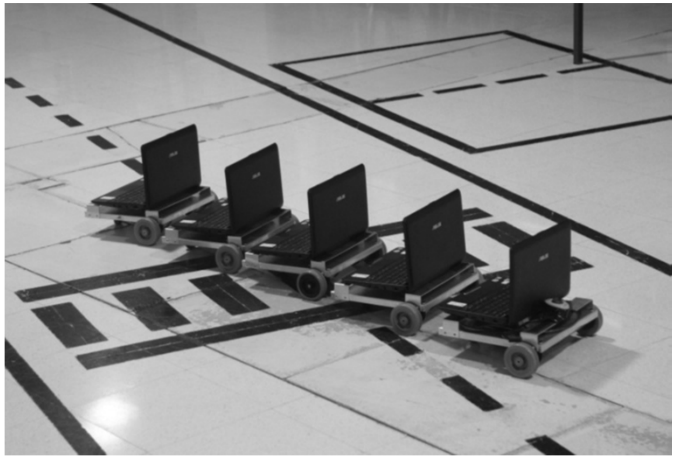
PCA-Based Localization System for Mobile Robots in Unstructured Environments

Fernando Carreira
Advanced Control Systems

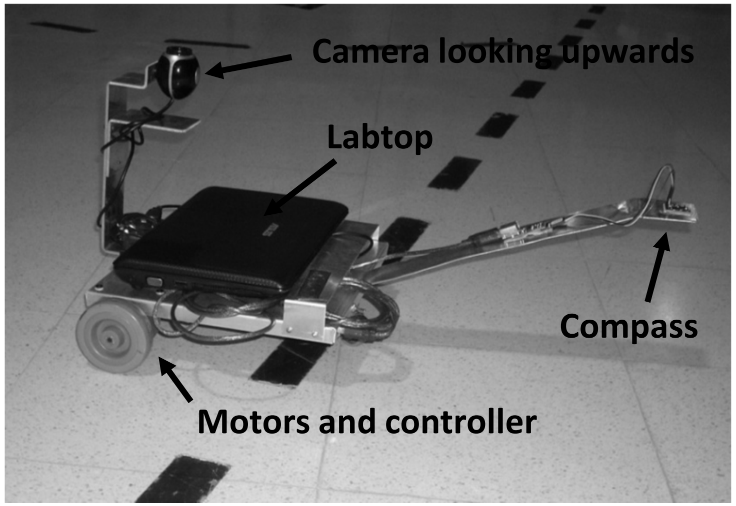
IST, January 7, 2013

INSTITUTO SUPERIOR TÉCNICO
idMEC IST engenharia mecânica
ISR
ISEL INSTITUTO SUPERIOR DE ENGENHARIA DE LISBOA

Motivation



To make **Teams** of mobile robots that **Cooperate** and **Navigate** in **Dynamic Human** populated **Unstructured** environments.



Assumed restrictions:

- Low cost platforms
- Mobile robot moves only with on-board sensors
- No features of environment

PCA-based position sensor

Consider M stochastic signals: $\mathbf{x}_i \in R^N, i = 1, \ldots, M$

Signals stored in array: $\mathbf{X} = [\mathbf{x}_1 \quad \ldots \quad \mathbf{x}_i \quad \ldots \quad \mathbf{x}_M]$

PCA compression data algorithm:

Mean of the captured signals: $$\mathbf{m}_x = \frac{1}{M}\sum_{i=1}^{M}\mathbf{x}_i$$

Covariance matrix: $$\mathbf{R} = \frac{1}{M-1}\sum_{i=1}^{M}(\mathbf{x}_i - \mathbf{m}_x)(\mathbf{x}_i - \mathbf{m}_x)^T$$

Eigenvalues: $\lfloor \boldsymbol{\lambda}_j, \mathbf{U}_j \rfloor = eig(\mathbf{R})$ ⇨ Choose the first $n << N$

During a mission:

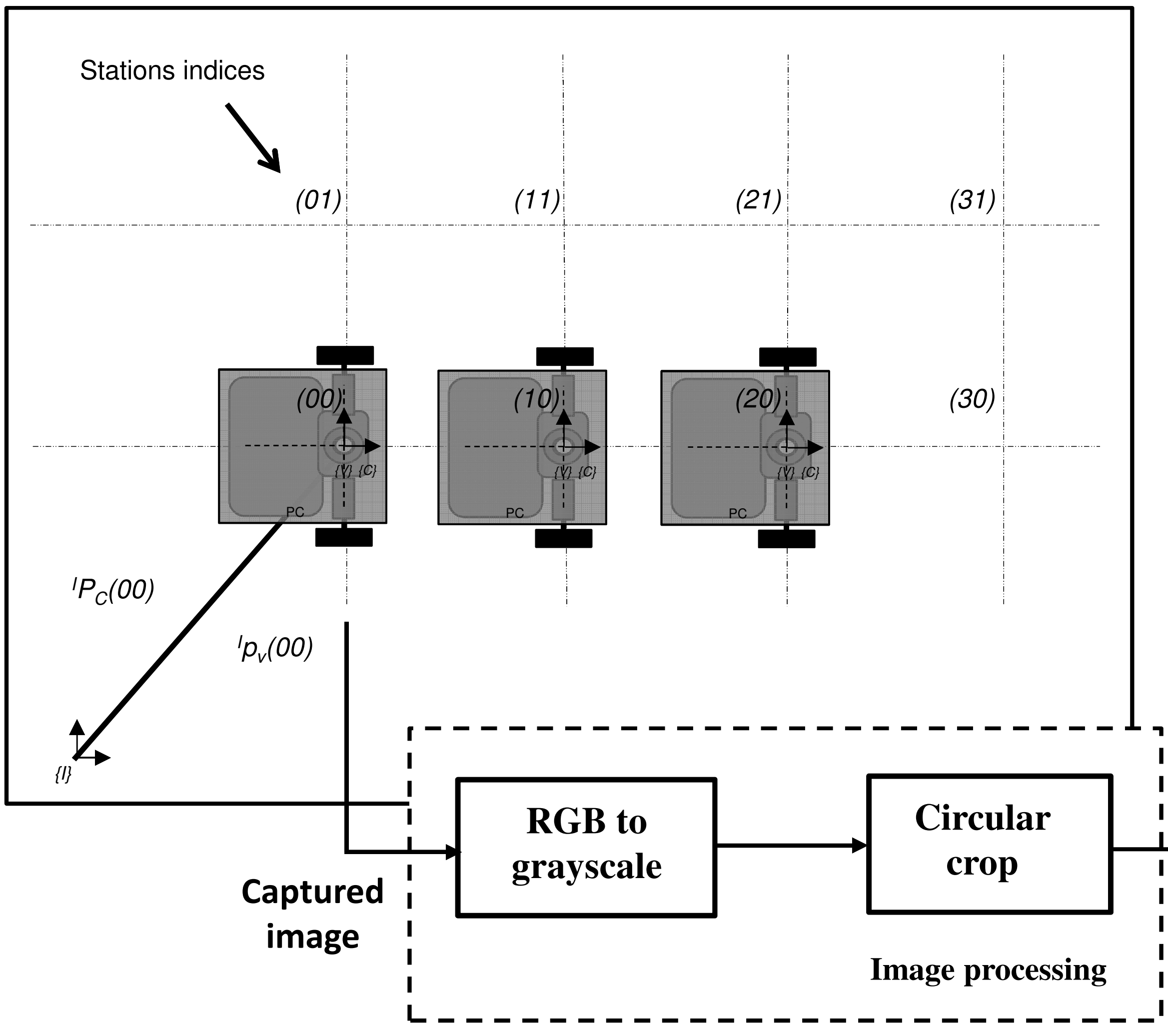
Decomposition of x in orthogonal space: $$\mathbf{v}_i = \mathbf{U}_n^T(\mathbf{x}_i - \mathbf{m}_x)$$

Obtain the image with the closest eigenvector:

$$\forall_i \left\| [\hat{x}, \hat{y}]^T - [x_i, y_i]^T \right\|_2 < \delta, \; r_{PCA} = \min_i \left\| \mathbf{v} - \mathbf{v}_i \right\|_2$$

PCA database

- **Area:** 5 x 4.5m
- **# images:** 125
- **Subsample pixels:** 1/10
- **PCA eigenvectors:** 85%
- **# PCA eigenvectors:** 22



Position estimation

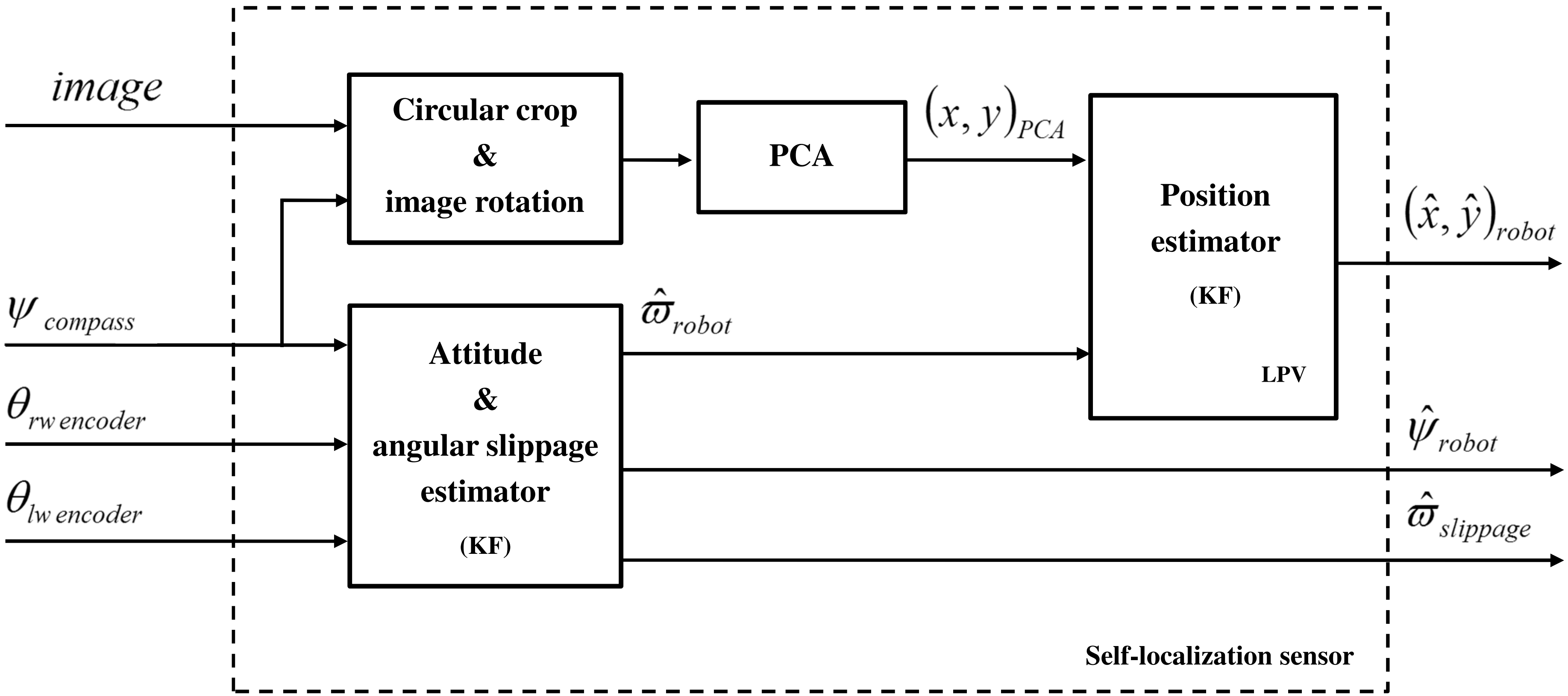
- **Sample time:** 0.2 s
- **Robot velocity:** 0.1 m.s^{-1}

PCA position sensor fused with with Kalman Filter

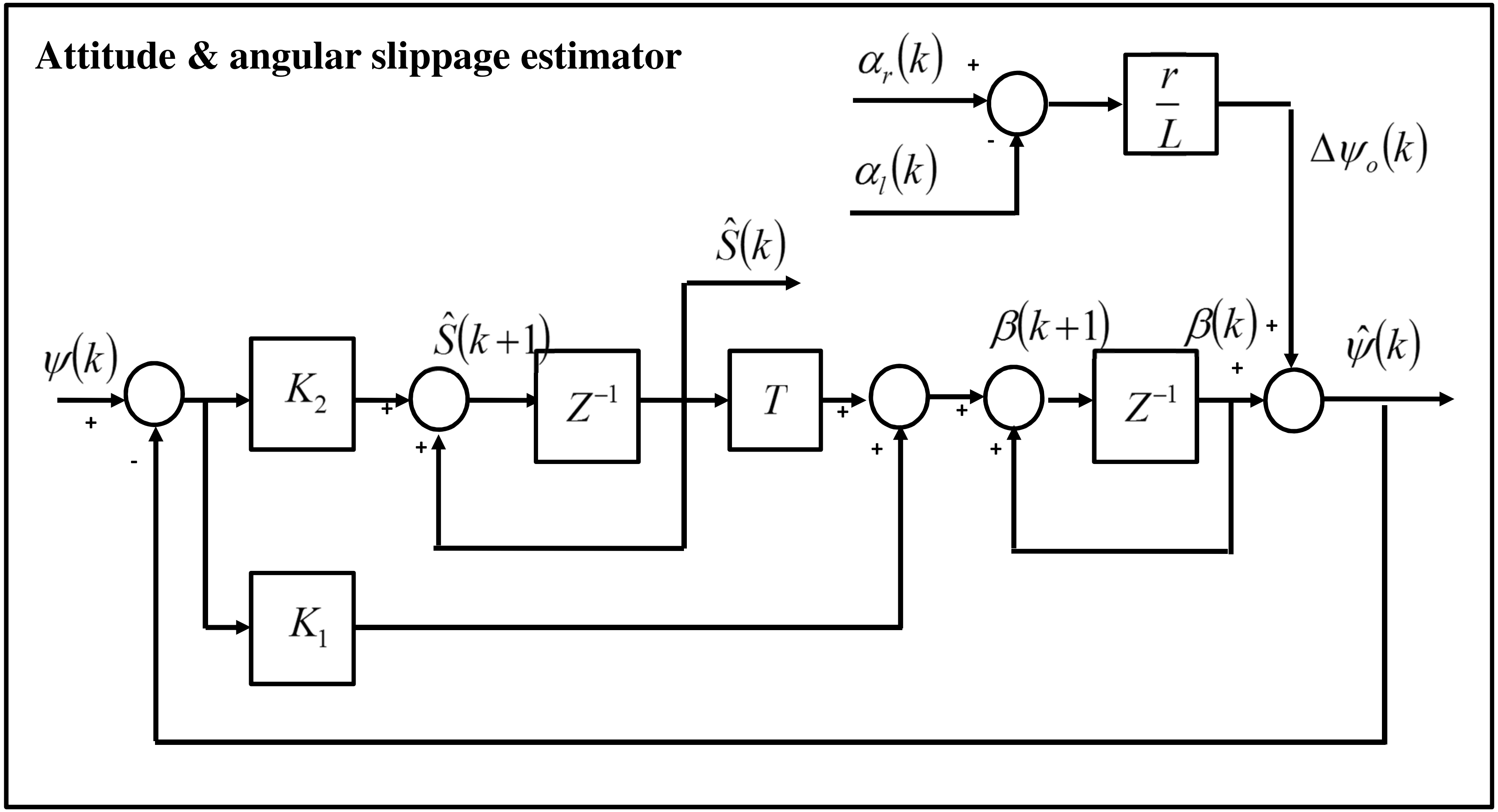
Assumptions:

- Mobile robot is commanded by a digital controller
- Zero Order Hold (ZOH) approximation
- Actuation is constant between two sample times
- Values of the sensors are constant between two sample times

Architecture of the self-localization system



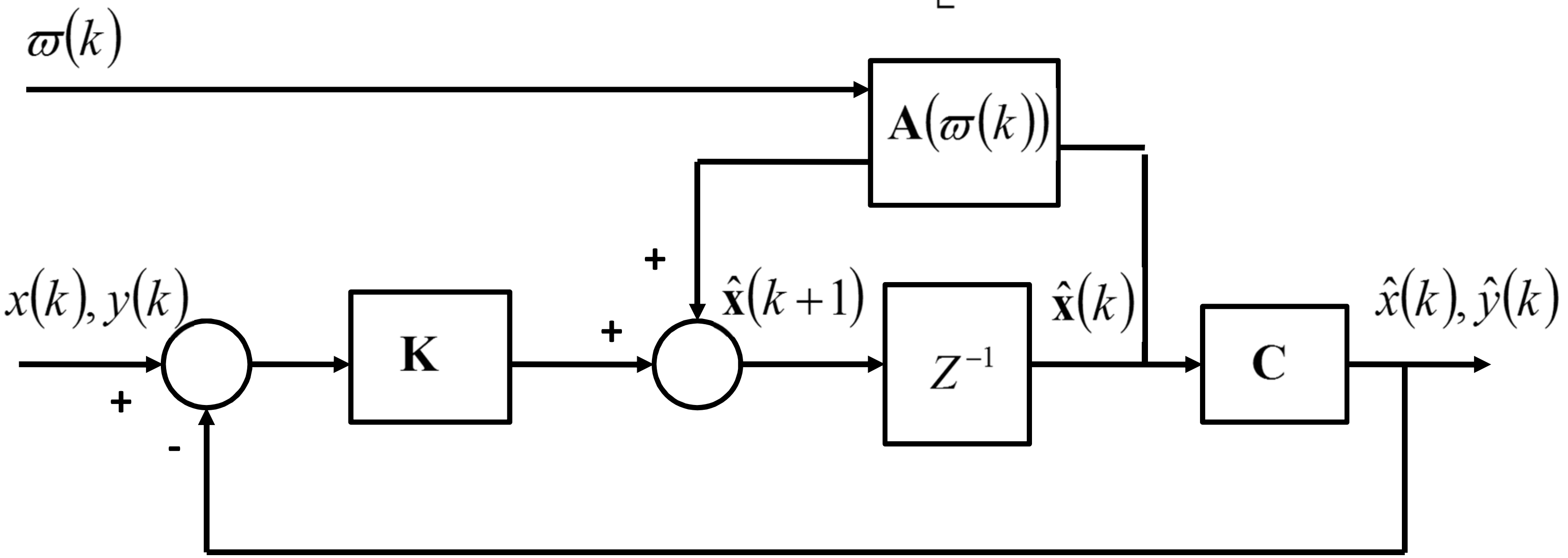
Architecture of the self-localization system



Architecture of the self-localization system

Position estimator

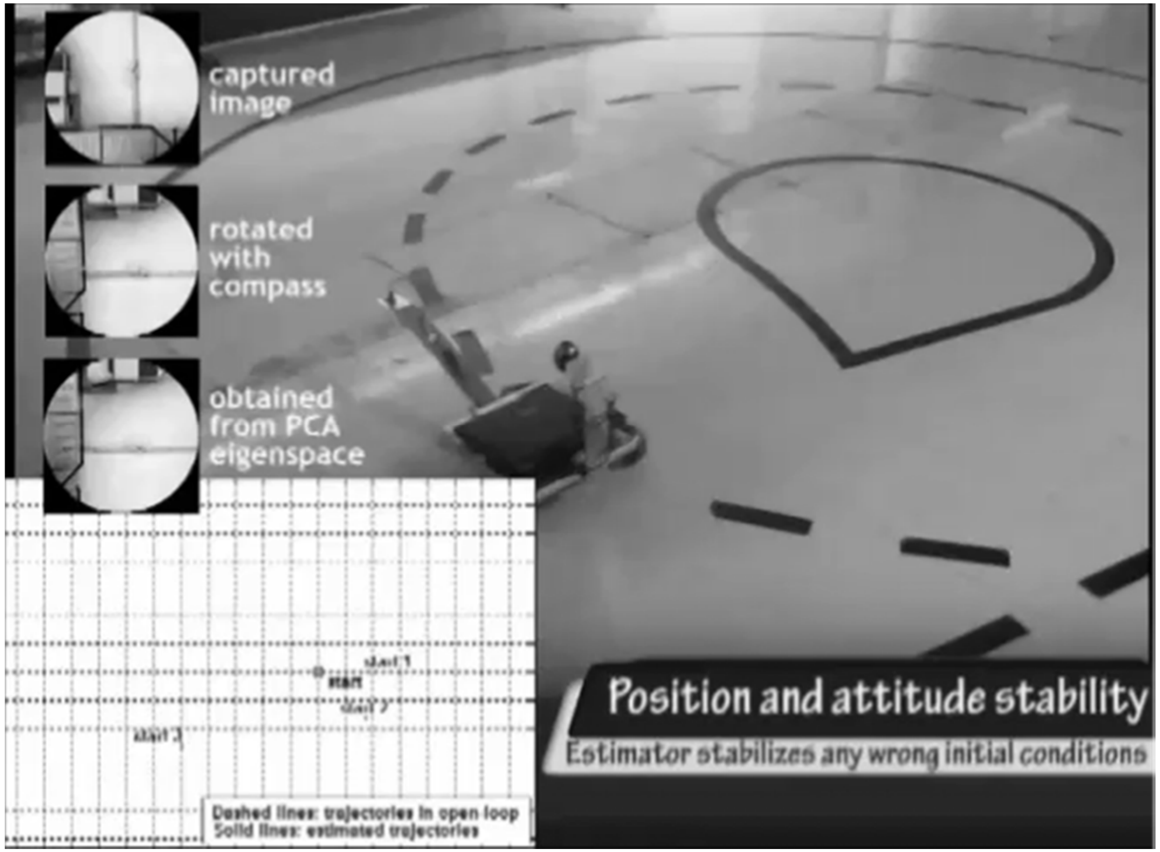
$$\mathbf{A}(\varpi(k)) = \begin{bmatrix} 1 & \dfrac{\sin \varpi T}{\varpi} & 0 & \dfrac{1}{\varpi} + \dfrac{\cos \varpi T}{\varpi} \\ 0 & \cos \varpi T & 0 & -\sin \varpi T \\ 0 & \dfrac{1}{\varpi} - \dfrac{\cos \varpi T}{\varpi} & 1 & \dfrac{\sin \varpi T}{\varpi} \\ 0 & \sin \varpi T & 0 & \cos \varpi T \end{bmatrix}$$



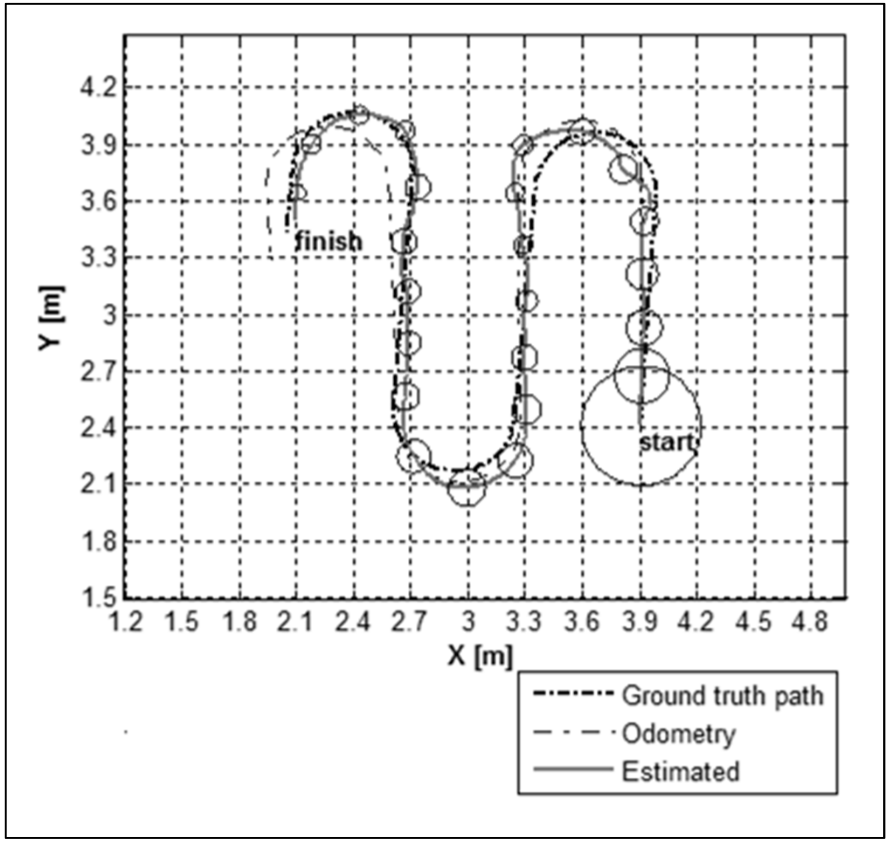
Global position and attitude stability

Estimator starts with different initial positions and attitudes:

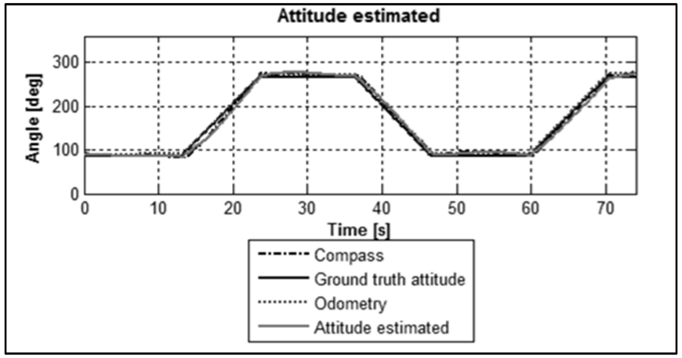
	Δx_0 [m]	Δy_0 [m]	$\Delta \psi_0$ [m]
Test 1	+ 0.5	0	- 90
Test 2	+ 0.5	- 0.5	- 125
Test 3	- 1.5	- 0.8	+ 125



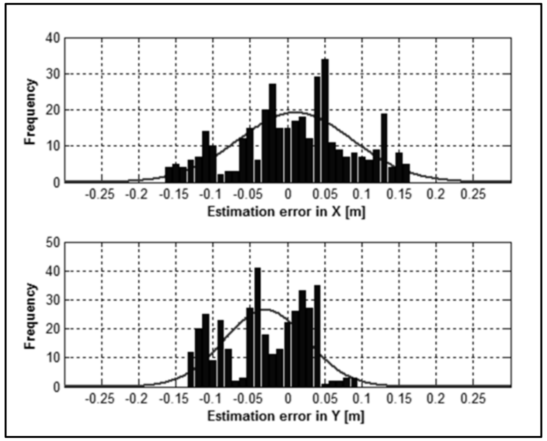
2D localisation results



Map with estimated position

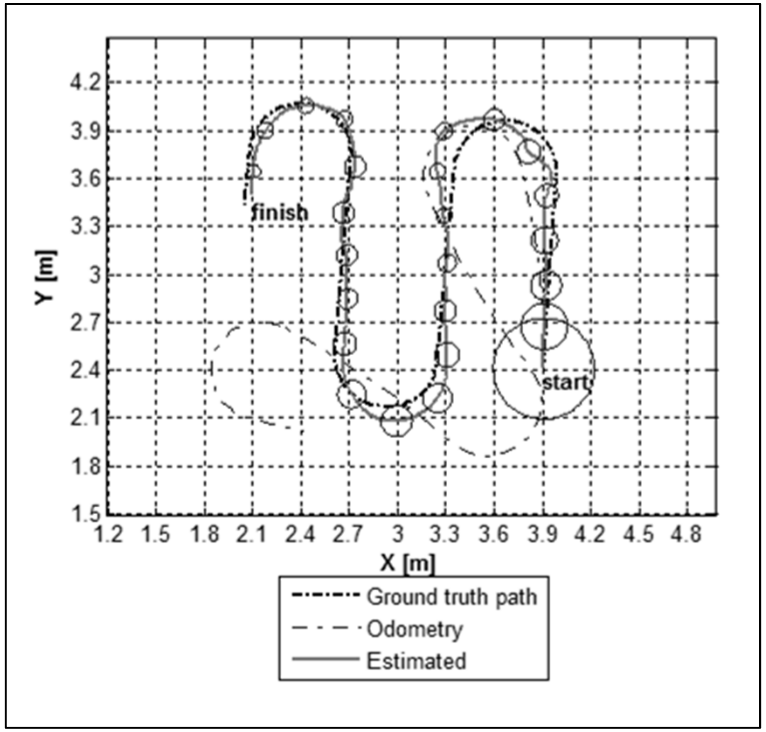


Attitude estimated along time

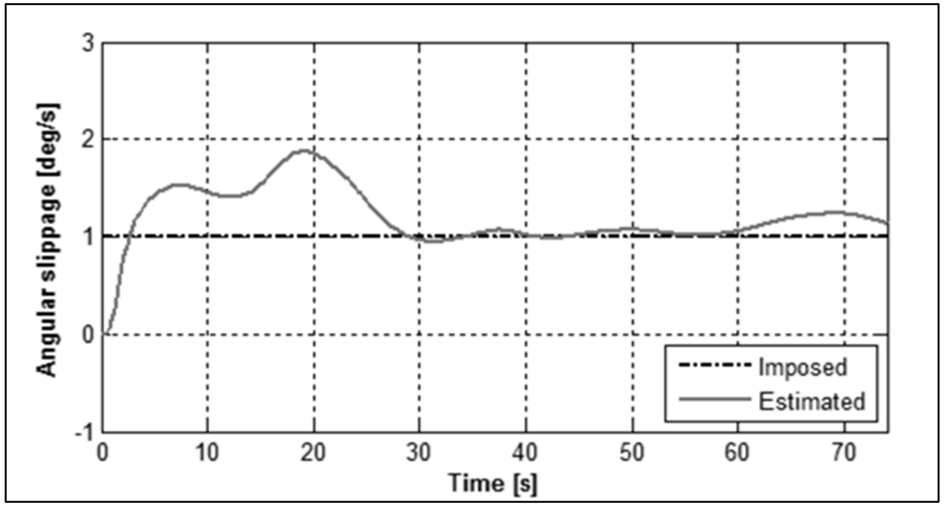


Distribution of estimated position error (PCA position grid with 0.3 m)

2D localization with imposed angular slippage

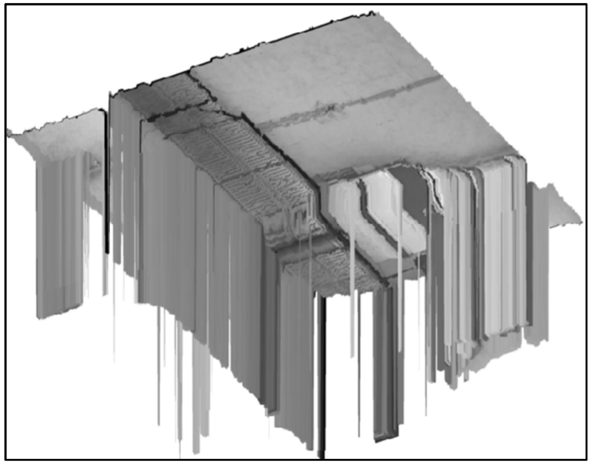


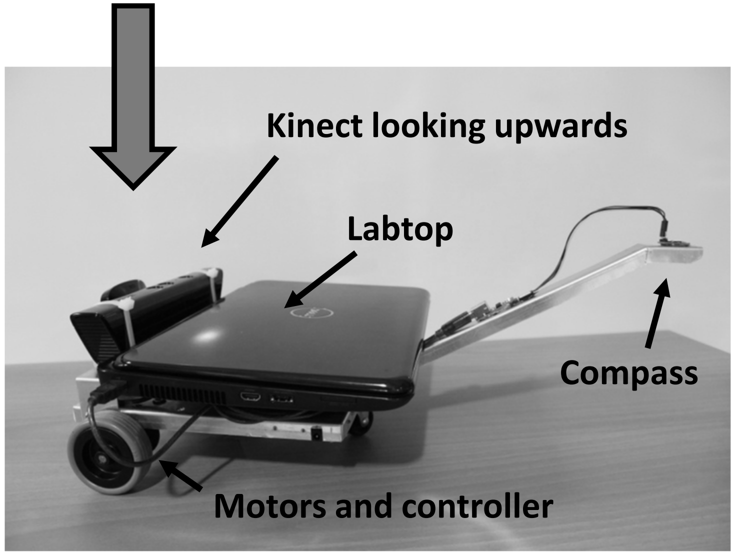
Map with estimated position



Imposed angular slippage estimation along time

RGB-D ceiling vision





Ceiling vision

- Depth signals

Advantage

- Work in any light conditions

Problem

- Signals is often corrupted
  with missing data
  (null distance: 0 *mm*)

PCA for signals with missing data

Consider M stochastic signals: $\mathbf{x}_i \in R^N, i = 1,\ldots,M$

Signals stored in array: $\mathbf{X} = [\mathbf{x}_1 \quad \ldots \quad \mathbf{x}_i \quad \ldots \quad \mathbf{x}_M]$

PCA compression data algorithm:

Binary vector: $$\mathbf{l}_i(j) = \begin{cases} 0 & \text{, if } x_i(j) \text{ is a missing data} \\ 1 & \text{, if } x_i(j) \text{ is a right signal} \end{cases}$$

Auxiliary counters: $\mathbf{c} = \sum_{i=1}^{M} \mathbf{l}_i$ and $\mathbf{C} = \sum_{i=1}^{M} \mathbf{l}_i \mathbf{l}_i^T$

Mean of the captured signals: $$\mathbf{m}_x(j) = \frac{1}{c(j)} \sum_{i=1}^{M} \mathbf{l}_i(j) \cdot \mathbf{x}_i(j) \ , \qquad j = 1,\ldots,N$$

Covariance matrix: $$\mathbf{R}(j,k) = \frac{1}{\mathbf{C}(j,k)-1} \sum_{i=1}^{M} \mathbf{l}_i(j)\mathbf{l}_i(k)(\mathbf{x}_i(j)-\mathbf{m}_x)(\mathbf{x}_i(k)-\mathbf{m}_x)$$
$$j,k = 1,\ldots,N$$

Eigenvalues: $[\boldsymbol{\lambda}_j, \mathbf{U}_j] = eig(\mathbf{R})$ $\Rightarrow$ Choose the first $n << N$

PCA for signals with missing data

Consider M stochastic signals: $\mathbf{x}_i \in R^N, i = 1,\ldots,M$

During a mission:

Mean substitution: $\mathbf{x}_i(j) = \begin{cases} \mathbf{m}_x(j) & \text{, if } x_i(j) \text{ is a missing data} \\ \mathbf{x}_i(j) & \text{, if } x_i(j) \text{ is a right signal} \end{cases}$

Decomposition of x in orthogonal space: $\mathbf{v}_i = \mathbf{U}_n^T(\mathbf{x}_i - \mathbf{m}_x)$

Obtain the image with the closest eigenvector:

$$\forall_i \left\| [\hat{x}, \hat{y}]^T - [x_i, y_i]^T \right\|_2 < \delta,\ r_{PCA} = \min_i \left\| \mathbf{v} - \mathbf{v}_i \right\|_2$$

$[x_i, y_i]^T$ **is selected has the mobile robot position**

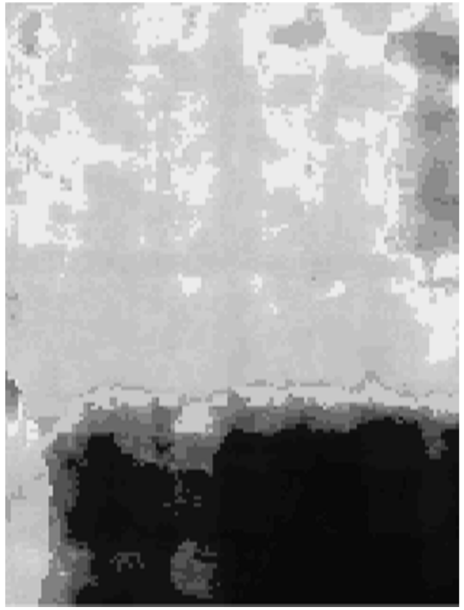
Image reconstruction

Captured depth image with corrupted data:

Corrupted data (dark blue):

- **Mapped area:** 5 *x* 4.5 m
- **# images:** 125
- **Subsample pixels:** 1/100
- **# data (after subsample):** 19200
- **PCA eigenvectors:** 85 %
- **# PCA eigenvectors:** 66

Reconstructed image:

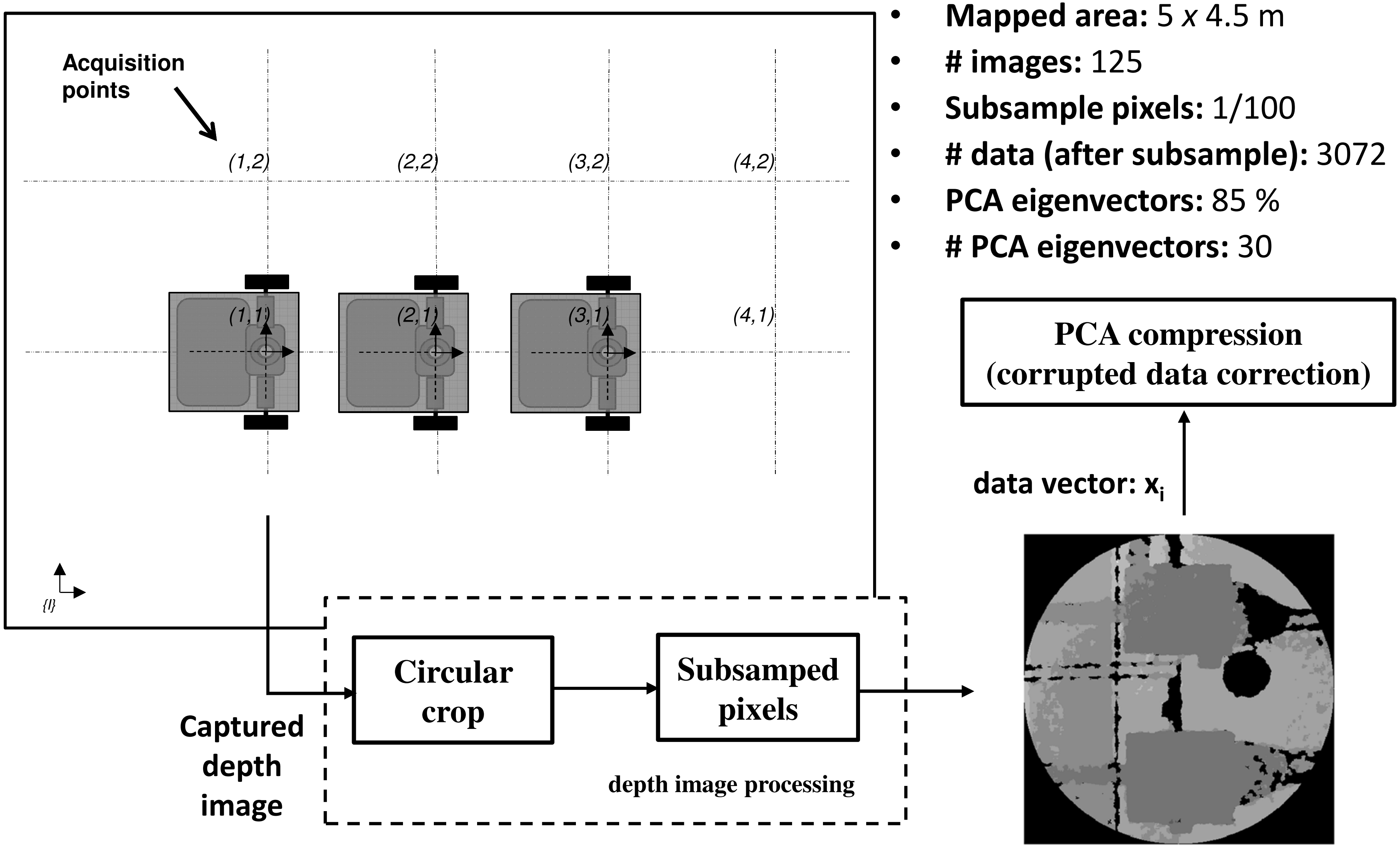


$$\mathbf{v} = \mathbf{U}_n^T\left(\mathbf{x} - \mathbf{m}_x\right)$$

$\mathbf{v}$

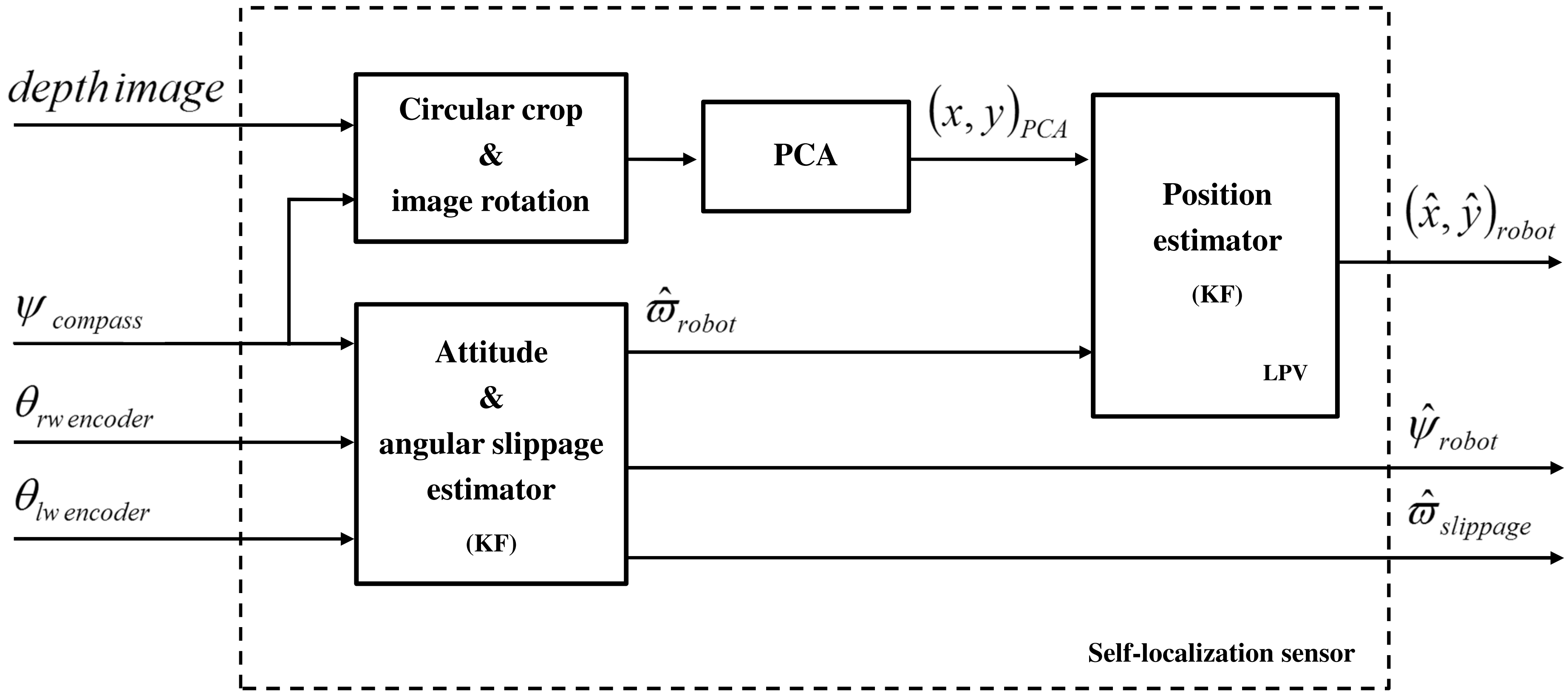
$$\mathbf{x}_r = \mathbf{U}_n \mathbf{v} + \mathbf{m}_x$$

PCA database (2D localization)



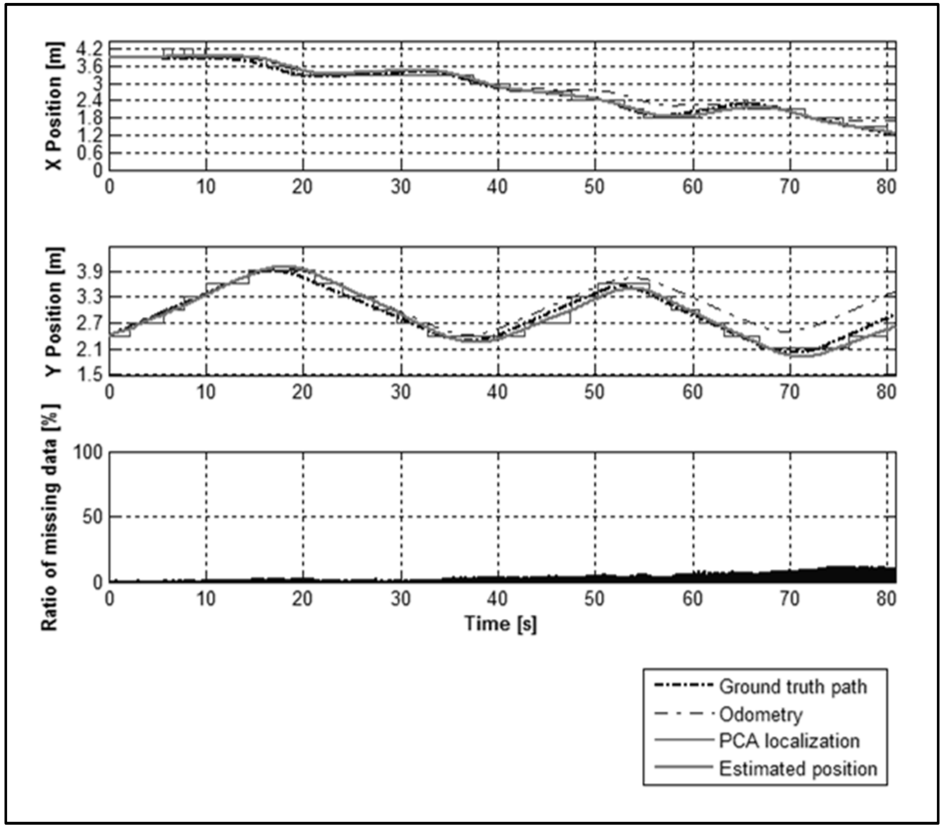
- **Mapped area:** 5 *x* 4.5 m
- **# images:** 125
- **Subsample pixels:** 1/100
- **# data (after subsample):** 3072
- **PCA eigenvectors:** 85 %
- **# PCA eigenvectors:** 30

Architecture of the self-localization system

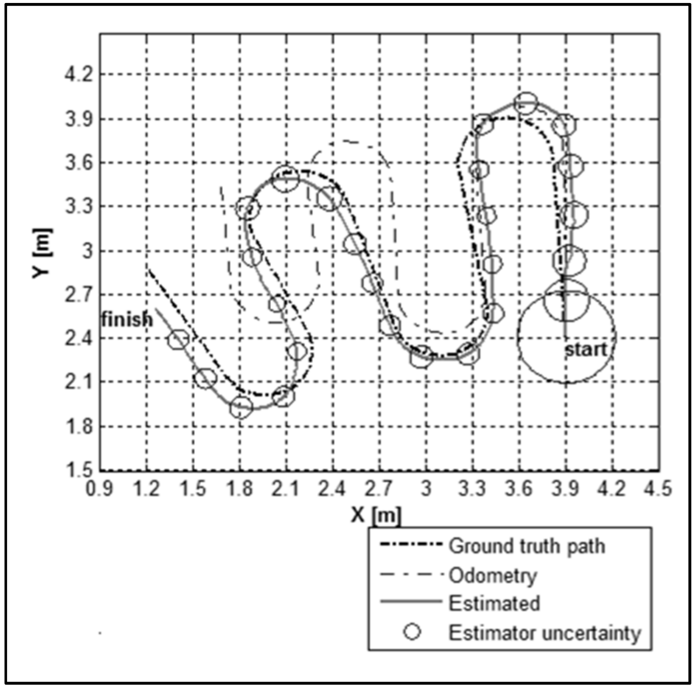


2D localization with missing data correction

- **Sample time:** 0.2 s
- **Robot velocity:** 0.1 $m.s^{-1}$



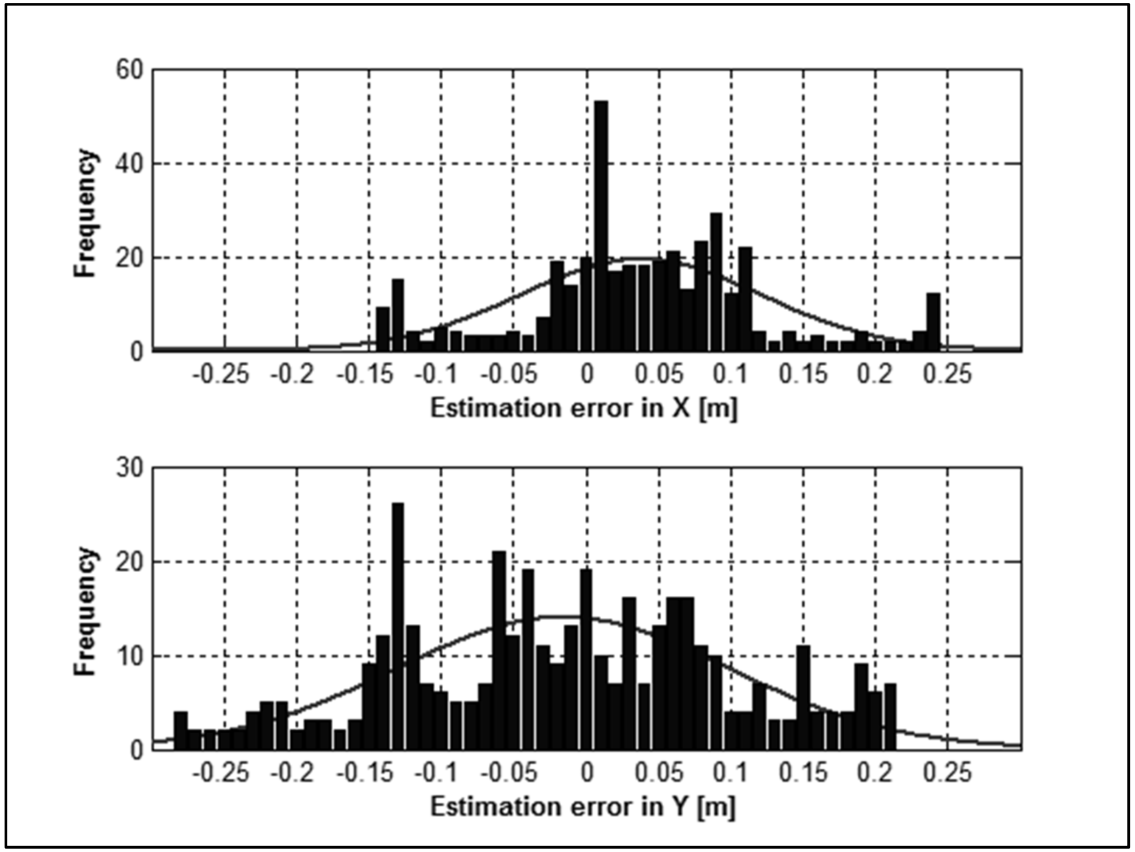
Estimated position along time



Map with estimated position

Distribution of the localization error

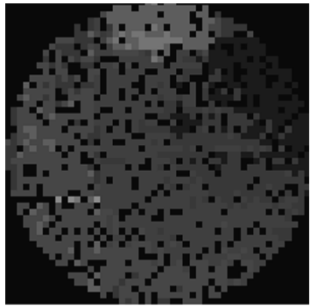
- **# total depth images:** 405
- **PCA localization with KF**

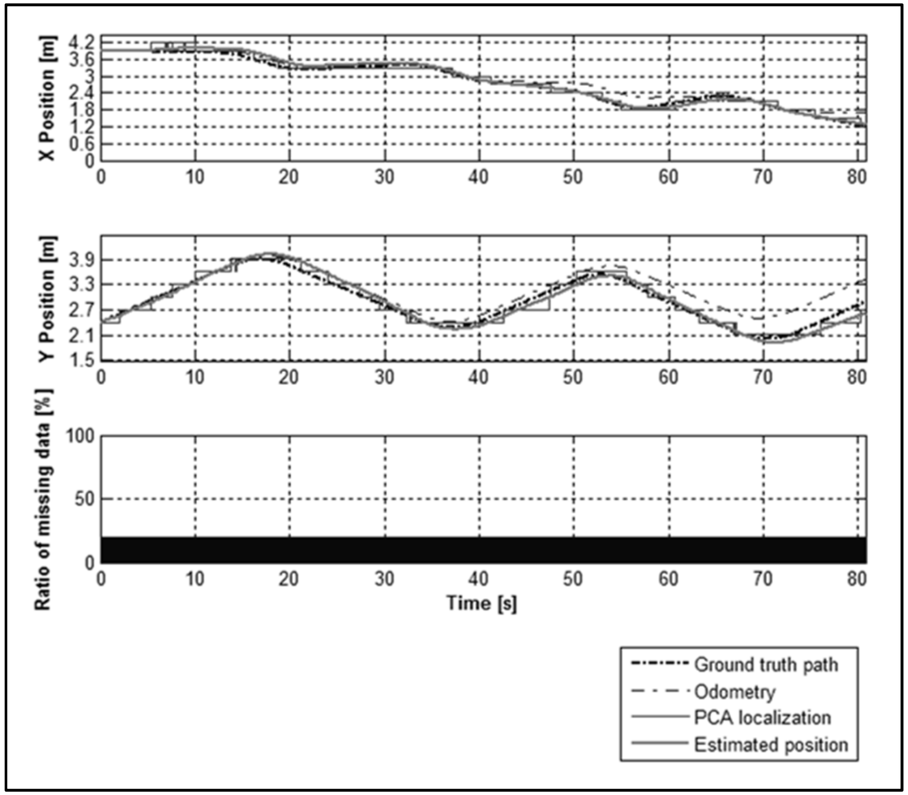


Distribution of estimated position error (PCA position grid with 0.3 m)

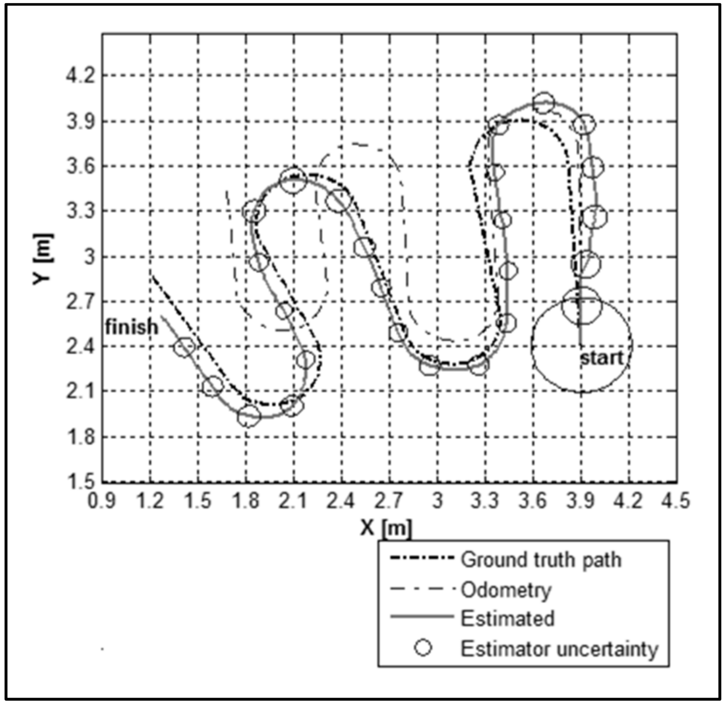
2D localization with imposed corrupted data

- **Sample time:** 0.2 s
- **Robot velocity:** 0.1 m.s^{-1}
- **Imposed corrupted data:** 20%





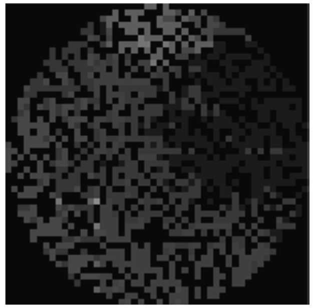
Estimated position along time

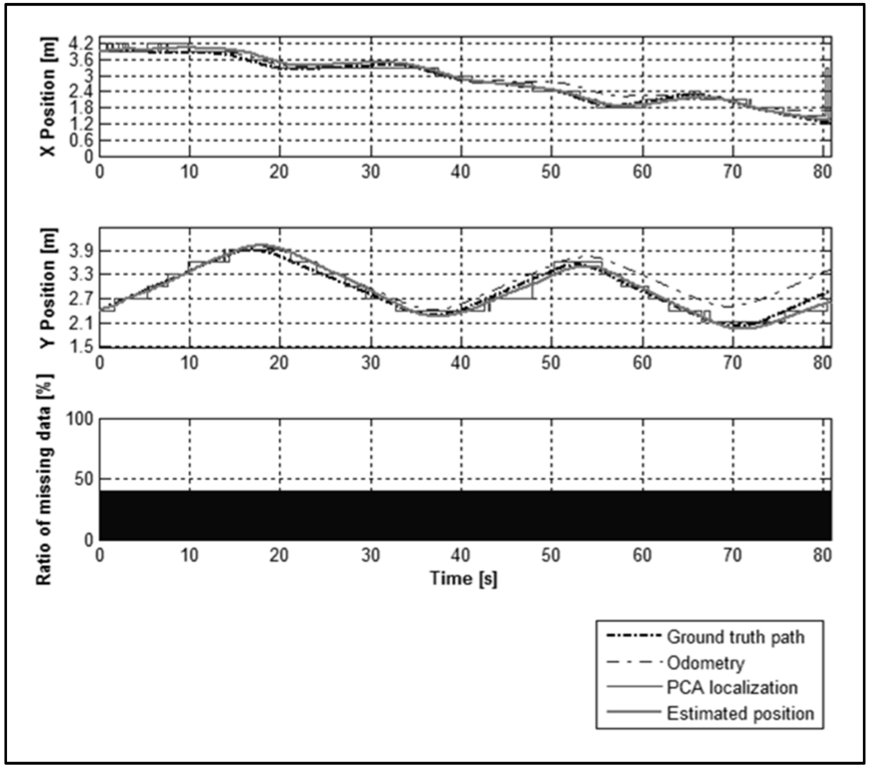


Map with estimated position

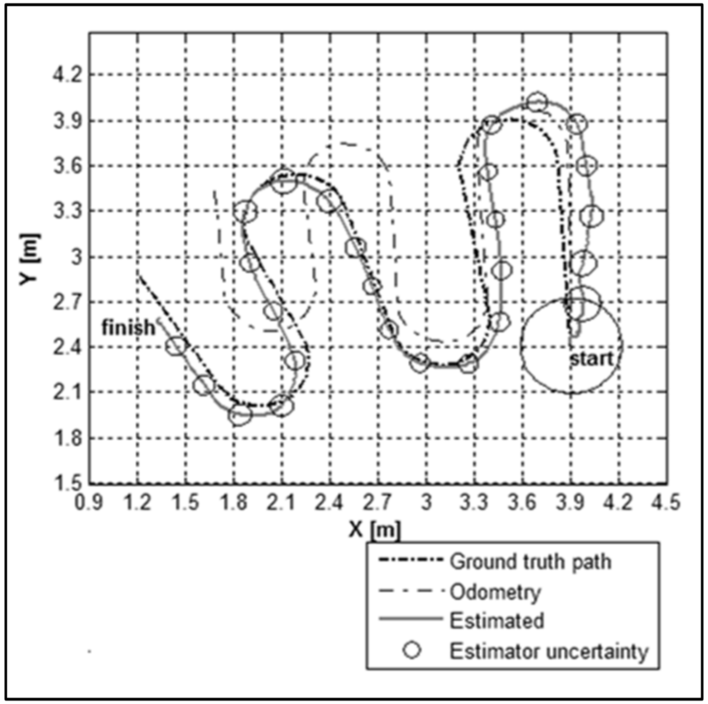
2D localization with imposed corrupted data

- **Sample time:** 0.2 s
- **Robot velocity:** 0.1 $m.s^{-1}$
- **Imposed corrupted data:** 40%





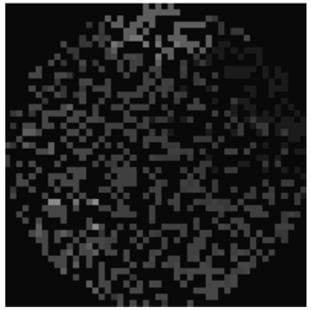
Estimated position along time

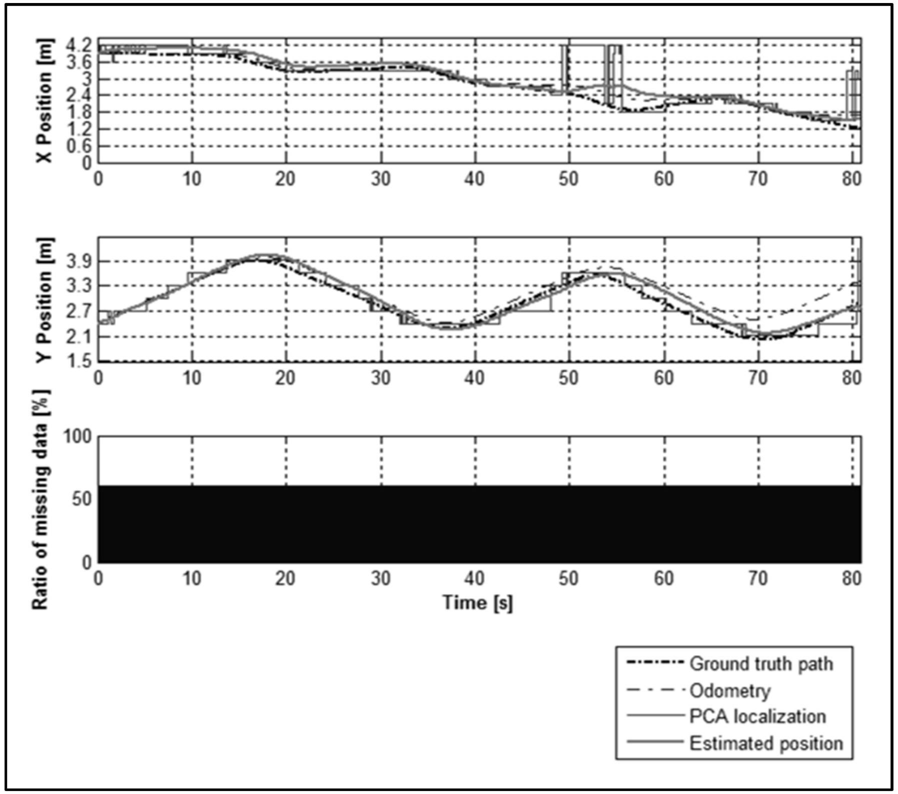


Map with estimated position

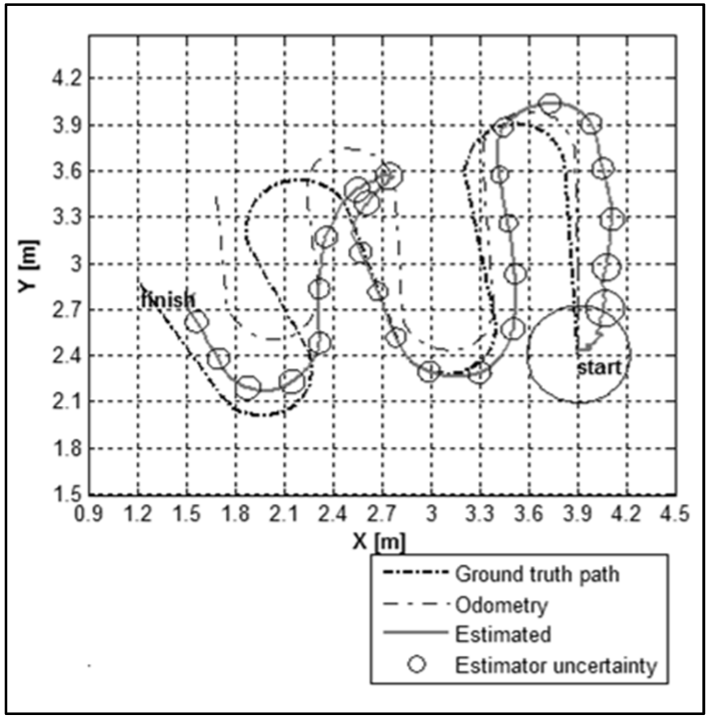
2D localization with imposed corrupted data

- **Sample time:** 0.2 s
- **Robot velocity:** 0.1 m.s^{-1}
- **Imposed corrupted data:** 60%





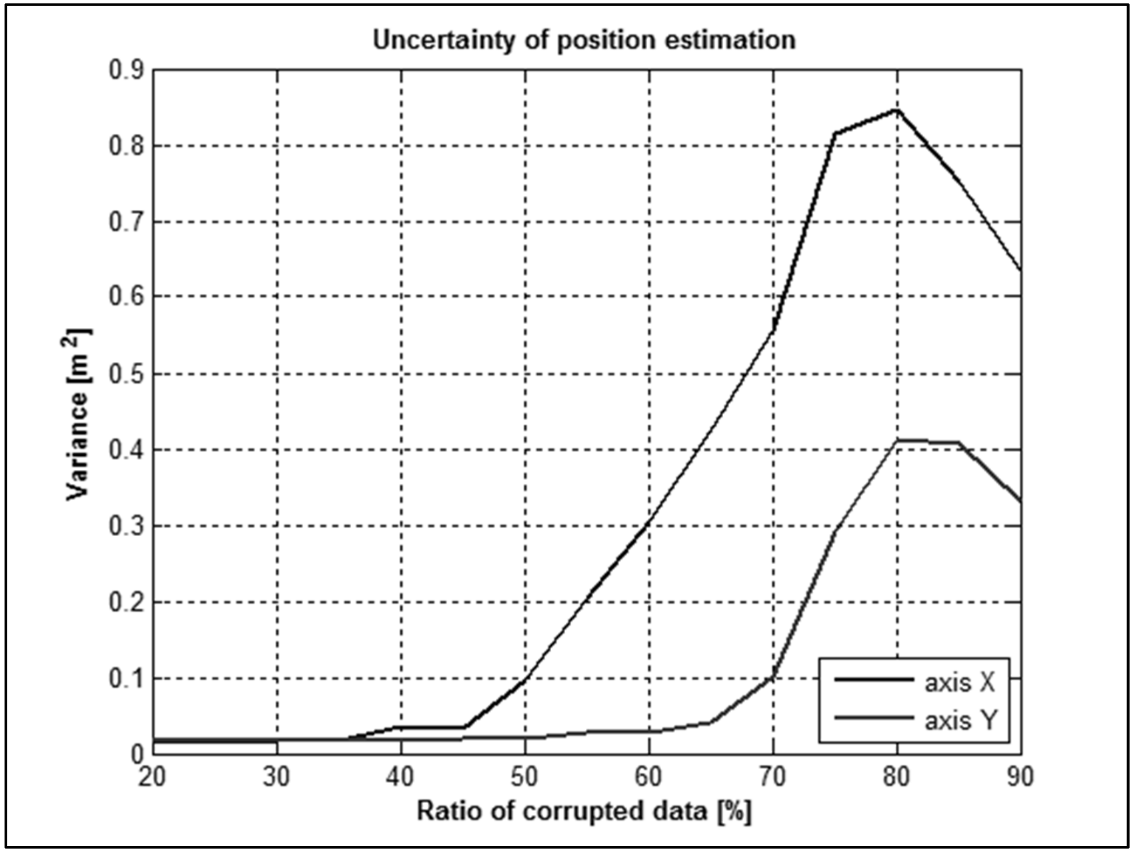
Estimated position along time



Map with estimated position

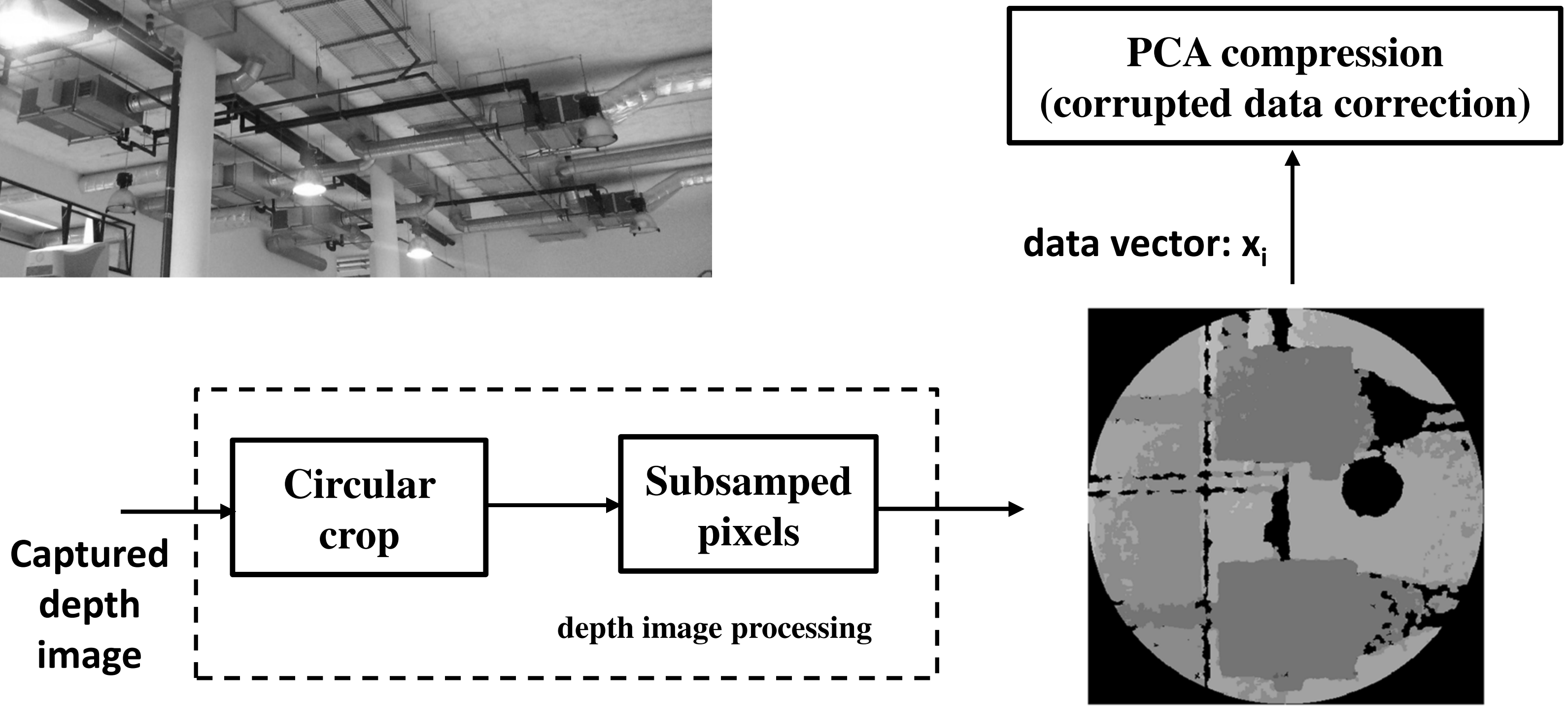
2D localization with imposed corrupted data

- **Imposed corrupted data:** 20% to 90%

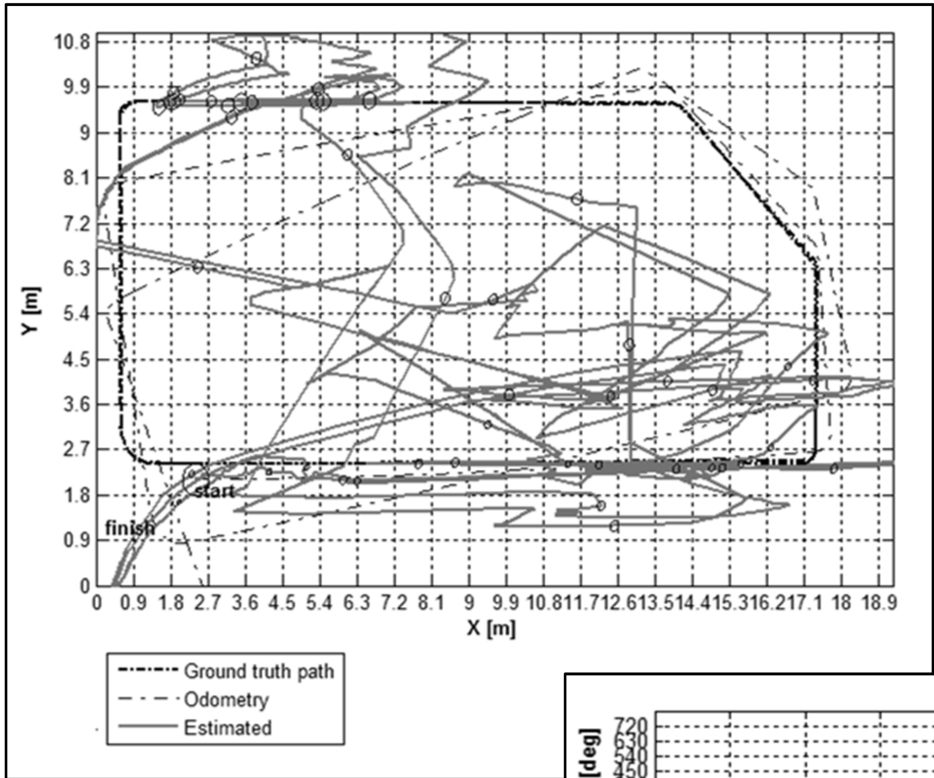


PCA database (2D localization)

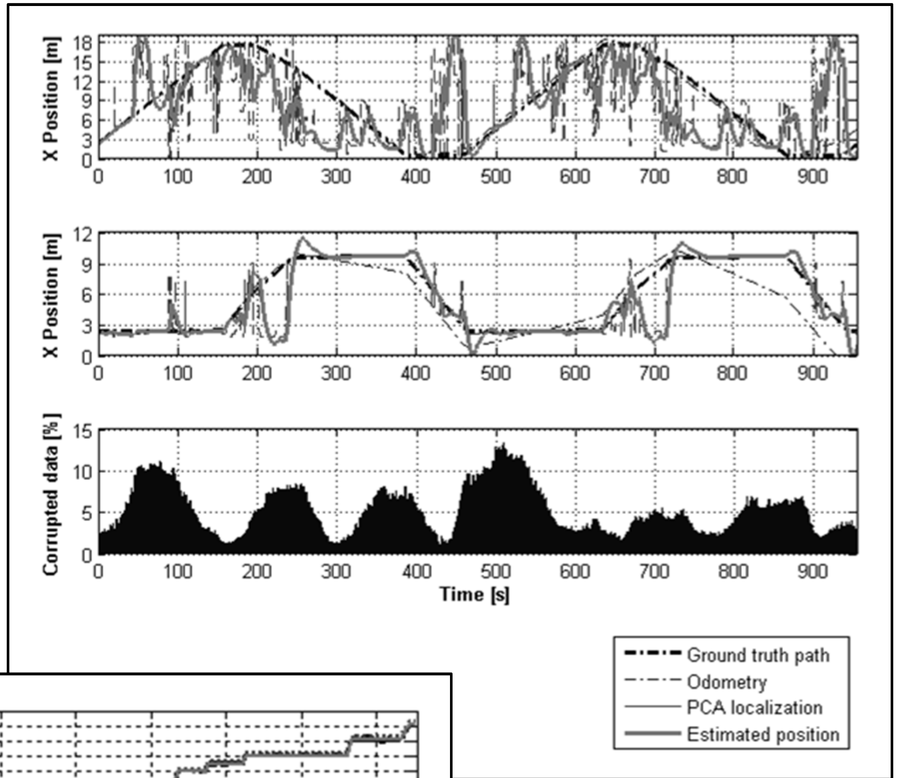
- **Mapped area:** 18.9 *x* 9.6 m
- **# images:** 1115
- **Subsample pixels:** 1/100
- **# data (after subsample):** 3072
- **# PCA eigenvectors:** 30



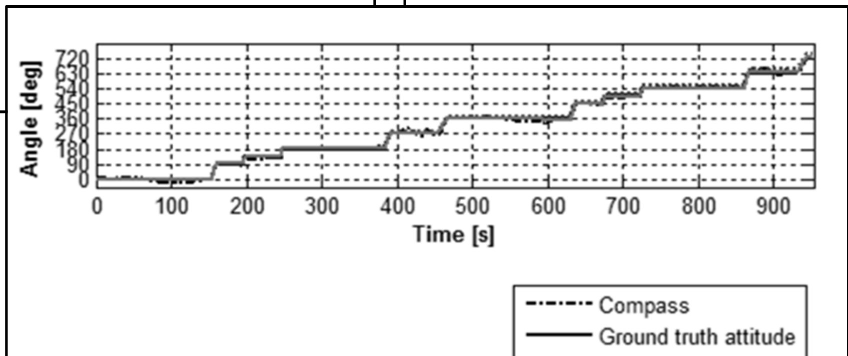
Problem of repetitive scenarios



Map with estimated position

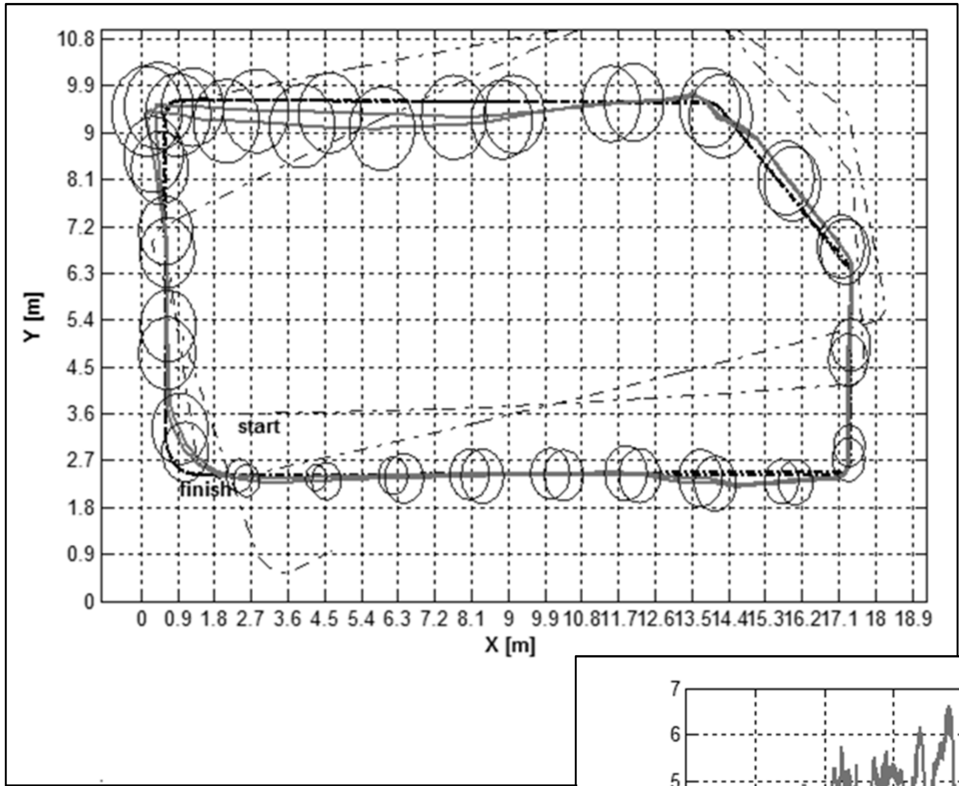


Position along time

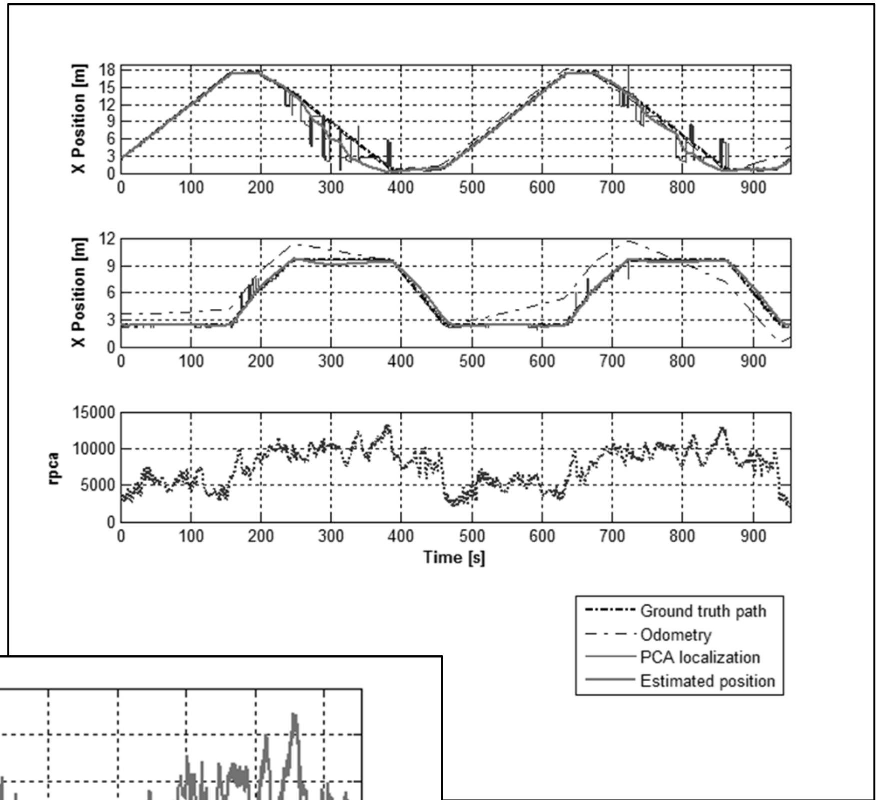


Estimated attitude

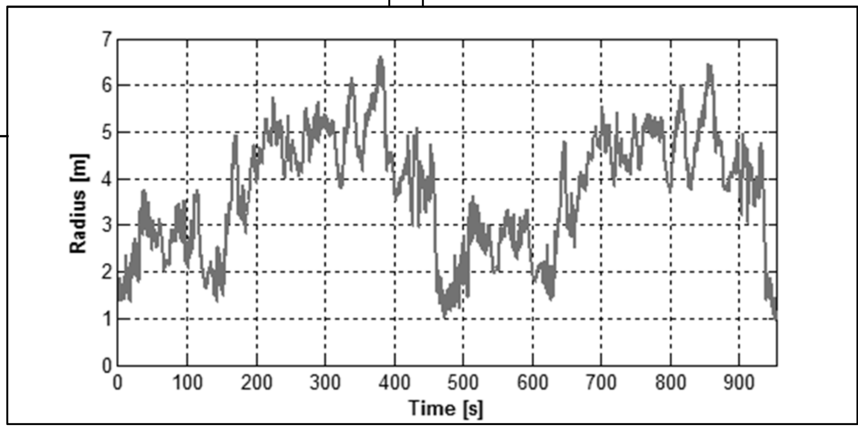
PCA with search in a Neighbourhood



Map with estimated position



Position along time



Radius of searching neighborhood

Conclusions

- a self-localization system is developed to estimate position, attitude and angular slippage of Differential Drive Car mobile robot
- no hypothesis is made about specific features in the environment
- the PCA position sensor is merged in an architecture with linear Kalman filter
- the estimators are globally stable and optimal under the Gaussian assumption
- with depth images the existence of missing data can induce a position system to an erroneous localization
- an extension of a PCA algorithm to avoid the negative impact of missing data is experimentally validated
- the searching in a neighborhood allows the mobile robot in repetitive scenarios

Thanks for your attention !!!

This work was partially supported by
Fundação para a Ciência e a Tecnologia, through IDMEC under LAETA.

INSTITUTO SUPERIOR TÉCNICO

idmec IST engenharia mecânica

ISR

ISEL INSTITUTO SUPERIOR DE ENGENHARIA DE LISBOA