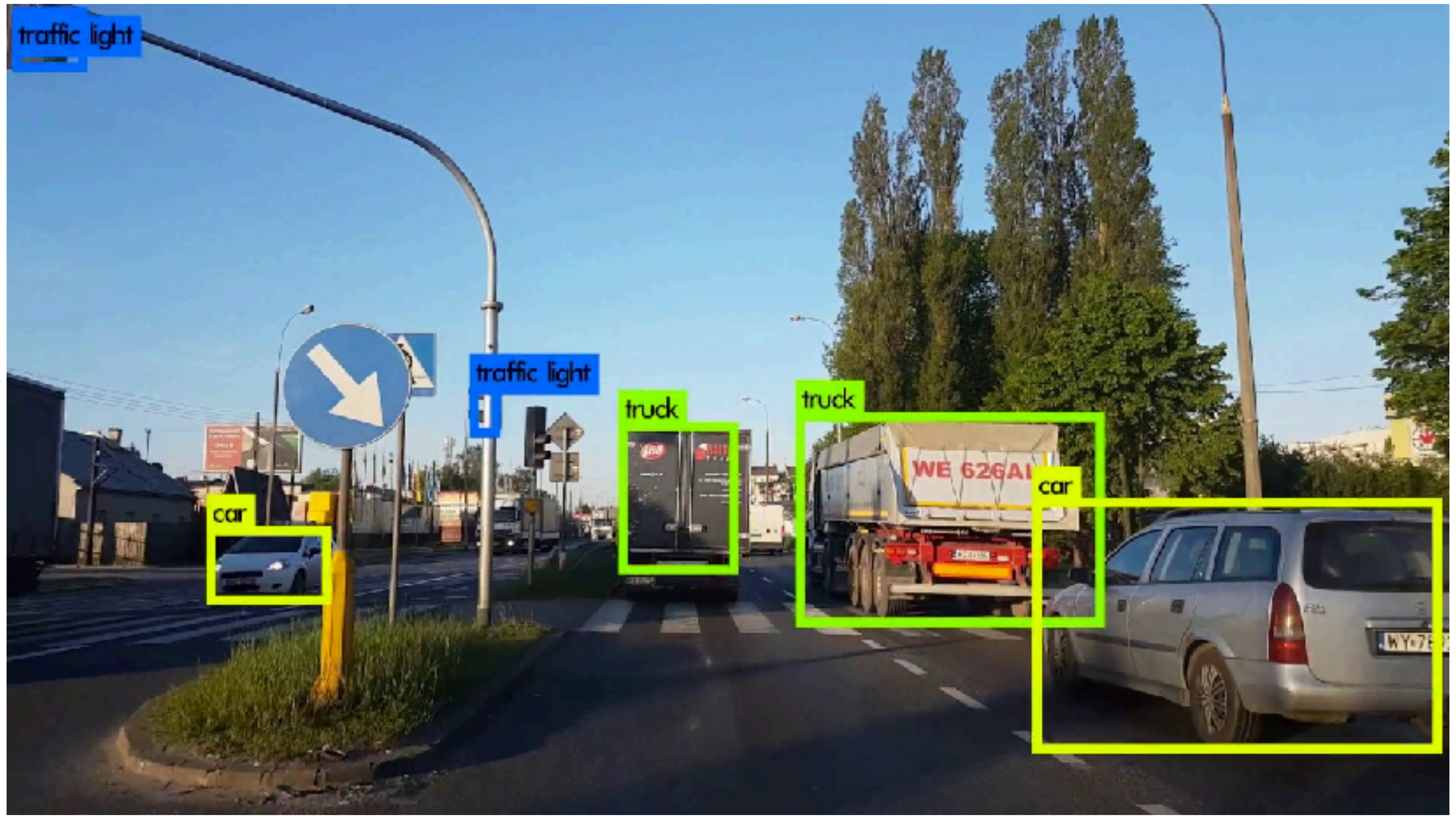
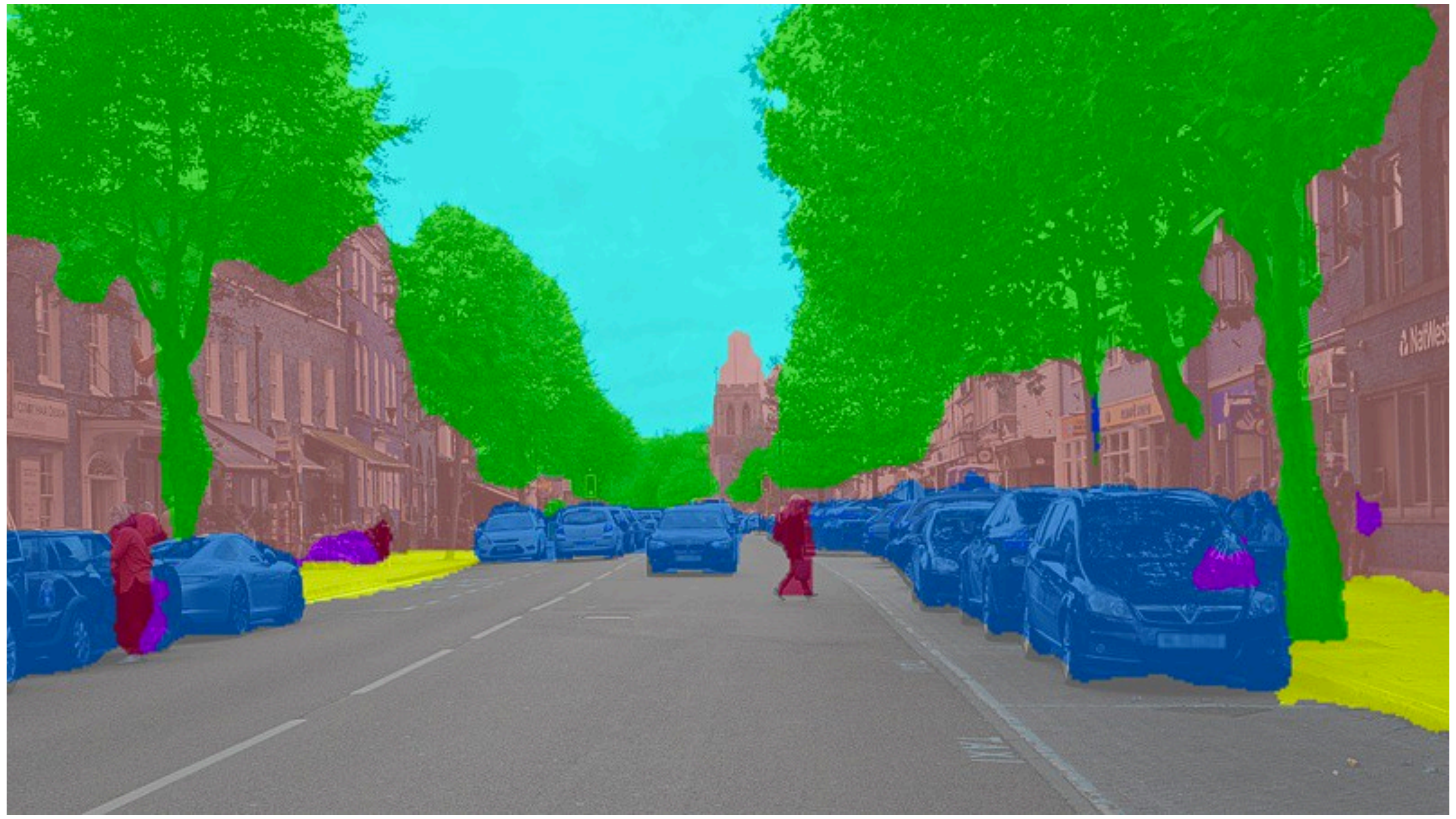
Tracking Measurements Produced by DNNs

Forrest Laine, Anish Muthali, Claire Tomlin

June 10, 2021

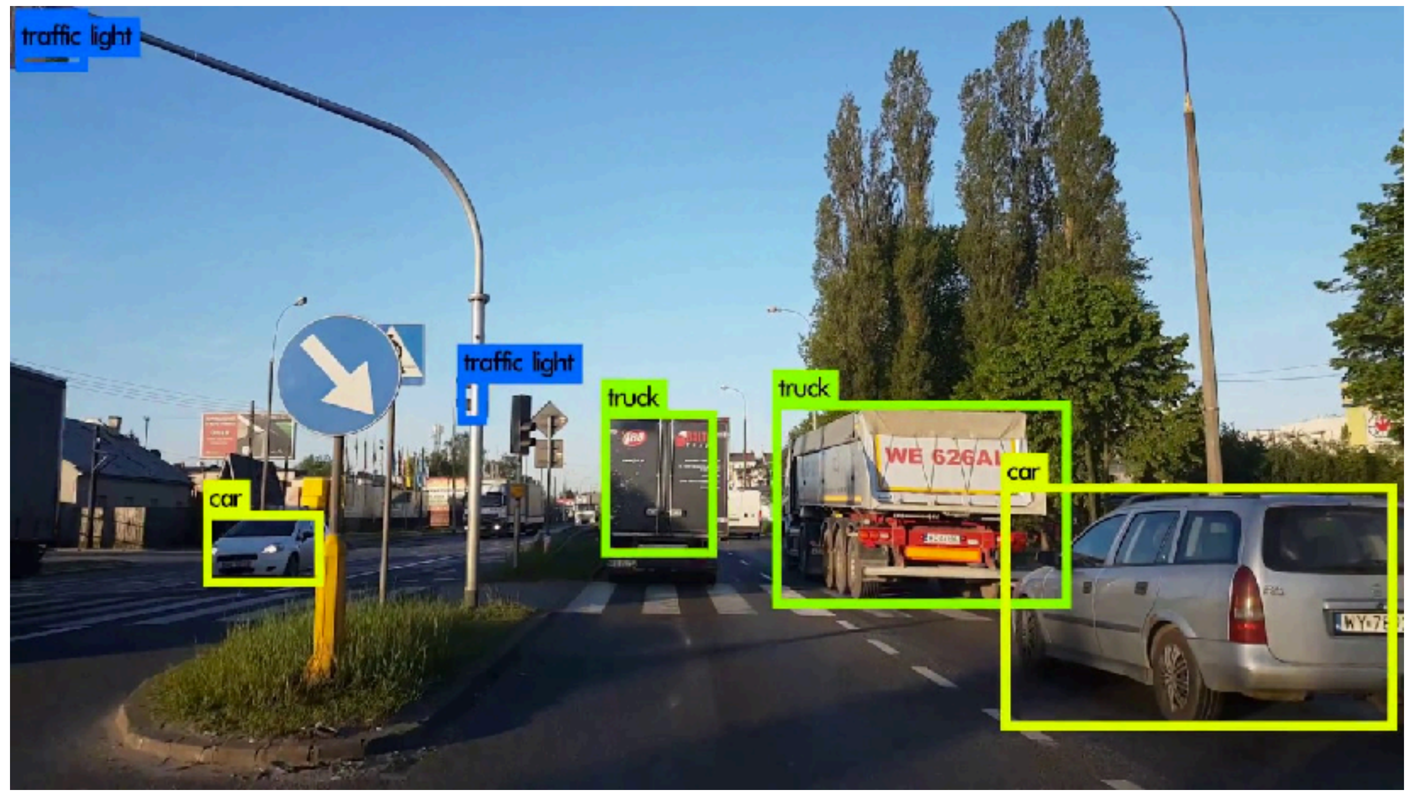
Object Detection from Cameras

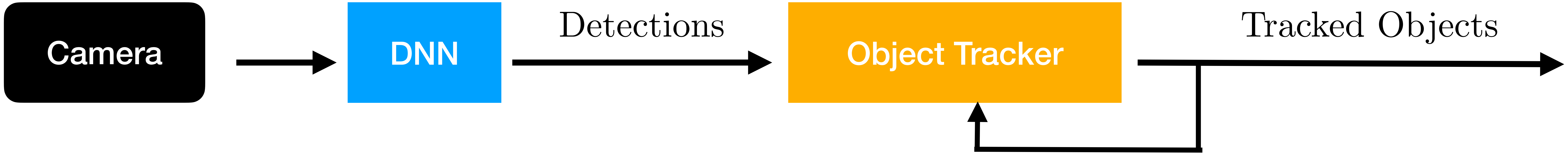




Object Tracking from Cameras

Bounding Box Detections





Chapman-Kolmogorov Updates:

$$p(x_k \mid z_{1:k-1}) = \int p(x_k \mid x_{k-1})\, p(x_{k-1} \mid z_{1:k-1})\, dx_{k-1}$$

$$p(x_k \mid z_{1:k}) = \frac{p(z_k \mid x_k)\, p(x_k \mid z_{1:k-1})}{p(z_k \mid z_{1:k-1})}$$

$$p(z_k \mid z_{1:k-1}) = \int p(z_k \mid x_k)\, p(x_k \mid z_{1:k-1}) dx_k$$

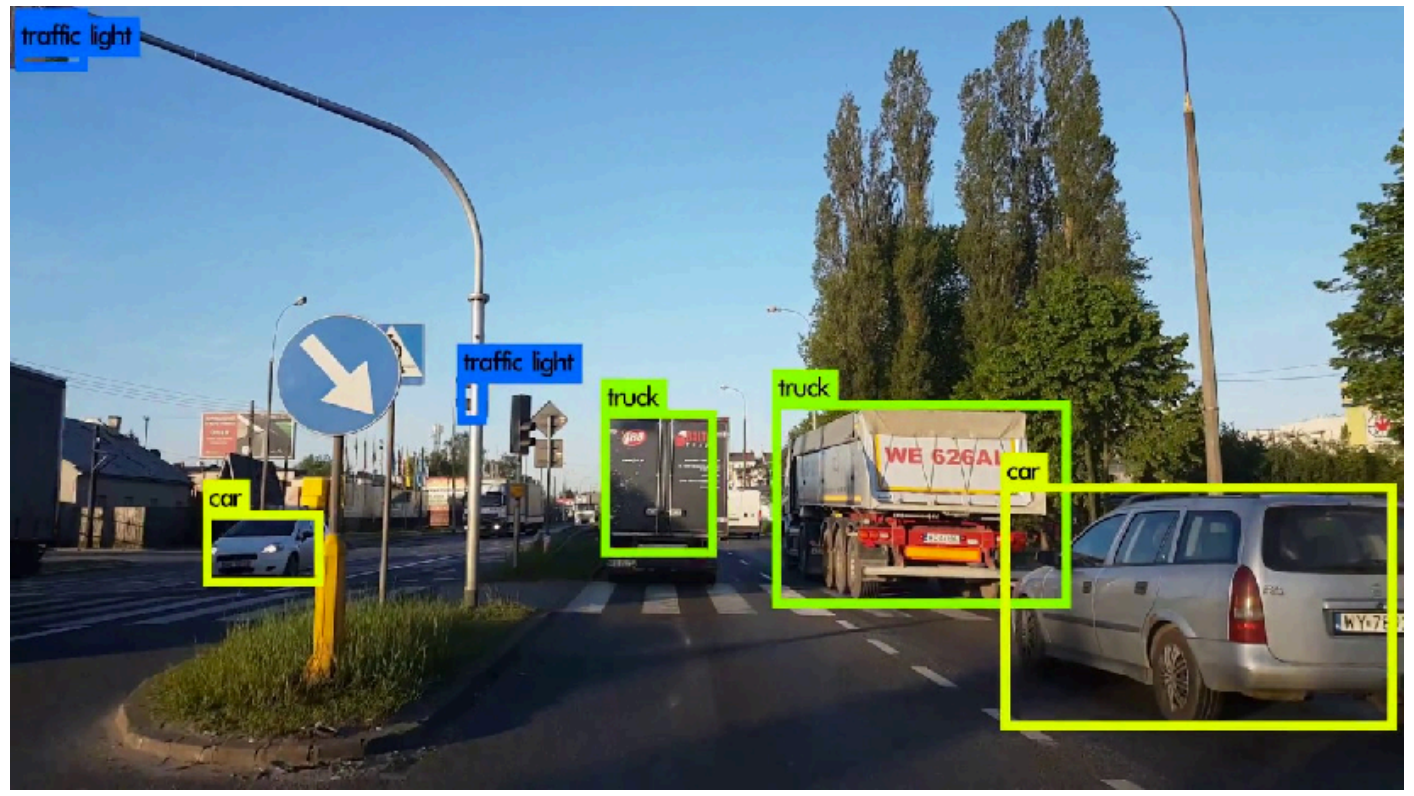
Object Distribution

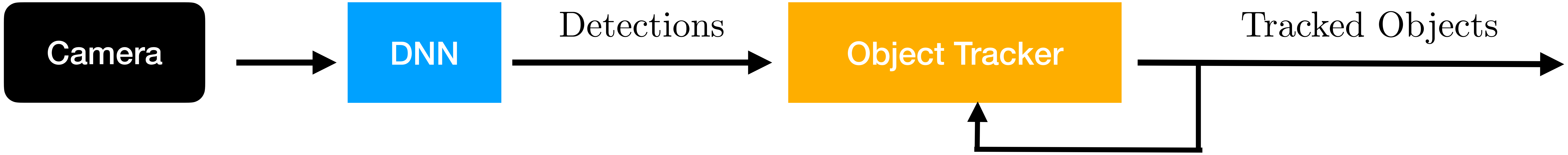
Process Model

Measurement Model

Object Tracking from Cameras

Bounding Box Detections





Each object track has the following state:

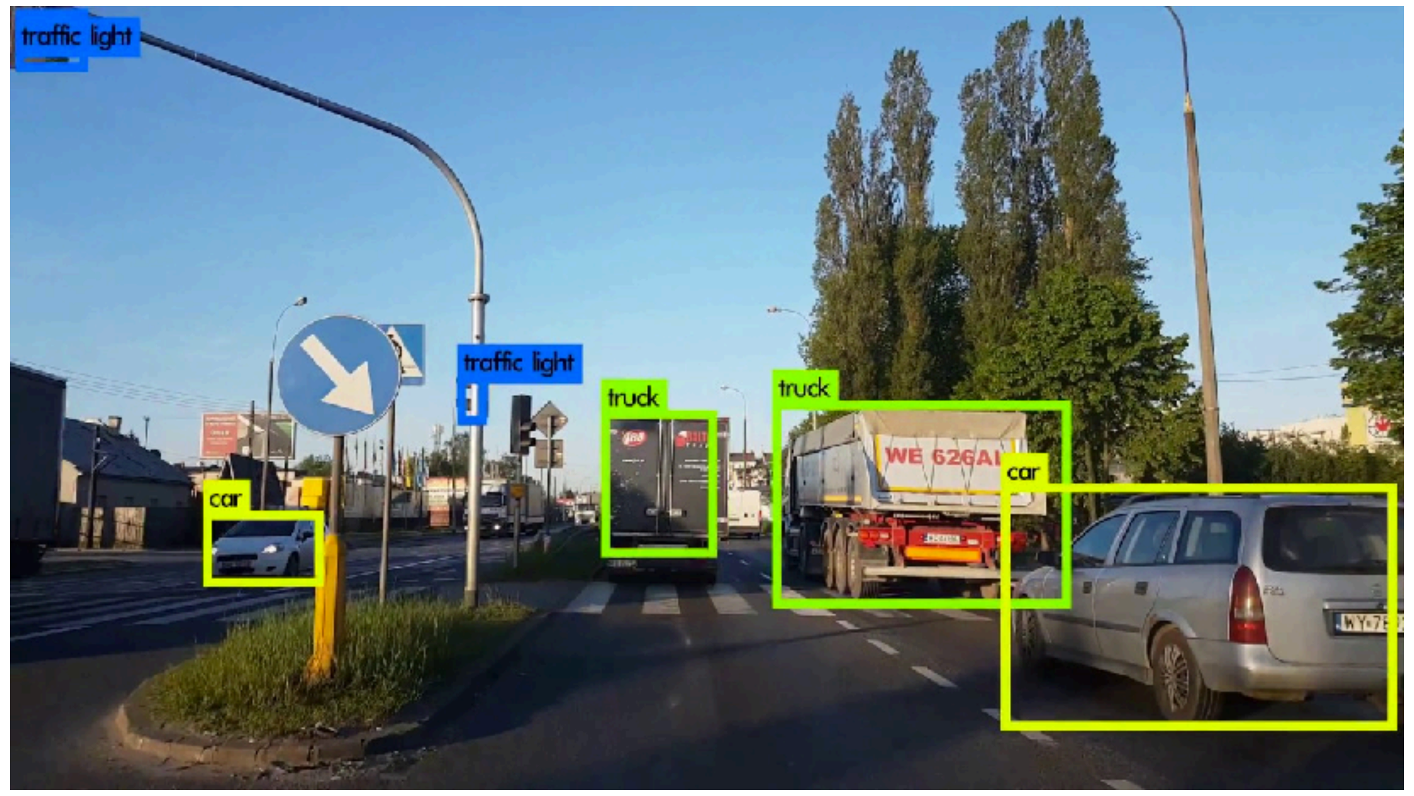
$x = (p_x, p_y, v_x, v_y, w, h)$ OR $\varnothing$

Detections measure:

$z = (p_x, p_y, w, h)$ OR $\varnothing$

Object Tracking from Cameras

Bounding Box Detections



Most important modeling choice in tracker design is measurement model:

$p(\bar{z}_k = z_k \,|\, \bar{x}_k = x_k)$ True Positive

$p(\bar{z}_k = z_k \,|\, \bar{x}_k = \varnothing)$ False Positive

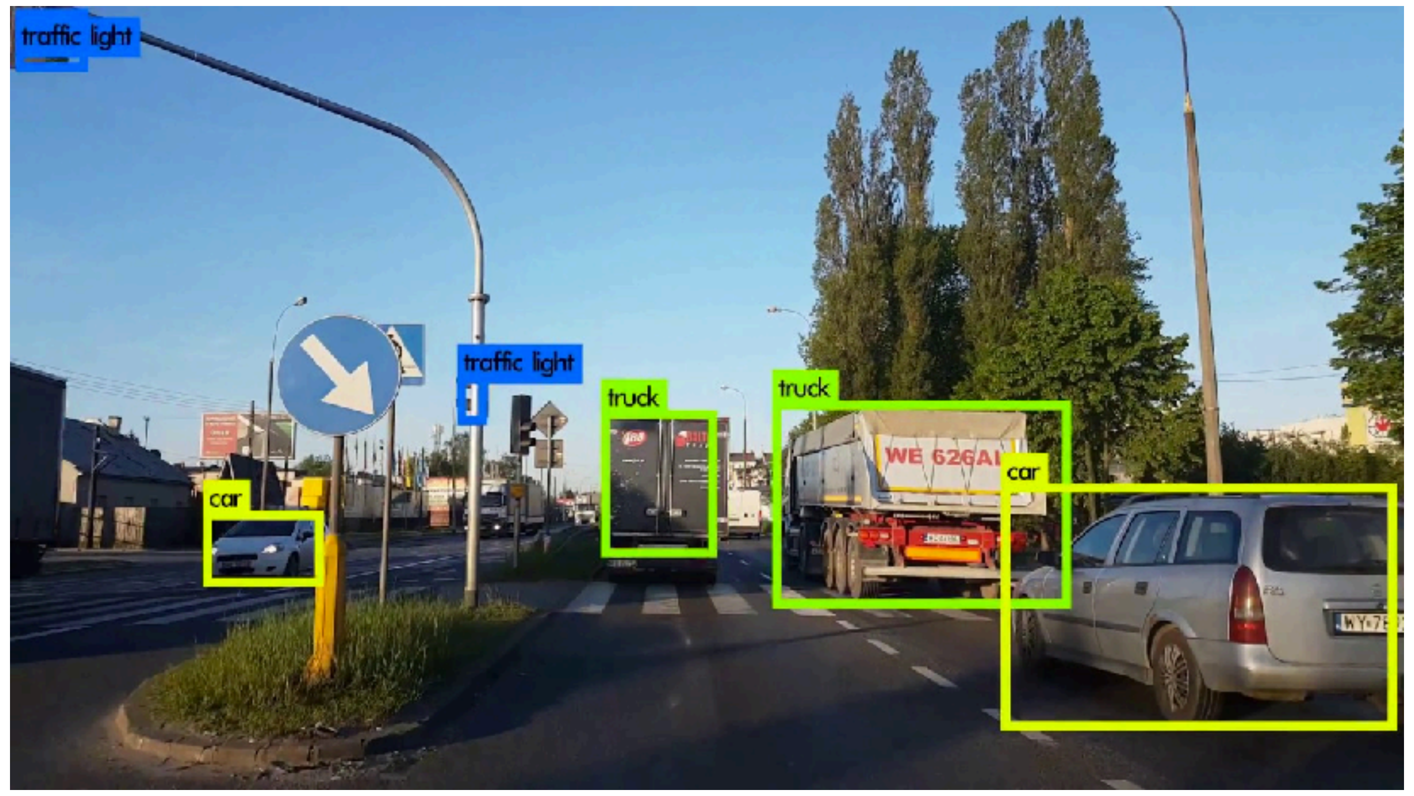
$p(\bar{z}_k = \varnothing \,|\, \bar{x}_k = \varnothing)$ True Negative

$p(\bar{z}_k = \varnothing \,|\, \bar{x}_k = x_k)$ False Negative

How should these be chosen? Especially for DNN measurements? OOD?

Object Tracking from Cameras

Bounding Box Detections



Ideally, for each measurement $\bar{z}_k$, we could estimate the uncertainty internal to the DNN regarding the sensor input:

$$p(\bar{z}_k \,|\, o_k, D)$$

Training Dataset

Camera input

Measurement

This distribution could serve to inform our measurement models used in tracking better

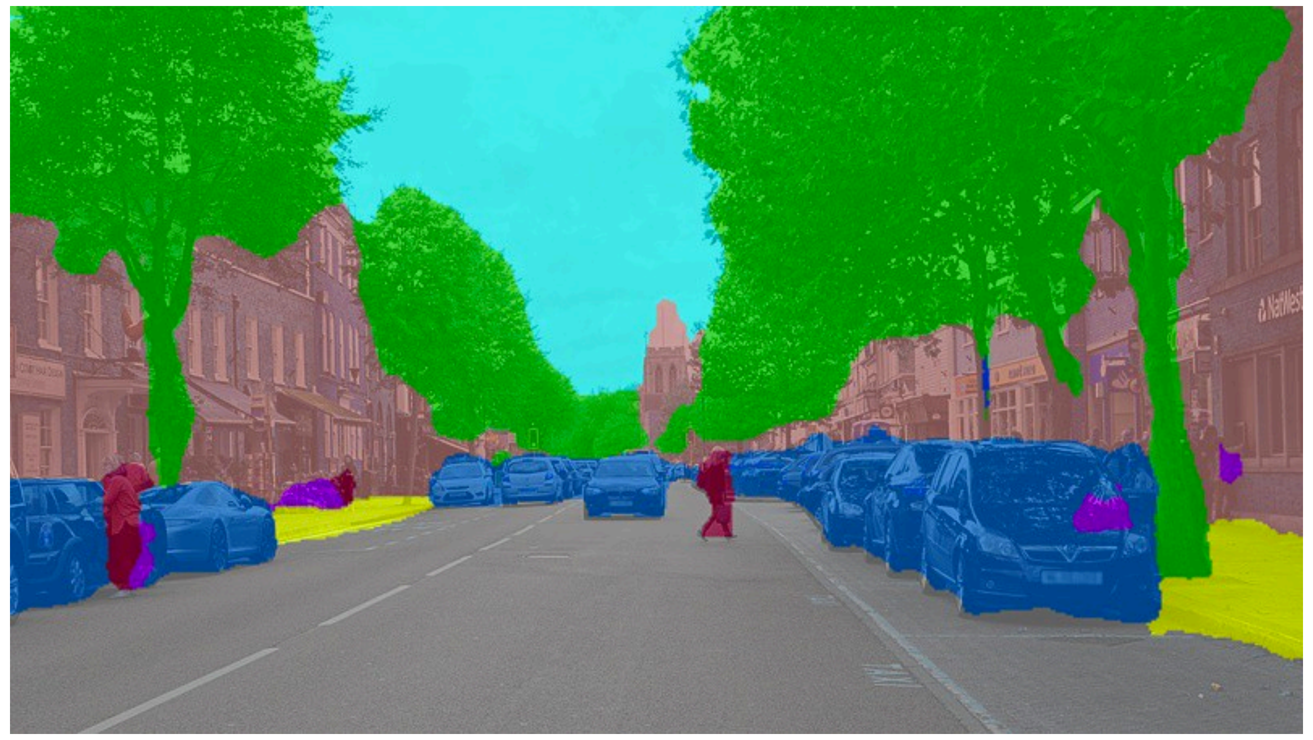
$$p(\bar{z}_k = z_k \,|\, \bar{x}_k = \varnothing)$$

$$p(\bar{z}_k = \varnothing \,|\, \bar{x}_k = x_k)$$

However, for bounding box detections, it is not clear how to extract this distribution.

Object Tracking from Cameras

Pixel-Level Detections

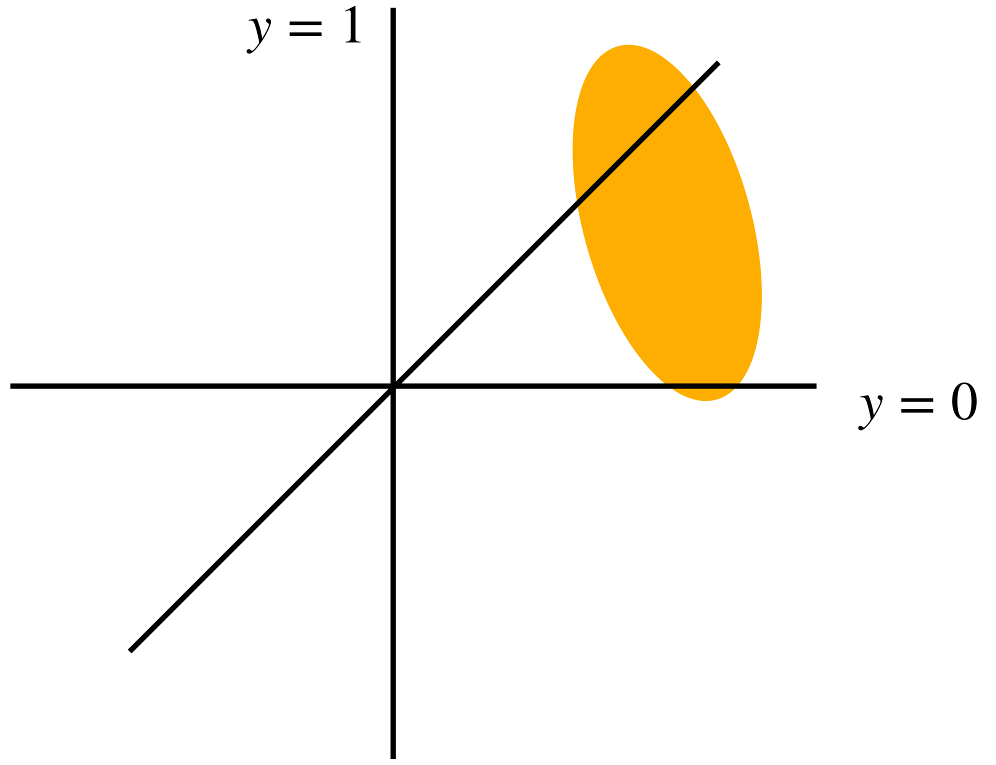


The output at each pixel is a 2D value $y_k^{i,j}$ which is arg-maxed to obtain a detection (e.g. 1=object, 0=not)

Using off-the-shelf variance propagators, we are able to obtain an estimate of $p(y_k^{i,j} \mid o_k, D)$ in the form of a gaussian distribution.

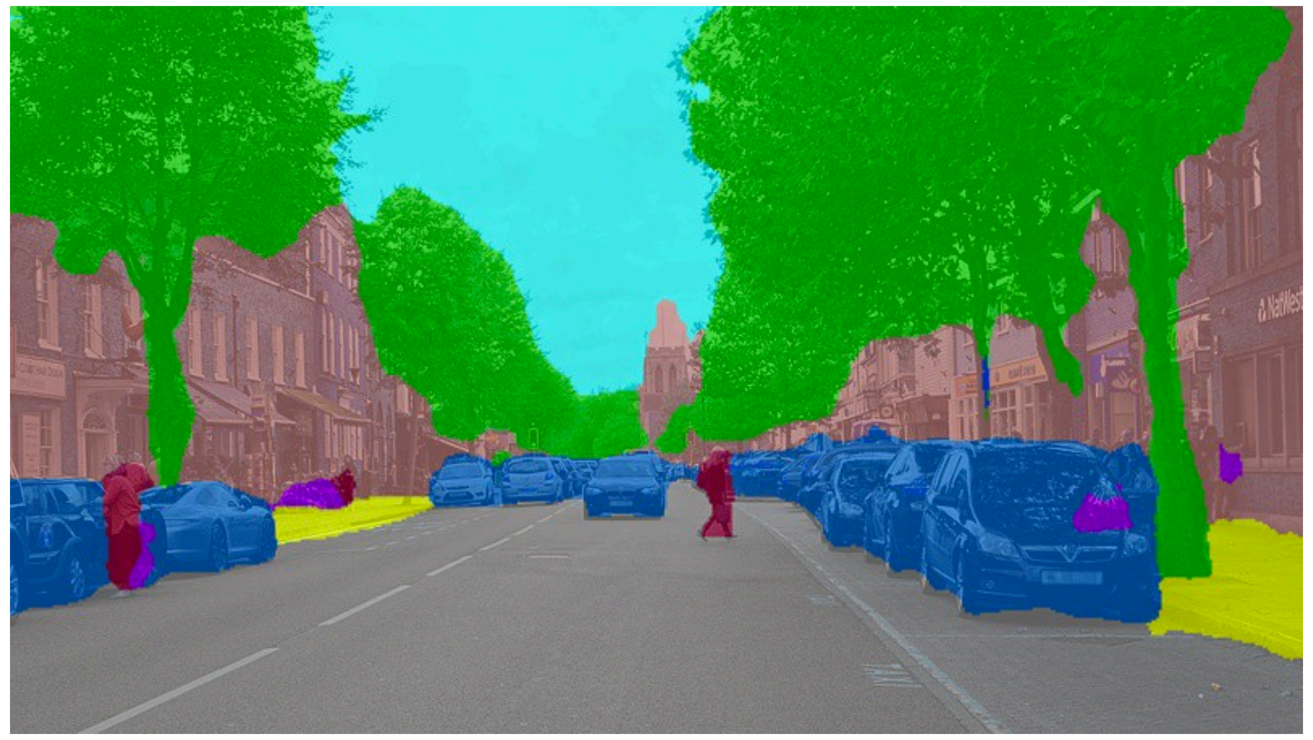
This distribution can then be propagated through the arg-max function (via particles or other) to obtain a probability of occupancy $p(z_k^{i,j} = 1 \mid o_k, D)$.

This distribution can be decomposed into *detection* and *uncertainty*



https://arxiv.org/abs/1908.00598

Object Tracking from Cameras
Pixel-Level Detections



The measurement models used with pixel-level detections (with uncertainties) needs some special care.

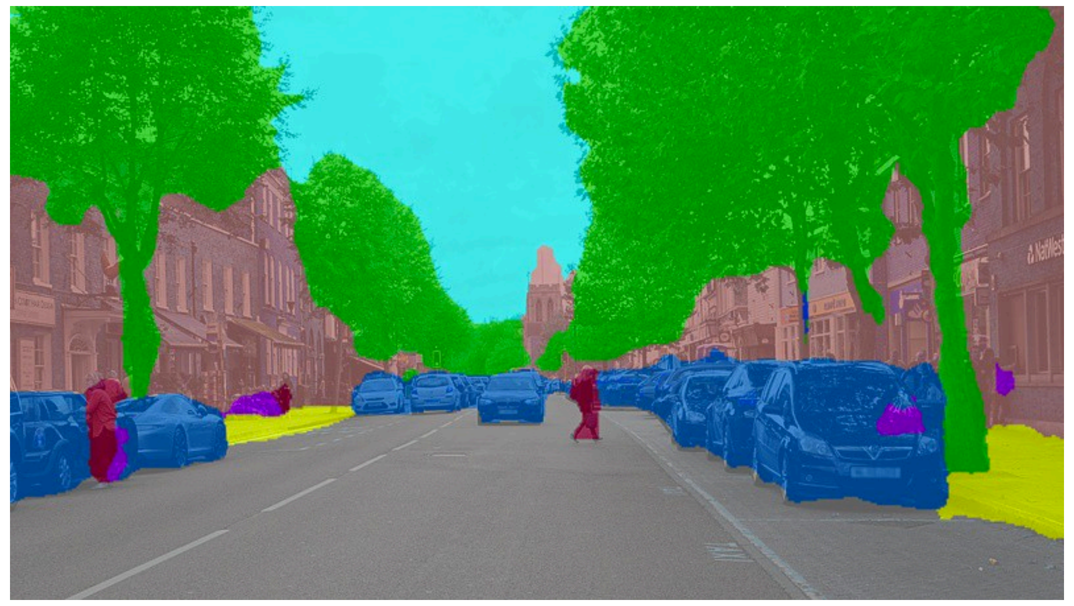
The track representation remains the same as before:
$x = (p_x, p_y, v_x, v_y, w, h)$, as is the form of $p(x_{k+1} | x_k)$.

However we now consider measurements to be *collections of pixel detections*, which are assumed to be independent of other neighboring measurements.

Need to define:

$p(\bar{z}_k = z_k | \bar{x}_k = x_k)$ Probability of True Positive

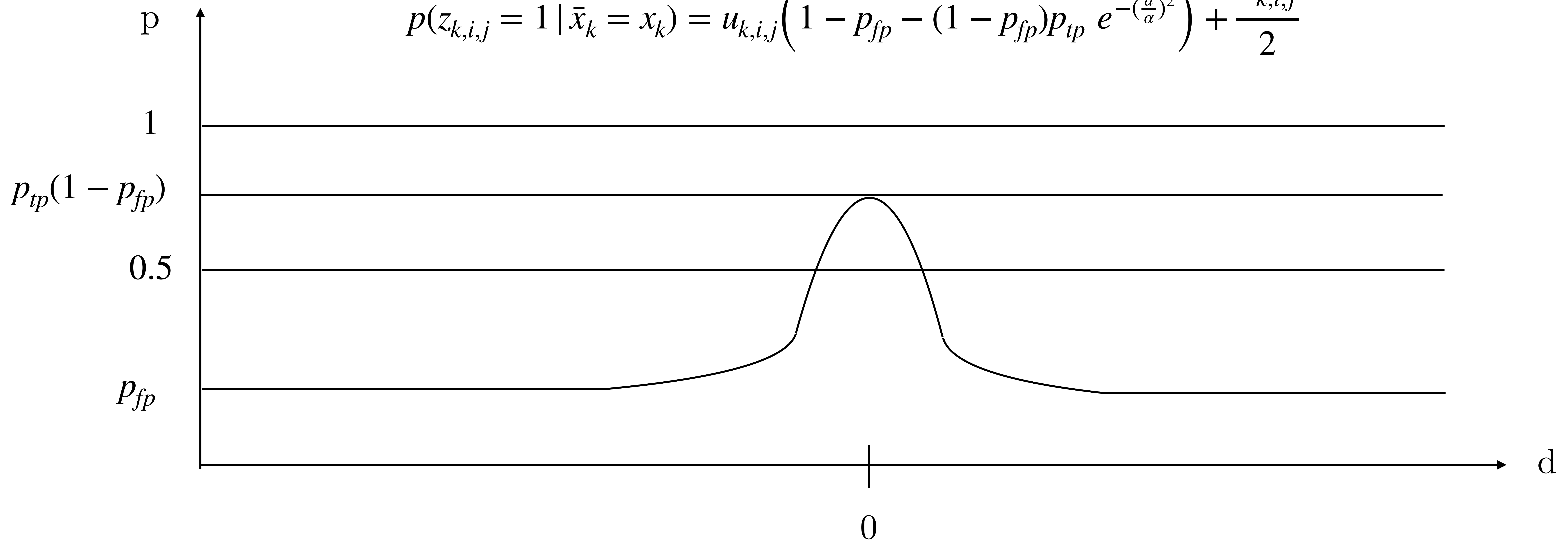
$p(\bar{z}_k = z_k | \bar{x}_k = \emptyset)$ Probability of False Positive



Object Tracking from Cameras

Probability of True Positive

$$p(z_{k,i,j} = 1 \,|\, \bar{x}_k = x_k) = u_{k,i,j}\left(1 - p_{fp} - (1 - p_{fp})p_{tp}\ e^{-(\frac{d}{\alpha})^2}\right) + \frac{u_{k,i,j}}{2}$$



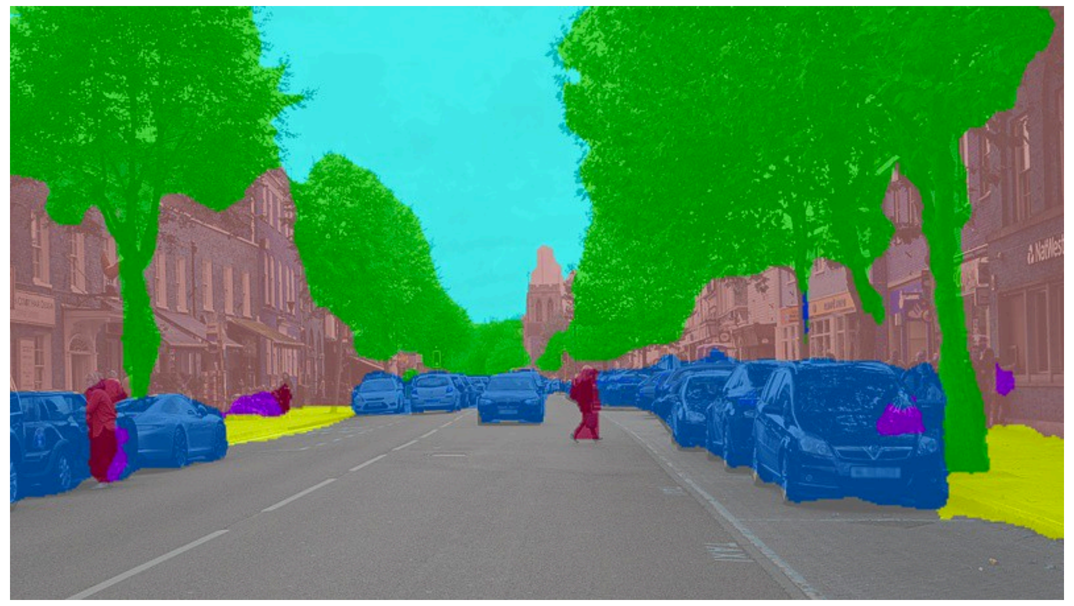
$$d = \min_{a,b} \|a - b\|$$
$$s.t.\ \ a \in B_a$$
$$b \in B_b$$

$B_a :=$ Box defined by track center, width, height

$B_b :=$ Box defined by pixel center, width, height

Object Tracking from Cameras

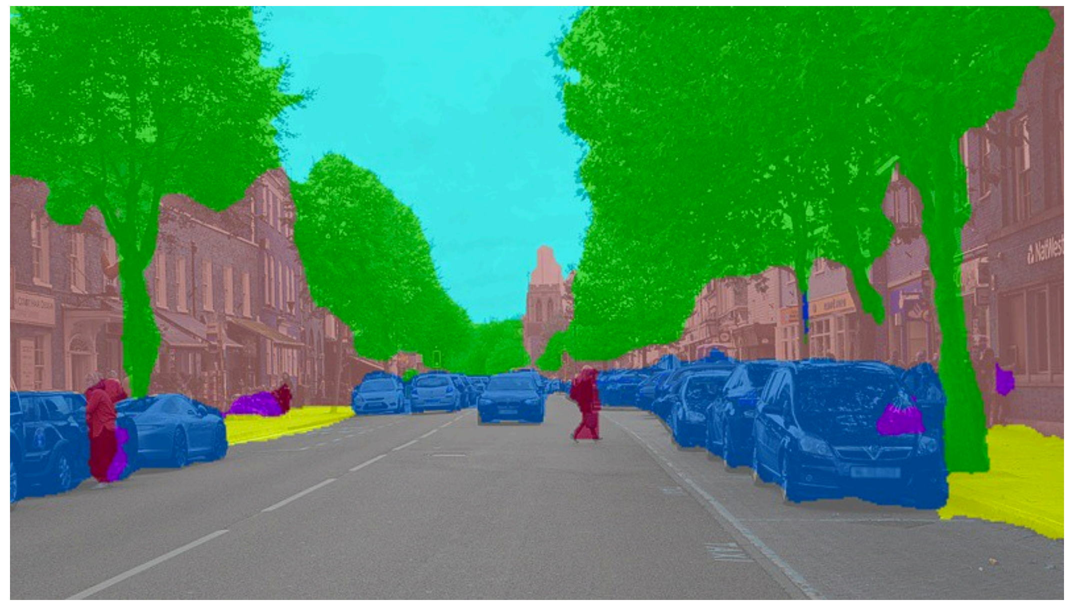
Probability of False Positive



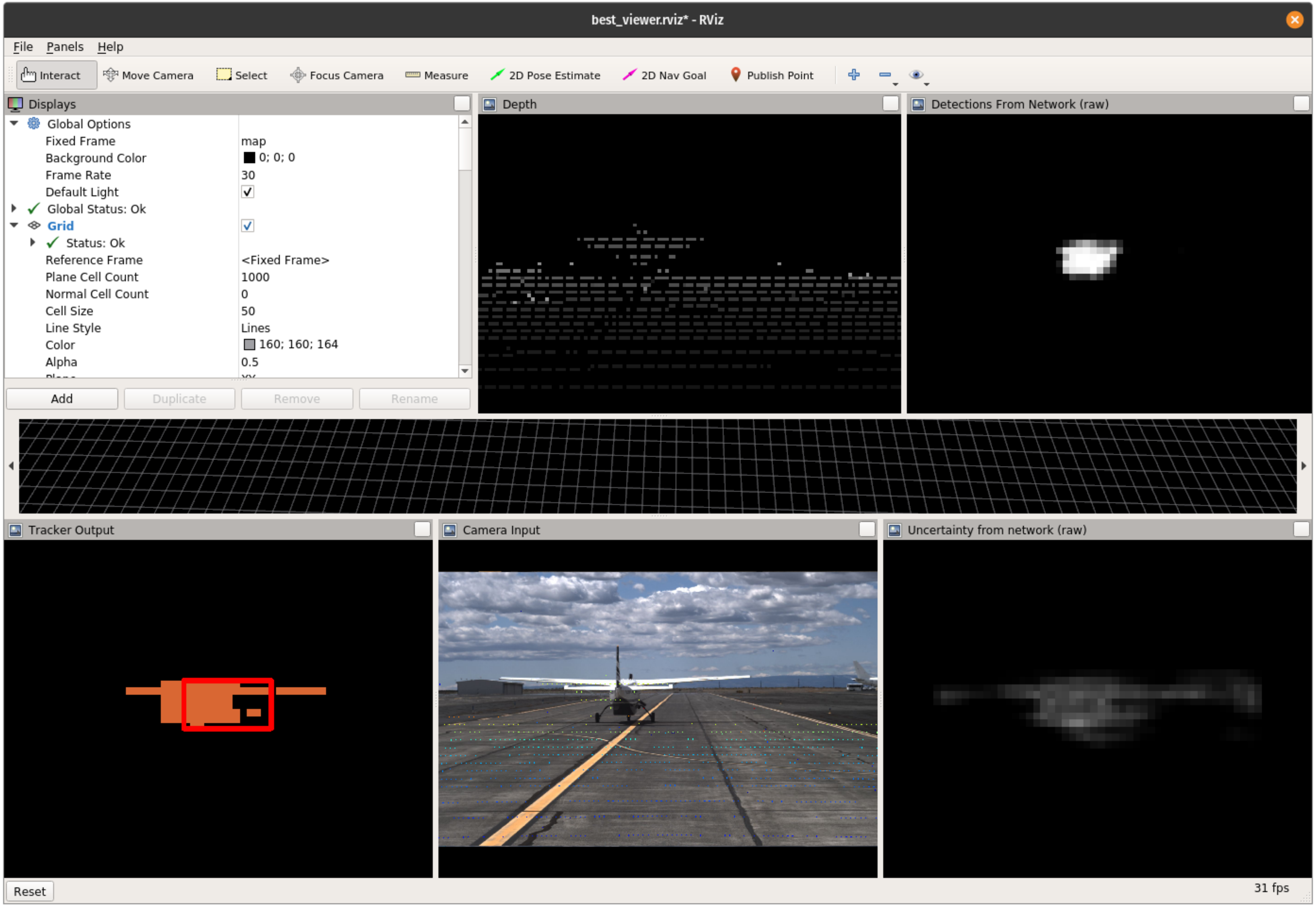
Probability of False Positive: $$p(z_{k,i,j} = 1 \mid \bar{x}_k = \varnothing) = u_{k,i,j}(1 - p_{fp}) + \frac{u_{k,i,j}}{2}$$

Object Tracking from Cameras

Tracking Overview



1. Cluster network output pixels in to groups
2. For each cluster, assign to an existing track (using M-distance) or birth new track
3. For each track, compute prediction distributions
4. For each track, sample particles from predicted distribution
5. For each particle $p_{k,l}$, compute weights $= p(\bar{z}_k \,|\, \bar{x}_k = p_{k,l}) = \Pi_{m=1}^{M} p(\bar{z}_{k,m} \,|\, \bar{x}_k = p_{k,l})$
6. Fit distribution to particles based on weights to normalize. Compute POE for track, and kill if needed


best_viewer.rviz* - RViz
File
Panels
Help
Interact
Move Camera
Select
Focus Camera
Measure
2D Pose Estimate
2D Nav Goal
Publish Point
Displays
Global Options
Fixed Frame
map
Background Color
0; 0; 0
Frame Rate
30
Default Light
Global Status: Ok
Grid
Status: Ok
Reference Frame
<Fixed Frame>
Plane Cell Count
1000
Normal Cell Count
0
Cell Size
50
Line Style
Lines
Color
160; 160; 164
Alpha
0.5
Add
Duplicate
Remove
Rename
Depth
Detections From Network (raw)
Tracker Output
Camera Input
Uncertainty from network (raw)
Reset
31 fps