Target Tracking and Multi-Robot Localization Methods in Unknown Environments Using KF Algorithms

Polychronis Kondaxakis
BEng, MSc, PhD, MIEEE, MIET
p.kondaxakis@ieee.org

Contents

1. Introduction- Who I am!
2. Multi-Robot Localization Problem
3. Target Tracking Problem
4. Conclusions
5. Future Work

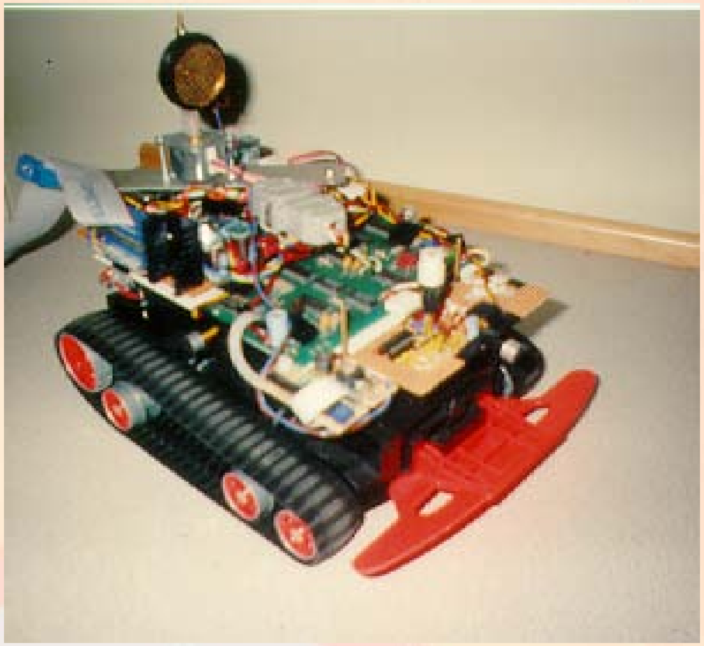
Introduction

- **Education**
  - PhD in Cybernetics (University of Reading)
  - MSc in Automation and Control (University of Newcastle Upon Tyne)
  - BEng in Robotics and Automated Manufacture (University of Sussex)
- **Experience**
  - Research scientist at Foundation for Reseach & Technology-Hellas (FORTH)
  - Interactive table developer at POLYMECHANON@Allou science park.
  - Research assistant at the University of Reading

Introduction

My first mobile robot: HERMES 1

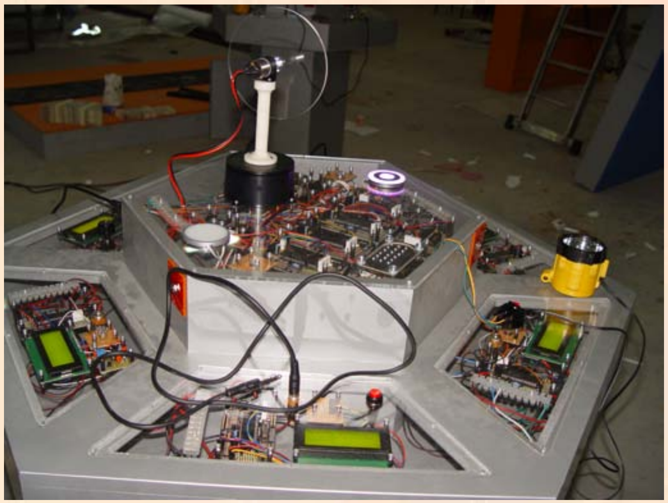
- BASIC STAMP 2 microcontroller
- 32-count optical encoders
- Bumper switches
- LDR sensors
- Ultrasound sensors

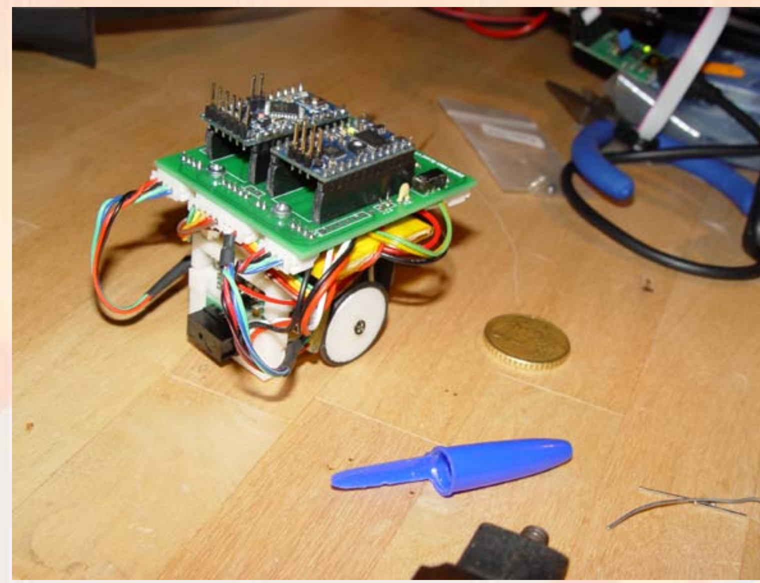


Introduction

INTERACTIVE TABLE

MICROBOT





Introduction

- **LOCALIZATION:**
  - Where am I?
- **TRACKING:**
  - Where are the Others?

Contents

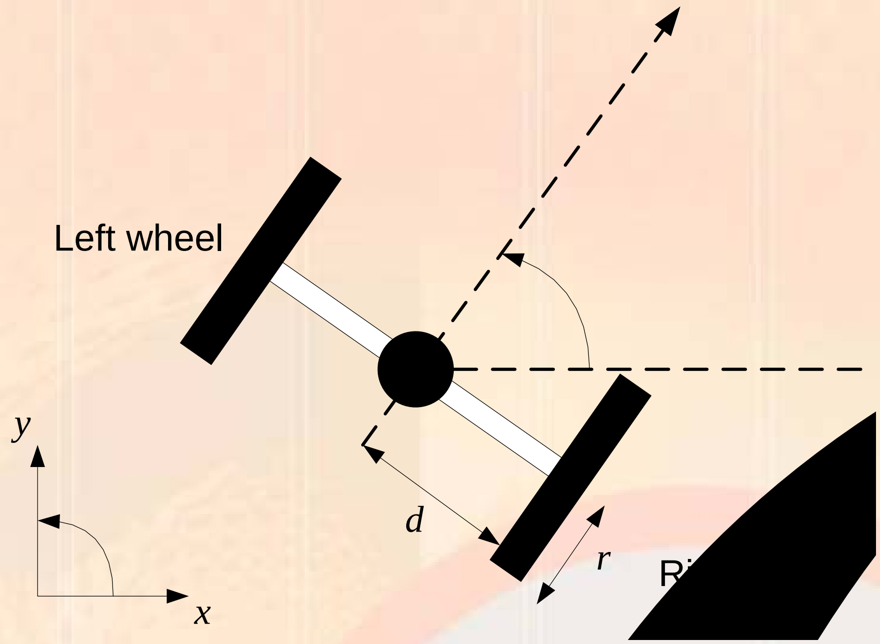
Localization Problem

The University of Reading

- Absolute or relative measurements
- Communication between robots
- Robots are viewed as a homogeneous group entity
- A centralized EKF is used for robots' state estimation

Localization Problem

Robots are described by differential drive configuration



$$SP(k) = \frac{\omega_L(k) + \omega_R(k)}{2} \times r \qquad \dot{\theta}(k) = \frac{\omega_L(k) - \omega_R(k)}{d} \times r$$

Localization Problem

The recursive state space model

$$\mathbf{X}_i(k+1) = f(\mathbf{X}_i(k), \mathbf{u}_i(k)) + \mathbf{w}_i(k) \quad i=1\ldots I \qquad \mathbf{z}_n(k+1) = h_n\big(\mathbf{X}_{i,j}(k+1)\big) + \mathbf{v}_n(k+1) \quad n=1,2\ldots N$$

States of Robot i

Gaussian noise

Non-linear measurement function

Non-linear state transition function

nth measurement vector

Gaussian noise

Each robot's state vector is composed by position (*x*,*y*) and orientation *θ* components

$$f(\mathbf{X}_i(k), \mathbf{u}_i(k)) = \begin{bmatrix} x_i(k) + SP_i(k)\cos\theta_i(k)T \\ y_i(k) + SP_i(k)\sin\theta_i(k)T \\ \theta_i(k) + \dot{\theta}_i(k)T \end{bmatrix}$$

Localization Problem

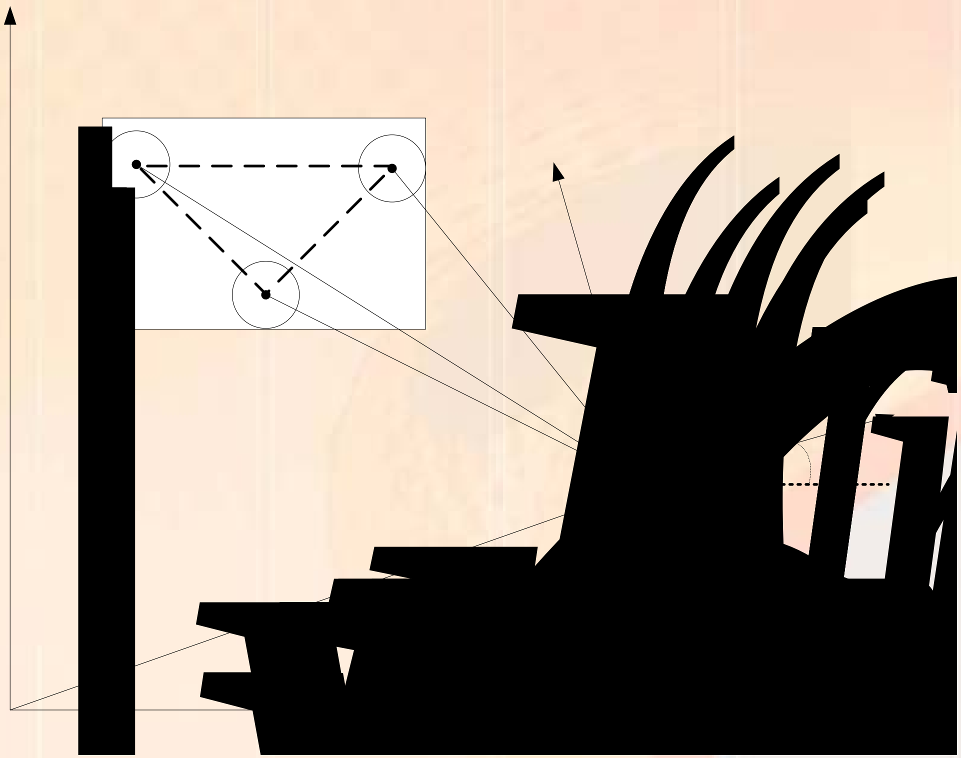
$$z_{i,j}(k) = \sqrt{\left(x_i(k) - x_j(k)\right)^2 + \left(y_i(k) - y_j(k)\right)^2}$$

$$\phi_{i,j}(k) = \arctan\frac{y_i(k) - y_j(k)}{x_i(k) - x_j(k)} - \theta_i(k)$$

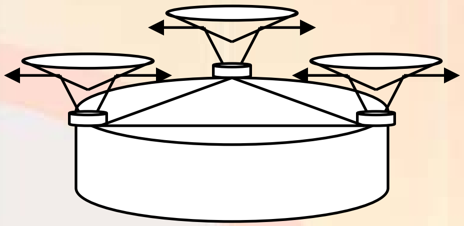
MIABOT Pro

Localization Problem

2D Ultrasound trilateration sensor for relative measurements



Reflection Cone



Tri-Transducer

Localization Problem

EKF needs a linearization procedure…

$$\Delta\hat{\mathbf{X}}_i(k+1)^- \approx \boldsymbol{\Phi}_i\left(\hat{\mathbf{X}}_i(k|k)\right)\Delta\hat{\mathbf{X}}_i(k)^+ + \mathbf{w}_i(k) \qquad i = 1\ldots I$$

State prediction error

State transition matrix

State estimated error

$$\Delta\hat{\mathbf{z}}_n(k+1) \approx \mathbf{H}_n\left(\hat{\mathbf{X}}_{i,j}(k+1|k)\right)\Delta\hat{\mathbf{X}}_{i,j}(k+1)^- + \mathbf{v}_n(k+1) \quad n = 1\ldots N$$

*n*th measurement error vector

Measurement matrix

State prediction error

c-EKF in block matrix form

Robot 1 Robot 2 Robot 3

$$\begin{bmatrix} \Delta\hat{\mathbf{X}}_1(k+1)^- \\ \Delta\hat{\mathbf{X}}_2(k+1)^- \\ \Delta\hat{\mathbf{X}}_3(k+1)^- \end{bmatrix} = \begin{bmatrix} \mathbf{\Phi}_1\left(\hat{\mathbf{X}}_1(k|k)\right) & 0 & 0 \\ 0 & \mathbf{\Phi}_2\left(\hat{\mathbf{X}}_2(k|k)\right) & 0 \\ 0 & 0 & \mathbf{\Phi}_3\left(\hat{\mathbf{X}}_3(k|k)\right) \end{bmatrix} \times \begin{bmatrix} \Delta\hat{\mathbf{X}}_1(k)^+ \\ \Delta\hat{\mathbf{X}}_2(k)^+ \\ \Delta\hat{\mathbf{X}}_3(k)^+ \end{bmatrix} + \begin{bmatrix} \mathbf{w}_1(k+1) \\ \mathbf{w}_2(k+1) \\ \mathbf{w}_3(k+1) \end{bmatrix}$$

2×3 –dimensional space

Localization Problem

Measurement model uses N = 6 measurement vectors

Robot 1

Robot 2

$$\Delta \mathbf{z}_c(k+1) = \begin{bmatrix} \Delta\hat{\mathbf{z}}_1(k+1) \\ \Delta\hat{\mathbf{z}}_2(k+1) \\ \Delta\hat{\mathbf{z}}_3(k+1) \\ \Delta\hat{\mathbf{z}}_4(k+1) \\ \Delta\hat{\mathbf{z}}_5(k+1) \\ \Delta\hat{\mathbf{z}}_6(k+1) \end{bmatrix} = \begin{bmatrix} \mathbf{H}_{1,2}(k+1) & -\mathbf{H}_{1,2}(k+1) & \vec{0}_{2\times3} \\ \mathbf{H}_{1,3}(k+1) & \vec{0}_{2\times3} & -\mathbf{H}_{1,3}(k+1) \\ -\mathbf{H}_{2,1}(k+1) & \mathbf{H}_{2,1}(k+1) & \vec{0}_{2\times3} \\ \vec{0}_{2\times3} & \mathbf{H}_{2,3}(k+1) & -\mathbf{H}_{2,3}(k+1) \\ -\mathbf{H}_{3,1}(k+1) & \vec{0}_{2\times3} & \mathbf{H}_{3,1}(k+1) \\ \vec{0}_{2\times3} & -\mathbf{H}_{3,2}(k+1) & \mathbf{H}_{3,2}(k+1) \end{bmatrix} \begin{bmatrix} \Delta\hat{\mathbf{X}}_1(k+1)^- \\ \Delta\hat{\mathbf{X}}_2(k+1)^- \\ \Delta\hat{\mathbf{X}}_3(k+1)^- \end{bmatrix} + \begin{bmatrix} \mathbf{v}_1(k+1) \\ \mathbf{v}_2(k+1) \\ \mathbf{v}_3(k+1) \\ \mathbf{v}_4(k+1) \\ \mathbf{v}_5(k+1) \\ \mathbf{v}_6(k+1) \end{bmatrix}$$

Robot 3

Localization Problem

Centralized EKF prediction cycle

$$\hat{\mathbf{X}}_c(k+1|k)=\begin{bmatrix}\hat{\mathbf{X}}_1(k+1|k)\\ \hat{\mathbf{X}}_2(k+1|k)\\ \hat{\mathbf{X}}_3(k+1|k)\end{bmatrix}=\begin{bmatrix}f_1\left(\hat{\mathbf{X}}_1(k|k),\mathbf{u}_1(k)\right)\\ f_2\left(\hat{\mathbf{X}}_2(k|k),\mathbf{u}_2(k)\right)\\ f_3\left(\hat{\mathbf{X}}_3(k|k),\mathbf{u}_3(k)\right)\end{bmatrix}\quad \mathbf{P}_c(k+1|k)=\begin{bmatrix}\mathbf{\Phi}_1\mathbf{P}_{11}(k|k)\mathbf{\Phi}_1^{\ T}+\mathbf{Q}_1 & 0 & 0\\ 0 & \mathbf{\Phi}_2\mathbf{P}_{22}(k|k)\mathbf{\Phi}_2^{\ T}+\mathbf{Q}_2 & 0\\ 0 & 0 & \mathbf{\Phi}_3\mathbf{P}_{33}(k|k)\mathbf{\Phi}_3^{\ T}+\mathbf{Q}_3\end{bmatrix}$$

Centralized EKF update cycle

$$\mathbf{S}_c(k+1)=\mathbf{H}_c\left(\hat{\mathbf{X}}_c(k+1|k)\right)\mathbf{P}_c(k+1|k)\mathbf{H}_c^{\ T}\left(\hat{\mathbf{X}}_c(k+1|k)\right)+\mathbf{R}_c(k+1)$$

$$\mathbf{K}_c(k+1)=\mathbf{P}_c(k+1|k)\mathbf{H}_c^{\ T}\left(\hat{\mathbf{X}}_c(k+1|k)\right)\mathbf{S}_c^{\ -1}(k+1)$$

$$\hat{\mathbf{X}}_c(k+1|k+1)=\hat{\mathbf{X}}_c(k+1|k)+\mathbf{K}_c(k+1)\left[\mathbf{z}_c(k+1)-h_c\left(\hat{\mathbf{X}}_c(k+1|k)\right)\right]$$

$$\mathbf{P}_c(k+1|k+1)=\mathbf{P}_c(k+1|k)-\mathbf{P}_c(k+1|k)\mathbf{H}_c^{\ T}\left(\hat{\mathbf{X}}_c(k+1|k)\right)\mathbf{S}_c^{\ -1}(k+1)\mathbf{H}_c\left(\hat{\mathbf{X}}_c(k+1|k)\right)\mathbf{P}_c(k+1|k)$$

Localization Problem

- **Distributed EKF framework:**
  - Each robot keeps its own state and error covariance matrix estimations
  - Error covariance interdependencies calculated locally
- **Communication requirements:**
  - Predicted error covariance matrix and state vector
  - The measurement vector

Localization Problem

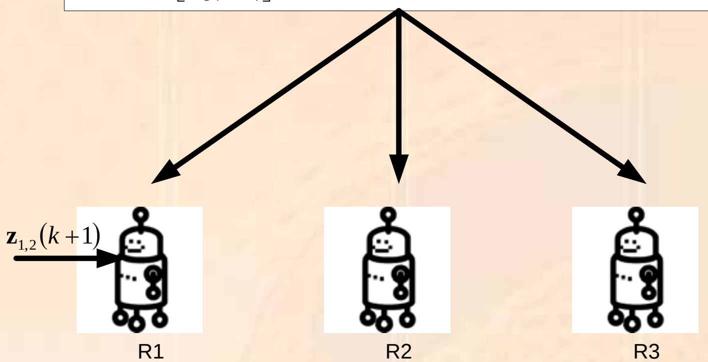
The Centralized EKF Predicted components

$$\hat{\mathbf{X}}_c^-(k+1)=\begin{bmatrix}\hat{\mathbf{X}}_1^-(k+1)\\ \hat{\mathbf{X}}_2^-(k+1)\\ \hat{\mathbf{X}}_3^-(k+1)\end{bmatrix}\qquad \mathbf{P}_c^-(k+1)=\begin{bmatrix}\mathbf{P}_{11}^-(k+1) & \mathbf{P}_{12}^-(k+1) & \mathbf{P}_{13}^-(k+1)\\ \mathbf{P}_{21}^-(k+1) & \mathbf{P}_{22}^-(k+1) & \mathbf{P}_{23}^-(k+1)\\ \mathbf{P}_{31}^-(k+1) & \mathbf{P}_{32}^-(k+1) & \mathbf{P}_{33}^-(k+1)\end{bmatrix}$$

The Centralized EKF Updated components

$$\hat{\mathbf{X}}_c^+(k+1)=\begin{bmatrix}\hat{\mathbf{X}}_1^+(k+1)\\ \hat{\mathbf{X}}_2^+(k+1)\\ \hat{\mathbf{X}}_3^+(k+1)\end{bmatrix}\qquad \mathbf{P}_c^+(k+1)=\begin{bmatrix}\mathbf{P}_{11}^+(k+1) & \mathbf{P}_{12}^+(k+1) & \mathbf{P}_{13}^+(k+1)\\ \mathbf{P}_{21}^+(k+1) & \mathbf{P}_{22}^+(k+1) & \mathbf{P}_{23}^+(k+1)\\ \mathbf{P}_{31}^+(k+1) & \mathbf{P}_{32}^+(k+1) & \mathbf{P}_{33}^+(k+1)\end{bmatrix}$$

$$\mathbf{K}_c(k+1)=\begin{bmatrix}\mathbf{K}_1(k+1)\\ \mathbf{K}_2(k+1)\\ \mathbf{K}_3(k+1)\end{bmatrix}\qquad \mathbf{S}_{12}(k+1)$$



Prediction	Prediction	Prediction
$\hat{\mathbf{X}}_1^-(k+1)$	$\hat{\mathbf{X}}_2^-(k+1)$	$\hat{\mathbf{X}}_3^-(k+1)$
$\mathbf{P}_{11}^-(k+1)$	$\mathbf{P}_{22}^-(k+1)$	$\mathbf{P}_{33}^-(k+1)$
$\sqrt{\mathbf{P}_{12}^-(k+1)}$	$\sqrt{\mathbf{P}_{21}^-(k+1)}$	$\sqrt{\mathbf{P}_{31}^-(k+1)}$
$\sqrt{\mathbf{P}_{13}^-(k+1)}$	$\sqrt{\mathbf{P}_{23}^-(k+1)}$	$\sqrt{\mathbf{P}_{32}^-(k+1)}$
Update	**Update**	**Update**
$\hat{\mathbf{X}}_1^+(k+1)$	$\hat{\mathbf{X}}_2^+(k+1)$	$\hat{\mathbf{X}}_3^+(k+1)$
$\mathbf{P}_{11}^+(k+1)$	$\mathbf{P}_{22}^+(k+1)$	$\mathbf{P}_{33}^+(k+1)$
$\mathbf{P}_{12}^+(k+1)$	$\mathbf{P}_{21}^+(k+1)$	$\mathbf{P}_{31}^+(k+1)$
$\mathbf{P}_{13}^+(k+1)$	$\mathbf{P}_{23}^+(k+1)$	$\mathbf{P}_{32}^+(k+1)$
$\mathbf{S}_{12}(k+1)$		

*

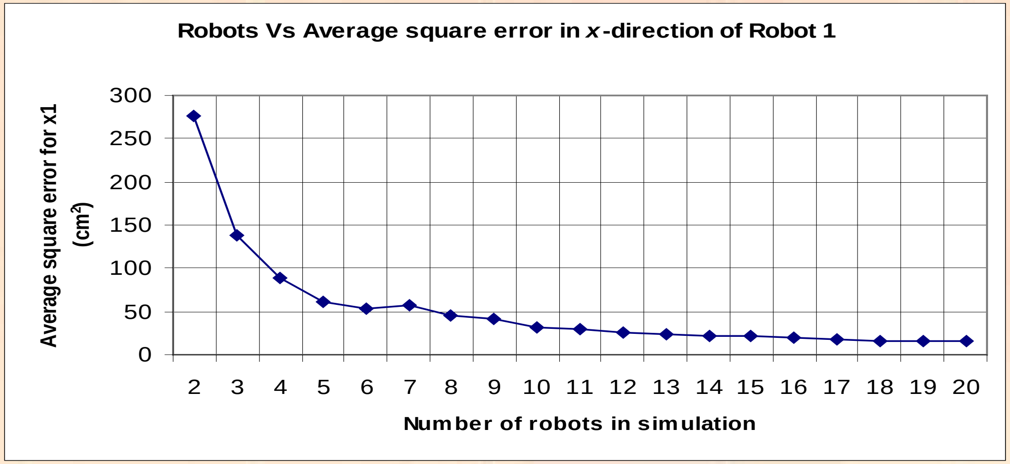
* The full cross-correlation components can be obtained as:

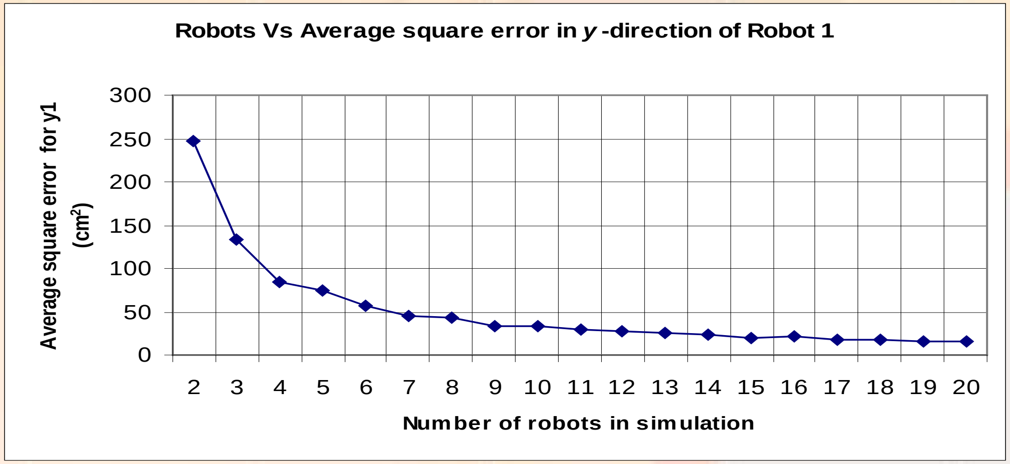
$$\mathbf{P}_{32}^-(k+1)=\sqrt{\mathbf{P}_{32}^-(k+1)}\sqrt{\mathbf{P}_{23}^-(k+1)}^{\,T}$$

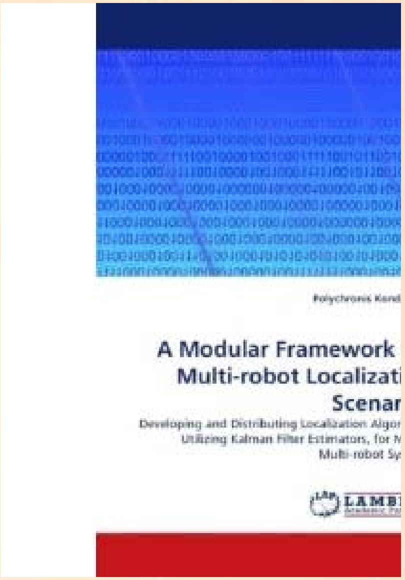
Distributed EKF framework

Localization Problem

Simulated localization results







Kondaxakis P., “A Modular Framework for Multi-robot Localization Scenarios: Developing and Distributing Localization Algorithms Utilizing Kalman Filter Estimators, for Mobile Multi-Robot Systems” *Lambert Academic Publishing*, ISBN: 978-3-8383-7481-9, 2010.

Contents

Target Tracking Problem

FORTH ICS

- **Target Tracking: Estimating the Trajectories of Moving Objects**
  - Using Laser Range Scanners
  - Using Digital Camera Arrangements
- **Why Target Tracking is necessary in Robotics?**
  - Improve their navigation capabilities in dynamic environments
  - Enhance the performance of map-building algorithms
  - Allow the implementation of sophisticated man-machine interaction behaviors

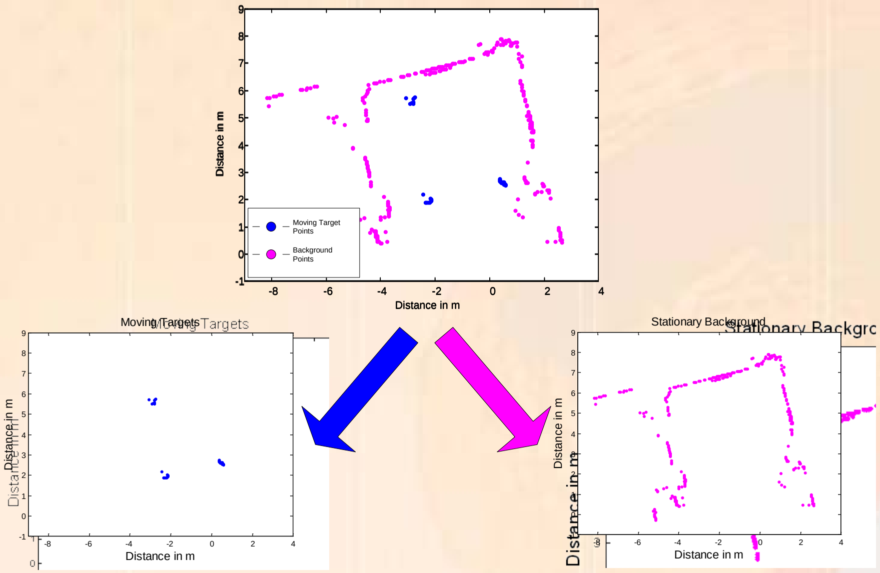
Target Tracking Problem

- **Estimating trajectories of moving objects**
  - Input from a laser range scanner
  - Mobile platform
- **Inherent difficulties**
  - Difficulty to maintain a consistent background model on a moving platform
  - Lack of models for moving objects gives rise to difficulties in:
    - Track initiation
    - Outlier detection/elimination
    - Object shape classification

Target Tracking Problem

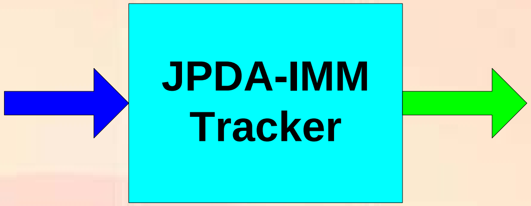
Problem: How can I detect and track moving objects in Dynamic environments?

DETECT



TRACK

$$\mathbf{Z}_{mov}(k) = \begin{bmatrix} \mathbf{z}_1(k) \\ \mathbf{z}_2(k) \\ \vdots \\ \mathbf{z}_s(k) \end{bmatrix}$$



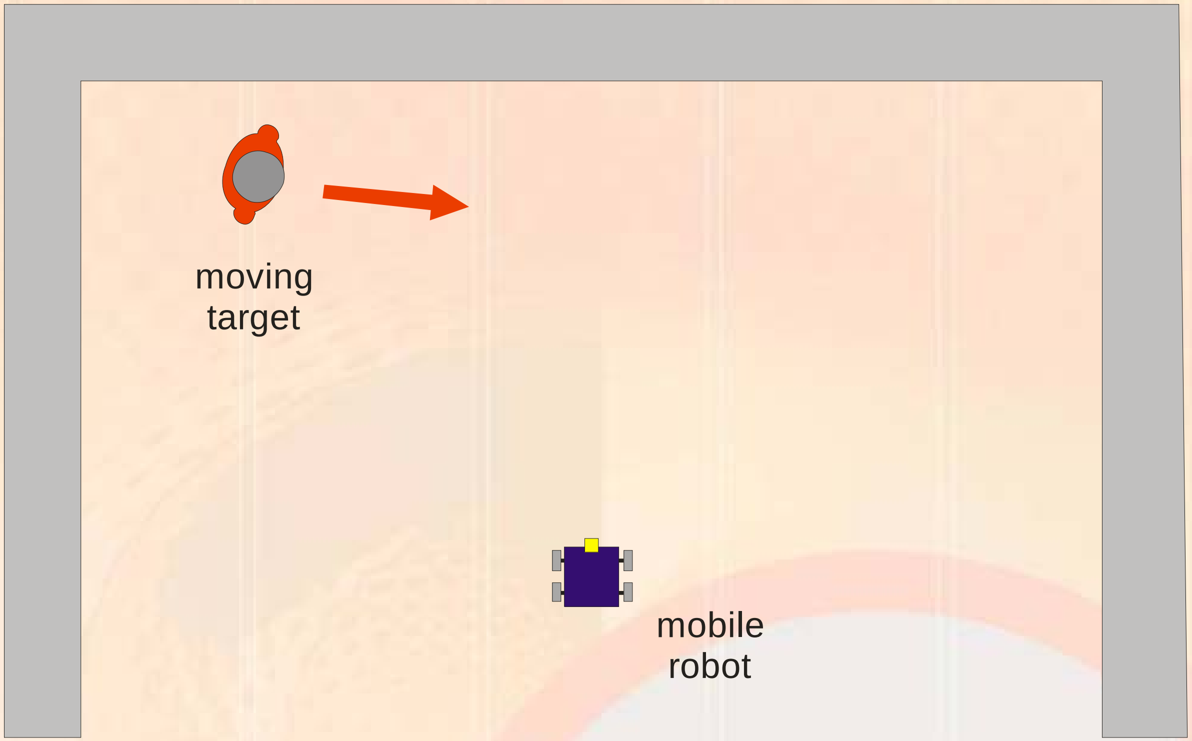
$$\hat{\mathbf{X}}_1(k|k) = \begin{bmatrix} \hat{x}_1(k|k) \\ \hat{v}_{x1}(k|k) \\ \hat{y}_1(k|k) \\ \hat{v}_{y1}(k|k) \end{bmatrix} \quad \hat{\mathbf{X}}_2(k|k) = \begin{bmatrix} \hat{x}_2(k|k) \\ \hat{v}_{x2}(k|k) \\ \hat{y}_2(k|k) \\ \hat{v}_{y2}(k|k) \end{bmatrix} \quad \vdots \quad \hat{\mathbf{X}}_n(k|k) = \begin{bmatrix} \hat{x}_n(k|k) \\ \hat{v}_{xn}(k|k) \\ \hat{y}_n(k|k) \\ \hat{v}_{yn}(k|k) \end{bmatrix}$$

with $\mathbf{z}_l(k) = [r_l(k), \varphi_l(k)]^T$

Background Subtraction

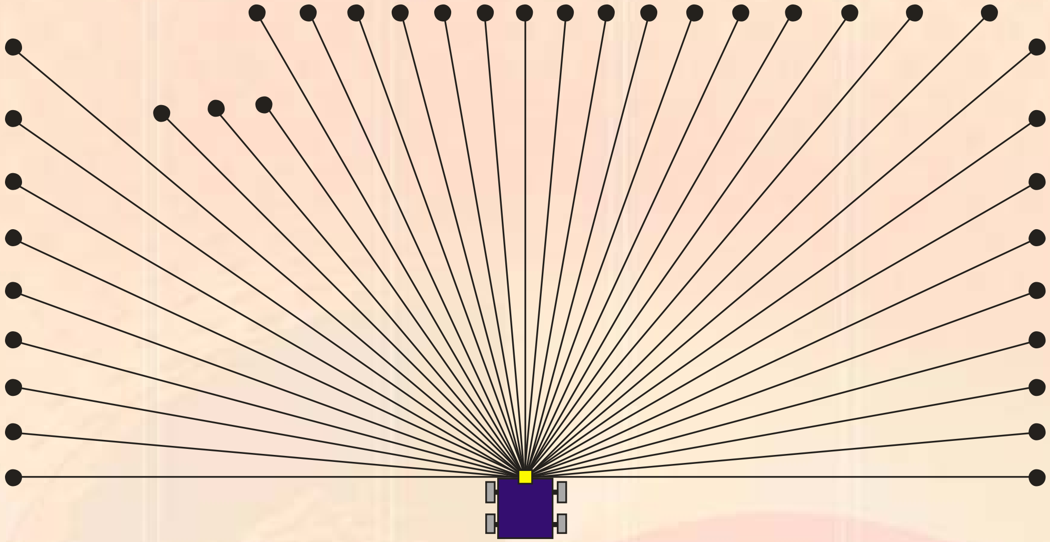
- **Background model is needed to accumulate evidence for moving objects.**
- **Models based on local occupancy grids are prone to:**
  - Quantization errors
  - Drift due to quantization
  - Higher demands for processing power
- **We employ a simple, yet effective background model based on a depth histogram.**

Target Tracking Problem



Assume a mobile robot in an unknown environment

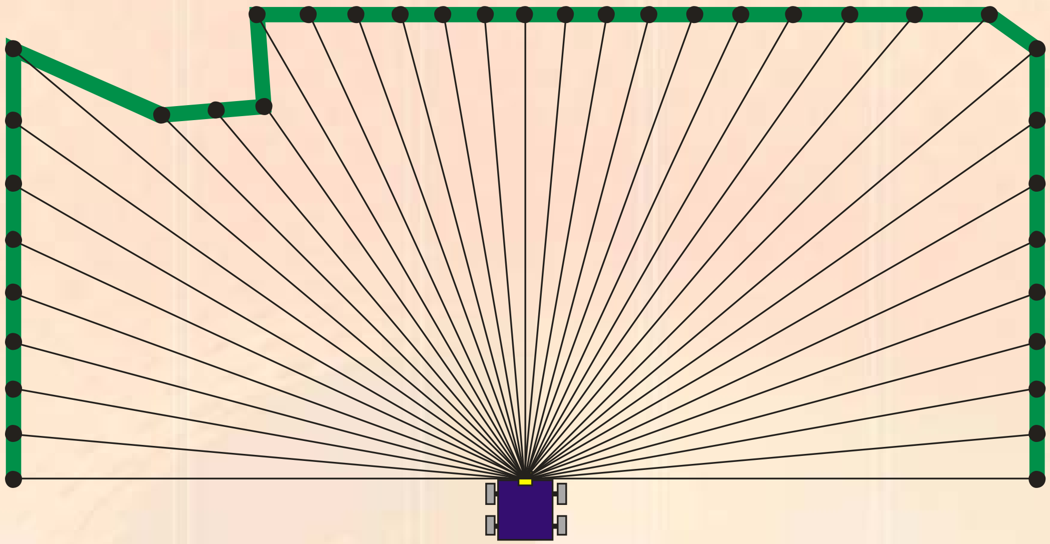
Target Tracking Problem



A 180   laser range scan is captured at time instant t_0.

(Angular resolution 0.5 degrees – 361 measurements total)

Target Tracking Problem



A 361-element vector accumulates background range data

Target Tracking Problem

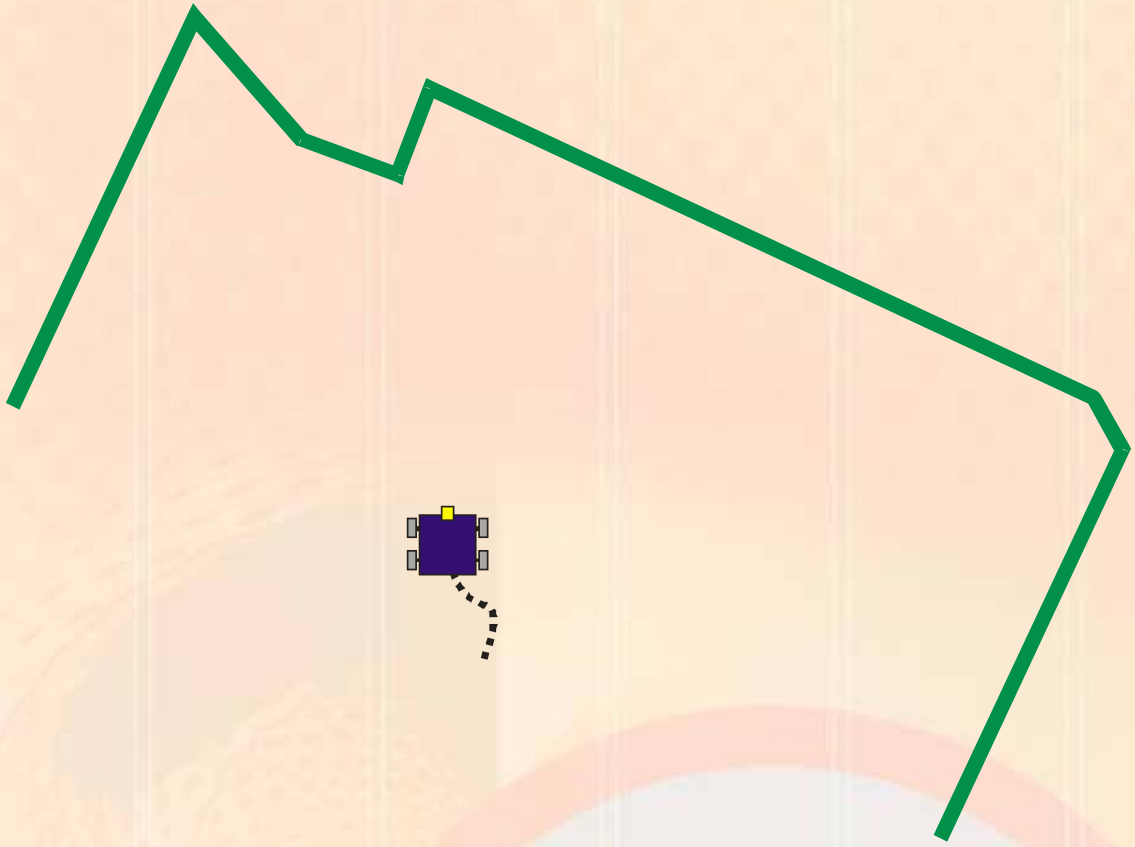
A 361-element vector accumulates background range data

Maximum certainty (line-thickness) is assigned to background model

Target Tracking Problem

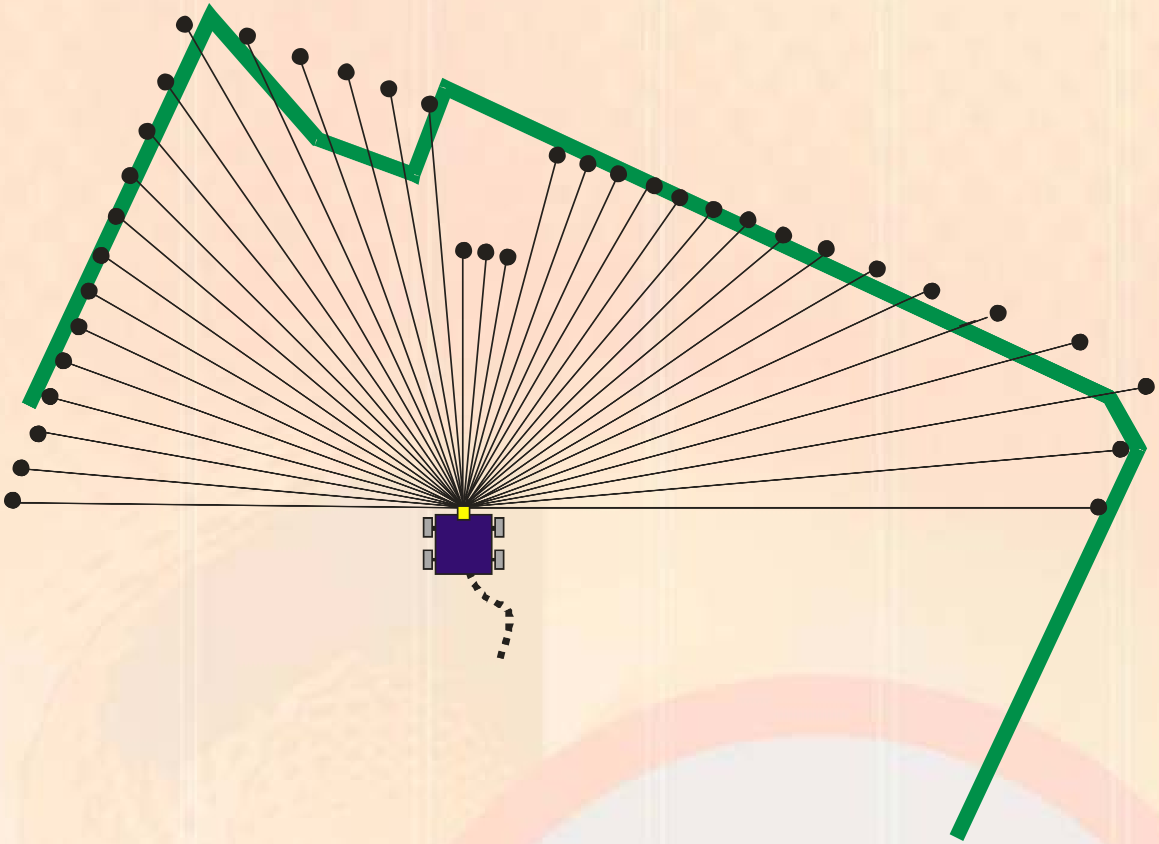
Robot moves...

Target Tracking Problem



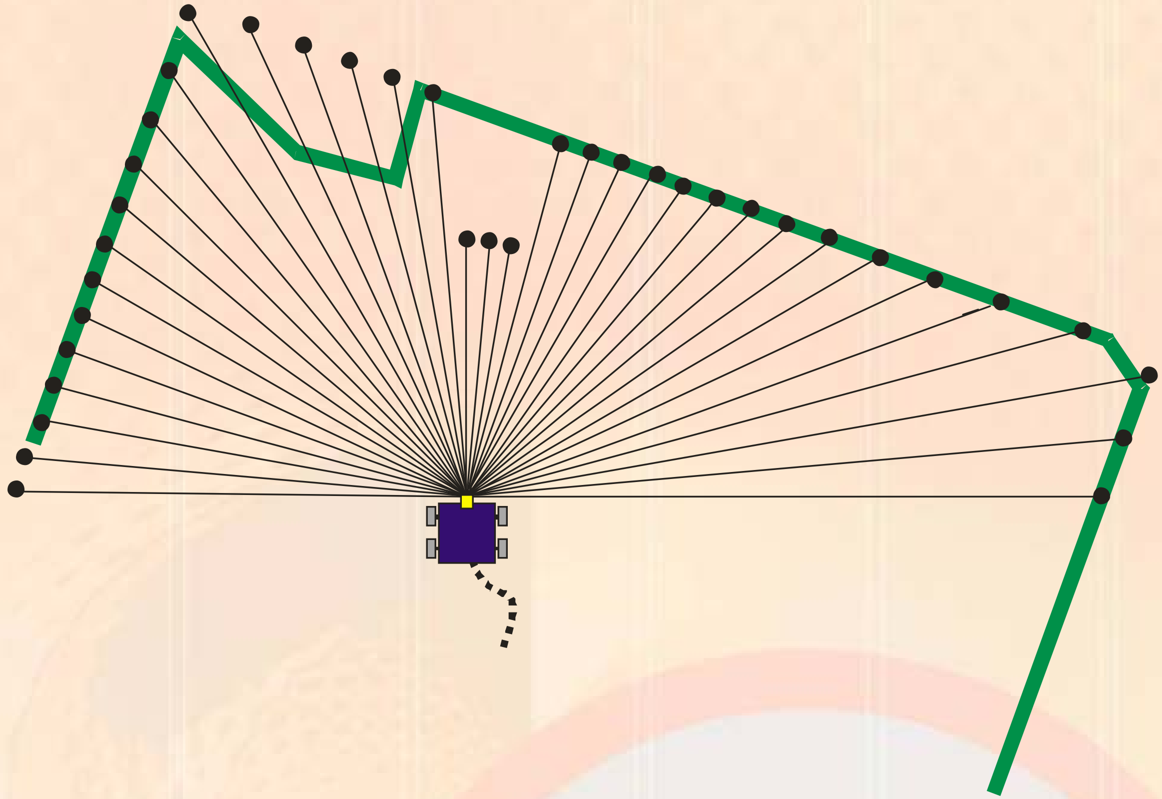
The background model of the previous frame is transformed using odometry

Target Tracking Problem



At time instant t_1, a new laser scan is captured

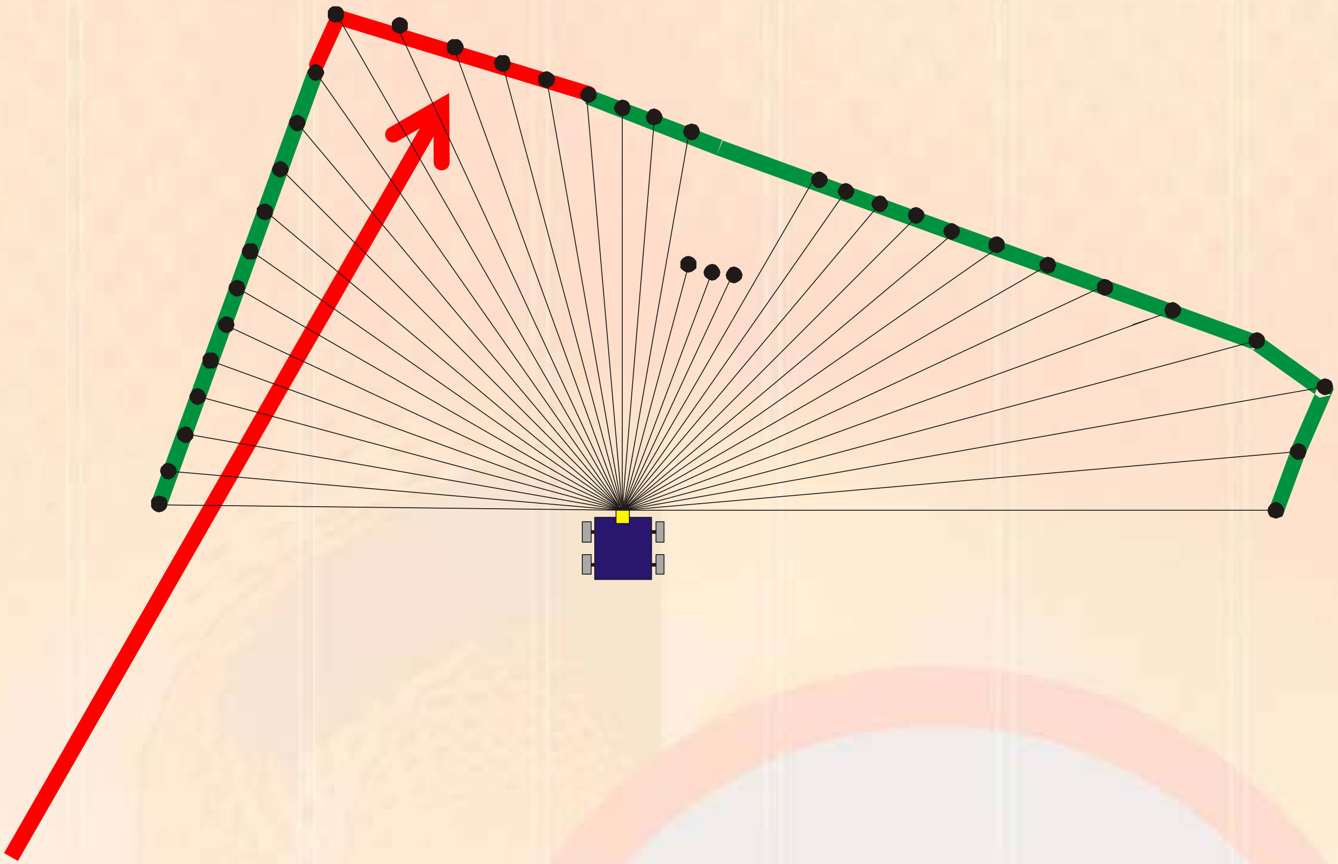
Target Tracking Problem



An ICP algorithm compensates odometry errors

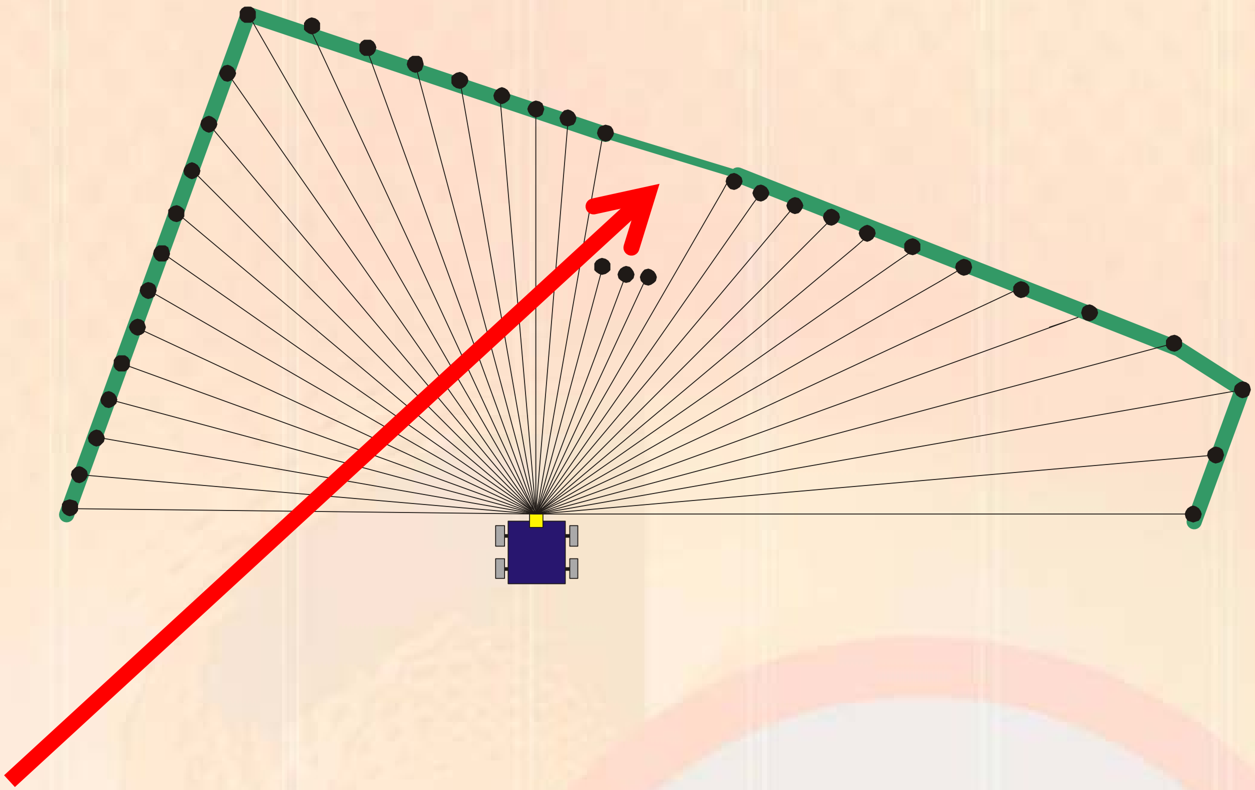
(aligns the background model with the current scan)

Target Tracking Problem



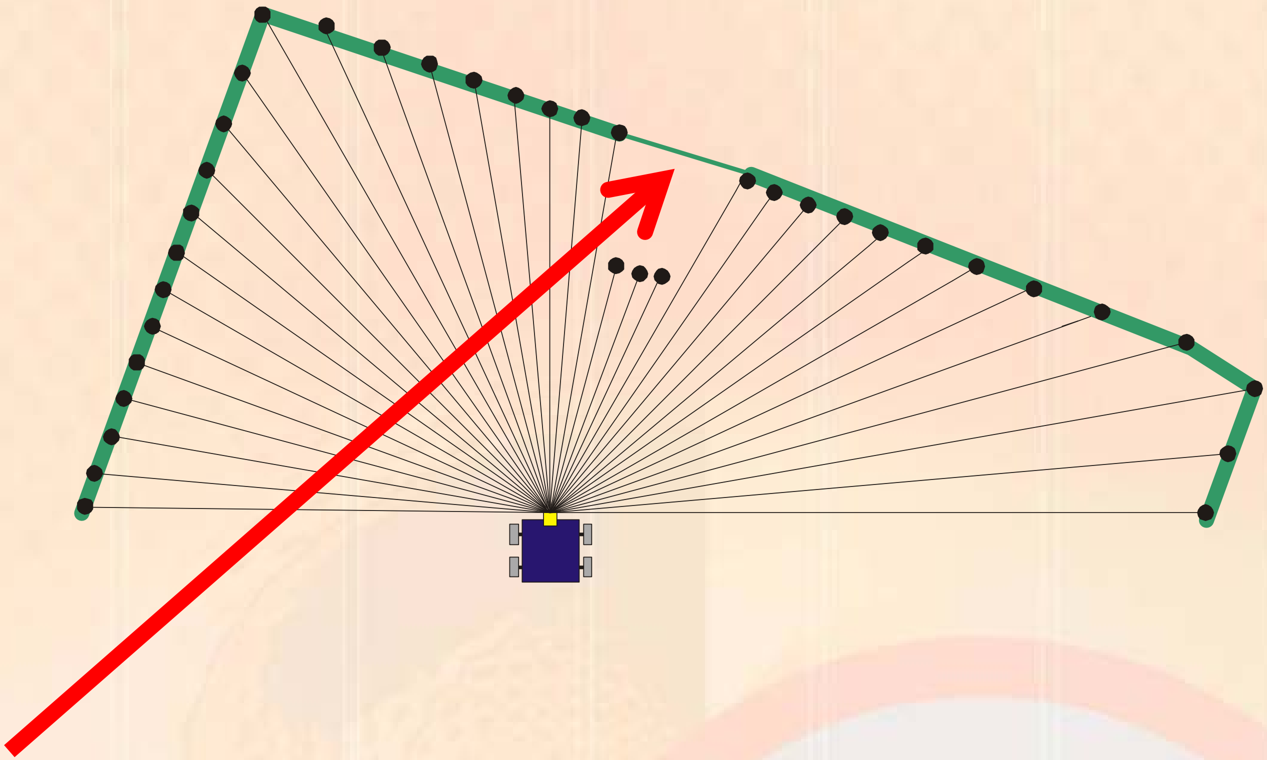
The new background model is obtained by max-distance readings

Target Tracking Problem



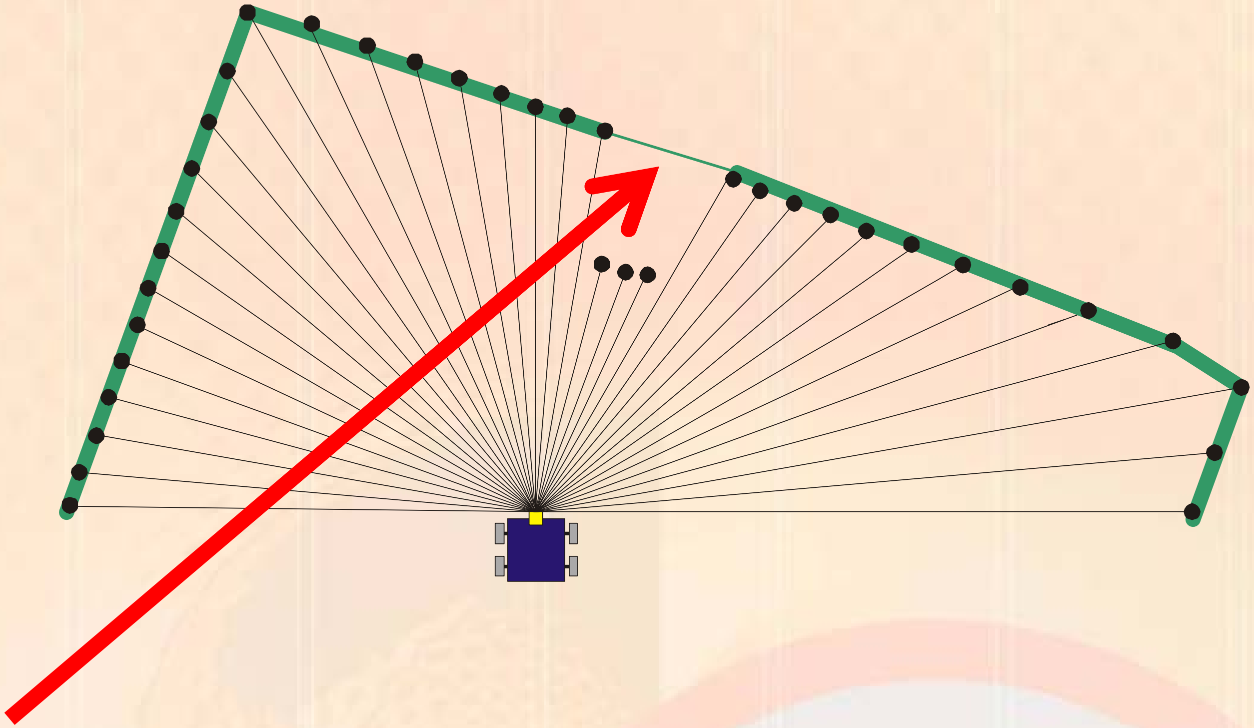
The background model is weighted according to consistency of measurements

Target Tracking Problem



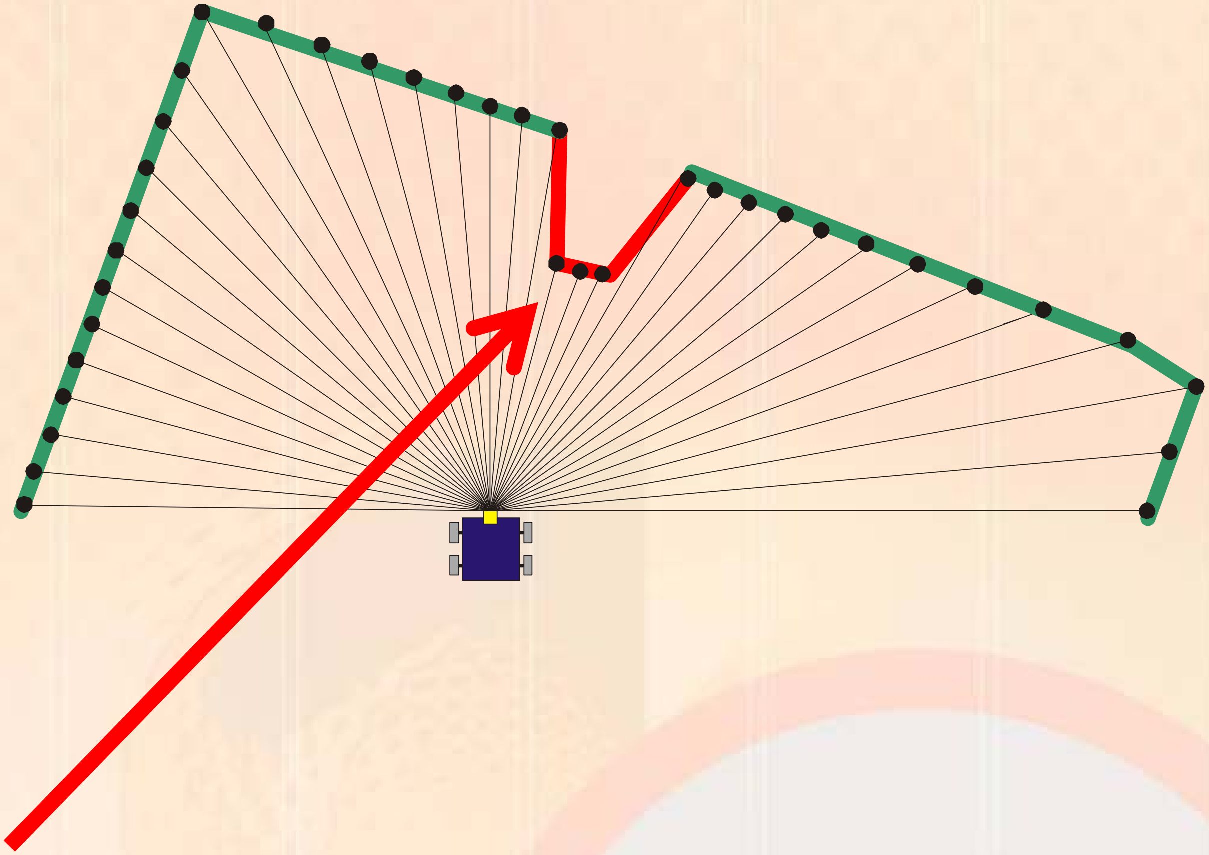
Laser measurements that read further to background model cause local certainty reduction

Target Tracking Problem



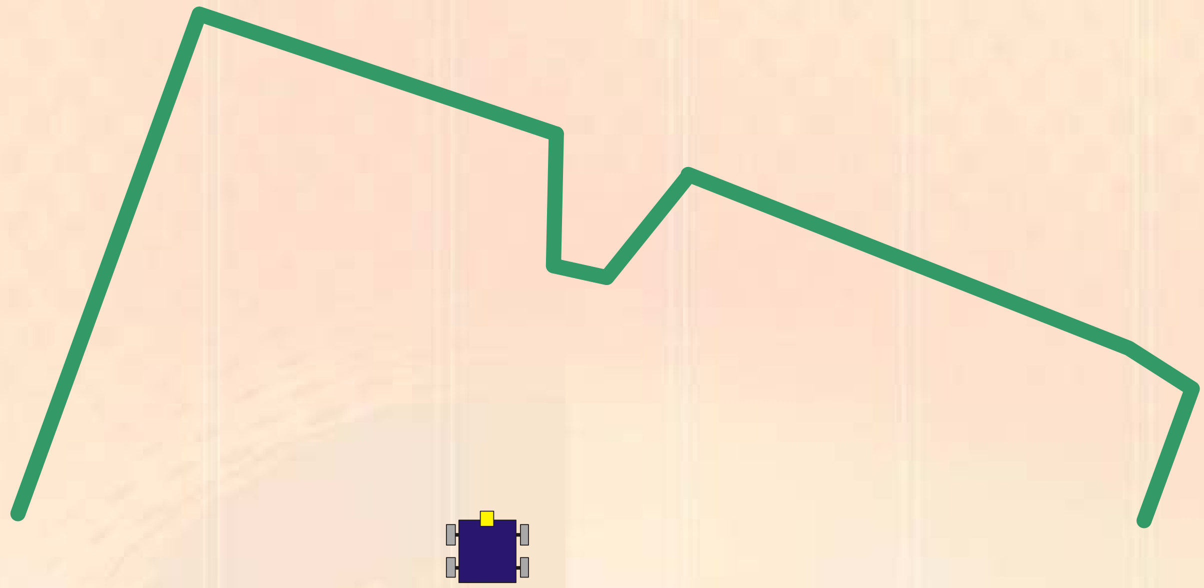
Laser measurements that read further to background model cause local certainty reduction

Target Tracking Problem



Background substitution with current measurements

Target Tracking Problem



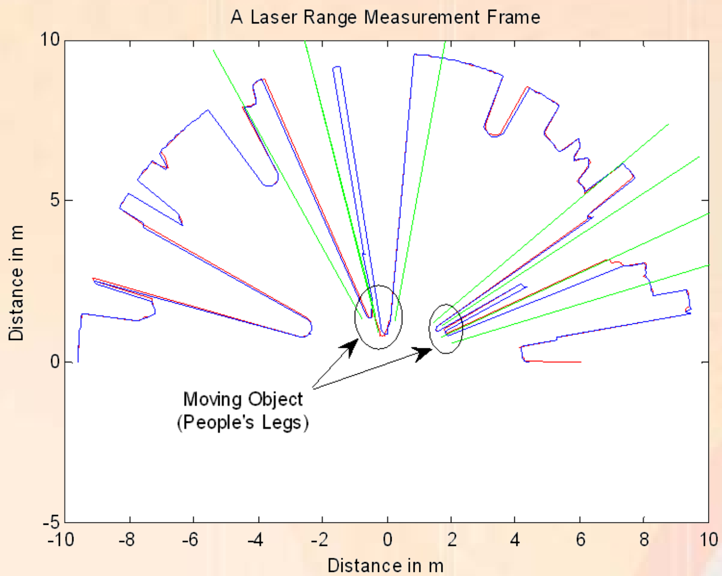
Obtained background

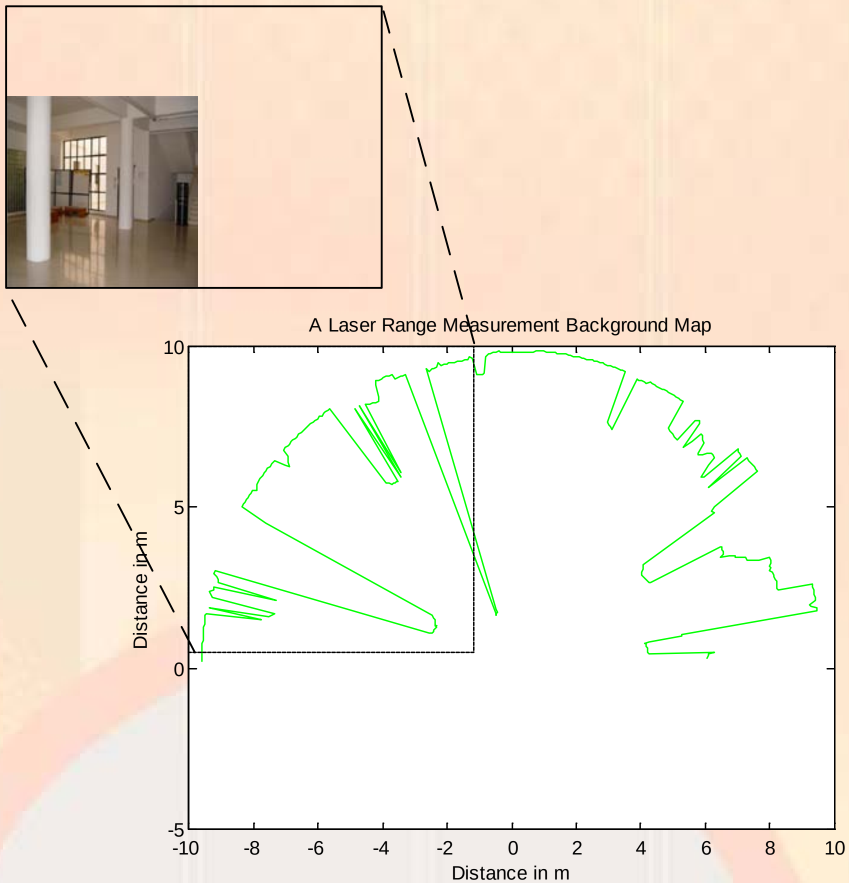
Keeping Background Map Consistency – Algorithmic Formulation

- A "background-weight array" accommodates a consistency belief of the background-map's elements ($\mathbf{w}_{weight}(k) = \{w_1(k), \ldots, w_{361}(k)\}$)
- Weights can take values in range [1, V_{max}] depending on background-point certainty (V_{max} denotes maximum certainty).
  - Weights are initialized at V_{max};
  - Measurements that are further to the ones suggested by the background model result in instant background change.
  - If the weight of a background point reduces to its minimum value this point is substituted with the latest range-reading and its weight is reinitialized to V_{max} .

Target Tracking Problem

Final Result



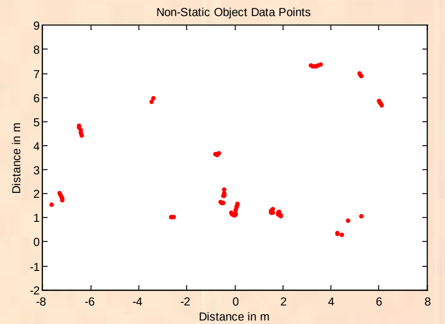


Track Initiation Process

- Compute **clusters of foreground** points
  - Simple geometric clustering using the Euclidean distance between consecutive points
- **Filter clusters** according to two criteria:
  - Cluster size in specified range
  - Clusters should be formed around at least one local depth minimum
- Clusters are **associated across frames** (Munkres algorithm)
- A **two-point track initiation** technique is used to create new tracks for non-stationary clusters

Target Tracking Problem

Available Data form Background Modeling



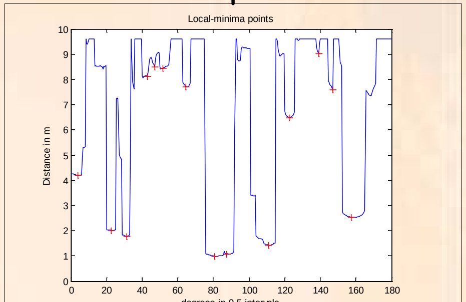
$$\mathbf{Z}^{*}_{mov}(k) = \{\mathbf{z}^{*}_{1}(k), \mathbf{z}^{*}_{2}(k), \ldots, \mathbf{z}^{*}_{s}(k)\}$$

Clustering Process

$$\mathbf{C}_{SEG}(k) = \{\mathbf{c}_1(k), \mathbf{c}_2(k), \ldots, \mathbf{c}_q(k)\}$$

$$\text{with } \mathbf{C}_{SEG}(k) \subseteq \mathbf{Z}^{*}_{mov}(k)$$

$$\tilde{\mathbf{C}}_{SEG}(k) \leftarrow \mathbf{c}_i(k) \quad \text{if} \quad \mathbf{c}_i(k) \cap \mathbf{L}^{*}_{\min}(k)$$

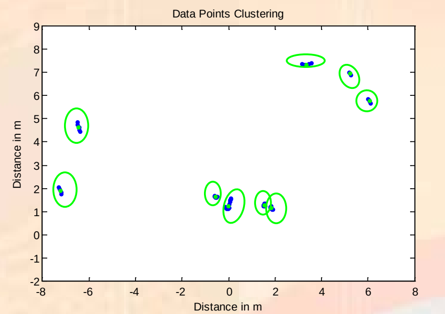


$$\mathbf{L}^{*}_{\min}(k) = \{\mathbf{l}^{*}_{1}(k), \mathbf{l}^{*}_{2}(k), \ldots, \mathbf{l}^{*}_{e \le 361}(k)\}$$

$$\text{with } \mathbf{L}^{*}_{\min}(k) \subseteq \mathbf{Z}^{*}_{laser}(k)$$

Fourteen Local-Minima Points Obtained from Unprocessed Data Frame

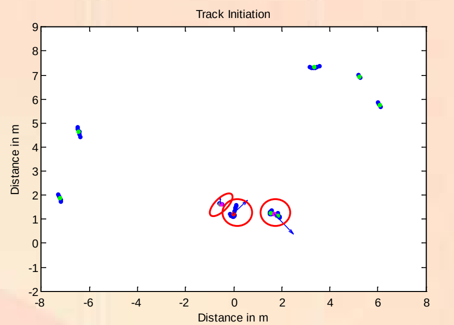
Available Dummy Tracks from Previous Time Instance (*k-1*)



$$\overline{\mathbf{C}}_{SEG}(k) = \{\overline{\mathbf{c}}_1(k), \overline{\mathbf{c}}_2(k), \ldots, \overline{\mathbf{c}}_q(k)\}$$

Nine Clusters Obtained

Two-point Track initiation Process



$$\hat{\mathbf{X}}_{existing}(k|k) = [\hat{\mathbf{x}}_1(k|k), \ldots, \hat{\mathbf{x}}_a(k|k)]$$

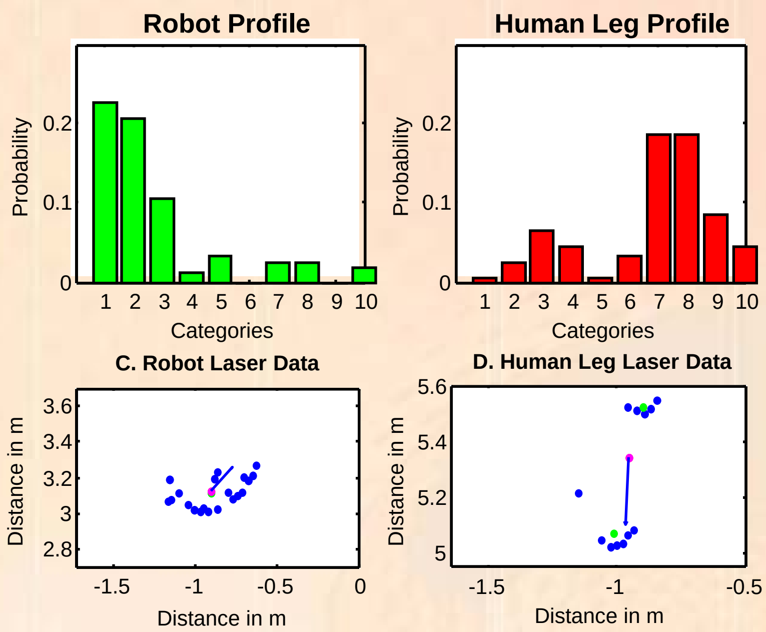
Three Tracks Initiated

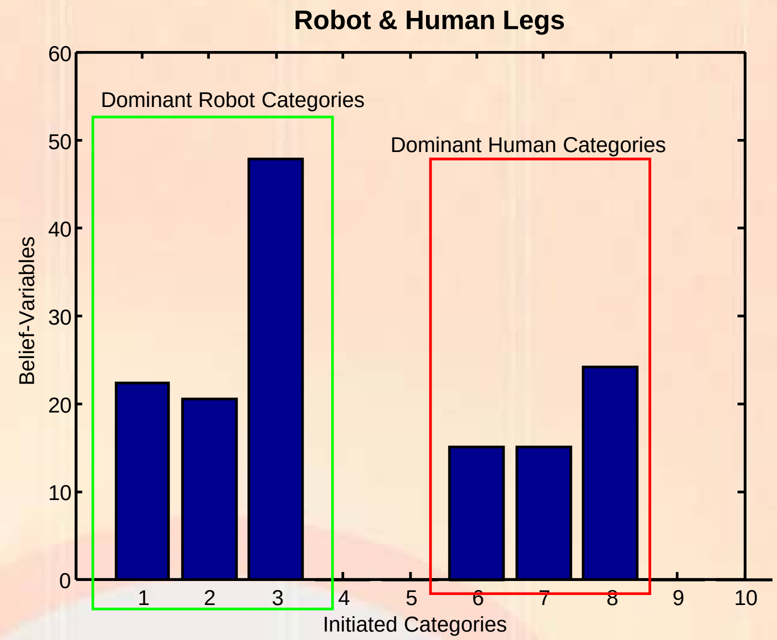
Target Tracking Problem

Moving-Object’s Shape Classification

- The learning/assignment is based on a geometric profile of each available cluster
- An unsupervised Fuzzy ART NN learns in real-time shape classes and simultaneously assigns clusters to learned classes
- Key assumption: The majority of clusters fed to NN represent moving objects
- A voting system that assigns a belief-state to the classes is employed to:
  - Characterize dominant classes
  - Detect –and therefore filter out- outliers

Target Tracking Problem





Target Tracking Problem

- **NEONOTIX ME-470 Robot**
  - Differential drive configuration
  - Max Driving Speed: 0.3m/sec
  - Max Rotational Velocity: 10deg/sec
- **SICK S300 Laser Scanner**
  - Angular Resolution: 0.5
  - Maximum Span: 270 (180 utilized)
  - 541 range readings per frame (361 utilized)
  - Indoor Maximum Range: 9.6m
  - Range Accuracy: 30-70mm
  - One complete range scan every 150ms

- **Pioneer 3-AT Mobile Robot Platform**
  - Skid-Steer drive configuration
  - Maximum Driving Speed: 0.3m/sec
  - Maximum Rotational Velocity: 10deg/sec
- **SICK LMS200 Laser Range Scanner**
  - Angular Resolution: 0.5
  - Maximum Span: 180
  - Delivers 361 range readings per frame
  - Indoor Maximum Range: 9.6m
  - Resolution: 5cm
  - One complete range scan every 200ms

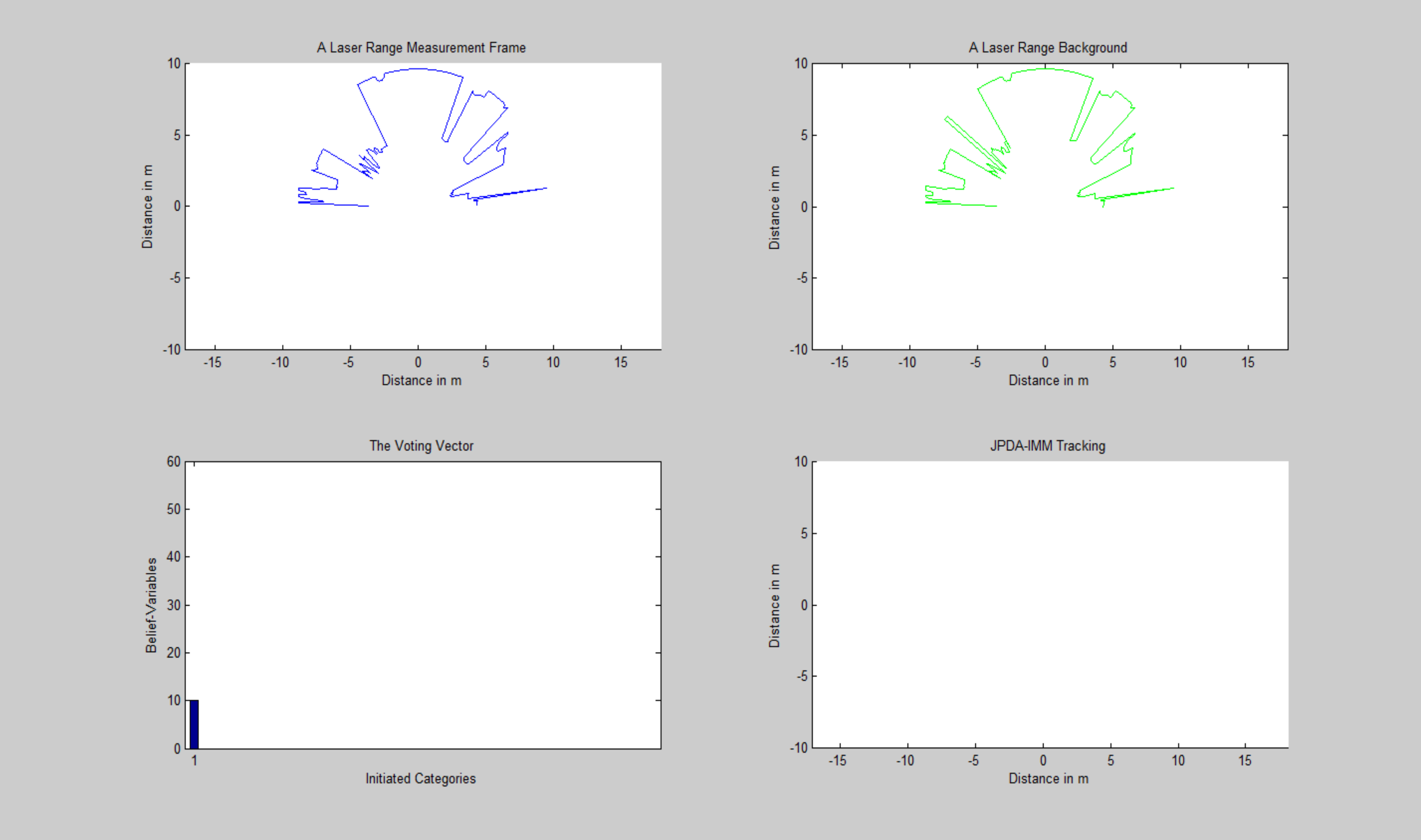
Target Tracking Problem

Result 1: FORTH's foyer

- Robot (moving): Neobotix NE470, Sick S300 laser scanner
- Moving objects: 3 people, 1 pushcart, 1 robotic platform (iRobot B21R)

Target Tracking Problem

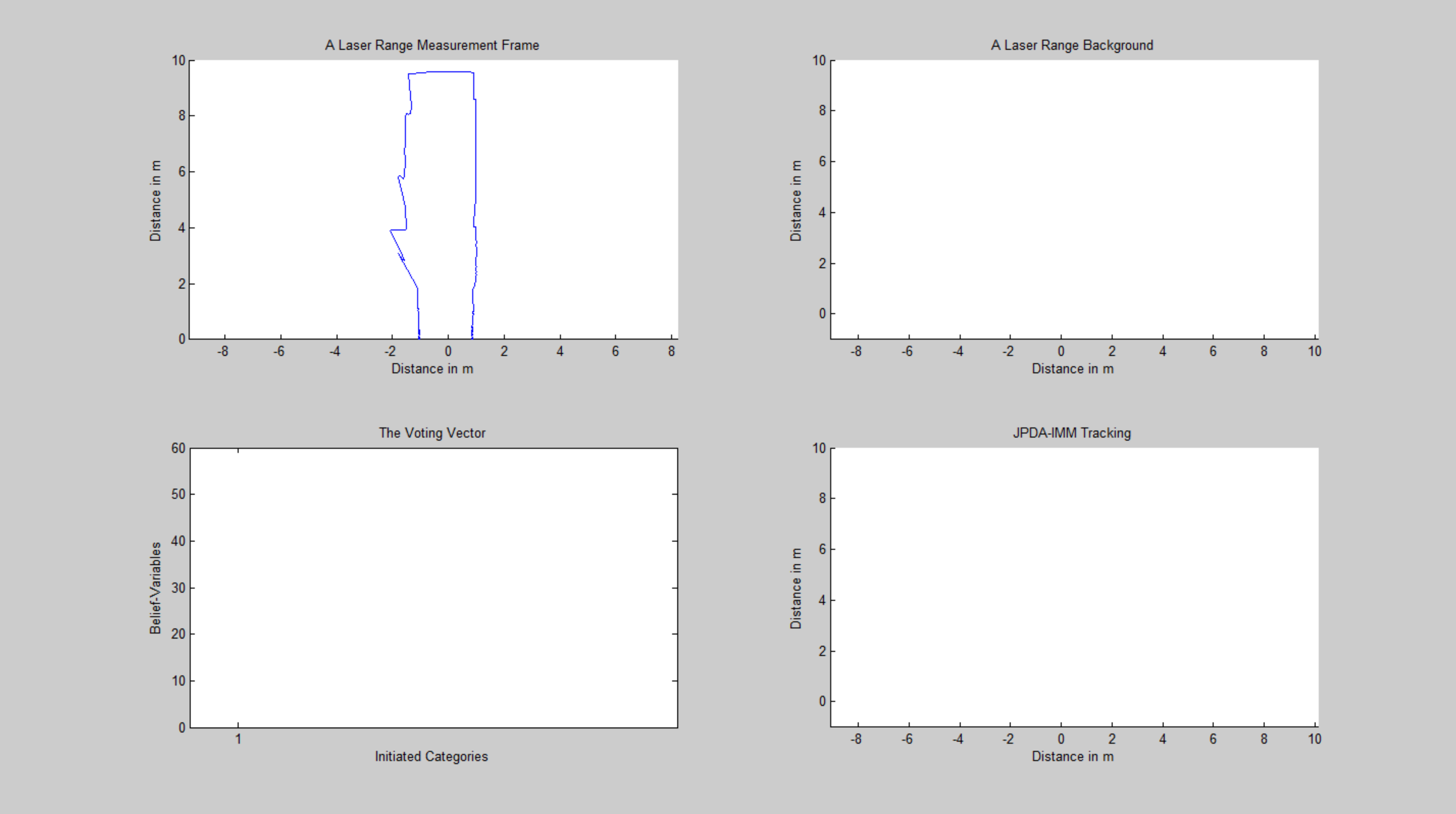
Result 1: FORTH's foyer



Target Tracking Problem

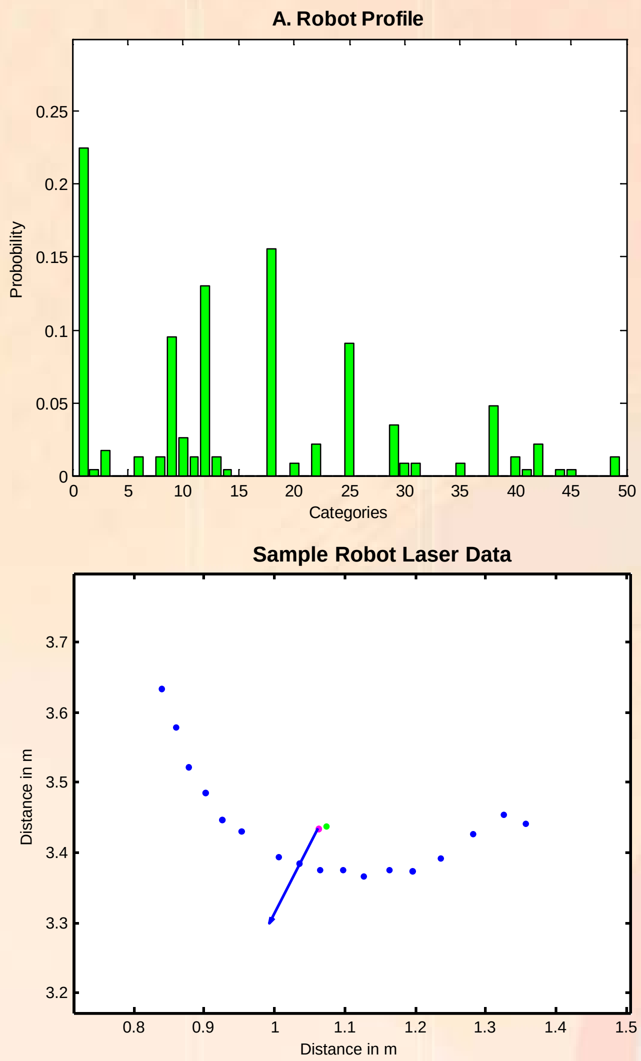
Result 2: corridor area outside our lab

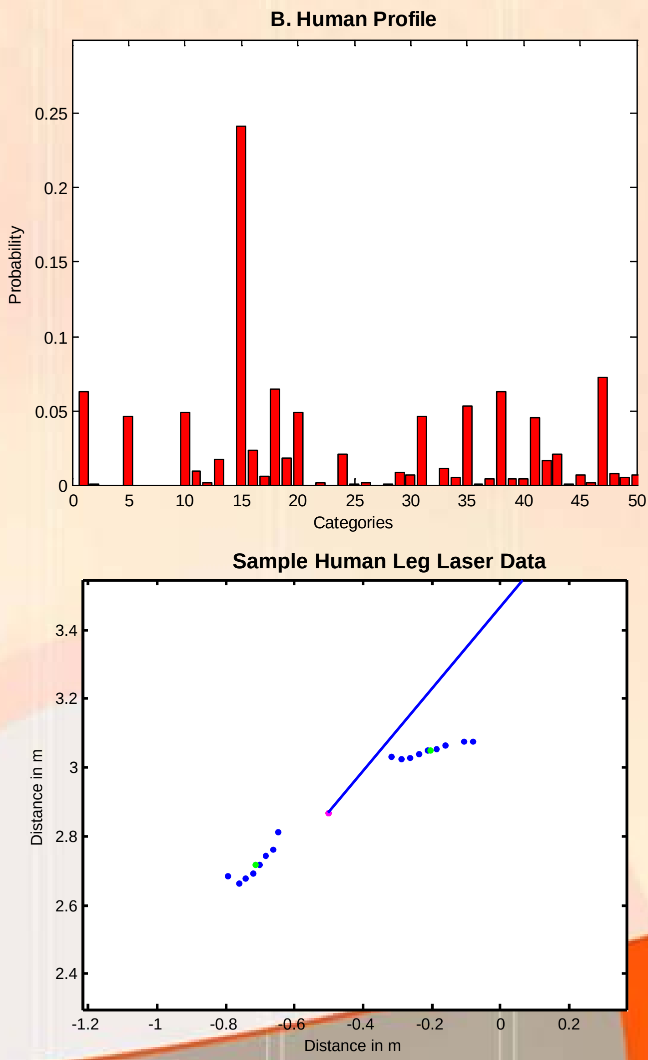
Moving objects: 2 people, 1 robotic platform (additional ME470)



Target Tracking Problem

Shape classification results





Contents

1. Introduction- Who I am!
2. Multi-Robot Localization Problem
3. Com-Loss in Robot Localization
4. Target Tracking Problem
5. Conclusions
6. Future Work

Conclusions

TRACKING

- Proposed method effectively detects and tracks multiple moving objects
- Compensates observer's (Robot) relative movement
- Learns/classifies in real-time the shape of new moving targets
- Moving targets with uncommon shapes are eliminated as target outliers

LOCALIZATION

- The proposed framework offers a modular localization tool
- The c-EKF can be re-arranged to accommodate additional robots
- The system autonomy is increased with the number of robots
- Can be implemented in large number of robots

Contents

1. Introduction- Who I am!
2. Multi-Robot Localization Problem
3. Com-Loss in Robot Localization
4. Target Tracking Problem
5. Conclusions
6. Future Work

Future Work

TRACKING

- Further analysis on track categorization using Fuzzy ART
- Further evaluation of the algorithm's performance using a number of different moving targets
- Implementation/comparison of different classification algorithms

LOCALIZATION

- Obtain a suboptimal solution to com-loss problem
- Implement and test different kind of linearization techniques such as the UKF (Unscented Kalman Filter) for more stable filtering
- Expansion of proposed distributed framework to SLAM scenarios
- Test large number of robots

My Robots!!!!!

Thank you!!!