“Sensor Fusion Techniques for Mobile Robotics”

Prof. Sauro Longhi

Dipartimento di Ingegneria Informatica, Gestionale, dell’Automazione
Università Politecnica delle Marche
Ancona - Italy

sauro.longhi@univpm.it
Saturday, September 3 (9h-10h30)
Sensor fusion techniques for mobile robotics

Monday, September 5 (9h-10h30)
Kalman filtering in sensor fusion

Outline

✓ Introduction
✓ Map building/ Localization
✓ Kalman filter
✓ SLAM
✓ Fault diagnosis
How combine data and information from multiple sensors in order to improve accuracies and inference about the “environment”

Multisensor data fusion

Technologies

Methodologies

Performances

Reliability

To improve
Multisensor data fusion is “new” for artificial intelligent systems

Background

Humans and animals have evolved the capability to use multiple senses (“natural sensors”) to improve their capability to “survive” (to “evolve”)!!!
In recent years sensors fusion has been deeply investigated in different fields of “sciences and engineering”

Example

Automated target recognition


Automatic landing guidance


Remote sensing


Monitoring of manufacturing processes


Robotics


Medical applications


There are a lot of contributions proposed by the
- Aeronautics researchers community
- Manufacturing (Robotics) researchers community

**Air Traffic Control**
- redundant radar antennas
- GPS
- aircraft transponders
- altimeters
- …

Military applications:
- AWACS aircraft
- verify aircraft identity (Identify Friend and Foe)
- …

**Autonomous robots**
- to improve industrial automation
- manipulators
- mobile robots
- to simplify the interaction with the environment
- reliable information about the environment
- a network of robot sensors

Intelligent autonomous robots
In the framework of Artificial Intelligent Systems

Control paradigm:

Artificial Intelligent System

sense → reason → act

environment

Introduction
Ancona: September 2005
Example:
Mobile robot

Mobile robots are not only “computer with wheels”, but autonomous and “intelligent” agents with the integration and application of many technologies and methodologies.

An autonomous mobile robot integrates intelligence with perception and action.

A mechatronics device which is able to move and to act autonomously in the environment.
✓ Automatic Guidance Vehicle (AGV) for industrial applications
✓ Assistive Mobile Base for rehabilitation applications (MOVAID, …)
✓ Unmanned Vehicles for Underwater activities (UVU, ROV, …)
✓ Unmanned Vehicles for military use (land-mine detection, scouting, …)
✓ Air Unmanned Vehicles for disaster management systems (fair detection, …)
✓ Planetary rovers (JPL rover, Mars Pathfinder, …)
✓ Hard exploration (Polar robots, …)
✓ Mobile base for maintenance and survey (cleaning, gardening, …)
✓ Entertainment robotics (RoboCup, Lego Dacta, Sony Entertainment robots, …)
✓ …
Automatic Guidance Vehicle (AGV) for industrial applications
MOVAID: a robot for assistive applications
Scuola Studi Superiori S. Anna, Pisa
Prof. P. Dario
Supporting cable
Launching ship
Main frame
Underwater vehicle
QuickTime™ e un decompressore TIFF (LZW) sono necessari per visualizzare quest’immagine.
Robots in Iraq and Afghanistan
Air Unmanned Vehicles for disaster management systems (fair detection, …)
Planetary rovers (JPL rover, Mars Pathfinder, …)
Hardware Integration: External
Introduction

Ancona: September 2005
Mobile robot

- mobility
- control

behavioural autonomy

- reasoning
- decision making

interaction with the environment

- sensing
- modelling

Introduction
Ancona: September 2005
Mobile robot

The classical Control Paradigm:
sense-reason-act
Performances are improved by the reduction of the gap between perception and action:

- parallel processing of sensor data
- to reduce the importance of the world modelling
- to limit symbolic reasoning

New control architecture:
- different levels
- independent modules
- direct access to sensor data and actuators
- reduced interactions between modules
In robotics (mobile robotics) an important aspect is the integration, optimization of the sense capabilities!!!

The fusion of sensor data for improving the autonomy and reliability of mobile robots.

- Base localization
- Obstacle detection
- Cooperative task
...

INTERACTIONS with the ENVIRONMENT
Intelligent autonomous robot

Where am I? → Localization problem

Where am I going? → Localization and/or place recognition problem

How do I get there? → Path planning problem

MOVAID

sensors
Sensor := it is a device that, in a given environment A, produces an information V at the time t:

\[ S(A,t) = \{ V(t), \varepsilon(t) \} \]

uncertainty (noise)

datum

The range of uncertainty is a priori known (a priori estimated)

It is possible to associate a probability to the sensor reading (there are many causes to alter the sensor reading)

\[ P(S,A,t) = < \text{prob. that the distance of } V(t) \text{ from the physical value measured in } A \text{ is less than } \varepsilon(t) > \]

\[ P(S,A,t) \text{ depends on } \varepsilon(t) \text{ and it is unknown (in many cases)}!!! \]
Information from sensors is: uncertain, erroneous, incomplete.

Many sensor measurements are uncertain.

The true physical sensing device does not operate in practice as we have modelled it. Systematic errors and random errors. Uncertainty and Erroneous are often used interchangeably in this context.

They depend on the parameter variations and uncertainties.

Incompleteness := a single sensor cannot sense all information.
The information from sensors is incomplete, erroneous and uncertain!!!

It is necessary to FUSE redundant information from multiple sensors!!!

Multiple sensors should measure (partly) the same information to REDUCE the **errors** and the **uncertainty** in this information and to solve the problem of **incomplete** information.
Main features of a sensor:

- Accuracy: how correct the sensor reading is
- Repeatability: how consistent the measurements are in the same circumstances
- Resolution: granularity of results (e.g. 1cm resolution)
To improve the accuracy a general solution is based on the integration of many different sensors.

Independent sensors used in a proper combined way are able to increment the capability for acquiring information on the environment.

Complementary sensors

Competitive sensors

Cooperative sensors

Virtual sensors

How do we combine the sensor readings?

Multiple sensors data fusion
The idea:

Fusion with a common representation
“Complementary” and “Competitive” fusion

intelligent autonomous robot
environment
intelligent autonomous robot

environment

Introduction
Ancona: September 2005
Classification by types of sensors data

Complementary fusion:

fusion of several disparate sensors which give partial information of the environment (this procedure resolves incompleteness of sensor data)

Competitive fusion:

fusion of uncertain sensors data from several sources (devices) (this procedure resolves uncertain and erroneous of sensor readings)

random and systematic errors are reduced

Cooperative fusion:

fusion of different sensors where one sensor uses the observation of the other sensors for generating the observation (stereo vision systems)

Classification by levels of representation

Signal level:

Fusion is performed by the combination of the signal of a group of sensors; the signal can be modelled as random variables corrupted by uncorrelated noise; the fusion is an estimation procedure.

Feature level:

Fusion is performed at the feature level; feature represents the sensor information; it increases the reliability of the processing results; the procedure involves the extraction of representation features from multiple sensor information; the fusion is based on a probabilistic procedure.

Symbolic level:

The information from multiple sensors are integrated at the highest level of abstraction (symbol level); it is a pattern recognition.

Classification by architectures

Centralized architecture:

The computable data from many sensors is combined (fusion) and evaluated by the system (interpretation) in a centralized solution.

The sensing process in many cases can be decomposed in:
- detection
- preprocessing
- fusion
- data interpretation
Fusion combines the individual readings to interpret the fused data.
Decentralized architecture:

The sensors are connected, forming clusters; each cluster has a number of sensors collecting redundant information and the related data are fused in first level cluster controller; this cluster controller transmits the fused data to another cluster controller that is a part of a network; finally the data are interpreted in a distribute architecture where local data interpretation modules are connected by a local area network.

To reduce the amount of data to be transmitted and to increase the reliability (fault tolerance)

Environment / Object

Sensor
preprocess
Data fusion Level 1
Data fusion of data from primary cluster controllers

Sensor
preprocess
Data fusion Level 1

Sensor
preprocess
Data fusion Level 2

Sensor
preprocess
Data fusion Level 2

Data fusion Level 2

Data interpretation and communication

Introduction
Ancona: September 2005
There are a lot of mathematical techniques for developing different fusion solutions.

In this summer school some significant examples will be presented and discussed.

My contribution is to recall the Kalman filter approach for sensor fusion in mobile robots, that is strictly related to our research activity in this field.
Part 2: “Map building”

Prof. Sauro Longhi

Dipartimento di Ingegneria Informatica, Gestionale, dell’Automazione
Università Politecnica delle Marche
Ancona - Italy

sauro.longhi@univpm.it
Intelligent autonomous robot

Where am I? \[\Rightarrow\] Localization problem

Where am I going? \[\Rightarrow\] Localization and/or place recognition problem

How do I get there? \[\Rightarrow\] Path planning problem
For the robot localization and/or navigation a map is required

- measures association
- path planning

A map of the environment (indoor environment) can be defined
- as a set of artificial (natural) markers
- a CAD model

Automatic acquisition of environment knowledge (map building) := automatic acquisition of maps of the environment based on natural features (i.e. walls, doors, …)
Some definitions:

Pose tracking := estimation of the robot position and orientation using sensors readings, where the initial estimate of the robot pose (position and orientation) is known

A map is required and the measures association is simple to be performed

Global localization := estimation of the robot position and orientation using sensors readings, but where the initial estimate of the robot pose (position and orientation) is unknown (pose initialization)

A map is required and the measures association is more difficult to be performed!!!

It depends on the size of the environment, on the level of symmetry; a large set of sensors data must be integrated over time
Map building := online construction of a map of the environment by the integration of sensor data acquired by the robot; the position of the robot must be known!!!

SLAM := Simultaneous Localization And Mapping; pose estimation and mapping at the same time (at the same time to build a map and to estimate the pose of the robot); it is a “chicken and egg” problem:
- for a pose estimation a map is required!!!
- for the map building the robot pose must be known!!!

It is an OPEN PROBLEM: there are different solutions for specific situations
Environment representations: there are 4 map representations (in-door):

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

geometric characteristics
**Metric Maps: Features maps**

The features maps are metric maps specified with elements (features) which characterize the environment: lines, corner, edges, … and they can parameterized of the function of colour, length, wide, …

$$M = \{ f_j \mid j=1, 2, \ldots, m \}$$

These maps are generally used for the robot localization, in many cases lines are considered as features of the maps.

Sonar sensors used for detecting environment lines and for comparing them with primitive lines stored in the map.


Natural geometric beacons that are easily observed by sonar readings and that are described by simple a geometric parameterization.


Artificial landmarks that are simple to be detected, then the localization is simple and efficient; this solution requires to introduce some modification on the environment.
Different features are used: lines, edges, corners, semi-planes

Stereo vision system and a CAD model of the environment.

Sonar sensors and walls as environment features.
Matric maps: Grid maps

The environment (the work space) is decomposed into a grid of different accuracy, where each cell represents a part of the environment (path planning, obstacle avoidance, ...)

This solution is generally used with sonar sensors; it represents a filtering tool for increasing the accuracy of sonar readings (at each cell is associated a probability to find an obstacle in that position)

Moravec and Elfes 1987, High resolution maps from wide angle sonar, Proc. IEEE ICRA, pp. 116-121


In this case, the localization of the robot can be done by correlating the acquired local map with global map; different correlation procedure have been developed for increasing the efficiency of the procedure.

Many different simplifications are considered for improving the velocity of the map building.


Many different solutions for correlating the local map with the global map:

i) the local map with the global map
ii) local line segments with the global map
iii) the local map with global line segments
iv) local line segments with global line segments

In the following our contributions to the on line construction of probabilistic grid map will be recalled and discussed, as a first example of sensor fusion in robotics.
Real-time Map Building for Mobile Robots: Experiments with Sonar Sensors

Navigation System for Mobile Robots (general hybrid solution)

- Two main modules:
  - motion planning module
  - collision avoidance module

- Motion planning operates off-line to produce a collision-free path (a-priori knowledge about the environment)

- Collision-avoidance module exploits sensorial information in real-time to avoid unexpected obstacles along the pre-planned trajectory

- Sensor systems (to update the knowledge of the environment)
To increase the autonomy of a mobile robot:

- real time operations on the sensor measures to update the knowledge of the environment

A low cost solution to the problem of on-line acquiring environment data: a set of sonar sensors

To improve the reliability and accuracy of the sonar measures:

- local map of the obstacles on the environment

To develop and to experimental evaluate three methods to fuse together the information carried by the sonar sensors in a mobile robot.
A navigation system of a mobile robot:

the motion planning module

and

the collision avoidance module

A hybrid navigation module with sonar sensors

An hybrid navigation module with sonar sensors

the critical feature is

the on-line construction of the environment map

this module modifies the planned path in order to avoid unexpected obstacles

the choice of the avoidance direction can be produced on the basis of the local information stored in the environment map

for simple obstacles, the environment map could improve the efficiency of the algorithm

with the use of an obstacle map of the environment

The reliability and robustness of the collision avoidance algorithm can be increased

a simple and low cost solution for detecting obstacles on the environment is based on sonar sensors.
Cross-talk reduction

13 sonar sensors

Quantity & Quality of the sonar sensor measures

**Quantity**

Sonar scanning times

\[ T_{\text{scan}} := T_n - T_{n+1} \]

**Quality**

Cross-talk reduction by changing the time interval between two consecutive sonar measures (Borenstein and Koren, 1991)

\[ D_{\text{cross}} = D_{\text{cross}} \pm n \Delta TD \]

This impulsive noise can be easily detected and filtered
Real-time map building

- Different approaches to the on-line map-building have been analyzed
  1. Probabilistic approach
  2. Evidential reasoning approach
  3. Fuzzy Logic approach
- A procedure for the reduction of the cross-talk interference has been used
- The above procedures have been analyzed by a set of experimental tests
- The same model of the sensor device and the same accuracy on the discretized map have been considered
The map is composed by square cells (10 cm each side) containing an obstacle probability. Each measure up-dates the obstacle probability of the cells in the measured zone by a obstacle probability. The central part of the cone of the sonar beam has been considered. An approximation of the Bayes theorem is introduced for reducing the computation efforts in the map up-dating.

The on-line up-dating of the map requires a low computation cost at each sampling instant.

The solution proposed is able to find a trade-off between accuracy and computation cost.

The main features of the implemented procedure are the following ones:

✓ The map is composed by square cells (10 cm each side) containing an obstacle probability

✓ Each measure up-dates the obstacle probability of the cells in the measured zone by a obstacle probability

✓ The central part of the cone of the sonar beam has been considered

✓ An approximation of the Bayes theorem is introduced for reducing the computation efforts in the map up-dating.
Obstacle probability of a cell after a sonar measure

1) Sonar with a return signal: (“in range” measure)
- high value of probability for the cells of the arc
- low value of probability for the cells inside the sonar beam
- 0.5 elsewhere

2) Sonar without a return signal: (“empty” measure)
- low value of probability for the cells of the beam
- 0.5 elsewhere

The new value to assign to the cell \( i \)

\[
P(o_i / S_j, M) = \frac{P(o_i / S_j) P(S_j / M)}{P(o_i / S_j) P(S_j / M) + P(\overline{o_i} / S_j) P(S_j / \overline{M})}
\]

\[
P(o_i / S_j, M) = \frac{P(S_j / o_i, M) P(o_i / M)}{P(S_j / o_i, M) P(o_i / M) + P(S_j / \overline{o_i}, M) P(\overline{o_i} / M)}
\]

This probability has been deduced by consideration on how sonar sensors work.

The following approximations have been introduced:

\[
P(S / o_i, M) \approx P(S / o_i)
\]

\[
P(S / \overline{o_i}, M) \approx P(S / \overline{o_i})
\]

Bayes theorem

grid-based model of sonar sensor
Experimental results

LabMate with 13 Polaroid sensors

Tests:

in-door environments with a proper set of unknown obstacles

paths free of collisions have been planned

A corridor
Probabilistic approach

Dark regions: = cells with an high probability of obstacle

Evidential reasoning approach

Dark regions: = cells with a lot of evidence supporting obstacle presence and few evidence supporting obstacle absence

Fuzzy logic approach

Dark regions: = cells with an high risk for robot navigation

The map of conflict = cells where conflicting sonar readings are detected (dark regions correspond to cells with a high lack of consensus between sonar readings)
Probabilistic approach

Unexplored area

Cells with a low probability of obstacle presence

Cells with a high probability of obstacle presence

Cells with an undetermined state
Evidential reasoning approach

- **Start labmate**
- **Goal labmate**
- **Obstacles**

- **Cells with a low evidence of obstacle presence**
- **Cells with a high evidence of obstacle presence**
- **Cells with an undetermined state**
- **Unexplored area**
Map building

Unexplored area

Cells with a low risk for robot navigation

Cells with a high risk for robot navigation

Cells with an undetermined state
Results

✓ In all the performed tests the second approach, based on the conditioned Dempster-Shafer theory, has produced the best results (the differences are “small”)

✓ This method allows to detect and to analyze the conflict readings (by the map of weight of conflict).

✓ Moreover the map of weight of conflict can be used to decide on the reliability of a sonar sensor and if it is necessary to exclude this sonar temporarily or permanently

Other contributions:

- To fuse the environment data in a multirobot system.
- The map building of the environment by fusing sonar data with video data.
The co-ordination problem in the path planning and in the obstacle avoidance can be solved by an hierarchical approach.

Problems:

- the navigation in presence of the other robots
- the perception of the other robots
- the transmission of information among the robots
- the fusion of the knowledge on the environment

The same hybrid structure for the navigation system of each robot:
path planner + obstacle avoidance module
The hierarchical approach requires:

- data communication among the mobile robots
- co-ordination

Reliability and robustness of this approach can be improved with a supervisor system which co-ordinates the activities and controls the status of different robots.

The supervisor system allows:
- to optimize the global task of the multirobot system
- to integrate the environment information acquired by all the mobile robots

### Diagram

- **Supervisor**: The central unit managing the tasks.
- **Local Area Network (LAN)**: Communicates tasks and information.
- **Hybrid Navigation Module**: Integrates navigation strategies.
- **Robots**: Mobile entities managing tasks.

**Protocol Management**: Controlled by the network supervisor.

**Local Area Network**: Imposes tasks and priorities, integrates environment knowledge.
Navigation system of each robot

Functional Level
- Perception module
- Localisation and Control module
- Motion Planning module

Decisional Level
- Obstacle Avoidance Module
- Data Module
- Communication Module
- Local Supervisor Module

Supervisor / other robots
Environment data fusion:

The global map of the environment can be updated with the sonar measures acquired by all the robots.

The data fusion on the supervisor:

1. a unique map on the supervisor

2. the supervisor integrates the single maps
   i. a simple average
   ii. a weighted average

The sonar readings of each robot are used for the real-time updating of the local map of each robot and of the global map of the supervisor.
This technique does not decrease the autonomy of each robot:

- Each navigation module updates its own local map.
- The supervisor integrates the single maps and communicates the global environment map to each robot.

The Dempster-Shafer method is used for the global map and the local maps.

The periodic data transmission is performed by radio link and a local area network (Ethernet LAN) with the User Datagram Protocol (UDP).

The Transmission Control Protocol (TCP) is used for the aperiodic data transmission.
The main features:

The aperiodic data transmission: the global map, the planned task (high reliability is required)

*The low complexity of the UDP* \rightarrow real-time acquisition on the supervisor of all sonar readings

A server/client solution is developed for the data communication:

A remote Internet connection by a compatible Java browser to analyze the global map to impose the tasks
Experimental results:

- 16 Polaroid sensors on the Labmate base
- 16 Polaroid sensors on the TGR Explorer (powered wheelchair transformed in a autonomous vehicle)
- Crosstalk reduction: sensors distribution Borenstein-Koren procedure
- Navigation system of each vehicle in two PC:
  - Functional level: PC486 104-bus
  - Decisional level: desk PC Pentium

The communication module with three independent sub-modules:
Server sub-module manages the TCP and UDP protocols to the supervisor / other robots;
Client sub-module manages the TCP and UDP protocols from the other robots / supervisor;
Local module manages the serial communication functional level / decisional level.
Two independent tasks free of collision

The proposed procedure improves the knowledge on the indoor environment

On the performed tests:
real-time acquisition of sonar readings by UDP

Internet connection: Java browser for acquiring the global map
Results

✓ A developed method (architecture) is able to fuse together the information acquired by each single robot.
✓ The purpose is that of obtaining a global environment map increasing the reliability and accuracy of the sonar measures.
✓ Results: experimental tests (LABMATE + TGR).
✓ In all the performed tests the procedure has produced satisfactory results.
✓ The proposed procedures are quite general so they are directly applicable to multirobot systems composed of any number of robots and characterized by different kinds of sensor equipment.

The map building of the environment by fusing sonar data with video data.
Autonomous mobile robot navigation:

+ perception of local environment
+ obstacle avoidance in a unknown environment
+ vehicle pose estimation
+ auto-localization on a known environment
+ sensor readings

Local map environment:

+ grid base modelling
+ feature based modelling

Our contribution is to improve the feature based modelling by fusion of ultrasonic range readings and digital video images
Ultrasonic range readings

Digital video images

2D modelling technique for indoor environments based on line features

For improving:
- accuracy on environment knowledge
- vehicle pose estimation

For reducing:
- the amount of computer memory in exploration of large environments
- the cost of the sensor system (by using data fusion techniques)
Sensor system

Sonar sensors: aggregation of sonar measures in a probabilistic map of the environment:
- only information about existence of an obstacle;
- no information about shape of the obstacle’s contour

Monocular Video sensor: extraction of straight lines from video images:
- accurate information about the contours of the objects in the image;
- it is not able to discern (by itself) obstacles from the contours of the objects

Accuracy improvement

data fusion gives to the system the capability to extract 2D obstacle’s contour as straight lines features
Proximity system: ultrasonic sensors

The ultrasonic sensors installed on the vehicle

Half ring of sonar sensors installed around the vehicle

Occupancy grid map:
- it is a 2D representation of the whole environment
- sensor readings are aggregated as occupancy probability values stored in the cells of a grid map

The considered framework used for representing range information is the occupancy grid map

Map building
Ancona: September 2005
Video system: monocular CCD camera

The *video sensor* installed on the vehicle it is a monocular CCD web-cam installed on the vehicle and pointed down toward the floor

**Digital image:**
- it is a 2D representation of a *part of the environment*
- sensor readings are not expressed in terms of probability values (probability measures must be computed)

The considered *framework* used for representing contour information of the obstacles is the *digital image*
Conditions for fusing ultrasonic and video data

1. To extract *common information* from both sets of sensor data:
   *LINE FEATURES (Hough Transform)*

2. To find a *common framework* for representing the common information:
   *VISIBLE SPACE OF THE CAMERA*

3. To find a *mathematical instrument* for relating line features of different framework:
   *PIN-HOLE CAMERA MODEL*

4. To match *line features* by using a *FUSION TECHNIQUE*
LINE FEATURES and the HOUGH TRANSFORM (HT)

The HT is an image processing technique used for detecting line features

✓ based on a polar representation of line features

\[ \rho = x \cdot \cos \theta + y \cdot \sin \theta \]

✓ the Hough accumulator (Hough space) is a square array which rows and columns represent quantized values of line parameters \( \rho \) and \( \theta \); each cell of the array stores a value representing existence evidence for the corresponding line.
Probabilistic Hough Transform

It allows to associate a **probability values to the line features** detected **from digital images** by computing the probability contribution of each edge pixel to the Hough space \( H_V(\theta, \rho) \)

Denote:

- \( \Theta_v = (\theta, \rho) \) a vector consisting of all possible **quantized values** of line parameters \( \theta \) and \( \rho \) in the Hough space.

- \( \hat{\Theta}_v = (\hat{\theta}, \hat{\rho}) \) be a vector representing the **estimated line parameters** from a pixel and assume \( \hat{\Theta}_v \) normal distributed as:
  \[
  \hat{\Theta}_v \approx N(\Theta_v, \sum_{\hat{\Theta}_v}) \text{ where } \sum_{\hat{\Theta}_v} \text{ is the covariance matrix.}
  \]

The **normal joint density distribution for** the estimation \( \hat{\Theta}_v \) has the form

\[
p(\hat{\Theta}_v | \Theta_v) = \frac{1}{2\pi |\sum_{\hat{\Theta}}|^{-\frac{1}{2}}} \exp\left( -\frac{1}{2} (\hat{\Theta}_v - \Theta_v)^T \cdot \sum_{\hat{\Theta}}^{-1} \cdot (\hat{\Theta}_v - \Theta_v) \right)
\]
Considerations about the Normal joint density distribution $p(\hat{\Theta}_V | \Theta_V)$

General situation:
2 correlated normal r.v. $\hat{\theta}, \hat{\rho}$

\[ \sum \hat{\Theta}_v \neq 0 \]

Singular situation:
2 linearly dependent normal r.v. $\hat{\theta}, \hat{\rho}$

\[ \sum \hat{\Theta}_v = 0 \]

bivariate normal distribution

$P(\hat{\Theta}_V | \Theta_V)$ = $\int \int p d\theta d\rho$

Q. Ji & R.M. Haralik, *Int. Conf. SMS 1988*
It was shown that

1. for \( \hat{\Theta}_v \) it results \( \sum \hat{\Theta}_v \) singular \( \left| \sum \hat{\Theta}_v \right| = 0 \)

2. \( p(\hat{\Theta}_v | \Theta_v) \) can be always computed by applying a simple co-ordinate transformation which transform \( p \) in the distribution of two independent r. v. (by means of a proper reference frame rotation)

3. by property 2 it can be showed that \( p(\hat{\Theta}_v | \Theta_v) \) can be computed as a function of only one r.v. (i.e. \( \theta \)) and then

\[
p(\hat{\Theta}_v | \Theta_v) = p(\hat{\theta} | \theta) = \frac{1}{\sqrt{2\pi}\sigma_{\hat{\theta}}} \exp\left(-\frac{1}{2} \frac{(\hat{\theta} - \theta)^2}{\sigma_{\hat{\theta}}^2}\right)
\]

\[
P(\hat{\Theta}_v | \Theta_v) = \frac{1}{\sqrt{2\pi}\sigma_{\theta}} \int_{\hat{\theta} - \frac{\Delta \theta}{2}}^{\hat{\theta} + \frac{\Delta \theta}{2}} \exp\left(-\frac{1}{2} \frac{(\hat{\theta} - \theta)^2}{\sigma_{\theta}^2}\right) d\theta
\]
Hough accumulator (Hough space) is a square array which rows and columns represent quantized values of line parameters $\rho$ and $\theta$; each cell of the array stores a value representing existence evidence for the corresponding line.

The contribution of each edge point to all the possible line is obtained by iterating the computation of the edge point probability $p(\hat{\Theta}_V | \Theta_V)$ for each possible set of $\Theta_v=(\theta,\rho)$ in the discrete array.

The peaks identify the parameters of the most likely lines and the values of these peaks are the probability of these lines.
The common framework

**Occupancy grid map**
- A 2D representation of the considered environment

**Digital image**
- A 2D representation of a part of the environment

**Camera visible space**
- (pin-hole camera model)

For defining which part of the occupancy grid is framed from the camera.
Features extraction from occupancy grid map

**sonar image**

bitmap representation of part of the occupancy grid in the visible space of the camera at time interval \( t \) during the robot motion

Occupancy probability values

0.0 0.5 0.9

Occupancy grid map at time \( t \)

Sonar image at time \( t \)

Vehicle trajectory

Vehicle position

Map building

Ancona: September 2005
Feature lines extracted by applying the Hough Transform to the Sonar Image

Sonar image at time \( t \)

Hough accumulator of the sonar image

The peaks stored in the accumulator are the probability values of the extracted lines.
Features extraction from digital images

Occupancy grid map at time $t$

Digital image at time $t$

Digital filtering

Edge detection of Digital image

straight lines
obstacle’s contour

Vehicle trajectory
Vehicle position

Ancona: September 2005
Feature lines extracted by applying the *probabilistic Hough Transform* to the filtered digital image.

Digital image at time $t$

Hough accumulator of the digital image

The peaks stored in the accumulator are probability values of the extracted lines.
The common framework

Before fusing the lines they will be defined in a common framework (the sonar image)

Lines extracted from Sonar image

Lines extracted from Video Image

Lines equation:
\[ x \cdot \cos \theta + y \cdot \sin \theta = \rho \]

Perspective projection (pin-hole model) of video straight lines on the sonar image

Visible space
Probabilistic Fusion technique

It is a probabilistic fusion of the Hough accumulators.

\[ P_1(\rho_i, \theta_i) \]

Hough accumulator of lines extracted from occupancy grid

\[ P_2(\rho_j, \theta_j) \]

Hough accumulator of lines extracted from video images

\[ P_{12}(\rho_i, \theta_i) \]

Fusion result

\[ P(\Theta_j | \Theta_{1j} \cup \Theta_{2j}) = \frac{P_1P_2}{P(\Theta_j) + (1-P_1)(1-P_2)} \]

Beyes theorem

Map building

Ancona: September 2005
In summary

- SONAR 1 . . . SONAR n

Signal level fusion

Environment sonar map

Local sonar map

Sonar Image

Probabilistic Hough transform

PIN-HOLE CAMERA MODEL

VISIBLE SPACE OF THE CAMERA

Sonar Image

Probabilistic Hough transform

fission of Hough accumulators (Bayes theorem)

Perspective projection of straight lines on the ground plane

Sonar image

Hough space
Sonar image

Digital image

Hough accumulator of the sonar image

Hough accumulator of the digital image
Hough accumulator of Sonar Image

Hough accumulator of Video Image

Probabilistic thresholding

IOP Fusion

IOP := Independent Opinion Pool

E.A. Bender, Mathematical Methods in AI, IEEE CS Press, 1996
Experimental test

Vehicle’s trajectory
Vehicle’s position

Detected obstacle

Vehicle’s actual position
Real obstacles
detected obstacles

Map building
Ancona: September 2005
Map building

Vehicle’s trajectory
Vehicle’s position

Detected obstacle

Vehicle’s actual position
Real obstacles
detected obstacles
Vehicle’s trajectory

Ancona: September 2005
Map building

Ancona: September 2005
Map building

Vehicle’s trajectory
Vehicle’s position

Detected obstacle

Vehicle’s actual position
Real obstacles
detected obstacles
Vehicle’s trajectory

Università Politecnica delle Marche - Ancona
Dip. Ingegneria Informatica, Gestionale, dell’Automazione

Ancona: September 2005
Vehicle’s trajectory
Vehicle’s position

Detected obstacle

Vehicle’s actual position
Real obstacles
detected obstacles
Vehicle’s trajectory

Map building
Ancona: September 2005
Map building
Ancona: September 2005

- Vehicle’s trajectory
- Vehicle’s position
- Detected obstacle
- Vehicle’s actual position
- Real obstacles
- detected obstacles
- Vehicle’s trajectory
Map building
Ancona: September 2005
Vehicle’s trajectory
Vehicle’s position

Detected obstacle

Vehicle’s actual position
Real obstacles
detected obstacles
Vehicle’s trajectory

Map building
Ancona: September 2005
Results:

✔ The sensor fusion increases the reliability of the acquired environment model

✔ The feature based modelling here described can be used for local path planning and reactive navigation of mobile robot

✔ Reduction of the memory demand for environment model

✔ Experimental results encourage further research

✔ Further developments of this work include the application of the proposed techniques for probabilistic localization problems


Obstacle avoidance

The on-line map building is the main tool also for obstacle avoidance: the map is used for detecting with the necessary reliability the obstacle on the environment.

Angeloni et al, “Real time collision avoidance for mobile robots”, Proc. SMCR, 1996

An algorithm has been developed on a proper extension of the Bug2 algorithm (introduced by Lumenlsky for the motion planning problem) with the introduction

✓ of visual capabilities by sonar sensors

✓ of a predefined trajectory

✓ of an avoidance strategy (the choice of the avoidance direction is produced on the basis of the available local information)
Data fusion between sonar readings and video data for improving the construction of a feature map to be used for the localization.
Part 3: “Kalman Filter”

Prof. Sauro Longhi

Dipartimento di Ingegneria Informatica, Gestionale, dell’Automazione
Università Politecnica delle Marche
Ancona - Italy

sauro.longhi@univpm.it
In mobile robotics sensor fusion is used for base localization and map building

Different sets of sensors are used (internal, external) proprioceptive and exteroceptive sensors

Problem to be solve is the integration of the different sensors readings

A simple and efficient solution is based on a proper designed Kalman filter

Main hypothesis: linearity and Gaussian distributions for modelling the uncertainties of the sensor readings
Intelligent autonomous robot

Where am I? Localization problem

There are many different techniques for estimating the pose of a mobile robot

\[ \mathbf{x} := [x, y, \theta]^T \]

\( \text{pose} := \text{robot position over the plane xy + orientation} \)
(Pose tracking if the initial robot pose is known)

For example in a **Bayesian approach**: to find $p(x_k \mid z_1, z_2, \ldots, z_k)$ where $x_k$ is a time varying signal and $\{z_i\}$ are measurements acquired by sensors.

A more complete description of the state of the robot is not possible.

When $p(x_k \mid z_1, z_2, \ldots, z_k)$ can be represented by a Gaussian, the mean and the covariance give a full description of the state.

Most of Localization procedures are based on different methods for approximating the distribution $p(x_k \mid z_1, z_2, \ldots, z_k)$.
In the robot localization, the main problem is the right association between sensor data and environment model.

If the data association can be performed, then the localization problem is a classical estimation problem, where the sources of all measures are known.

Then the pose estimation can be decomposed in two main steps: prediction + updating.

For example:

- **Proprioceptive sensors** → **Prediction** → **Model of the environment** → **Updating** → **Esteroceptive sensors**
The prediction step is introduced for avoiding “ambiguity” on the sensor data (video, sonar, laser… exteroceptive sensors), it is always necessary!!!

**Data Association**

It is necessary for determining the measurement “source”, the “object” of the measurement

Some measurements must be neglected, because they are related to objects of the environment that are not modelled in the map, and therefore they cannot be used for the localization (unknown obstacles…)

\[ M = \{ f_j | j=1, 2, \ldots, m \} \] is the environment model

(Feature map)

In the data association there are two possibilities for \( z \):

i) it is a measurement of an element \( f_j \) of \( M \)

ii) it is a measurement of an object not included in the map (to be neglected?)
**Pose estimation** (a possible classification)

✓ *Dead-Reckoning*
✓ *Topological*
✓ *Gaussian*
✓ *Position Probability Grids*
✓ *Monte Carlo Methods*

This possible classification is based on the considered representations for modelling the uncertainty on the robot pose.
Gaussian

The Gaussian distributions are the most considered tools to represent the uncertainty of the robot (there are many different independent sources of uncertainty).

The **Kalman Filter** is generally used in this context together with a metric map as a model of the environment (the model is generally based on edges, lines, planes...).

The **Kalman Filter** is a simple and optimal recursive data processing algorithm. For the considered problem the key point is the “recursive” aspect: the measures are processed as they are acquired.

The **Kalman Filter** provides an optimal estimation of $x$ given the measurements $z$ if:
- the system is linear
- the model noise is white and Gaussian
- the measurement noise is white and Gaussian

Kalman filter
Ancona: September 2005
This solution (Kalman Filter for pose tracking) will be deeply analyzed in the following

There are many contributions based on this approach:

Crowley, 1985, navigation for an intelligent mobile robot, IEEE J. of Robotics and Automation, pp 31-41


Fabrizi et al., An integrated sensing guidance system for robotized wheelchair, Proc. Of the 14th IFAC World Congress, pp. 457-462

…
A simple classification:

<table>
<thead>
<tr>
<th></th>
<th>Accuracy</th>
<th>Complexity</th>
</tr>
</thead>
<tbody>
<tr>
<td>Topological</td>
<td>low</td>
<td>low</td>
</tr>
<tr>
<td>Gaussian</td>
<td>high</td>
<td>low</td>
</tr>
<tr>
<td>Pose grid</td>
<td>medium</td>
<td>high</td>
</tr>
<tr>
<td>MCM</td>
<td>high</td>
<td>medium</td>
</tr>
</tbody>
</table>

Kalman filter
Ancona: September 2005
Kalman Filter for the pose tracking of mobile robot

- Different set of sensors
- Feature model of the environment (the metric map it is assumed to be known)
- Different hypotheses of the measurement errors
- Experimental validation
Kalman filter

Linear system with state noise $w_k \sim N(0, Q_k)$ and measurement noise $v_k \sim N(0, R_k)$ (independent)

$$x_k = Fx_{k-1} + Gu_{k-1} + w_{k-1}$$
$$z_k = Hx_k + v_k$$

Prediction: a priori estimation of the state and of the error covariance

$$\hat{x}_{k|k-1} = F\hat{x}_{k-1|k-1} + Gu_{k-1}$$
$$P_{k|k-1} = FP_{k-1|k-1}F^T + Q_{k-1}$$
Updating: the measure $z_k$ is used to correct the estimate

$$S_k = HP_{k|k-1}H^T + R_k$$

covariance of the innovation process

$$K_k = P_{k|k-1}H^T S_k^{-1}$$

filter gain

$$\nu_k = z_k - H\hat{x}_{k|k-1}$$

innovation process

$$\hat{x}_{k|k} = \hat{x}_{k|k-1} + K_k\nu_k$$

$$P_{k|k} = (I - K_kH)P_{k|k-1}$$


Bar-Shalom and Fortman, 1988, Tracking and Data Association, Accademic Press.
Initialization: the initial state and initial value of the error covariance matrix $P_{0|0}$

These choices depend on the available a priori knowledge on the considered estimation problem.
On the **Kalman filter** it is assumed known a priori the time-invariant system dynamics, the measurement equations, the model of the noise.

The **Kalman filter** it one of the most important results of the use of state space methods in modern control theory.

Under the assumptions introduced in the Kalman filter (linearity and Gaussian distributions) the **Kalman filter** is equivalent to the maximum likelihood, Bayesian, least square, and minimum mean-square error estimations.

This solution can be also extended to the non-linear case (Extended Kalman Filter) by a simple linearization technique.

In this case no results exist on the filter convergence.

The performance depends on the accuracy and significance of the linear model used.

For the problems here considered: the robot model is not linear (for example the kinematic model), in this case a linearization technique can be introduced and this case the solution of the estimation problem is based on a Extended Kalman Filter.
Non linear model:

\[ \mathbf{x}_k = f(\mathbf{x}_{k-1}, \mathbf{u}_{k-1}, \mathbf{w}_{k-1}) \]
\[ \mathbf{z}_k = h(\mathbf{x}_k, \mathbf{v}_k). \]

defining:

\[ F_k = \left. \frac{\partial f}{\partial \mathbf{x}_k} \right|_{\hat{\mathbf{x}}_{k-1|k-1}, \mathbf{u}_{k-1}, 0} \]
\[ W_k = \left. \frac{\partial f}{\partial \mathbf{w}_k} \right|_{\hat{\mathbf{x}}_{k-1|k-1}, \mathbf{u}_{k-1}, 0} \]

Prediction: a priori estimation of the state and of the error covariance

\[ \hat{\mathbf{x}}_{k|k-1} = f(\hat{\mathbf{x}}_{k-1|k-1}, \mathbf{u}_{k-1}, 0) \]
\[ P_{k|k-1} = F_k P_{k-1|k-1} F_k^T + W_k Q_{k-1} W_k^T \]
Updating: the measure \( z_k \) is used to correct the estimate

\[
H_k = \left. \frac{\partial h}{\partial x_k} \right|_{\hat{x}_{k|k-1,0}} \\
V_k = \left. \frac{\partial h}{\partial v_k} \right|_{\hat{x}_{k|k-1,0}}
\]

\[
S_k = H_k P_{k|k-1} H_k^T + V_k R_k V_k^T
\]

\[
K_k = P_{k|k-1} H_k^T S_k^{-1}
\]

\[
\nu_k = z_k - h(\hat{x}_{k|k-1,0})
\]

\[
\hat{x}_{k|k} = \hat{x}_{k|k-1} + K_k \nu_k
\]

\[
P_{k|k} = (I - K_k H_k) P_{k|k-1}
\]
A note:
in this algorithm the linear model is time-varying along the state trajectory and therefore also the filter equations have time-varying matrices!!!
Data association:

For determining the origin of the measurement, the object of the measurement

Some sensor readings must be rejected; when the measurements are referred to objects of the environment that are not modelled in the map and therefore such measurements can not used for the localization (unknown obstacles, furniture, …)

\[ \rho_k = \nu_k S_k^{-1} \nu_k^T \]

(Mahalanobis distance) the difference between the measure \( z_k \) and its expected value \( h(x_{klk+1,0}) \)

\[ \rho_k \leq \gamma \]

Bar-Shalom and Fortmann, 1988, Tracking and Data Association, Academic Press
Approaches to localization using the Kalman filter

✓ Adaptive estimation of robot location through an Adaptive Extended Kalman Filter integrating odometric and sonar measures


✓ An alternative adaptive approach, where the adaptation algorithm is based on fuzzy logic

Jetto et al., 1999, Localization of a wheeled mobile robot by sensor data fusion based on a fuzzy logic adapted Kalman filter, Control Engineering Practice, pp. 763-771

✓ Integration of odometric, FOG and laser scanner measures; the high accuracy of gyroscope measures gives rise to a singular filtering problem

Ippoliti et al., A properly designed extended Kalman filtering approach for robot localization by sensors with different degree of accuracy, Proc. of the fourth International Symposium on Robotics an Automation (ISRA 2004), Queretaro, Mexico, 2004
Adaptive Extended Kalman Filter

The performance of the filter is improved by an on line adjustment of the input and measurement noise covariances obtained by a suitably defined estimation algorithm.

Data fusion algorithms can provide a reliable estimate of the robot location exploiting the advantages offered by different and complementary sensors (in this case, incremental encoders for relative localization and sonar sensors for absolute localization).

A data fusion of odometric and sonar measures is performed through an Adaptive Extended Kalman Filter (AEKF).

A sonar reading selection is introduced.
Unicycle-like Mobile Robot

Localization requires knowledge of:

- coordinates $x$ and $y$ of the midpoint between the two driving wheels
- angle $\theta$ between the main axis of the robot and the $X$-direction

The kinematic model of the unicycle vehicle is described by the equations:

\[
\begin{align*}
\dot{x}(t) &= \nu(t) \cos \theta(t) \\
\dot{y}(t) &= \nu(t) \sin \theta(t) \\
\dot{\theta}(t) &= \omega(t)
\end{align*}
\]

$\nu(t)$: displacement velocity of the robot
$\omega(t)$: angular velocity of the robot
Odometric measures

Position and orientation of the robot:

\[
x(t_{k+1}) = x(t_k) + \bar{v}(t_k)\Delta t_k \frac{\sin \frac{\bar{\omega}(t_k)\Delta t_k}{2}}{\bar{\omega}(t_k)\Delta t_k} \cos \left( \theta(t_k) + \frac{\bar{\omega}(t_k)\Delta t_k}{2} \right),
\]
\[
y(t_{k+1}) = y(t_k) + \bar{v}(t_k)\Delta t_k \frac{\sin \frac{\bar{\omega}(t_k)\Delta t_k}{2}}{\bar{\omega}(t_k)\Delta t_k} \sin \left( \theta(t_k) + \frac{\bar{\omega}(t_k)\Delta t_k}{2} \right),
\]
\[
\theta(t_{k+1}) = \theta(t_k) + \bar{\omega}(t_k)\Delta t_k.
\]

\(\bar{v}(t_k)\Delta t_k\) and \(\bar{\omega}(t_k)\Delta t_k\) depend on the incremental distances, measured by the encoders, covered by the right and left wheels of the robot.

Really available values:

\[
y_v(t_k) = \bar{v}(t_k)\Delta t_k + n_v(t_k)
\]
\[
y_\omega(t_k) = \bar{\omega}(t_k)\Delta t_k + n_\omega(t_k)
\]

Measurement errors:

\[n_v(\cdot) \sim N(0, \sigma_v^2)\]
\[n_\omega(\cdot) \sim N(0, \sigma_\omega^2)\]
Sonar Measures

The distance readings by sonar sensors are related to the indoor environment model and to the configuration of the mobile robot.

Position of the $i$-th sonar referred to the inertial coordinate system:

\[
\begin{align*}
    x_i(t_k) &= x(t_k) + x'_i \sin \theta(t_k) + y'_i \cos \theta(t_k) \\
    y_i(t_k) &= y(t_k) - x'_i \cos \theta(t_k) + y'_i \sin \theta(t_k) \\
    \theta_i(t_k) &= \theta(t_k) + \theta_i
\end{align*}
\]
Sonar Measures

The walls and the obstacles in an indoor environment are represented by a proper set of planes orthogonal to the plane $X$-$Y$ of the inertial system.

Each plane is represented by the triplet:

- $P_r^j$: normal distance of the plane from the origin
- $P_n^j$: angle between the normal line to the plane and the $X$-direction
- $P_v^j \in \{-1, 1\}$ defines the face of the plane reflecting the sonar beam

Expected distance of the sonar from the plane:

$$d_i^j(t_k) = P_v^j(P_r^j - x_i(t_k) \cos P_n^j - y_i(t_k) \sin P_n^j)$$

$$P_n^j \in [\theta_i(t_k) - \delta/2, \theta_i(t_k) + \delta/2]$$
Adaptive estimation of robot location

Denote with:

- \( X(t) := [x(t) \, y(t) \, \theta(t)]^T \) the robot state
- \( U(t) := [\nu(t) \, \omega(t)]^T \) the robot control input

The kinematic model of the robot can be written in the following compact form:

\[
\begin{align*}
\frac{dX(t)}{dt} &= F(X(t), U(t)) dt + d\eta(t) \\
Z((k+1)T) &= G(X((k+1)T), \Pi) + v(kT)
\end{align*}
\]

\( d\eta(t)/dt \) is a white noise process \( \sim N(0, Q(t)) \)
\( v(kT) \) is a white sequence \( \sim N(0, R(kT)) \)

The dimension \( p_k \) of \( Z(kT) \) is not constant
Adaptive estimation of robot location

Linearization and discretization result in the following EKF:

\[
\hat{X}(k+1,k) = \hat{X}(k,k) + B_d(k)B(k)F(\hat{X}(k,k), U(k-1)) + B_d(k)(U(k) - U(k-1)),
\]

\[
P(k+1,k) = A_d(k)P(k,k)A_d^T(k) + Q_d(k),
\]

\[
K(k+1) = P(k+1,k)C^T(k+1)S(k+1)^{-1},
\]

\[
S(k+1) = [C(k+1)P(k+1)C^T(k+1) + R(k+1)],
\]

\[
\hat{X}(k+1,k+1) = \hat{X}(k+1,k) + K(k+1)\Gamma(k+1),
\]

\[
\Gamma(k+1) = [Z(k+1) - G(\hat{X}(k+1,k))],
\]

\[
P(k+1,k+1) = [I - K(k+1)C(k+1)]P(k+1,k),
\]
Adaptive estimation of robot location

... where

\[ A_d(k) = e^{A(k)T}, \]

\[ A(k) = \left[ \frac{\partial F(X(t), U(t))}{\partial X(t)} \right]_{X(t) = \hat{X}(k, k)}, \]

\[ B(k) = \left[ \frac{\partial F(X(t), U(t))}{\partial U(t)} \right]_{X(t) = \hat{X}(k, k), U(t) = U(k-1)}, \]

\[ C(k) = \left[ \frac{\partial G(X(k))}{\partial X(k)} \right]_{X(k) = \hat{X}(k, k-1)}, \]

\[ B_d(k) = \int_{kT}^{(k+1)T} e^{A(k)((k+1)T-\tau)} d\tau, \]

\[ Q_d(k) = \int_{kT}^{(k+1)T} e^{A(k)((k+1)T-\tau)} Q(\tau) e^{A^T(k)((k+1)T)} d\tau. \]
Adaptive estimation of robot location

The covariance matrices $Q(\tau), \tau \in [kT, (k+1)T]$ and $R(k)$ are assumed to have the following form:

\[
Q(\tau) = \sigma_\eta^2(k)I_3, \tau \in [kT, (k+1)T] \\
R(k) = \text{diag}[^2 \sigma_v, 1(k), \ldots, ^2 \sigma_v p_k(k)]
\]

From which

\[
Q_d(k) = \sigma_\eta^2(k)\bar{Q}(k), \\
\bar{Q}(k) = \int_{kT}^{(k+1)T} e(A(k)((k+1)T-\tau) \\
e(A^T(k)((k+1)T-\tau)) d\tau.
\]

so that $Q_d(k)$ is completely known save that for the multiplicative scalar $\sigma_\eta^2(k)$.
Adaptive estimation of robot location

Processing each measure one at a time results in

\[
\hat{X}(k+1, k+1) = \hat{X}(k+1, k) + K_i \gamma_i(k+1).
\]

\[
\gamma_i(k+1) = z_i(k+1) - G_i(k+1)(\hat{X}(k+1, k)).
\]

The final state estimate is given by

\[
\hat{X}(k+1, k+1) = \hat{X}(k, k) + B_d(k)B(k)F(\hat{X}(k, k), U(k-1)) + B_d(k)(U(k) - U(k-1)) + \sum_{i=1}^{P_k} K_i(k+1)\gamma_i(k+1).
\]
Adaptive estimation of $Q_d(k)$ and $R(k)$

The following assumptions are made:

1. The parameters $\sigma_{v,i}^2(k), i = 1, \ldots, p_k$ and $\sigma_\eta^2(k)$ are nearly constant over $n_v \geq 2$ and $n_\eta \geq 2$ samples respectively.

2. Residuals $\gamma_i(k + 1), i = 1, \ldots, p_k$ are called the innovation process samples and are assumed to be well described by a white sequence $\sim \mathcal{N}(0, s_i(k+1))$,

$$s_i(k+1) = C_i(k+1)[A_d(k)P(k,k)A_d^T(k)$$

$$+ \sigma_\eta^2(k)\bar{Q}(k)]C_i^T(k+1) + \sigma_{v,i}^2(k)$$

The parameters $\sigma_\eta^2(k)$ and $\sigma_{v,i}^2, i = 1, \ldots, p_k$ are estimated imposing the condition of consistency between the innovation process samples and their predicted statistics.
Adaptive estimation of $Q_d(k)$ and $R(k)$

As the $p_k$ measures are processed one at a time, at each step, $p_k$ estimates $\hat{\sigma}^2_{\eta, i}(k)$ of the same parameter $\sigma^2_{\eta, i}(k)$ are obtained.

They are determined by the operation $\max \frac{1}{\epsilon_i(k + 1) \times \gamma_i(k + 1)^2 \gamma_i(k + 1) \gamma_i(k + 1)}$

One has

$$\hat{\sigma}^2_{\eta, i}(k) = \max \left\{ \epsilon_i(k + 1) \times \left[ \gamma_i(k + 1)^2 \gamma_i(k + 1) \gamma_i(k + 1) \right] \right\}$$

where

$$\epsilon_i(k + 1) = \frac{1}{C_i(k + 1) \bar{Q}_d(k) C_i^T(k + 1)}$$
Adaptive estimation of $Q_d(k)$ and $R(k)$

The following recursive smoothed estimate is obtained:

$$\tilde{\sigma}_{\eta}(k) = \tilde{\sigma}_{\eta}(k-1) + \frac{1}{(l_{\eta} + 1)p_k} \times \left[ \sum_{i=1}^{p_k} \left( \tilde{\sigma}_{\eta,i}(k) - \tilde{\sigma}_{\eta,i}(k-(l_{\eta} + 1)) \right) \right]$$

Analogously, \( \max \text{ prob} \sigma_{v,i}(k+1) \geq \gamma_i(k+1) \)

The following recursive smoothed estimate is obtained:

$$\tilde{\sigma}_{v,i}(k+1) = \tilde{\sigma}_{v,i}(k) + \frac{1}{l_v + 1} (\tilde{\sigma}^2(k+1) - \hat{\sigma}^2(k-l_v))$$

where

$$\hat{\sigma}_{v,i}(k+1) = \max \left\{ \gamma_i^2(k+1), \right.$$

$$\left. -(C_i(k+1)A_d(k)P(k,k)A_d^T(k)C_i^T(k+1) + \tilde{\sigma}_{\eta,i}(k)Q(k)), 0 \right\},$$
Sonar Reading Selection

Reduce the probability of an inadequate interpretation of erroneous sensor data

- Interferences produced by
  - presence of unknown obstacles in the environment
  - uncertainty on the sensor readings

Compare the actual sensor readings with their expected values

\[ \gamma_i(k) = z_{3+i}(k) - \hat{d}_i \]
\[ \gamma_i(k) \sim N(0, s_i(k)) \]

The current value \( z_{3+i}(k) \) is accepted if \( |\gamma_i(k)| \leq 2\sqrt{s_i(k)} \)

- Variable threshold chosen as twice the standard deviation of the innovation process
Experimental Results

The experimental tests have been performed on the LabMate mobile base in an indoor environment with different geometries.

This mobile robot has two driving wheels.
**Control algorithm**

The control algorithm is based on a proper discrete-time implementation of the control algorithm based on the kinematic inversion approach

\[
\begin{align*}
\nu(t) &= K\nu(y_d(t)\cos\theta(t) - x_d(t)\sin\theta(t)) + C\nu e(t), \\
\omega(t) &= K\omega(\theta_d(t) - \theta(t)) + C\omega e_n(t),
\end{align*}
\]

where

\[
\begin{align*}
e(t) &= (y_d(t) - y(t))\cos\theta(t) - (x_d(t) - x(t))\sin\theta(t), \\
e_n(t) &= (y_d(t) - y(t))\sin\theta(t) + (x_d(t) - x(t))\cos\theta(t).
\end{align*}
\]

The system is connected directly with the robot low level controller and the proximity system by a standard serial port RS232

The proposed localization algorithm has been tested with relatively long trajectories in an indoor environment completely represented by a suitable set of planes orthogonal to the plane XY of the inertial system
Experimental Results

In order to test the limitations of odometric measures, the planned trajectory is composed of a large set of orientation changes.
Experimental Results

Pose tracking
Ancona: September 2005
Experimental Results

The adaptation algorithm is able to reduce the initial position error to zero by better exploiting the information carried by sonar readings in correspondence of an obstacle.
Experimental Results

If an unknown obstacle which is not introduced in the environment map is placed at the side of a wall, the corresponding readings are seen as undesired interferences, and are eliminated by the proposed procedure.
Fuzzy Adaptive Extended Kalman Filter

In this case the adaptive estimation of noise statistics is based on a fuzzy logic approach.

The principles of fuzzy expert systems lead to a valid on-line adaptive algorithm which improves the performance of the Extended Kalman Filter.
Adaptive estimation of $Q_d(k)$

Warning: poor estimates of $Q_d(k)$ may lead to filter divergence.

The heuristic adaptation algorithm is based on rules:

$$\text{IF } <\text{antecedent}> \text{ THEN } <\text{consequent}>$$

A fuzzy implication between the parameter $\sigma^2_{\eta}(k)$ and a parameter relative to the expected value of the sample:

$$r(k+1) = y(k+1) - C(k+1)\hat{X}(k+1, k)$$

In the linear case $r(k+1)$ is called the “innovation process” and is a white discrete sequence $\sim \mathcal{N}(0, \hat{S}(k+1))$, where

$$\hat{S}(k+1) = C(k+1)[A_d(k)P(k, k)A_d(k)^T + \sigma^2_{\eta}(k)\bar{Q}(k)]C^T(k+1) + R(k+1)$$
Adaptive estimation of $Q_d(k)$

An heuristic extension of these properties to the linearized model:

a good estimate $\hat{Q}_d(k)$ of $Q_d(k)$ produces innovation process samples consistent with their statistics

In other words:

If the innovation process sample $r(k+1)$ is observed and its absolute value $|r(k+1)|$ is within the range predicted from the theory, this means that the current value of $\hat{Q}_d(k)$ is large enough

On the contrary, if $|r(k+1)|$ is so large to exceed the predicted range, then $\hat{Q}_d(k)$ should be increased (as a consequence, also $P(k,k)$ and $K(k)$ are increased and filter divergence is prevented)
Adaptive estimation of $Q_d(k)$

This suggests the following adaptation mechanism:

If $|r(k + 1)|$ is neither too small nor too large, then leave $\hat{Q}_d(k)$ nearly unchanged

If it is very small, then reduce $\hat{Q}_d(k)$

If it is large, then increase $\hat{Q}_d(k)$
A Fuzzy Logic Approach

Compute:  \[ \beta(k + 1) := \frac{\bar{r}(k + 1)}{\bar{s}(k + 1)} \]

\[ \bar{r}(k + 1) := \frac{1}{p_k} \sum_{i=1}^{p_k} |r_i(k + 1)| \]

\[ \bar{s}(k + 1) := \frac{1}{p_k} \sum_{i=1}^{p_k} \tilde{s}_{i,k}(k + 1)^{1/2} \]

At each step, parameter \( \sigma_\eta^2(k + 1) \) is estimated from \( \beta(k + 1) \) by:

1. If \( \beta(k + 1) \) is very small, then \( \sigma_\eta^2(k + 1) \) is small enough
2. If \( \beta(k + 1) \) is small, then \( \sigma_\eta^2(k + 1) \) is near to 1
3. If \( \beta(k + 1) \) is neither small nor large, then \( \sigma_\eta^2(k + 1) \) is a bit larger than 1
4. If \( \beta(k + 1) \) is large enough, then \( \sigma_\eta^2(k + 1) \) is somewhat larger than 1
5. If \( \beta(k + 1) \) is very large, then \( \sigma_\eta^2(k + 1) \) is very large
A Fuzzy Logic Approach

The universes of the discourse relative to $\beta(k + 1)$ and $\sigma^2_\eta(k + 1)$ are $U_\beta$ and $U_{\sigma^2_\eta}$.

The characteristic functions of $\beta(k + 1)$ and $\sigma^2_\eta(k + 1)$ are $\mu_\beta$ and $\mu_{\sigma^2_\eta}$.

The discretization of the universes and the shape of the characteristic functions have been fixed:

- on the basis of the above mentioned heuristic knowledge that the probability density function of $r(k + 1)$ can be well approximated by a normal density.
- and on the basis of a wide working expertise on understanding the phenomenon of filter divergence.
A Fuzzy Logic Approach

Representation of fuzzy sets:
A Fuzzy Logic Approach

To simplify the defuzzification procedure

- The characteristic function $\mu\sigma^2_\eta$ of $\sigma^2_\eta(k + 1)$ has been considered piecewise constant over intervals $\Delta = 0.01$
- The numerical value $\sigma^2_\eta(k + 1)$ has been computed according to the center of area method:

$$\sigma^2_\eta(k + 1) = \frac{1}{2} \sum_{i=1}^{n} \mu\sigma^2_{\eta,i} \frac{x_i^2 - x_{i-1}^2}{\sum_{i=1}^{n} \mu\sigma^2_{\eta,i} (x_i - x_{i-1})}$$

$x_{i-1}$ and $x_i$ are the extreme points of the $i$-th interval of amplitude $\Delta$

$\mu\sigma^2_{\eta,i}$ is the constant value of the characteristic function $\mu\sigma^2_\eta$ on the interval

$n$ is the number of intervals
A Fuzzy Logic Approach

To avoid excessive oscillations of the numerical value of \( \sigma^2(\eta)(k+1) \), the smoothed estimate is computed:

\[
\bar{\sigma}^2(\eta)(k+1) = \bar{\sigma}^2(\eta)(k) + \frac{1}{l+1}(\sigma^2(\eta)(k+1) - \sigma^2(\eta)(k-l))
\]

where \( l \) is a positive integer.

The final estimate is

\[
\hat{Q}(k+1) = \bar{\sigma}^2(\eta)(k+1)\bar{Q}(k)
\]
Sonar Reading Selection  (same solution …)

Reduce the probability of an inadequate interpretation of erroneous sensor data

✓ Interferences produced by
✓ presence of unknown obstacles in the environment
✓ uncertainty on the sensor readings

Compare the actual sensor readings with their expected values

\[ \gamma_i(k) = z_{\frac{2}{3}+i}(k) - \hat{d}_i \quad \gamma_i(k) \sim N(0, s_i(k)) \]

The current value \( z_{\frac{2}{3}+i}(k) \) is accepted if \( |\gamma_i(k)| \leq 2\sqrt{s_i(k)} \)

✓ Variable threshold chosen as twice the standard deviation of the innovation process
Experimental Results

In order to test the limitations of odometric measures, the planned trajectory is composed of a large set of orientation changes.

- Localization with only odometric measures
- Localization with AEKF
- Localization with FLAEKF

real trajectory
planned trajectory

actually used sonar measures (based on the sonar reading selection algorithm)
Nearly Singular Filtering

Different sensors are characterized by different accuracies

Some measures could be more reliable than others

When fusing the measures in the Kalman filter these aspects must be considered: the measure with an high reliability could introduce problems of singularity in the Kalman filter

For example if we combine (fuse) in the Kalman filter odometric measures with FOG measures (characterized by an high reliability), the matrix $R$ is singular and in this case a lower order non singular EKF can be derived
Nearly Singular Filtering

The algorithm operates in a stochastic framework, and is based on the ascertainment that the angular measure $\theta_g(t_k)$ provided by the FOG is much more reliable than the angular measure $y_\omega(t_k)$ obtainable by encoders.

The estimation algorithm is an EKF defined on the basis of:

- a stochastic state-space model derived from:
  - the kinematic model of the unicycle vehicle
- a measure equation containing:
  - the incremental measure of the encoders $y_\nu(t_k)$
  - the angular measure of the gyroscope $\theta_g(t_k)$
  - distance measures provided by the set of sonar sensors from environment planes
FOG Measures

The Fiber Optic Gyroscope (FOG) installed on the robot produces an accurate estimation of robot orientation.

The FOG readings are denoted by:

\[ \theta_g(\cdot) = \theta^r_g(\cdot) + n_\theta(\cdot) \]

\( n_\theta(\cdot) \sim N(0, \sigma^2_\theta) \)
Stochastic State-Space Model

State vector: \[ X(t) = [X_1(t)^T, \theta(t)]^T \text{ where } X_1(t) = [x(t), y(t)]^T \]

Measurement vector: \[ Z(k) := [Z_1(k)^T, Z_2(k)^T]^T \]

\[ Z_1(k) = [z_1(k), z_2(k)]^T \]

✓ Measure provided by the encoders: \[ z_1 = \bar{\nu}(t_k) \Delta t_k + n_{\nu}(t_k) \]

✓ Measure provided by the FOG: \[ z_2 = \theta_r^g(\cdot) + n_\theta(\cdot) \]
Odometric-inertial localization vs mere odometry

A localization algorithm which integrates odometric and inertial measures instead of using odometry only performs much better when the robot trajectory is long and winding, which is shown by the comparison of these figures:
Laser Measures

The distance readings by laser scanner are related to the indoor environment model and to the configuration of the mobile robot

Position of the laser scanner referred to the inertial coordinate system:

\[ x_s(t_k) = x(t_k) + l \cos \theta(t_k) \]
\[ y_s(t_k) = y(t_k) + l \sin \theta(t_k) \]
\[ \theta_s(t_k) = \theta(t_k) \]
Laser Measures

The walls and the obstacles in an indoor environment are represented by a proper set of planes orthogonal to the plane X-Y of the inertial system.

Each plane is represented by the triplet:
- $P_r^j$: normal distance of the plane from the origin
- $P_n^j$: angle between the normal line to the plane and the X-direction
- $P_p^j \in \{-1, 1\}$ defines the face of the plane reflecting the laser beam

Expected distance of the laser center from the plane:

$$d_i^j(t_k) = \frac{P_r^j(P_r^j - x_s(t_k)\cos P_n^j - y_s(t_k)\sin P_n^j)}{\cos \theta_d^j}$$
Stochastic State-Space Model

State vector: \( X(t) = [X_1(t)^T, \theta(t)]^T \) where \( X_1(t) = [x(t), y(t)]^T \)

Measurement vector: \( Z(k) := [Z_1(k)^T, Z_2(k)^T]^T \)

\[ Z_1(k) = [z_1(k), z_2(k)]^T \]

✓ Measure provided by the encoders: \( z_1 = \bar{v}(t_k) \Delta t_k + n_v(t_k) \)

✓ Measure provided by the FOG: \( z_2 = \theta^r_g(\cdot) + n_\theta(\cdot) \)

\[ Z_2(k) = [z_3(k), z_4(k), \cdots, z_{2+n_s}(k)]^T \]

✓ Measure provided by the \( i \)-th laser reading: \( z_{2+i}(k) = d_i^j(k) + n_{r,i}(k) \)
Stochastic State-Space Model

Observation noise: \( V(k) = [n_\nu(k), n_\theta(k), n_{r,1}(k), \ldots, n_{r,n_s}(k)] \)

- White sequence \( \sim N(0, R) \)
- Covariance matrix of the independent errors affecting the sensor measures

\[ R := \text{block diag } [R_1, R_2] \]
- \( R_1 := \text{diag} [\sigma^2_\nu, \sigma^2_\theta] \)
- \( R_2 := \text{diag} [\sigma^2_{r,1}, \sigma^2_{r,2}, \ldots, \sigma^2_{r,n_s}] \)

The measure \( z_2(\cdot) \) provided by the FOG is much more reliable than \( z_1(\cdot) \) and \( z_{2+i}(\cdot) \)

- \( \sigma^2_\theta << \sigma^2_\nu, \sigma^2_\theta << \sigma^2_{r,i} \) \( \Rightarrow \) singularity of \( R \) \( \Rightarrow \) \( z_2(\cdot) = \theta^r(\cdot) \)
- Lower order non singular EKF assuming \( \sigma^2_\theta = 0 \)

Only \( X_1(\cdot) \) is estimated as a function of \( z_1(\cdot) \) and \( Z_2(\cdot) \)

Kalman Filter

Ancona: September 2005
Stochastic State-Space Model

The measure provided by the encoders is in terms of increments. The extended state is defined as:
\[ \tilde{X}(k) := [X_1(k)^T, X_1(k-1)^T]^T \]

The linearized and discretized dynamic state-space equation is:
\[ \tilde{X}(k+1) = \tilde{A}(k) \tilde{X}(k) + \tilde{L}(k)U(k) + \tilde{B}(k)\theta^r_g(k) + \tilde{D}(k) + \tilde{W}(k) \]  

Measure equations:
- \[ z_1(k) = C_1(k) \tilde{X}(k) + v_1(k) \]  
  expressed as a function of \( x(k+1) - x(k) \) and \( y(k+1) - y(k) \)
- Linearization of \( Z_2(k) \)
  \[ \tilde{Z}_2(k) = C_2(k) \tilde{X}(k) + V_2(k), \quad C_2(k) := [c_1(k)^T, c_2(k)^T, \ldots, c_{\tilde{p}_k}(k)^T]^T \]

Equations (a), (b) and (c) represent the linearized and discretized state-space form to which the classical EKF algorithm has been applied.
Laser Scanner Reading Selection

Reduce the probability of an inadequate interpretation of erroneous sensor data

- Interferences produced by
  - presence of unknown obstacles in the environment
  - uncertainty on the sensor readings

Compare the actual laser sensor readings with their expected values

\[ \gamma_i(k) = z_{2+i}(k) - \hat{d}_i \quad \gamma_i(k) \sim N(0, s_i(k)) \]

The current value \( z_{2+i}(k) \) is accepted if \( |\gamma_i(k)| \leq 2\sqrt{s_i(k)} \)

- Variable threshold chosen as twice the standard deviation of the innovation process
EKF Implementation

The EKF has been implemented on the navigation module of the TGR Explorer powered wheelchair.
EKF Implementation

The EKF has been implemented on the navigation module of the TGR Explorer powered wheelchair.
**Experimental Results**

- 115 m length trajectory
- Uncertainties on the environment model
- $T_C=0.4$ s

---

**Pose tracking**

Ancona: September 2005
Experimental Results

- 115 m length trajectory
- Uncertainties on the environment model
- $T_C = 0.4\ s$

EKF without laser measures

EKF with laser measures

Pose tracking
Ancona: September 2005
Experimental Results

- 115 m length trajectory
- Uncertainties on the environment model
- $T_C=0.4 \text{ s}$
The proposed algorithms show several applications of Kalman filtering to mobile robot localization, in situations where different kinds of sensory devices are available.

Adaptively estimating the unknown state noise statistics from data improves the convergence and performance of the Kalman filter.

When the available sensor accuracies differ significantly from each other, this fact must be suitably taken into account.

Tests have been performed in different known environments and suitable trajectories have been realized (also with different set of sensors and real-time implementations).

A suitable integration of proprioceptive and exteroceptive sensory data allows a highly accurate estimate of robot location.
Part 4: “A Kalman Filter for Simultaneous Localization And Map building”

Prof. Sauro Longhi

Dr. Alessia La Manna (PhD student)

Dipartimento di Ingegneria Informatica, Gestionale, dell’Automazione
Università Politecnica delle Marche
Ancona - Italy

sauro.longhi@univpm.it

Perception and Sensor Fusion in Mobile Robotics
Ancona: September 2005
Simultaneous Localization and Map Building (SLAM)

It is an “open problem”: (I) for the localization of a mobile robot a map of the environment is required; (II) for the map building it is necessary to have a good estimate of the robot pose.

Two approaches are mainly investigated:

i) Extended Kalman Filter and “stochastic map” (feature-based SLAM)

ii) Probabilistic approach (particle filter)
In the following our approach to the feature-based SLAM is presented and discussed. Also for this problem the main tool is the Extended Kalman Filter (EKF).

\[ \text{Robot} \rightarrow \text{Localization system} \rightarrow \text{Map building} \rightarrow \text{Measurement system} \rightarrow \text{z} \]

Background:


- EKF and landmarks (laser and vision system)

- EKF and map defined by landmarks (outdoor application with range sensor)

Gamini et al., 2001, Map management for efficient simultaneous localization and mapping (SLAM), Autonomous robots, pp. 267-286

(Outdoor application)


- EKF and landmarks (laser and vision system)

Zunino and Christensen, 2001, navigation in realistic environment, Proc. SIRS

- EKF and stochastic map (sonar)

Folkesson, Jensfelt, Christensen, Vision SLAM and the Measurement Subspace, Proc. ICRA, Barcelona, April 2005
Methodologies for SLAM

Technologies for SLAM
Our approaches to SLAM

- The robot pose and environment feature locations can be simultaneously estimated using odometric, inertial and sonar range measures
  
  Ippoliti et al., Improving the robustness properties of robot localization procedures with respect to environment features uncertainties, Proc. of ICRA 2005, Barcelona, Spain, pp. 1463-1470

- Sonar measures can be used not only to provide distance information, but also to obtain an occupancy grid map; a probabilistic method is proposed for fusing the features extracted from it with data provided by a video camera, thus improving the performance with respect to an approach using odometric and sonar range measures only

  Bonci et al., Sonar and video data fusion for robot localization and environment feature estimation, proc. of CDC-ECC’05, Seville, Spain
Simultaneous estimation of robot location and environment features

State-space model whose state vector contains both the state variables of the wheelchair model and the state variables of landmarks

Algorithm for the simultaneous on-line estimation of both the vehicle position and the environment features

Computationally efficient solution to the localization problem
Simultaneous estimation of robot location and environment features

The algorithm operates in a stochastic framework, and is based on the ascertainment that the angular measure $\theta_g(t_k)$ provided by the FOG is much more reliable than the angular measure $y_\omega(t_k)$ obtainable by encoders.

The estimation algorithm is an EKF defined on the basis of:

- a stochastic state-space model derived from:
  - the kinematic model of the unicycle vehicle
  - environment geometric features
- a measure equation containing:
  - the incremental measure of the encoders $y_\nu(t_k)$
  - the angular measure of the gyroscope $\theta_g(t_k)$
  - distance measures provided by the set of sonar sensors from environment planes
Simultaneous estimation of robot location and environment features

The measure \( z_1(\cdot) \) provided by the encoders is in terms of increments.

Extended state \( \tilde{X}(k) := [X_1(k)^T, X_1(k-1)^T]^T \)

\( \tilde{X}(k) := [\tilde{X}^T(k), X'^T(k)]^T \)

where \( X'(k) := [P^1_r, P^1_n, P^2_r, P^2_n, \ldots, P^n_{r^p}, P^n_{n^p}]^T \)

Dynamic state-space equation:

\[
\tilde{X}(k + 1) = \tilde{A}(k)\tilde{X}(k) + \tilde{L}(k)U(k) + \tilde{B}(k)\theta^r_g(k) + \tilde{D}(k) + \tilde{W}(k) \quad (a)
\]

Measure equations:

\[
z_1(k) = \tilde{C}_1(k)\tilde{X}(k) + v_1(k), \quad \tilde{C}_1(k) = [C_1(k), \ 0_{1,2n^p}] \quad (b)
\]

expressed as a function of \( x(k+1) - x(k) \) and \( y(k+1) - y(k) \)

\[
\tilde{Z}_2(k) = \tilde{C}_2(k)\tilde{X}(k) + V_2(k), \quad \tilde{C}_2(k) = [\tilde{c}_1(k)^T, \ \tilde{c}_2(k)^T, \ \ldots, \ \tilde{c}_{p_k}(k)^T]^T \quad (c)
\]

Equations (a), (b) and (c) represent the linearized and discretized state-space form to which the classical EKF algorithm has been applied.
Experimental results

The EKF has been implemented on the navigation module of the TGR Explorer powered wheelchair.
Experimental results

Trajectory of the vehicle estimated by the proposed algorithm

<table>
<thead>
<tr>
<th></th>
<th>$x$</th>
<th>$y$</th>
<th>$\theta$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$AC$</td>
<td>0.91 m</td>
<td>0.145 m</td>
<td>6.2832 rad</td>
</tr>
<tr>
<td>$EC$</td>
<td>0.8935 m</td>
<td>0.1446 m</td>
<td>6.2873 rad</td>
</tr>
<tr>
<td>$E$</td>
<td>0.0165 m</td>
<td>0.0004 m</td>
<td>0.0041 rad</td>
</tr>
</tbody>
</table>

Detected estimate errors ($E$) in correspondence of the final vehicle configuration = distance between the actual ($AC$) and the corresponding estimated ($EC$) configuration
Experimental results

- The continuous lines are the actual (true) locations of environment map landmarks.
- Dotted and dashed lines represent the initial and final estimates of environment map landmarks respectively.
- The dots are the actually used sonar measures.

<table>
<thead>
<tr>
<th></th>
<th>$L_1$</th>
<th>$L_2$</th>
<th>$L_3$</th>
<th>$L_4$</th>
<th>$L_5$</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>IE</strong></td>
<td>0.12 m</td>
<td>0.105 m</td>
<td>0.11 m</td>
<td>0.095 m</td>
<td>0.095 m</td>
</tr>
<tr>
<td><strong>FE</strong></td>
<td>0.0475 m</td>
<td>0.0476 m</td>
<td>0.07 m</td>
<td>0.0523 m</td>
<td>0.034 m</td>
</tr>
</tbody>
</table>

Differences in module between the actual locations of environment landmarks and their initial (IE) and final (FE) estimates.

A typical estimate (continuous line) of environment geometric feature location.

- As the vehicle progresses through the environment, the error in landmark location estimate decreases monotonically to a lower bound.
- The dashed line represents the actual environment geometric feature location.
Data Fusion

The performance of the considered approach to SLAM can be improved by the use of more reliable measures.

For example the reliability of exteroceptive measures could be increased by the data fusion of complementary sensors, as considered in the problem of map building: data fusion between sonar readings and video data.
The CCD video camera can be used for extracting the walls and obstacle borders, which are straight lines in the $XY$ plane.

For detecting straight lines representing environment features, the Hough Transform (HT) is used.

**Straight line equation:**

\[ \rho = u \cos \phi + v \sin \phi \]

- $\rho$ : distance between the straight line and the origin of the coordinate system in the image plane
- $\phi$ : orientation of the line
- $u, v$ : coordinates of whatever edge point belonging to the line

The edge points are detected by an orthogonal differential operator.

For each detected edge point the parameters $(\rho, \phi)$ are estimated and quantized, and the accumulator array $H(\rho, \phi)$ is increased accordingly. Eventually, the peaks of $H(\rho, \phi)$ identify the parameters of the highest probability lines.
**Occupancy grid map**

The values of the cells in the grid are updated by a probabilistic fusion of sonar readings.

Straight line segments can be found in the occupancy grid as aligned cells with high probability of occupation.

By interpreting a grid and its probabilities as an image with different levels of intensity (grey level), it is possible to apply the HT to detect straight lines and to associate a probability to each detected line.
Video and ultrasonic data fusion

Matching between the lines extracted from the video data and from occupancy grid is performed for environment features belonging to the part of the floor visible from the CCD camera (here called “visible space”)

The proposed matching algorithm is based on the combination of the line probabilities encoded in both Hough accumulators

All the straight lines detected from the portion of the occupancy grid corresponding to the portion of floor falling in the “visible space” are matched with the straight lines detected from the video image by projecting the straight lines of the video image on the floor plane

The existence probability of the straight lines is stored in both Hough accumulators

The matching algorithm is based on the combination of the line probabilities encoded in both Hough accumulators; a Bayesian estimator is considered

The idea is to validate the straight lines detected by the vision system with the sonar probabilistic map
Also data video measures are included in the measurement equation. Vector $Z(k)$ is composed of three subvectors:

- $Z_1(k) = [z_{o,1}(k), z_{o,2}(k), z_{o,3}(k)]^T$ (odometric measures)

$$z_{o,1}(k) = x(k) + v_{o,1}(k) \quad z_{o,1}(k) = \hat{x}(k-1|k-1) + \Delta x(k)$$

$$z_{o,2}(k) = y(k) + v_{o,2}(k) \quad z_{o,2}(k) = \hat{y}(k-1|k-1) + \Delta y(k)$$

$$z_{o,3}(k) = \theta(k) + v_{o,3}(k) \quad z_{o,3}(k) = \hat{\theta}(k-1|k-1) + \Delta \theta(k)$$

- $Z_2(k) = [z_{f,1}(k), z_{f,2}(k), \ldots, z_{f,2n_{fm}}(k)]^T$ (data fusion measures)

$$z_{f,2i-1}(k) = P_r^i(k) + v_{f,2i-1}(k) \quad z_{f,2i-1}(k) = \rho_i(k)$$

$$z_{f,2i}(k) = P_n^i(k) + v_{f,2i}(k) \quad z_{f,2i}(k) = \phi_i(k)$$

- $Z_3(k) = [z_{s,1}(k), z_{s,2}(k), \ldots, z_{s,n_{sm}}(k)]^T$ (sonar measures)

The observation noise is a white sequence with zero mean and covariance $R$, where $R$ is a diagonal matrix.

Fusion measure selection is analogous to the procedure previously recalled.
Ultrasonic and video data fusion

- Black lines: actual (true) locations of environment map landmarks
- Cyan lines: initial estimates of environment map landmarks
- Magenta lines: final estimates of environment map landmarks
- Green dots: actually used sonar measures
- Red lines: actually used fusion measures

The algorithm using fusion data performs better than the one using sonar range measures only, especially when the initial estimate of landmarks differs from the real value by about 1 m.
Ultrasonic and video data fusion

Even in a less structured environment the algorithm using fusion data shows satisfactory performances, while the one using sonar measures only is unable to correct an initial erroneous estimate of landmark locations.
Comments

- The proposed algorithms for SLAM operate in a stochastic state-space form where sensor and model uncertainties are intrinsically taken into account.

- The vehicle trajectory estimate is highly accurate and robust with respect to uncertain physical parameters and not exactly known initial conditions.

- All available information and uncertainties are collected into the state-space representation.

- The solution has a recursive structure and requires a modest computational effort.