深蓝学院
shenlanxueyuan.com

感知介绍

- 概览
- 感知四要素
- 如何处理一套合适的感知系统
- Apollo感知
- 总结

概览

感知四要素

如何处理一套合适的感知系统

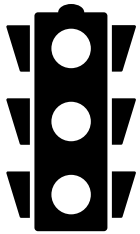
Apollo感知

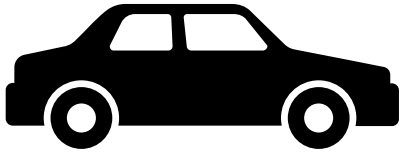
总结

感知概览

❑ 在自动驾驶中的角色

- 回忆你在开车的时候，关注什么？

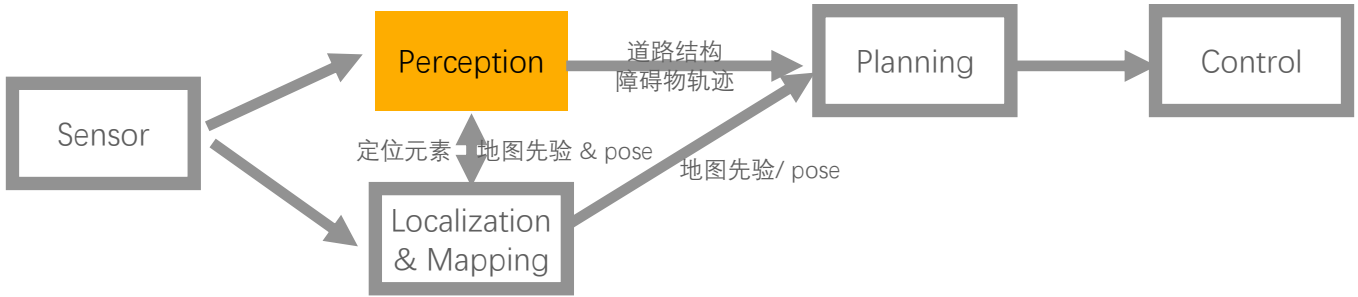




感知概览

❑ 在自动驾驶中的角色

- 提供周围360度的环境信息
  - 看环境：障碍物检测、红绿灯检测、车道线检测
  - 理解环境：多传感器障碍物融合、红绿灯识别、车道线跟踪
  - 信息预判：轨迹预测



概览

感知四要素

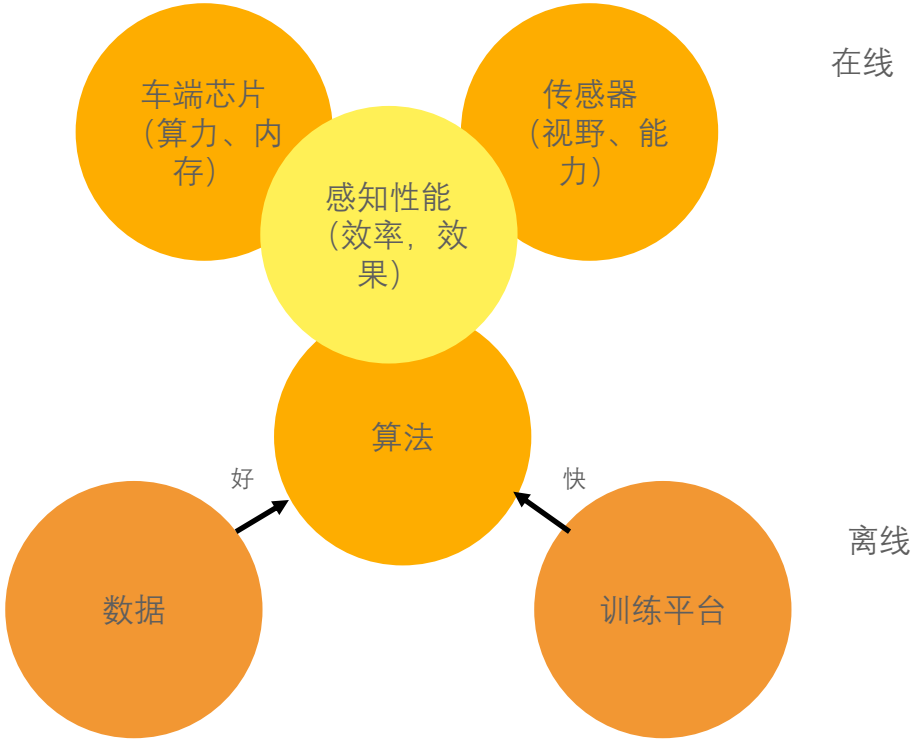
如何处理一套合适的感知系统

Apollo感知

总结

感知的要素

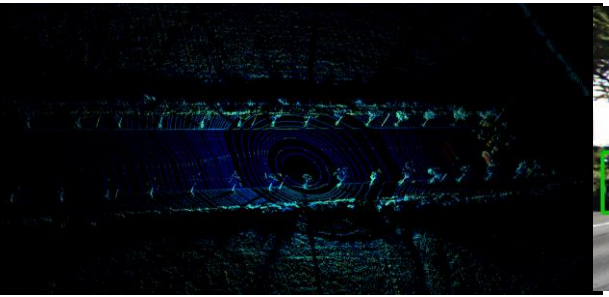
- 传感器
- 算法
- 数据
- 计算平台
  - 在线：车端芯片
  - 离线：训练平台

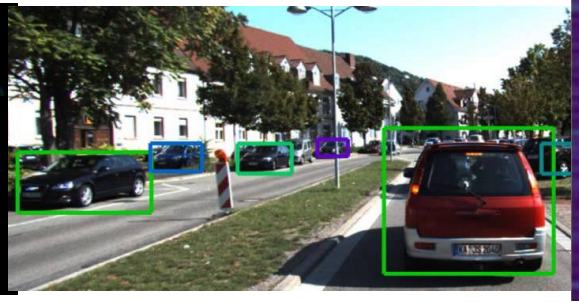


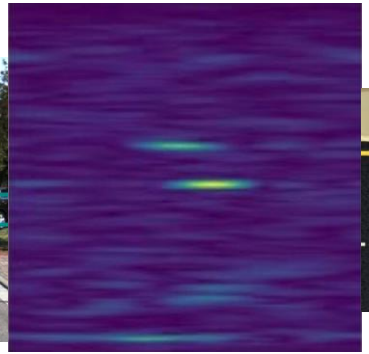
感知的要素

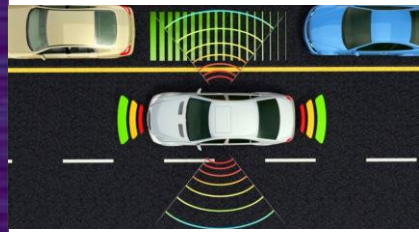
❑ 传感器

- 激光雷达（LiDAR，Light Detection And Ranging）
- 相机（Camera）
- 毫米波雷达（常称Radar，Radio Detection And Ranging，实际是millimeter wave Radar）
- 超声波雷达（Ultrasonic Radar）









感知的要素

❑ 各传感器优劣势分析

	camera	lidar	radar	sonar
Object detection	🟡	🟢	🟡	🔴
Object classification	🟢	🟡	🔴	🔴
distance	🟡	🟢	🟢	🔴
velocity	🔴	🟡	🟢	🔴
Range of visibility	🟡	🟡	🟢	🔴
Feature / Texture	🟢	🟡	🔴	🔴
Lane tracking	🟢	🔴	🔴	🔴
Functionality in bad weather	🔴	🟡	🟢	🔴
Functionality in poor lighting	🟡	🟢	🟢	🟢

Why Lidar:

- 提供深度信息，高精度bev环境描述

Why camera:

- 提供颜色纹理信息，处理更复杂的环境;
- 更远的观测，提前决策;

Why Radar:

- 提供速度观测，处理突然出现的障碍物;
- 恶劣天气的鲁棒处理;

Why sonar:

- 近处泊车预警

感知的要素

❑ 算法

Camera

lidar

Radar

...

object detection

segmentation

lane detection

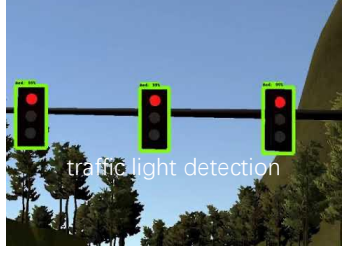
traffic light

object tracking

lanemark detection

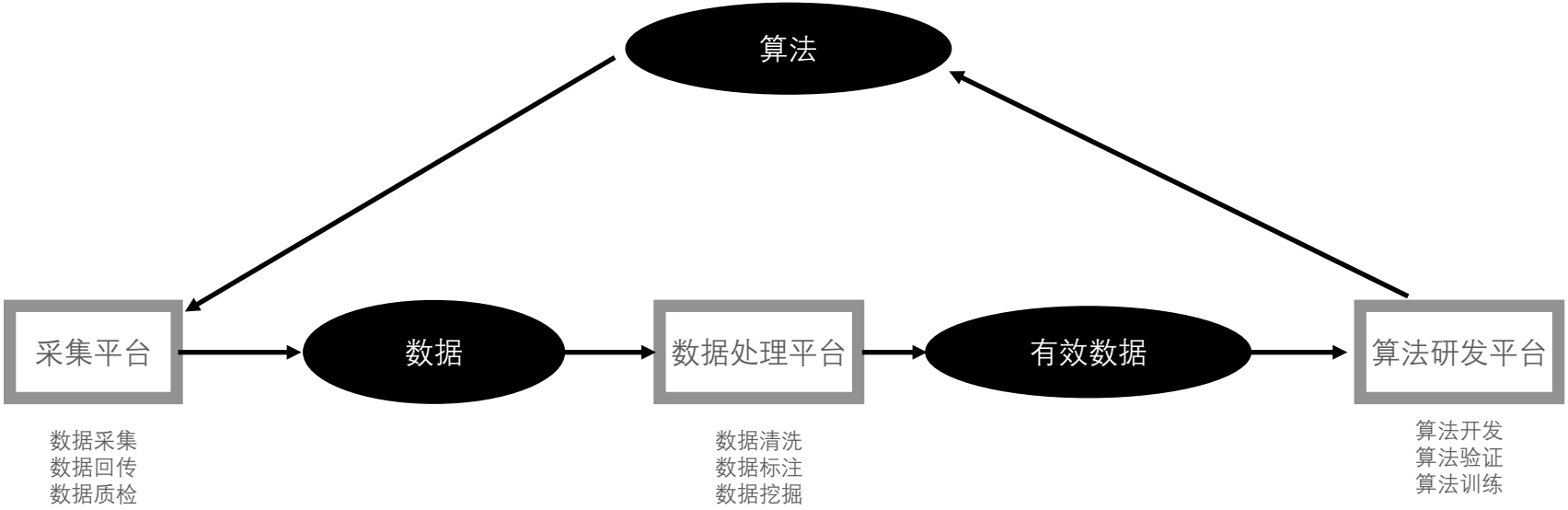
Trajectory prediction

...



感知的要素

❑ 数据

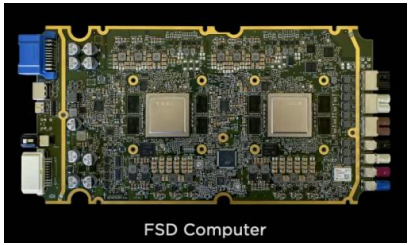


- 数据是自动驾驶效果的基石
- 数据的多样性决定了产品的成熟度（demo->产品）
- 数据的挖掘效率和速度决定了产品迭代的速度

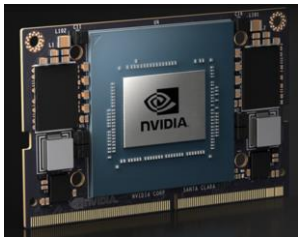
感知的要素

❑ 计算平台

- 车端：例如orin、Xavier
  - gpu / fpga
  - cpu
- 离线平台：例如dojo
  - 训练平台
  - 验证平台



Tesla FSD computer



xavier

orin

dojo：5760个GPU

Tesla AI day：https://www.youtube.com/watch?v=j0z4FweCy4M&t=6130s

概览

感知四要素

如何处理一套合适的感知系统

Apollo感知

总结

如何搭建一套合适的感知系统

❑ 行业主流方案

如何搭建一套合适的感知系统

❑ 行业主流方案

如何搭建一套合适的感知系统

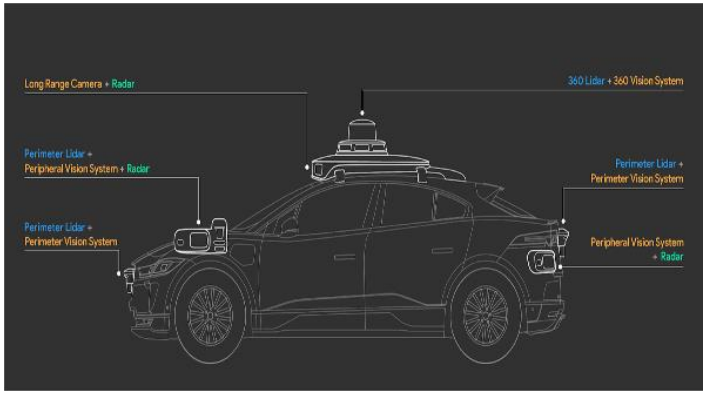
❑ 行业主流方案

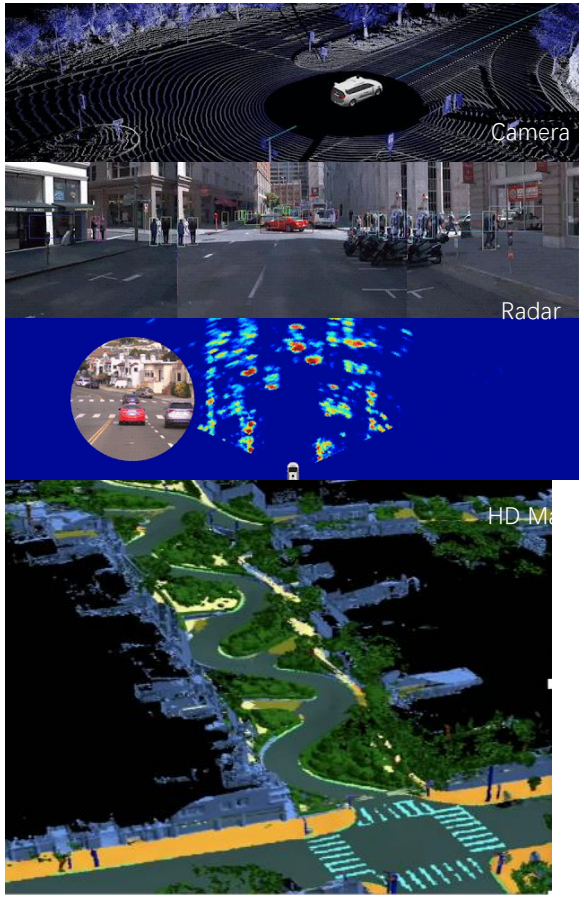
	应用领域	代表公司	技术路线	特点
跨越式路线	Robotaxi	Waymo / 百度	lidar为主的多传感器融合方案	高成本 场景复杂度高 强依赖高精度地图 特定区域
	末端物流	Nuro / 阿里 / 美团	lidar为主的多传感器融合方案	
渐进式路线	车企	Tesla	纯视觉方案	低成本 场景复杂度低 弱/无依赖高精度地图 规模化
		蔚来/小鹏/长城	视觉为主的多传感器融合方案	
	OEM	Mobileye	视觉子系统 + lidar / radar子系统	

跨越式路线

❑ waymo

- 输入：传感器（lidar、camera、radar）+ 高精度地图 （传感器深度自研）
- 技术路线：直奔L4，lidar为主，camera、radar为辅，依赖高精度地图先验
- 现状：指定区域（凤凰城、San Francisco）的无人驾驶
- 计算平台：Xeon处理器 + Arria FPGA
- 数据：自采为主(详细数据量未透露)

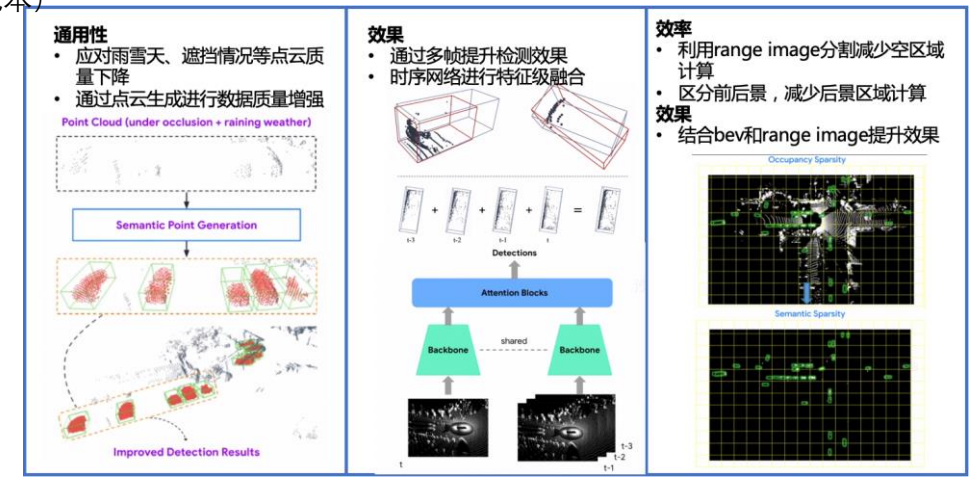




跨越式路线

❑ waymo

- 车端/在线系统
  - 感知算法效果
  - 计算效率（算力、内存）
  - 通用性（城市、天气、长尾问题）
- 离线系统
  - 数据灵活可用（数据量[真实数据，仿真数据]、存储成本）
  - 自动标注（时间成本、金钱成本）
- 车端与离线系统的交互
  - 离线系统提供更多的数据
  - 车端系统的问题指导离线模型的迭代

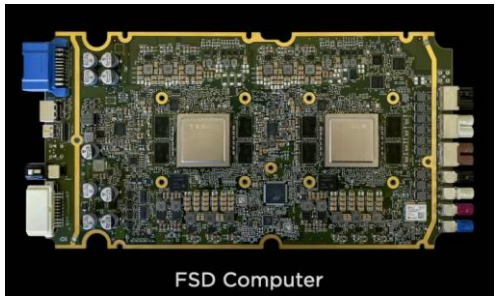


https://arxiv.org/abs/2108.06709
https://arxiv.org/abs/2103.16054
https://arxiv.org/abs/2106.13365

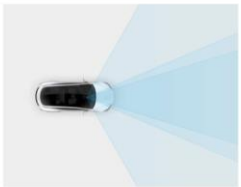
渐进式路线

❑ Tesla

- 输入：传感器（camera、radar，宣称去除radar）+ 弱高精度地图
- 技术路线：L2+ -> L3+，纯视觉感知方案
- 现状：大多数区域的辅助驾驶
- 计算平台：自研FSD，144Tops
- 数据：100w个10秒视频数据、60亿个标记物体，总计达1.5petabytes的数据（量产回传为主）



dojo：5760个GPU



前视摄像头

侧视摄像头

后视摄像头

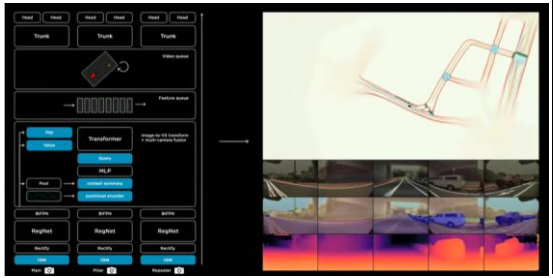
深度定制相机模组（1280x960 12bit HDR 36Hz）

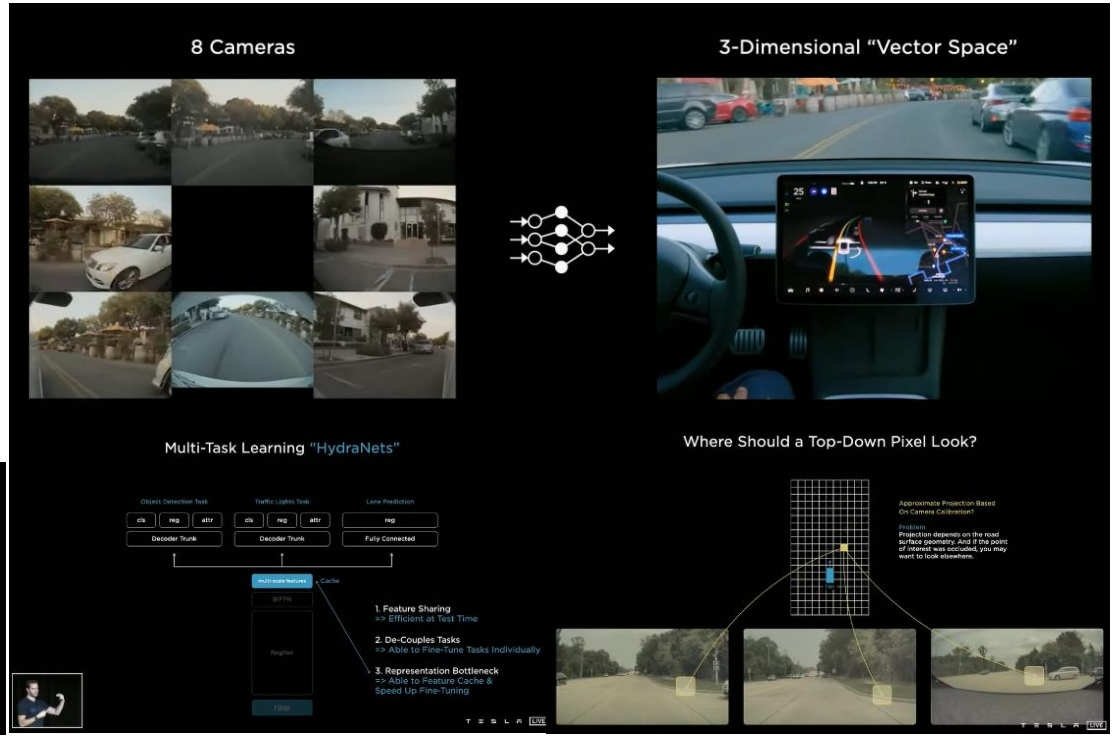
Tesla AI day：https://www.youtube.com/watch?v=j0z4FweCy4M&t=6130s

渐进式路线

❑ Tesla

- 多任务框架：HydraNets
  - 共享backbone，高效推理
  - 多head解耦任务，独立迭代
- 多相机融合
  - 直接检测输出vector space
  - transformer进行跨相机特征融合
- learn based 时序模型
  - RNN连接时序



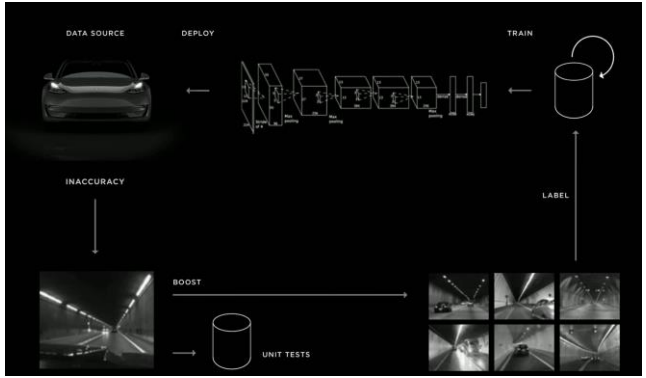


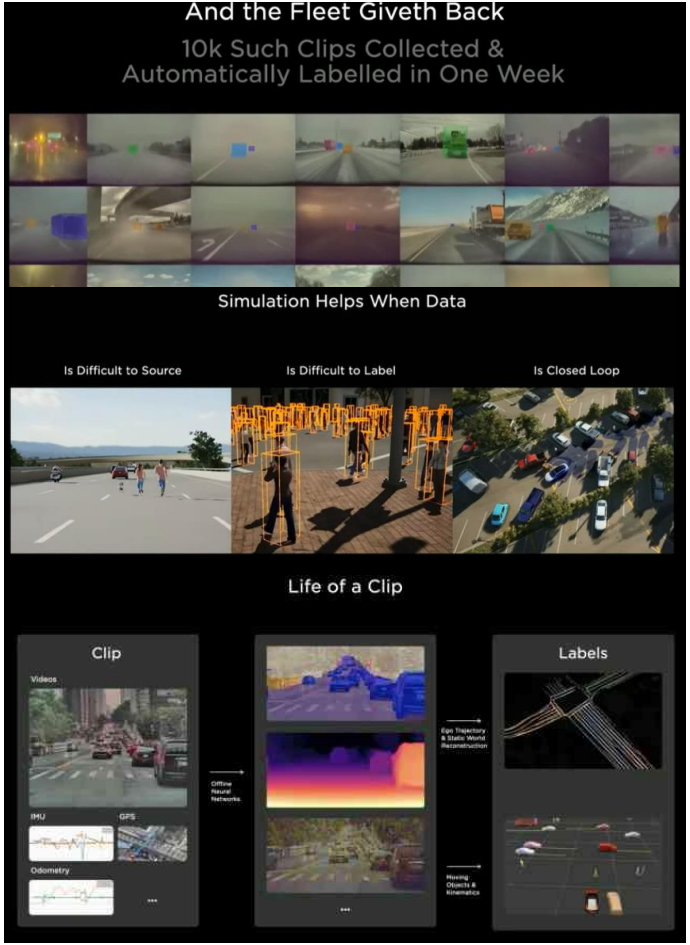
Tesla AI day：https://www.youtube.com/watch?v=j0z4FweCy4M&t=6130s

渐进式路线

❑ Tesla

- 数据闭环
  - 影子模式（快速收集case）
- 4D 数据标注获取
  - 自动标注流程
  - 三维重建
  - 仿真引擎





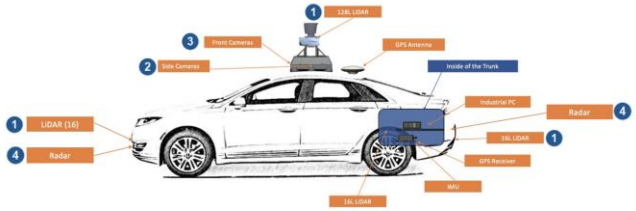
Tesla AI day：https://www.youtube.com/watch?v=j0z4FweCy4M&t=6130s

- 概览
- 感知四要素
- 如何处理一套合适的感知系统
- Apollo感知
- 总结

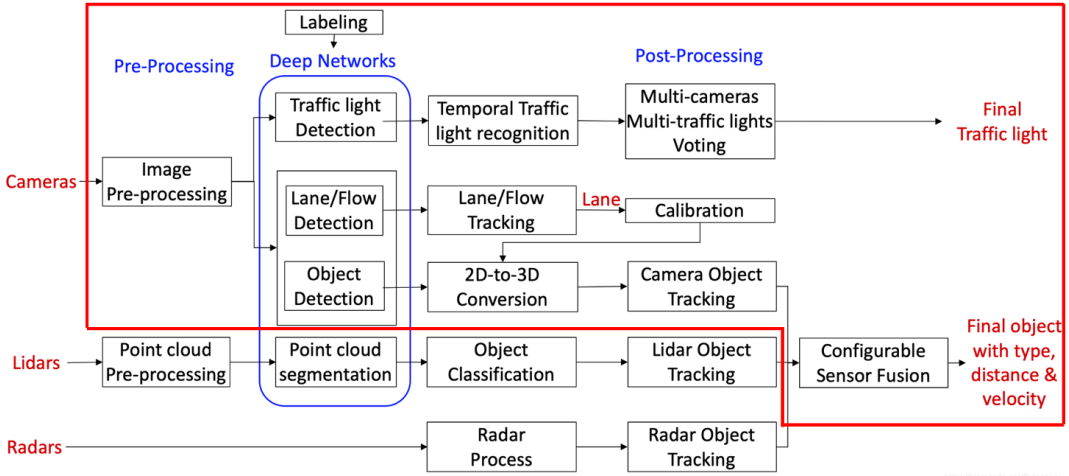
Apollo感知

❑ 大纲

- 视觉模块
- 激光雷达模块
- 融合模块



1. 1x 360° LiDAR+n个补盲LiDAR：有效检测距离约150m；
2. 4x 侧视camera，FOV 90°，分别：
   - 2x 测前视，
   - 2x 侧后视；
3. 3x 前视camera，分别：
   - FOV 30°，最远距离250米，
   - FOV 60 °，最远检测距离150米，
   - FOV 120 °，最远检测距离60米；
4. 1x前视radar+1x后视radar：最远检测距离160m。

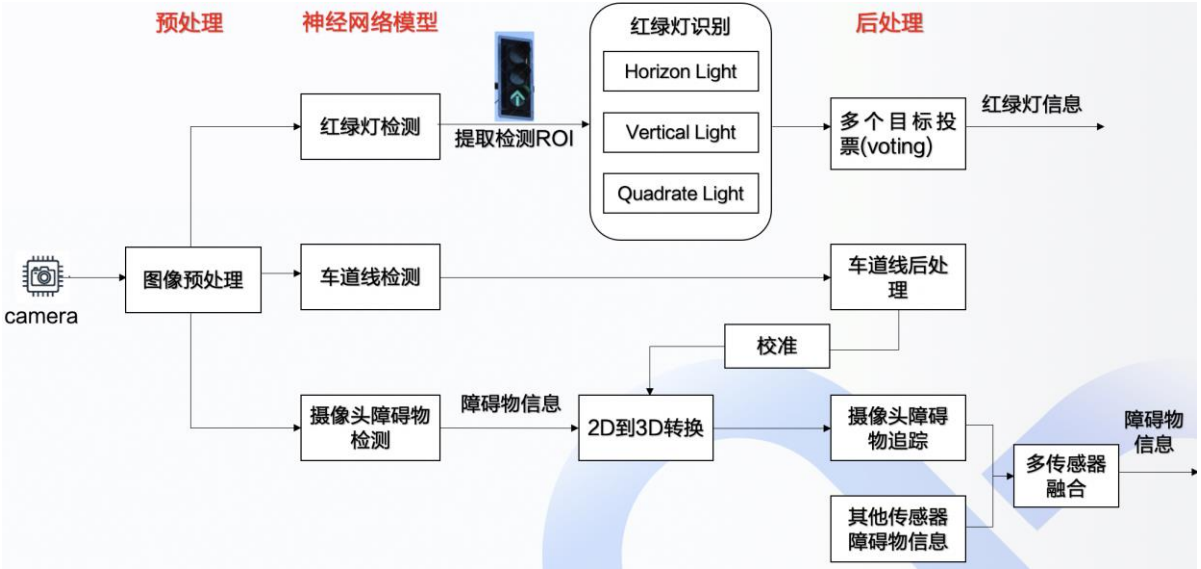


https://apollo.auto/Apollo-Homepage-Document/Apollo_Doc_CN_6_0

Apollo感知

❑ 视觉感知

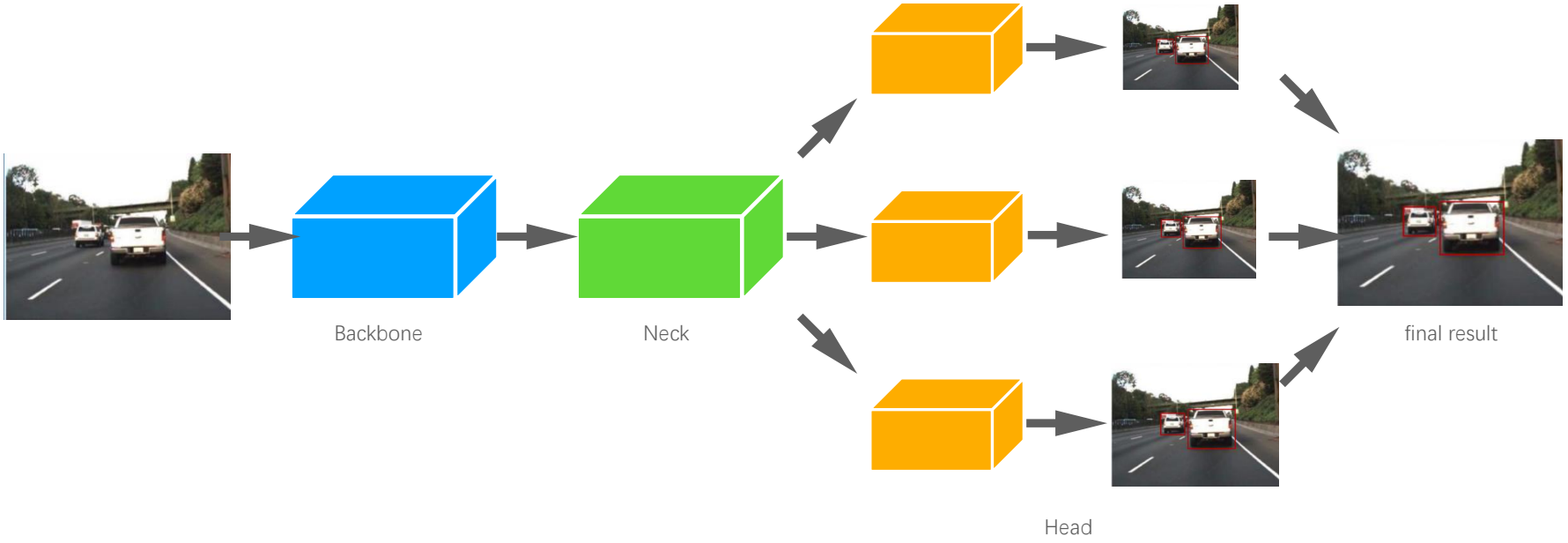
- 核心功能：看周围的世界
  - 红绿灯
  - 车道线
  - 障碍物



https://apollo.auto/Apollo-Homepage-Document/Apollo_Doc_CN_6_0/上机使用教程/上机实践Apollo视觉感知能力/Apollo视觉感知能力介绍

Apollo感知

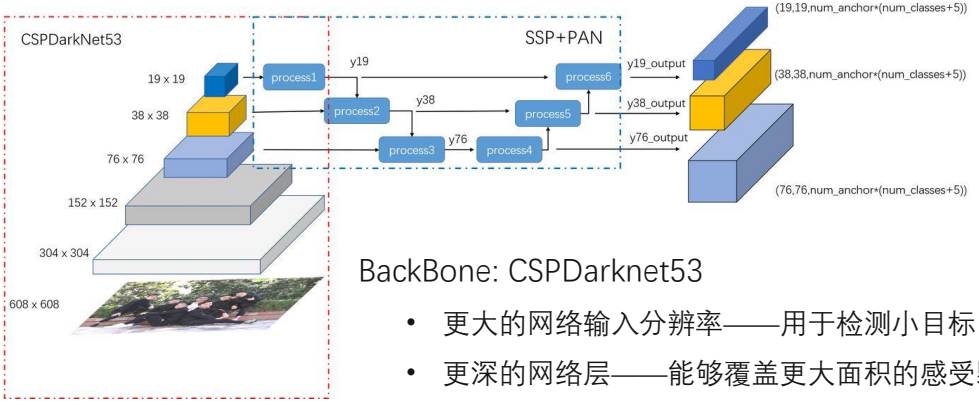
❑ 视觉感知-障碍物检测网络

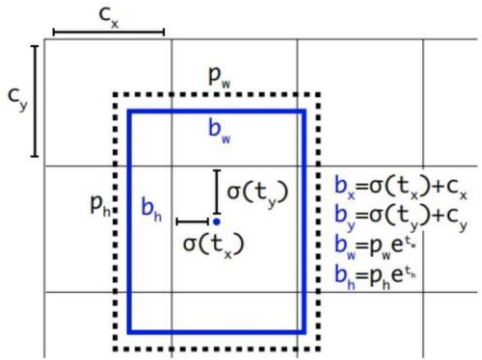


Apollo感知

❑ 视觉感知- YOLO （You Only Look Once）

tx, ty, tw, th, prob





BackBone: CSPDarknet53

- 更大的网络输入分辨率——用于检测小目标
- 更深的网络层——能够覆盖更大面积的感受野
- 更多的参数——更好的检测同一图像内不同size的目标

Neck: SPP & PAN

- 不同尺度特征的处理 —— 增大感受野

Head

- 对图像特征进行预测，生成边界框并预测类别

(cx, cy) : 中心点所处区域的左上角点

(pw, ph) : anchor的宽高

(bx, by)：预测框中心点

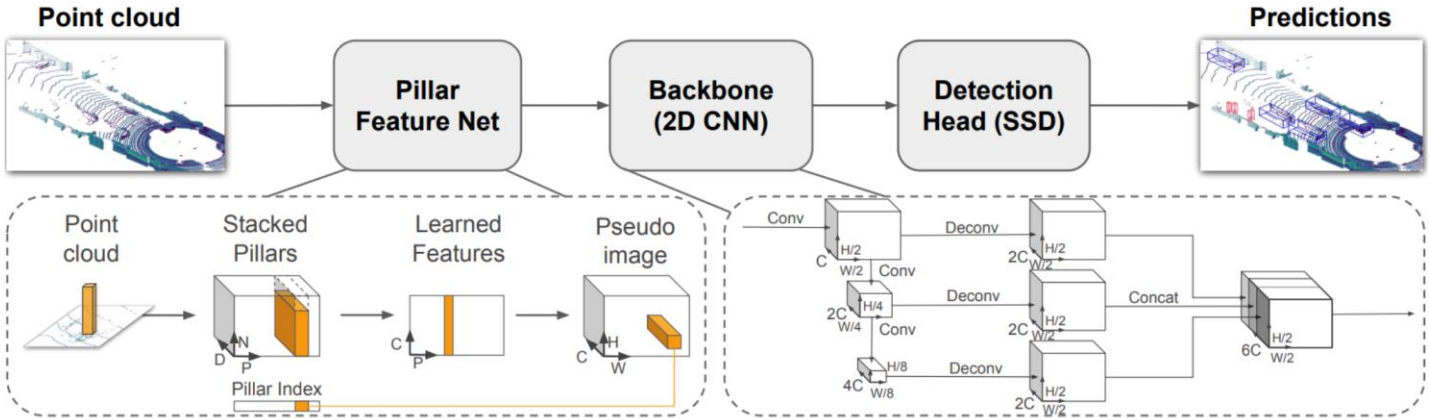
(bw, bh)：预测框宽高

https://zhuanlan.zhihu.com/p/159209199

Apollo感知

❑ 激光雷达感知

- pillar feature net：提取点特征(点视图)，然后进行pillar化(俯视图)
- backbone：Feature Pyramid Network
- head：SSD



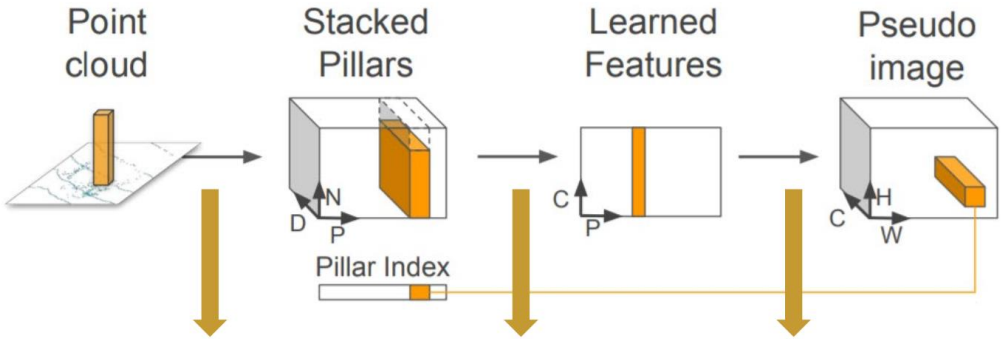
https://arxiv.org/abs/1812.05784

Apollo感知

❑ 激光雷达感知

- pillar feature net：提取点特征(点视图)，然后进行pillar化(俯视图)

输入：x, y, z, intensity



- X-Y平面网格量化
  - 一个网格 = 一个Pillar
  - 97%Pillar为空（KITTI, 0.16m）
- 每个点用9维特征表示
  - x, y, z, r, x_c, y_c, z_c, x_p, y_p
- Pillar的表示: Tensor（P, N, D）
  - D=9, P=非空Pillar数量，N=Pillar中的点数

- PointNet提取点特征
  - (P, N, **D**) => (P, N, **C**)
- MaxPooling
  - (P, **N**, C) => (P, C)

- Pillar映射回X-Y平面
  - (P, C) + Pillar的索引 => (H, W, C)

Apollo感知

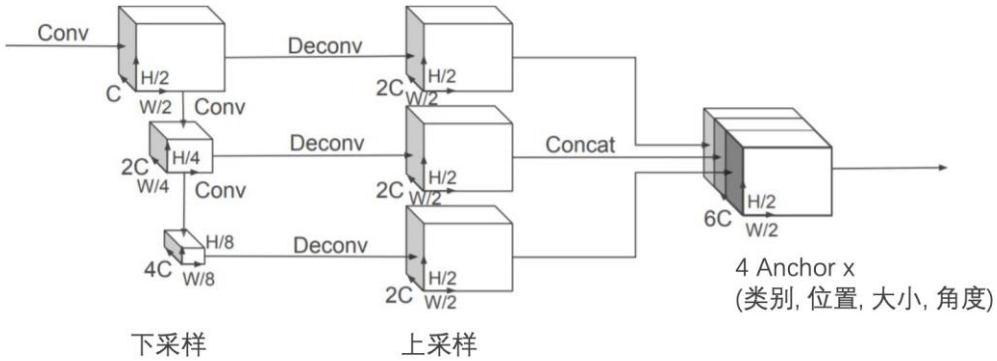
❑ 激光雷达感知

- backbone：Feature Pyramid Network

- head：SSD

  输出：

  - box中心：x, y, z
  - box大小：w, l, h
  - box角度：dir
  - box类别：class



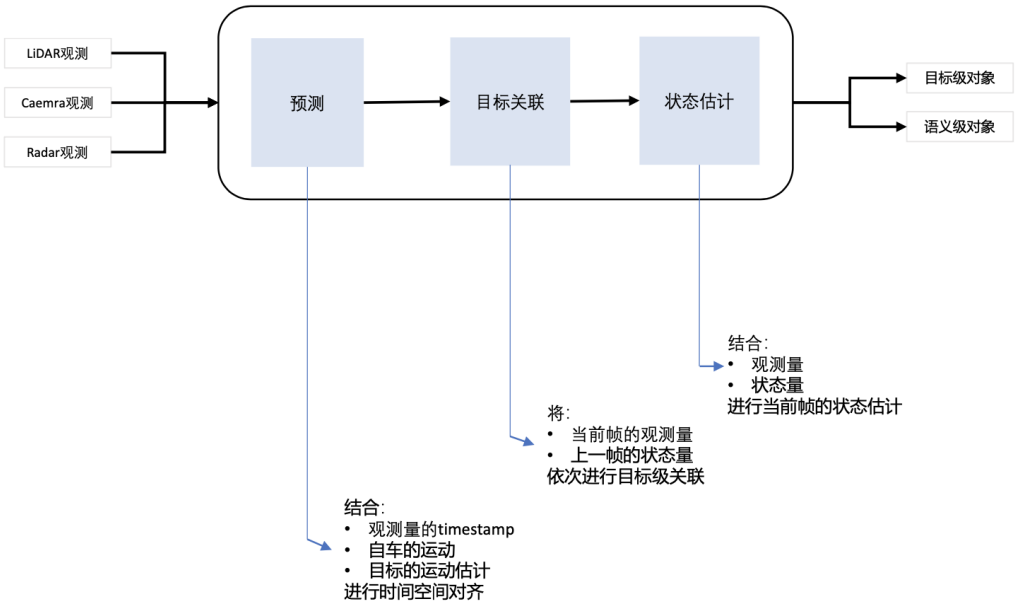
- loss $\mathcal{L} = \frac{1}{N_{pos}}(\beta_{loc}\mathcal{L}_{loc} + \beta_{cls}\mathcal{L}_{cls} + \beta_{dir}\mathcal{L}_{dir})$

  Bbox loss　　类别loss　　朝向角loss

Apollo感知

❑ 融合模块

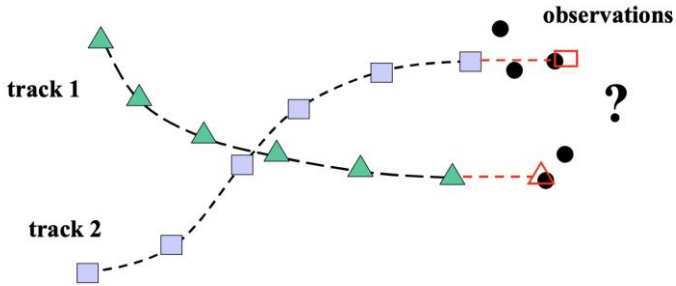
- 核心功能：理解世界
  - 关联
  - 状态估计

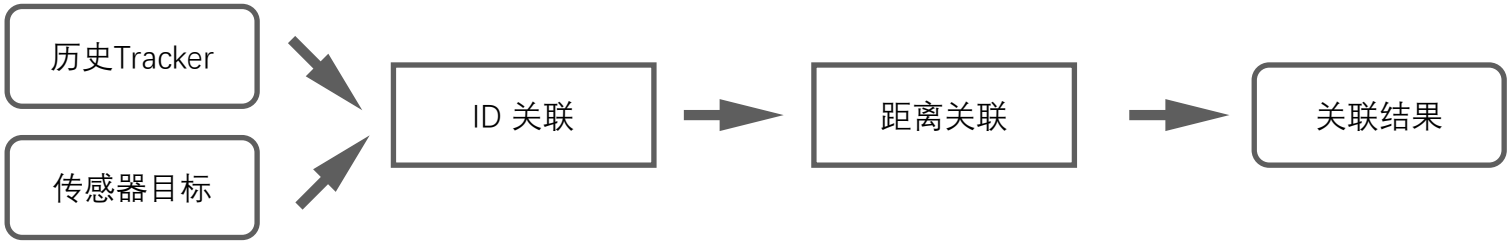


Apollo感知

❑ 融合模块

- 关联： 对当前帧观测量与历史Tracker状态量进行时空对齐，输出关联关系
  - HM（Hungarian Match）
  - Greedy Match
  - JPDAF（Joint Probabilistic Data Association Filter）
  - MHT（Multiple Hypothesis Tracker）

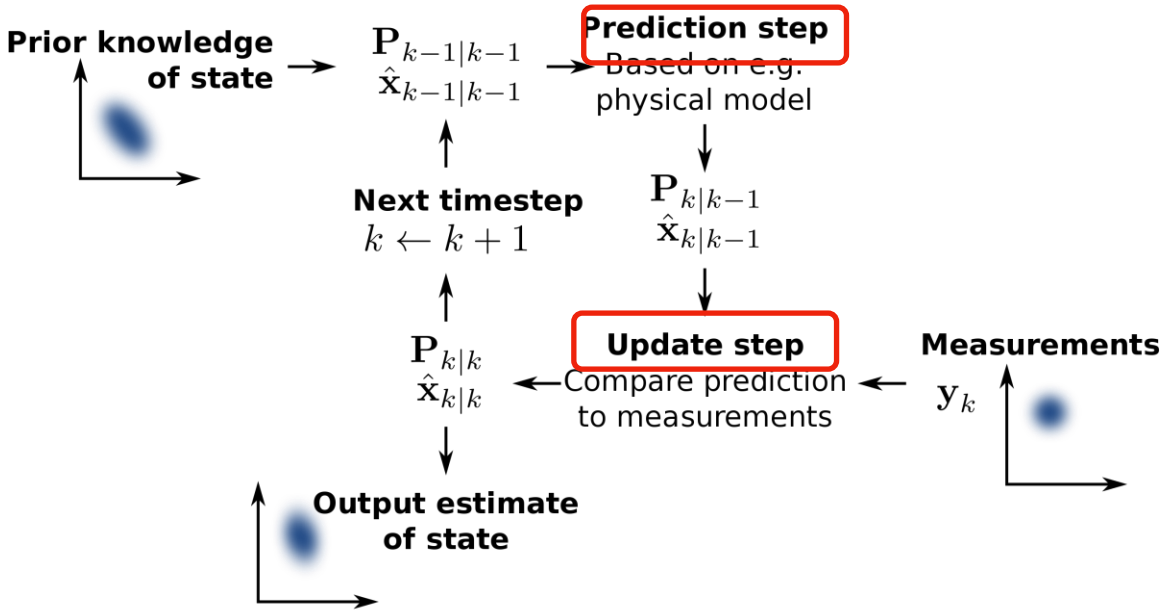




Apollo感知

❑ 融合模块

- 状态估计
  - 运动状态融合
  - 存在性融合
  - 形状融合
  - 轨迹融合
  - 类别融合

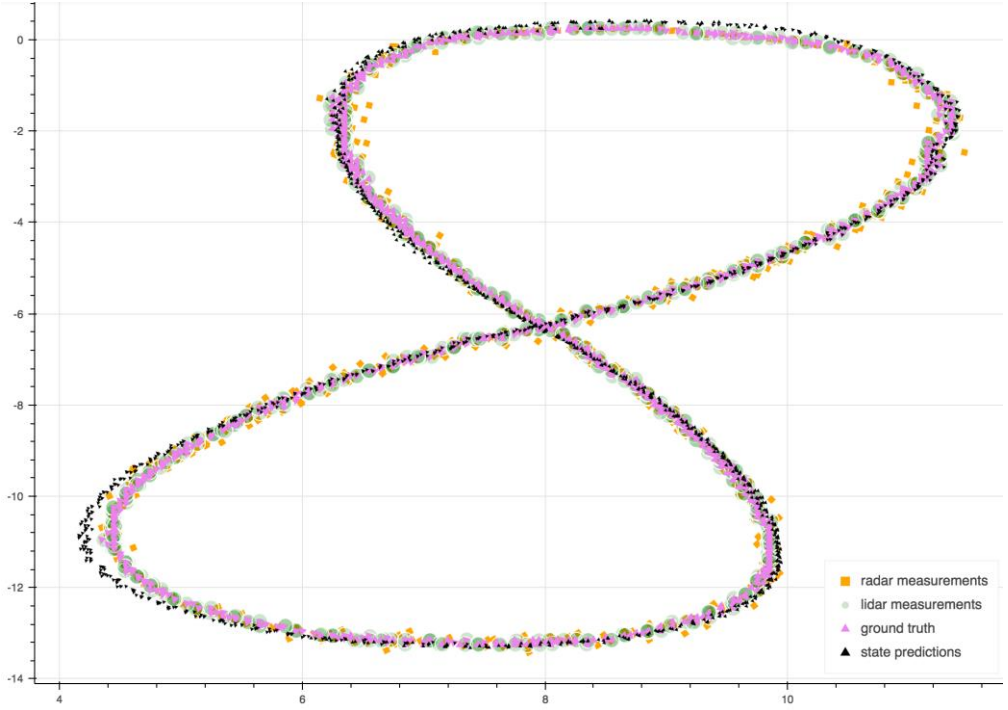


https://en.wikipedia.org/wiki/Kalman_filter

Apollo感知

❑ 融合模块

- why we need:
  - 更完整的fov
  - 更鲁棒（冗余）
  - 更稳定（噪声更小）



Apollo感知算法

❑ 代码概览

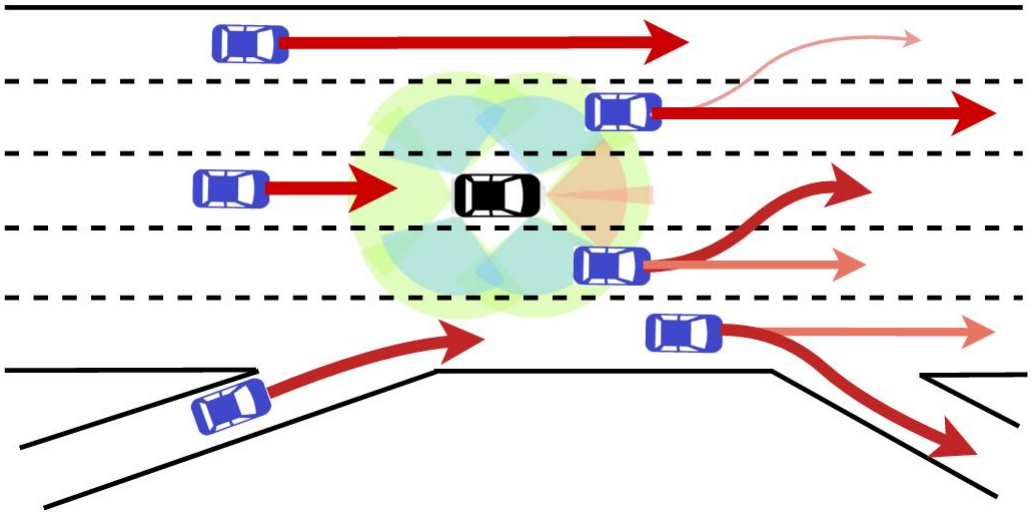
```
.
├── BUILD
├── Perception_README_3_5.md
├── README.md
├── base
├── camera
├── common
├── data
├── fusion
├── inference
├── lib
├── lidar
├── map
├── onboard
├── production
├── proto
├── radar
├── testdata
└── tool
```

base——感知模块的基础类定义，例如障碍物Object;
camera——视觉感知模块;
common——感知模块的基础操作，例如几何信息计算等;
data——感知模块使用的数据;
fusion——融合模块;
Interface——感知模块的接口定义;
lib——感知模块的算法定义;
lidar——激光雷达感知模块;
map——感知模块与高精度地图相关的操作，例如提取ROI;
onboard——实车运行程序，其中component文件夹是感知模块的入口;
production——感知模块的配置参数定义;
proto——感知模块的protobuf定义;
Radar——毫米波雷达感知模块;
Testdata——一些用于测试的数据，如点云pcd文件;
Tool——一些工具方法;

Apollo感知算法

❑ 预测

- 核心功能：信息预判
  - 帮助提前决策



概览

感知四要素

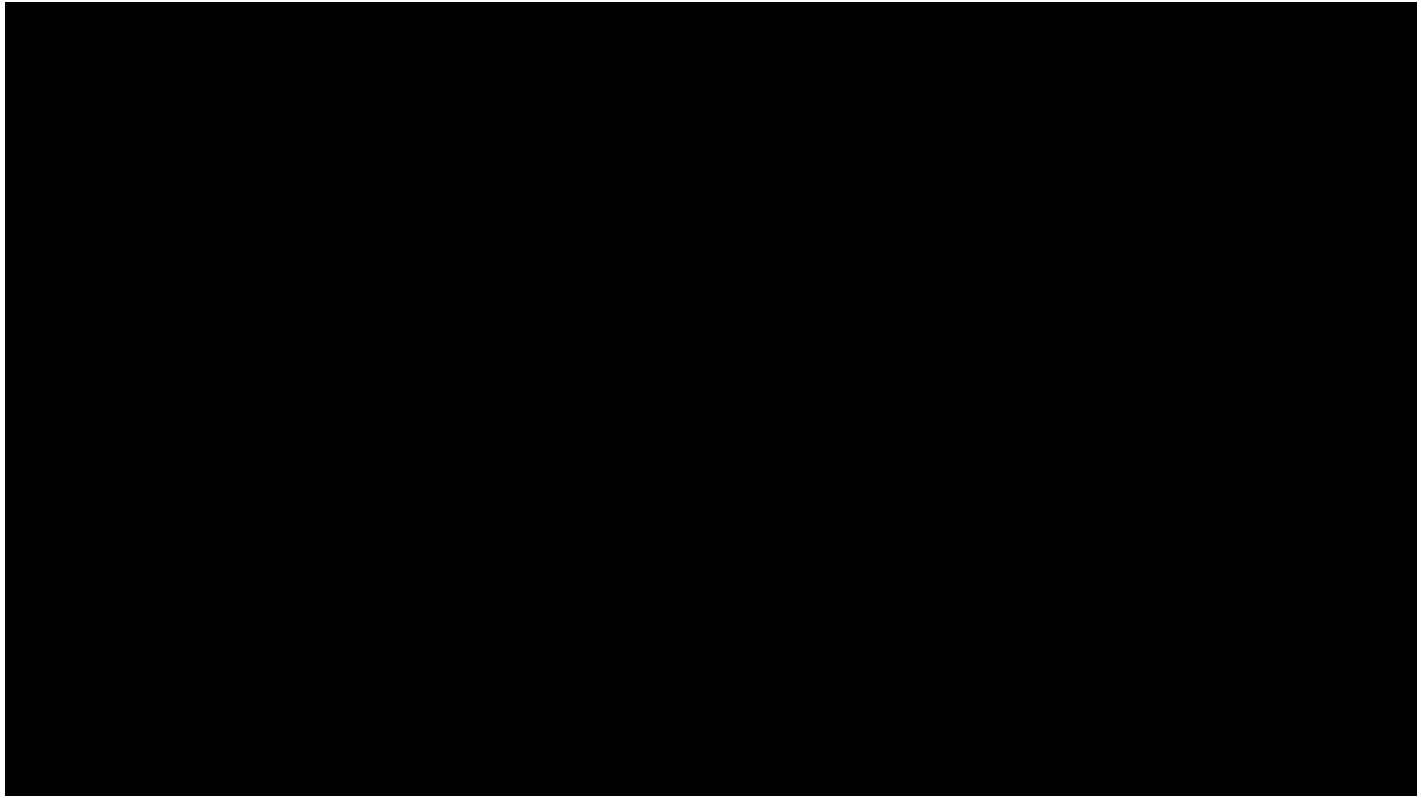
如何处理一套合适的感知系统

Apollo感知

总结

感知

❑ 总结



❑ 总结

- 面对的挑战
  - 层出不穷的复杂场景
  - 恶劣天气情况
  - 成本与效果的权衡
  - …
- 学术界/未来方向
  - 长尾问题的解决
  - 信息功能安全
  - 多模态bev的感知 / 前融合
  - …

多谢观看