Mobile Robots and Autonomous Vehicles

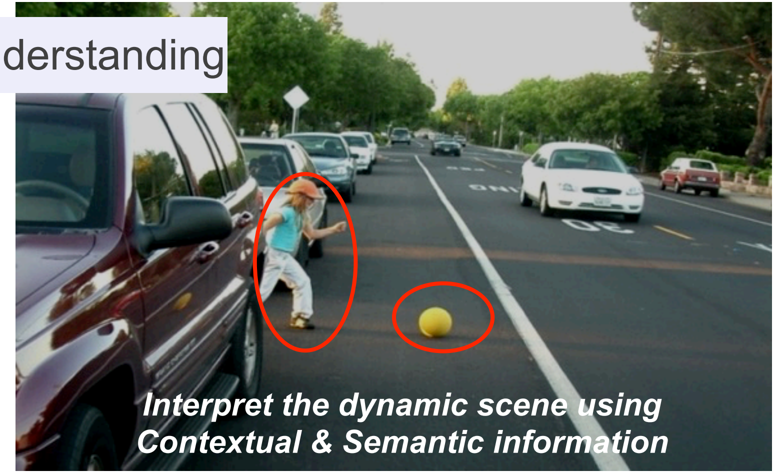
1. Objectives, Challenges, State of the Art, Technologies
2. Bayes & Kalman Filters
3. Extended Kalman Filter, Observability properties
4. **Perception & Situation Awareness & Decision Making**
5. Behavior Modeling & Learning

W4. Perception & Situation Awareness & Decision making

- **Robot Perception for Dynamic environments: Outline & DP-Grids concept**
- Dynamic Probabilistic Grids – Bayesian Occupancy Filter concept
- Dynamic Probabilistic Grids – Implementation approaches
- Object level Perception functions (SLAM + DATMO)
- Detection and Tracking of Mobile Objects – Problem & Approaches
- Detection and Tracking of Mobile Objects – Model & Grid based approaches
- Embedded Bayesian Perception & Short-term collision risk (DP-Grid level)
- Situation Awareness – Problem statement & Motion / Prediction Models
- Situation Awareness – Collision Risk Assessment & Decision (Object level)

Perception & Situation Awareness in Dynamic Human Environments – *Reminder of main features (see week 1)*

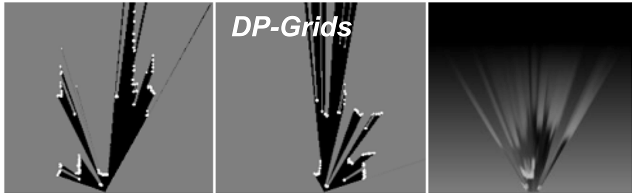
Traffic scene understanding



New Models & Algorithmic Tools

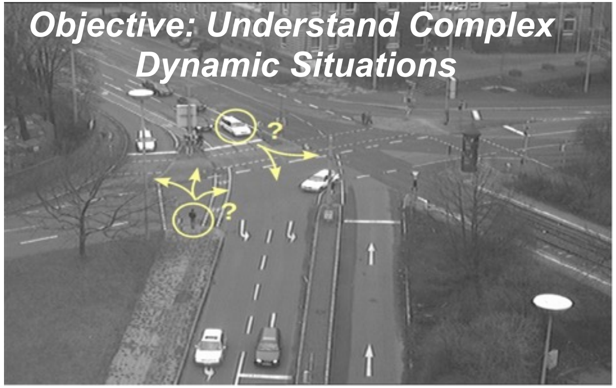
- Dynamicity & Uncertainty
  - ➔ *Space & Time + Probabilities*
- Interpretation ambiguities
  - ➔ *History, Context & Semantics, Prior knowledge + Sensor fusion*
- Prediction of future states
  - ➔ *Behaviors, prediction models*
- Real-time Embedded Perception
  - ➔ *Miniaturization & Software / Hardware integration*

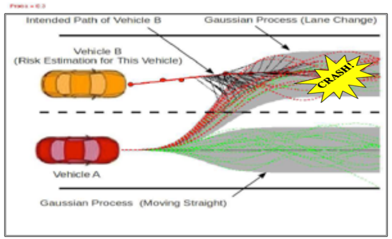
Process 1 – Embedded Bayesian Perception



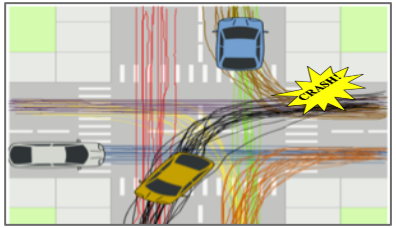
1. **Monitor the Dynamic Environment & Robot States** using **on-board sensors** *(Vision, Stereo Vision, Lidar, Radar, IMU, GPS, Odometry…)*
2. **Perform data fusion** of multiple sensors by means of ***"Dynamic Probabilistic Grids" (DP-Grids)*** *=> see sessions 2 & 3*
3. **Process dynamic scenes in real time** to ***Detect & Localize & Track & Classify*** multiple moving objects *=> see sessions 4 & 5 & 6*
4. **Improve efficiency** by ***Software / Hardware integration*** *… while reducing size & cost & energy consumption factors* *=> see session 7*

Process 2 – Situation Awareness & Risk Assessment





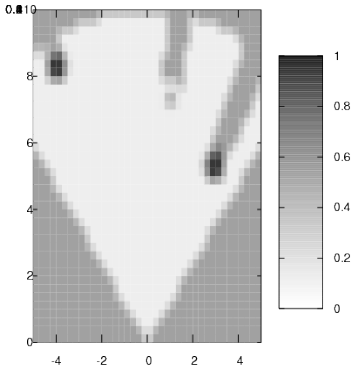
Prediction & Risk



1. **Model & Learn scene participants Behaviors** using the *"Learn & Predict paradigm" => see week 5*
2. **Predict future scene changes & Evaluate Collision Risks** using *Stochastic variables & Motion models & Trajectories prediction => see sessions 8 & 9*
3. **Prevent future collisions** (Alarm / Vehicle control) using various strategies *=> see sessions 9*

Dynamic Probabilistic Grids: *DP-Grids Principle*

Basic idea: *Combining two approaches*

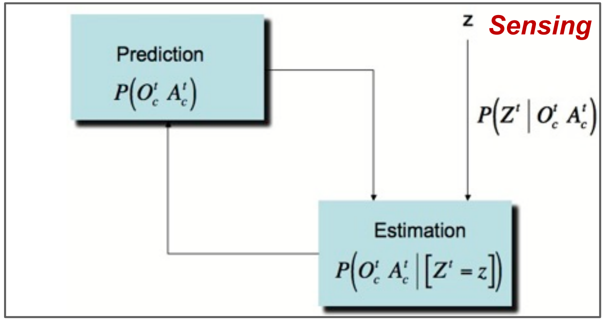


- **Occupancy Grid** for representing ***uncertain static*** environments *[Moravec 89]*

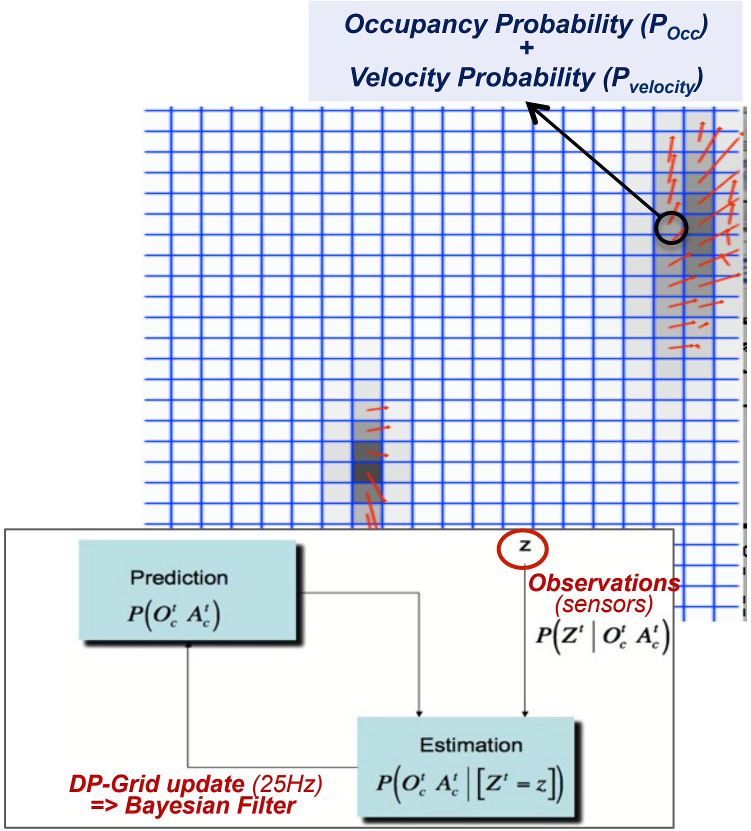
  => Each cell has a probability of being occupied by an object (or part of an object)

- **Bayesian Filter** for updating world states in ***uncertain dynamic*** environments

  => Grid Update using ***Sensor data & Prediction / Estimation loop using***



DP-Grids – *Bayesian Occupancy Filter approach (BOF)*



- **Processing Dynamic Environments using *DP-Grids***
  ➔ *Occupancy & Velocity Probabilities*
- ***DP-Grid Update*: *Bayesian Filter*** *(Inference using Probabilistic Sensor & Dynamic Models)*
  ➔ *More robust to Sensing Errors & Temporary Occultation*
- **Highly parallel processing**
  ➔ *Hardware implementation: GPU, Many-core architecture, SoC …*

[Coué et al IJRR 05] [Laugier et al ITSM 2011]

DP-Grids: *The Sensor Model*

- **Sensors data:**
  - *Incomplete (objects states are only partially measurable)*
  - *Uncertain (measures are noisy)*

- **The sensor model**

  => Modeling the relationship between ***objects true states*** *and the* ***corresponding observations*** *made by sensors*

- **Probabilistic representation** *(Thrun 2005)*

  => Inverse sensor model: $P\,(Z_t \mid S_t)$

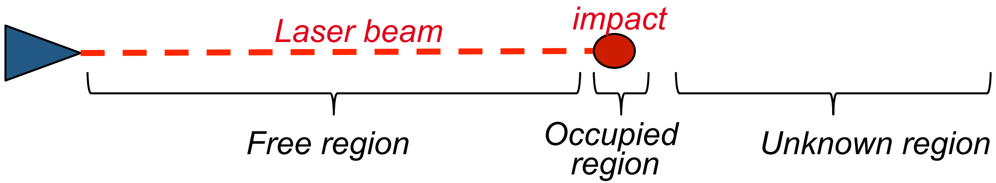
  Observation given by the sensor

  Object is at state S_t

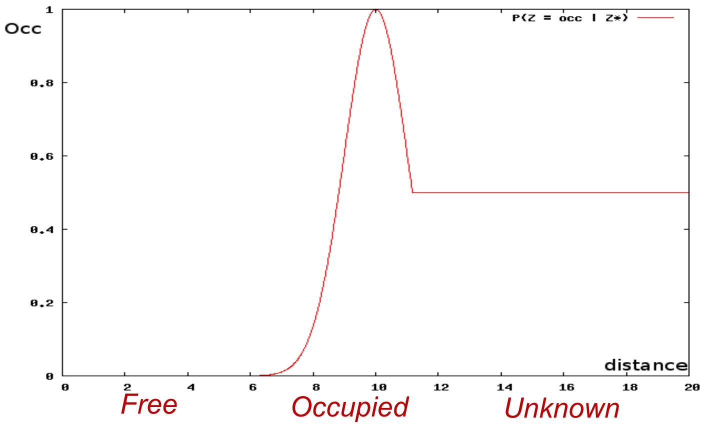
[Probabilistic Robotics, Thrun 2005]

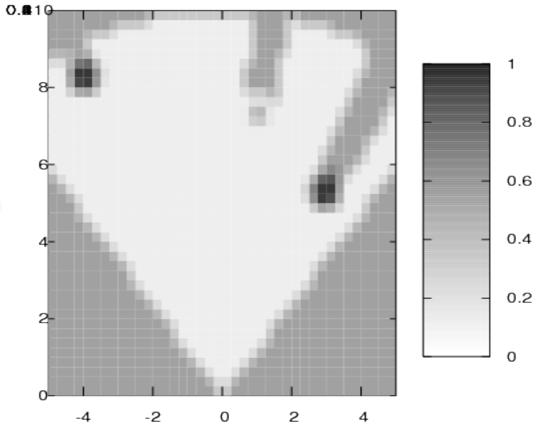
DP-Grids: *The Lidar Sensor Model*

- **Basic idea:** The Laser beam split the space in 3 regions (Free, Occupied, unknown)



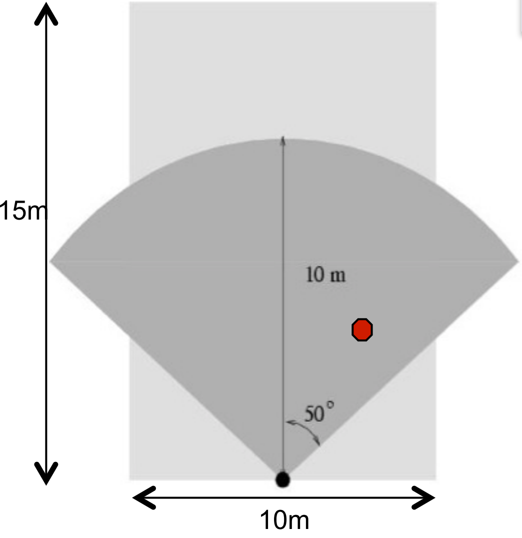
- **Probabilistic modeling:** The independent beam model



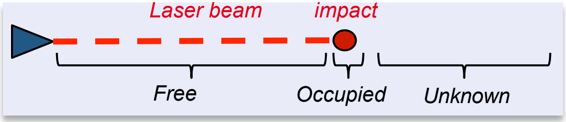


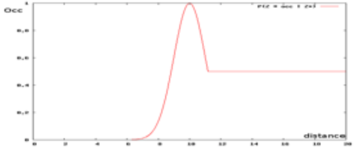
[Probabilistic Robotics, Thrun 2005]

DP-Grids example



1 Sensor (laser scanner)
1 Stationary object





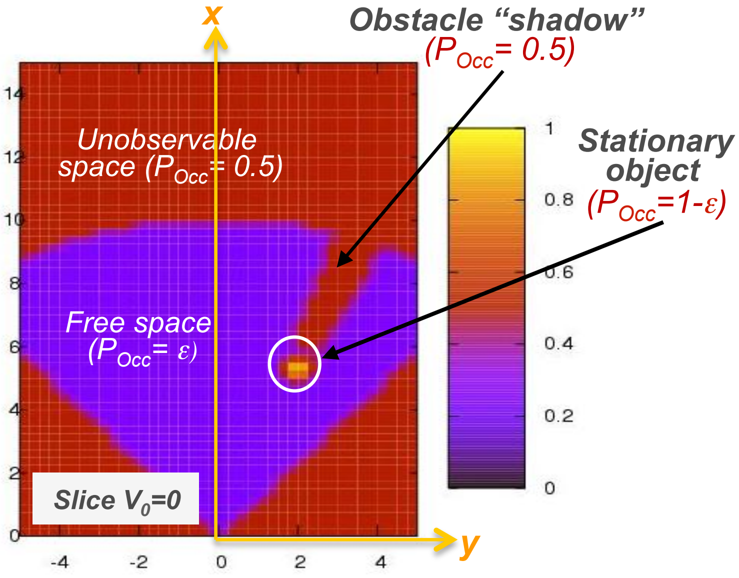
Independent beam model

Sensor observation: z = (5, 2, 0, 0)

Cell characterization: c = [x, y, 0, 0]

Position *Velocity (v_x, v_y)*

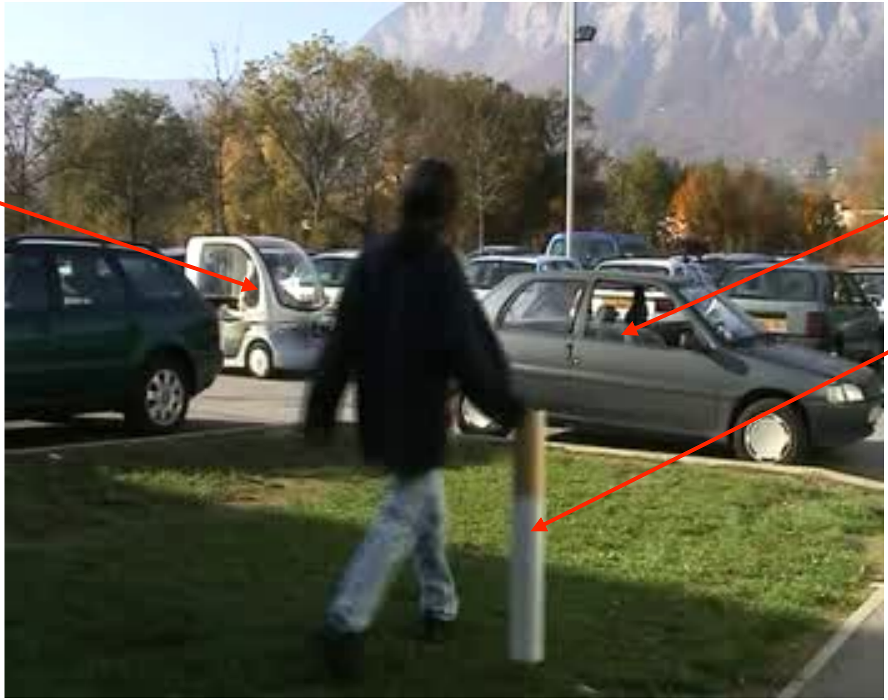
$$P_{Occ} = P([O_c=occ] \mid z \; c)$$



Underlying Conservative Prediction Capability

➔ *Application to Conservative Collision Anticipation*

Autonomous Vehicle (Cycab)



Parked Vehicle (occultation)

Beacon to improve detection accuracy (detection issue not addressed here)

Thanks to the conservative prediction capability of the BOF technology, the Autonomous Vehicle **"anticipates"** the behavior of the pedestrian and brakes (even if the pedestrian is temporarily hidden by the parked vehicle)

[Coue et al IJRR 05]

Dynamic Probabilistic Grids: *Practical implementation*

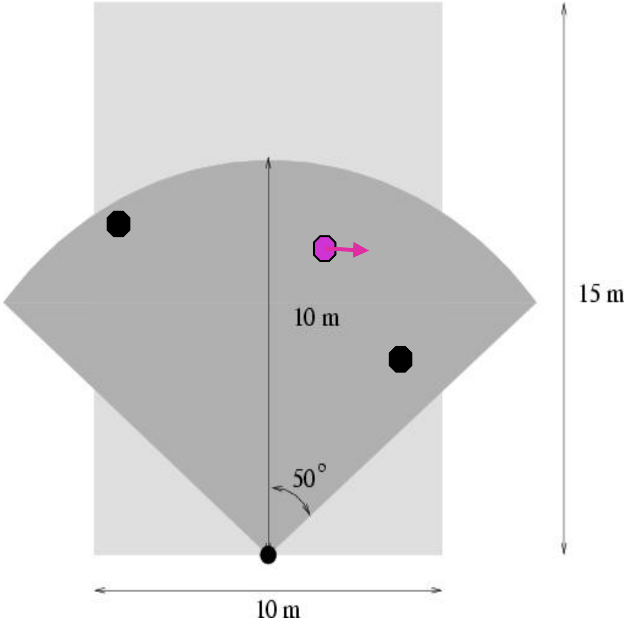
DP-Grids implementation will be illustrated using patented approaches

- ***Bayesian Occupancy Filter (BOF)***
  [Coué et al IJRR 05] [Laugier et al ITSM 2011]

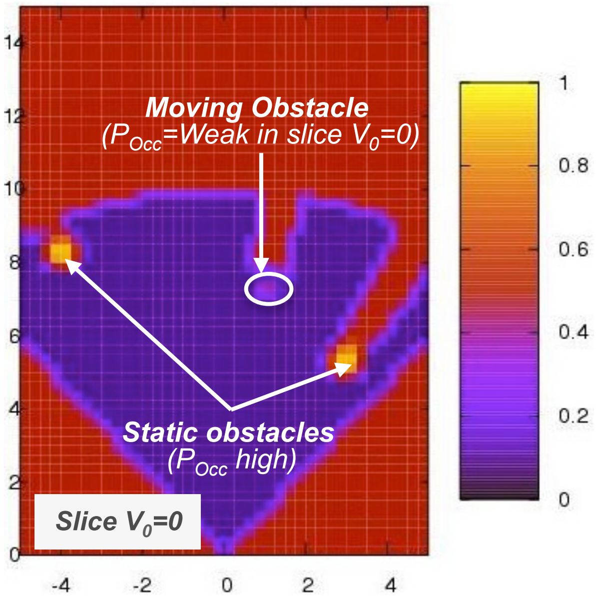
- ***Hybrid Sampling BOF (HSBOF)***
  [Negre et al 2014]

=> *More details in sessions 2 & 3*

DP-Grids example 2



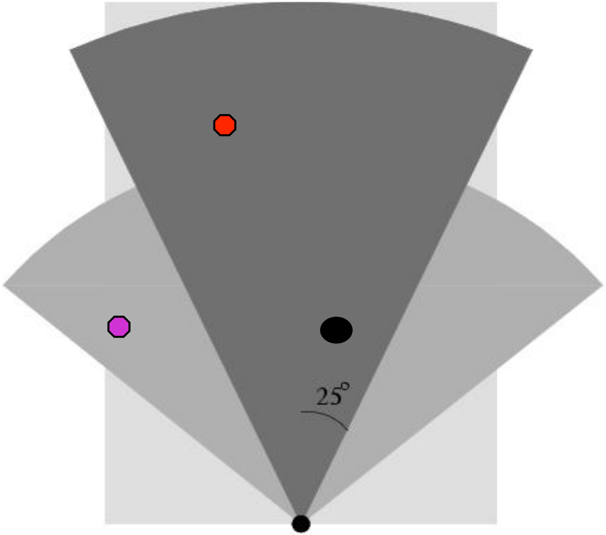
1 Sensor
2 Stationary objects
1 Moving object



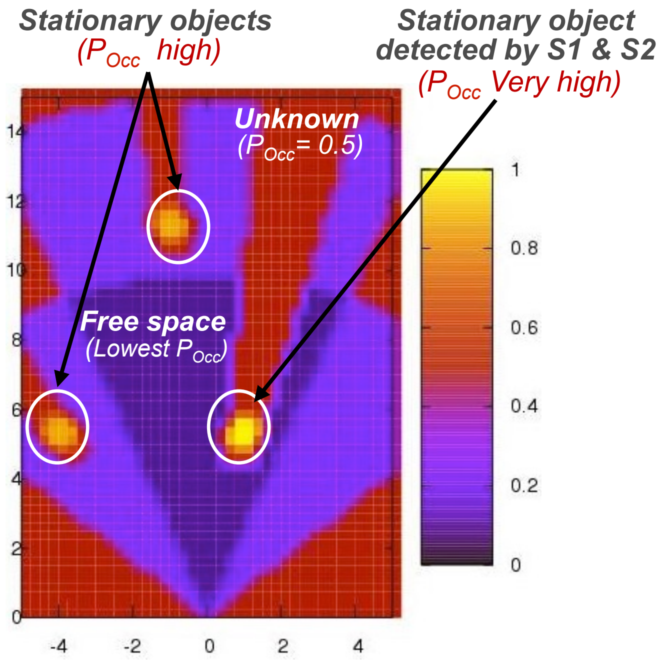
$$P_{Occ} = P([O_c=occ] \mid z_1\ z_2\ z_3\ \ c)$$

z_1 = (8.3, -4, 0, 0)
z_2 = (7.3, 1.9, 0, 0.8)
z_3 = (5, 3, 0, 0)
c = [x, y, 0, 0] => in velocity slice V_0=0

Dynamic Probabilistic Grids: *DP-Grids example 2*



2 Sensors (S1 & S2)
3 Stationary objects
=> Black object detected by S1 & S2



$$P_{Occ} = P([O_c = occ] \mid z_{1,1}\ z_{1,2}\ z_{2,1}\ z_{2,2}\ \ c\,)$$

$z_{1,1} = (5.5, -4, 0, 0)$ $\mathbf{z_{1,2} = (5.5, 1, 0, 0)}$
$z_{2,1} = (11, -1, 0, 0)$ $\mathbf{z_{2,2} = (5.4, 1.1, 0,0)}$
$c = [x, y, 0, 0]$