is the Multi hypothesis Tracking Algorithm (MHT).

Scanned by CamScanner

⇒ The MHT, can represent a belief by multiple Gaussians, that is, the posterior is represented by a mixture.

$$bel(x_t) = \frac{1}{\sum_l \psi_{t,l}} \sum_l \psi_{t,l} \det(2\pi\Sigma_{t,l})^{-\frac{1}{2}}$$
$$\exp\left\{-\frac{1}{2}(x_t-\mu_{t,l})^T \Sigma_{t,l}^{-1} (x_t-\mu_{t,l})\right\}$$

⇒ Here l is the index of the mixture component.

↳ Each such component, or "track" in MHT slang, is itself a Gaussian with mean $\mu_{t,l}$ and covariance $\Sigma_{t,l}$.

7.8) Practical Consideration

⇒ The EKF localization algorithm and its close relative, MHT localization, are popular technique for position tracking.

⇒ There exist a large number of variations of these algorithm that enhance their efficiency and robustness.

→ Doesn't consider negative information into account.