MCE 412- Autonomous Robotics

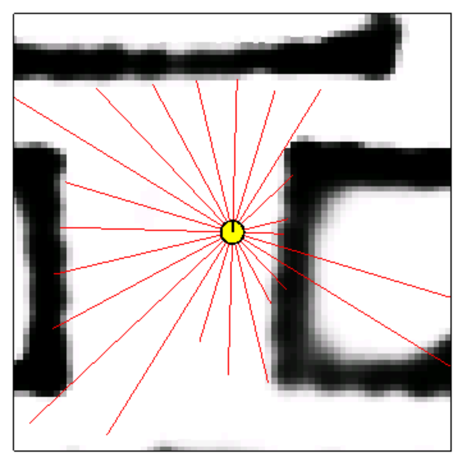
Probabilistic Sensor Models

Sensors for Mobile Robots

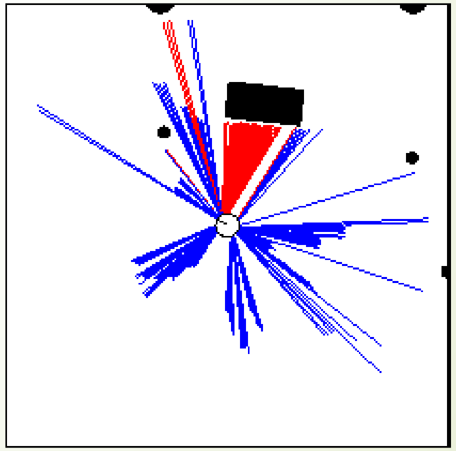
- Contact sensors: Bumpers
- Proprioceptive sensors
  - Accelerometers (spring-mounted masses)
  - Gyroscopes (spinning mass, laser light)
  - Compasses, inclinometers (earth magnetic field, gravity)
- Proximity sensors
  - Sonar (time of flight)
  - Radar (phase and frequency)
  - Laser range-finders (triangulation, tof, phase)
  - Infrared (intensity)
- Visual sensors: Cameras
- Satellite-based sensors: GPS

Proximity Sensors

- The central task is to determine $p(z|x)$, i.e., the probability of a measurement z given that the robot is at position x.
- Question: Where do the probabilities come from?
- Approch: Let's try to explain a measurement



Sonar Scan



Laser Scan

Beam-based Sensor Model

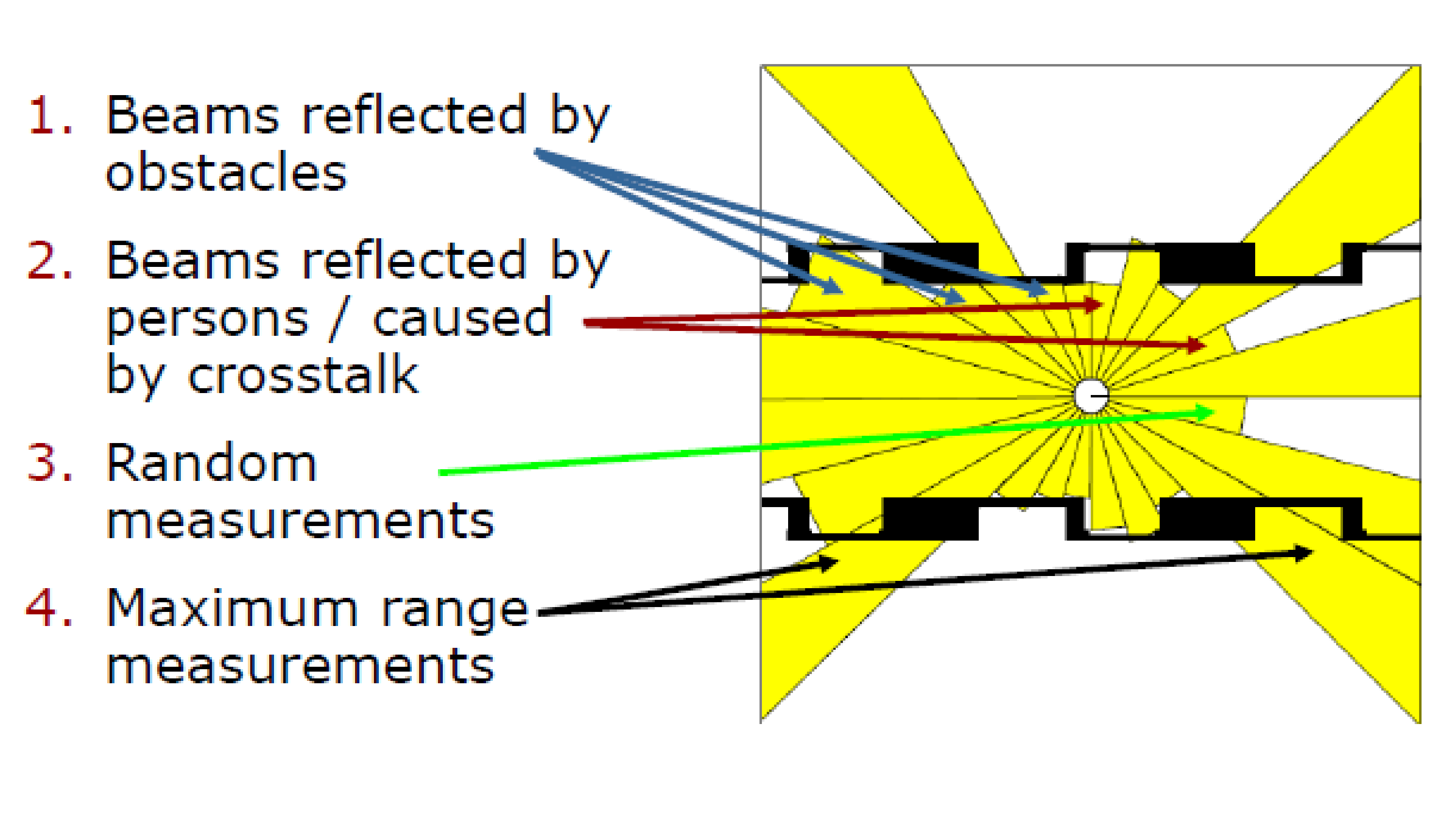
- Scan z consists of K measurements

$$z = \{z_1, z_2, \ldots, z_K\}$$

- Individual measurements are independent given the robot position

$$P(z|x,m) = \prod_{k=1}^{K} P(z_k|x,m)$$

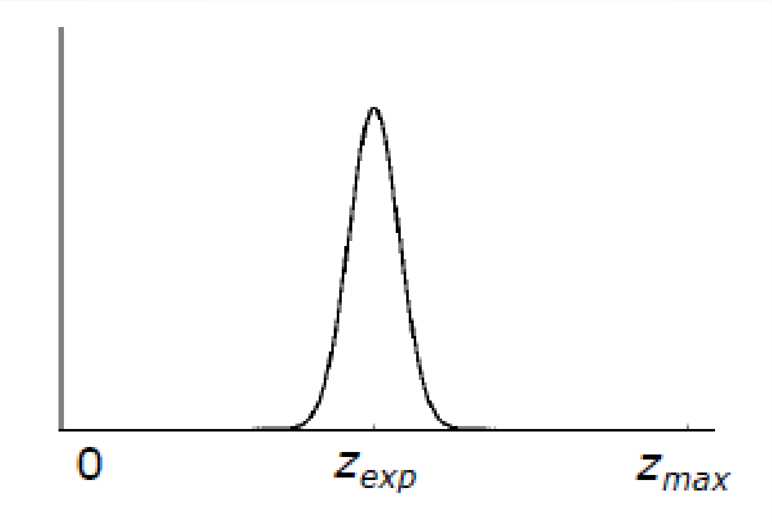
Typical measurement Errors of Range Measurements



Proximity Measurement

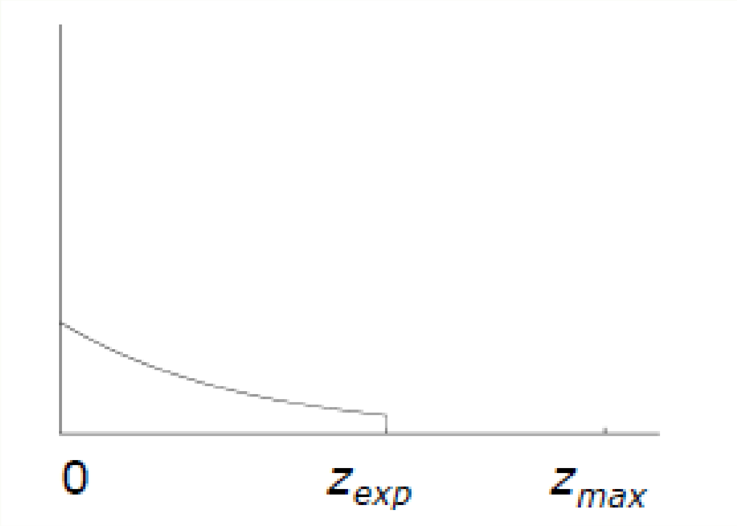
- Measurement can be caused by
  - A known obstacle
  - Cross-talk
  - An unexpeceted obstacle (people, furniture, ..)
  - Missing all obstacles (total reclection, glass, ..)
- Noise is due to uncertainty
  - in measuring distance to known obstacle
  - in position of known obstacles
  - in position of additional obstacles
  - Whether obstacle is missed

1- Correct range with local measurement noise



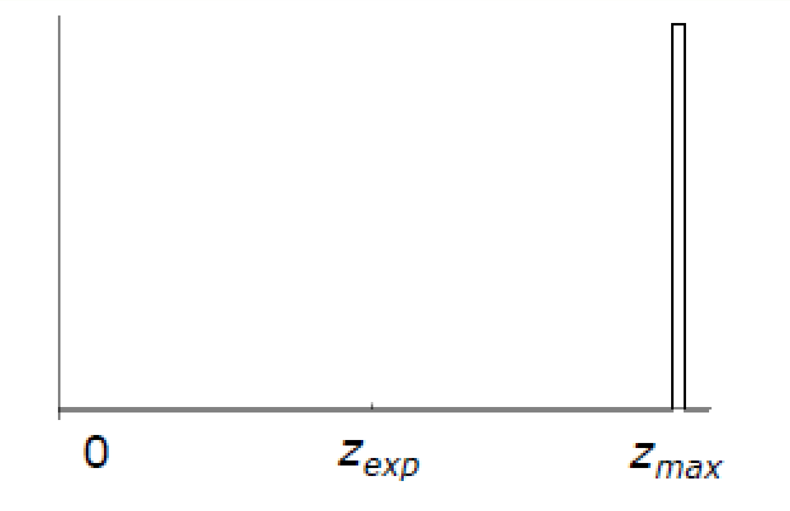
$$p_{hit}(z|x,m) = \eta \frac{1}{\sqrt{2\pi\sigma_{hit}^2}} e^{-\frac{1}{2}\frac{(z - z_{exp})^2}{\sigma_{hit}^2}}$$

2 – Unexpected Obstacles



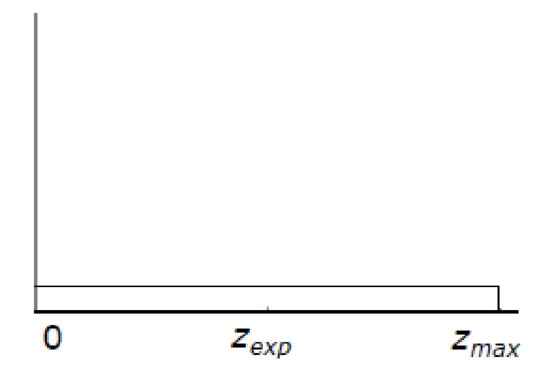
$$p_{unexp}(z|x,m) = \begin{cases} \eta\lambda e^{-\lambda z} & z < z_{exp} \\ 0 & otherwise \end{cases}$$

3 - Failures



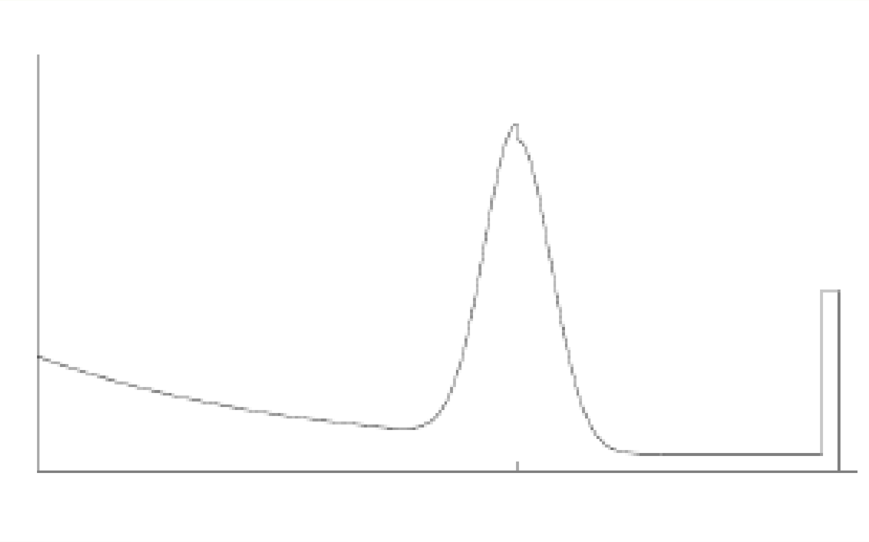
$$p_{max}(z|x,m) = \eta \frac{1}{z_{small}}$$

4 – Random Measurements



$$p_{rand}(z|x,m) = \eta \frac{1}{z_{max}}$$

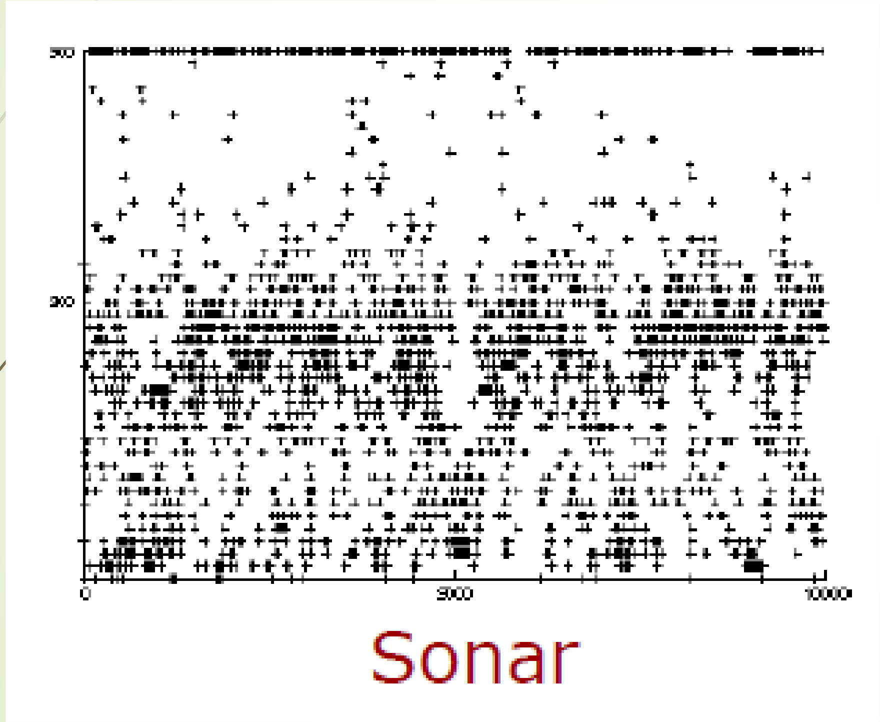
Resulting Mixture Density

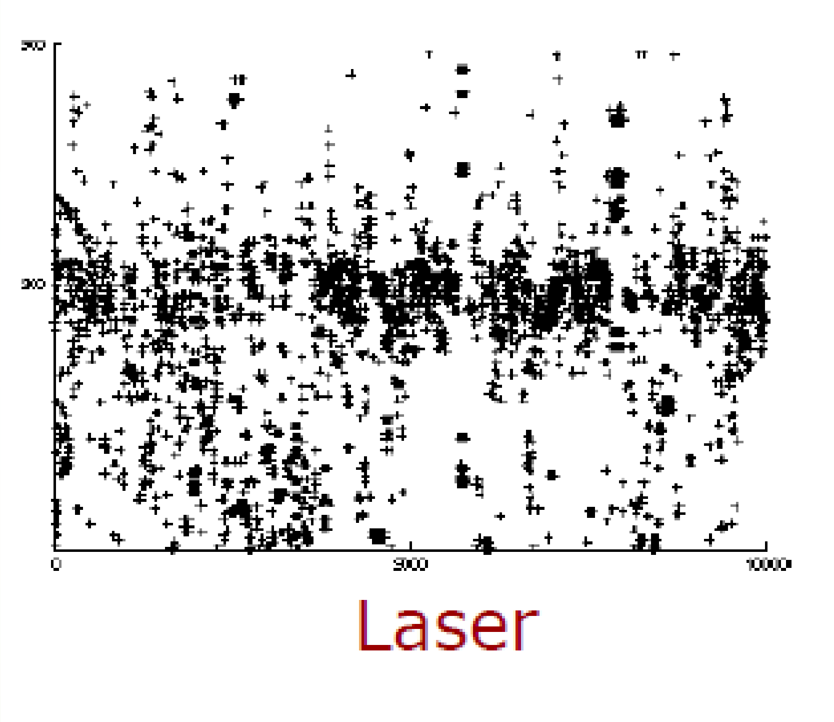


$$p(z|x,m) = \begin{pmatrix} \alpha_{hit} \\ \alpha_{unexp} \\ \alpha_{max} \\ \alpha_{rand} \end{pmatrix}^T \cdot \begin{pmatrix} p_{hit}(z|x,m) \\ p_{unexp}(z|x,m) \\ p_{max}(z|x,m) \\ p_{rand}(z|x,m) \end{pmatrix}$$

How can we determine the model parameters?

Raw Sensor Data





Measured distances for expected distance of 300 cm

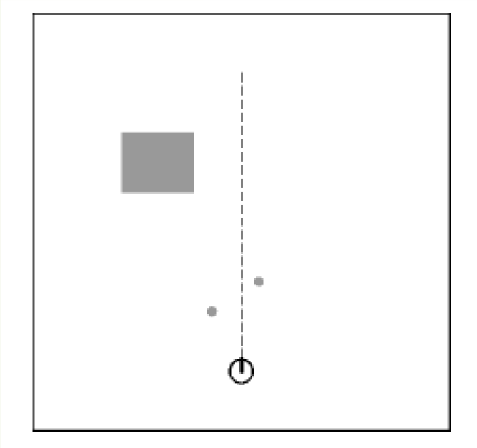
Scan-based Model (Likelihood Field Model)

- Beam-based model is
  - Not smooth for small obstacles and at edges
  - Nor very efficient
- Idea: Instead of following along the beam, just check the end point.

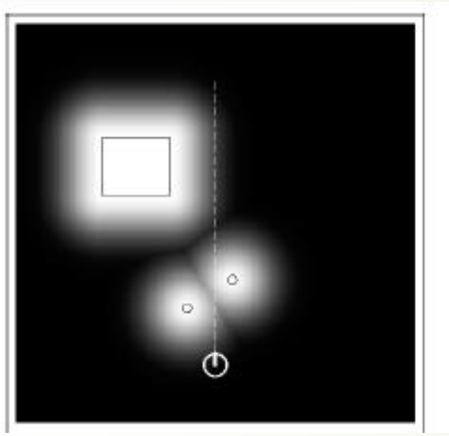
Scan-based Model

- Probability is a mixture of …
  - A Gaussian distribution with mean at distance to closest obstacle
  - A uniform distribution for random measurements and,
  - A small uniform distribution for max range measurements
- Again, independence between different components is assumed.

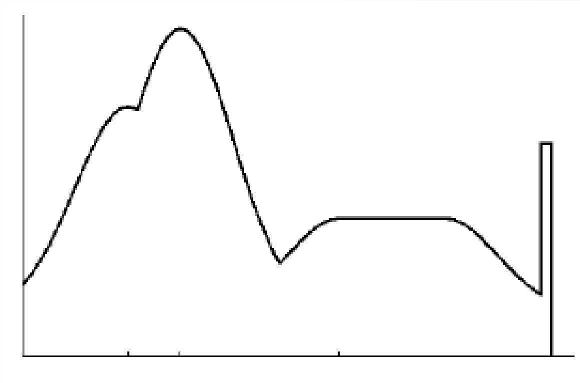
Example



Map m



Likelihood field



P(z|x,m)

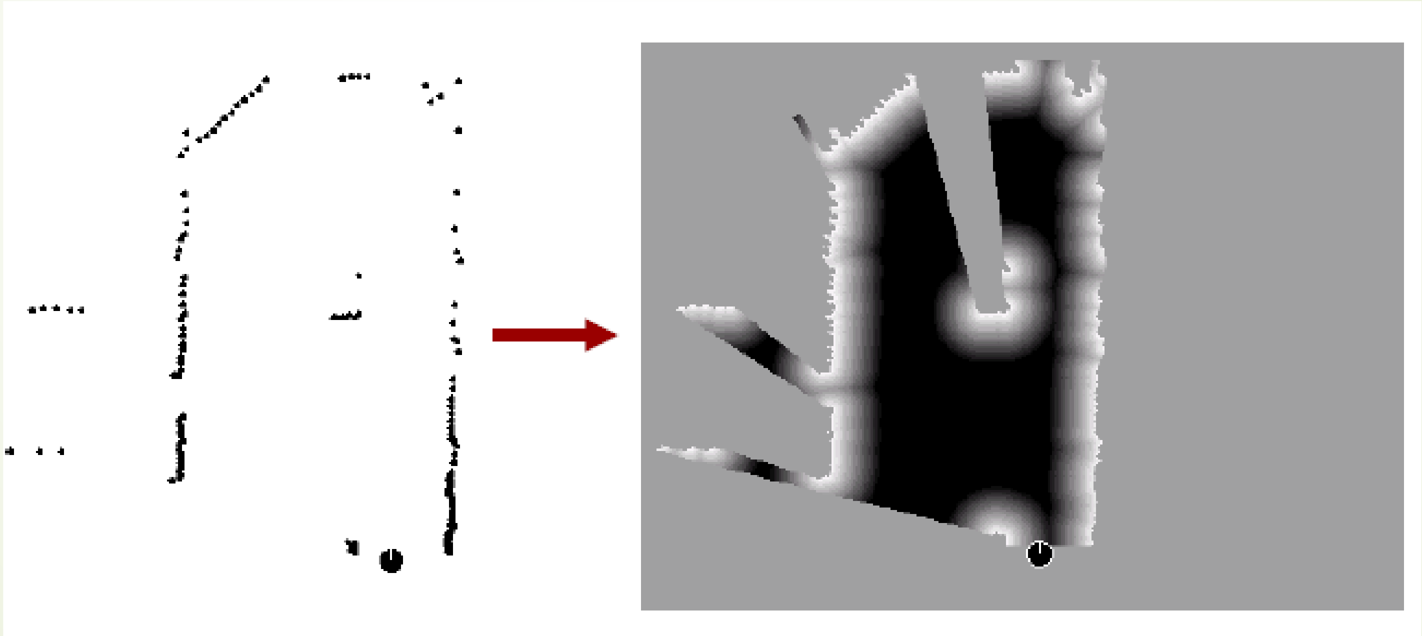
San Jose Tech Museum

Occupancy grid map

Likelihood field

Scan Matching

- Extract likelihood field from scan and use it to match different scan



Properties of Scan-based Model

- Highly efficient, uses 2D tables only
- Distance grid is smooth w.r.t. to small changes in robot position
- Allows gradient descent, scan matching.
- Ignores physical properties of beams

Additional Models of Proximity Sensors

- Map matching (sonar, laser): generate small, local maps from sensor data and match local maps against global model
- Scan matching (laser): map is represented by scan endpoints, match scan into this map
- Features (sonar, laser, vision): Extract features such as doors, hallways from sensor data
  - For range sensors: identify lines, corners
  - For cameras: identify edges, distinct patterns

Landmarks

- Active beacons (e.g. Radio, GPS)
- Passive (e.g., visual, Retro-reflective)

- Sensor provides
  - Distance, or
  - Bearing, or
  - Distance and bearing
  - A signature may be generated

Calculation the probability of a feature with known corresponce

Algorithm landmark_model_known_correspondence(f_t^i, c_t^i, x_t, m):

$j = c_t^i$

$\hat{r} = \sqrt{(m_{j,x} - x)^2 + (m_{j,y} - y)^2}$

$\hat{\phi} = \text{atan2}(m_{j,y} - y, m_{j,x} - x)$

$q = \textbf{prob}(r_t^i - \hat{r}, \sigma_r^2) \cdot \textbf{prob}(\phi_t^i - \hat{\phi}, \sigma_\phi^2) \cdot \textbf{prob}(s_t^i - s_j, \sigma_s^2)$

return q

Table 6.4 Algorithm for computing the likelihood of a landmark measurement. The algorithm requires as input an observed feature $f_t^i = (r_t^i \; \phi_t^i \; s_t^i)^T$, and the true identity of the feature c_t^i, the robot pose $x_t = (x \; y \; \theta)^T$, and the map m. It's output is the numerical probability $p(f_t^i \mid c_t^i, m, x_t)$.

Summary of Sensor Models

- Explicitly modeling uncertainty in sensing is key to robustness.
- In many cases, good models can be found by the following approach:
  - Determine parametric model of noise free measurement
  - Analyze sources of noise
  - Add adequate noise to parameters (eventually mix in densities for noise)
  - Learn (and verify) parameters by fitting model to data
  - Likelihood of measurement is given by probabilistically comparing the actual with expected measurement.
- This holds for motion models as well.
- It is extremely important to be aware of the underlying assumptions!

References

- Probabilistic Robotics, Sebastian Thrun, Wolfram Burgard and Dieter Fox, Chapter 6