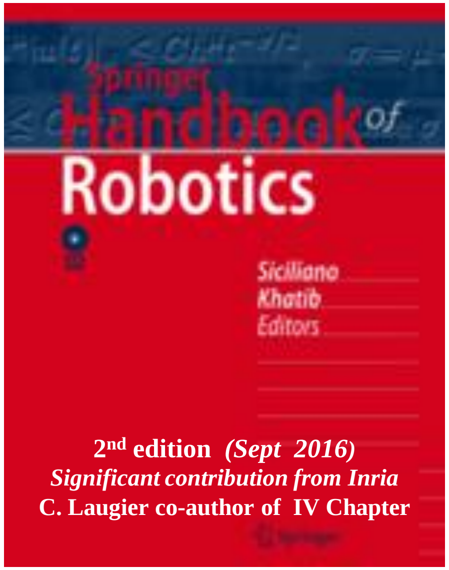
Lecture on Bayesian Perception & Decision-making for Autonomous Vehicles and Mobile Robots

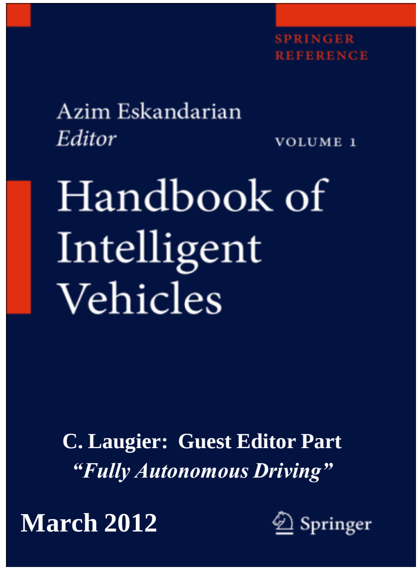
Christian Laugier

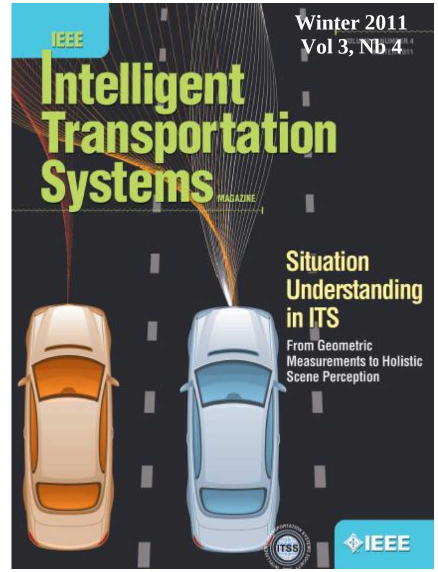
Lecture on
Bayesian Perception & Decision-making for Autonomous Vehicles and Mobile Robots

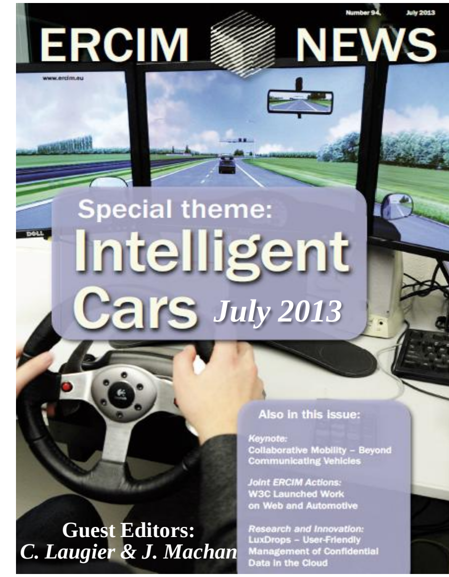
Dr. HDR Christian LAUGIER, Research Director at Inria
INRIA Grenoble Rhône-Alpes, Chroma team
Christian.laugier@inria.fr

Lecture at Beijing Institute of Technology
Intelligent Vehicle Research Center, School of Mechanical Engineering
May 8th 2017

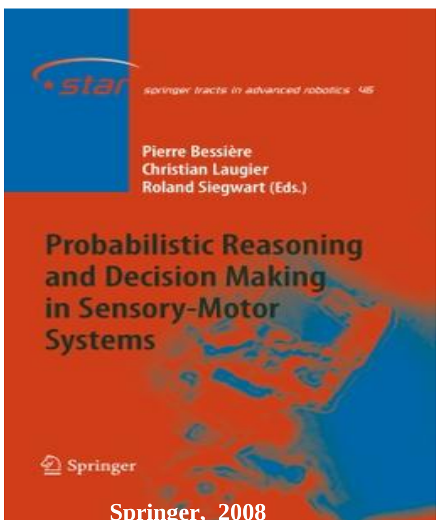


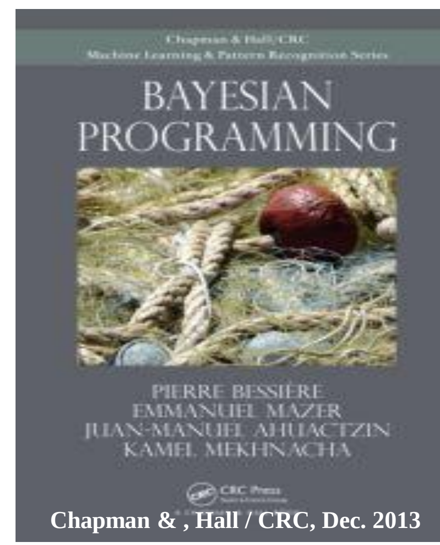






Relevant Literature on Robotics & IV & ITS





Content of the Lecture

- ❑ **Socio-economic & Technological Context**
- ❑ Decisional & Control Architecture: Outline => Not presented !
- ❑ Bayesian Perception *(Key Technology 1)*
- ❑ Embedded Perception & Experimental results
- ❑ Bayesian Risk Assessment & Decision-making *(Key Technology 2)*
- ❑ Conclusion & Perspectives

Cars & Human Mobility
A Psychological & Technological evolution

A quick on-going change of the role & concept of **private car** in human society !

Ownership & Feeling of Freedom
Affective behaviors & Shown Social position
Driving pleasure ... but less and less true !

Next cars generation *=> Focus on* ***Technologies*** *for*
Safety & Comfort & Reduced Pollution
Driving Assistance v/s Autonomous Driving

- **Context**
  - ⇒ *Expected 3 Billions vehicles & 75% population in cities in 2050 (current model not scalable !!!)*
  - ⇒ *Accidents: ~1.2 Million fatalities/Year in the world*
  - ⇒ *Driving safety & Nuisance issues (pollution, noise, traffic jam, parking ...) are becoming* ***a major issue for both Human Society & Governments & Industry***

Cars & Human Mobility
A Psychological & Technological evolution

A quick on-going change of the role & concept of **private c**

Ownership & Feeling of Freedom
Affective behaviors & Shown Soci
Driving pleasure … but less c

*…eration => Focus on **Technologies** for*
…ety & Comfort & Reduced Pollution
…riving Assistance v/s Autonomous Driving

Towards less ownership & More shared mobility systems
&
Increased Autonomy

❖ Technology & … hange de **mobility habits** of people
=> Share… pooling, more ADAS & Autonomy (e.g. Tesla autopilot)
N… uCar, Robot Taxis (Uber, Nutonomy)…

❖ … ket for Automotive Industry
*… in 2012 & Expected **$261 billions** in 2020 (f)*

(f) For… bal Market for ADAS Systems by 2020. ABI Research. 2013.

Autonomous Vehicles: 30 years R&D

Pioneer work at INRIA in the 90's : *Autonomous parking, Platooning in cities, People mover (Cycab)*

1986 VaMors (Dickmann Munich U)
First autonomous véhicle on a road,
Followed by EU project Prometheus

2011 Google Car project
Fleet of 6 automated Toyota Prius
140 000 miles covered on California roads
with occasional human interventions

2004 Darpa Grand Challenge
Significant step towards Motion Autonomy
… But still some uncontrolled behaviors !!!

2007 Darpa Urban Challenge
97 km, 50 manned & unmanned vehicles, 35 teams

Sustainable Mobility – *Cybercar technologies*

- **An EU driven concept since the 90's: "Cybercars"**

  ✓ ***Autonomous Self Service Urban & Green*** *Vehicles moving at* ***low speed***

  ✓ *Numerous R&D projects in Europe during the* ***past 25 years*** *(Several European* ***cities involved****)*

  ✓ *Some* ***Start-ups & Commercial products*** *for semi-protected areas (e.g. airports, industrial areas, amusement parks …), e.g. Robosoft & Easymile, 2GetThere , Induct & Neavia…*

- **Several early large scale public experiments in Europe**

Floriade 2002, Amsterdam
(2GetThere & Inria)

Cybus experiment, La Rochelle 2012
(CityMobil Project & Inria)

Autonomous Cars & Driverless Vehicles

- Strong involvement of Car Industry & Large media coverage
- An expected market of 500 B€ in 2035
- Numerous recent & on-going real-life experiments for validating the technologies

Tesla Autopilot based on Radar & Mobileye

Costly 3D Lidar & Dense 3D mapping

Cybus experiment, La Rochelle 2012
(CityMobil Project & Inria)

Drive Me trials
- *100 Test Vehicles in Göteborg, 80 km, 70km/h*
- *No pedestrians & Plenty of separations between lanes*

Driverless Taxi testing in Pittsburgh *(Uber)* & Singapore *(nuTonomy)*
=> Mobility Service, Numerous Sensors … Engineer in the car during testing

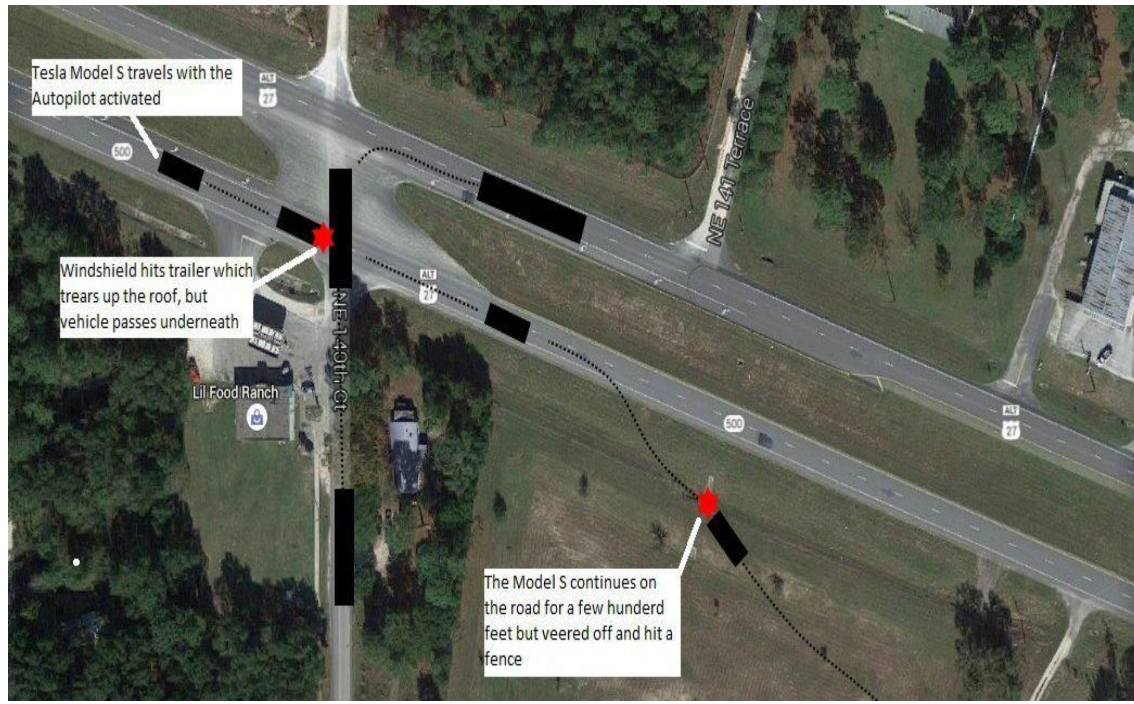
Several benign & serious accidents in past few months (Tesla, Google, Uber ..)

Safety is still not guaranteed !

Safety issues: *Example of the Tesla accident*

- **Safety is still insufficient** *(a false sense of Safety for users ?)*
  - *=> Still some Perception & Situation Awareness errors (even with commercial systems)*
  - *=> On May 7th 2016, **Tesla driver killed** in crash with Autopilot active*

Tesla Model S – Autopilot
Front perception:
Camera (Mobileye)+ Radar + US sensors



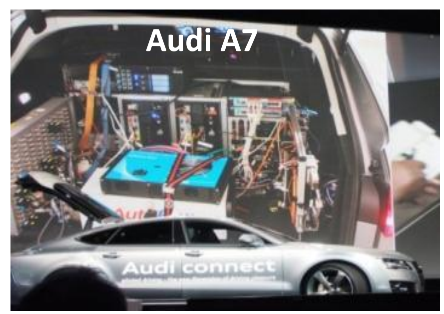
=> The Autopilot didn't detected the trailer as an obstacle *(see internet for available comments on NHTSA investigation & Tesla conjecture)*

Perception: State of the Art & Today's Limitations

- Despite significant improvements during the last decade of both Sensors & Algorithms, **Embedded Perception** is still one of the major bottleneck for Motion Autonomy

  => Obstacles detection & classification errors, incomplete processing of mobile obstacles, collision risk weakly address, scene understanding partly solved…

- **Lack of Robustness & Efficiency & Embedded integration** is still a significant obstacle to a full deployment of these technologies



Lack of Robustness & Efficiency

Lack of Integration into Embedded Sw/Hw

Trunk still full of electronics & computers & processor units
High computational capabilities are still required

Perception: Required system capabilities

Understanding Complex Dynamic Scenes

Situation Awareness & Decision-making

Dealing with unexpected events
e.g. Road Safety Campaign, France 2014

Anticipation & Prediction
for avoiding upcoming accidents

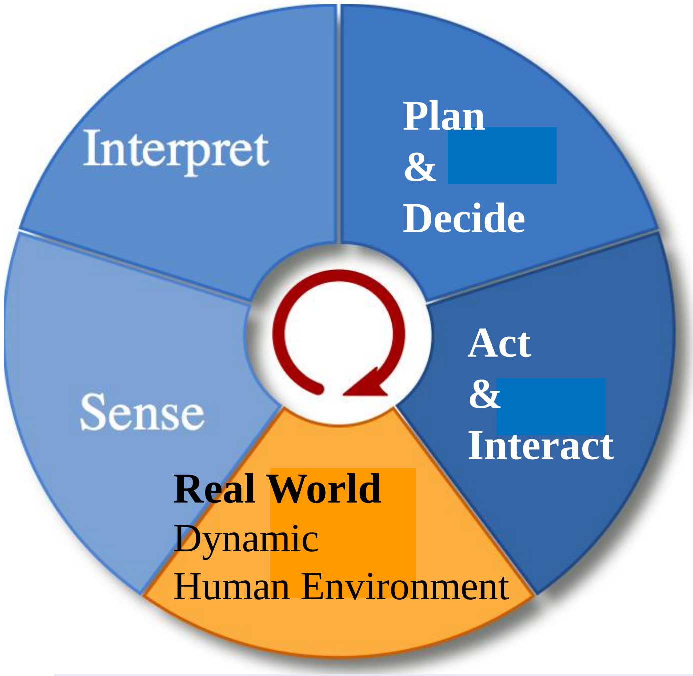
Main features

- ✓ Dynamic & Open Environments => *Real-time processing*
- ✓ Incompleteness & Uncertainty => *Appropriate Model & Algorithms (probabilistic approaches)*
- ✓ Sensors limitations => *Multi-Sensors Fusion*
- ✓ Human in the loop => *Interaction & Social Constraints (including traffic rules)*
- ✓ Hardware / Software integration => *Satisfying Embedded constraints*

Content of the Lecture

- ❑ Socio-economic & Technological Context
- ❑ **Decisional & Control Architecture : Outline**
- ❑ Bayesian Perception *(Key Technology 1)*
- ❑ Embedded Perception & Experimental results
- ❑ Bayesian Risk Assessment & Decision-making *(Key Technology 2)*
- ❑ Conclusion & Perspectives

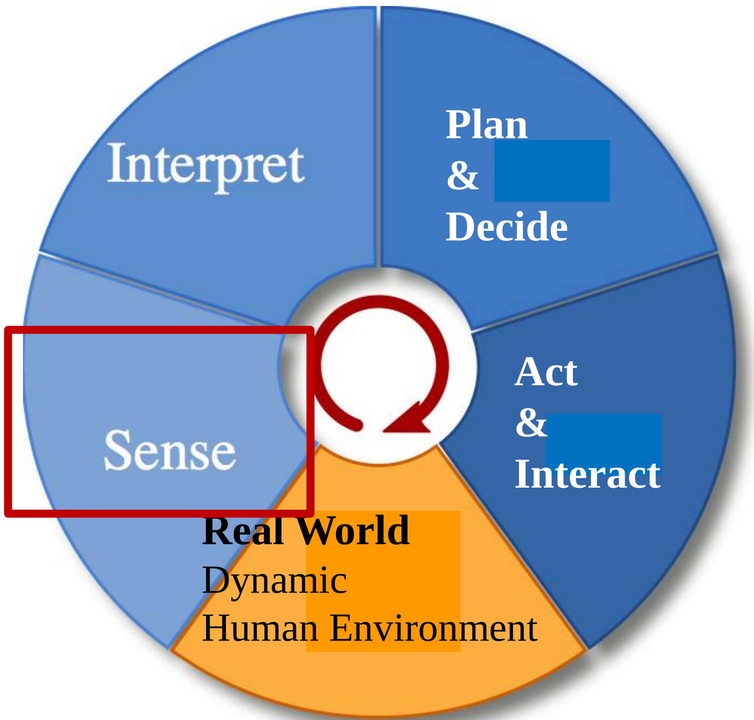
Decisional & Control Architecture (1)

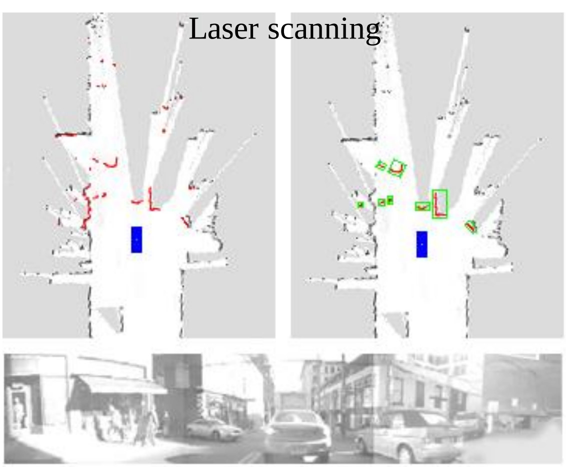


- ❑ **How to control Robot actions in a Dynamic world populated by Human Beings ?**

- ❑ **Combining & Adapting interdependent functions for:**
  - ✓ Sensing the environment using various sensors
  - ✓ Interpreting the dynamic scene (Semantics)
  - ✓ Planning Robot motions & Deciding of the most appropriate action to be executed
  - ✓ Acting & Interacting in the real world (Safety & Acceptability)

Decisional & Control Architecture (2)





❑ Objective

Perceive what is happening in the Dynamic Scene using various sensors

❑ Main Difficulty

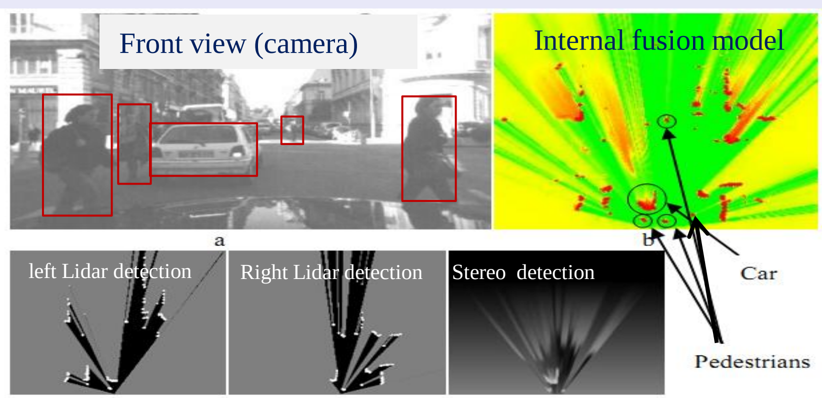
- ✓ Huge heterogeneous sensory data
- ✓ Sensing errors & Uncertainty
- ✓ Real-time processing

❑ Main Functions

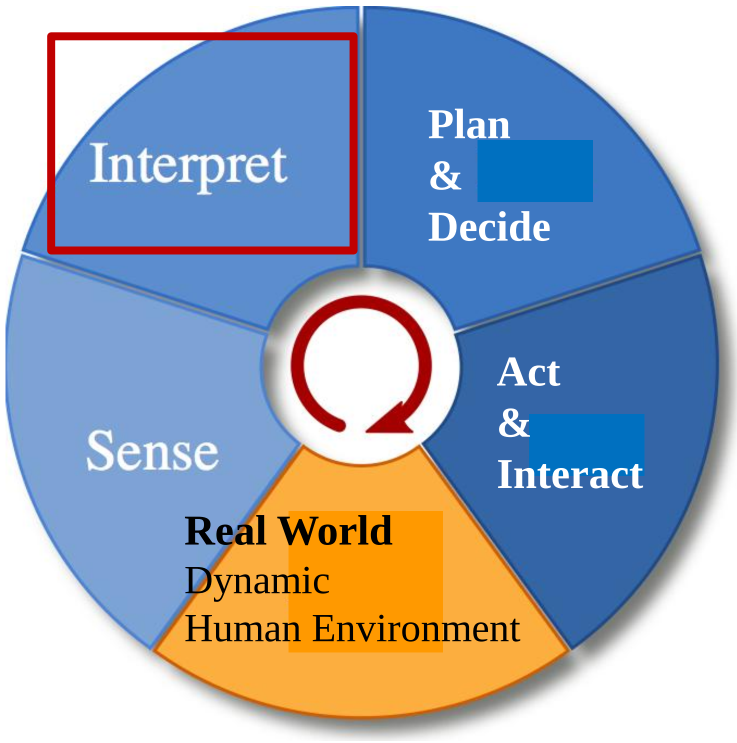
- ✓ Localization & Mapping (SLAM)
- ✓ Static & Mobile Objects Detection

❑ Main Models & Algorithms

- ✓ Bayesian Filtering
- ✓ Feature based & Grid based approaches



Decisional & Control Architecture (3)

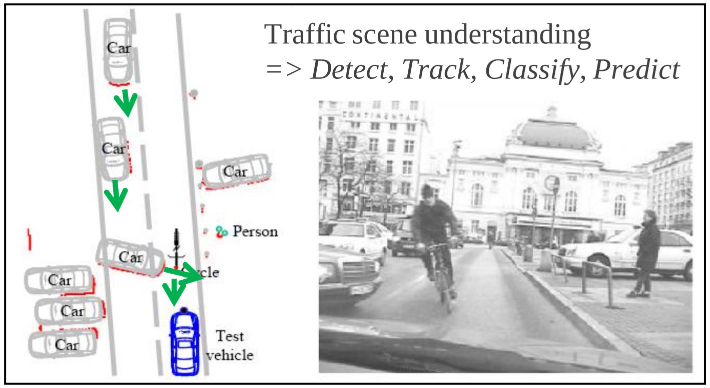


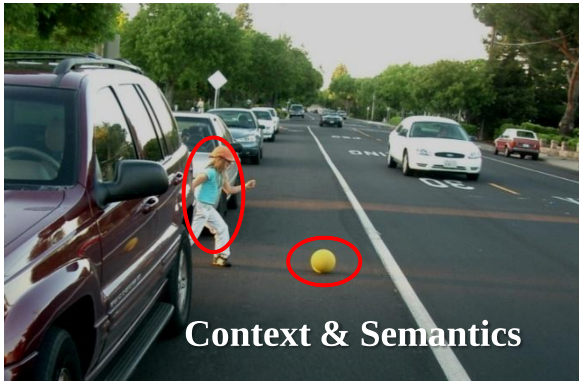
❑ Objective

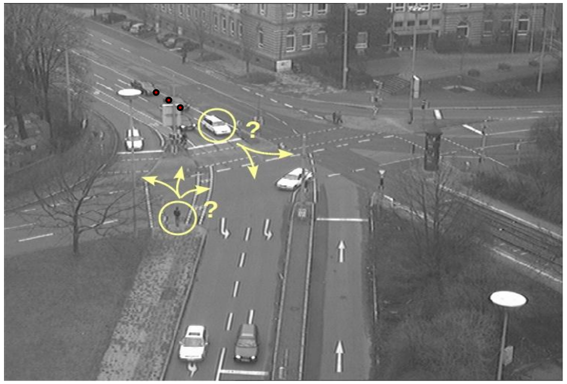
Understand the content of the Dynamic Scene using **Contextual & Semantic knowledge**

❑ Main Difficulty

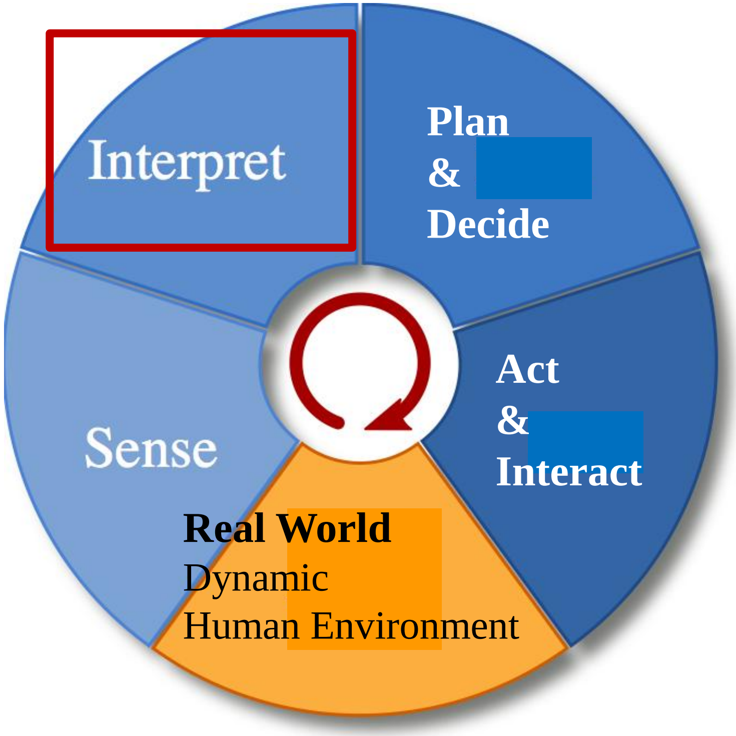
- ✓ Uncertainty
- ✓ Real-time processing
- ✓ Reasoning about various knowledge (history, context, semantics, prediction)







Decisional & Control Architecture (3 bis)

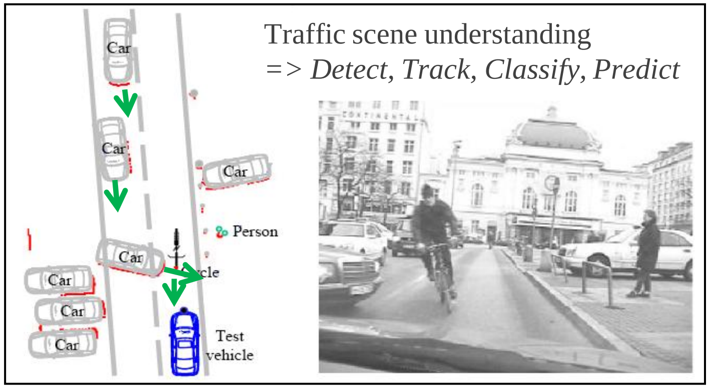


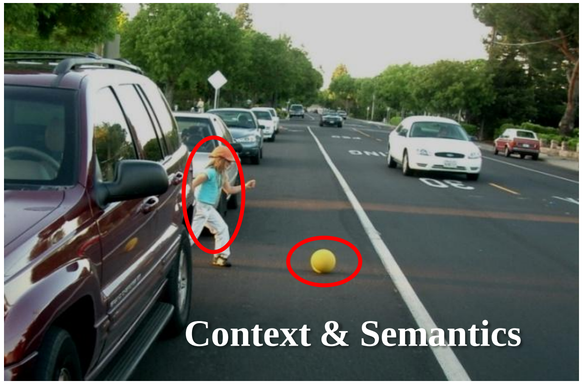
❑ Main Functions

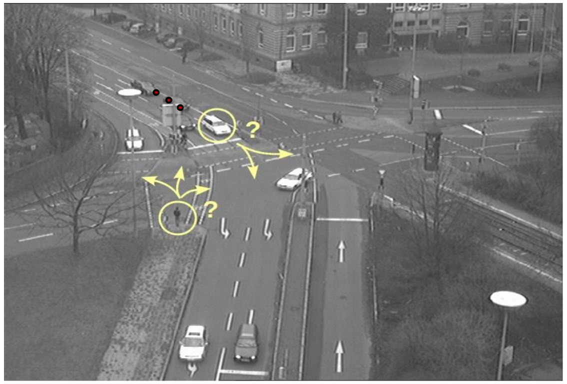
- ✓ Detection & Tracking of Mobile Objects (DATMO)
- ✓ Objects classification (recognition)
- ✓ Prediction & Risk Assessment (avoiding future collisions)

❑ Main Models & Algorithms

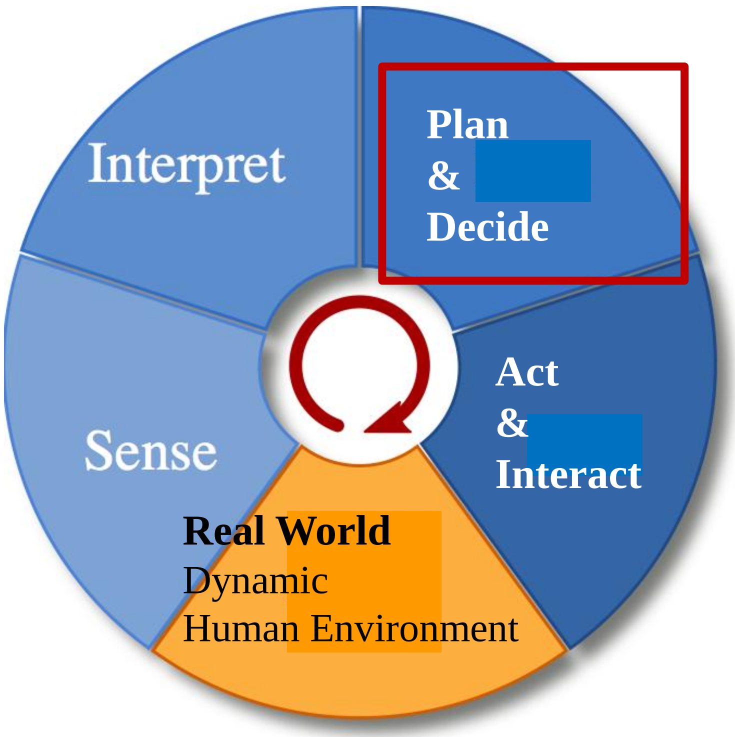
- ✓ Bayesian Perception Paradigm
- ✓ Behaviors modeling & learning
- ✓ Bayesian approaches for Prediction & Risk Assessment







Decisional & Control Architecture (4)



Objective

Planning robot motions & Deciding of the most appropriate action to be executed by the robot (Goal & Context & Risk)

Main Difficulty & Functions

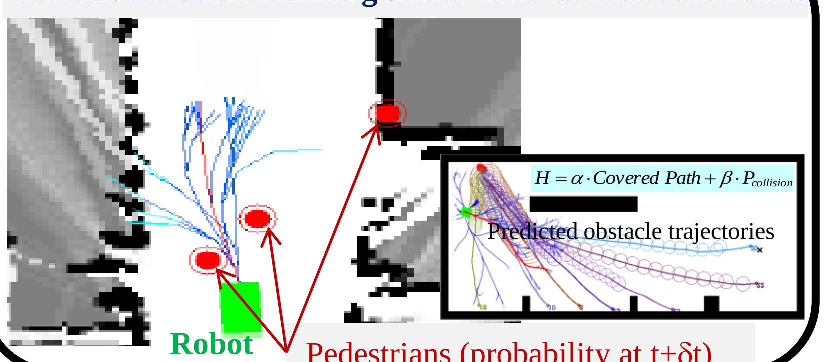
✓ On-line Motion Planning under various constraints (time, kinematic, dynamic, uncertainty, collision risk, social)

✓ Decision making under uncertainty using contextual data (history, semantics, prediction)

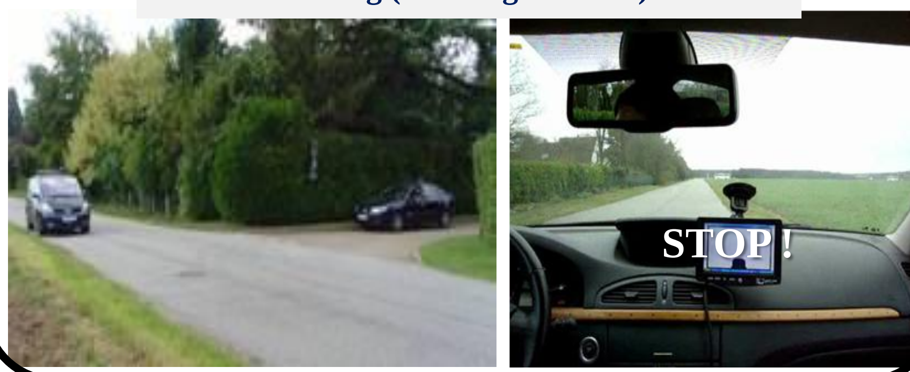
Main Models & Algorithms

✓ Iterative Risk-based Motion Planning (e.g. Risk-RRT)
✓ Decision making using Contextual data & Bayesian networks

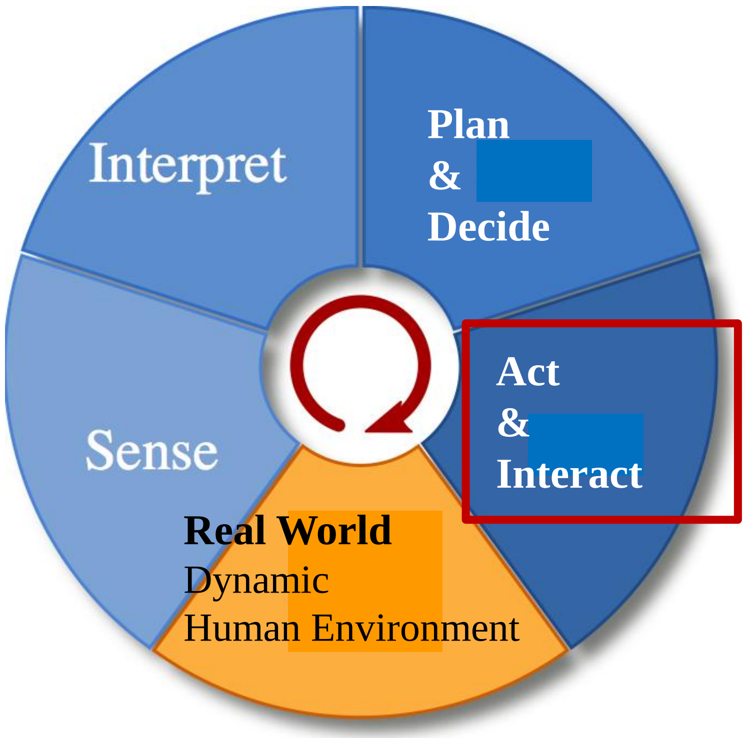
Iterative Motion Planning under Time & Risk constraints



Decision making (avoiding collision)



Decisional & Control Architecture (5)



- ❑ **Objective**

  Controlling the robot for executing **Safe & Socially Acceptable robot actions**, while taking into account the related **Human – Robot Interactions**

- ❑ **Main Difficulty & Functions**
  - ✓ Robot navigation while taking into account both Safety & Social constraints
  - ✓ **Human in the loop**

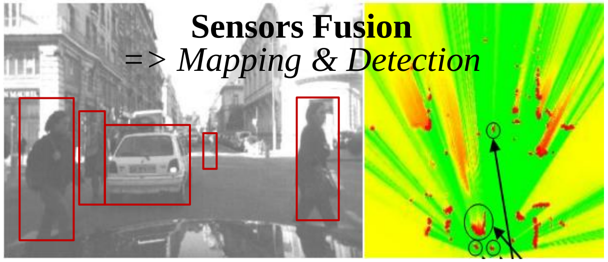
- ❑ **Main Models & Algorithms**
  - ✓ Human-Aware Navigation paradigm (safety & social filters)
  - ✓ Intuitive Human-Robot Interaction

Content of the Lecture

- ❑ Socio-economic & Technological Context
- ❑ Decisional & Control Architecture : Outline
- ❑ **Bayesian Perception *(Key Technology 1)***
- ❑ Embedded Perception & Experimental results
- ❑ Bayesian Risk Assessment & Decision-making *(Key Technology 2)*
- ❑ Conclusion & Perspectives

Key Technology 1: Embedded Bayesian Perception

Embedded Multi-Sensors Perception
⇒ Continuous monitoring of the dynamic environment



❑ Main challenges

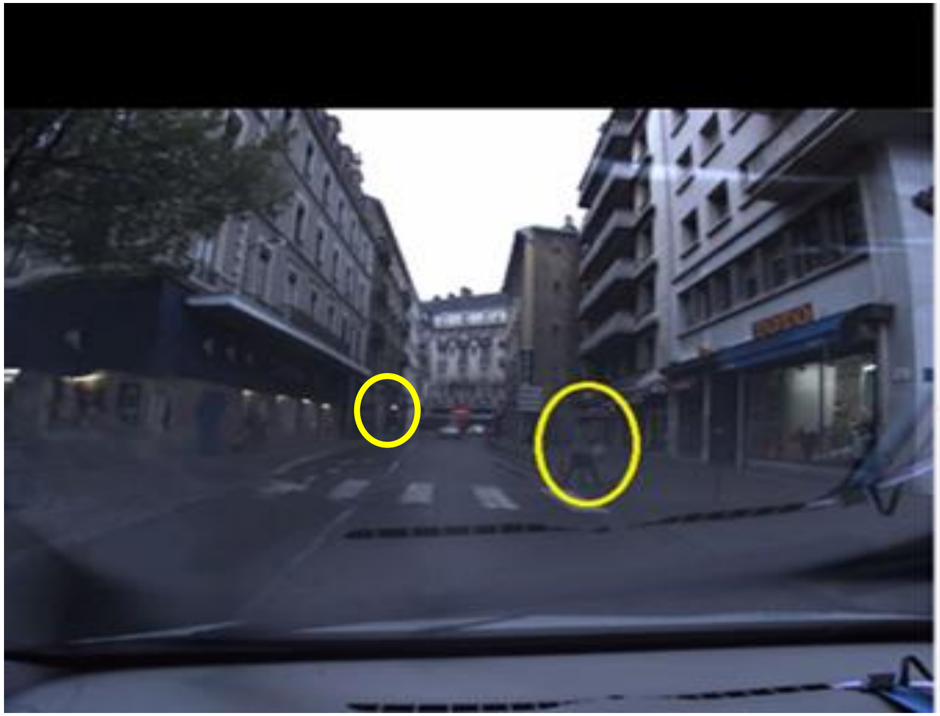
- ✓ *Noisy data, Incompleteness, Dynamicity, Discrete measurements*
- ✓ ***Strong Embedded & Real time constraints***

❑ Approach: Embedded Bayesian Perception

- ✓ *Reasoning about* ***Uncertainty & Time window*** *(Past & Future events)*
- ✓ ***Improving robustness*** *using* ***Bayesian Sensors Fusion***
- ✓ ***Interpreting the dynamic scene*** *using* ***Contextual & Semantic*** *information*
- ✓ ***Software & Hardware integration*** *using GPU, Multicore, Microcontrollers…*

Improving robustness using Multi-modality sensing

Camera Image at Dusk *(Pedestrians not detected)*



Camera output depends on lighting conditions
Cheap & Rich information & Good for classification

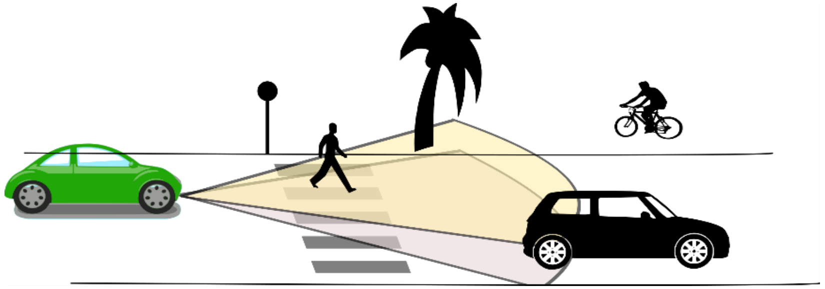
Processed Lidar data *(Pedestrians detected)*

Lidar more accurate & can work at night
*Good for fine detection of objects … but **still Expensive***

- *Develop **Robust & Efficient Multi-Sensor Fusion** approaches using probabilistic models*
- ***Good news:** A new generation of **affordable "Solid State Lidars"** will arrive soon on the market !*
  - *=> No mechanical component & Expected cost less than 1000 US$ (before mass production)*
  - *=> Numerous announcements since Spring 2016*

QUANERGY LeddarTech MASTERING LIDAR SENSOR TECHNOLOGY ZF ibeo automotive INNOVIZ TECHNOLOGIES

Bayesian Perception : Basic idea

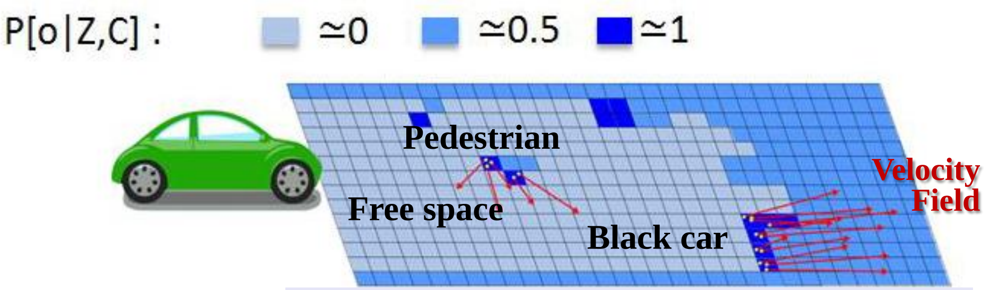


- ❑ Multi-Sensors Observations
  Lidar, Radar, Stereo camera, IMU …

Bayesian
Multi-Sensors Fusion

Real-time

- ❑ Probabilistic Environment Model
  - ✓ *Sensor Fusion*
  - ✓ *Occupancy grid integrating uncertainty*
  - ✓ *Probabilistic representation of Velocities*
  - ✓ *Prediction models*



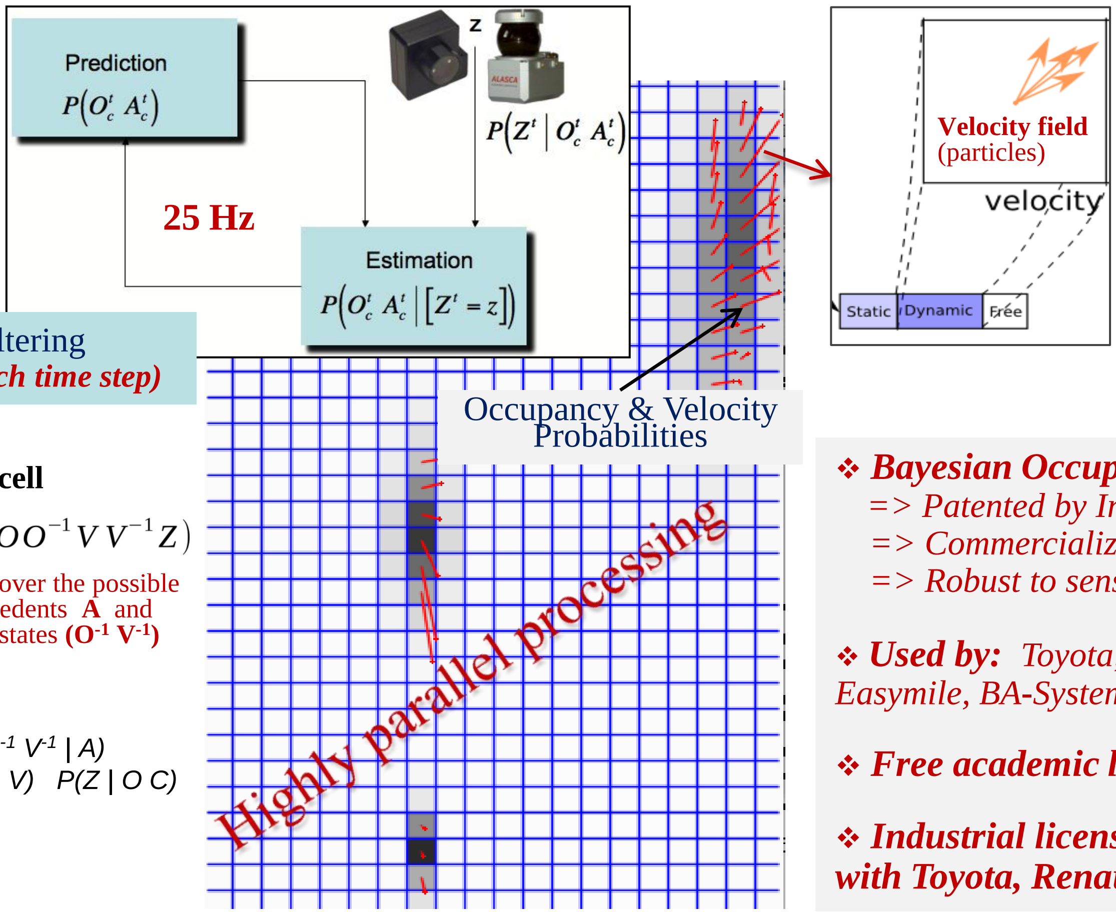
Concept of Dynamic Probabilistic Grid
⇒Occupancy & Velocity probabilities
⇒Embedded models for Motion Prediction

- ❑ Main philosophy
  Reasoning at the grid level as far as possible for both :
  - ***Improving efficiency*** *(highly parallel processing)*
  - ***Avoiding traditional object level processing problems*** *(e.g. detection errors, wrong data association…)*

A new framework: Dynamic Probabilistic Grids
=> *A clear distinction between Static & Dynamic & Free components*

[Coué & Laugier IJRR 05] [Laugier et al ITSM 2011] [Laugier, Vasquez, Martinelli Mooc uTOP 2015]

Sensing *(Observations)*



Bayesian Filtering
(Grid update at each time step)

Solving for each cell

$$P(O\,V \mid Z\,C) = \lambda \sum_{A\,O^{-1}\,V^{-1}} P(C\,A\,O\,O^{-1}\,V\,V^{-1}\,Z)$$

Sum over the possible antecedents **A** and their states **(O^{-1} V^{-1})**

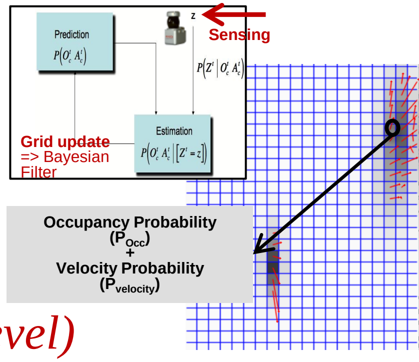
Joint Probability decomposition:

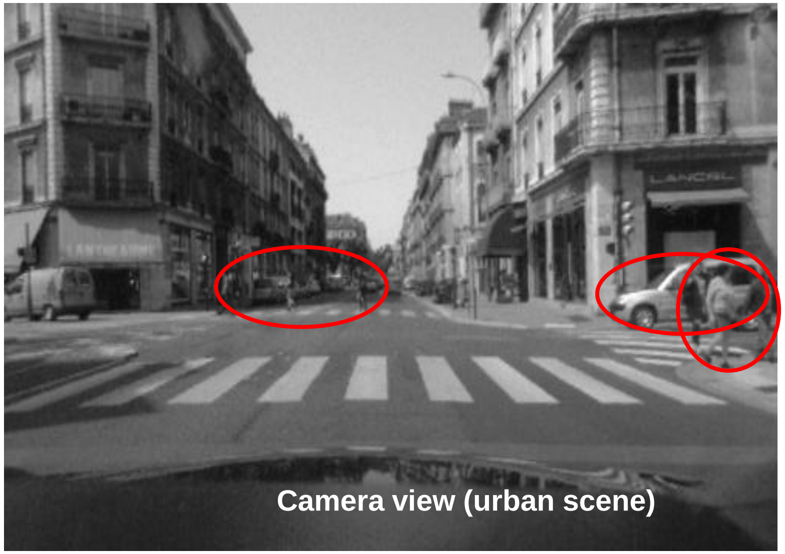
$$P(C\,A\,O\,O^{-1}\,V\,V^{-1}\,Z) = P(A)\ \ P(O^{-1}\,V^{-1} \mid A)\ \ P(O\,V \mid O^{-1}\,V^{-1})\ \ P(C \mid A\,V)\ \ P(Z \mid O\,C)$$

- ***Bayesian Occupancy Filter (BOF)***
  - *=> Patented by Inria & Probayes*
  - *=> Commercialized by Probayes*
  - *=> Robust to sensing errors & occultation*
- ***Used by:*** *Toyota, Denso, Probayes, Easymile, BA-Systems, IRT Nanoelec / CEA*
- ***Free academic license available***
- ***Industrial license under negotiation with Toyota, Renault, Easymile***

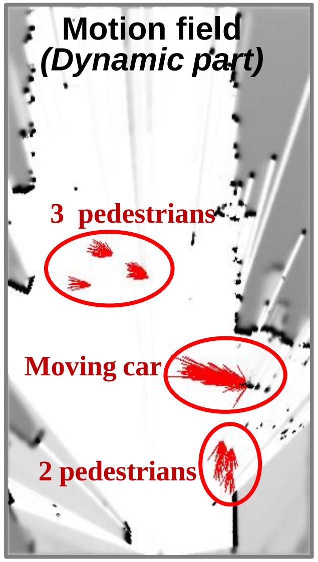
Bayesian Occupancy Filter (BOF) – *Main Features*

- Estimate **Spatial occupancy for each cell of the grid** ***P (O | Z)***
- **Grid update** is performed in each cell in parallel *(using BOF equations)*
- **Extract Motion Field** *(using Bayesian filtering & Fused Sensor data)*
- **Reason at the Grid level** *(i.e. no object segmentation at this reasoning level)*





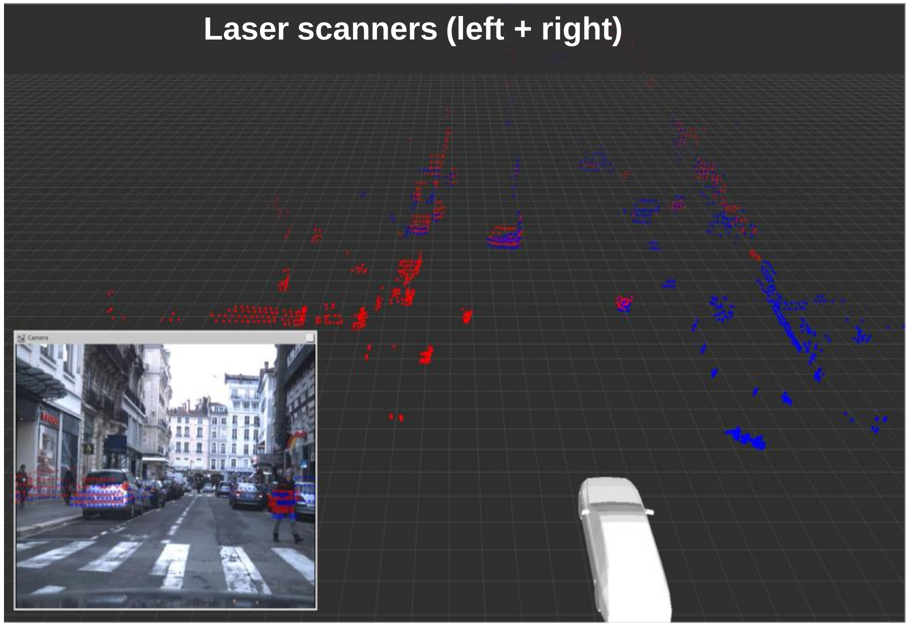
Sensors data fusion
+
Bayesian Filtering

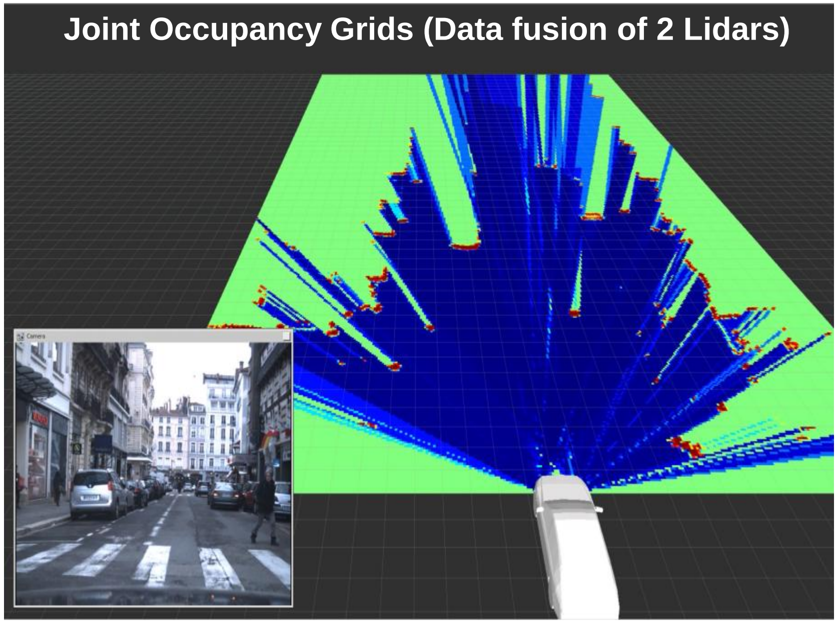


Exploiting the Dynamic information for improving Scene Understanding !!

Bayesian data fusion: *The joint Occupancy Grid*

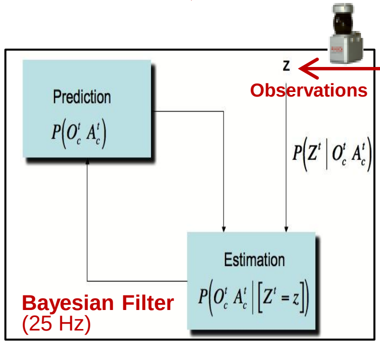
- **Observations Z_i** are given by each sensor ***i*** *(Lidars, cameras, etc)*
- For each set of observation Z_i , **Occupancy Grids are computed:** ***$P(O \mid Z_i)$***
- Individual grids are **merged into a single one:** ***$P(O \mid Z)$***

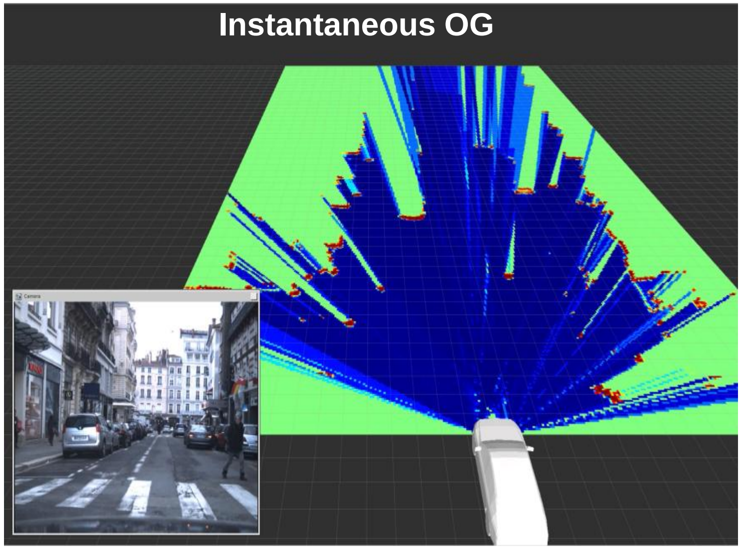


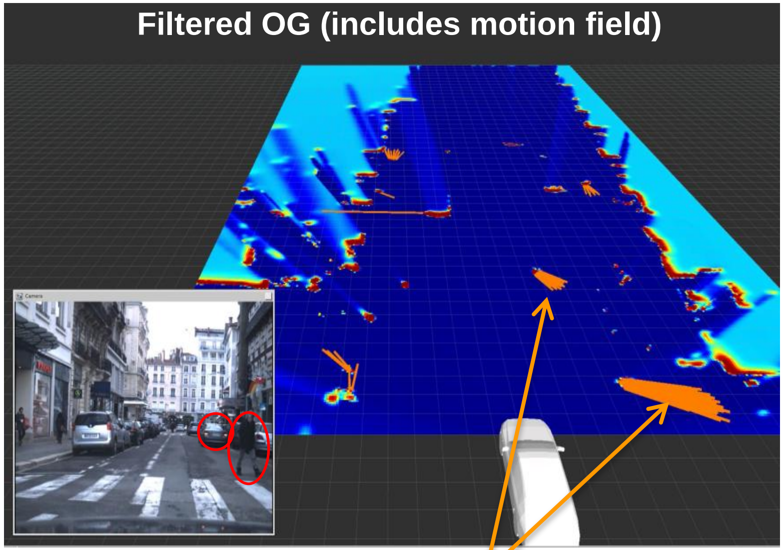


Taking into account dynamicity:
Filtered Occupancy Grid (Bayesian filtering)

- **Filtering** is achieved through the ***prediction/correction loop*** **(Bayesian Filter)**
  *=> It allows to take into account **grid changes over time** !*
- **Observations** are used to update the environment model
- **Update is performed in each cell <u>in parallel</u>** *(using BOF equations)*
- **Motion field** is constructed from the resulting filtered data





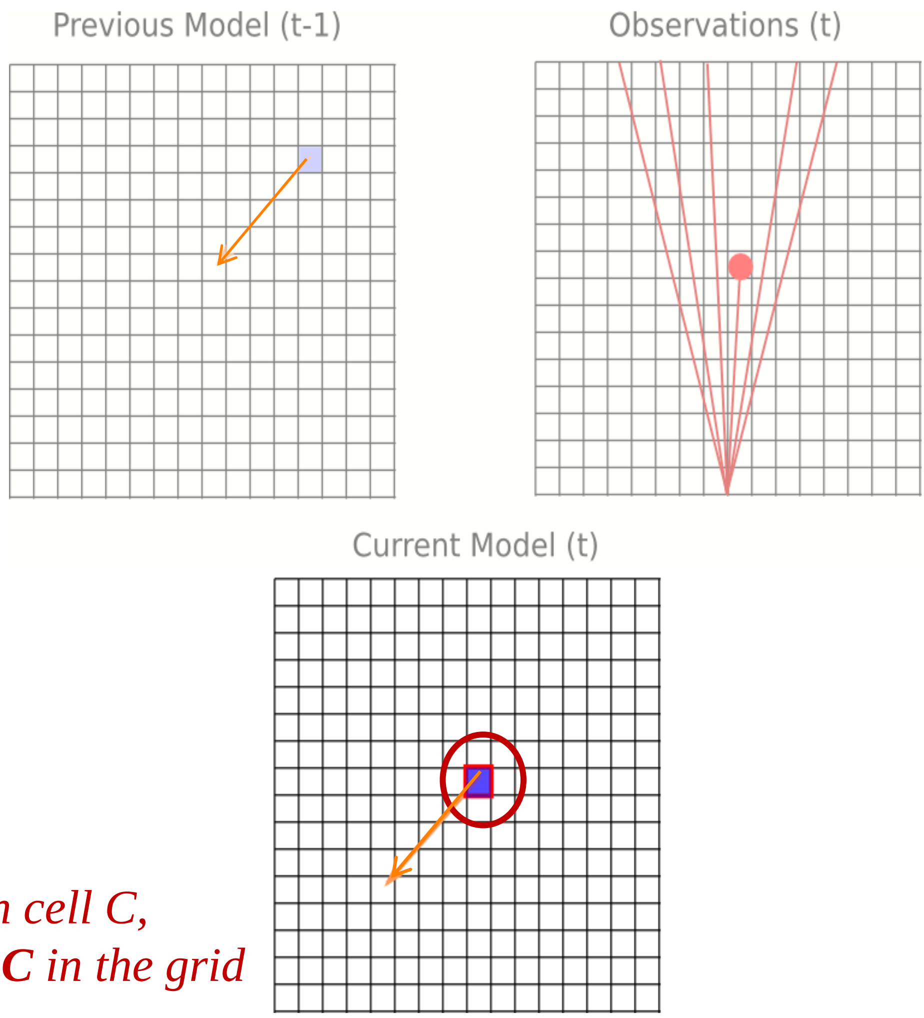


Motion fields are displayed in orange color

Bayesian Occupancy Filter – *Formalism*

Variables:

- C : current cell
- A : antecedent cell, i.e. the cell from which the occupancy of the current cell comes from
- O : occupancy of the current cell C
- O^{-1} : previous occupancy in the antecedent cell
- V : current velocity
- V^{-1} : previous velocity in the antecedent
- Z : observations (sensor data)



Objective:

Evaluate P (O V | Z C)

*=> Probability of **Occupancy & Velocity** for each cell C,*
*Knowing the **observations Z** & the **cell location C** in the grid*

Bayesian Occupancy Filter

How to theoretically compute P (O V | Z C) ?

$$P(O\,V\,|\,Z\,C) = \lambda \sum_{A\,O^{-1}\,V^{-1}} P(C\,A\,O\,O^{-1}\,V\,V^{-1}\,Z)$$

Sum over the possible antecedents **A** and their states **(O^{-1} V^{-1})** at time t-1

The joint probability term can be re-written as follows:

$$P(C\,A\,O\,O^{-1}\,V\,V^{-1}\,Z) = P(A)\;\; P(O^{-1}\,V^{-1}\,|\,A)\;\; P(O\,V\,|\,O^{-1}\,V^{-1})\;\; P(C\,|\,A\,V)\;\; P(Z\,|\,O\,C)$$

Joint probability => *used for the update of P (O V | Z C)*

P(A) *:* Selected as **uniform** *(every cell can a priori be an antecedent)*

$P(O^{-1}\,V^{-1}\,|\,A)$ *:* Result from the previous iteration

$P(O\,V\,|\,O^{-1}\,V^{-1})$ *:* **Dynamic model**

P(C | A V) *:* **Indicator function** of the cell **C** corresponding to the **"projection"** in the grid of the antecedent **A** at a given velocity **V**

P(Z | O C) *:* **Sensor model**

Bayesian Occupancy Filter

How to theoretically compute P (O V | Z C) ?

$$P(O\,V\,|\,Z\,C) = \lambda \sum_{A\,O^{-1}\,V^{-1}} P(C\,A\,O\,O^{-1}\,V\,V^{-1}\,Z)$$

Sum over the possible antecedents **A** and their states **(O^{-1} V^{-1})** at time t-1

The joint probability term can be re-written as follows:

$$P(C\,A\,O\,O^{-1}\,V\,V^{-1}\,Z) = P(A)\;\; P(O^{-1}\,V^{-1}\,|\,A)\;\; P(O\,V\,|\,O^{-1}\,V^{-1})\;\; P(C\,|\,A\,V)\;\; P(Z\,|\,O\,C)$$

But, computing this expression is difficult in practice

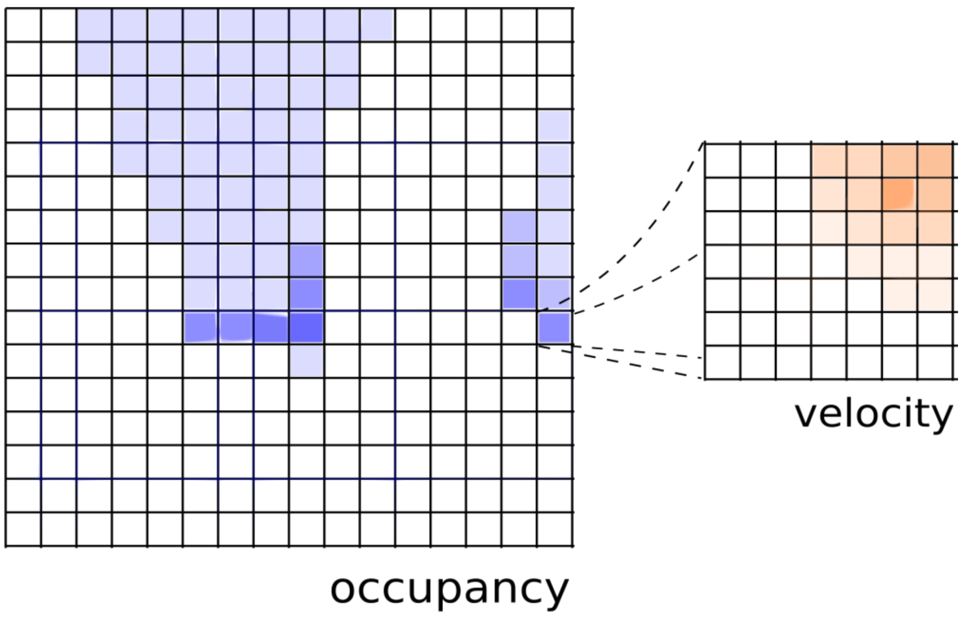
=> Huge range of possible antecedents

=> Strongly depends on Grid size & Velocity range

How to compute P(OV | Z C) in practice?

Initial approach: The classic BOF process

- **Regular grid**
- **Transition histograms** for every cell (for representing velocities)

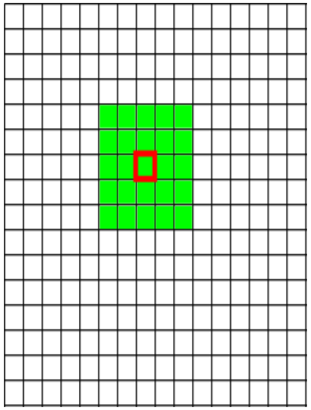


How to compute P(OV | Z C) in practice?

Initial approach: The classic BOF process

$$P(O\,V\,|\,Z\,C)=\lambda\sum_{A\,O^{-1}\,V^{-1}}P(C\,A\,O\,O^{-1}\,V\,V^{-1}\,Z)$$

Sum over the possible antecedents A
and their states (O^{-1} V^{-1})



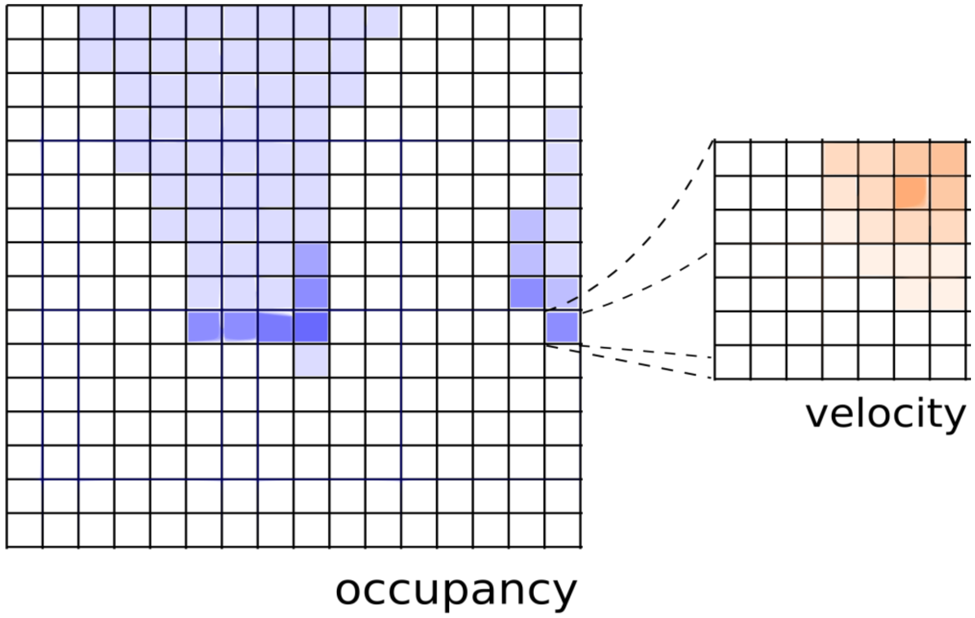
Practical computation:

→ Sum over the **neighborhood,** with a **single possible velocity per antecedent A** of equation:

$$P(C\,A\,O\,O^{-1}\,V\,V^{-1}\,Z) = P(A)\;\;P(O^{-1}\,V^{-1}\,|\,A)\;\;P(O\,V\,|\,O^{-1}\,V^{-1})\;\;P(C\,|\,A\,V)\;\;P(Z\,|\,O\,C)$$

How to compute P(OV | Z C) in practice?
Initial approach: Drawbacks (1)

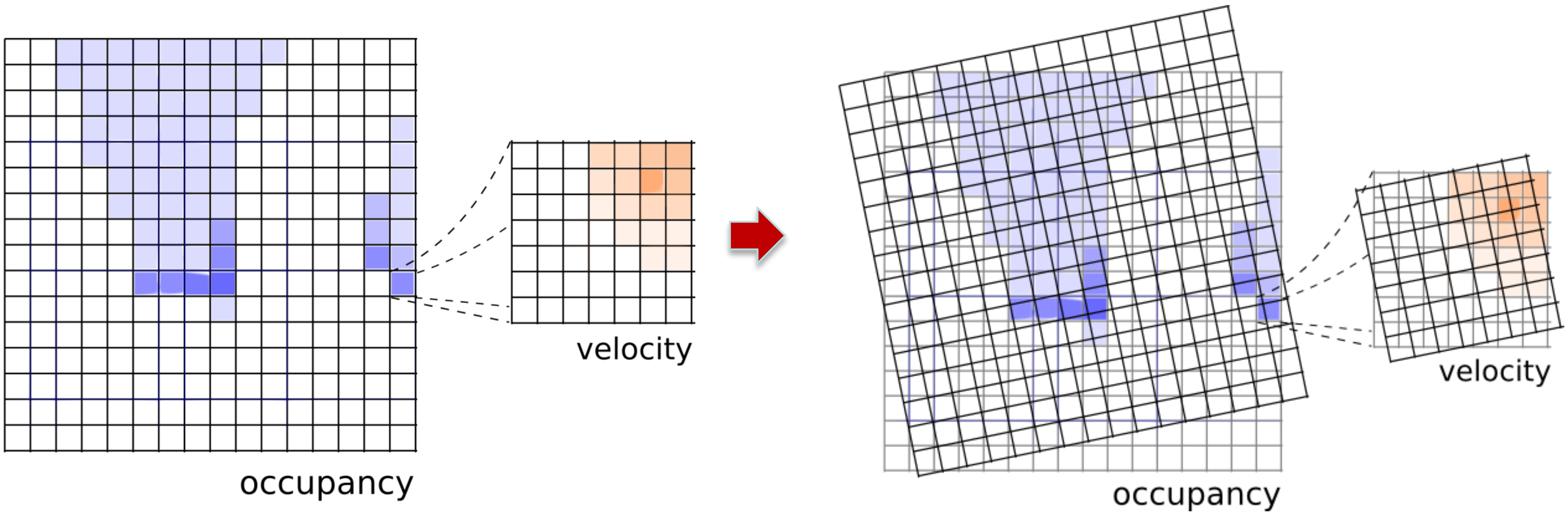
- **Velocity histogram** needs to be accurate on both **low** & **high velocities**
  - ⇒ *Its resolution has to be high, while being **mostly empty !***
  - ⇒ *It requires a **large memory size** (but in practice the **accuracy is still weak**) !*



⇒ Large memory size required
⇒ Weak accuracy

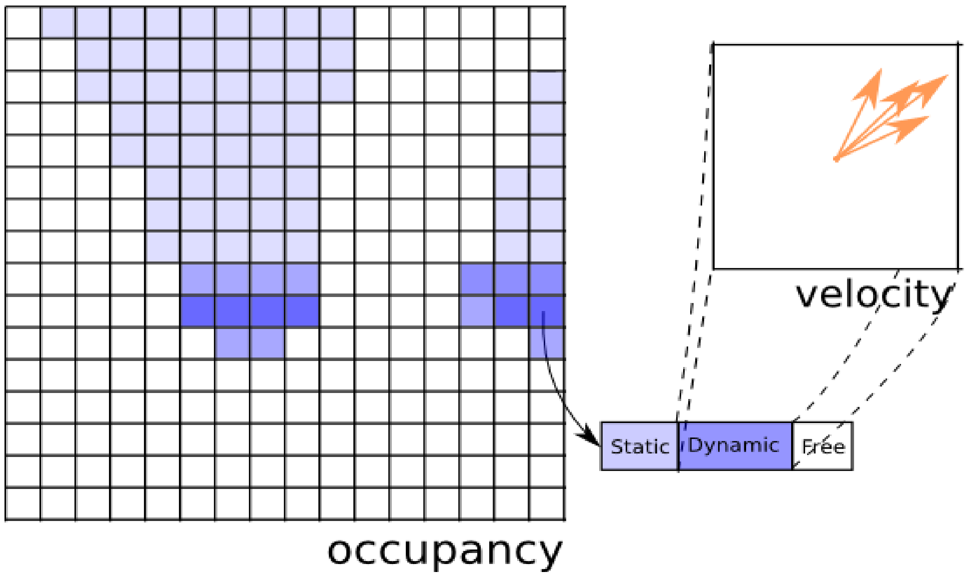
How to compute P(OV | Z C) in practice?
Initial approach: Drawbacks (2)

- **Temporal aliasing** *=> Due to* ***update & real frequencies synchronization***
- **Spatial aliasing** (moving grid) *=>* ***High complexity*** *due to 4-dimension interpolation*
  => Approximations required in practice



How to compute P(OV | Z C) in practice?
HSBOF updating process (principle)

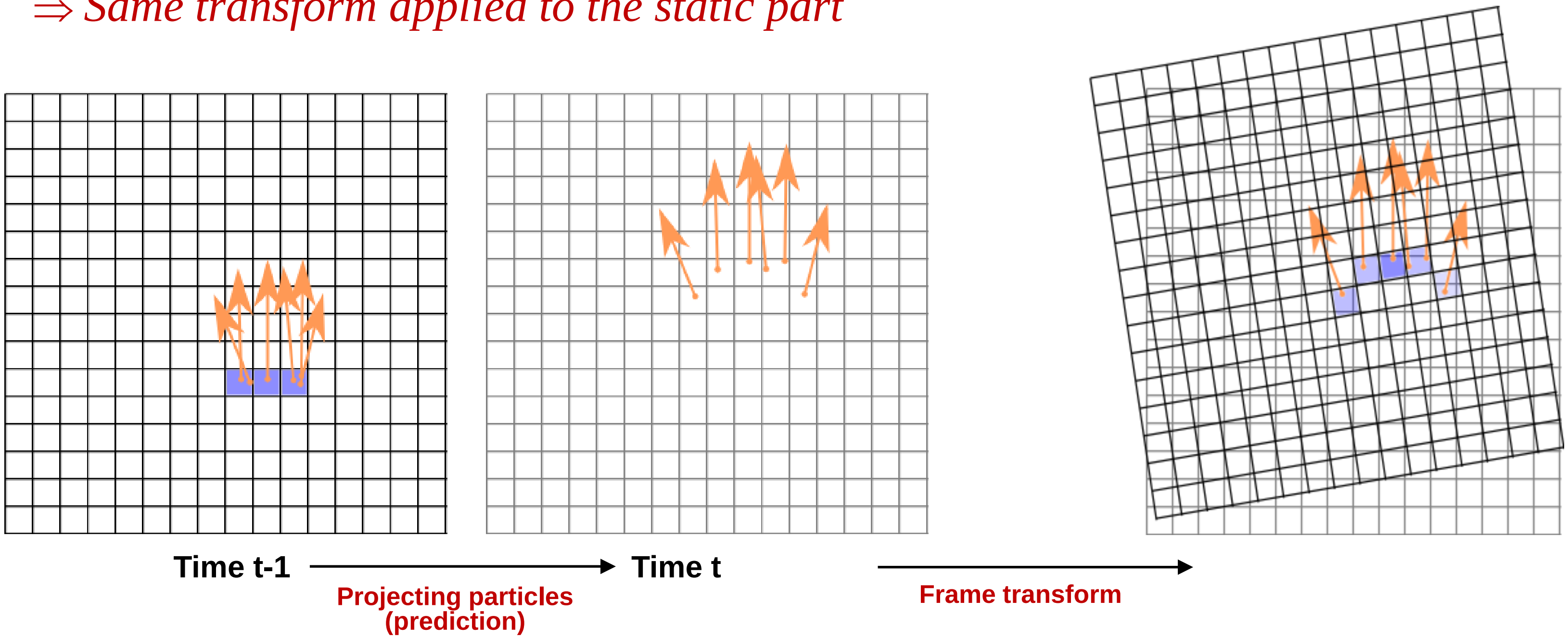
- **Basic idea**: *Modify the representation structure to avoid the previous computational problems*
  - ✓ Making a clear distinction between **Static & Dynamic & Free** components
  - ✓ Modeling velocity using **Particles** *(instead of histogram)*
  - ✓ Making an **adaptive repartition** of those particles in the grid



How to compute P(OV | Z C) in practice?
HSBOF updating process (principle)

- **Introducing a Dynamic model for "projecting" particles in the grid** *($S_{t-1} \rightarrow S_t$)*
  - ⇒ *Immediate antecedent association*
  - ⇒ *Simplified velocity prediction to the cells*

- **Updating Grid Reference Frame**
  - ⇒ *Translation & Rotation values* ***provided by sensors*** *(Odometry + IMU)*
  - ⇒ *Same transform applied to the static part*



How to compute P(OV | Z C) in practice?

HSBOF updating process (outline of the algorithm)

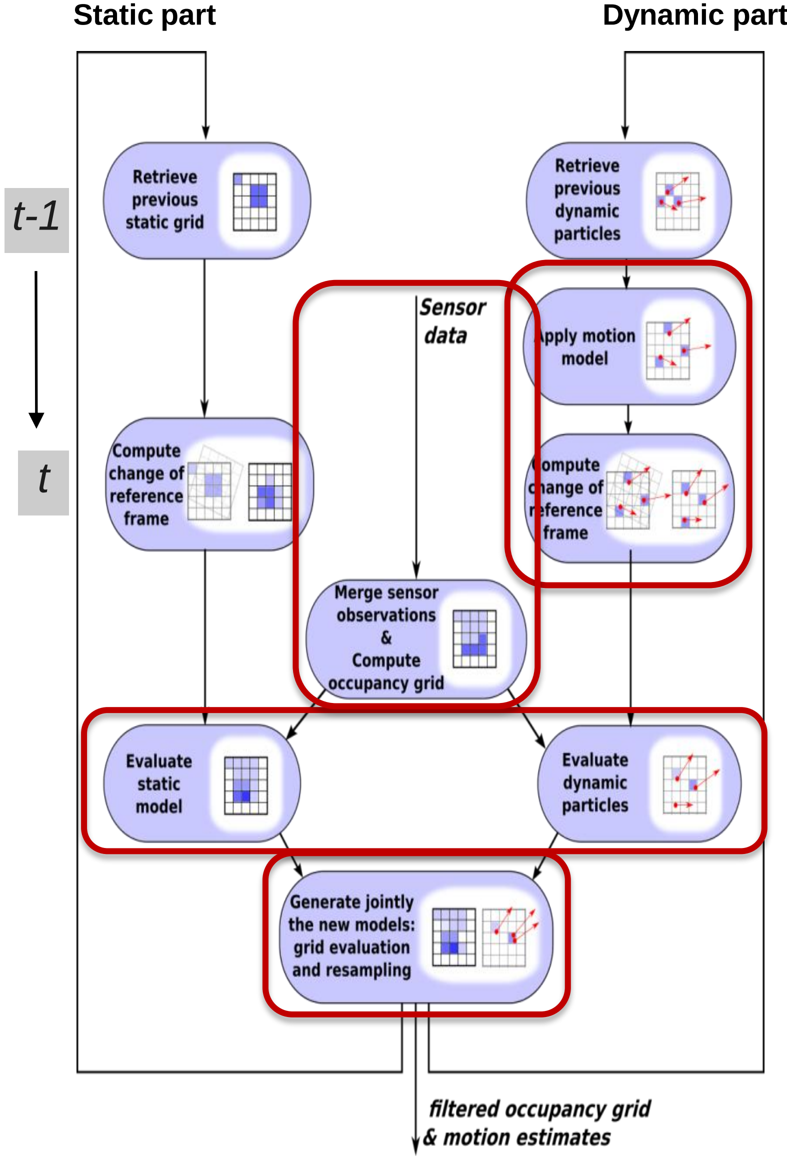
Main steps in the updating process

• Dynamic part (particles) is "**projected**" in the grid using motion model => *motion prediction*

• Both Dynamic & Static parts are expressed in the **new reference frame** => *moving vehicle frame*

• The two resulting representations are confronted to the **observations** => *estimation step*

• **New representations (static & dynamic)** are jointly evaluated and particles re-sampled



How to compute P(OV | Z C) in practice ?

HSBOF filtering calculation

$$P(O\,V\,|\,Z\,C) = \lambda \sum_{A\,O^{-1}V^{-1}} P(C\,A\,O\,O^{-1}\,V\,V^{-1}\,Z)$$

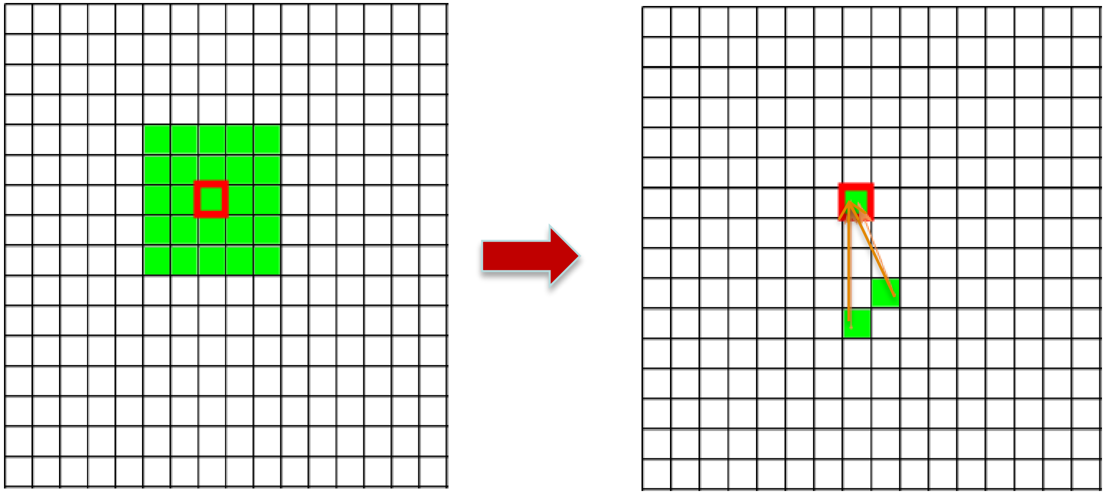
~~Sum over the neighborhood, with a single velocity per antecedent~~

A more efficient computation approach :

*=> Sum over **the particles projected in the cell** & their **related static parts***

$$P(C\,A\,O\,O^{-1}\,V\,V^{-1}\,Z) = P(A)\;\; P(O^{-1}\,V^{-1}\,|\,A)\;\; P(O\,V\,|\,O^{-1}\,V^{-1})\;\; P(C\,|\,A\,V)\;\; P(Z\,|\,O\,C)$$

Previous computation approach (histograms)



New computation approach (particles)

HSBOF: *Main Features (summary)*

- Empty & Static components => ***Occupancy Grid***

- Dynamic components => ***Sets of Particles*** *(Motion field)*
  - ✓ Smooth integration of the ego-motion (IMU & Odometry)
  - ✓ Propagation of sets of particles in the Grid (using dynamic models)
  - ✓ Joint estimation of distributions

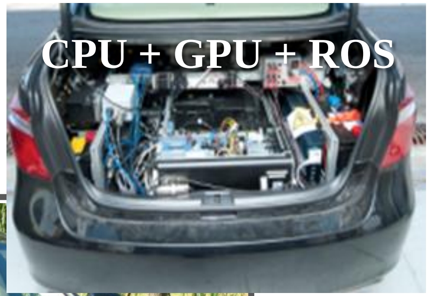
- More efficient (computation & memory) & Better estimation of velocities (more accurate)

Content of the Lecture

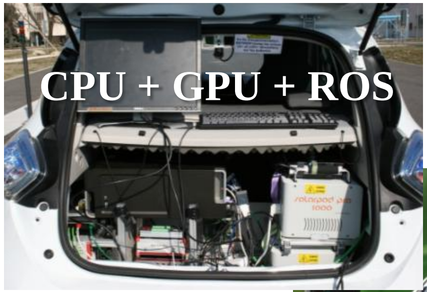
- ❑ Socio-economic & Technological Context
- ❑ Decisional & Control Architecture: Outline
- ❑ Bayesian Perception *(Key Technology 1)*
- ❑ **Embedded Perception & Experimental results**
- ❑ Bayesian Risk Assessment & Decision-making *(Key Technology 2)*
- ❑ Conclusion & Perspectives

Experimental Vehicles & Embedded Perception Units

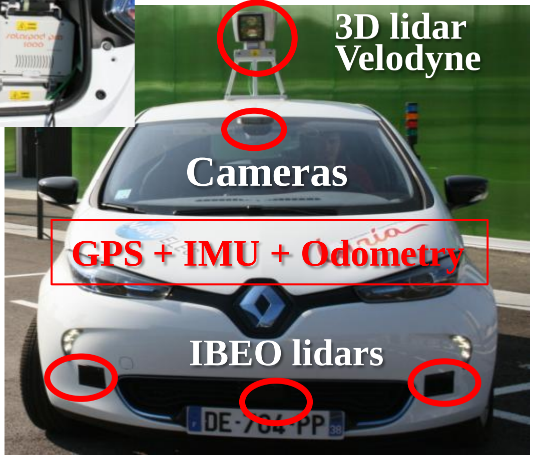
Toyota Lexus 2010



ROS



Renault Zoé 2014

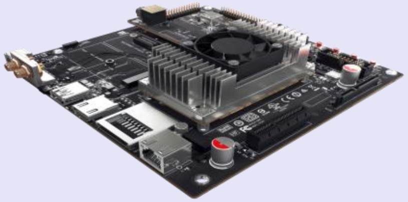


Connected Perception Unit

Nvidia GTX Titan X
Generation Maxwell

Nvidia GTX Jetson TK1
Generation Maxwell

Nvidia GTX Jetson TX1
Generation Maxwell



Embedded Perception Units & Hardware

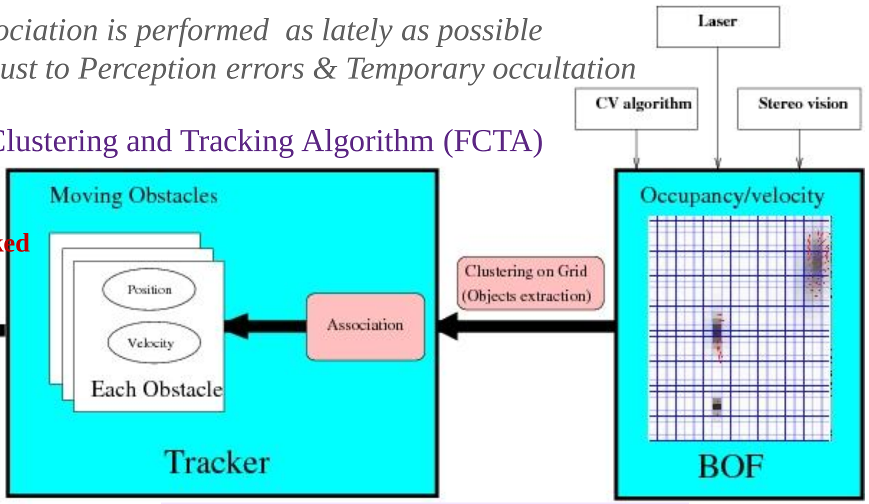
Former Bayesian Perception Architecture (outline)

Bayesian Sensor Fusion + Detection & Tracking

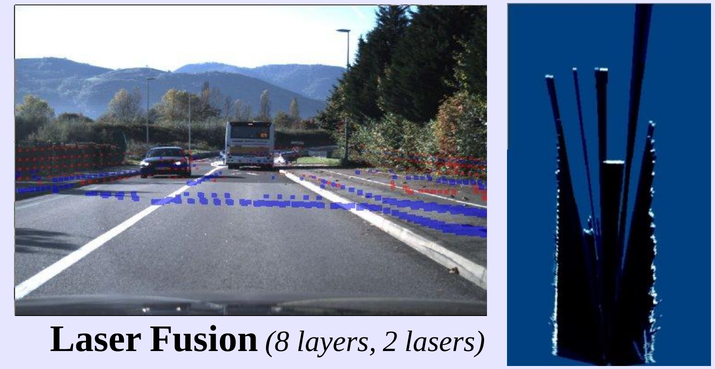
- *Data association is performed as lately as possible*
- *More robust to Perception errors & Temporary occultation*

Fast Clustering and Tracking Algorithm (FCTA)

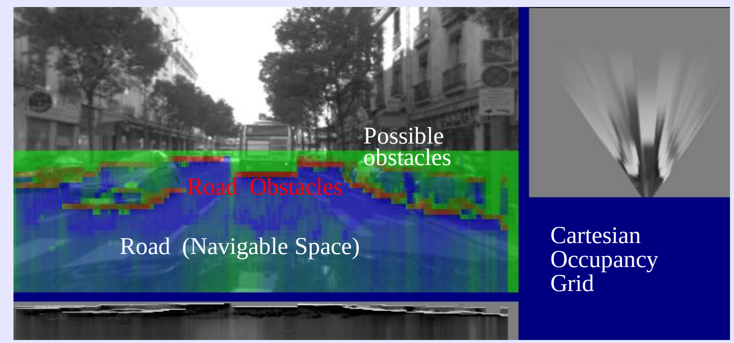
Detected & Tracked Objects



[Mekhnacha 09, Laugier et al ITSM'11]

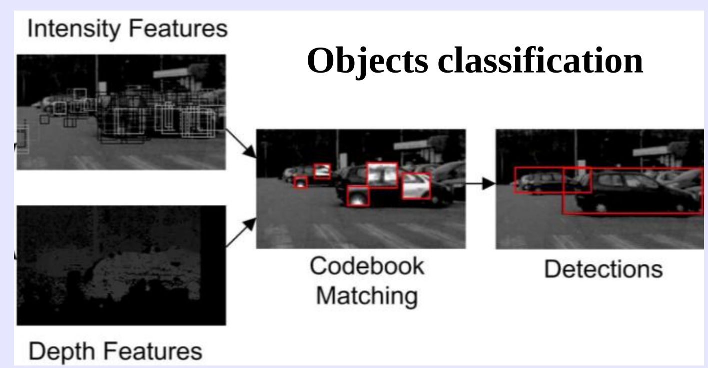


Laser Fusion *(8 layers, 2 lasers)*



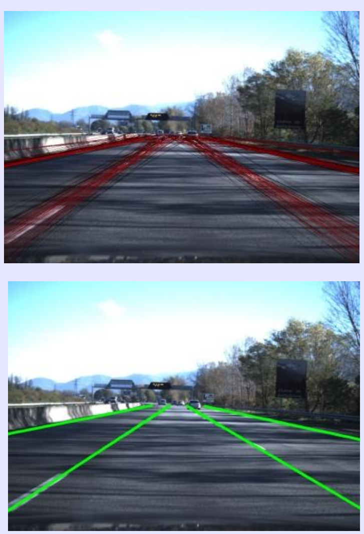
Stereo-vision *(U-disparity OG+ Road/obstacle classif.)*

Objects classification



Motion Detection

=> Dynamic grid filtering using Motion data (IMU + Odometry)

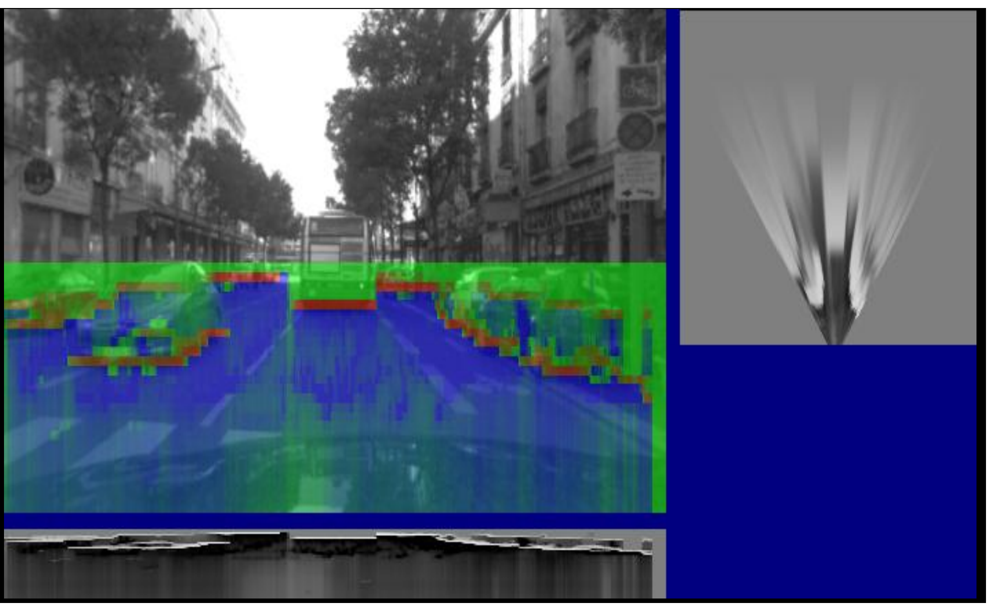


Multi-Lane tracker

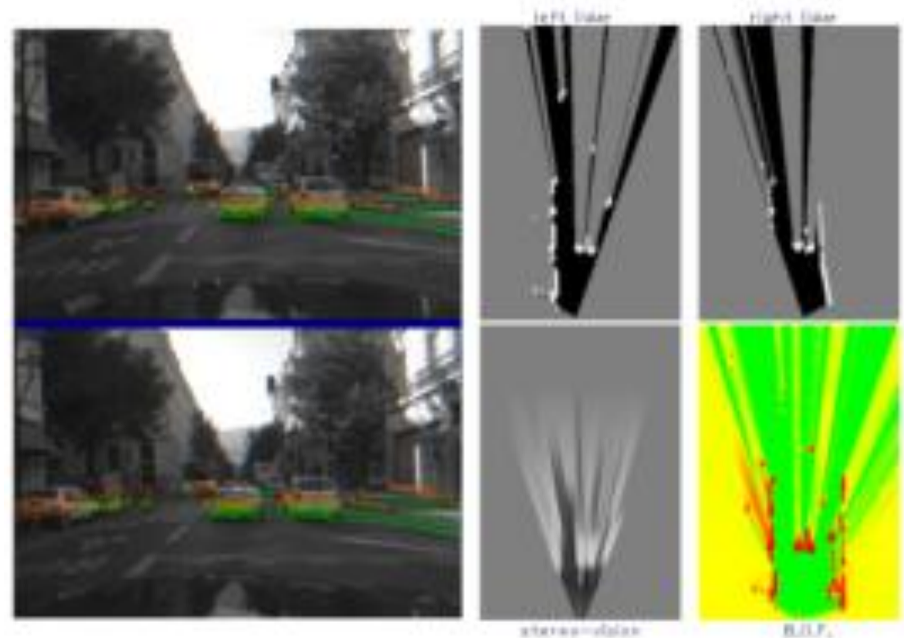
Former experimental results *(Inria – Toyota Lexus)*

Multiple sensors Fusion (Stereo vision & Lidars)

[Perrollaz et al 10] [Laugier et al ITSM 11]
IROS Harashima Award 2012



Stereo Vision
(U-disparity OG + Road / Obstacles classification)



Bayesian Sensor Fusion *(Stereo Vision + Lidars)*

Embedded Perception

Objectives & Achievements 2013 -17

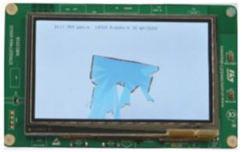
Fusion on many-core architecture **microcontroller**

Embedded Hardware (STHORM)

Experimental Platform

Automotive Standard
Multicore.
Dual cortex A9@ 800Mhz

Microcontroller
STM 32 Cortex
M7@200 MHz



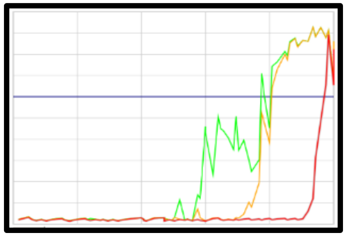
ICRA 2016 & CES 2017

Nvidia Jetson Tk1

Nvidia Jetson TX1

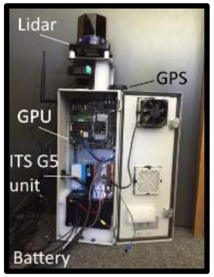
GTC Europe 2016 & 2017

BOF	HSBOF	CMCDOT	CMCDOT *Cuda Optimization on Tegra*
2013	2014	2015	2016 - 17

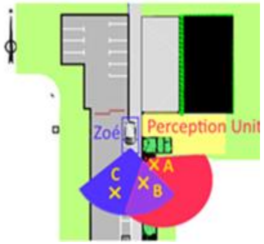


Risk assessment system

Experimental scenario (crash-test equipment)



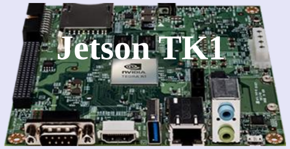
Connected Perception Unit



Distributed Perception (V2X)

Zoe Automatization

DP-Grids: *Recent implementations & Improvements*

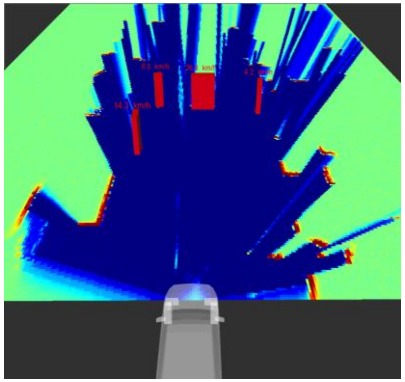


*Several implementations (models & algorithms) more and more adapted to **Embedded constraints & Scene complexity***

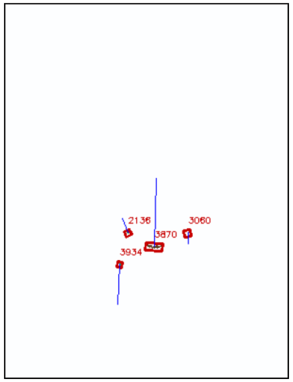
- Hybrid Sampling Bayesian Occupancy Filter (HSBOF, 2014) *[Negre et al 14] [Rummelhard et al 14]*
  *=> Drastic **memory size reduction** (factor 100) + Increased **efficiency** (complex scenes)*
  *+ More **accurate Velocity estimation** (using Particles & Motion data from ego-vehicle)*

- Conditional Monte-Carlo Dense Occupancy Tracker (CMCDOT, 2015) *[Rummelhard et al 15]*
  ***=> Increased efficiency using** "state data" (Static, Dynamic, Empty, **Unknown**) + Integration of a **"Dense Occupancy Tracker"** (Object level, Using particles propagation & ID)*

- CMCDOT + Ground Estimator (under Patenting, 2017) *[Rummelhard et al 17]*
  => Ground shape estimation & Improve obstacle detection (avoid false detections on the ground)



Grid & Pseudo-objects



Tracked Objects

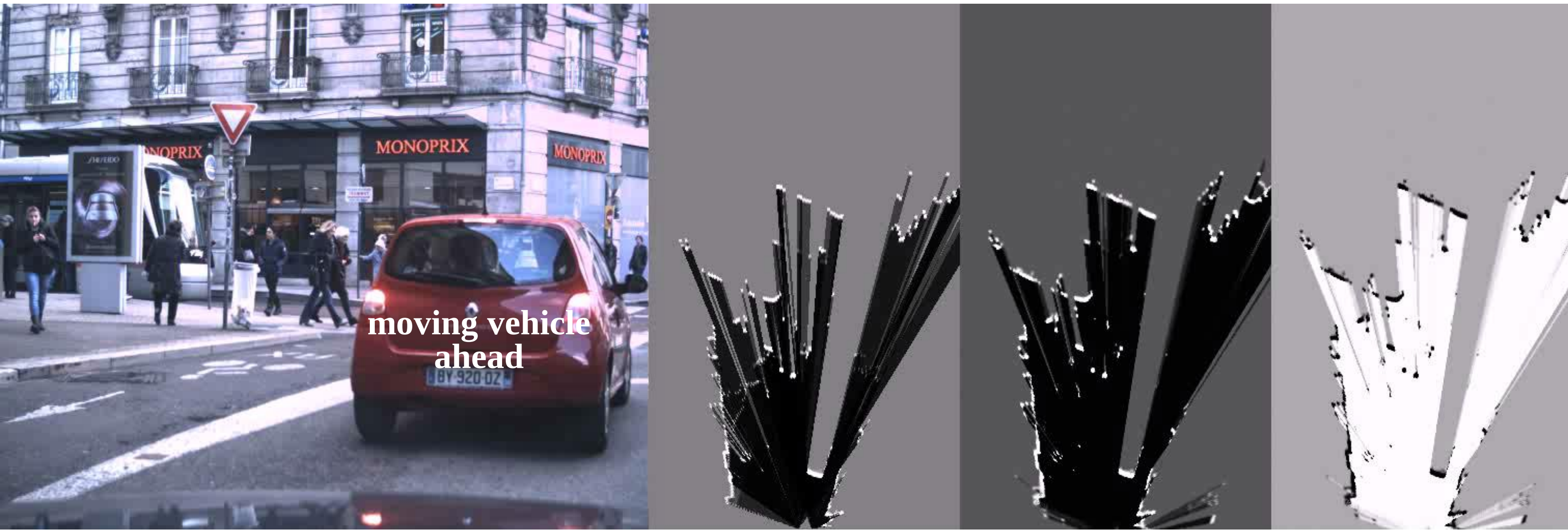
Classification *(using Deep Learning)*

Detection & Tracking & Classification

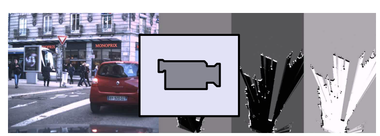
HSBOF

Experimental Results in dense Urban Environments

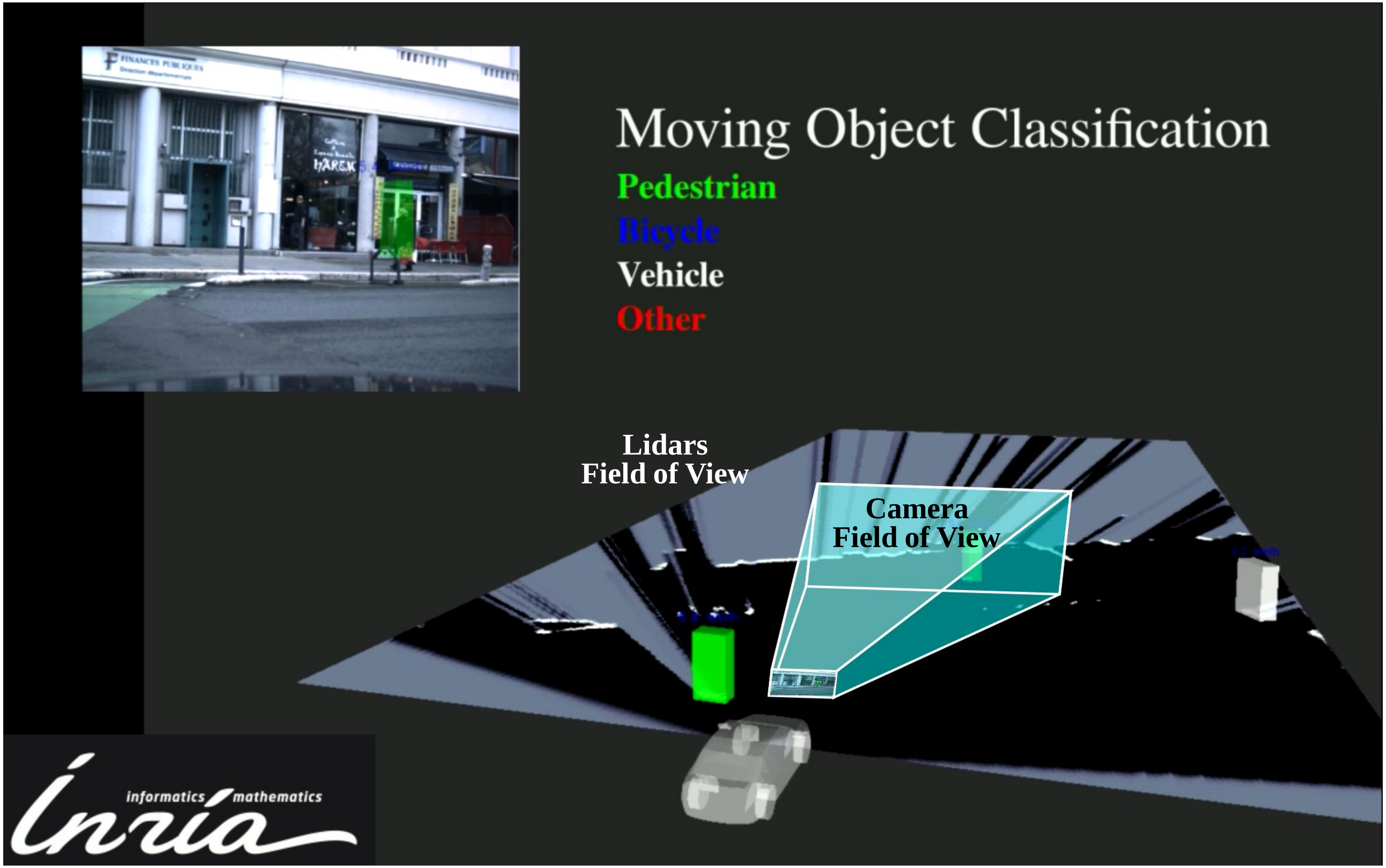
Observed Urban Traffic scene



Ego Vehicle *(not visible on the video)* | OG Left Lidar | OG Right Lidar | OG Fusion + Velocity Fields

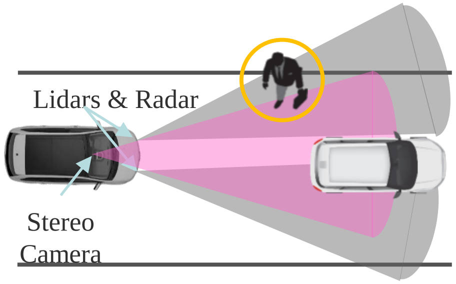


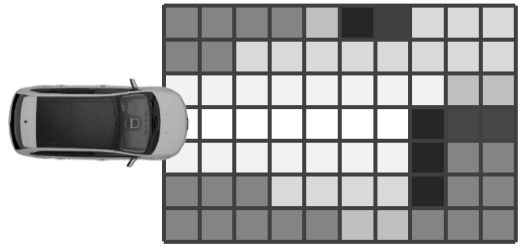
CMCDOT
Detection & Tracking & Classification results



Software / Hardware Integration – *Motivation*

PhD Tiana Rakotovao





OGs in practice:
- High number of cells
- Several sensors

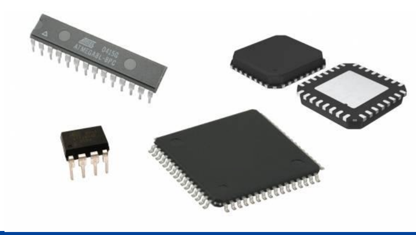
~ Billions Floating-point operations per sec

Hardware accelerators

High-end GPUs & CPUs

Not adapted for automotive industry

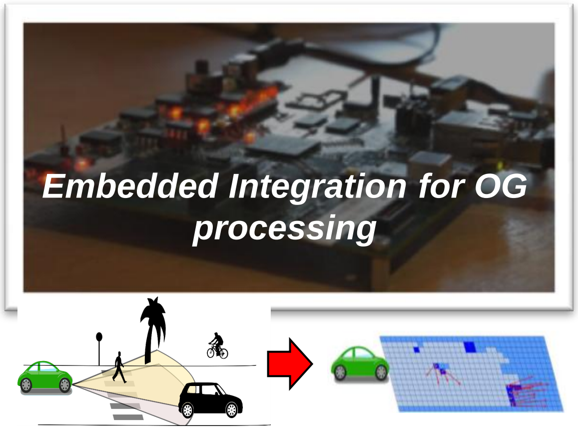
Electronic Control Units
Microcontroller, FPGA …



How to integrate
computing requirements of OGs
into embedded ECUs ?

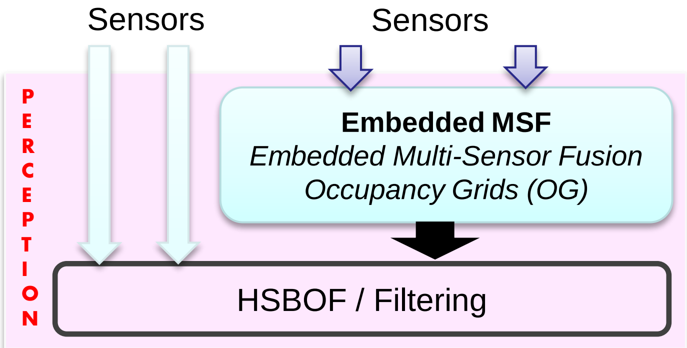
PhD Tiana Rakotovao

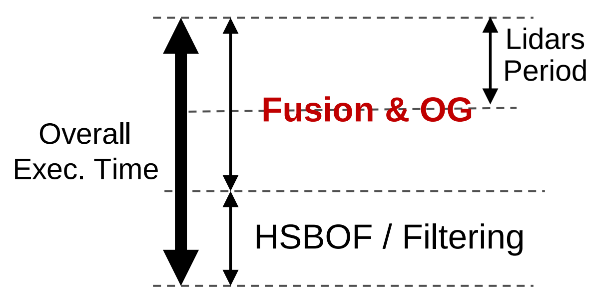
❑ The challenge: *How to cope with contradictory requirements & constraints ?*

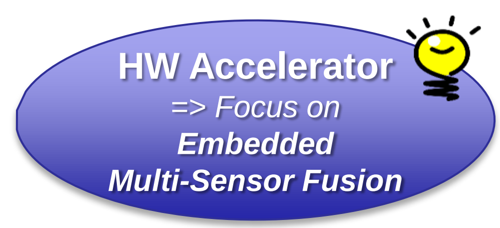


- **Embedded characteristics:** ***Low*** *computing power, Low memory space, Low bandwidth*
- **Embedded constraints:** ***Low*** *purchase cost, Low energy consumption, Small physical size*
- **Algorithmic constraints:** ***High*** *computing requirement,* ***High*** *memory space & bandwidth requirement*

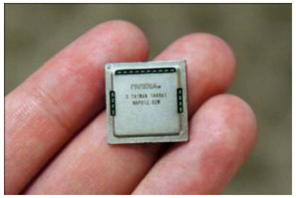
❑ Time Performance Analysis







Software / Hardware Integration – *GPU*



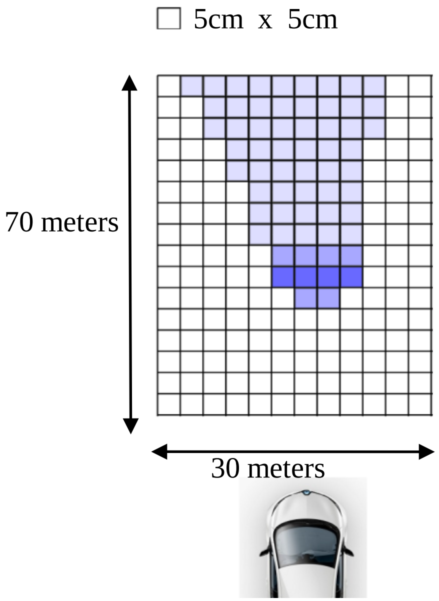
- Highly parallelizable framework, **27 kernels** over cells and particles
  - *=> Occupancy, speed estimation, re-sampling, sorting, prediction*
- Real-time implementation (20 Hz), optimized using Nvidia profiling tools

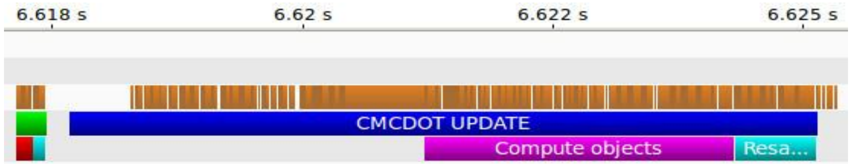
Results:

- Configuration with 8 Lidar layers (2x4)
- Grid: 1400 x 600 (840 000 cells) **+** Velocity samples: 65 536

=> Jetson TK1: *Grid Fusion **17ms**, CMCDOT **70ms***

=> Jetson TX1: *Grid Fusion **0.7ms**, CMCDOT **17ms***





Software / Hardware Integration – *µController*

PhD Tiana Rakotovao

Implementing MSF (OG without filtering) on µController *[1][2]*

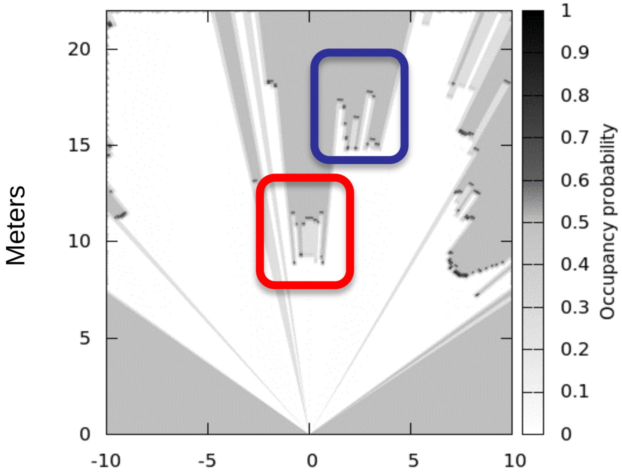
⇒ *Low Cost / Energy / Size (widely used in Industrial product)*

⇒ *Implementation based on **Integer Arithmetic** (Quantized Occupancy Probability)*

⇒ *Time performance: **Increased by a factor 5-10***

⇒ *Energy Consumption: **Decreased by a factor 100***

Inria /Renault Zoé (4 Lidars @ 25Hz)

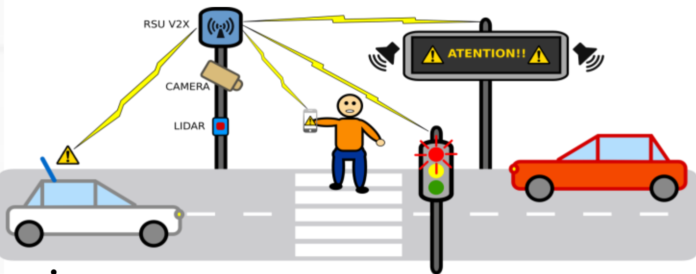


Multi-Sensor Fusion on **µController**

ARM Cortex-M3 @48MHz
- No floating-point
- Power consumption < 1Watt
- Low-cost < 2 €

[1] T. Rakotovao, et al. Multi-Sensor Fusion of Occupancy Grids based on Integer Arithmetic. IEEE ICRA 2016
[2] PhD Thesis T. Rakotovao, Feb 2017

Experimental Platforms & V2X



~200 m long
Parking lots, Road, Intersection
Traffic Lights
Cameras, Road sensors, V2X

Autonomous Renault Zoé

2 Renault Twizy

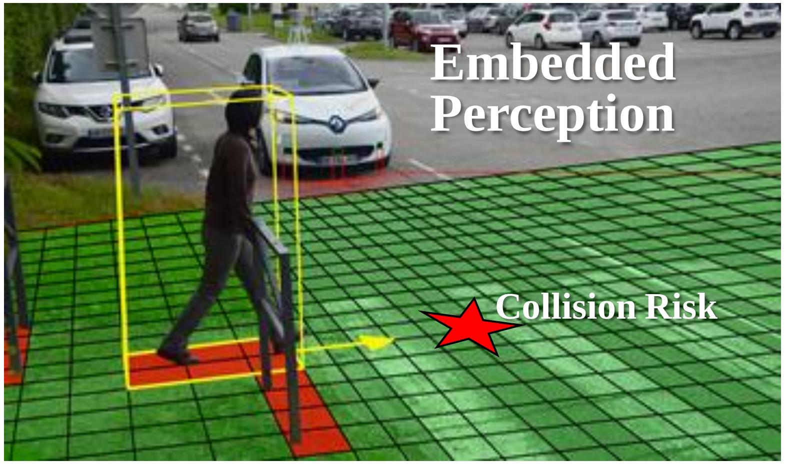
Pedestrian Crash-test platform

Connected Perception Unit

Connected Traffic Cone

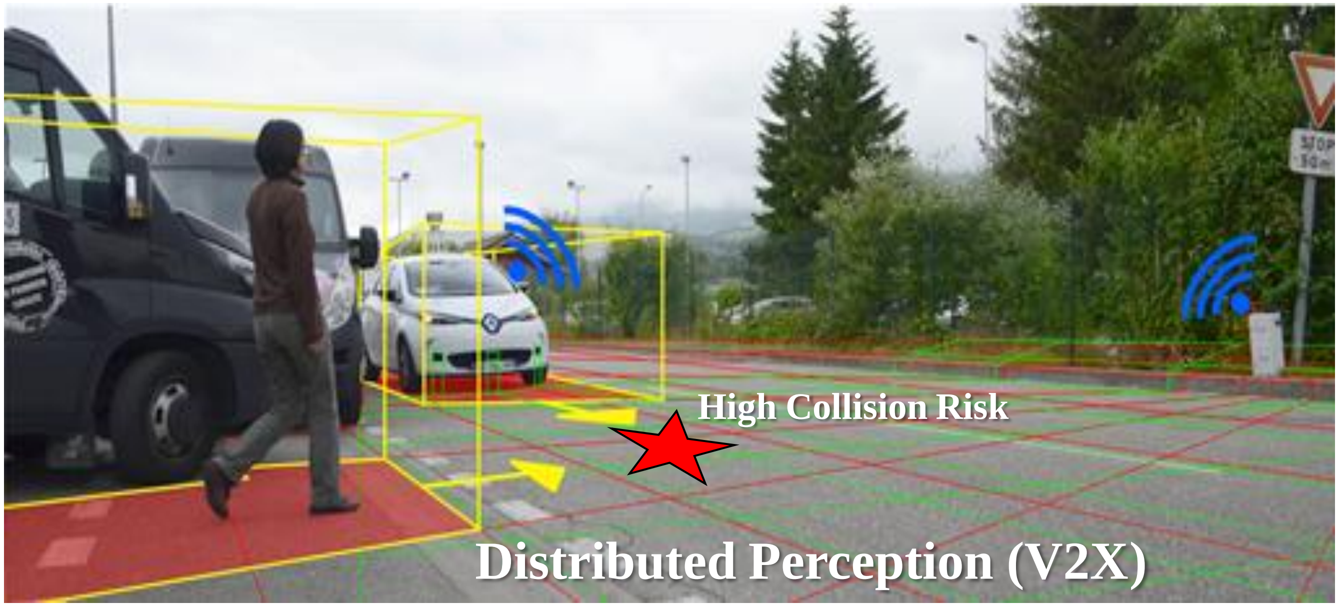
V2X: Extended Collision Risk Assessment

Concept of Distributed Perception



Detection & Collision Risk using embedded Perception

Detection & Collision Risk using Infrastructure Sensors & V2X



V2X: Data exchange & Synchronization

❑ Data exchange

ROS

ITRI

GPS position + Velocity + Bounding box (broadcast)

Collision Risk (CMCDOT) (space & time localization, probability)

ITRI

ITS-G5 (Standard ITS Geonetworking devices)
Basic Transport Protocal IEEE 802.11p

Same perception system (DP-Grids + Risk)

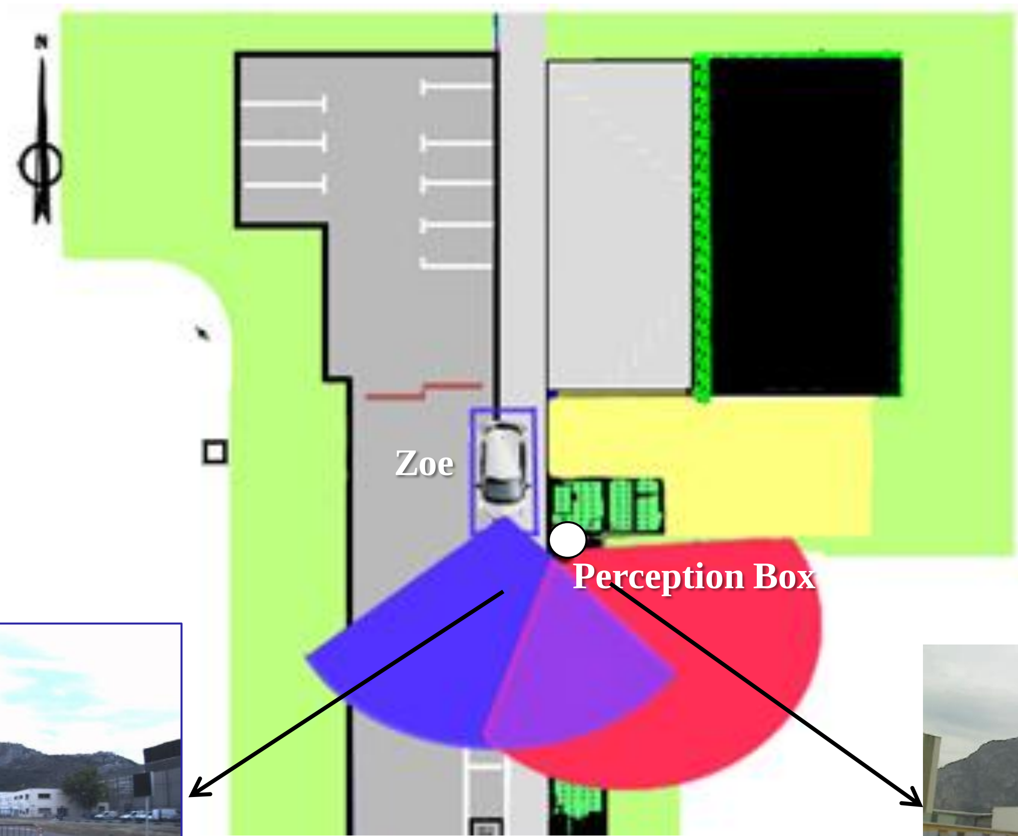
❑ Synchronization

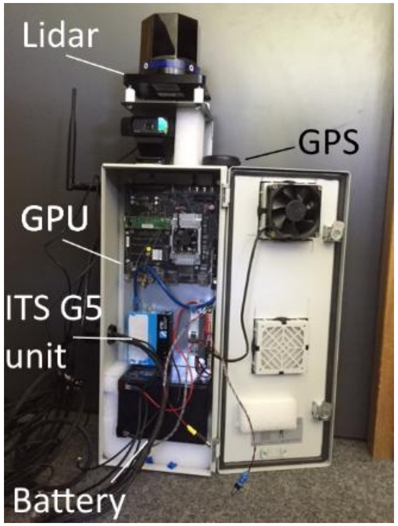
Chrony (**N**etwork **T**ime **P**rotocol)
GPS Garmin + PPS Signal (1 pulse per second)

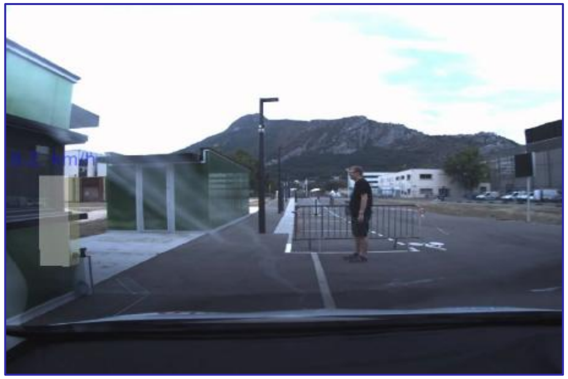
Serial Port

GPIO + UART

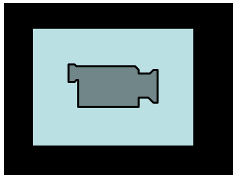
V2X: Distributed Perception Experiment







Camera Image provided by the
Zoe vehicle



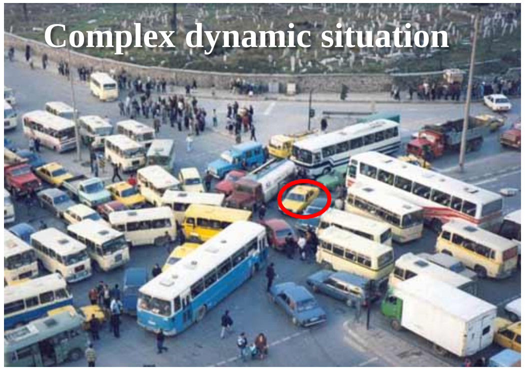
Camera Image provided by the
Perception box

Content of the Lecture

- ❑ Socio-economic & Technological Context
- ❑ Decisional & Control Architecture: Outline
- ❑ Bayesian Perception *(Key Technology 1)*
- ❑ Embedded Perception & Experimental results
- ❑ **Bayesian Risk Assessment & Decision-making *(Key Technology 2)***
- ❑ Conclusion & Perspectives

Key Technology 2: Risk Assessment & Decision

=> Decision-making for avoiding Pending & Future Collisions



❑ Main challenges

*Uncertainty, Partial Knowledge, World changes, **Human in the loop + Real time***

❑ Approach: Prediction + Risk Assessment + Bayesian Decision-making

- ✓ *Reason about **Uncertainty & Contextual Knowledge** (using **History** & **Prediction**)*
- ✓ ***Estimate probabilistic Collision Risk at a given time horizon $t+\delta$***
- ✓ ***Make Driving Decisions** by taking into account the **Predicted behavior** of all the observed surrounding traffic participants (cars, cycles, pedestrians …) & **Social / Traffic rules***

DP-Grids: *Underlying Conservative Prediction Capability*
=> Application to Conservative Collision Anticipation

[Coué & Laugier IJRR 05]

Autonomous Vehicle (Cycab)

Parked Vehicle (occultation)

Pioneer Results (2005)

Thanks to the 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)*

Short-term collision risk – *Main features*

=> Grid level & Conservative motion hypotheses (proximity perception)

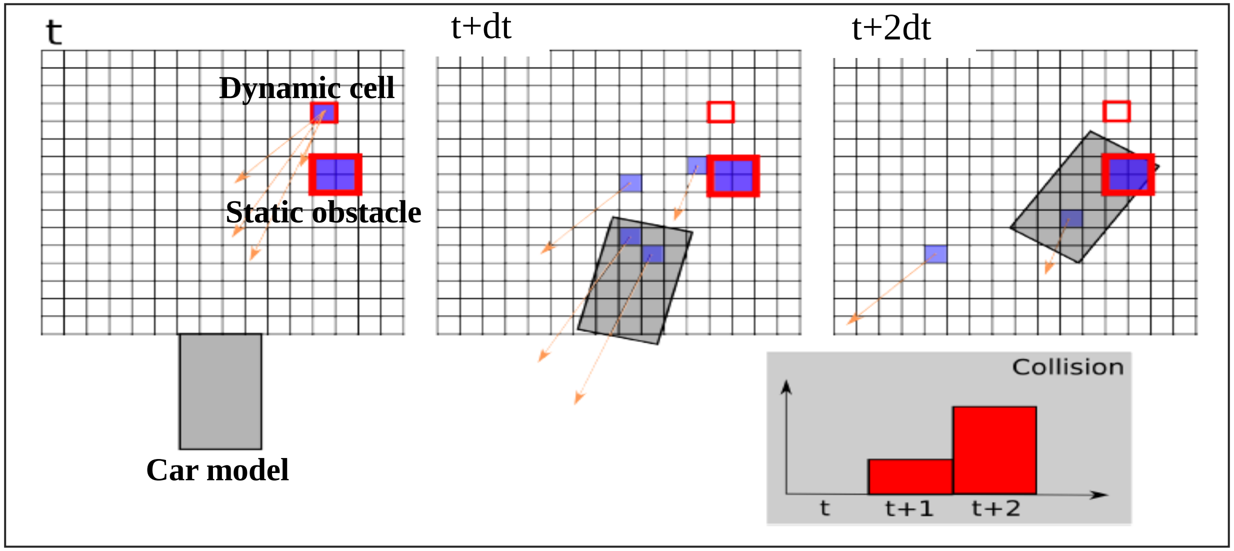
Main Features

Proximity perception: d <100m and t <5s
δ= 0.5 s => Precrash
δ= 1 s => Collision mitigation
δ > 1.5s => Warning / Emergency Braking

- *Detect **"Risky Situations"** a few seconds ahead (0.5 to 3s)*
- *Risky situations are **localized in <u>Space & Time</u>***
  - ⇒ ***Conservative Motion Prediction** in the grid (Particles & Occupancy)*
  - ⇒ ***Collision checking** with **Car model** (Shape & Velocity) for **<u>every future time steps</u>** (horizon h)*
- *Resulting information can be used for choosing **Avoidance Maneuvers***

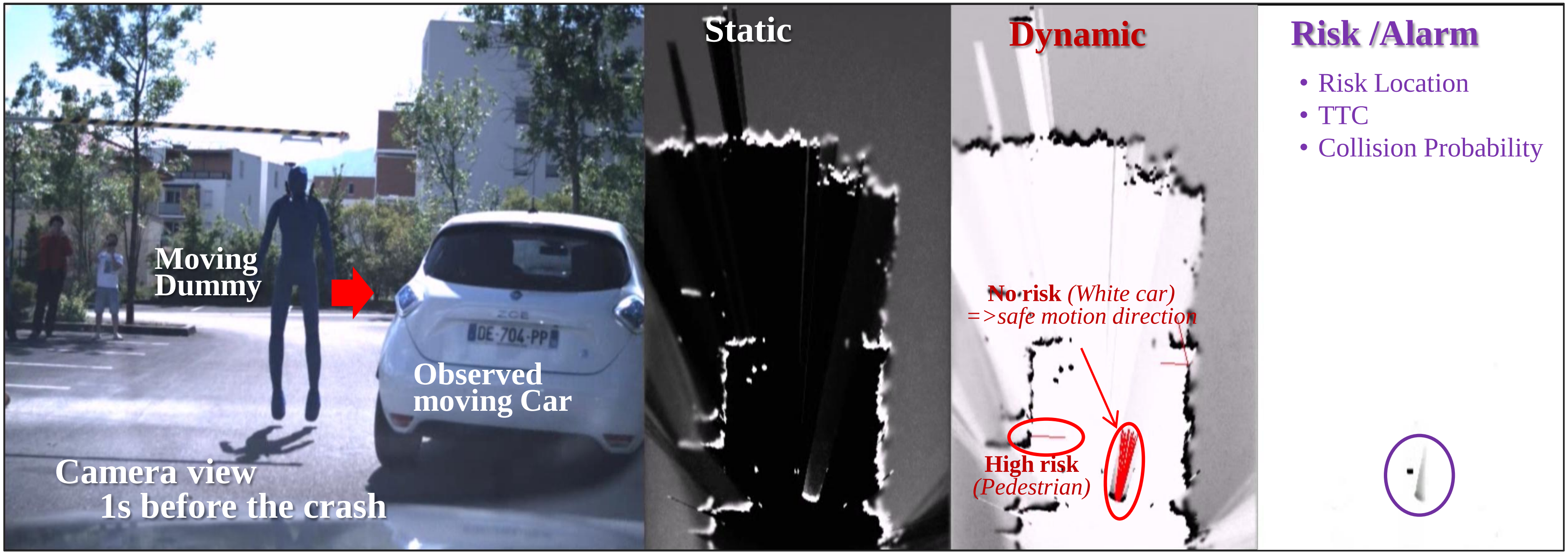
Collision Risk Estimation

✓ *Projecting over time the estimated **Scene changes** (DP-Grid) & **Car Model** (Shape + Motion)*
✓ *Evaluate the Collision Risk for every next time step*
✓ *Integration of risk over a time range [t t+δ]*



Short-term collision risk – *System outputs*

=> Static & Dynamic grids + Risk assessment



Short-term collision risk – *Experimental results*

Objectives:

- ✓ *Detect most of potential future collisions*
- ✓ *Reduce drastically false alarms*

Urban street experiments

=> Almost no false alarm (car, pedestrians …)

Crash scenario on test tracks

=> Almost all collisions predicted before the crash (0.5 – 3 s before)

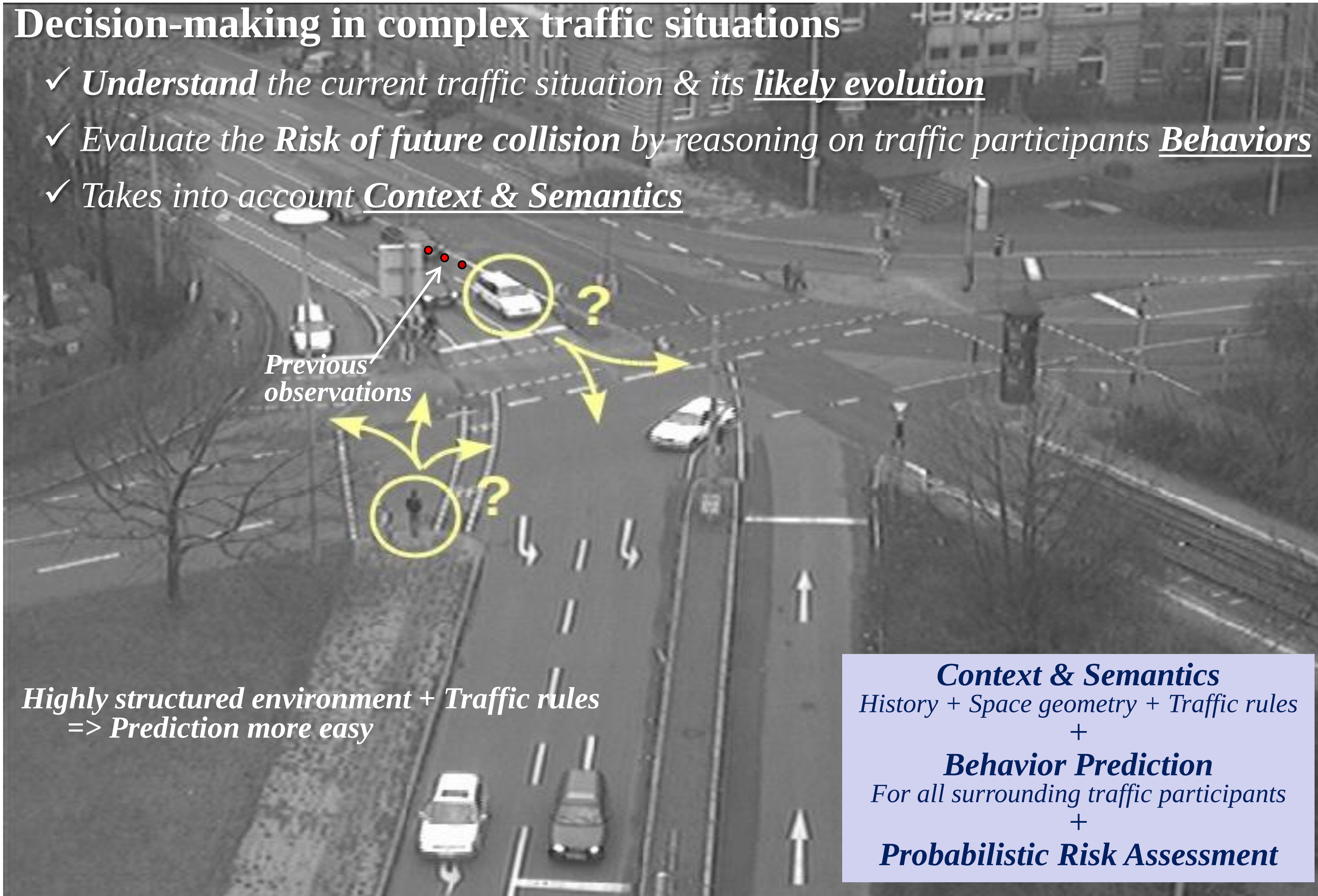
Generalized Risk Assessment (Object level)

=> Increasing time horizon & complexity using ***context & semantics***

=> Key concept: ***Behaviors Modeling & Prediction***

Decision-making in complex traffic situations

- ✓ ***Understand*** *the current traffic situation & its* ***likely evolution***
- ✓ *Evaluate the* ***Risk of future collision*** *by reasoning on traffic participants* ***Behaviors***
- ✓ *Takes into account* ***Context & Semantics***



Highly structured environment + Traffic rules
=> Prediction more easy

Context & Semantics
History + Space geometry + Traffic rules
+
Behavior Prediction
For all surrounding traffic participants
+
Probabilistic Risk Assessment

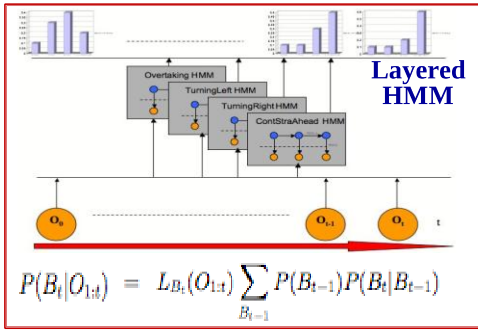
Behavior-based Collision risk *(Object level)*

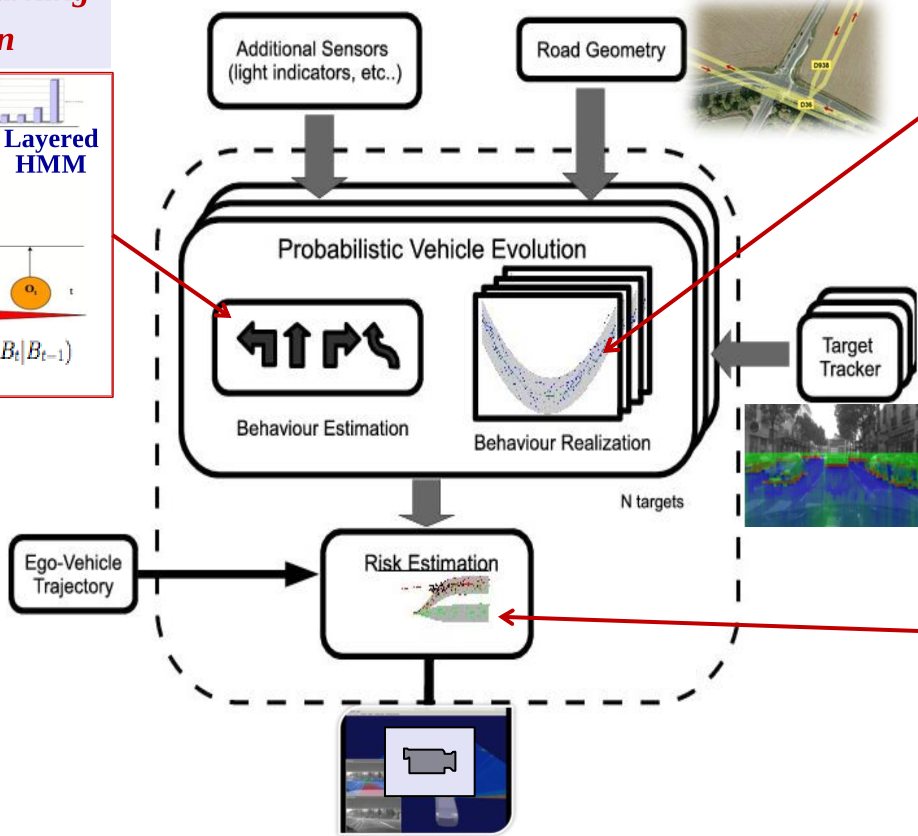
Approach 1: Trajectory prediction & Collision Risk Assessment

[Tay thesis 09] [Laugier et al 11]

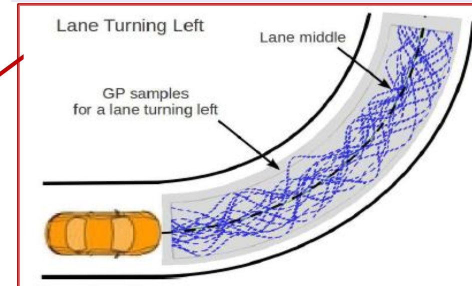
Patent Inria & Toyota & Probayes 2010

Behavior modeling & learning
+
Behavior Prediction



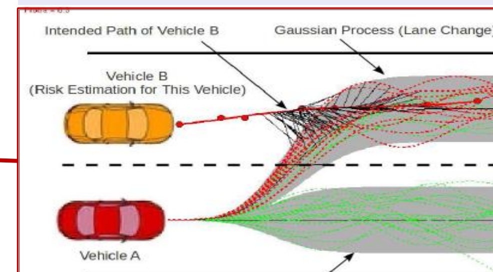


From behaviors to trajectories

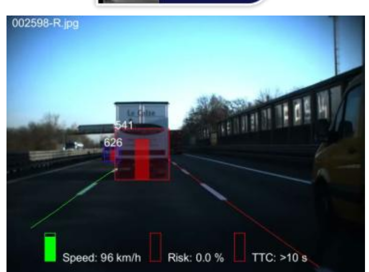


Gaussian Process + LSCM

Collision risk assessment
(Probabilistic)



MC simulation



Courtesy Probayes

Experimental Results
Behavior prediction & Risk Assessment on highway
Probayes & Inria & Toyota

Behavior-based Collision risk *(Object level)*

Approach 2: Intention & Expectation comparison

*=> Complex scenarios with **interdependent behaviors & human drivers***

[Lefevre thesis 13] [Lefevre & Laugier IV'12, Best student paper]

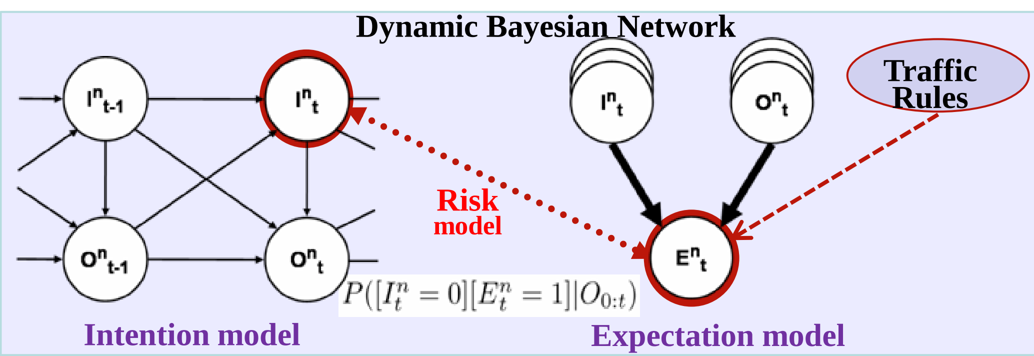
Patent Inria & Renault 2012 *(risk assessment at road intersection)*
Patent Inria & Berkeley 2013 *(postponing decisions for safer results)*

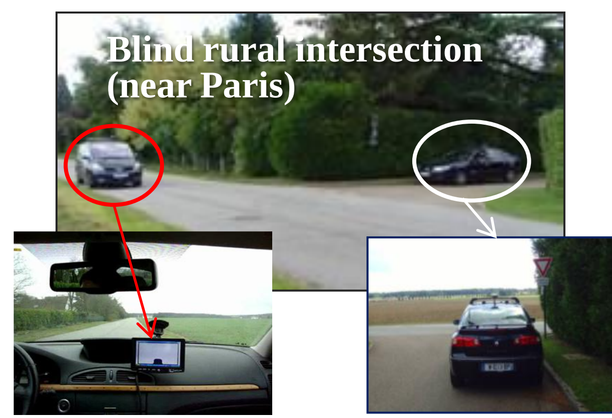
A Human-like reasoning paradigm *=> Detect Drivers Errors & Colliding behaviors*

- ✓ *Estimating **"Drivers Intentions"** from Vehicles States Observations (X Y θ S TS) => Perception or V2V*
- ✓ *Inferring **"Behaviors Expectations"** from Drivers Intentions & Traffic rules*
- ✓***Risk** = Comparing Maneuvers **Intention & Expectation***

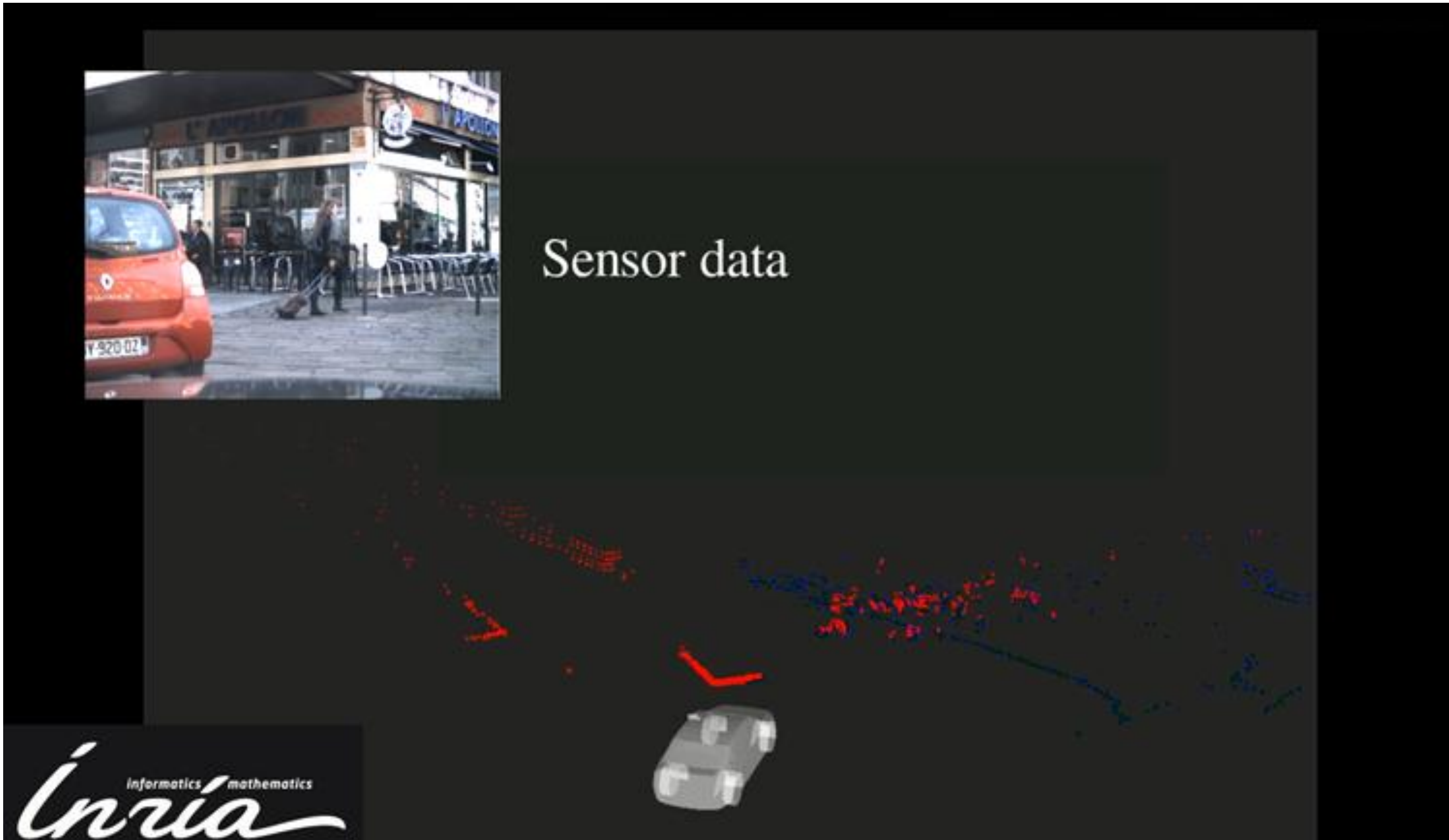
*=> Taking **traffic context** into account (Topology, Geometry, Priority rules, Vehicles states)*

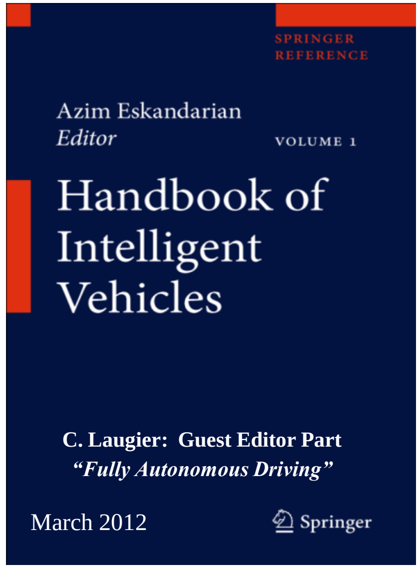
*=> **Digital map** obtained using "Open Street Map"*

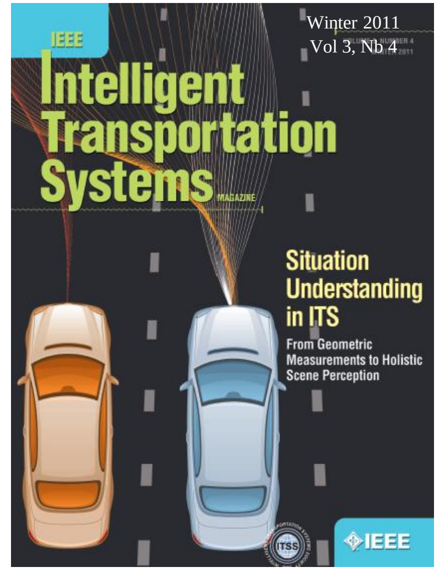


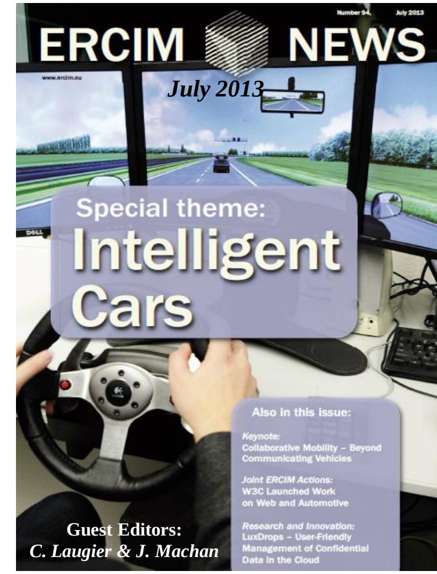


CMCDOT – Experimental results in urban environment
Annotated Video

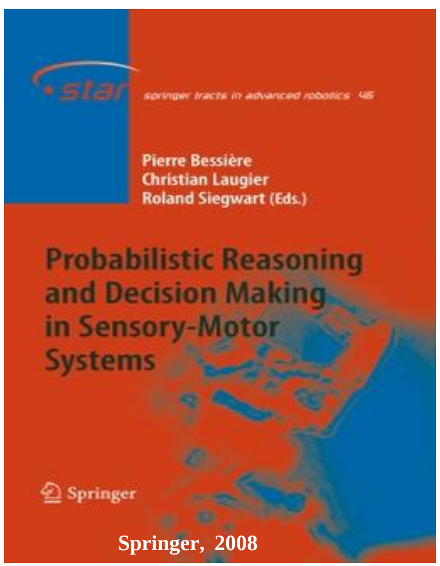


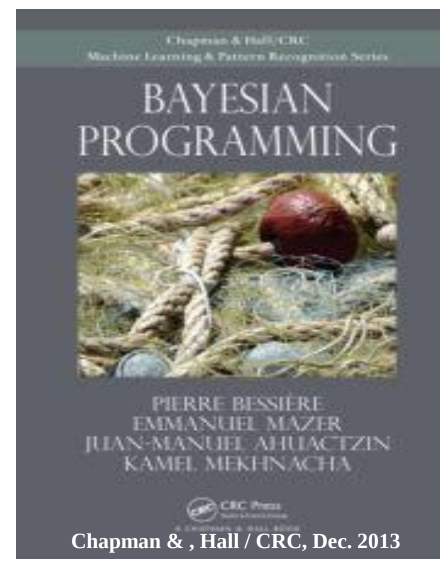






Thank You Any questions ?





christian.laugier@inria.fr