



07 / Using Detected Objects

Using Detected Objects, An Overview

We want the minimum amount of information needed to produce the correct results:

- Preprocessing for:
 - Classification
 - Further feature extraction
- Input to tracking
- Collision detection

Classification

Object detection only says *where* objects are, not *what* they are

- Detected objects/clusters can be passed to a classifier to obtain a class
- Or combined with some camera-based classifier/detector to produce objects in 3D space with classes
- Knowing the class of an object can inform:
 - Prediction estimates
 - Special forms of planning wrt certain kinds of objects (e.g. cyclists, pedestrians)



Object Detection as Preprocessing

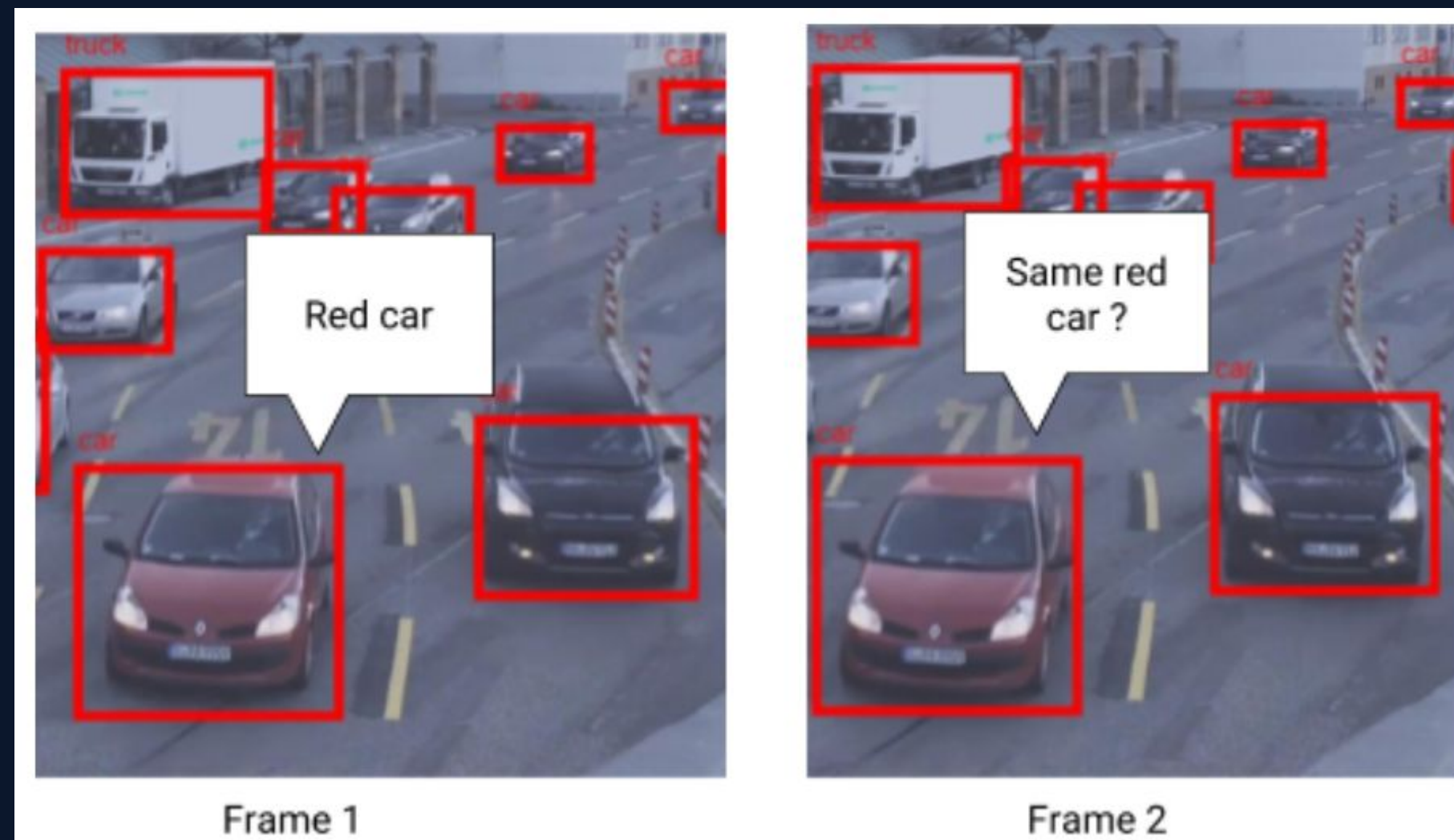
More generally, clusters or objects can be used as regions of interest for further processing:

- Marker detection/classification
- Signage understanding
- Localization landmarks
- etc.

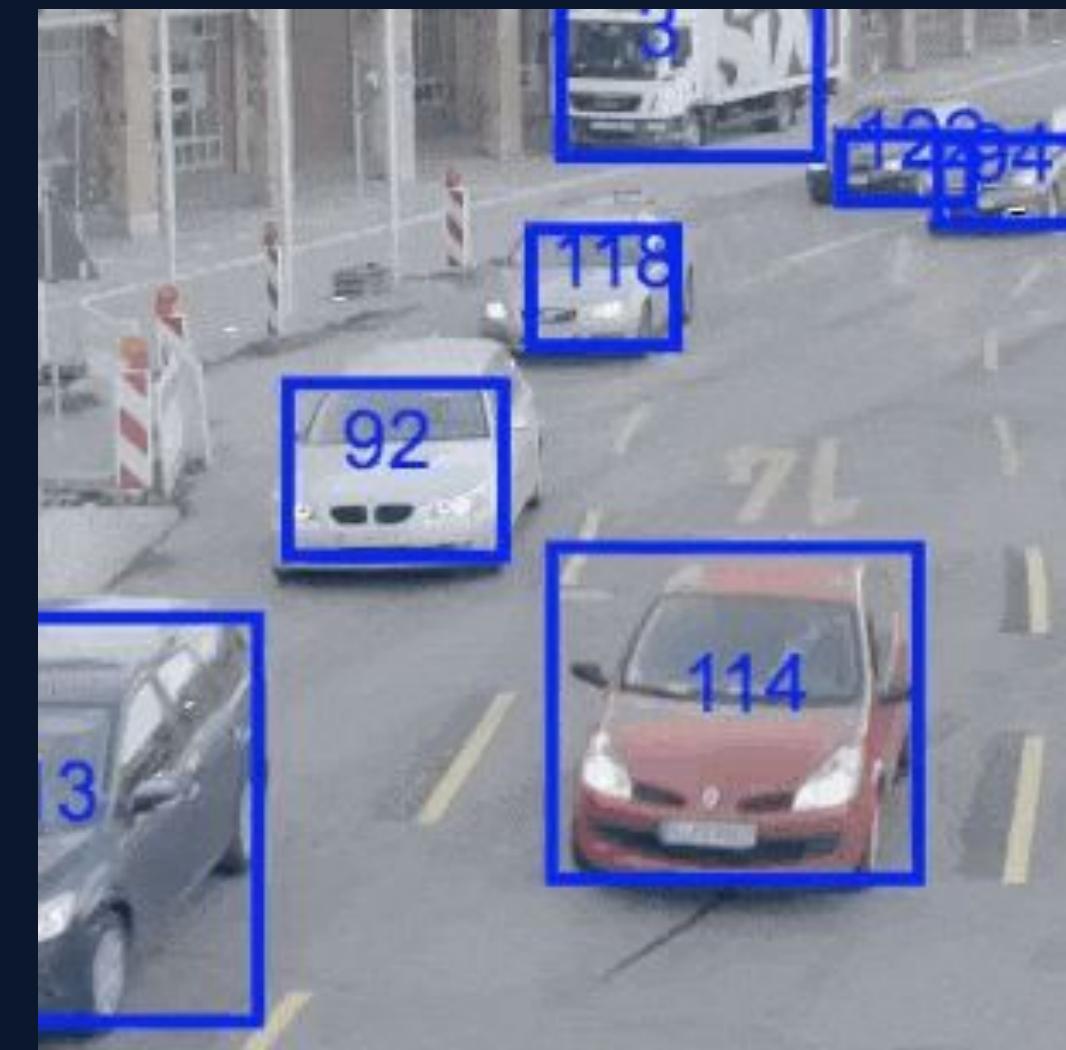


Tracking

- Tracking can be thought of as temporally fusing instantaneous detections over time
- Huge wealth of literature and approaches on this topic, e.g.:
 - Tracking-by-assignment
 - Data association methods (Hungarian algorithm, GNN)
 - State estimation (Kalman Filter, Particle Filter)
 - Combined tracking methods, e.g.:
 - Multiple Hypothesis Tracking (Khan et al, Kim et al)



The problem



Tracking output example

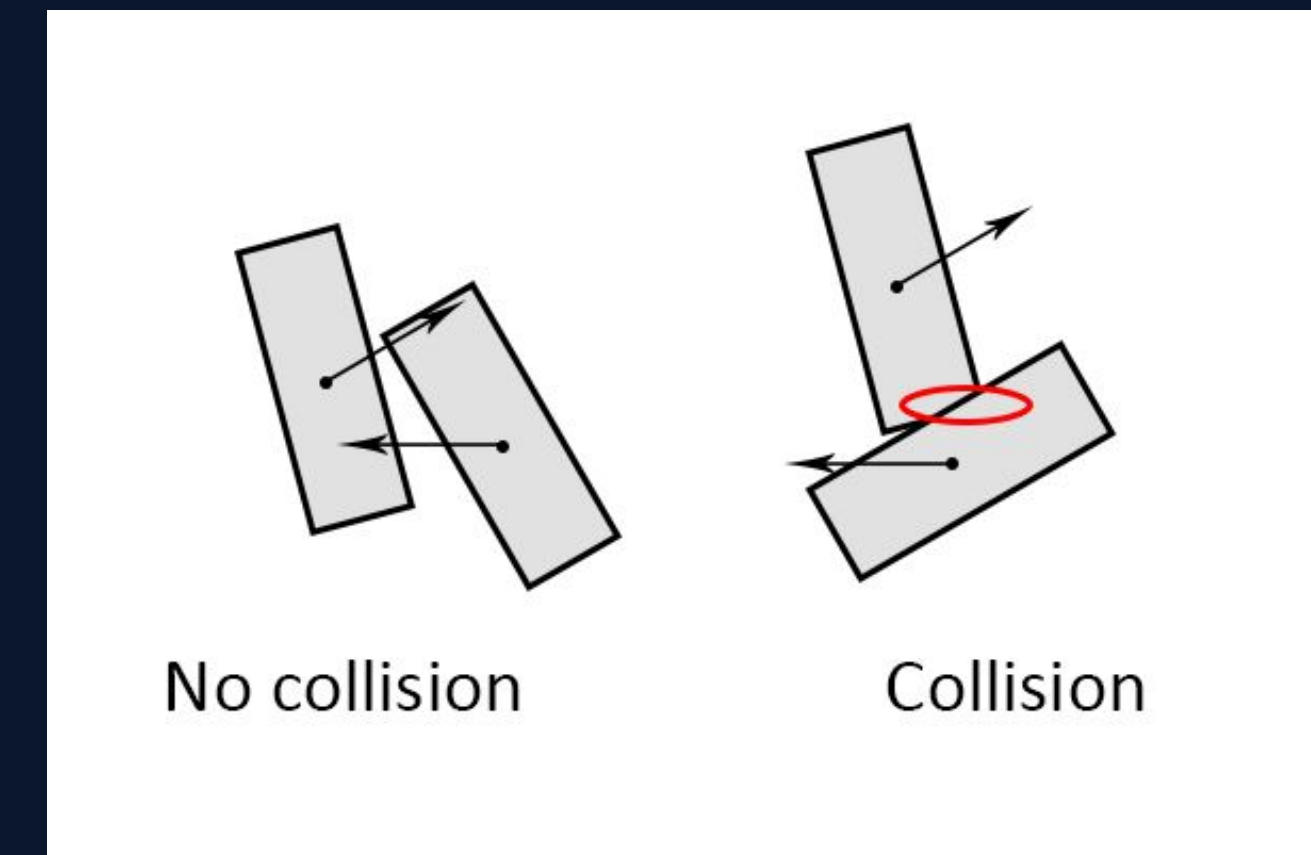
Collision Detection

Algorithms for collision detection:

- SAT
- GJK (+EPA)

Many ways to efficiently use them

- See motion planning literature, e.g.:
 - Mirtich, 1997
 - Lin and Gottschalk
 - etc...



Using Detected Objects

Four main use cases for the result of object detection:

- General preprocessing/region of interest detection
- Classification
- Tracking (fusing detections temporally)
- Collision detection

Object Detection and Autoware.Auto - Summary

Object Detection is a key part of the autonomous driving stack

- Fundamental to the basic use case
- Encompassing the sensing and understanding part of autonomy
- Outputs are used in many places

Autoware.Auto uses a classical object detection stack:

- Simpler and more robust
- Geometric and optimization-based methods for:
 - Ground filtering
 - Clustering
 - Shape Extraction

