Deep Learning Techniques For Autonomous Driving

Mingcan Xiang
Sep. 18, 2021

University of Massachusetts Amherst

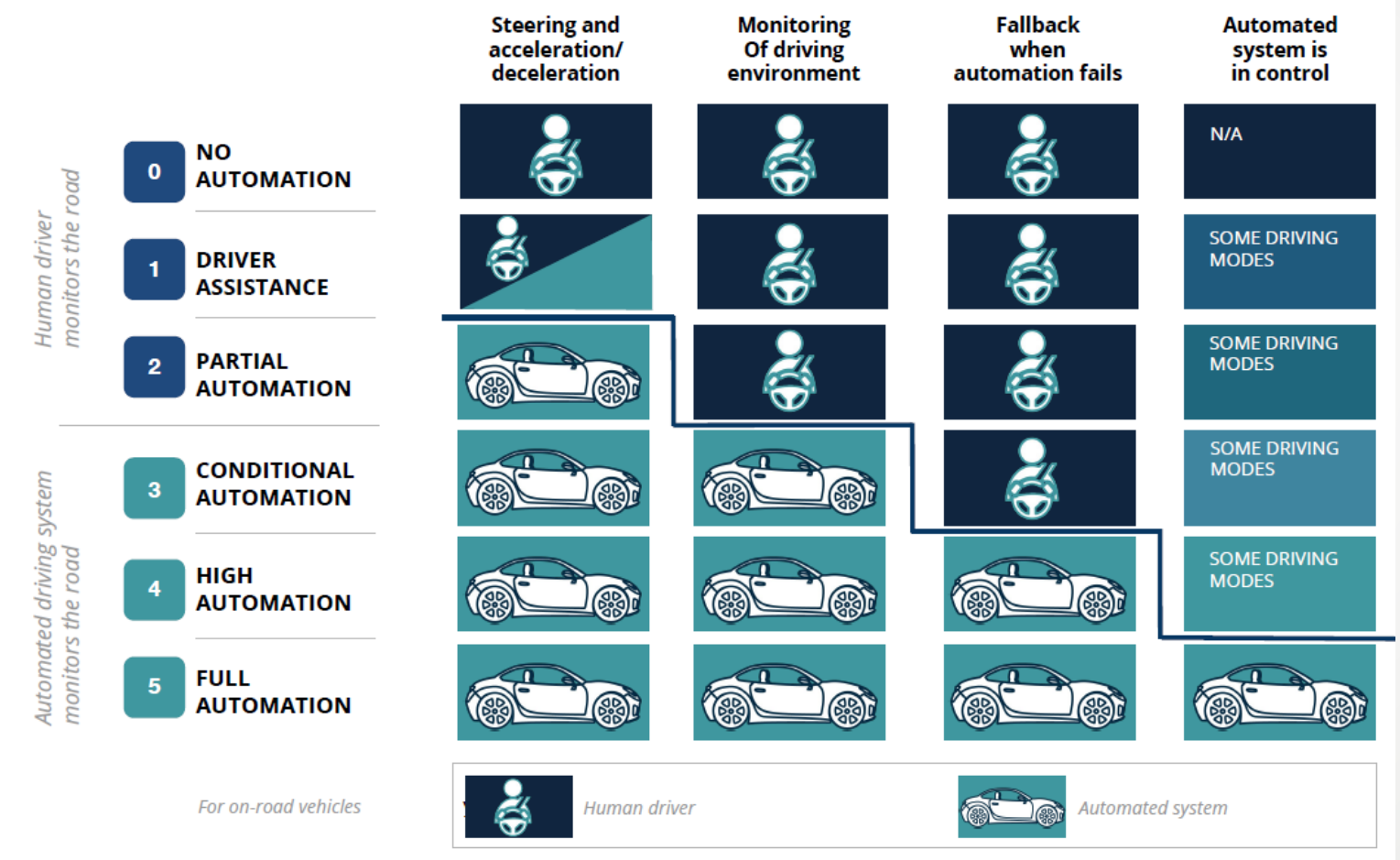
ROADMAP

- **Introduction**
- **Architecture**
- **Four major tasks**
  - ✓ Localization
  - ✓ Perception
  - ✓ Prediction
  - ✓ Planning
- **Datasets**
- **Evaluation**
- **Multi-task opportunities**

INTRODUCTION

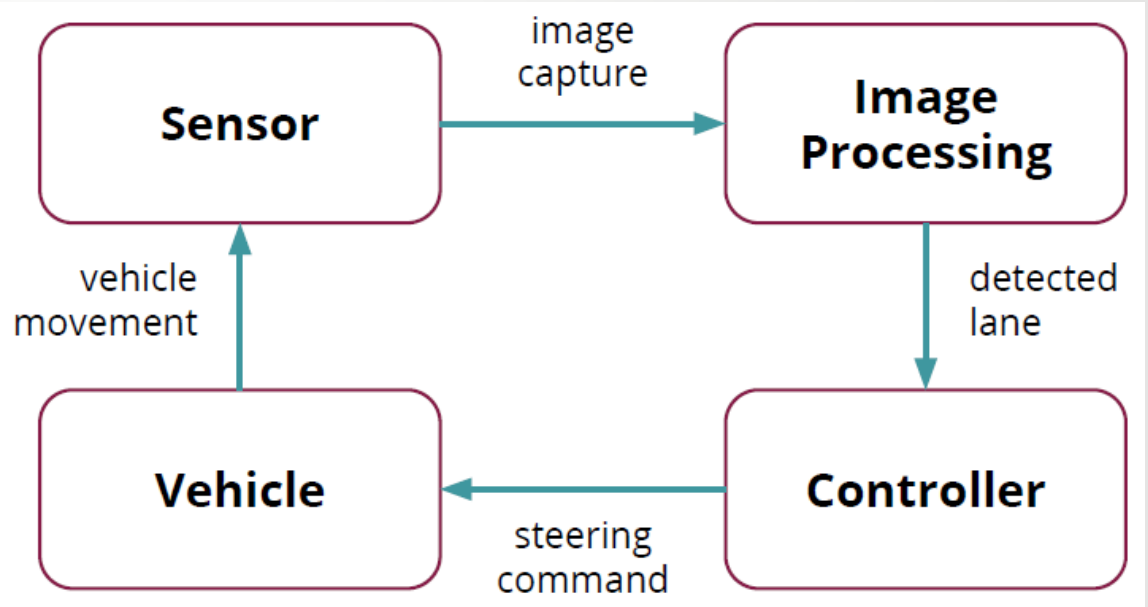
- **Why Autonomous Driving?**
  - Approximately 1.35 million people die in road accidents each year
    - 94% of these are caused due to human error
  - Reduce the traffic congestion
  - Relieve drivers' stress
- **Major Challenges**
  - Process large data
  - Real-time requirement
  - Improve the reliability
    - handle normal and abnormal situations
    - fault detection and diagnosis
    - defense the network attack

INTRODUCTION



ARCHITECTURE

1. Localization — Where I am
2. Perception — What is around me
3. Prediction — Where everything is going
4. Planning — What to do next



LOCALIZATION

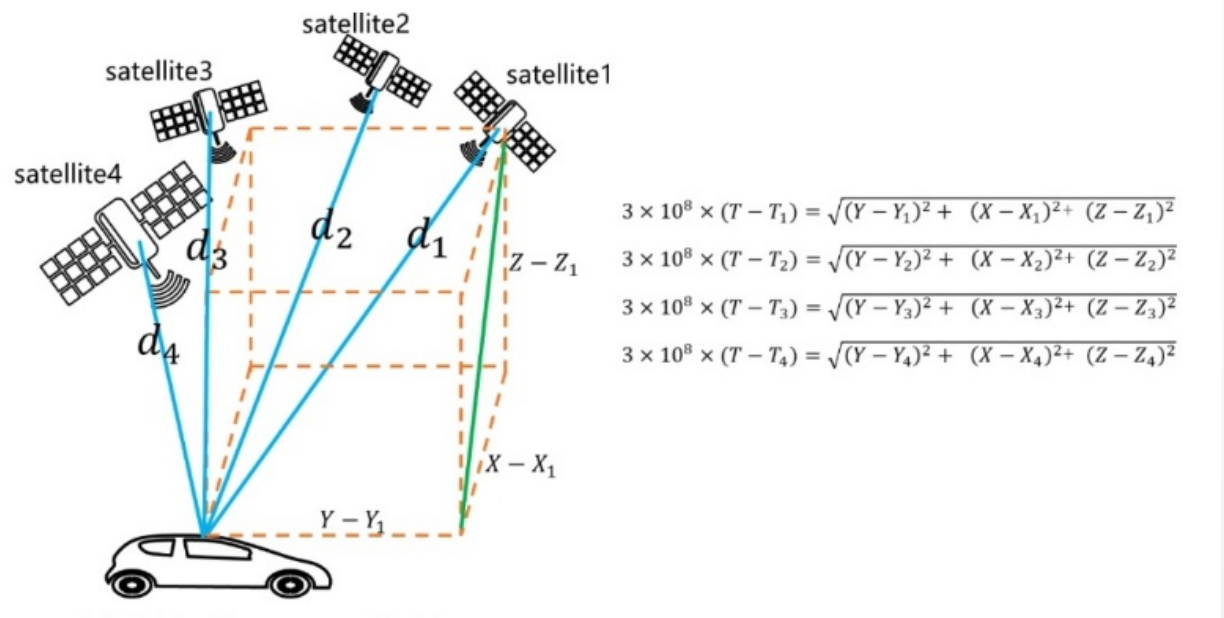
Localization algorithms aim at calculating the position and orientation of the autonomous vehicles.

Currently, there are two kinds of high precision positioning technology used in automatic driving:

- Localization based on electronic signals, such as GNSS (Global Navigation Satellite System)
  - e.g. GPS, GLONASS, Beidou
- Environment feature matching, calculates the current position and direction according to the position and direction of the previous time;
  - Visual Odometry (VO)
  - Simultaneous Localization and Mapping (SLAM)

LOCALIZATION

- **GNSS** is Global navigation satellite system. The principle is easy to understand, which is to determine the position of the ground receiver through four fix-position satellites.



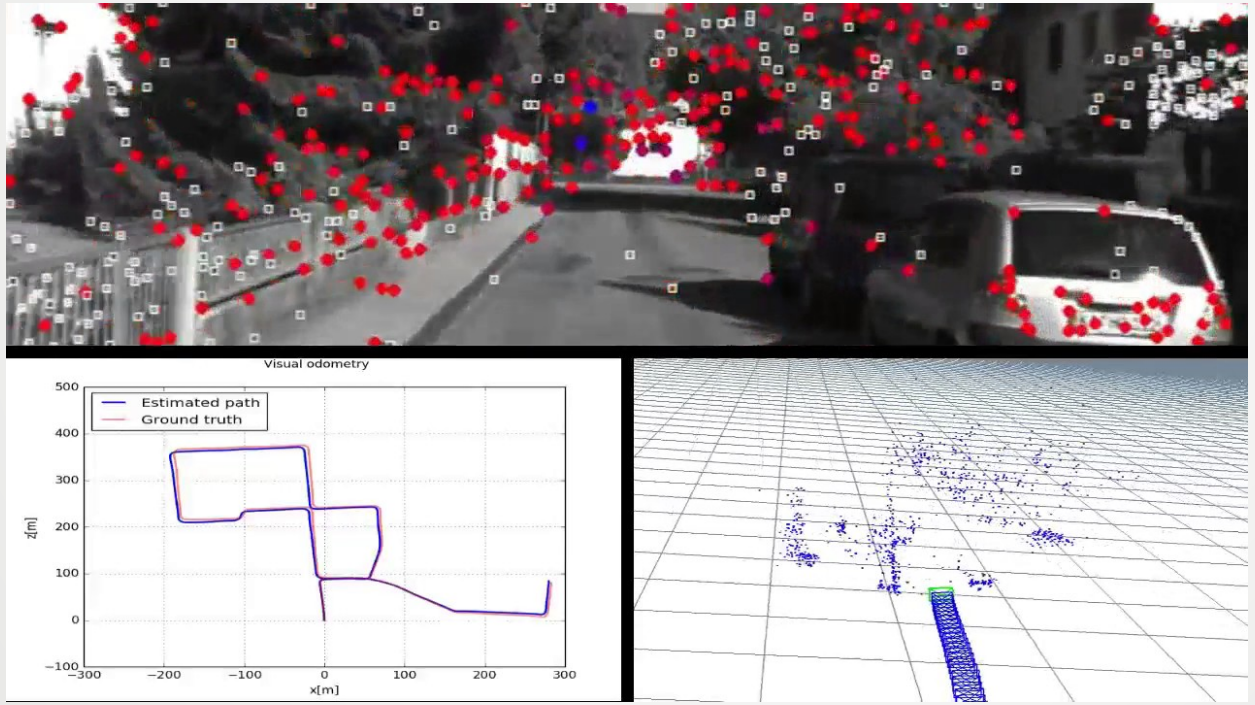
$$3 \times 10^8 \times (T - T_1) = \sqrt{(Y - Y_1)^2 + (X - X_1)^2 + (Z - Z_1)^2}$$
$$3 \times 10^8 \times (T - T_2) = \sqrt{(Y - Y_2)^2 + (X - X_2)^2 + (Z - Z_2)^2}$$
$$3 \times 10^8 \times (T - T_3) = \sqrt{(Y - Y_3)^2 + (X - X_3)^2 + (Z - Z_3)^2}$$
$$3 \times 10^8 \times (T - T_4) = \sqrt{(Y - Y_4)^2 + (X - X_4)^2 + (Z - Z_4)^2}$$

Advantage:

- accuracy,
- redundancy
- availability at all time

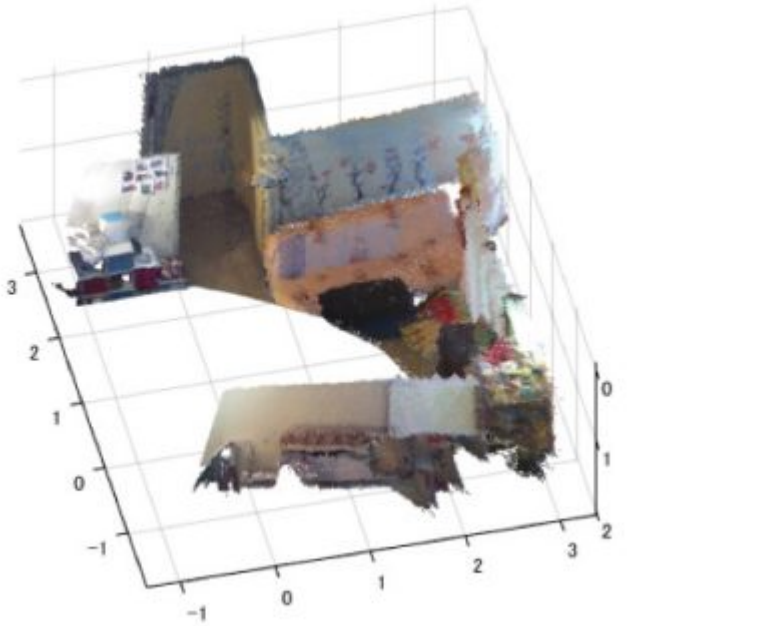
LOCALIZATION

- Visual Localization, also known as **Visual Odometry** (VO), is typically determined by matching keypoint landmarks in consecutive video frames. Given the current frame, these keypoints are used as input to a perspective-n-point mapping algorithm for computing the pose of the vehicle with respect to the previous frame.



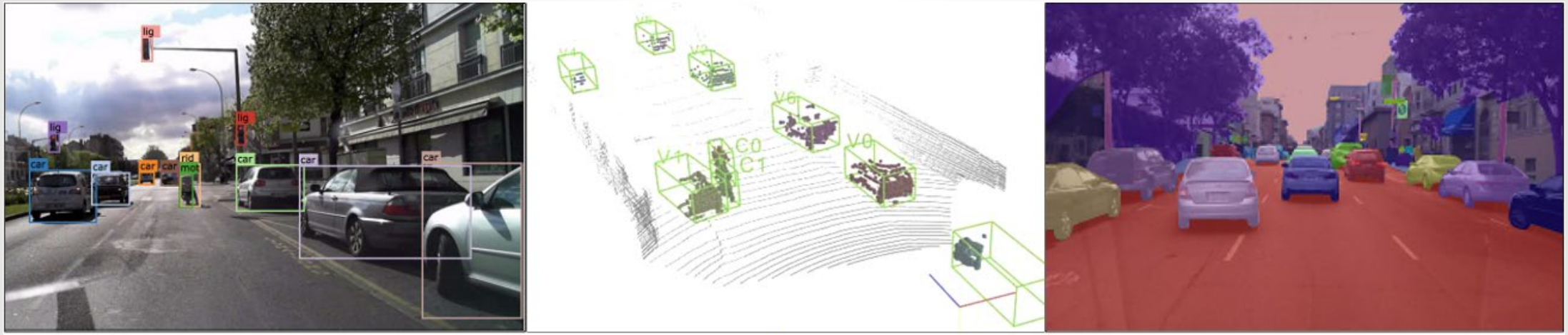
LOCALIZATION

- **SLAM** (Simultaneous Localization and Mapping) algorithms allow the vehicle to map out unknown environments. Engineers use this map information to carry out tasks such as path planning and obstacle avoidance.



PERCEPTION

Autonomous driving technology achieves no manual involvement by perceiving the environment and detecting the response from sensing hardware.



(a) (b) (c)

PERCEPTION

There is a wide range of sensors, including **LiDAR**, **camera**, and **radar**.

- **LiDAR** (Waymo): 3D point clouds representation
- **Camera** (Tesla): Pixel representation – shape & texture

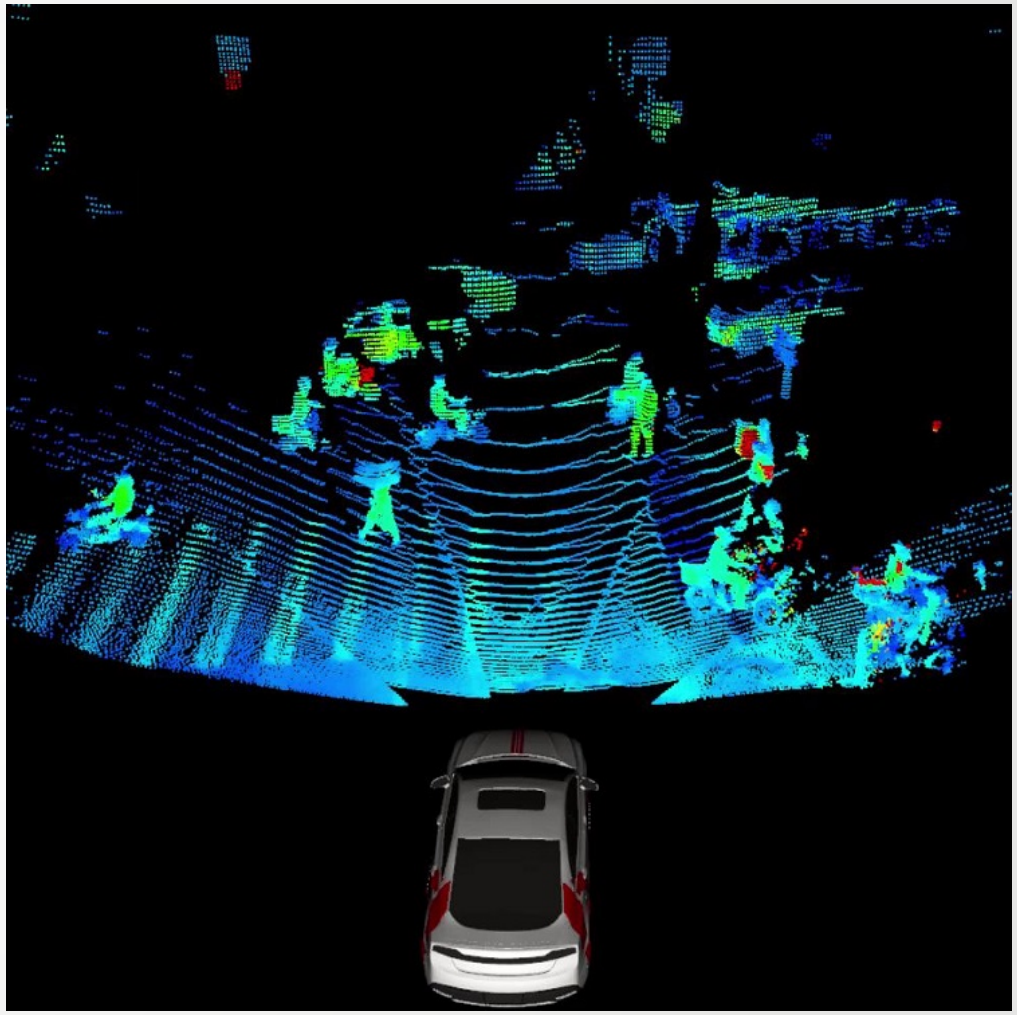
	Advantages	Disadvantages
Camera	Readily available and inexpensive.	Lack depth perception ability, be vulnerable to weather and light condition
LiDAR	360 degree points of view, precise distance measurements, resistant to light condition	much expensive and large equipment, raw point clouds can't provide texture information

PERCEPTION

This is the **LiDAR** 3D point clouds representation

How Does Lidar Work?

A typical lidar sensor emits pulsed light waves into the surrounding environment. These pulses bounce off surrounding objects and return to the sensor. The sensor uses the time it took for each pulse to return to the sensor to calculate the distance it traveled.



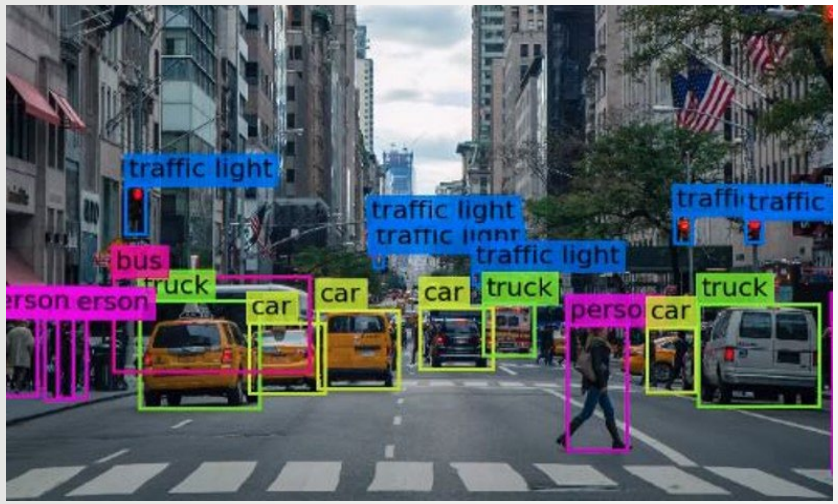
PERCEPTION

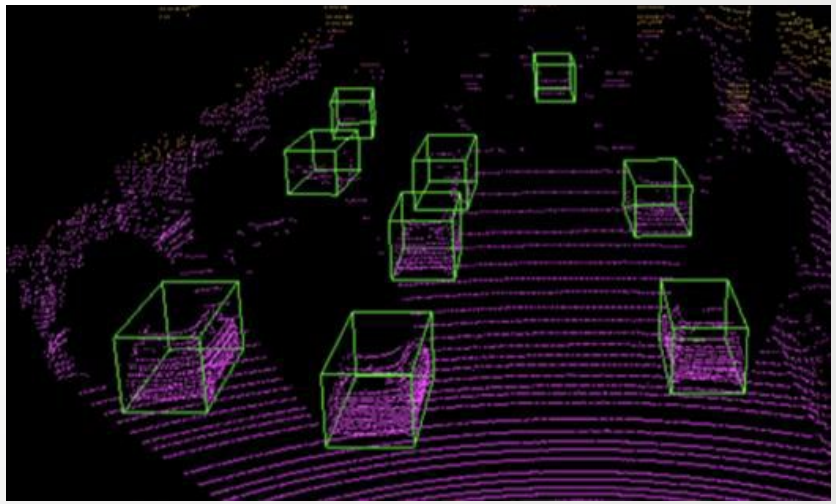
- **Object detection**

Autonomous cars should be able to detect potential obstacles with high accuracy before making any decisions.

Scenarios & Challenges:

- Urban areas: more complex environment and more diverse classes
- Highway: well structured lanes with vehicles following a standard orientation
- Occlusion : partial or complete invisibility of the object





PERCEPTION

- **Object detection**

Despite the aforementioned challenges, the performance of object detection methods for autonomous driving has greatly improved, leveraging deep learning-based perception, in particular Convolutional Neural Networks (CNNs).

Single-stage algorithms:

- **faster speed, less accuracy**
- e.g. Yolo, SSD, CornerNet, RefineNet

Two-stage algorithm:

- **higher accuracy, less speed**
- e.g. RCNN, Faster-RCNN, and R-FCN

In the two-stage method, firstly, a set of **sparse candidate bounding box** are generated, and then they are further classified and regressed to get the **precious bounding box**. The two-stage approach has performed best on several worldwide challenges or competition, including Pascal VOC and MS COCO.

PERCEPTION

- **Semantic Segmentation**

Semantic segmentation is carried out at the pixel level, which needs to classify all pixel points in the image and assign pixels of the same kind to the same label.

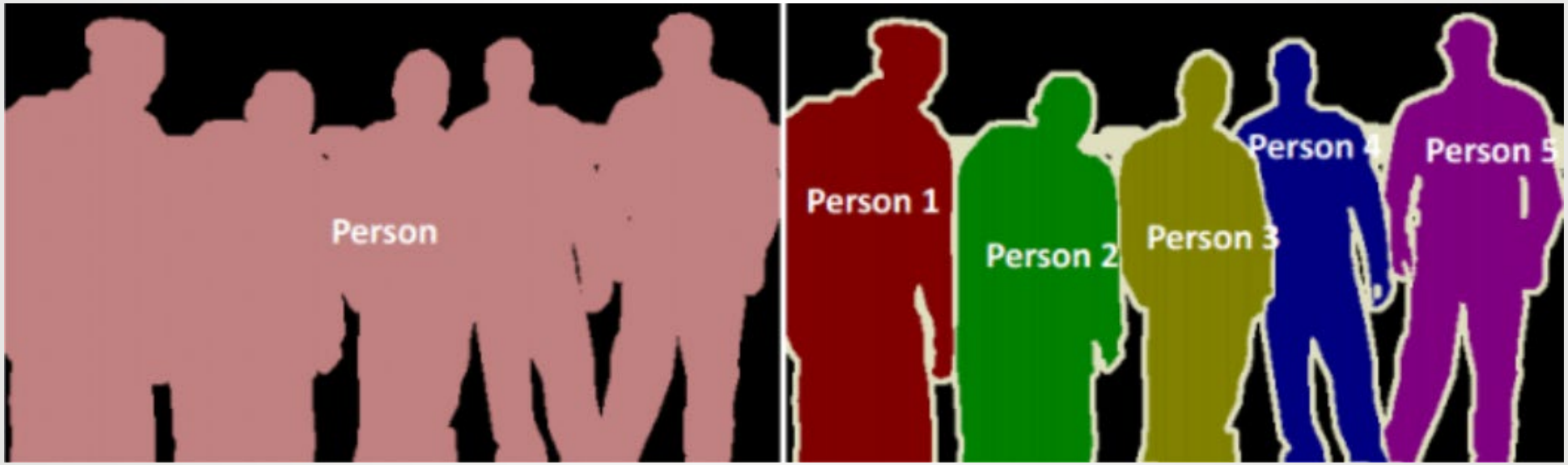
In the autonomous driving scenario, pixels can be marked with categorical labels representing:

- Buildings,
- Pedestrians,
- Drivable area,
- Traffic participants

PERCEPTION

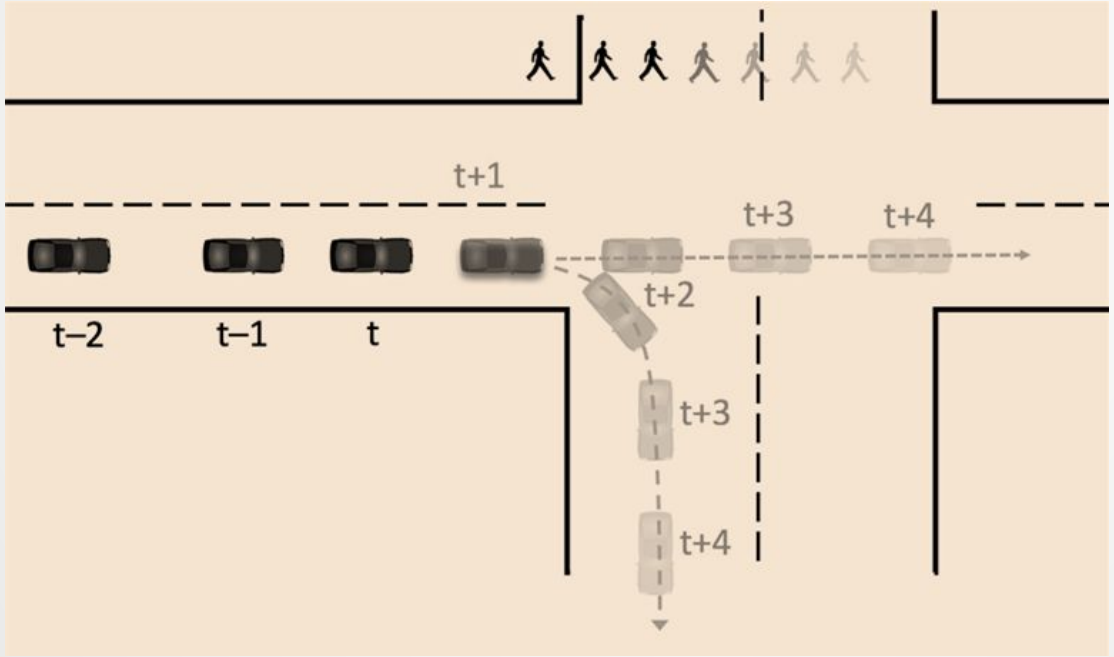
- **Instance Segmentation**

  Instance segmentation is one of the most difficult tasks among the four classic visual tasks. Compared with semantic segmentation, instance segmentation not only needs to classify all the pixels in the image but also needs to distinguish different individuals in the same category



PREDICTION

The Prediction (Motion Prediction) module mainly solves the problem of collaborative interaction between autonomous vehicles and other moving objects (vehicles, pedestrians, etc.) in the surrounding environment.

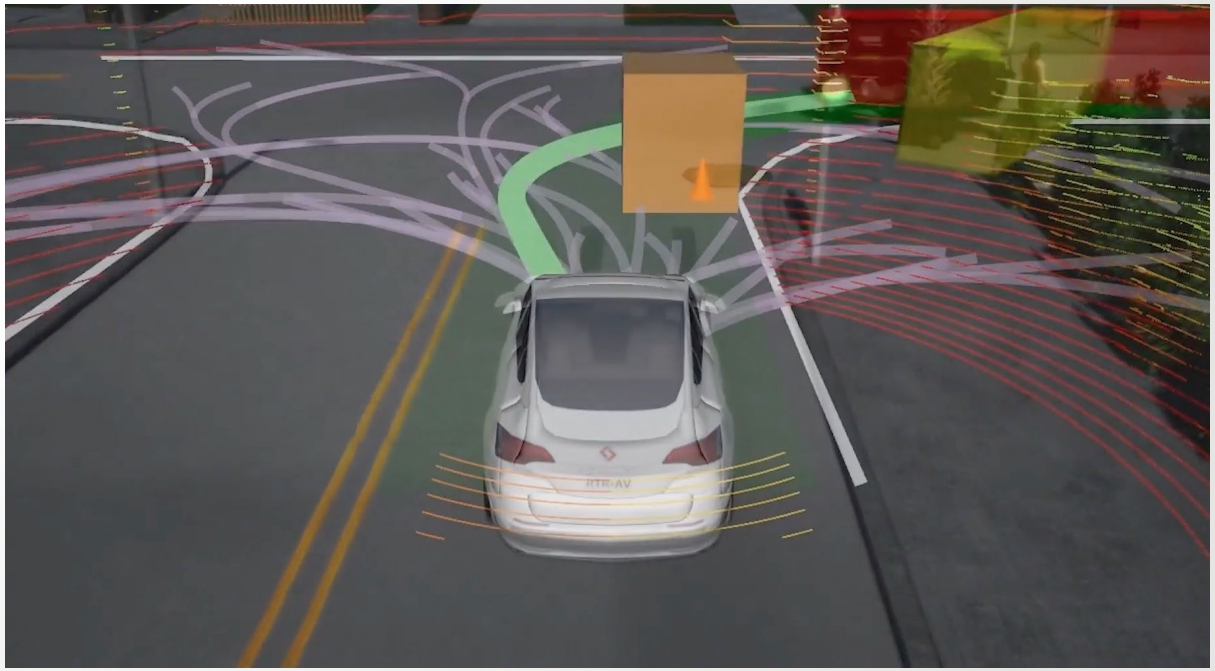


PLANNING

Given a paired points, such as start and terminate points, path planning is to find the best route for the vehicle by considering the vehicle motion model, the rules that the vehicle should follow, the constraints of traffic environment, static obstacles and dynamic obstacles.

Deep learning algorithms:

- Imitation learning
- Deep reinforcement learning



PLANNING

Imitation Learning
The goal in Imitation Learning is to learn the behavior of a human driver from recorded driving experiences. The strategy implies a vehicle teaching process from human demonstration. Thus, we can employ CNNs to learn planning from imitation.

Deep reinforcement learning
DRL for path planning deals mainly with learning driving trajectories in a simulator. The real environmental model is abstracted and transformed into a virtual environment, based on a transfer model.

PLANNING

Imitation Learning

Pros:
It can be trained with data collected from the real-world

Cons:
This data is scarce on corner cases (e.g. driving off-lanes, vehicle crashes, etc.), making the trained network's response uncertain when confronted with unseen data.

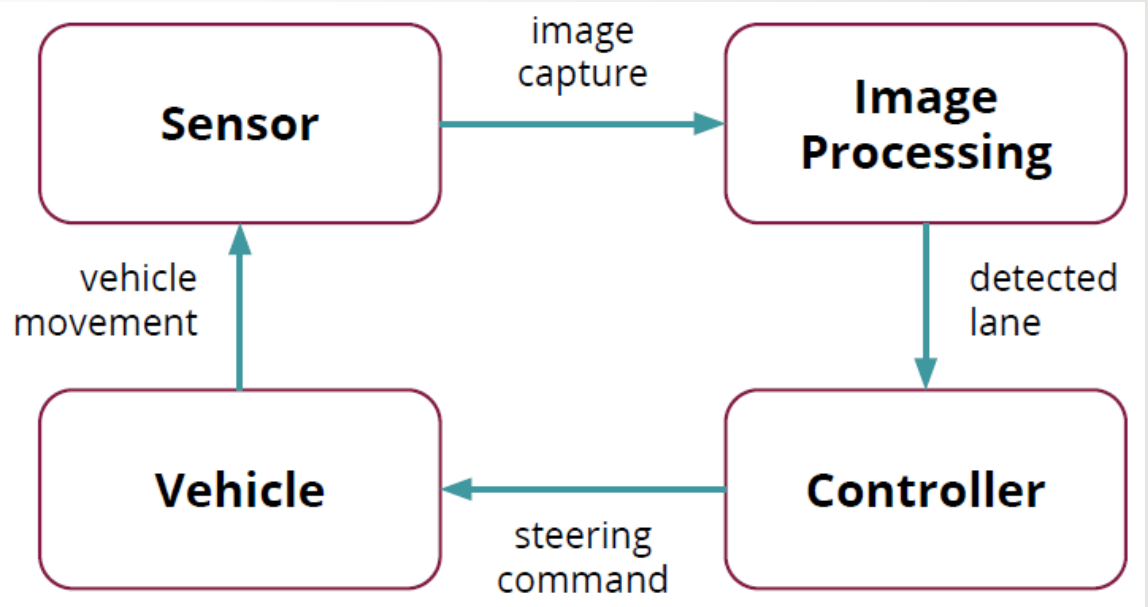
Deep reinforcement learning

Pros:
DRL systems are able to explore different driving situations within a simulated world

Cons:
These models tend to have a biased behavior when ported to the real-world.

CONTROL CYCLE

1. Localization — Where I am
2. Perception — What is around me
3. Prediction — Where everything is going
4. Planning — What to do next



DATASETS

- **The Oxford**
- **Udacity dataset**
- **H3d dataset**
- **Cityscapes**
- **KITTI dataset**
- **NuScenes dataset**

• The Oxford dataset.
Provided by Oxford University, UK. The dataset collection spanned over 1 year, resulting in over 1000 km of recorded driving with almost 20 million images collected from 6 cameras mounted to the vehicle, along with LIDAR, GPS and INS ground truth. Data were collected in all weather conditions, including heavy rain, night, direct sunlight, and snow. Since the vehicle frequently drove the same route over years, it's capable for researchers to investigate long-term localization and mapping for autonomous vehicles in real-world, dynamic urban environments, bounding boxes.

• Udacity dataset.
The vehicle sensor setup contains monocular color cameras, GPS and IMU sensors, as well as a Velodyne 3D Lidar. The data is labeled and the user is provided with the corresponding steering angle that was recorded during the test runs by the human driver. traffic scenes.

• H3d dataset.
Honda Research Institute released its driverless direction data set in March 2019. This data set uses a large full surround 3D multi-target detection and tracking data set collected by 3D lidar scanner. It contains 160 crowded and highly interactive traffic scenes.

DATASETS

- **The Oxford**
- **Udacity dataset**
- **H3d dataset**
- **Cityscapes**
- **KITTI dataset**
- **NuScenes dataset**

• Cityscapes dataset.
Provided from Germany. It focuses on the semantic understanding of urban street scenes. The diversity of the images is very large: 50 cities, different seasons (spring, summer, fall), various weather conditions, and different scene dynamics.
There are 5000 images with fine annotations and 20000 images with coarse annotations.

• KITTI Vision Benchmark dataset (KITTI).
Provided by the Karlsruhe Institute of Technology (KIT) from Germany, this dataset fits the challenges of benchmarking stereo-vision, optical flow, 3D tracking, 3D object detection, or SLAM algorithms. It is known as the most prestigious dataset in the domain of self-driving vehicles.

• NuScenes dataset.
Constructed by nuTonomy, this dataset contains 1000 driving scenes collected from Boston and Singapore, two cities known for their dense traffic and highly challenging driving situations. In order to facilitate common computer vision tasks, such as object detection and tracking, the providers annotated 25 object classes with accurate 3D

EVALUATION

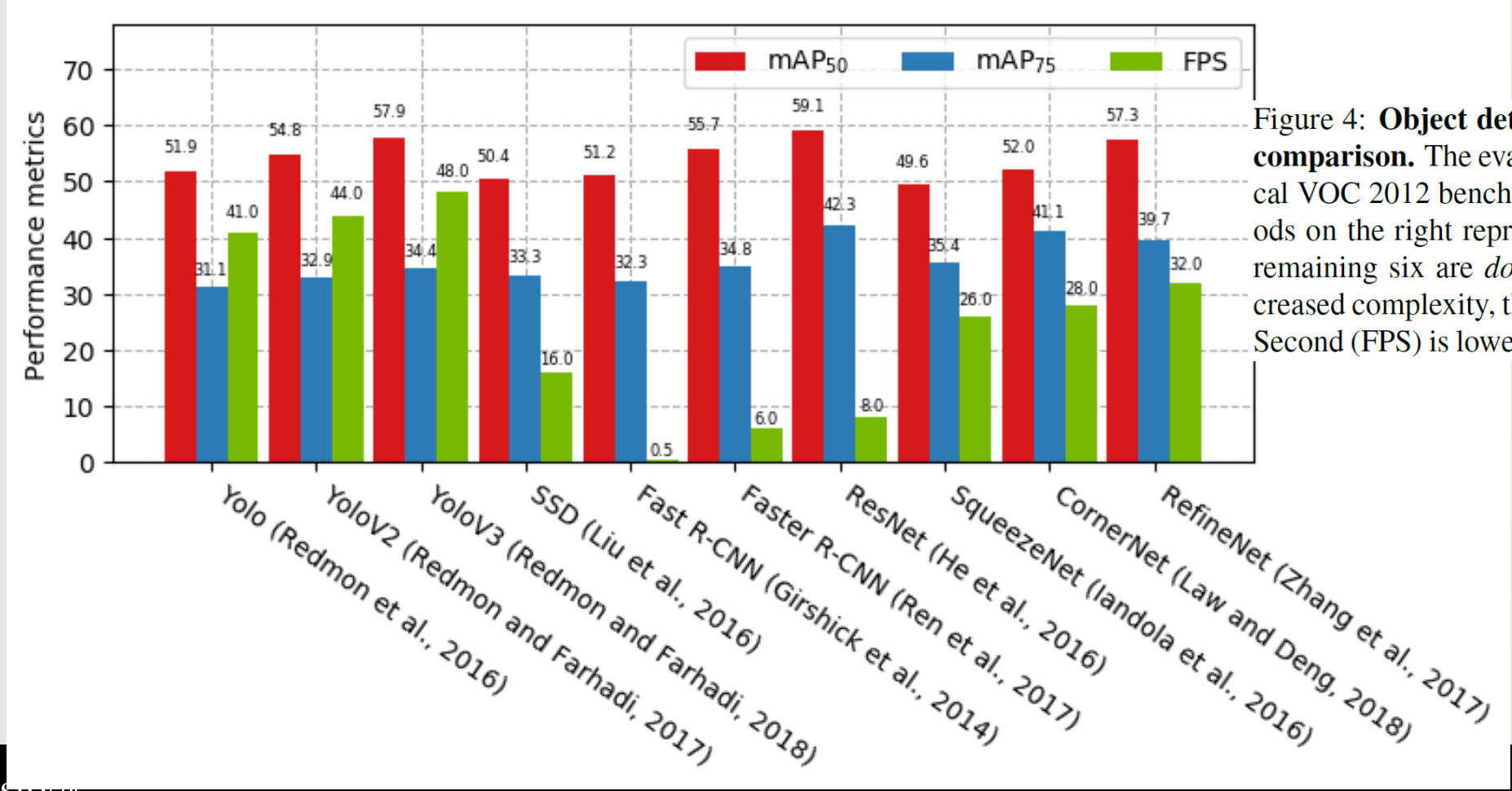


Figure 4: **Object detection and recognition perform**
comparison. The evaluation has been performed on the
cal VOC 2012 benchmarking database. The first four r
ods on the right represent *single stage* detectors, whil
remaining six are *double stage* detectors. Due to the
creased complexity, the runtime performance in Frames
Second (FPS) is lower for the case of double stage dete

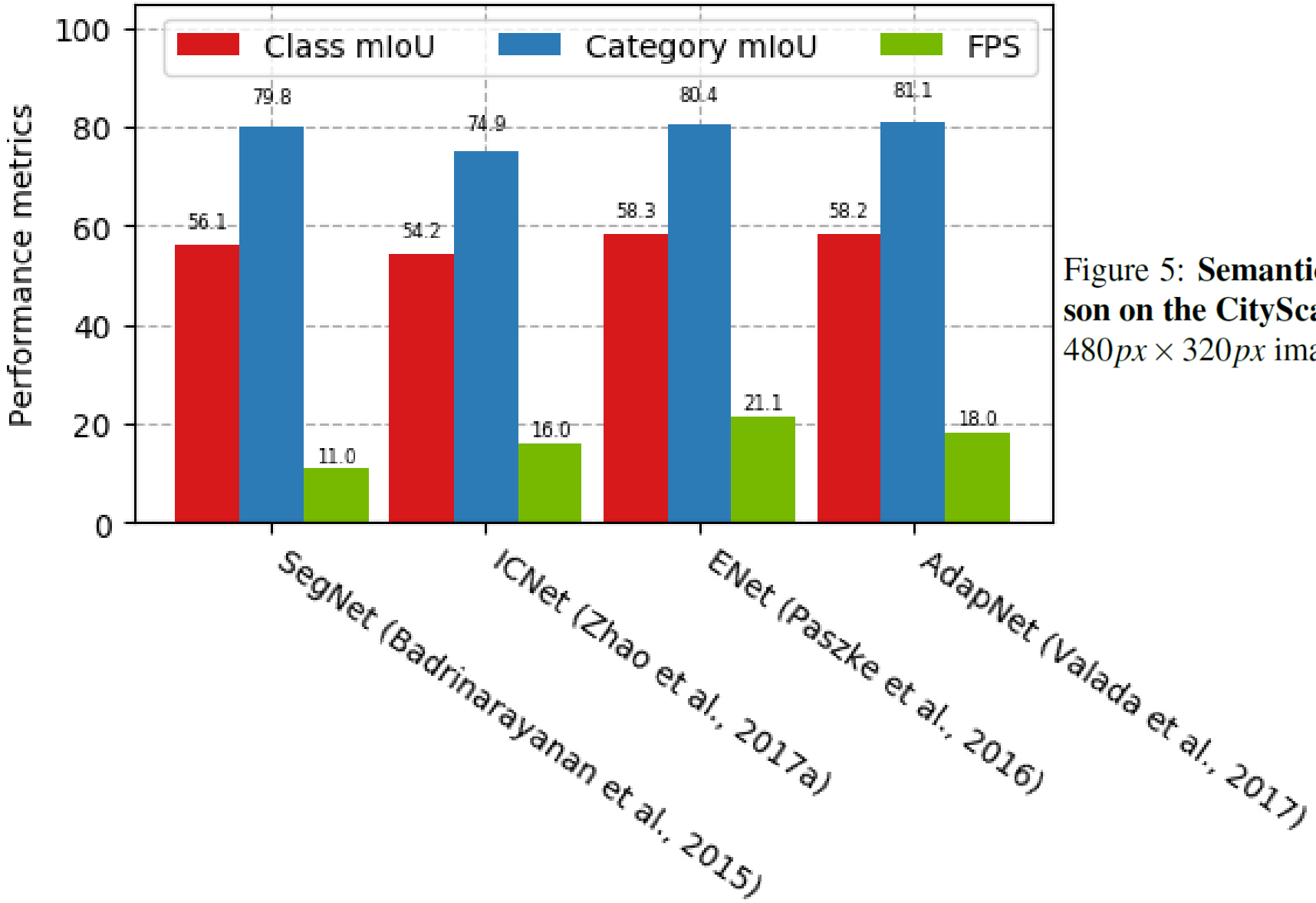
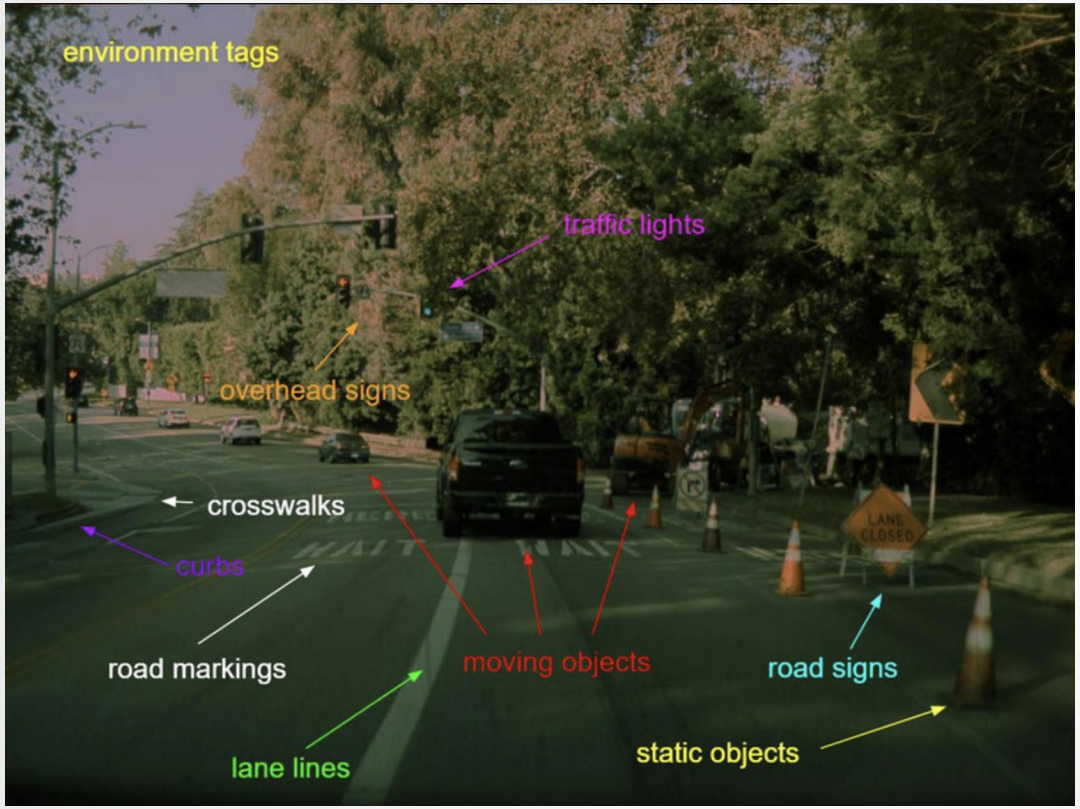


Figure 5: **Semantic segmentation performance comparison on the CityScapes dataset** [74]. The input samples are $480px \times 320px$ images of driving scenes.

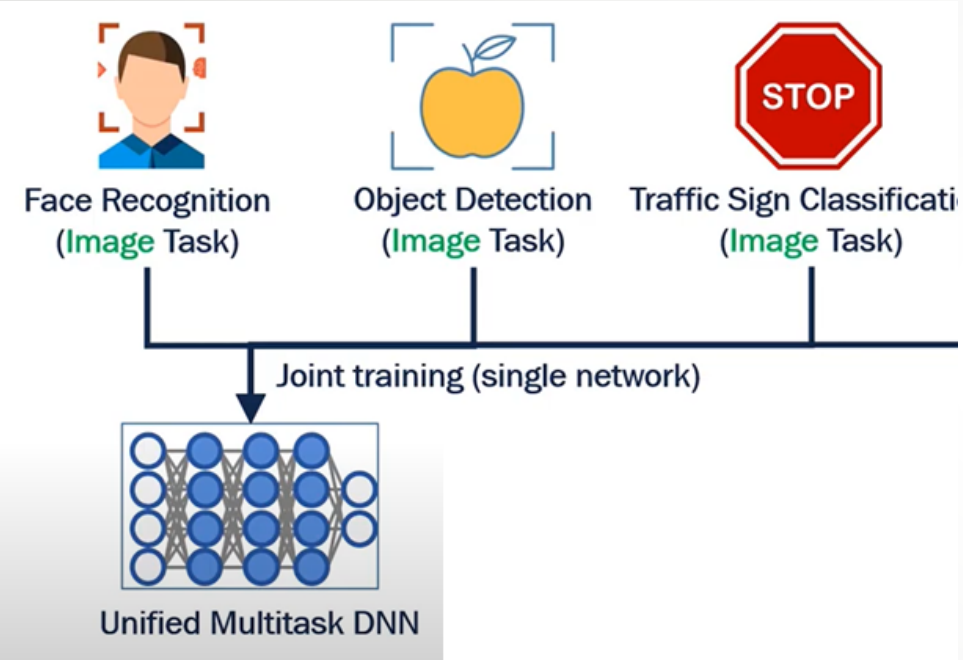
MULTI-TASKS


environment tags
traffic lights
overhead signs
crosswalks
curbs
road markings
moving objects
road signs
lane lines
static objects
LANE CLOSED

MULTI-TASKS


SCHOOL BUS
Each task has additional "sub-tasks", e.g. object types, vehicle classes, blinkers, breaks, parked bit, ...

MULTI-TASKS LEARNING



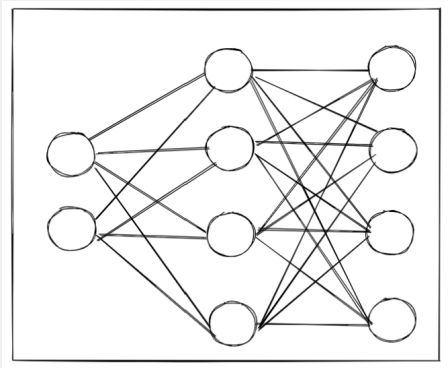
Objectives

- ✓ Packing multiple DNNs into a limited main memory. Largely alleviate memory occupation, especially for mobile IoT devices.
- ✓ Reduce inference time. Enabling fast in-memory execution of the multi-task DNNs.
- ✓ Retain the DNN models inference accuracy.

MULTI-TASKS LEARNING

- **All-in-one - Multi-task Learning**

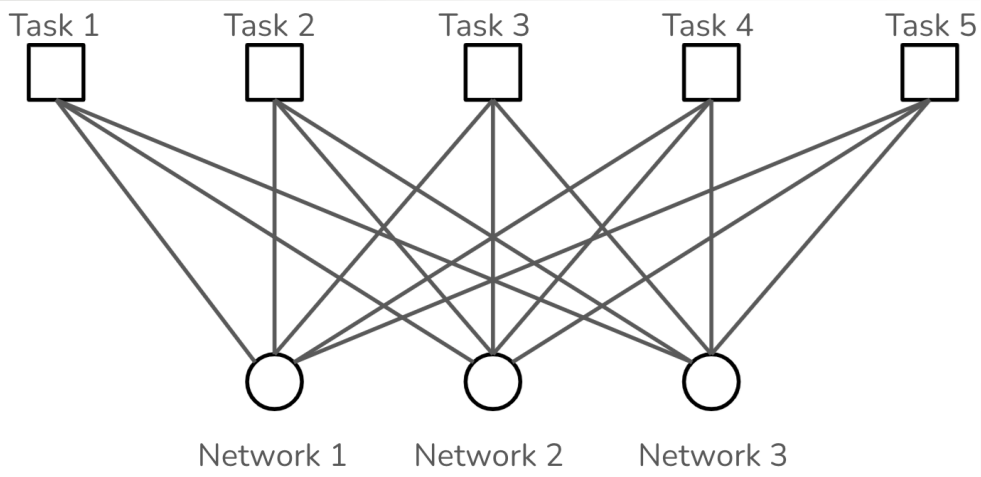
Moving Objects + Static Objects + Signs + Traffic Lights



- Pros:
  - Decreased inference time
  - More compact models
  - Better learned representations
- Cons:
  - Fully coupled functionality
  - Tasks may learn at different rates
  - One task may dominate learning
  - Gradients may interfere
  - The optimization landscape may be more difficult

MULTI-TASKS LEARNING

- **Which tasks are chosen to train together?**



Task grouping goal

- Assign tasks to networks
- Maximize performance
- Keeping inference time to a given fixed budget

Task Grouping Framework

- Create a set of candidate networks
- Select the best networks for our budget
- Proposed a branch and bound like algorithm finds the optimal solution to problems quickly (< 1 sec)

ROADMAP

- **Introduction**
- **Architecture**
- **Four major tasks**
  - ✓ Localization
  - ✓ Perception
  - ✓ Prediction
  - ✓ Planning
- **Datasets**
- **Evaluation**
- **Multi-task opportunities**