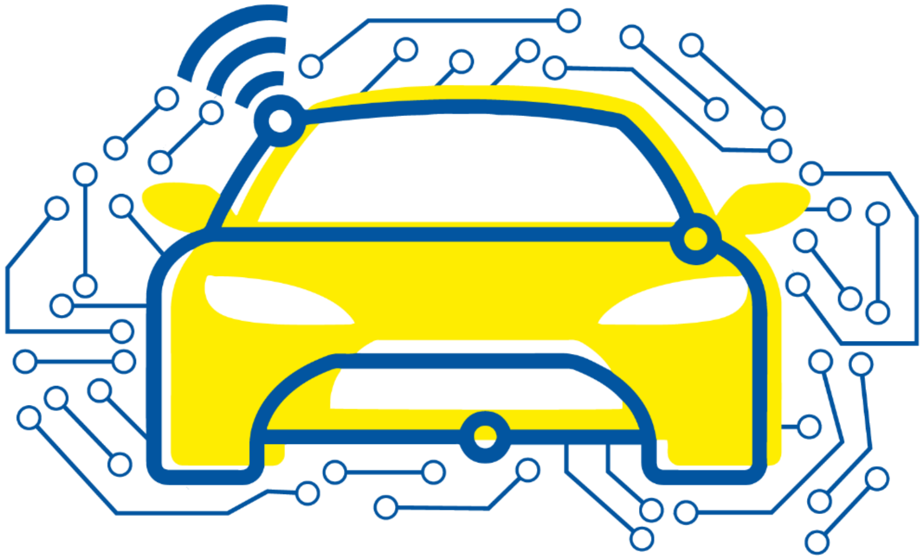
Automated and Connected Driving Challenges

Section 3 – Object Fusion and Tracking

Object Prediction



Bastian Lampe

Institute for Automotive Engineering

Object Fusion and Tracking

Mathematical Notation

Type

$\widehat{(.)}$	Estimate of the true value of $(.)$
$(.)^T$	Transpose of $(.)$

Indices

G	Results from a global fusion algorithm / global environment model
S	Originates from sensor S

Object Fusion and Tracking

Object state vector

$\boldsymbol{O}_i = (\hat{\boldsymbol{x}} \;\; \boldsymbol{P})^T$	(object)
$\hat{\boldsymbol{x}} = \left(x\ y\ z\ v_x\ v_y\ a_x\ a_y\ l\ w\ h\right)^T$	(estimated object *state* vector)
$\boldsymbol{P} = cov(\hat{\boldsymbol{x}}_{err}, \hat{\boldsymbol{x}}_{err})$	(uncertainty of object state; described by its error *covariance* matrix)
x, y, z	(bounding box center position)
v_x, v_y, a_x, a_y	(velocities and accelerations)
l, w, h	(bounding box length, width, height)
$\hat{\boldsymbol{x}}_S$	measured state vector (“S”=sensor)
$\hat{\boldsymbol{x}}_G$	global/fused state vector (“G”=global)

Object Fusion and Tracking

Object state vector

$\boldsymbol{O}_i = (\hat{\boldsymbol{x}} \;\; \boldsymbol{P})^T$ (object)

$\hat{\boldsymbol{x}} = \left(x\ y\ z\ v_x\ v_y\ a_x\ a_y\ l\ w\ h\right)^T$ (estimated object *state* vector)

$\boldsymbol{P} = cov(\hat{\boldsymbol{x}}_{err}, \hat{\boldsymbol{x}}_{err})$ (uncertainty of object state; described by its error *covariance* matrix)

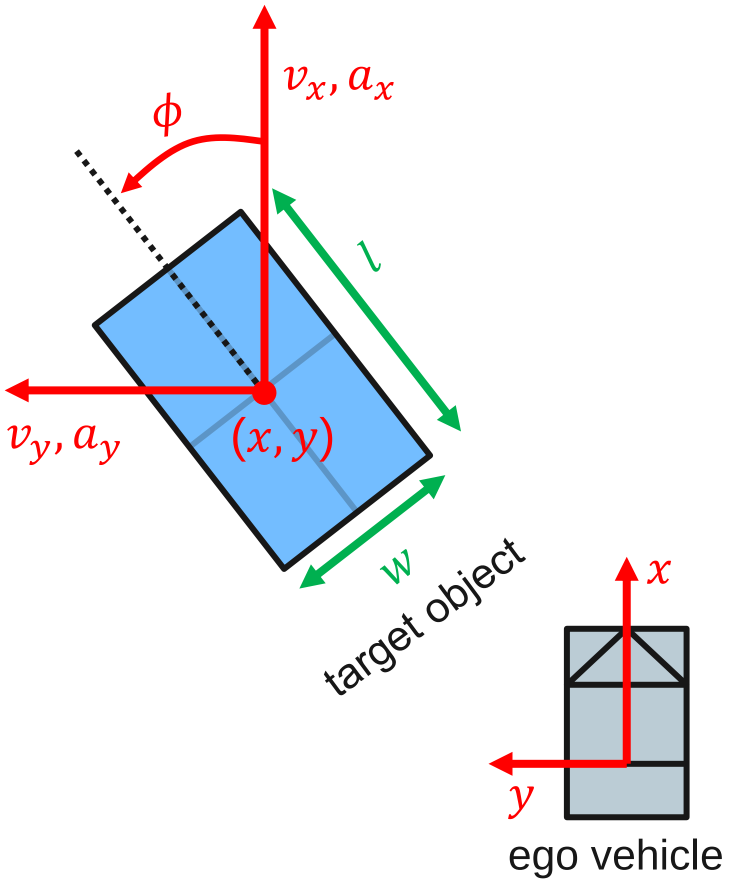
x, y, z (bounding box center position)

v_x, v_y, a_x, a_y (velocities and accelerations)

l, w, h (bounding box length, width, height)

$\hat{\boldsymbol{x}}_S$ measured state vector ("S"=sensor)

$\hat{\boldsymbol{x}}_G$ global/fused state vector ("G"=global)



Object Fusion and Tracking

Object state vector error covariance

$\boldsymbol{P} = cov(\hat{\boldsymbol{x}}_{err}, \hat{\boldsymbol{x}}_{err})$ (uncertainty of object state; described by its error ***covariance*** matrix)

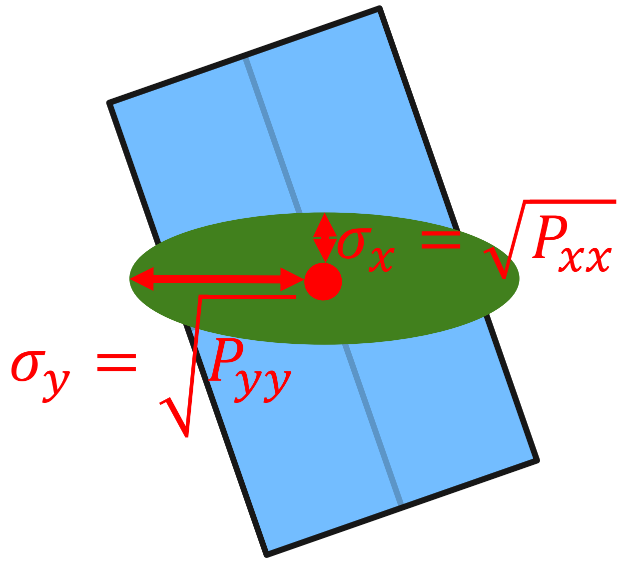
→ A small $\boldsymbol{P}$ means that the error in $\hat{\boldsymbol{x}}$ is small!

Object Fusion and Tracking

Object state vector error covariance

$\boldsymbol{P} = cov(\hat{\boldsymbol{x}}_{err}, \hat{\boldsymbol{x}}_{err})$ (uncertainty of object state; described by its error ***covariance*** matrix)

→ A small $\boldsymbol{P}$ means that the error in $\hat{\boldsymbol{x}}$ is small!



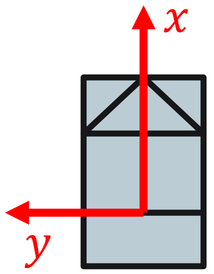
Visualization:

- Boundary of green ellipse shows one standard deviation of the error
- (variances are squared standard deviations)

In this example:

- Variance P_{xx} of estimated x rather *small*, e.g. $0.2m$
- Variance P_{yy} of estimated y rather *large*, e.g. $2.5m$

→ Estimate $\hat{\boldsymbol{x}}$ is more certain in x-direction



In the code, we also consider variances for velocities, accelerations etc.

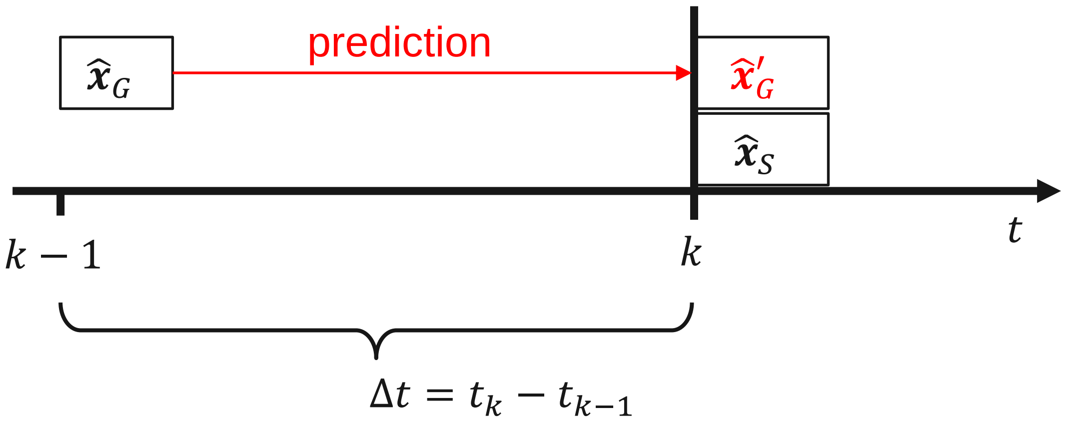
Object Fusion and Tracking

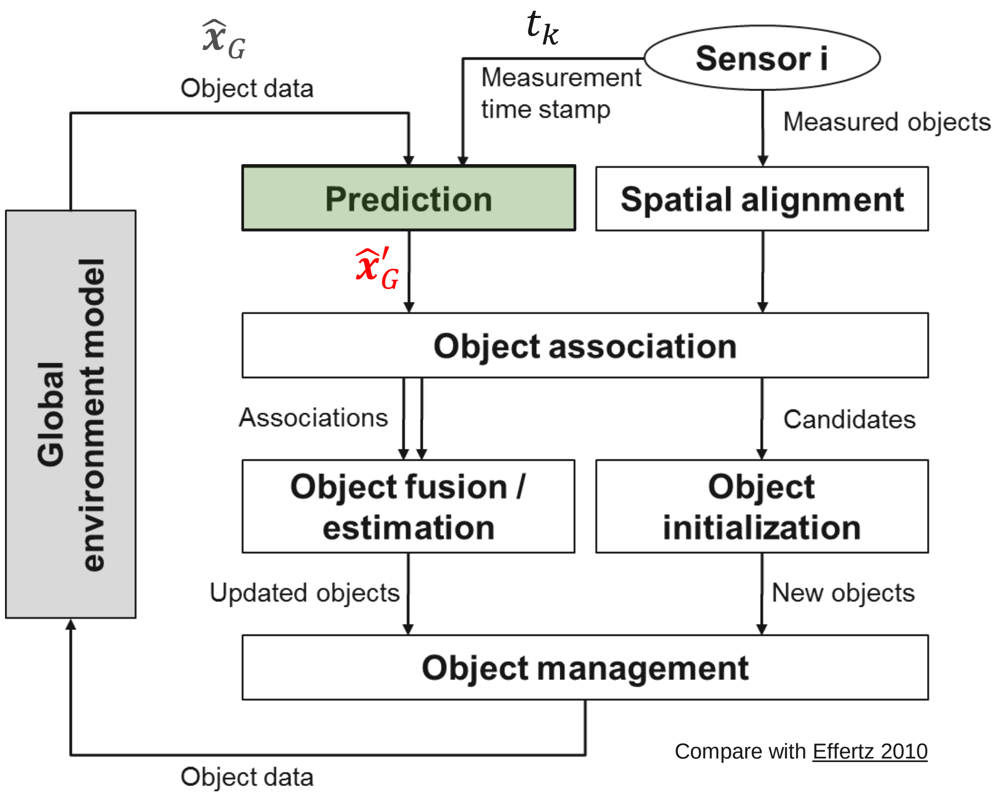
Prediction

- Kalman filter prediction step
- Reason: comparison of objects only makes sense if valid at the same time stamp

→ Predict older object from t_{k-1} to the newer time stamp t_k

  - The older object is the „global“ object G from the previous algorithm run (e.g. 50ms ago)
  - The newer object is the measured object S („sensor-level“) that e.g. the lidar system has just detected





Compare with Effertz 2010

Object Fusion and Tracking

Prediction

$\hat{\boldsymbol{x}}_G \coloneqq \boldsymbol{F}\,\hat{\boldsymbol{x}}_G$ (predict estimated object state)

with

$\boldsymbol{F}$ **Motion model matrix:**
Describes e.g. constant velocity or constant acceleration assumption.

Object Fusion and Tracking

Prediction

$\hat{\boldsymbol{x}}_G \ \coloneqq \ \boldsymbol{F}\,\hat{\boldsymbol{x}}_G$ (predict estimated object state)

with

$\boldsymbol{F}$ **Motion model matrix:**
Describes e.g. constant velocity or constant acceleration assumption.

$\Delta t \ = t_k - t_{k-1}$ „Prediction gap“ between currently measured object and already existing global object (see previous slide).

$$\boldsymbol{F}_{\boldsymbol{constVel}} = \begin{array}{c} \begin{array}{cccccccccc} x & y & z & v_x & v_y & a_x & a_y & l & w & h \end{array} \\ \begin{bmatrix} 1 & & & \Delta t & & & & & & \\ & 1 & & & \Delta t & & & & & \\ & & 1 & & & & & & & \\ & & & 1 & & & & & & \\ & & & & 1 & & & & & \\ & & & & & 1 & & & & \\ & & & & & & 1 & & & \\ & & & & & & & 1 & & \\ & & & & & & & & 1 & \\ & & & & & & & & & 1 \end{bmatrix} \end{array}$$

positions gets predicted by velocities

Object Fusion and Tracking

Prediction

$\hat{\boldsymbol{x}}_G \; := \; \boldsymbol{F}\,\hat{\boldsymbol{x}}_G$ (predict estimated object state)

$\boldsymbol{P}_G \; := \boldsymbol{F}\boldsymbol{P}_G\boldsymbol{F}^T + \boldsymbol{Q}$ (predict estimated object state error covariance)

with

$\boldsymbol{F}$ **Motion model matrix:** Describes e.g. constant velocity or constant acceleration assumption.

$\boldsymbol{Q}$ **Motion noise matrix:** Inaccuracy of the motion model assumption. Also “process noise”.

$\Delta t \; = t_k - t_{k-1}$ „Prediction gap“ between currently measured object and already existing global object (see previous slide).

$$\boldsymbol{F}_{\boldsymbol{constVel}} = \begin{array}{c} \begin{array}{cccccccccc} x & y & z & v_x & v_y & a_x & a_y & l & w & h \end{array} \\ \begin{bmatrix} 1 & & & \Delta t & & & & & & \\ & 1 & & & \Delta t & & & & & \\ & & 1 & & & & & & & \\ & & & 1 & & & & & & \\ & & & & 1 & & & & & \\ & & & & & 1 & & & & \\ & & & & & & 1 & & & \\ & & & & & & & 1 & & \\ & & & & & & & & 1 & \\ & & & & & & & & & 1 \end{bmatrix} \end{array}$$

positions gets
predicted
by velocities

Object Fusion and Tracking

Prediction

$\hat{\boldsymbol{x}}_G \;\coloneqq\; \boldsymbol{F}\,\hat{\boldsymbol{x}}_G$	(predict estimated object state)
$\boldsymbol{P}_G \;\coloneqq\; \boldsymbol{F}\boldsymbol{P}_G\boldsymbol{F}^T + \boldsymbol{Q}$	(predict estimated object state error covariance)

with

$\boldsymbol{F}$	**Motion model matrix:** Describes e.g. constant velocity or constant acceleration assumption.
$\boldsymbol{Q}$	**Motion noise matrix:** Inaccuracy of the motion model assumption. Also “process noise”.
$\Delta t \;= t_k - t_{k-1}$	„Prediction gap“ between currently measured object and already existing global object (see previous slide).

$$
\boldsymbol{F}_{constVel} = \begin{array}{c}
\begin{array}{cccccccccc} x & y & z & v_x & v_y & a_x & a_y & l & w & h \end{array} \\
\begin{bmatrix}
1 & & & \Delta t & & & & & & \\
 & 1 & & & \Delta t & & & & & \\
 & & 1 & & & & & & & \\
 & & & 1 & & & & & & \\
 & & & & 1 & & & & & \\
 & & & & & 1 & & & & \\
 & & & & & & 1 & & & \\
 & & & & & & & 1 & & \\
 & & & & & & & & 1 & \\
 & & & & & & & & & 1
\end{bmatrix}
\end{array}
$$

positions gets
predicted
by velocities