Robot Localization

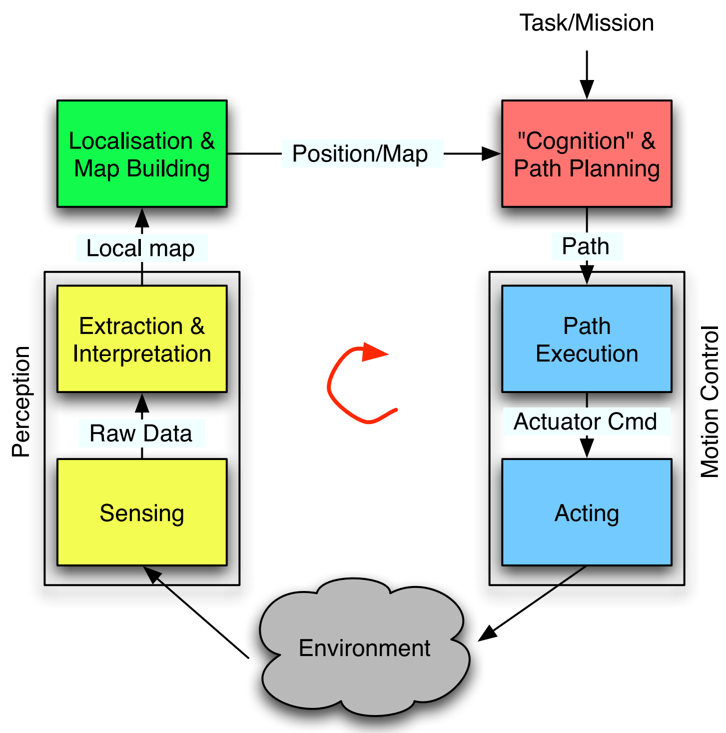
Henrik I Christensen

Contextual Robotics
UC San Diego
La Jolla, CA
hichristensen@ucsd.edu

Outline

1. Introduction
2. Taking it apart
3. Representations
4. Features
5. Prediction
6. Matching
7. Position Estimation
   - Topological Pose Estimation
   - Gaussian PDF
   - Monte-Carlo Methods
8. Examples
9. Wrap-up

System Context



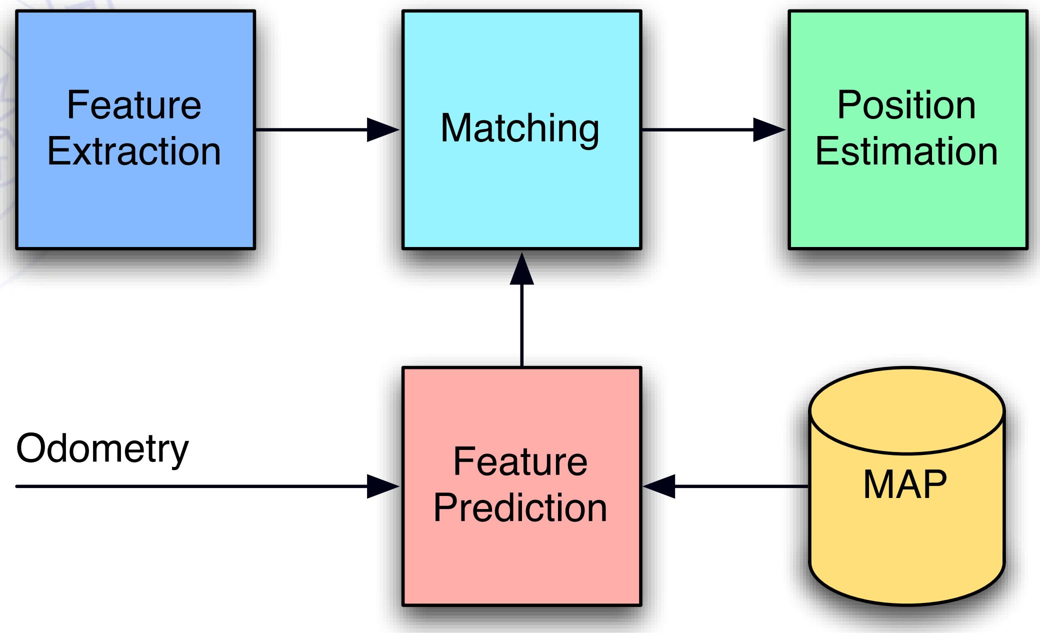
Problems in Mobile Robotics

1. Where am I?
2. Where do I need to go?
3. When have I arrived?

Outline

1. Introduction
2. Taking it apart
3. Representations
4. Features
5. Prediction
6. Matching
7. Position Estimation
   - Topological Pose Estimation
   - Gaussian PDF
   - Monte-Carlo Methods
8. Examples
9. Wrap-up

The process



What do we need?

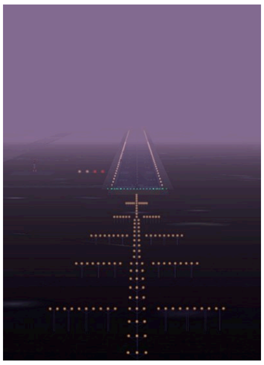
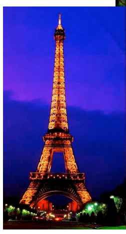
Question:

What do we need to complete the process?

The components

- Map representation
- Selection of features
- A strategy for matching of features to maps
- A way to predict vehicle position
- A way to predict feature locations

Map/Setup

Odometry	Instrumented Environments	High-level features
 Treasure map	 airport light	Elevator  Entrance  Eiffel tower
Unreliable	Expensive	Requires recognition challenging

Outline

1. Introduction
2. Taking it apart
3. Representations
4. Features
5. Prediction
6. Matching
7. Position Estimation
   - Topological Pose Estimation
   - Gaussian PDF
   - Monte-Carlo Methods
8. Examples
9. Wrap-up

Map representations

Question:

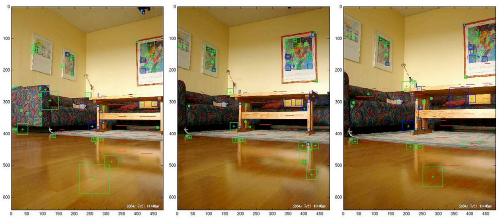
What would be good map representations?

Map representation

- Appearance based maps
- Topological maps
- Grid based maps
- Feature based maps

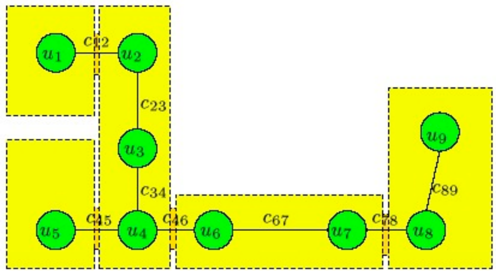
Environmental Maps

Landmarks

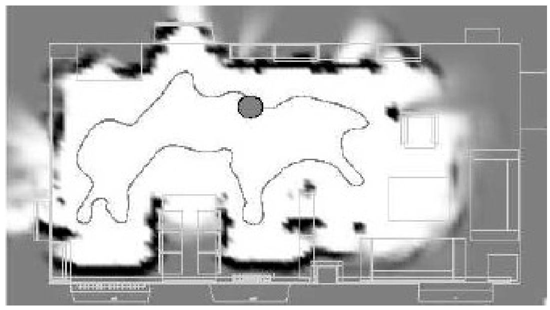


Feature Maps

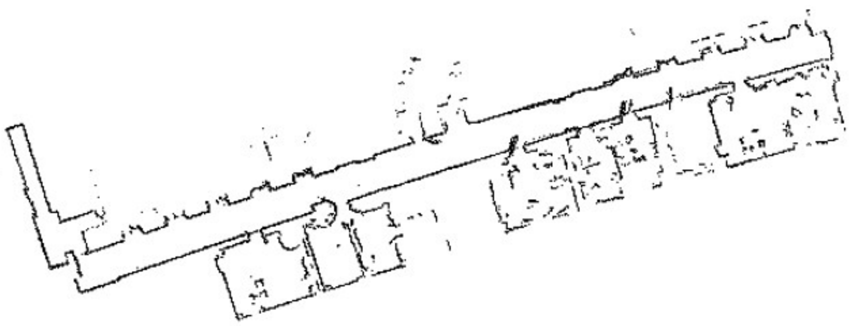
Topological Map



Grid Maps



Appearance based maps



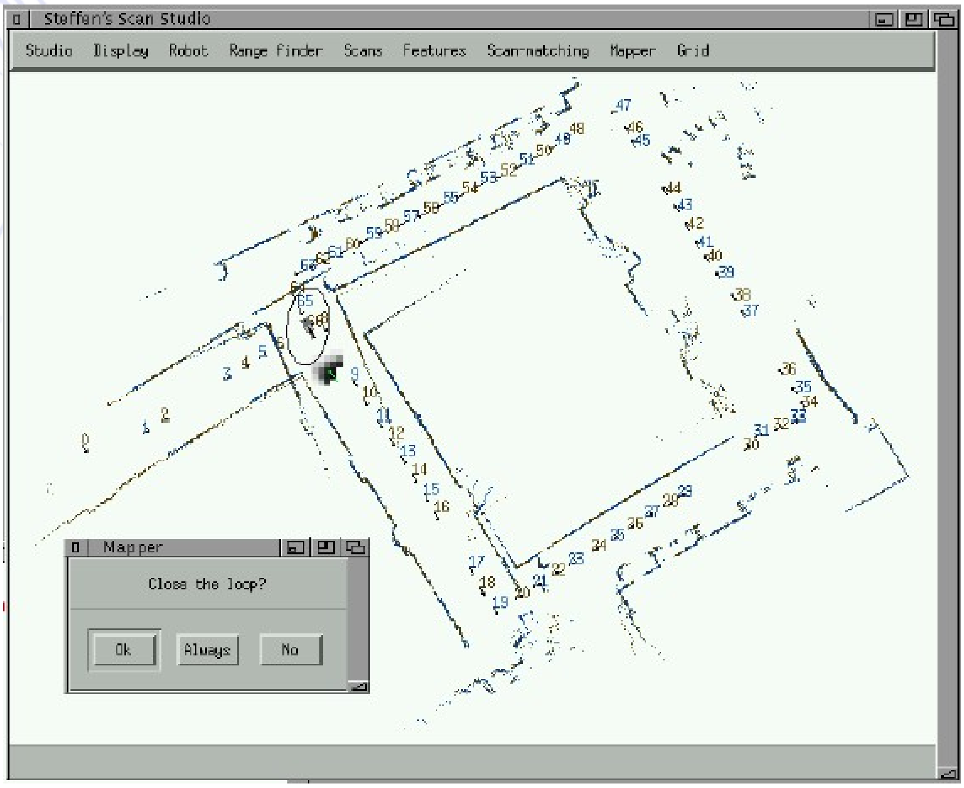
Pros

- Direct alignment of sensor data
- Easy to model

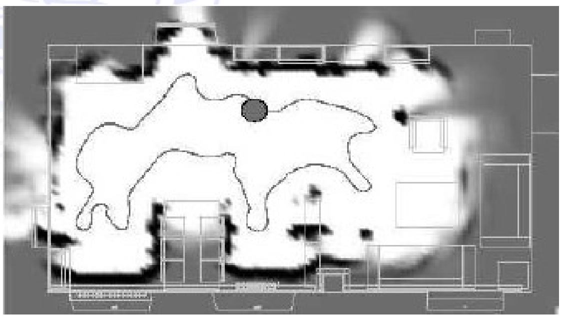
Cons

- Generalizes poorly
- One sensor system
- Ex: ScanStudio by Gutmann

ScanStudio



Gridmaps

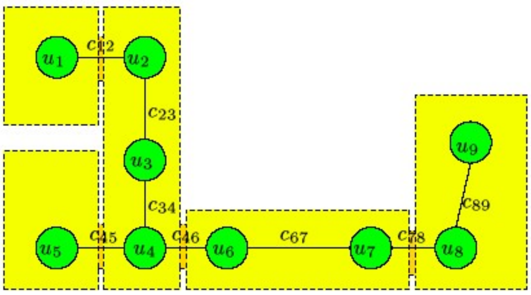


- Easy to understand
- O(env^2) in size
- Easy to update (might be slow)

Feature Maps

- Discrete feature map
- Easy for multi feature intg
- Easy to handle
- O(*features*)

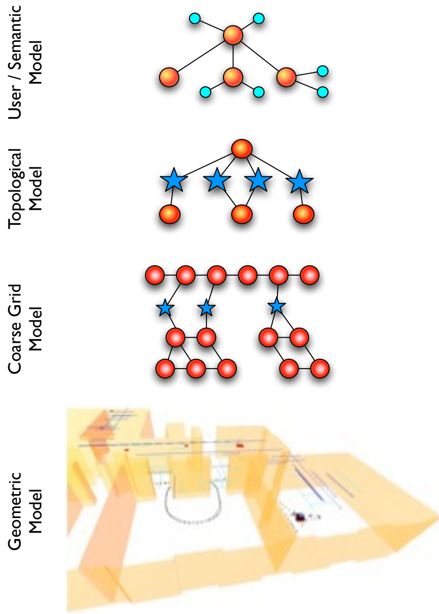
Topological Maps



- Graph representation
- Place recognition
- O(*places*)
- Coarse localisation
- A good planning rep.

Mixed Representations

- Mixed maps are gaining in importance
- Topology for overall layout
- Sub-maps for detailed models
- A way to handle complexity



Outline

1. Introduction
2. Taking it apart
3. Representations
4. Features
5. Prediction
6. Matching
7. Position Estimation
   - Topological Pose Estimation
   - Gaussian PDF
   - Monte-Carlo Methods
8. Examples
9. Wrap-up

Features

- Already covered in a separate lecture(s)
- Which features for which maps?

Method	Grid	Appearance	Feature	Topology
Raw Data	yes	YES		
Points		yes	YES	
Lines			YES	yes
Geometry			YES	yes
Object	yes		YES	YES

Features

- Feature models include
  - ID, Parameters, Uncertainty
- Gaussian Model: $\mathcal{N}(\mu, \Sigma)$
- Generic Sensor Model: $p(z|x)$
- Matching $\leftrightarrow$ Estimation $\leftrightarrow$ Detection

Outline

1. Introduction
2. Taking it apart
3. Representations
4. Features
5. Prediction
6. Matching
7. Position Estimation
    - Topological Pose Estimation
    - Gaussian PDF
    - Monte-Carlo Methods
8. Examples
9. Wrap-up

Pose / Map Prediction

- Initial estimation of new pose based on odometry data
    - Already covered as part of uncertainty modelling
    - Covariance propagation

$$p' = \begin{bmatrix} x \\ y \\ \theta \end{bmatrix} + \begin{bmatrix} \Delta s \cos\left(\theta + \frac{\Delta\theta}{2}\right) \\ \Delta s \sin\left(\theta + \frac{\Delta\theta}{2}\right) \\ \frac{\Delta s_r - \Delta s_l}{2l} \end{bmatrix}$$

$$\Sigma_{p'} = \nabla_p f \Sigma_p \nabla_p f^T + \nabla_{\Delta_{rl}} f \Sigma_\Delta \nabla_{\Delta_{rl}} f^T$$

- Propagation of uncertainty to map features
    - Kinematic Update
    - Pose uncertainty as your system noise (Q)

Outline

1. Introduction
2. Taking it apart
3. Representations
4. Features
5. Prediction
6. Matching
7. Position Estimation
   - Topological Pose Estimation
   - Gaussian PDF
   - Monte-Carlo Methods
8. Examples
9. Wrap-up

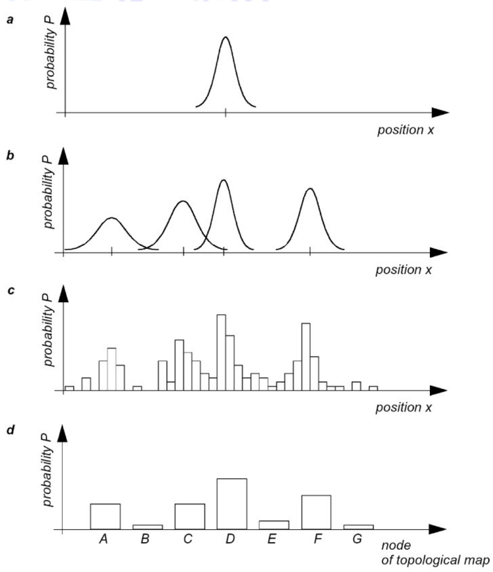
Matching

- Tied to the map representation
- Grid based - measure correlation/match
- Grid based - voting based matching
- Appearance - voting / scan correlation
- Feature based
  - Nearest neighbor
  - Mahalanobis / Probabilistic
  - Voting based

Outline

1. Introduction
2. Taking it apart
3. Representations
4. Features
5. Prediction
6. Matching
7. Position Estimation
   - Topological Pose Estimation
   - Gaussian PDF
   - Monte-Carlo Methods
8. Examples
9. Wrap-up

Pose updating – Uncertainty Model



- The selection of an uncertainty model
  - Single hypothesis
  - Sum of Gaussians
  - Probability grid
  - Topological Graph
  - Particle Based

Pose updating - Uncertainty Model

- The selection of an uncertainty model influences the updating methodology
- The uncertainty model is coupled to the environmental representation
- The model influences strongly the computational requirements

Uncertainty Modelling – Markov Approach

- Assume the world is divided into places/states $s \in P$
- Estimation of $p(s_t)$ given s_{t-1} and sensory data z_t
- Formally

$$p(s_t|z_t) = \int p(s_t|s'_{t-1}, z_t)p(s_{t-1})ds'_{t-1}$$

Integration needed as s_t could be reached from multiple locations

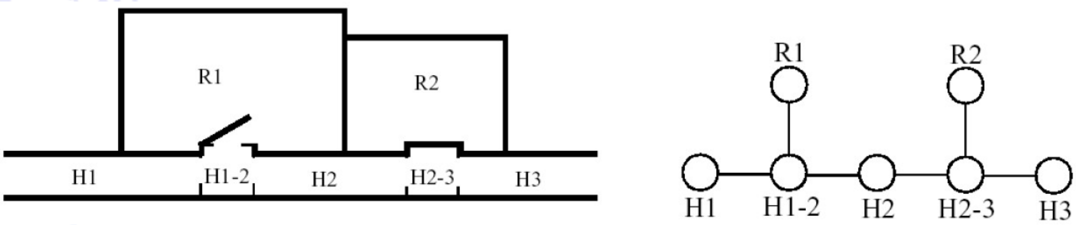
Uncertainty modelling – Markov Approach

- Markov assumption: all knowledge encoded in the pose/state estimate
- There is a probability model for motion updating
- There is a model for $p(z|s)$ i.e. a sensor model, as

$$p(s|z) = \frac{p(z|s)p(s)}{p(z)}$$

  where $p(s)$ is location model and $p(z)$ is the sensor noise model
- These assumptions are relative weak

Topological modelling – dervish example



	Wall	Closed door	Open door	Open hallway	Foyer
Nothing detected	0.70	0.40	0.05	0.001	0.30
Closed door detected	0.30	0.60	0	0	0.05
Open door detected	0	0	0.90	0.10	0.15
Open hallway detected	0	0	0.001	0.90	0.50

Topological modelling – dervish example

- Here the probability updating is used for direct lookup of $p(s|z)$, where s is any of the nodes in the topological map
- As robot moved through environment the graph is updated with new information
- The probability table is small and efficient to handle
- The localisation is coarse (location oriented)

Pose estimation with Gaussian Model

- The pose is approximated by a single Gaussian function

$$p(s) = \frac{1}{\sqrt{2\pi\Sigma_s}} \exp\left(-\frac{1}{2}(s-\bar{s})\Sigma_s^{-1}(s-\bar{s})^T\right)$$

- s is here a continuous function and Σ_s is the associated uncertainty estimate
- Updating is normally performed using a Kalman filter model

Kalman filter – State space model

$$
\begin{aligned}
s_t &= F s_{t-1} + G u_t + w_t \\
z_t &= H s_t + v_t
\end{aligned}
$$

- where F is the system model, G is the deterministic input, H is a prediction of where features are in the world, w is the system noise, and v is the measurement noise

Monte-Carlo Based Methods

- Monte-Carlo based methods is using a sample model for approximation of the pose estimate
- Using a grid model as presented earlier
- Assume with have a number of particles in a collection

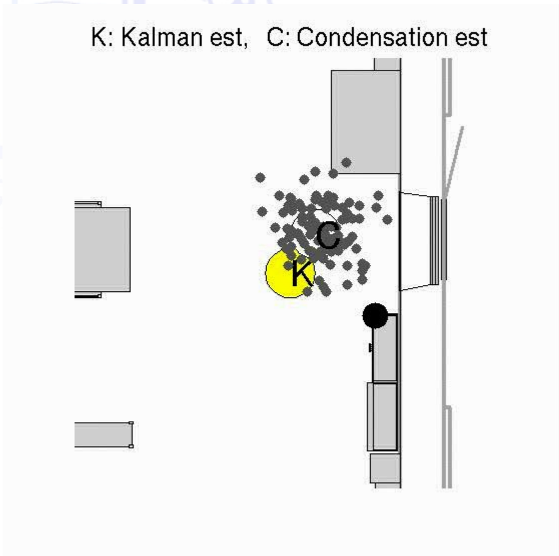
$$S_t = \left\{ (s_t^{(i)}, \pi_t^{(i)}) | i = 1..N \right\}$$

  each particle is a hypothesis for the position of the robot, and $\pi_t^{(i)}$ is an associated weight
- We can now approximate $p(s_t | z_0, z_1, ...z_t)$ for any distribution of the pose hypotheses

Monte-Carlo Strategy

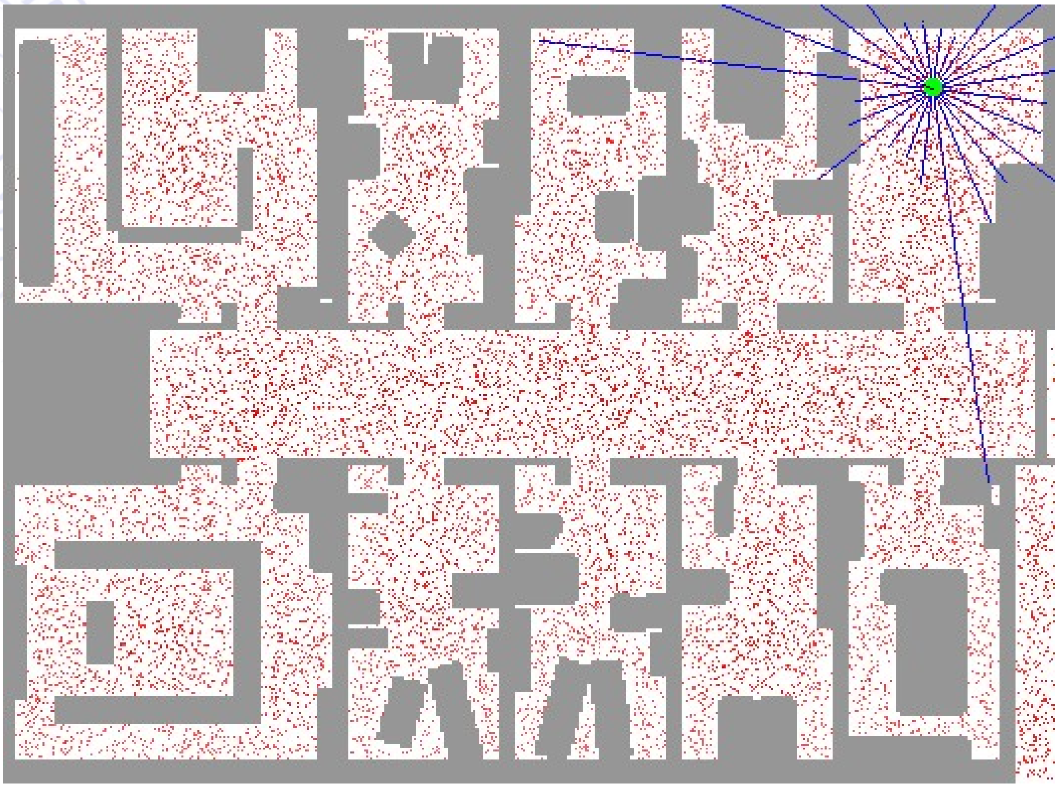
1. Draw N samples from an initial PDF. Typically a uniform distribution. Give each sample a weight of $\frac{1}{N}$
2. Propagate the motion information and draw a new sample from the distribution $p(s_{t+1}^{(i)}|s_t^{(i)}, o_t)$
3. Set the weight of the sample to $\pi_{t+1}^{(i)} = p(z_{t+1}|s_{t+1}^{(i)}) * \pi_t^{(i)}$ based on sensory input
4. Generate a new sample set by drawing samples from the current set and a basis distribution (typically uniform). Normalize the weights
5. Go back to step 2

Monte-Carlo Example



- Example of particle distribution about estimate of position
- Sonar readings for update of the position
- Video of system in operation

Monte-Carlo Example – Burgard, Fox & Thrun



Monte-Carlo Discussion

- Efficient to approximate any distribution of the pose
- The number of particles can be adopted to a particular platform
- Can be used both for simple and multi robot localisation

Outline

1 Introduction

2 Taking it apart

3 Representations

4 Features

5 Prediction

6 Matching

7 Position Estimation
- Topological Pose Estimation
- Gaussian PDF
- Monte-Carlo Methods

8 Examples

9 Wrap-up

Localisation / Mapping Example

- Now mapping and localisation is also integrated to allow for autonomous operation in general environments
- The mapping and localisation can be integrated to generate – Simultaneous Localisation and Mapping (SLAM)
- Indoor example VIDEO
- Outdoor example VIDEO

Outline

1. Introduction
2. Taking it apart
3. Representations
4. Features
5. Prediction
6. Matching
7. Position Estimation
   - Topological Pose Estimation
   - Gaussian PDF
   - Monte-Carlo Methods
8. Examples
9. Wrap-up

Wrap-Up

- Localisation is a fundamental competence in mobile robotics
- Involves two major steps
  - Prediction of motion (kinematic modelling)
  - Updating of pose estimate(s)
- The method used depends upon the adopted model for handling of uncertainty and the associated world model
- Brief introduction to the main methods for estimation
- A few illustrative examples