Using Fast Classification of Static and Dynamic Environment for Improving Bayesian Occupancy Filter (BOF) and Tracking

Qadeer Baig, Mathias Perrollaz, Jander Botelho Do Nascimento, Christian Laugier

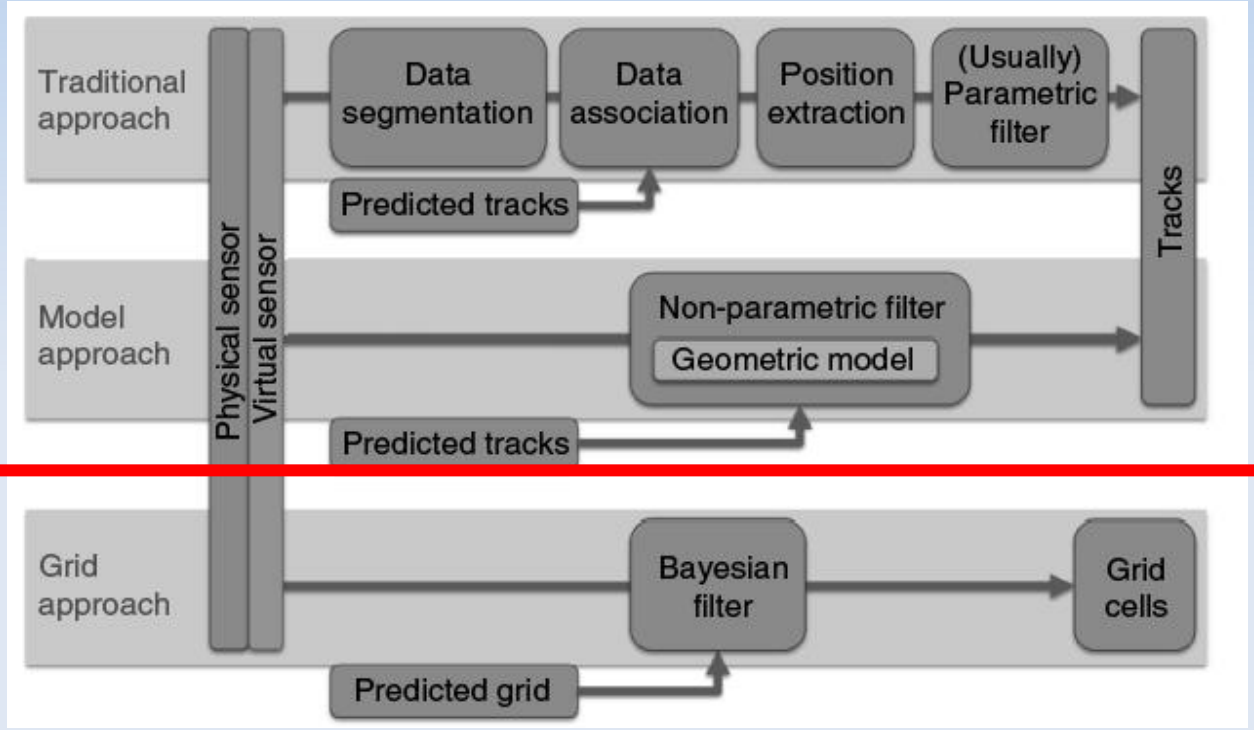
e-Motion team,
Inria Rhône-Alpes (Grenoble), France

Outlines

- Environment monitoring in the Bayesian Occupancy Filter (BOF) framework and FCTA
- Fast Motion Detection
- Integration with BOF and FCTA
- Results

Grid based DATMO

- For mobile robots: Detection and Tracking of Moving Objects (DATMO) is essential for navigation
- For intelligent vehicles : DATMO is essential for risk estimation
- Three main approaches for DATMO [*Petrovskaya11*]:



Bayesian Occupancy Filter (BOF)

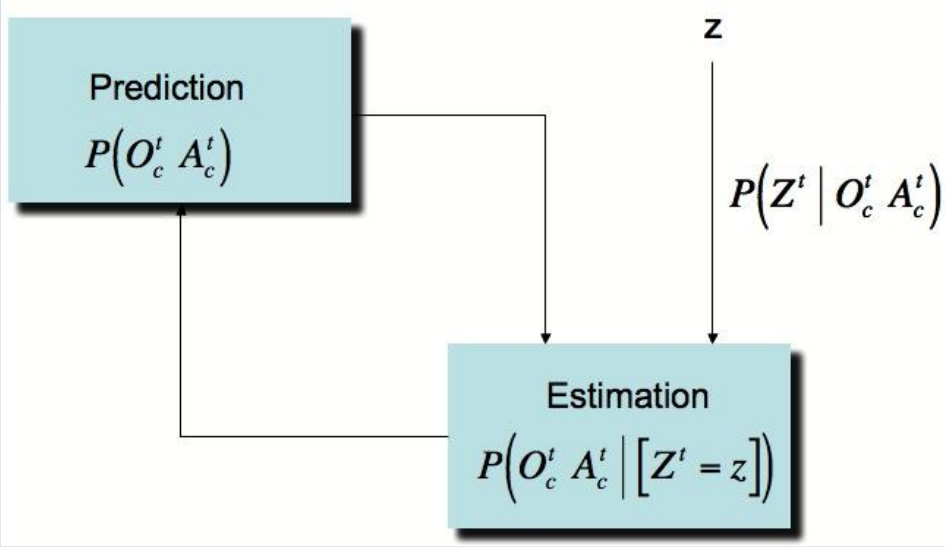
[Coué IJRR 2005]

- Grid-based approach for Bayesian Filtering
  - **Prediction/estimation** loop
  - Each cell has an estimated **occupancy** and a probability distribution over possible **velocities**

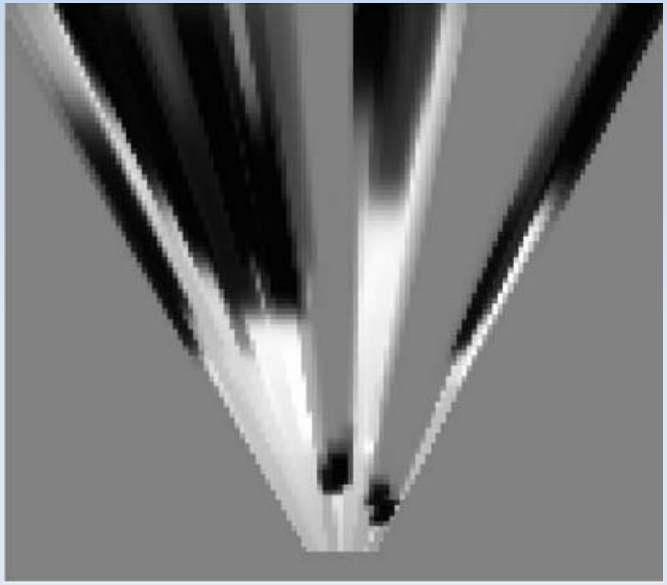
Allows to estimate velocities from grid measurements

- **Prediction:** propagates occupancy and velocity to neighboring cells, using dynamic models

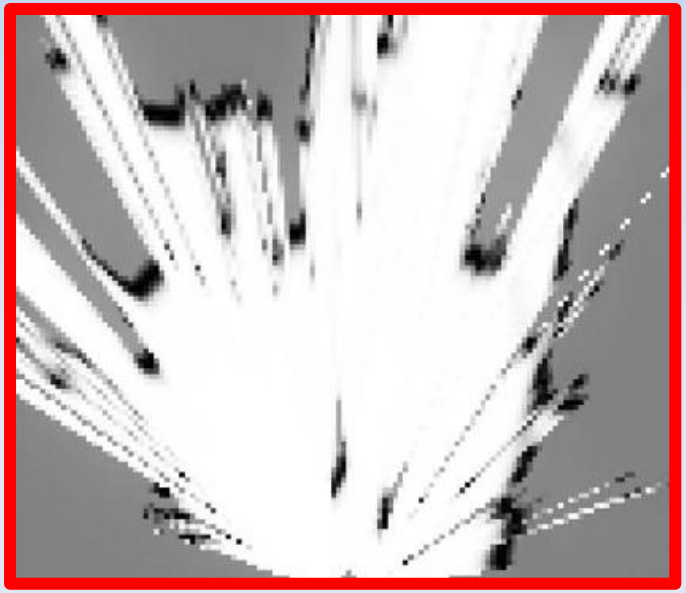
- **Estimation:** corrects predicted grids using observation grids computed using sensor model



BOF : input grids



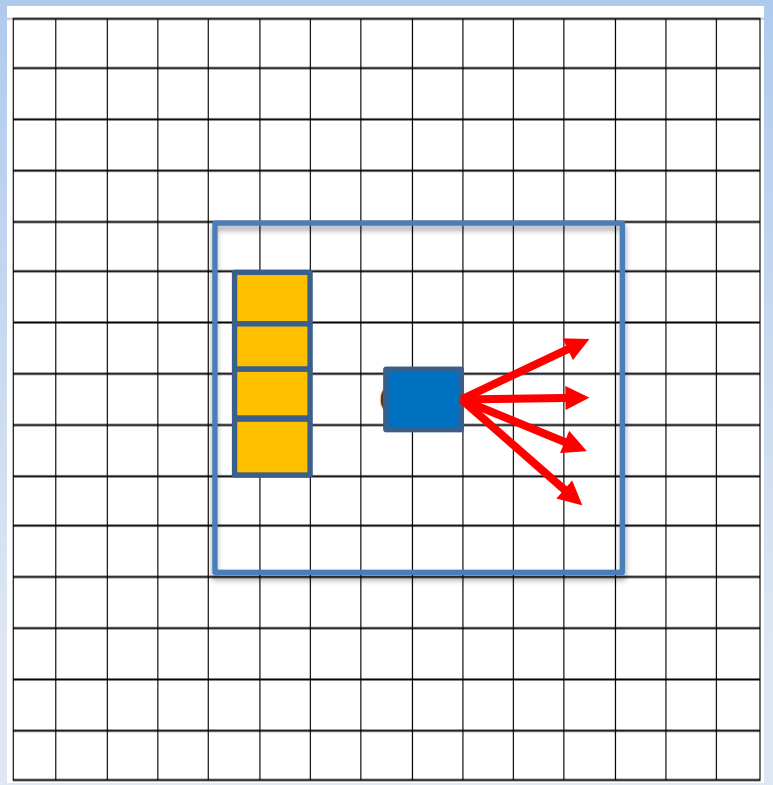
Using stereo-vision
Perrollaz, T-ITS 2012



Using multi-layers laser scanners
Adarve, ICRA 2012

BOF Velocities and FCTA

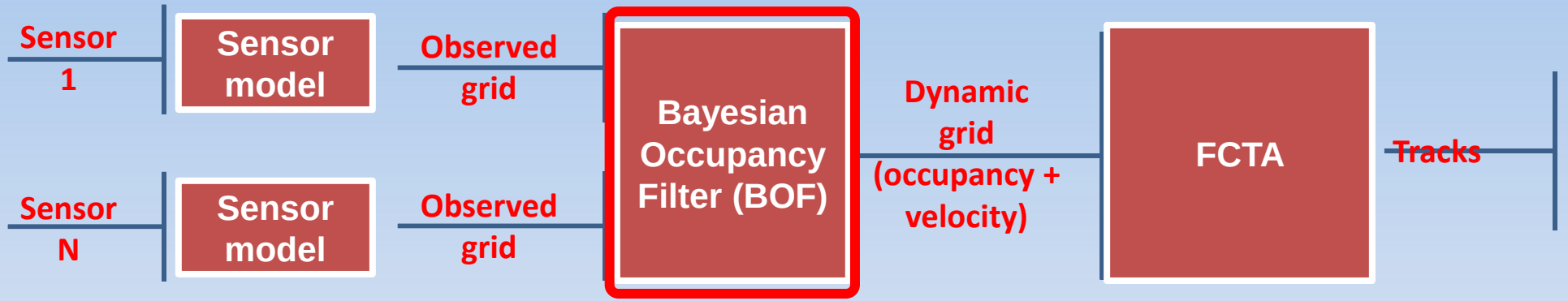
- Cell antecedents: knowing antecedent of a cell tells its velocity
  - Antecedent is the cell at t-1
  - Antecedent only in a neighborhood
  - Distribution over all antecedents
  - Relative velocities

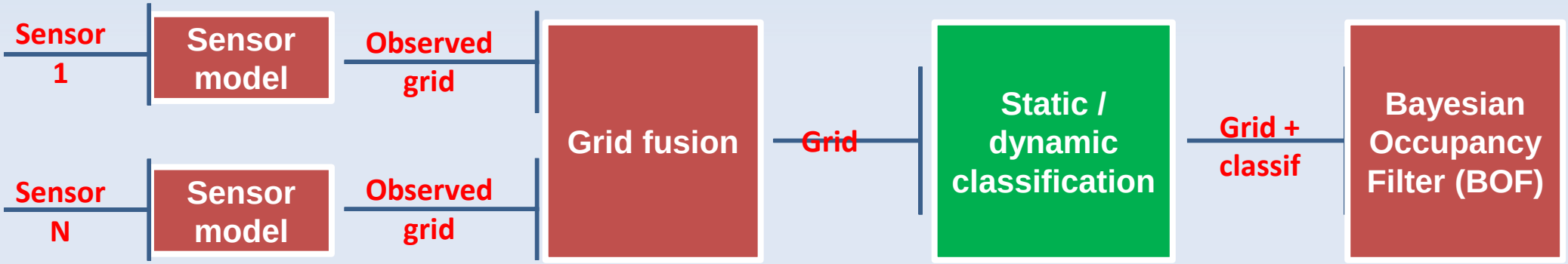


- FCTA: *[Mekhnacha 2008]*
  - Fast Clustering and Tracking Algo
  - Clustering based on cell occ and vel
  - Too many tracking hypotheses
  - Static objects are also tracked
  - Many parameters to tune
  - Results depend on parameters values
  - Convergence is slow in large regions

Solution: Finding static parts before using the BOF

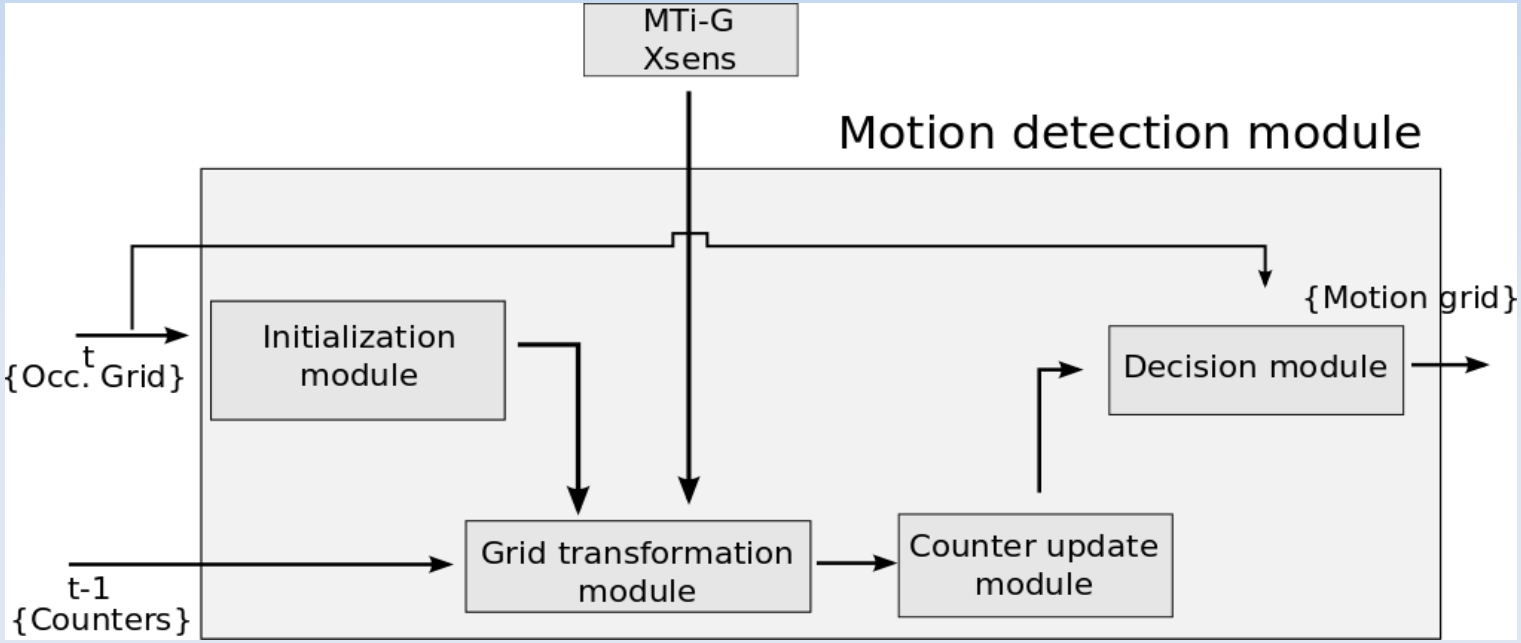
System Architecture





Fast Motion Detection

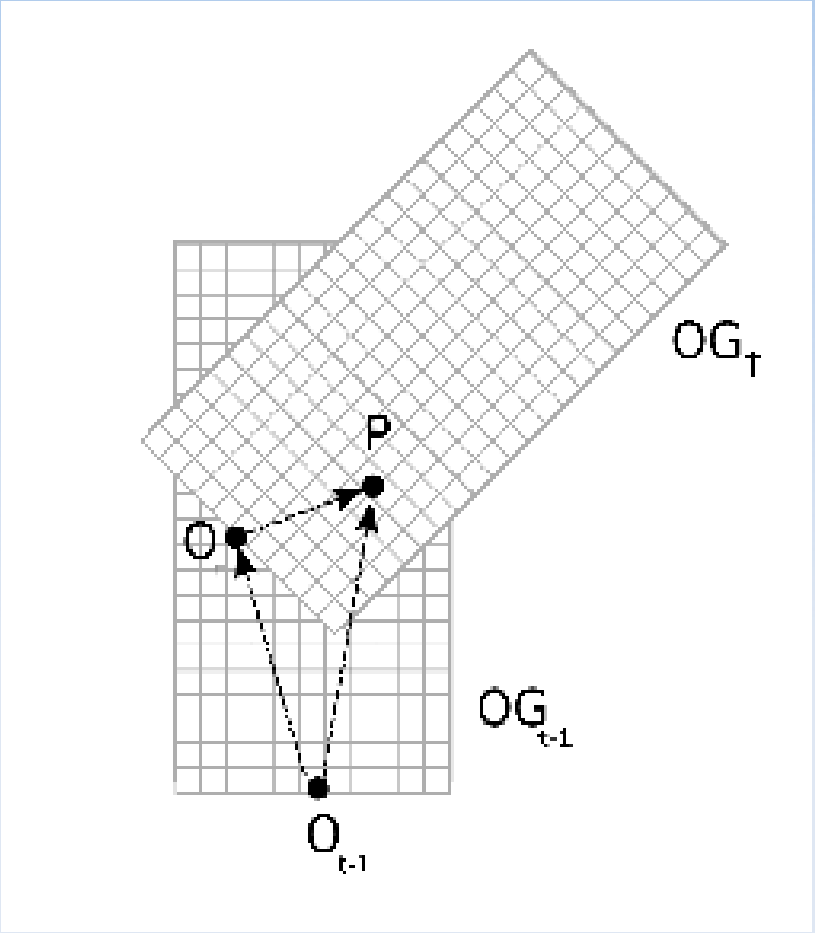
- Main idea: How many times a cell is observed as free and how many times occupied, in a global coordinate system
  - Use free/occupiecounters for each cell
  - Map cells from t-1 to t, using robot's motion
  - Update counters at each timestep
  - Framework:



Grid Transformation

- The objective is to map a cell *j* in grid OGt-1 to cell *i* in grid Ot, with the hypothesys of static environnement

- Method :
  - Using motion data from IMU
  - Velocity, Angular velocity
  - and circular motion model find
  - pose of Ot w.r.t Ot-1



Initialization and update

- Initialization:
  - FreeCountert[i] = 1, if OGt[i] < 0.5
  - OccupiedCountert[i] = 1, if OGt[i] > 0.5

- Updating counters from previous time step :
  - Mapping of cells of grid at time t-1 to grid at t
  - Update counters:
  - FreeCountert[i] += FreeCountert-1[j]
  - OccupiedCountert[i] += OccupiedCountert-1[j]

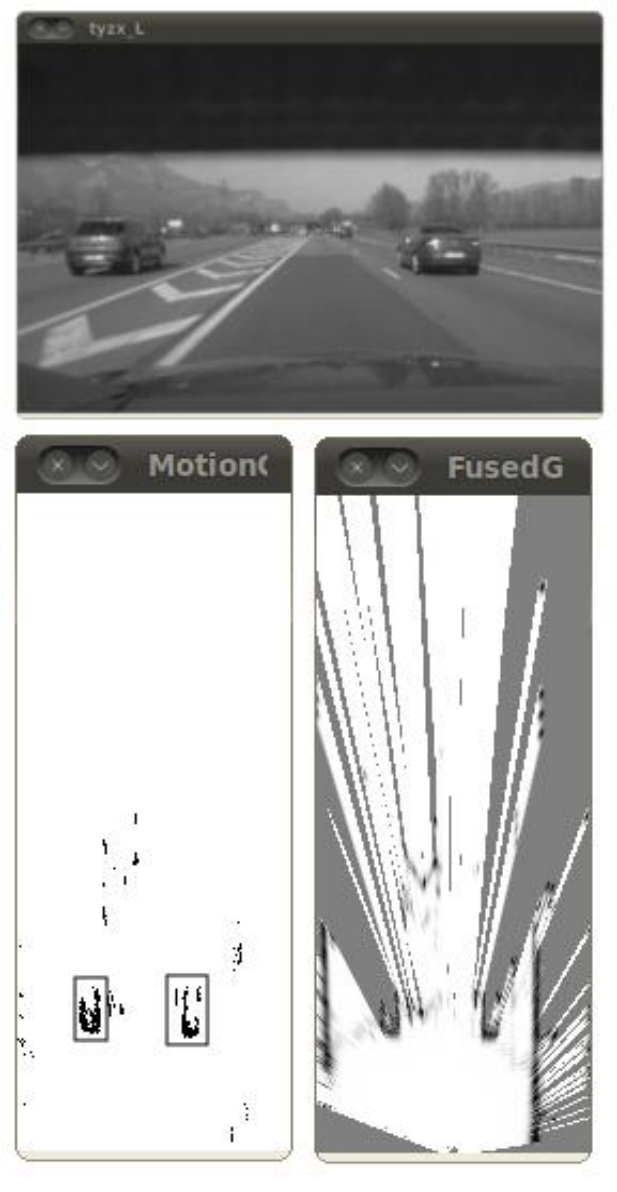
- Decision
  - MotionGrid[i] = F(OGt[i], FreeCountert[i], OccupiedCountert[i])

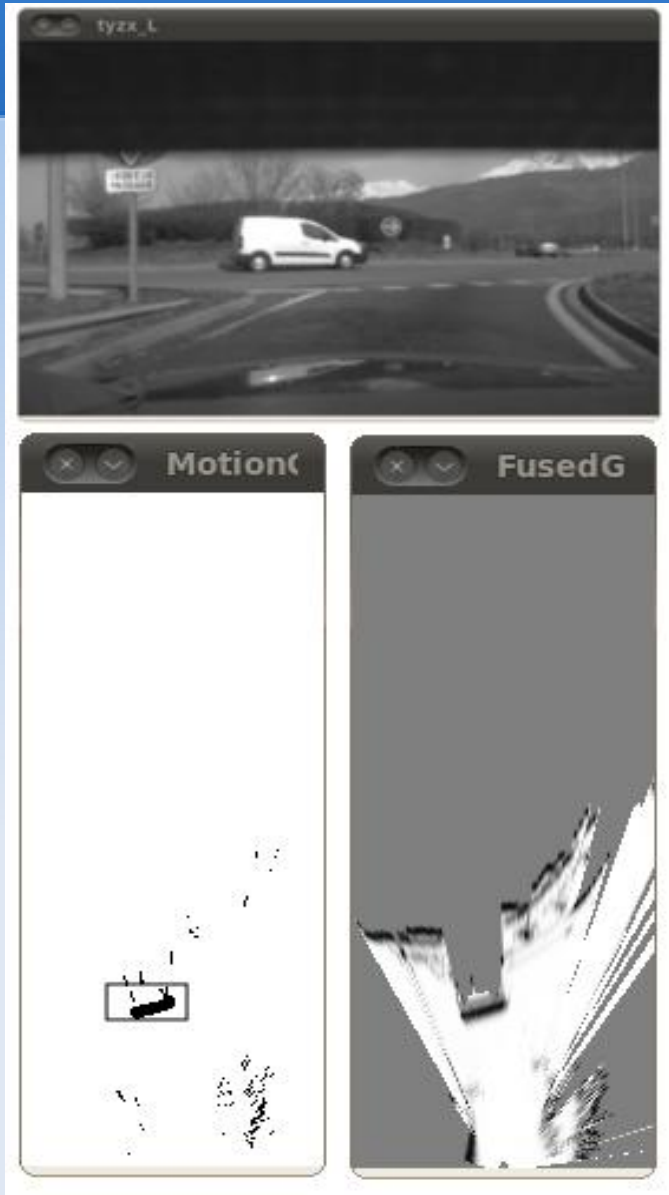
Very fast: Do not solve complete SLAM

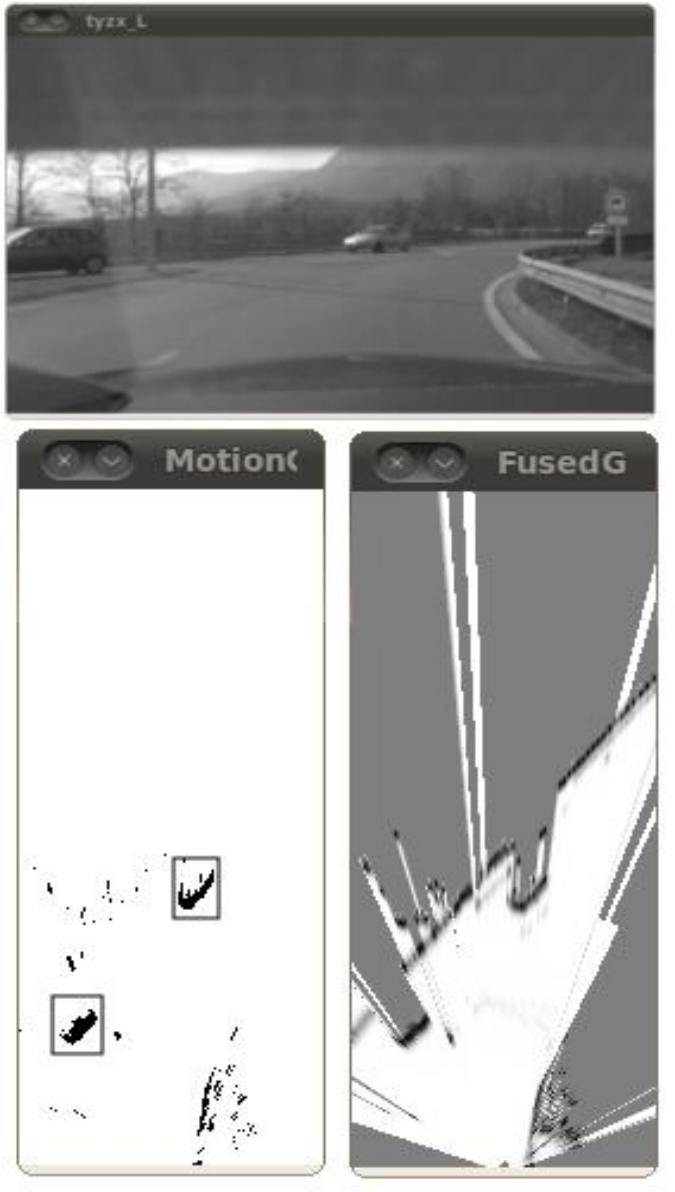
Results: Plateform

- 2 IBEO Lux laser scanners
- (4 layers each)
- 1 TYXZ stereo camera
- (baseline 22cm)
- Xsens MTI-G inertial sensor

Motion Detection Results







Integration with BOF and FCTA

With BOF: Modified velocity update step, update only if cell is moving. This means remove velocity information for static cells

$$P(A_i^t|A_i^{t-1}) = \begin{cases} (1-\epsilon)P(A^{t-1}_{A_i^{t-1}}) + \epsilon/\|A_i\| \\ \quad \text{if } MotionGrid_t[i] > 0 \\ \\ 1/\|A_i\| \quad \text{otherwise} \end{cases}$$

With FCTA: Modified clustering step, take a cell into account for clustering if it has velocity information

Result: More then 78% false positives removed with relaxed FCTA parameters

Results

Videos

Thank You!

Questions?