Grid Map

Feature-based Map

-> You need to give the system the ability to identify the feature.

{You need to have some knowledge of the environment the robot will be deployed in}

Volumetric Map

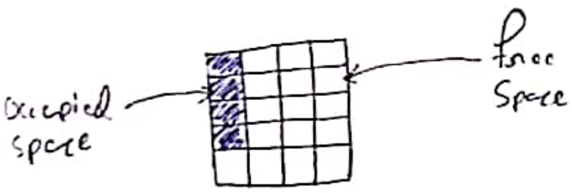
-> Take raw data to map the environment.

Fox;

* Features Map
  - -> Natural choice for Kalman filter-based SLAM System.
  - -> Compact representation
  - -> Multiple landmark observations improve the landmark position estimate (EKF)

* Grid Map
  - -> Discretize the world into cells
  - -> Grid structure is rigid.
  - -> Each cell is assumed to be occupied or free space.
  - -> Non-parametric model
  - -> Large maps require substantial memory resources.
  - -> Do not rely on a feature detector.

* Assumption 1



⇒ Each cell is a binary random variable that models the occupancy.

$P(m_i) \rightarrow 1$ {occupied}

$P(m_i) \rightarrow 0$ {free}

$P(m_i) \Rightarrow 0.5$ {No Knowledge}

★ **Assumption 2**

⇒ The World is Static.

↳ Cell that is occupied is always occupied and Cell that is free is always free.

★ **Assumption 3**

⇒ The cells (the random variables) are independent of each other.

↳ No dependency between the cell.

$$\underset{\text{map}}{P(m)} = \prod_i \underset{\text{Cell}}{P(m_i)} \quad \{\text{because of Assumption 3}\}$$

★ **Estimating a Map from data**

$$P(m \mid Z_{1:t}, x_{1:t}) = \prod_i P(m_i \mid Z_{1:t}, x_{1:t})$$

{Mapping with Known Pose}

⇒ Binary Bayes filter (for a static state)

* Static State Binary Bayes Filter

$$P(m_i|z_{1:t},x_{1:t}) = \frac{P(z_t|m_i,z_{1:t-1},x_{1:t})\,P(m_i|z_{1:t-1},x_{1:t})}{P(z_t|z_{1:t-1},x_{1:t})}$$

{Bayes rule}

$$= \frac{P(z_t|m_i,x_t)\,P(m_i|z_{1:t-1},x_{1:t-1})}{P(z_t|z_{1:t-1},x_{1:t})}$$

{Markov Assumption}

{Bayes rule}

$$\frac{P(m_i|z_t,x_t)\,.P(z_t|x_t)}{P(m_i|x_t)}$$

$$= \frac{P(m_i|z_t,x_t)\,P(z_t|x_t)\,P(m_i|z_{1:t-1},x_{1:t-1})}{P(m_i|x_t)\,P(z_t|z_{1:t-1},x_{1:t})}$$

$$\to P(m_i)$$

$$P(m_i|z_{1:t},x_{1:t}) = \frac{P(m_i|z_t,x_t)\cdot P(z_t|x_t)\,P(m_i|z_{1:t-1},x_{1:t-1})}{P(m_i)\,P(z_t|z_{1:t-1},x_{1:t})} \quad (1)$$

⇒ Do exactly the same for the opposite event:

$$P(\neg m_i|z_{1:t},x_{1:t}) = \frac{P(\neg m_i|z_t,x_t)\,P(z_t|x_t)\,P(\neg m_i|z_{1:t-1},x_{1:t-1})}{P(\neg m_i)\,P(z_t|z_{1:t-1},x_{1:t})} \quad (2)$$

Scanned by CamScanner

⇒ Computing ratio of Eq① and Eq②:

$$\frac{P(m_i \mid z_{1:t}, x_{1:t})}{P(\neg m_i \mid z_{1:t}, x_{1:t})} = \frac{P(m_i \mid z_t, x_t)\, P(m_i \mid z_{1:t-1}, x_{1:t-1})\, P(\neg m_i)}{P(\neg m_i \mid z_t, x_t)\, P(\neg m_i \mid z_{1:t-1}, x_{1:t-1})\, P(m_i)}$$

$$= \underbrace{\frac{P(m_i \mid z_t, x_t)}{1 - P(m_i \mid z_t, x_t)}}_{\text{uses } z_t} \cdot \underbrace{\frac{P(m_i \mid z_{1:t-1}, x_{1:t-1})}{1 - P(m_i \mid z_{1:t-1}, x_{1:t-1})}}_{\text{recursive term}} \cdot \underbrace{\frac{1 - P(m_i)}{P(m_i)}}_{\text{Prior}}$$

$$\Rightarrow P(m_i \mid z_{1:t}, x_{1:t}) =$$

$$\left[1 + \frac{1 - P(m_i \mid z_t, x_t)}{P(m_i \mid z_t, x_t)} \cdot \frac{1 - P(m_i \mid z_{1:t-1}, x_{1:t-1})}{P(m_i \mid z_{1:t-1}, x_{1:t-1})} \cdot \frac{P(m_i)}{1 - P(m_i)}\right]^{-1}$$

⇒ For reason of efficiency, one performs the calculations in the log odds notation.

$$\boxed{l(m_i \mid z_{1:t}, x_{1:t}) = l(m_i \mid z_t, x_t) + l(m_i \mid z_{1:t-1}, x_{1:t-1}) - l(m_i)}$$

{ where $\boxed{l(x) = \log \frac{P(x)}{1 - P(x)}}$ }

$$\boxed{P(x) = 1 - \frac{1}{1 + e^{l(x)}}}$$

{Inverse Sensor model}

{recursive term}

⇒ Or in short,

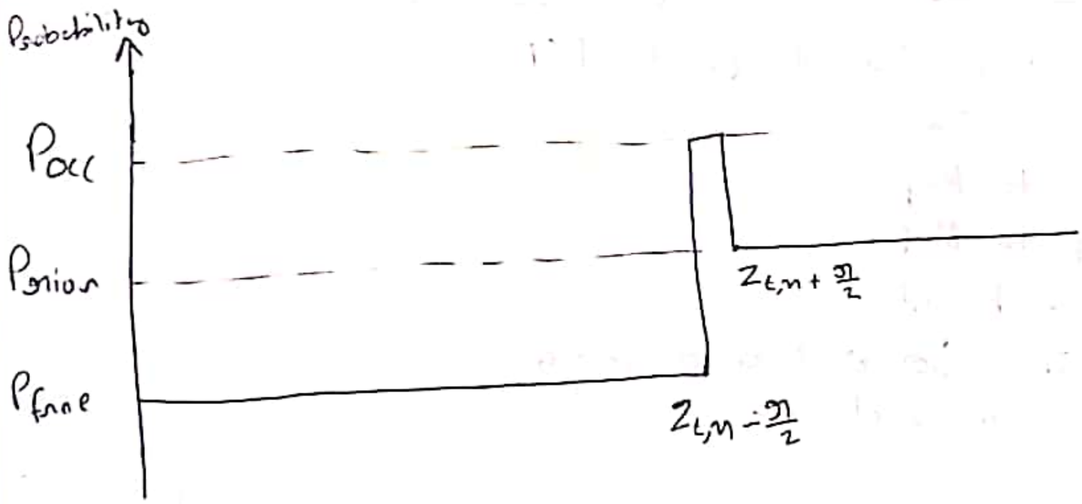
$$\boxed{l_{t,i} = \text{inv\_sensor\_model}(m_i, x_t, z_t) + l_{t-1,i} - l_0}$$

<u>Occupancy_grid_mapping</u>$(\{l_{t-1,i}\}, x_t, z_t)$:

```
for all cells m_i do
    if m_i in perceptual field of z_t then
        l_t,i = l_t-1,i + inv_sensor_model(m_i, x_t, z_t) - l_0
    else
        l_t,i = l_t-1,i
    endif
endfor
return {l_t,i}
```

⇒ Moravec and Elfes proposed occupancy grid mapping in the mid 80'ies.

* <u>Inverse Sensor model for laser range finder</u>



* Incremental Scan Alignment

⇒ Motion is noisy, we cannot ignore it.

⇒ Often, the Sensor is rather precise.

⇒ Scan-matching tries to incrementally align two scans or a map to a scan without revising the past/map.

* Pose Correction using Scan matching

⇒ Maximize the likelihood of the current pose relative to the previous pose and map.

$$\boxed{x_t^* = \underset{x_t}{\operatorname{argmax}} \left\{ \underbrace{p(z_t \mid x_t, m_{t-1})}_{\text{Current measurement}} \; \underbrace{p(x_t \mid u_{t-1}, x_{t-1}^*)}_{\text{robot motion}} \right\}}$$

Current measurement

robot motion

map Constructed so far

* Various different ways to realize Scan-matching

- Iterative Closest point (ICP)
- Scan to Scan
- Scan to Map
- Map to Map
- Feature-based
- RANSAC for outlier rejection
- Correlative matching.

⇒ Often Scan-matching is not sufficient to build a (large) consistent map.