

强化学习

第十讲：深度强化学习与自动驾驶

教师：赵冬斌 朱圆恒 张启超

中国科学院大学
中国科学院自动化研究所



June 11, 2021

- ◆ 自动驾驶简介
- ◆ 自动驾驶软件架构
- ◆ 深度强化学习与自动驾驶应用
 - ◆ 视觉输入端到端控制
 - ◆ 基于深度强化学习的决策控制
- ◆ 总结

自动驾驶简介



截至5.28日，新型冠状病毒目前已导致全球**35万人**丧生



“自动驾驶之父”的Sebastian Thrun，未来无人驾驶将会大大降低交通事故，提高安全性能

自动驾驶简介



减缓交通拥堵



减少空气污染

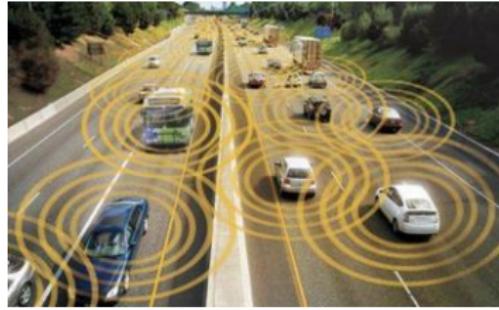


提升乘员舒适性



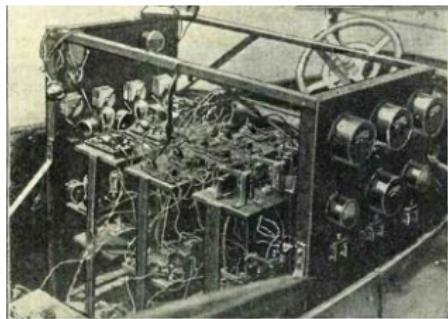
资本市场对于利益的追逐大大加速了自动驾驶产业发展

提升社会经济效益



自动驾驶简介

1925年，无线电设备公司Houdina Radio Control，设计了一辆“无人”驾驶汽车American Wonder，无线电接收设备接收信号，进而操作车的方向盘、制动器、加速器等



1956年，美国通用公司的Firebird II概念车



自动驾驶简介

1961年，斯坦福研发第1辆“自动驾驶”汽车 Stanford Cart



2004/07年开始，美国DAPRA举办沙漠/城市无人驾驶挑战赛



1971年，英国道路研究实验室展示了第一辆无人驾驶车



2009年，首届中国“智能车未来挑战赛”，至今举办10届



自动驾驶简介

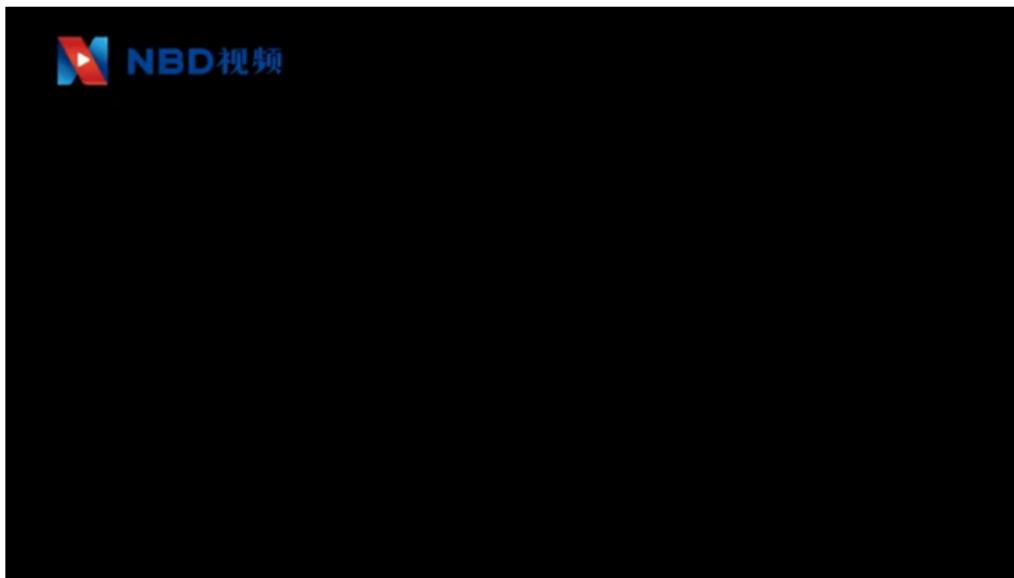
2009年开始，Google在DAPRA的支持下开始无人驾驶研究，2016年底成立Waymo，2020年累计无人驾驶里程达2千万英里，2020年计划造车2万辆



自动驾驶简介



2015年，百度宣布其无人驾驶车已在国内首次实现城市、环路及高速道路混合路况下的全自动驾驶。随后推出了Apollo计划，开放无人驾驶软硬件平台，2018年，自动驾驶里程近14万公里。

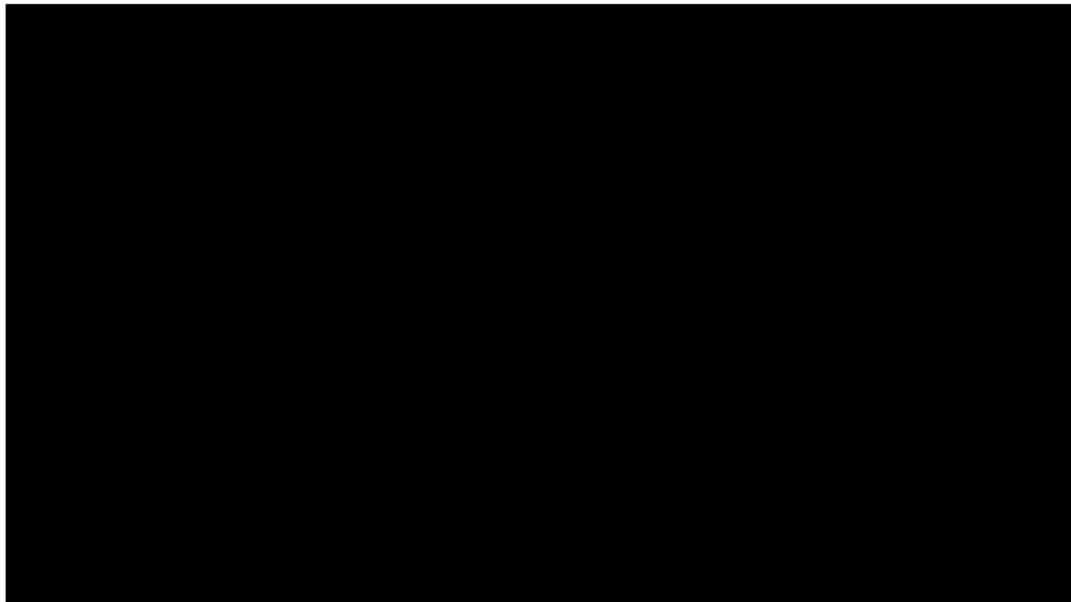


自动驾驶简介



Wayve是剑桥大学博士生创建的自动驾驶公司，旨在利用深度强化学习实现复杂的无人驾驶任务，实现了强化学习在实车上的成功应用。

Wayve: Learning to drive in a day



自动驾驶简介

Wayve: Learned Urban Driving

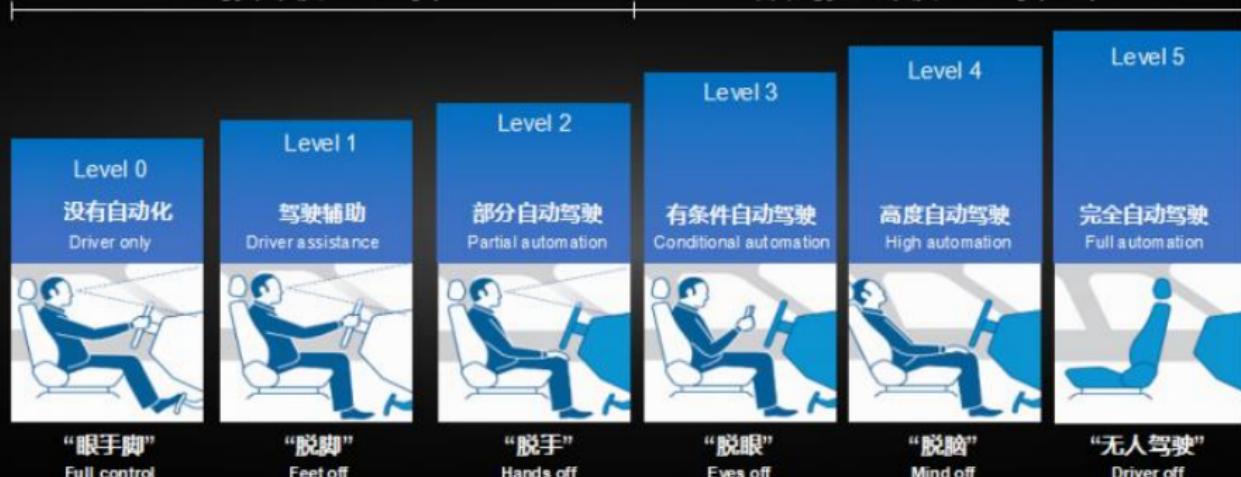
致力于通过强化学习实现自动驾驶



SAE自动驾驶等级划分

驾驶员负责 Monitoring by driver

自动驾驶系统负责 Monitoring by AD system

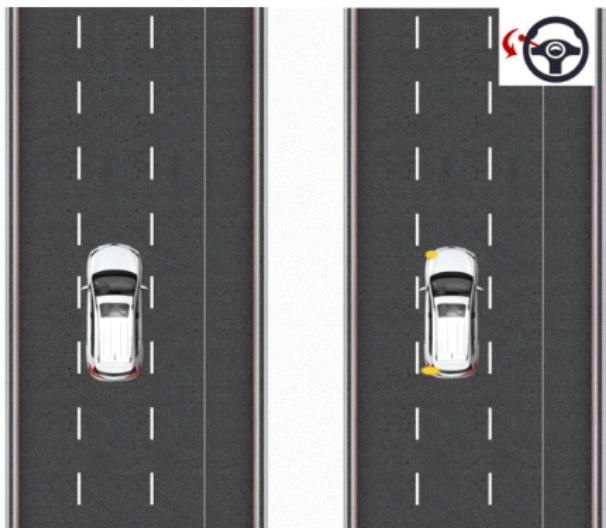


报警类功能

ADAS功能
ACC/AEB...Tesla的
Autopilot根据系统请
求，驾驶员
适当应答Robotaxi
矿车/清扫车

L1级别驾驶辅助功能

预警类



车道偏离

车道偏离预警

控制类



自适应巡航控制/自动紧急刹车

自动驾驶简介

L2级别部分自动驾驶

特斯拉开发的**L2级别辅助驾驶功能**Autopilot，借助于Mobileye的视觉技术和毫米波雷达，2019年累计里程达10亿英里，特别是针对高速路场景。



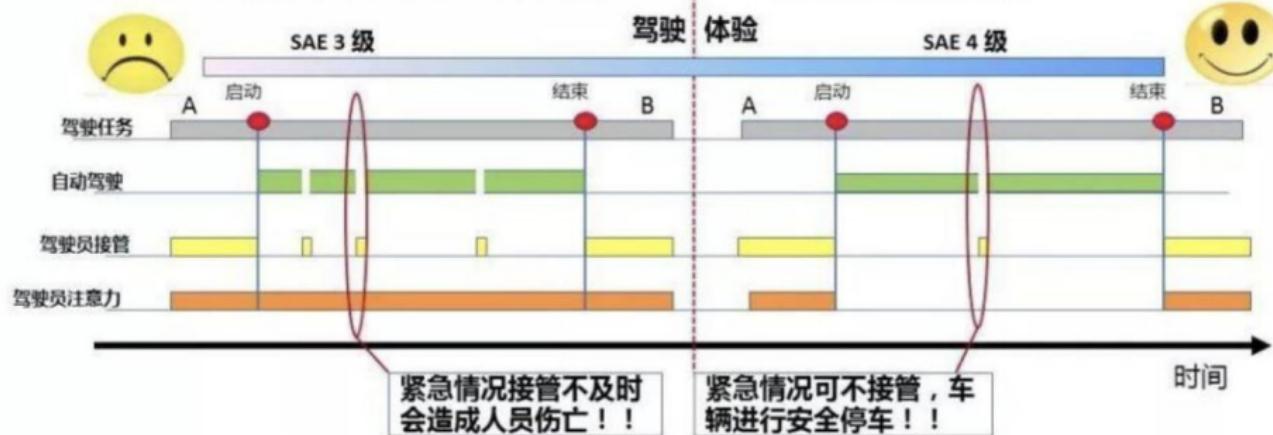
自动驾驶简介

L3级别人机共驾

L3是相对比较模糊的分级，自动驾驶的商业化应用以L3为节点，L3以下的方案称为驾驶员辅助系统(ADAS)，**驾驶员为主导，责任在人而不在车。** L3以上的自动驾驶，以互联网公司为主导，在封闭/开放道路下商业运营，**责任在车而不在人。**

- 驾驶员随时准备接管车辆；
- 若驾驶员未能及时接管车辆，可能产生安全问题。

- 解除驾驶员全程监控；
- 紧急情况可安全停车；



自动驾驶简介

L3级别人机共驾

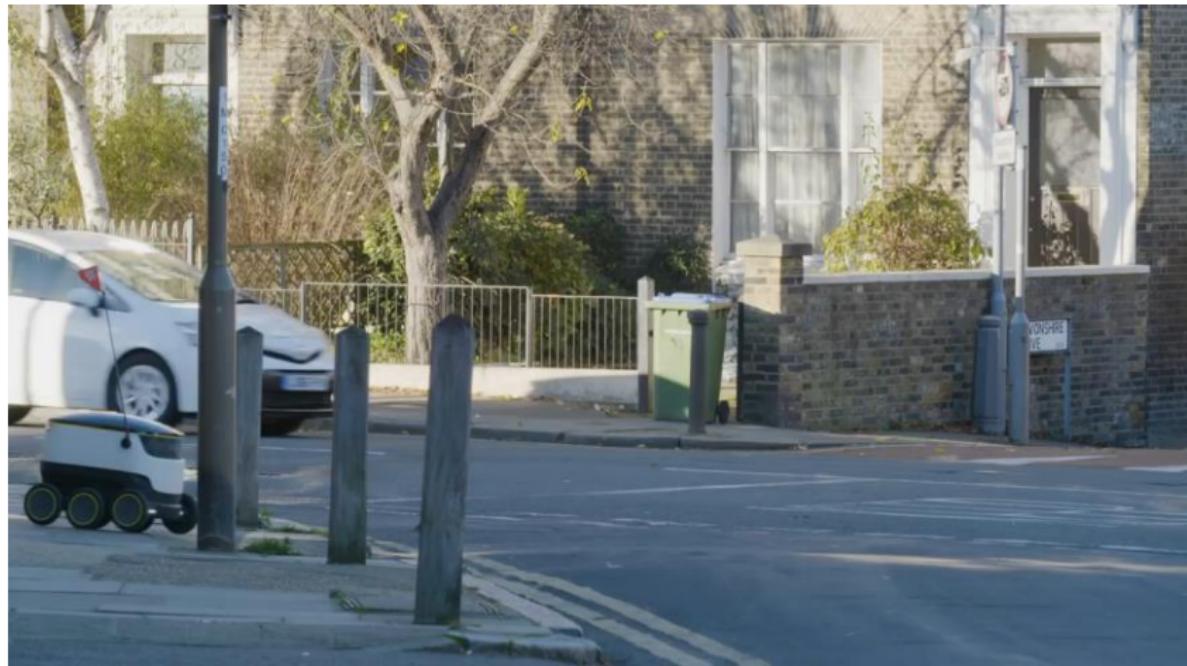
MIT Human-Centered Autonomous Vehicle



自动驾驶简介

L4级别无人驾驶-限定区域无人驾驶

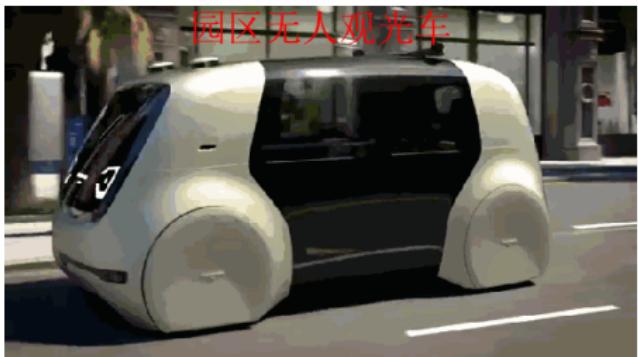
Starship: 目前国外很成功且商业化运营的送餐送货无人小车公司



自动驾驶简介



L4级别无人驾驶-限定区域无人驾驶



无人驾驶重卡



矿区无人驾驶



港口无人驾驶



自动驾驶简介

L4级别无人驾驶-限定区域无人驾驶

除了载人/载客之外？ 无人驾驶清扫车



自动驾驶简介



L4级别无人驾驶-限定区域无人驾驶

5G远程无人驾驶



车车通信/车路协同



汽车智能化与网联化
发展已成为当前趋势

自动驾驶简介



L5级别无人驾驶-完全无人驾驶

2018. 3. 15，谈到无人车何时能真正上路，李彦宏在受访时表示，之前工信部部长苗圩预言无人车上路还需要八到十年时间。而他本人的预期要更乐观一些，预计再有**三到五年**时间，在完全开放的道路上完全替代司机的无人驾驶车就会出现。

2020. 5. 21，全国政协委员、百度董事长李彦宏在接受中新社记者采访时表示，L5级别的自动驾驶估计还要**十年左右**的时间才能实现。他表示，目前来看，自动驾驶替代现有的汽车也需要一段较长的时间。

2020. 9，百度世界大会上，百度CEO李彦宏预测自动驾驶**5年后全面商用**。

2021. 5，百度第一季度财报电话会，作为自动驾驶最坚定的支持者李彦宏表示“我认为已经广为人知的是，自动驾驶在未来十年、二十年都不会真正做到成熟”。

自动驾驶简介

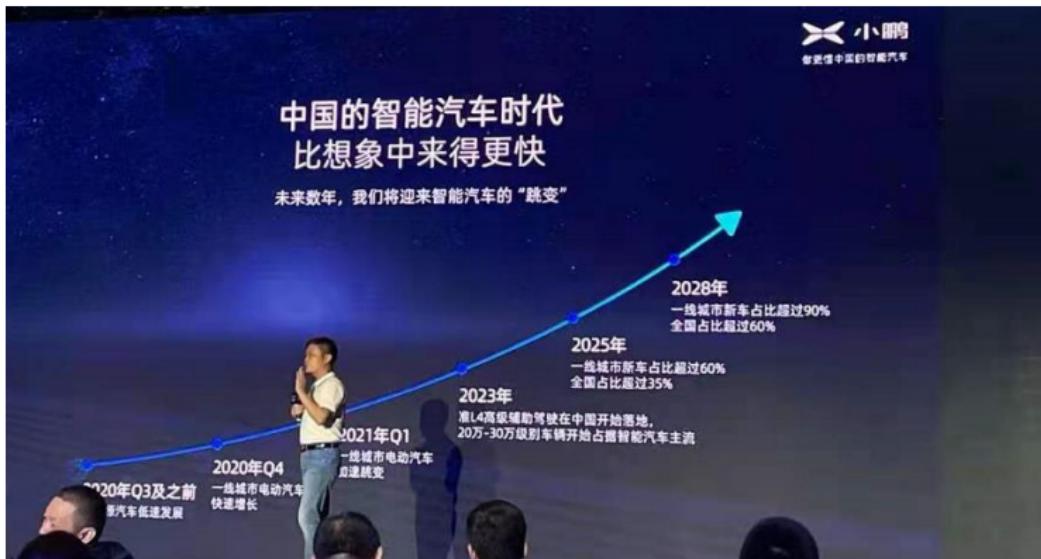
华为极狐



华为智能汽车解决方案BU总裁王军：“华为力争在**2025年实现**无人驾驶，真正颠覆性的技术是改变座舱”。

自动驾驶简介

小鹏/蔚来/理想



禾多倪凯：自动驾驶领域的“可量产”指的是在未来 5 - 10 年内，实现百万台级的交付，覆盖百万平方公里级的全场景数据回传。

自动驾驶简介



自动驾驶(关注车体智能)



智能座舱(关注人车交互)



城市大脑(关注智慧交通)



智慧出行(关注车路协同)



自动驾驶简介



自动驾驶始终难以破局的原因何在？

自动驾驶技术“长尾效应”突出

安全性要求极高

测试体系难建立

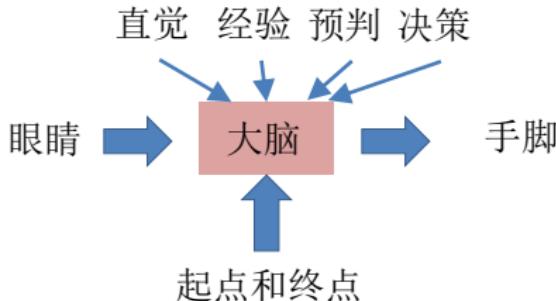
政策与法规的不完善

...



- ◆ 自动驾驶简介
- ◆ 自动驾驶软件架构
- ◆ 深度强化学习与自动驾驶应用
 - ◆ 视觉输入端到端控制
 - ◆ 基于深度强化学习的决策控制
- ◆ 总结

人类驾驶员开车过程

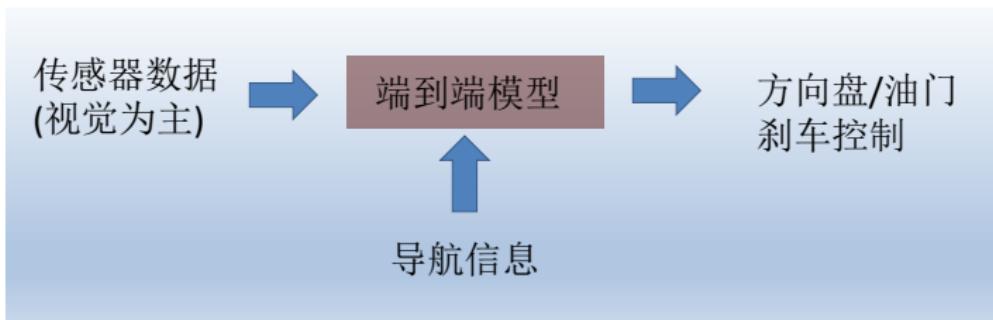
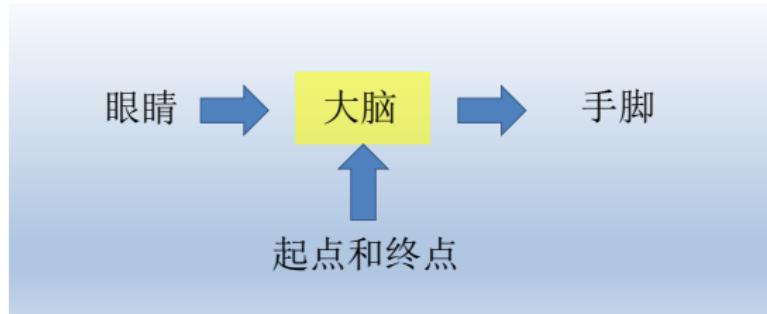


是否需要精确感知
前方车辆距离？

是否需要精确预测
周围车辆轨迹？

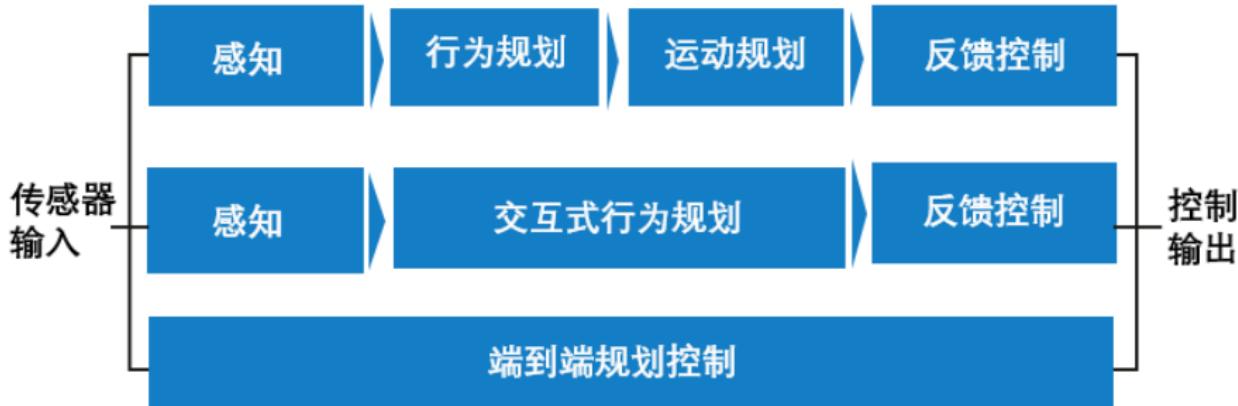
人类大脑具有超强
的认知/决策/应急处
理能力！

端到端架构



目前的端到端模型性能远不及人类大脑, 主要区别在**认知和决策能力**

交互式架构



自动驾驶是一个序列型决策，开车、转弯、变道、刹车等等都是序列型决策的动作，目前的认知与决策偏弱，可以借助深度强化学习来提高。

为解决长尾问题，自动驾驶仿真体系是真正实现无人驾驶的必由之路，也为深度强化学习训练提高了很好的仿真器。

不同场景泛化性，仿真与实际的迁移性，及DRL的效率、安全性等问题



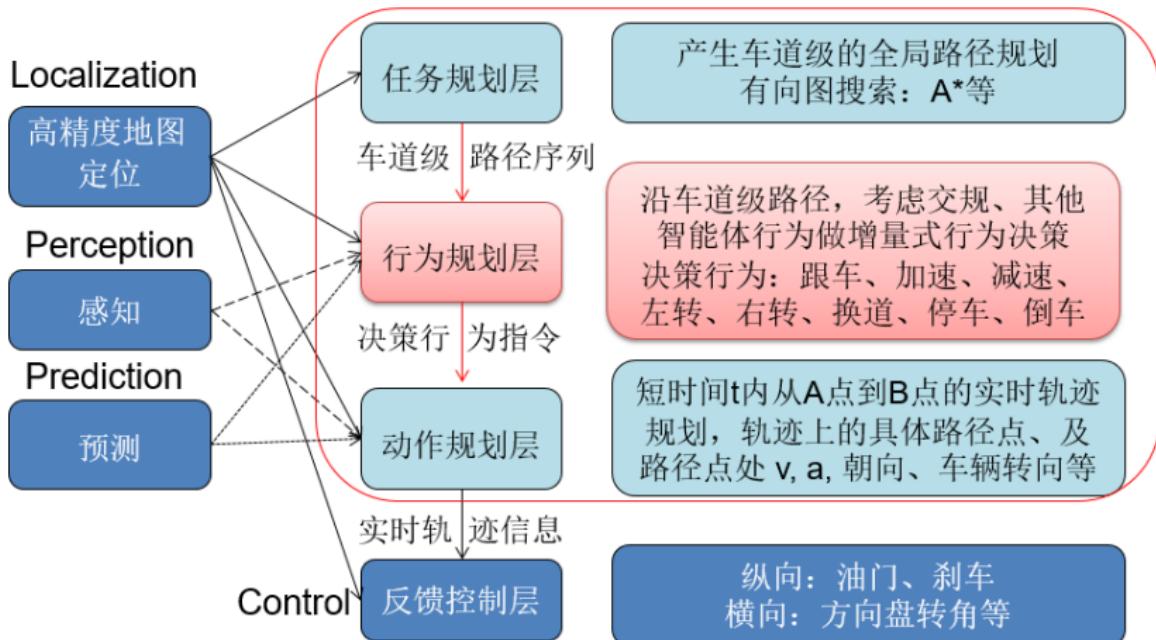
中国科学院自动化研究所

无人驾驶送餐

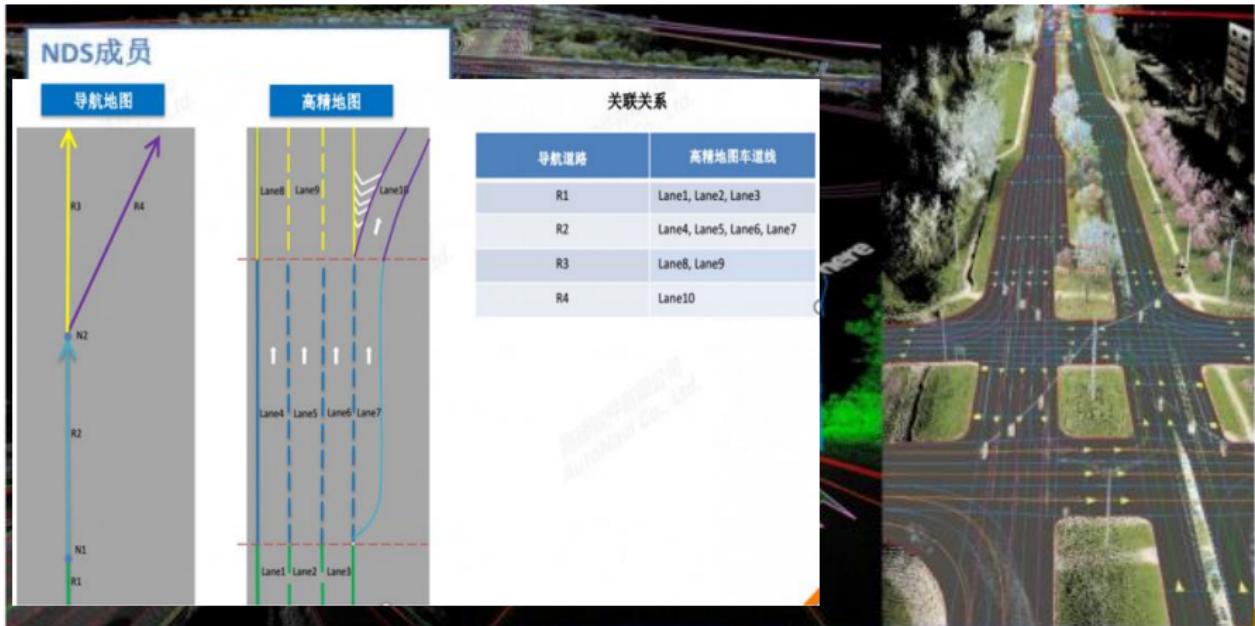
深度强化学习团队

高鸿飞，张启超，赵冬斌





高精度地图：导航/定位/感知/预测/规划/控制

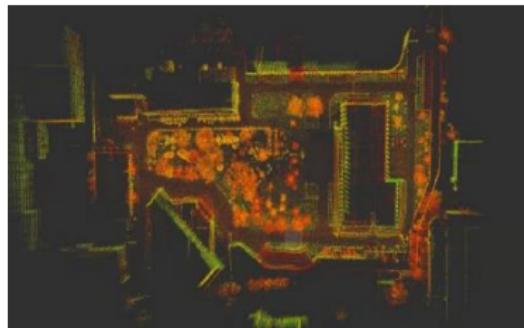


Routing: 基于高精地图输出车道级别的路径

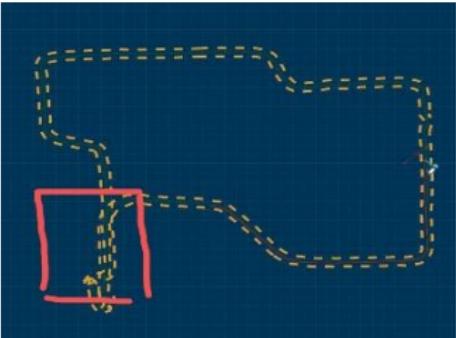
建图与定位



高精度地图：导航/定位/规划



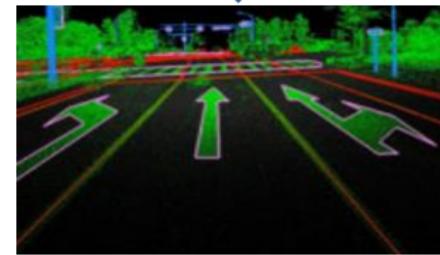
基于激光雷达建立点云底图



建立园区简易车道地图



OpenDrive格式的高精度地图



加入交通标志、减速带等图层信息

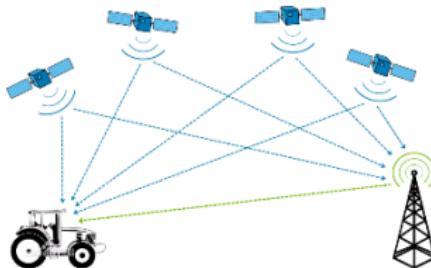
定位：厘米级定位，让车辆在高精度地图中找到自己的准确位置



IMU: 100Hz

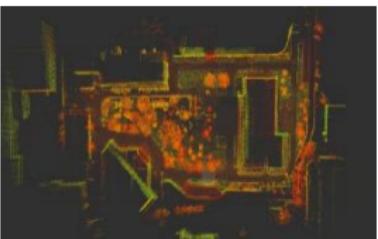


融合
定位



差分GPS数据: 10Hz

GPS, m级
RTK: 10cm级



对于园区环境，由于楼层遮挡，GPS
定位精度发生漂移

感知

利用多传感器融合技术**稳定实时**的检测和追踪移动障碍物、交通灯、车道线、可行驶区域、位置障碍物等，给出移动障碍物3D信息



感知

传感器设备

1. 多路单目相机

价格低廉

环境感知必备的传感器



根据焦距的不同又可分为短(8mm)
/中/长(25mm)距单目相机



感知模型的压缩与在嵌入式芯片的实时性是DL技术应用的关键

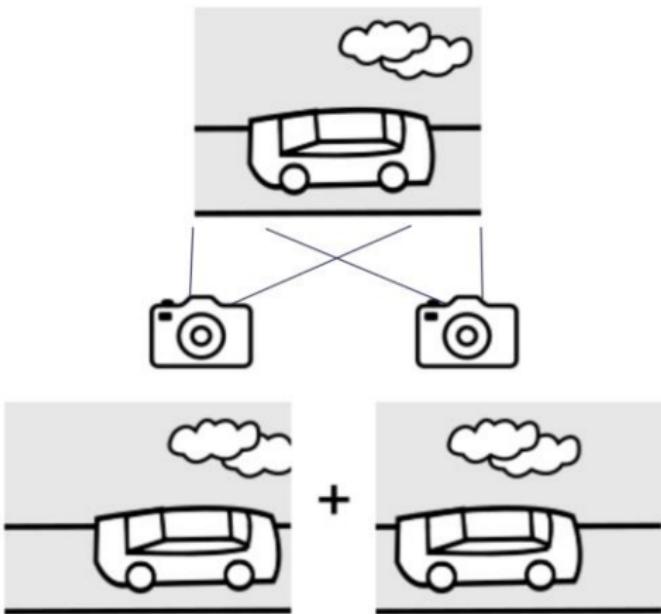
感知

2. 双目相机

可以有效估计深度信息
更准确的预测距离信息



60-70m, 1-2m测距误差



感知

3. 激光雷达 (Lidar)

单线-4线-16线-32线-64线-
128线，价格昂贵

激光多普勒效应，得到点云数据，点
云包含目标的姿态信息和反射强度

- 360度感知范围
- 可以准确测距离
- 测距范围与激光线数相关



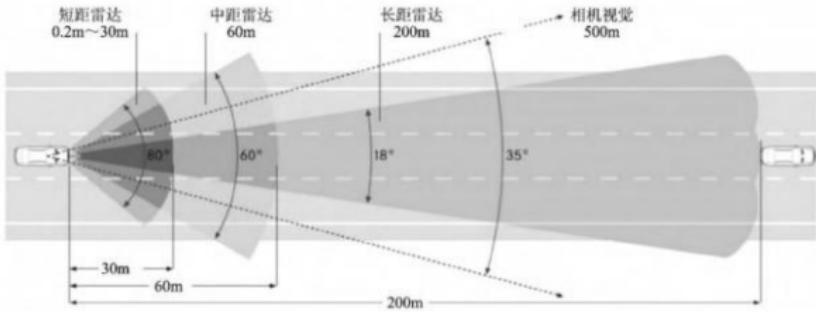
感知

4. 毫米波雷达 (Radar)

价格低廉

距离策略，速度估计精度高
环境感知必备传感器

宽视野，近距离
窄视野，远距离



感知

5. 超声波传感器

近距离检测障碍物
价格低廉
环境感知必备传感器



安装在无人车的四周



三类传感器功能比较

功能	摄像头	激光雷达	毫米波雷达
车道线检测	✓	✓	✗
路沿检测	✓	✓	✗
障碍物相对位置、距离检测	✓	✓	✓
障碍物运动状态判断	✓	✓	✓
障碍物识别、跟踪	✓	✓	✓
障碍物分类	✓	✗	✗
红绿灯、交通标志识别	✓	✗	✗
Slam地图创建及定位	✓	✓	✗

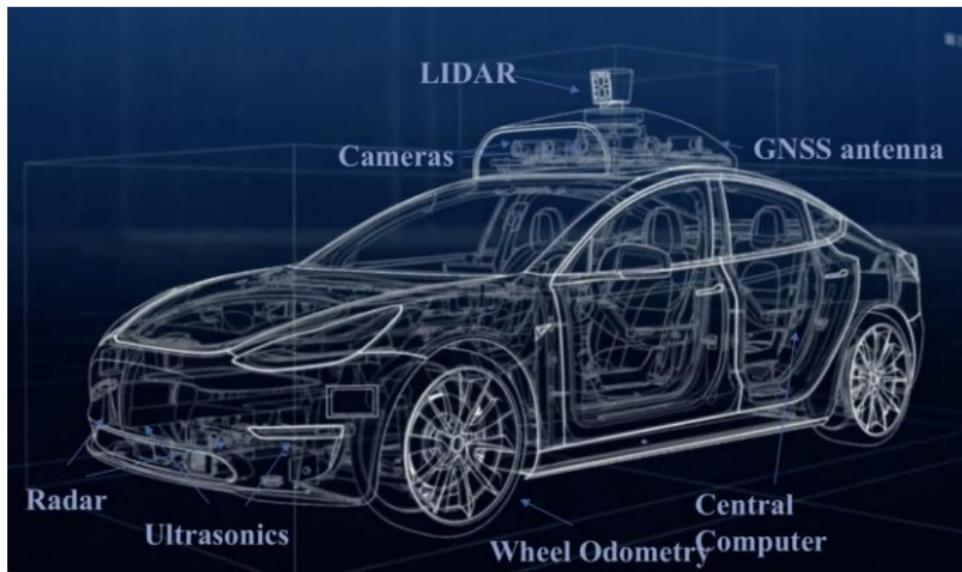
感知

三类传感器优缺点比较

传感器类型	特点	优点	缺点
摄像头	<ul style="list-style-type: none">像素级别的颜色、偏移、距离等信息	<ul style="list-style-type: none">成本低信息量丰富、特征识别好	<ul style="list-style-type: none">受环境光限制比较大速度、距离分辨率差
激光雷达	<ul style="list-style-type: none">3D扫描获取三维信息	<ul style="list-style-type: none">测距远、精确分辨率高、探测范围广	<ul style="list-style-type: none">成本高受雨雪雾烟尘影响较大
毫米波雷达	<ul style="list-style-type: none">电磁波TOF探距多普勒频移测速	<ul style="list-style-type: none">成本较低测速、测距精确不受光照影响，可穿透雾烟尘，全天候工作	<ul style="list-style-type: none">角度分辨率低障碍物特征识别差

感知

传感器设备-园区低速场景下的基本配置



1台激光雷达

1台毫米波雷达

3台摄像头

多个超声波

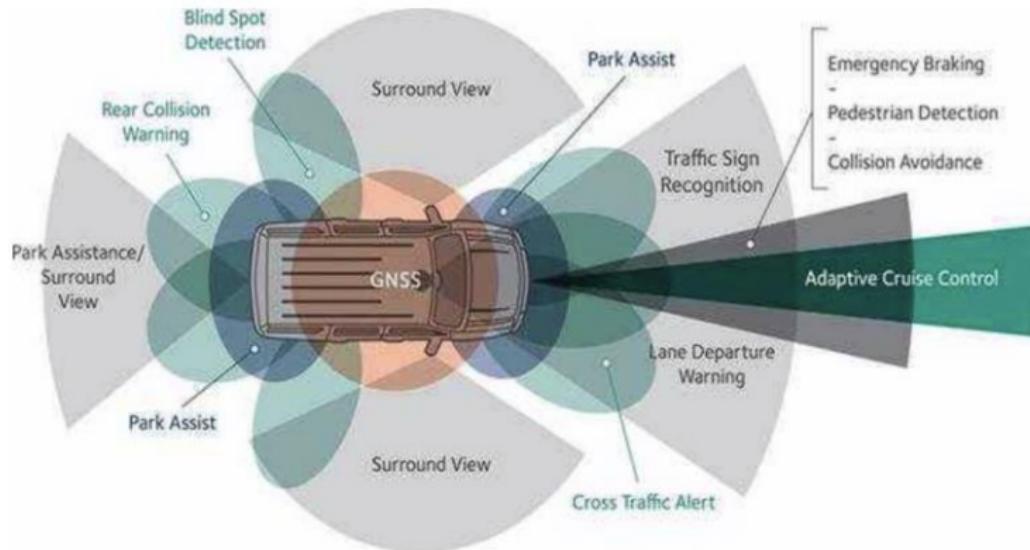
GSNN/组合惯导

轮速计

工控机

感知

传感器设备-复杂城市场景配置



2-3台激光雷达

8台毫米波雷达

6台摄像头

2个超声波

GSNN/组合惯导

轮速计

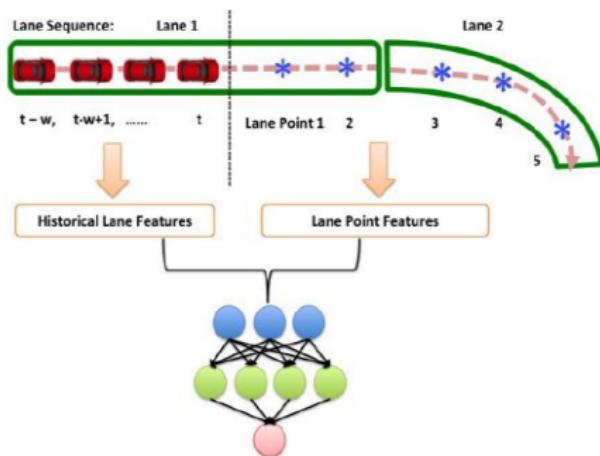
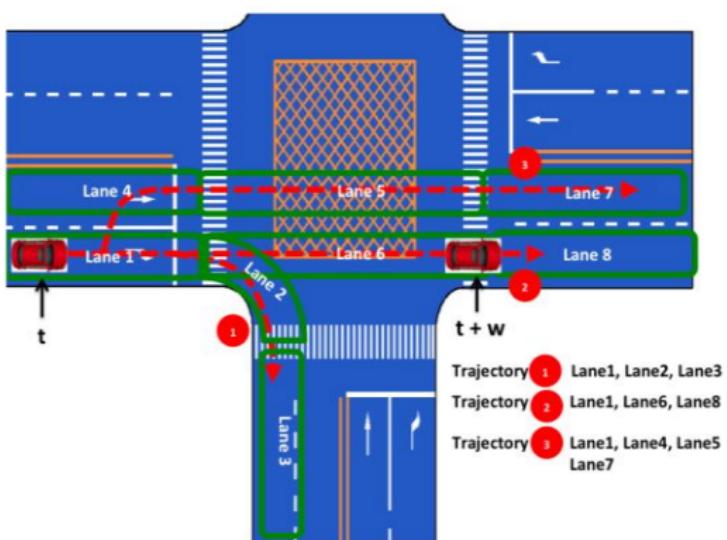
工控机

■ Long-Range Radar ■ Short/Medium Range Radar ■ LIDAR ■ Camera ■ Ultrasound ■ GNSS

预测

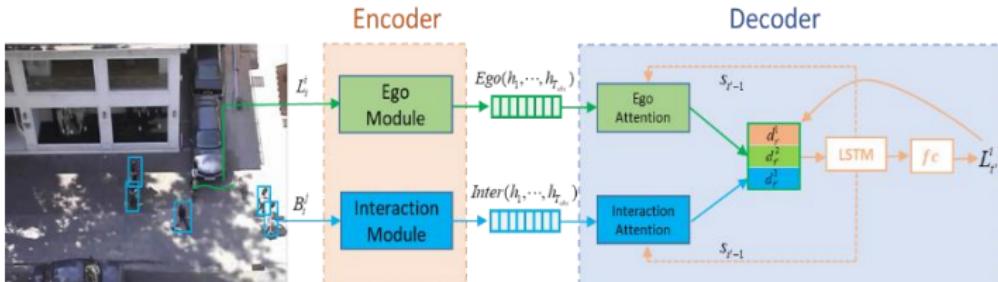
车辆轨迹预测

有交通规则和车道约束，可基于EKF与DL等方法结合来提高性能



预测

行人/自行车人轨迹预测



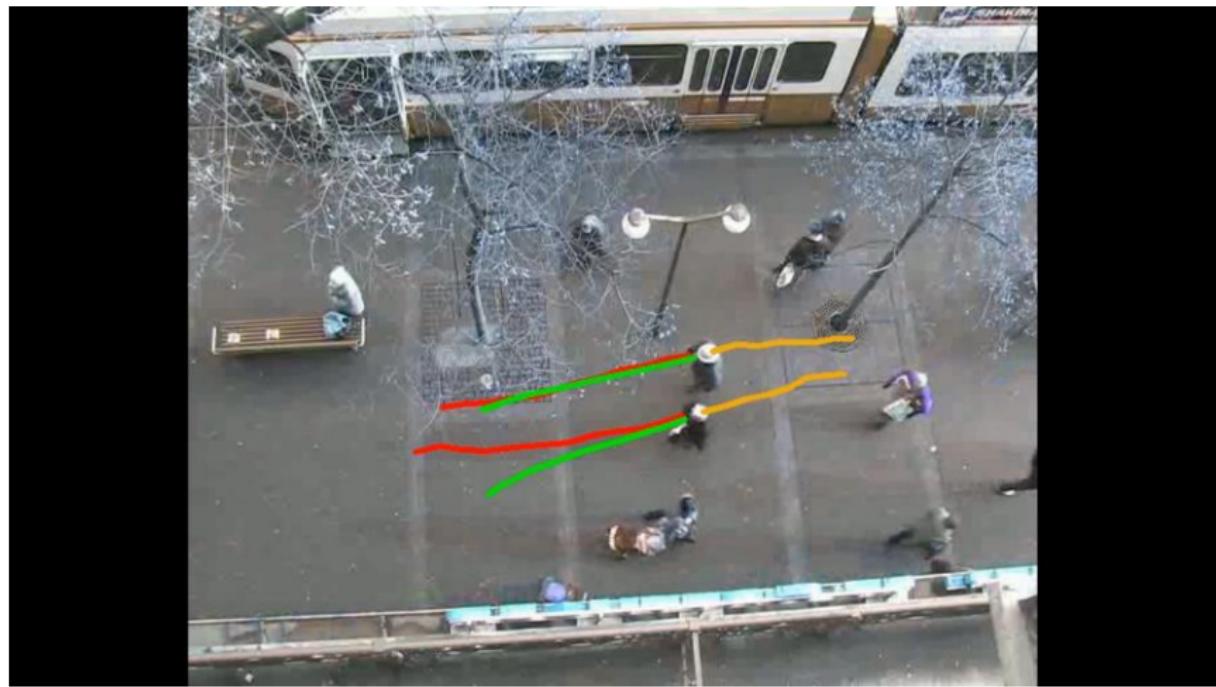
Rank	Method	WSADE	ADlw	ADlp	ADlb	WSADE	FDlw	FDlp	FDlb
1	TrafficPredict	8.5881	7.9467	7.1811	12.8805	24.2262	12.7757	11.1210	22.7912
2	sense_prediction	1.2387	2.1373	0.7102	1.8149	2.2697	3.7510	1.3658	3.3061
3	SimpleNet	1.2463	2.1041	0.7508	1.7726	2.2592	3.6582	1.4406	3.1457
4	GRIP	1.2524	2.0861	0.7142	1.8024	2.3440	3.9805	1.3732	3.4155
5	Panda	1.2671	2.2881	0.7195	1.8373	2.3738	4.0290	1.3773	3.4964
6	test_method1	1.2907	2.2371	0.7549	1.8431	2.4072	4.0353	1.4376	3.4833
7	test_ck	1.2907	2.2632	0.7314	1.8812	2.4045	4.0878	1.3894	3.5503
8	Scene_fusion	1.2988	2.2340	0.7617	1.8648	2.3870	3.9683	1.4527	3.4125
9	Scene_sum	1.3037	2.2262	0.7760	1.8543	2.4049	3.9735	1.4865	3.4000
10	vian2	1.3080	2.3242	0.7814	1.7726	2.3499	4.0916	1.4475	3.1457
11	SLUtm	1.3104	2.2721	0.7684	1.8651	2.4112	4.0437	1.4691	3.4107

http://apolloscape.auto/leader_board.html

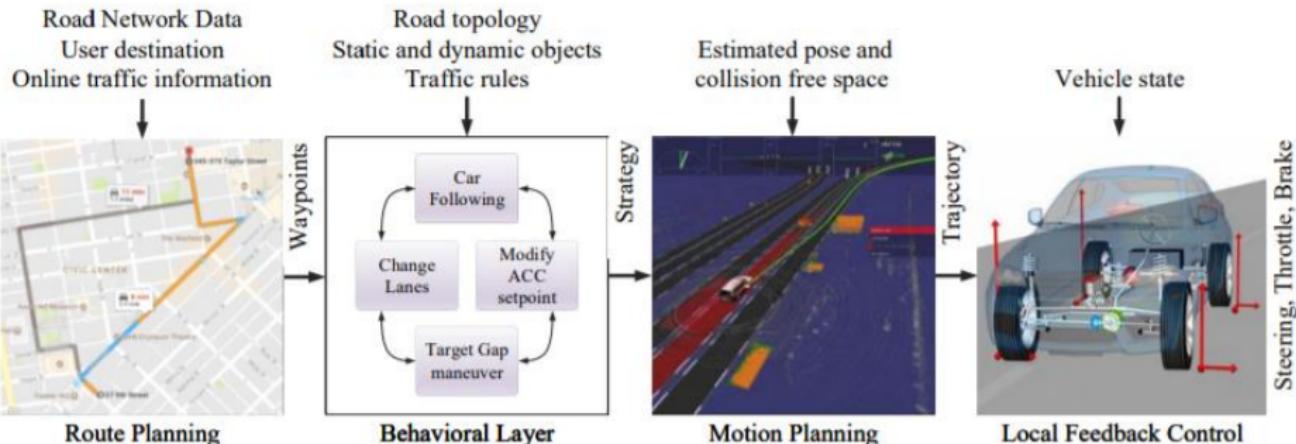
行人和自行车人由于没有车道约束，行进方向不确定性强，成为目前预测中的难点，而人流场景也恰好是目前低速物流车最常见的场景。

预测

行人/自行车人轨迹预测



决策



决策层输入：低维状态、栅格图、local map图等

决策层输出：行为规划层输出离散高层动作
 运动规划层的轨迹目标点
 控制层的连续/离散控制动作(方向盘/油门刹车)

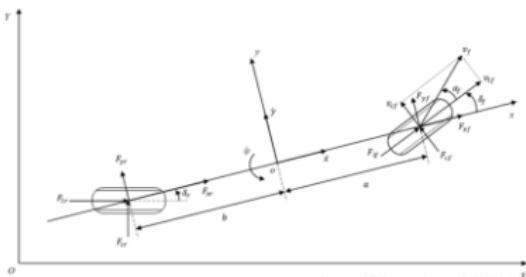
决策



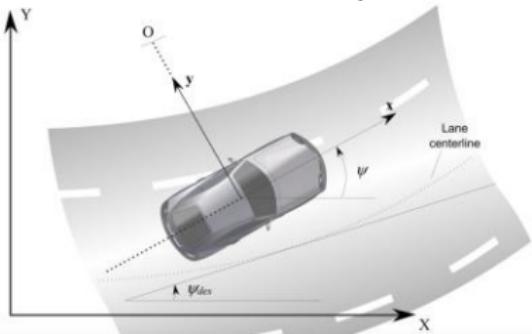
控制

横向控制：通过控制方向盘使得车辆跟踪期望轨迹

横向动力学模型--简化版的自行车模型



$$e = Ae + Bu + C\rho$$

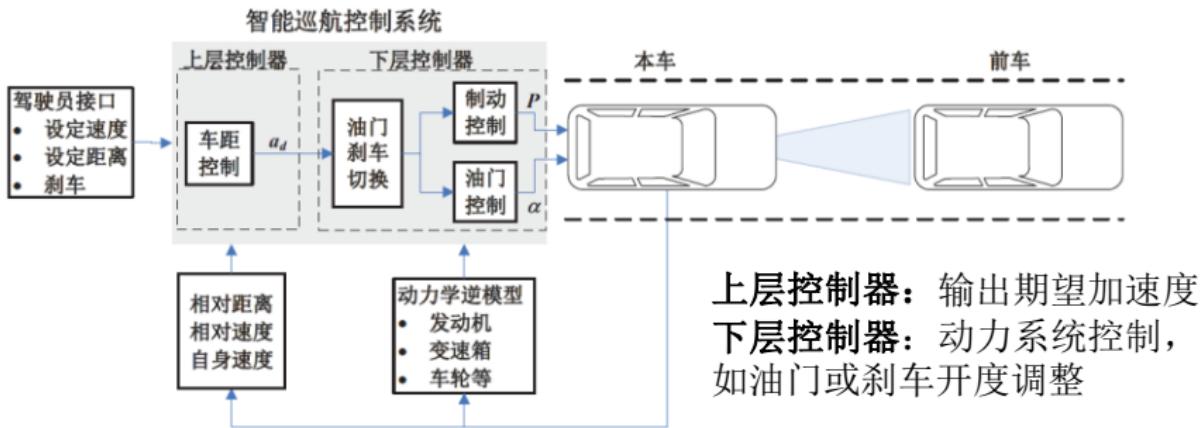
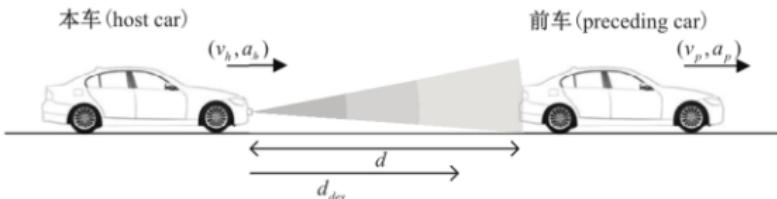


状态分别为：横向误差，横向误差率，航向误差，航向误差率

可以利用PID、LQR、MPC等方法来完成横向控制任务，计算出期望方向盘转角

控制

纵向控制：通过测量两车间距与速度，调整车距为驾驶员期望距离



控制

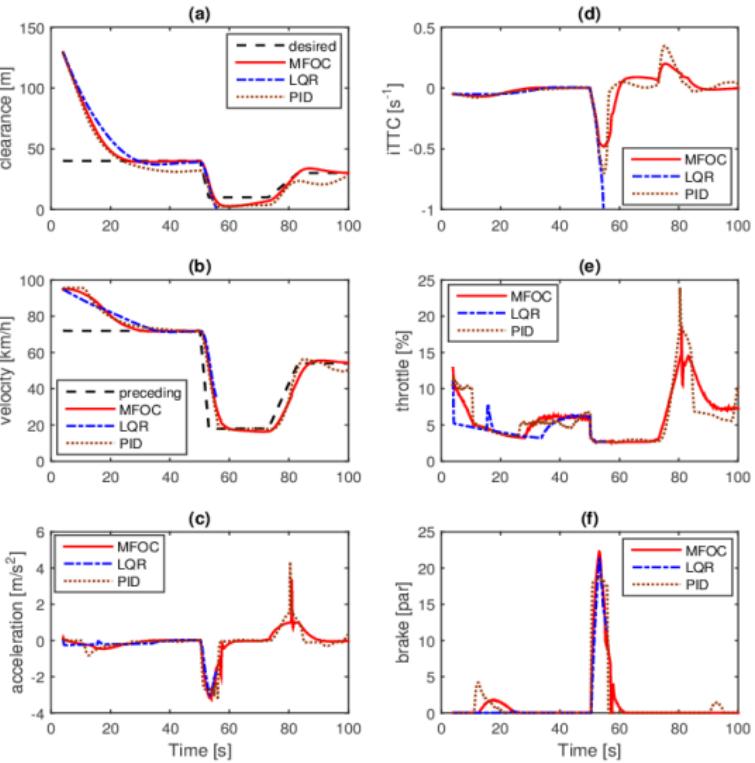
仿真测试平台：Dspace



控制

紧急刹车场景

时间	前车状态
[0, 50)s	72km/h
(50, 53)s	$-5m/s^2$
(53, 78)s	18km/h
(78, 83)s	$2m/s^2$
(83, 100)s	54km/h



控制效果

- MFOC控制平缓精确
- PID存在控制误差
- LQR发生碰撞

汽车状态变化曲线

- ◆ 自动驾驶简介
- ◆ 自动驾驶软件架构
- ◆ 深度强化学习与自动驾驶应用
 - ◆ 视觉输入端到端控制
 - ◆ 基于深度强化学习的决策控制
- ◆ 总结

视觉输入端到端控制

Comma One--利用VAE和GAN数据增广，基于真实数据做预测任务

状态 $X_t = \{x_{t-n+1}, x_{t-n+2}, \dots, x_t\}$ 历史图像序列

动作： $S_t = \{s_{t-n+1}, s_{t-n+2}, \dots, s_t\}$ 本车车速

$A_t = \{a_{t-n+1}, a_{t-n+2}, \dots, a_t\}$ 驾驶员给的方向盘转角

目标： $x_{t+1} = F(X_t, S_t, A_t)$ 下一时刻的图像

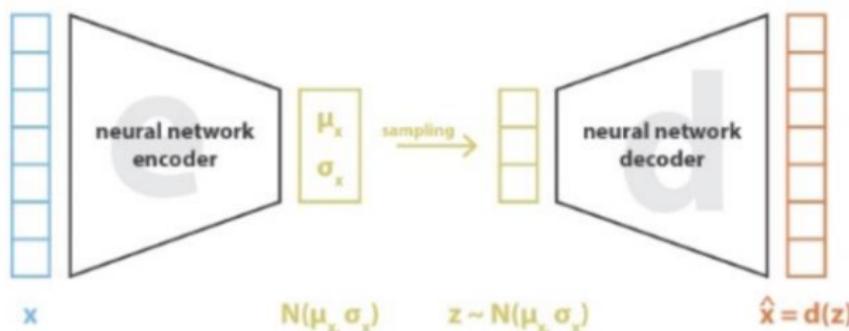
利用RNN构建一个预测模型F，实现基于历史数据预测未来



Eder Santana, George Hotz, Learn a driving simulator, 2016

视觉输入端到端控制

变分自编码器(VAE, Variational autoencoder)



$$\text{loss} = \|x - \hat{x}\|^2 - \text{KL}[N(\mu_x, \sigma_x), N(0, I)] = \|x - d(z)\|^2 - \text{KL}[N(\mu_x, \sigma_x), N(0, I)]$$

通过将输入编码到潜空间的一个分布上(一般选择高斯分布)，可以减缓自动编码器的过拟合问题，使得潜空间到输出的解码性能更好

视觉输入端到端控制

VAE效果

解码器输出 目标图像 解码器输出 目标图像

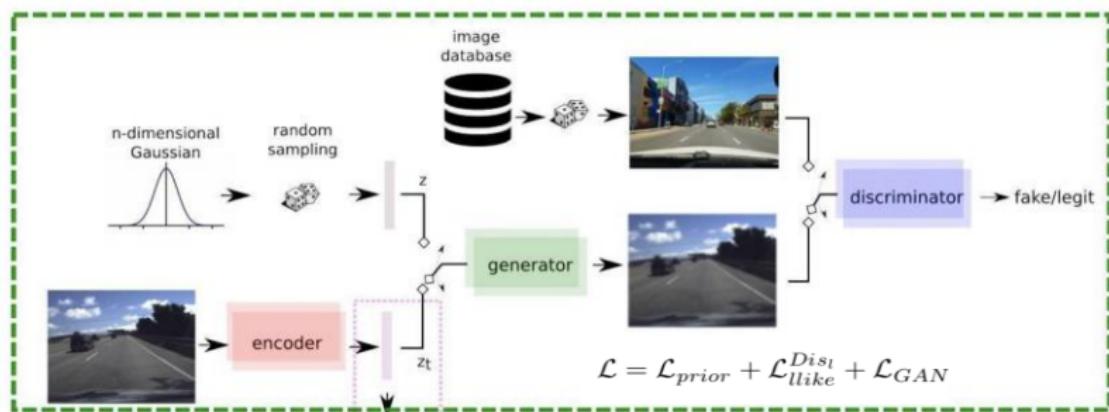


单纯使用VAE
训练效果不
佳，图片看
起来很模糊

潜空间变量
服从高斯分
布的假设不
完全符合

视觉输入端到端控制

VAE+GAN



$$\begin{array}{ccc}
 \vdots & \vdots & \\
 \tilde{z}_{t+1} & \xrightarrow{\hspace{1cm}} & \text{VAE损} \\
 & \downarrow & \text{失函数} \\
 \tilde{z}_{t+n} & \xrightarrow{\hspace{1cm}} & \text{判别网络I} \\
 & & \text{层损失函数} \\
 & & \\
 \mathcal{L}_{prior} = D_{KL}(q(z|x) || p(z)) & & \mathcal{L}_{llike}^{Dis_l} = \mathbb{E} [(y_l - \tilde{y}_l)^2]
 \end{array}$$

GAN网络
损失函数

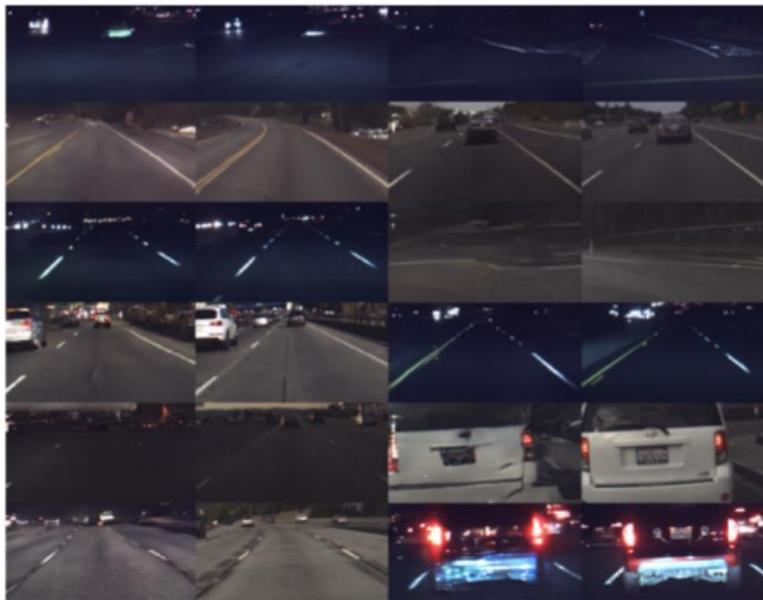
$$\mathcal{L}_{GAN}^{Gen} = \log(Dis(Dis(u))) + \log(Dis(Dis(Enc(x))))$$

$$\mathcal{L}_{GAN}^{Dis} = \log(Dis(x)) + \log(1 - Dis(Dis(u))) + \log(1 - Dis(Dis(Enc(x))))$$

视觉输入端到端控制

VAE+GAN效果

解码器输出 目标图像 解码器输出 目标图像



得到一个性能非常好的编码器

$$x_t \mapsto z_t$$

可以得到当前状态到潜空间分布

RNN预测模型

Encoder: $x_t \mapsto z_t$

RNN模型: $z_t, h_t, c_t \mapsto z_{t+1}$

$$h_{t+1} = \tanh(W h_t + V z_t + U c_t)$$

$$\tilde{z}_{t+1} = A h_{t+1}$$

h_t 为RNN网络的隐层状态

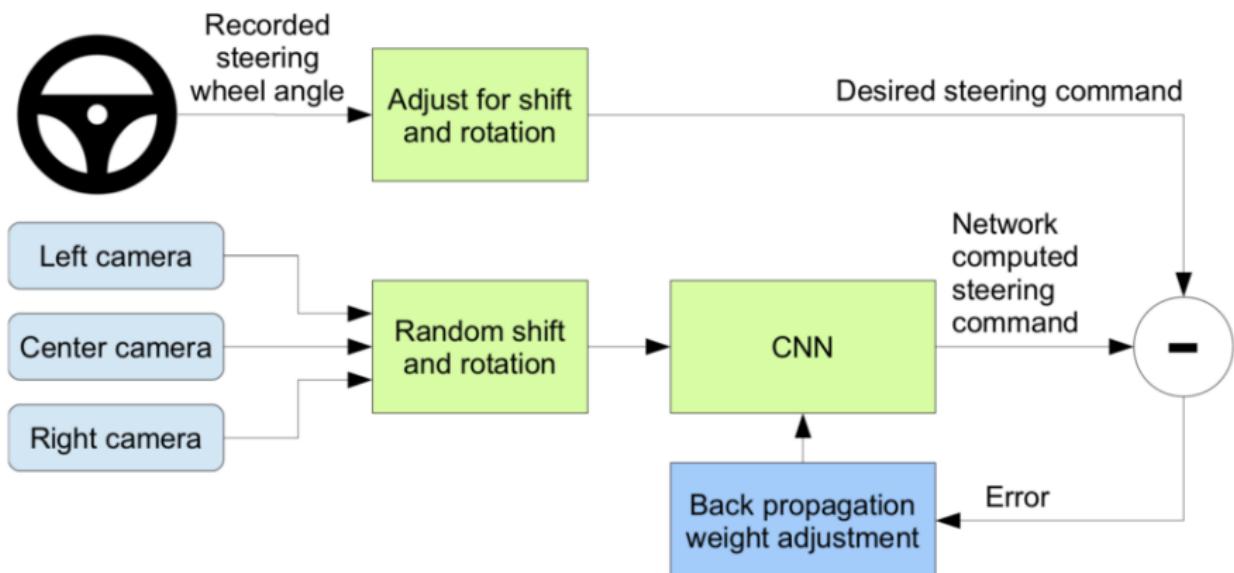
c_t 为速度和方向盘角度信息

损失函数为: $\mathcal{L}_{RNN} = \mathbb{E} [(z_{t+1} - \tilde{z}_{t+1})^2]$

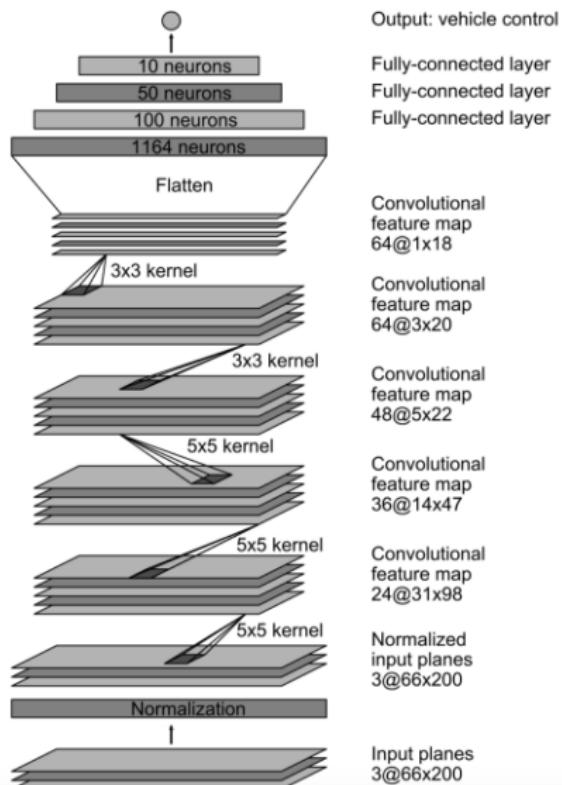
Decoder: $\tilde{z}_{t+1} \mapsto x_{t+1}$

视觉输入端到端控制

NVIDIA--利用输入数据源增广改进行为克隆方法



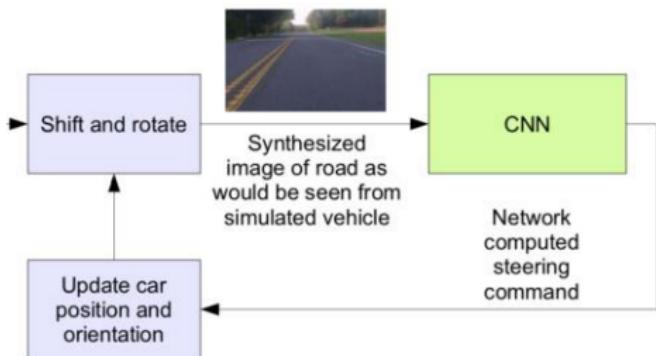
视觉输入端到端控制



数据训练
CNN
模型

仿真测试
CNN
模型

实车验证



视觉输入端到端控制

如果存在路口的情况怎么办？



有其他车辆的城市路况怎么办？



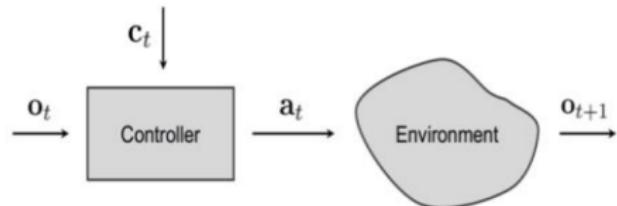
前面的两类方法都很难处理，只是实现了基于高维输入的车道保持任务，而车道保持仅是L1-L2级自动驾驶。

视觉输入端到端控制

条件模仿学习(Conditional Imitation Learning)

如果存在路口的情况怎么办？

存在一个高层指令(直行/左转/右转), 可以来自驾驶员意图, 可以来自规划模块的行为决策指令等, 如何将该指令结合入网络?



o_t 为当前观测状态
 c_t 为高层决策指令

条件模仿学习(Conditional Imitation Learning)

已知数据集: $\mathcal{D} = \langle o_i, c_i, a_i \rangle, i=1, \dots, N, \quad o = \langle i, m \rangle$

训练目标: $\underset{\theta}{\text{minimize}} \sum_i \ell(F(\mathbf{o}_i, \mathbf{c}_i; \theta), \mathbf{a}_i)$

i 为图像数据

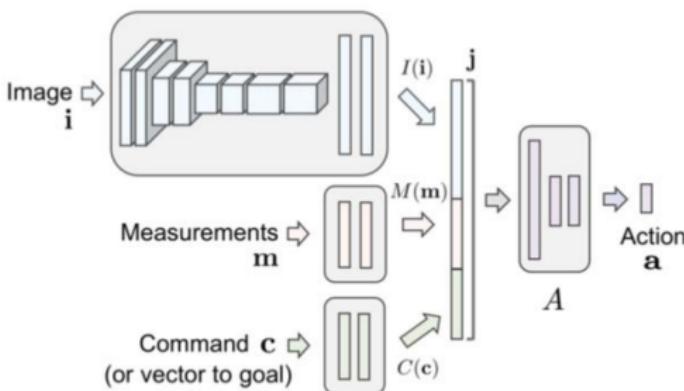
m 为低维测量数据, 如速度/位置等信息

c 指令输入时表示为独热编码向量

a 为方向盘转角和加速度, 横向和纵向都进行控制

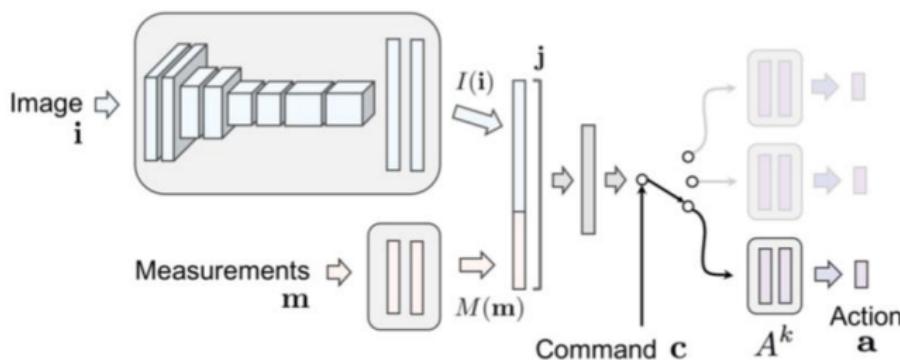
条件模仿学习(Conditional Imitation Learning)

直观想法:

实际效果不佳，网络还是很难考虑指令 c 带来的影响

条件模仿学习(Conditional Imitation Learning)

网络结构设计:

输入图像参考的
NVIDIA的三输入

考虑到高层决策指令数量有限，对每一个指令设计一个输出分支
branch，可以对每一个分支学习其策略，取得了很好的效果

条件模仿学习(Conditional Imitation Learning)

End-to-end Driving via Conditional Imitation Learning

Felipe Codevilla, Antonio López - Computer Vision Center (CVC)
Matthias Müller - King Abdullah University of Science and Technology (KAUST)
Vladlen Koltun, Alexey Dosovitskiy - Intel Visual Computing Lab

We propose conditional imitation learning which allows an autonomous vehicle trained end-to-end to be directed by high-level commands.

Experiments in simulation and on a physical vehicle show that the method allows for goal-directed navigation guided by a topological planner or a user.



Our first finding is that even with 30 million examples, and even with mid-level input and output representations that remove the burden of perception and control, pure imitation learning is not sufficient. As an example, we found that this model would get stuck or collide with another vehicle parked on the side of a narrow street, when a nudging and passing behavior was viable. The key challenge is that we need to run the system closed-loop, where errors accumulate and induce a shift from the training distribution (Ross et al.

仅仅依靠传统的模仿学习是行不通的，特别是窄道通行等场景

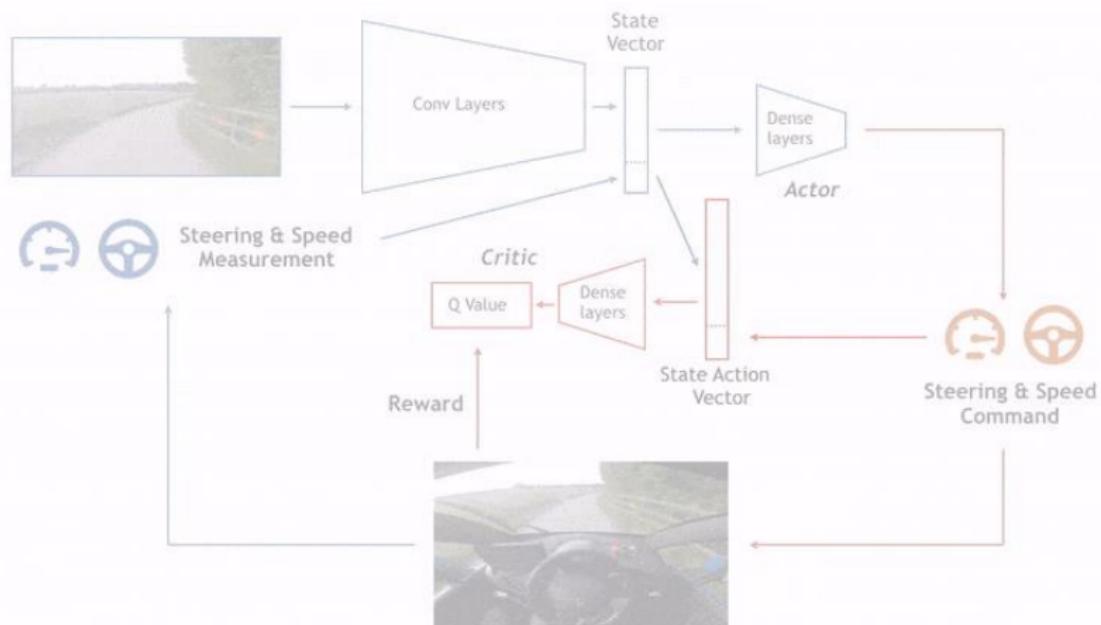
and data augmentation, first in simulation. We then show how our final model successfully drives a car in the real world and is able to negotiate situations involving other agents, turns, stop signs, and traffic lights. Finally, it is important to note that there are highly interactive situations such as merging which may require a significant degree of exploration within a reinforcement learning (RL) framework. This will demand simulating other (human) traffic participants, a rich area of ongoing research. Our contribution can be viewed as pushing the boundaries of what you can do with purely offline data and no RL.

强交互的场景如汇车/换道等，需要依赖强化学习的框架

Waymo & Google brain, ChauffeurNet: Learning to Drive by Imitating the Best and Synthesizing the Worst, 2018.

基于深度强化学习的决策控制

- DDPG 无需高精度地图，无需手写规则



Wayve, Learning to drive in a day, 2018

基于深度强化学习的决策控制



- DDPG



在仿真器中选择折扣因子、学习率、噪声等

奖赏信号的设置：

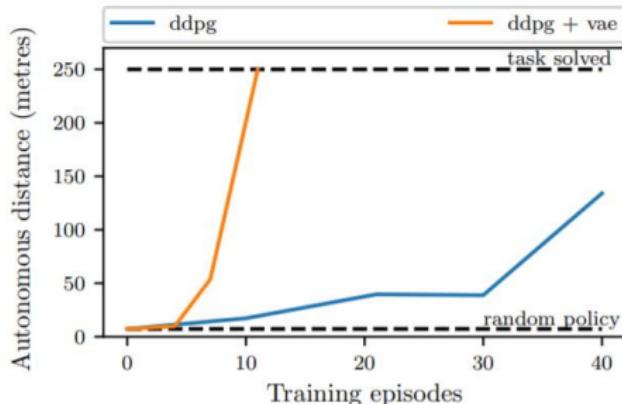
r =安全行驶距离

状态：驾驶员视角图像输入

动作：连续方向盘&油门控制量
 $-1 \leq \gamma \leq 1, 0 \leq a \leq 1$

基于深度强化学习的决策控制

DDPG+VAE



(a) Algorithm results

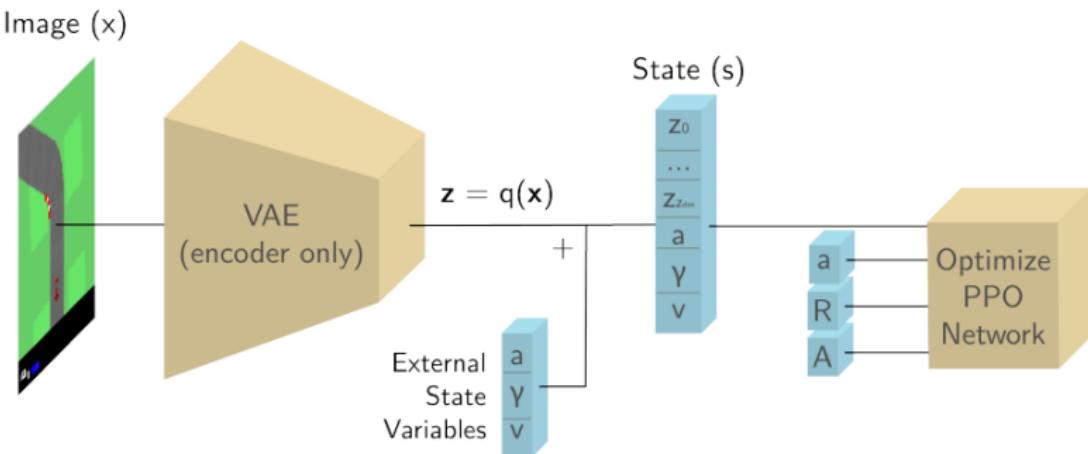


(b) Route

Model	Training			Test	
	Episodes	Distance	Time	Meters per Disengagement	# Disengagements
Random Policy	-	-	-	7.35	34
Zero Policy	-	-	-	22.7	11
Deep RL from Pixels	35	298.8 m	37 min	143.2	1
Deep RL from VAE	11	195.5 m	15 min	-	0

基于深度强化学习的决策控制

PPO+VAE



加速度
方向盘转角
速度

状态：驾驶员视角图像输入

动作：连续方向盘&油门控制量
 $-1 \leq \gamma \leq 1, 0 \leq a \leq 1$

基于深度强化学习的决策控制

PPO+VAE

1: 与速度成正比，超速终止

$$r(v) = \begin{cases} 0 & \text{if } v > v_{target} \text{ or on infraction} \\ v & \text{otherwise} \end{cases}$$

2: 与速度成正比，超速非终止

$$r(v) = \begin{cases} -10 & \text{on infraction} \\ v_{norm} & v_{norm} \leq 1 \\ (2 - v_{norm}) & v_{norm} > 1 \end{cases}$$

3: 在2基础上保持在车道中心

$$r(v, d) = \begin{cases} -10 & \text{on infraction} \\ v_{norm} + (1 - d_{norm}) & v_{norm} \leq 1 \\ (2 - v_{norm}) + (1 - d_{norm}) & v_{norm} > 1 \end{cases}$$

4: 在3基础上接近目标速度留余地

$$r(v, d) = \begin{cases} -10 & \text{on infraction} \\ \frac{v}{v_{min}} + (1 - d_{norm}) & v < v_{min} \\ 1 + (1 - d_{norm}) & v_{min} \leq v < v_{target} \\ (1 - \frac{v - v_{target}}{v_{max} - v_{target}}) + (1 - d_{norm}) & v \geq v_{target} \end{cases}$$

基于深度强化学习的决策控制

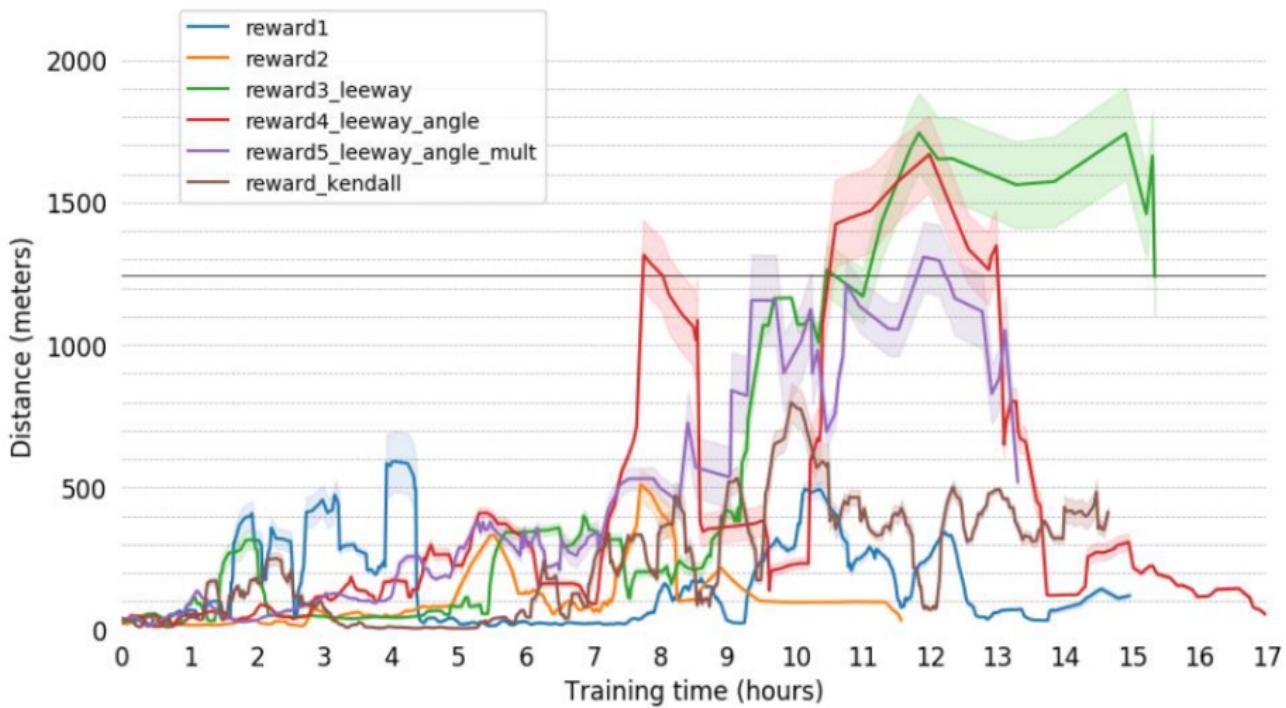
5：在4基础上加入偏航角信息

$$r(v, d) = \begin{cases} -10 & \text{on infraction} \\ \frac{v}{v_{min}} + (1 - d_{norm}) + \alpha_{rew} & v < v_{min} \\ 1 + (1 - d_{norm}) + \alpha_{rew} & v_{min} \leq v < v_{target} \\ (1 - \frac{v - v_{target}}{v_{max} - v_{target}}) + (1 - d_{norm}) + \alpha_{rew} & v \geq v_{target} \end{cases}$$

6：相乘积的行驶

$$r(v, d) = \begin{cases} -10 & \text{on infraction} \\ \frac{v}{v_{min}} * (1 - d_{norm}) * \alpha_{rew} & v < v_{min} \\ 1 * (1 - d_{norm}) * \alpha_{rew} & v_{min} \leq v < v_{target} \\ (1 - \frac{v - v_{target}}{v_{max} - v_{target}}) * (1 - d_{norm}) * \alpha_{rew} & v \geq v_{target} \end{cases}$$

基于深度强化学习的决策控制



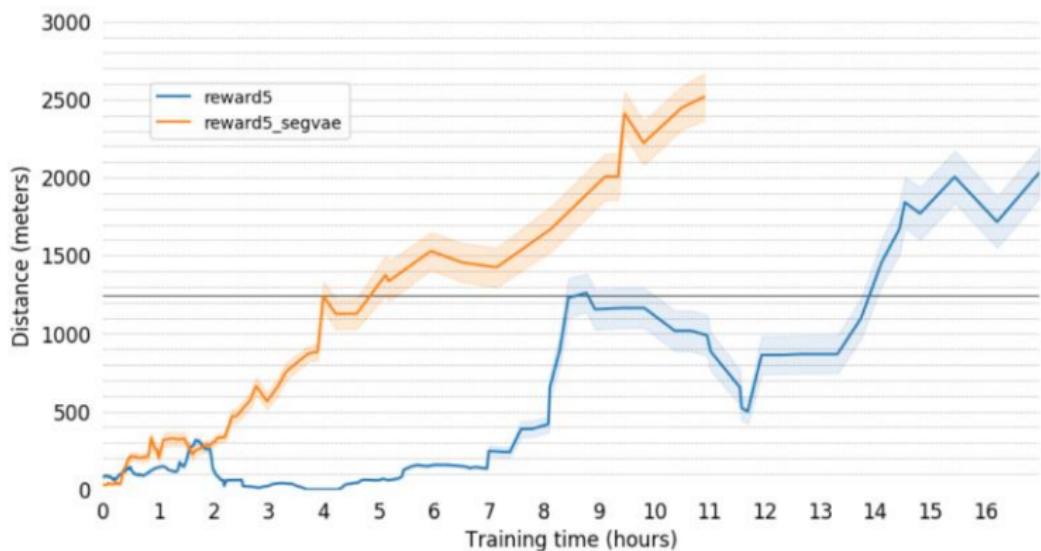
基于深度强化学习的决策控制



(a)



(b)

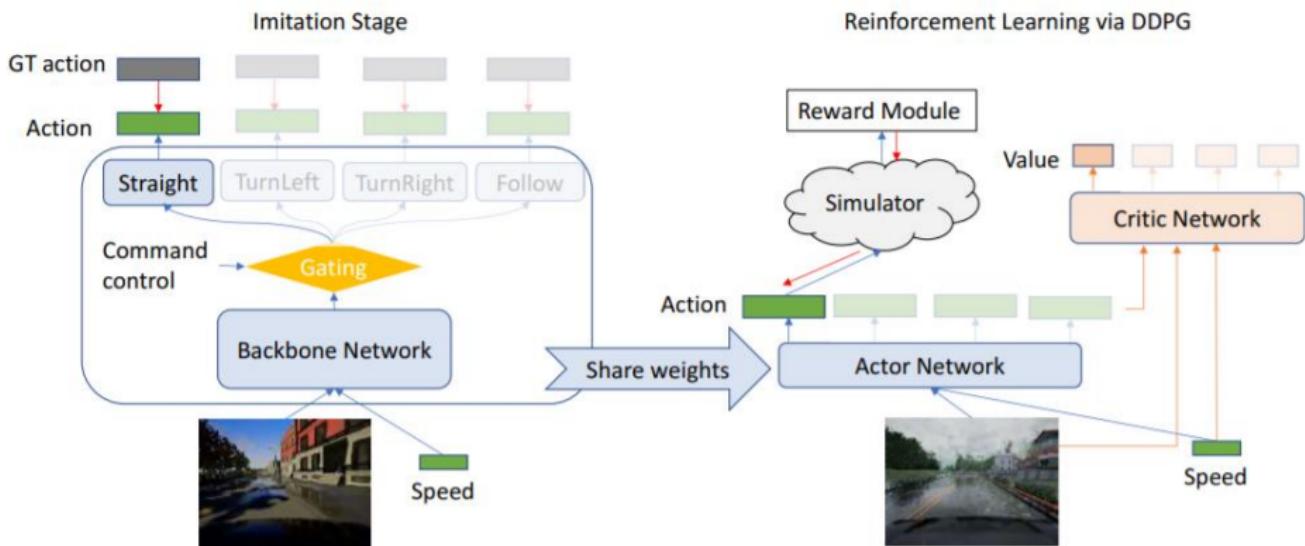


基于深度强化学习的决策控制



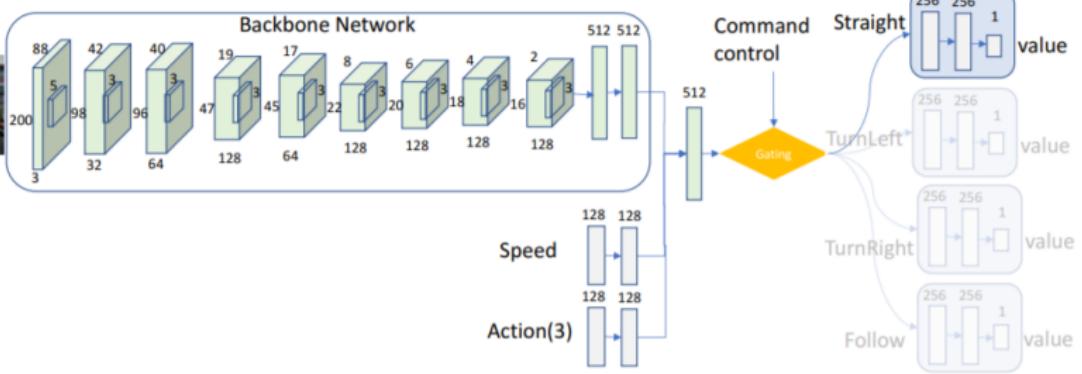
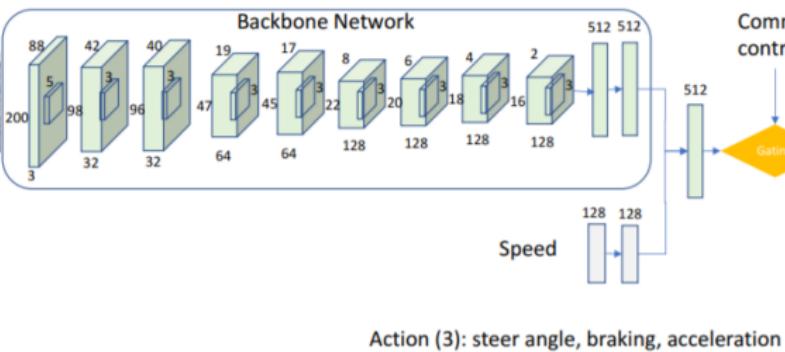
基于深度强化学习的决策控制

条件依赖的DDPG



基于深度强化学习的决策控制

条件依赖的DDPG



基于深度强化学习的决策控制

条件依赖的DDPG

Straight



One-turn



Navigation

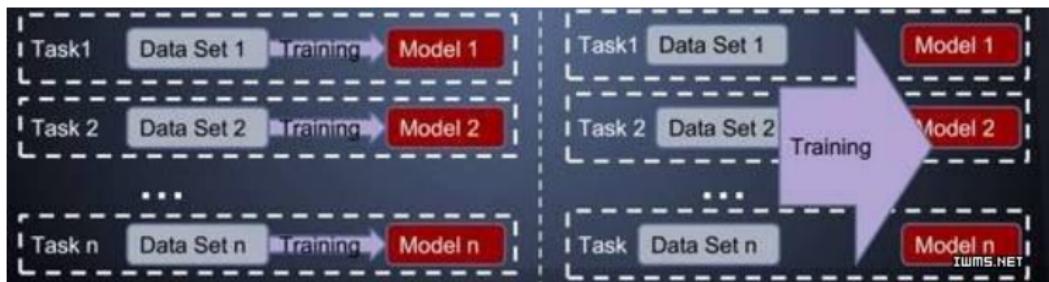


Navigation dynamic



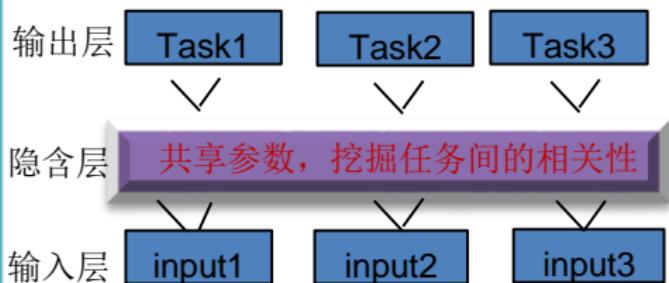
Task	Training conditions				New town				New weather				New town/weather			
	MP	IL	RL	CIRL	MP	IL	RL	CIRL	MP	IL	RL	CIRL	MP	IL	RL	CIRL
Straight	98	95	89	98	92	97	74	100	100	98	86	100	50	80	68	98
One turn	82	89	34	97	61	59	12	71	95	90	16	94	50	48	20	82
Navigation	80	86	14	93	24	40	3	53	94	84	2	86	47	44	6	68
Nav. dynamic	77	83	7	82	24	38	2	41	89	82	2	80	44	42	4	62

多任务学习



单任务学习

多任务学习

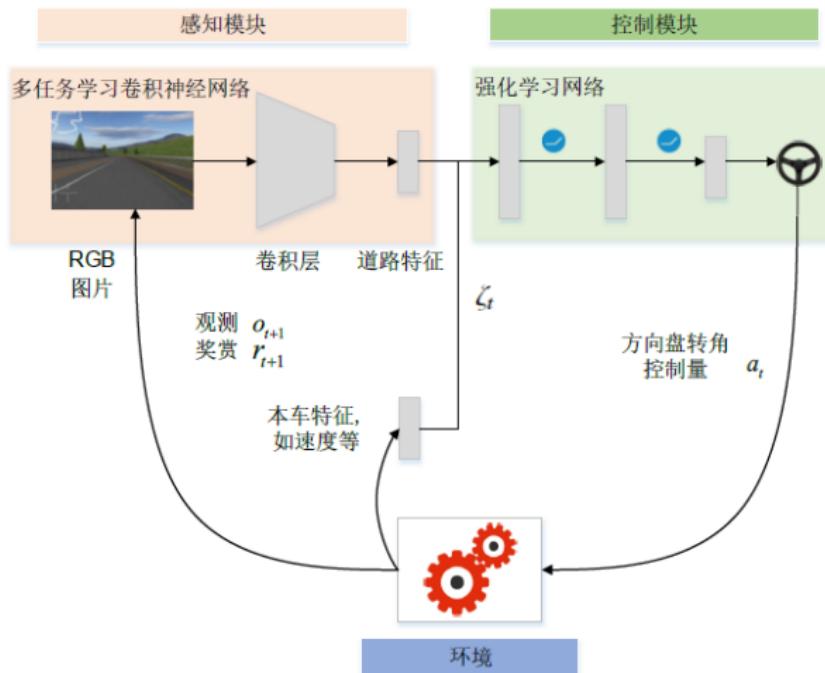


多任务学习的损失函数为：

$$L = \sum_{i=1}^N \alpha_i L_i$$

通过联合学习多个相关学习任务，挖掘任务间的共享特征，防止模型过拟合。

智能驾驶视觉横向控制框架



系统主要包含3个模块：

- 多任务学习(MTL)感知模块
- 强化学习(RL)决策模块
- 智能驾驶模拟器

□ 感知模块：

$$\zeta_t = \phi(o_t; \theta_p)$$

□ 控制模块

$$a_t = \phi(s_t; \theta_\mu)$$

多任务学习环境感知模块

强化学习控制模块依赖感知模块对于道路关键特征的预测精度，当预测不准时车辆在运行过程中会不断出现**转向角抖动**现象。尤其在弯道处很可能对车辆失去控制，驶出车道。

因此，采用多任务学习方法提升环境感知性能。

多学习任务的选择

- 本车与车道线距离回归任务
- 偏航角回归任务
- 前方车道朝向分类任务
- 道路上是否有其他车辆
- 车道数量
- ...

多任务学习网络预测

驾驶员视角图像

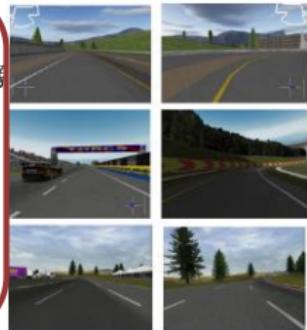
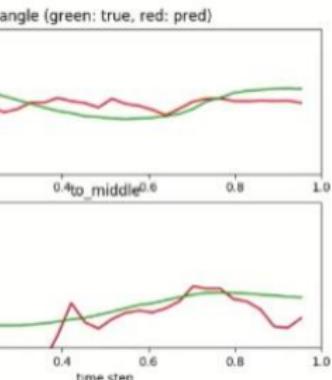
$$\hat{d}_t, \hat{\psi}_t, \hat{p}(c_{i,t}) = \phi(o_t; \theta_p)$$

道路关键特征：

距离: \hat{d}_t

偏航角: $\hat{\psi}_t$

朝向概率: $\hat{p}(c_{i,t})$



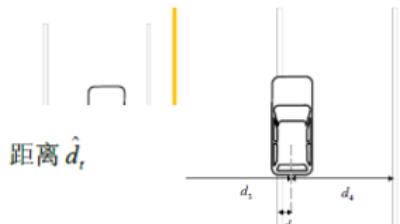
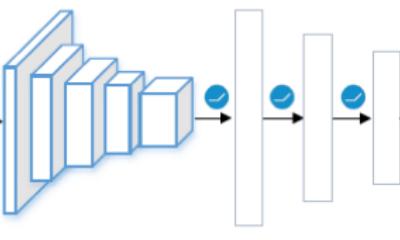
基于深度强化学习的决策控制

多任务学习环境感知模块

□ 距离预测任务

最小化本车与车道线距离向量 $d = [d_1, d_2, d_3, d_4, d_5]$
的预

□ 偏最小

输入图像 σ_t 

□ 车道路



□ 多任务学习损失

三种学习任务的加权和：

$$\mathcal{L}(\theta_p) = \sum_{k=1}^3 \lambda_k \mathcal{L}_k + \Phi(\theta_p), \quad \Phi(\theta_p) = \|\theta_p\|_2^2$$

强化学习控制模块

确定策略梯度算法

- 目标函数

$$J = \mathbb{E}_{s,a,r} \left[\sum_{t=1}^T \gamma^{t-1} r_t \right]$$

- Critic网络，估计值函数

$$\mathcal{L}(\theta_Q) = \mathbb{E}_{s_t, a_t, r_{t+1}, s_{t+1}} [(y_t - Q(s_t, a_t; \theta_Q))^2]$$

$$y_t = r_t + \gamma Q(s_{t+1}, \mu(s_{t+1}; \theta_\mu^-); \theta_Q^-)$$

- Actor网络，根据确定策略梯度定理，最大化目标函数

$$\nabla_{\theta_\mu} J = \mathbb{E}_{s_t} [\nabla_a Q(s_t, a; \theta_Q)|_{a=\mu(s_t; \theta_\mu)} \nabla_{\theta_\mu} \mu(s_t; \theta_\mu)]$$



来自Critic的梯度

来自Actor的梯度

基于深度强化学习的决策控制

强化学习控制模块

强化学习决策系统

- 状态 s_t 为感知模块预测的道路关键特征:

$$s_t = [\hat{d}_t, \hat{\psi}_t, v_{x,t}, v_{y,t}]$$

横纵向速度

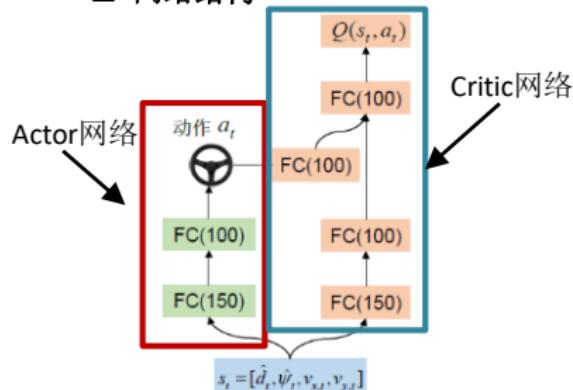
- 动作为方向盘转角量:

$$a_t = \begin{cases} \mu(s_t; \theta_\mu), & \text{如果 } p > \epsilon, \\ \mu(s_t; \theta_\mu) + \beta \mathcal{N}(0, 0.05^2), & \text{其他.} \end{cases}$$

- 奖赏函数:

$$r = \begin{cases} \cos(\psi) - \lambda \sin(|\psi|) - \frac{d}{w}, & \text{如果 } |\psi| < \frac{\pi}{2}, \\ -2, & \text{其他.} \end{cases}$$

- 网络结构



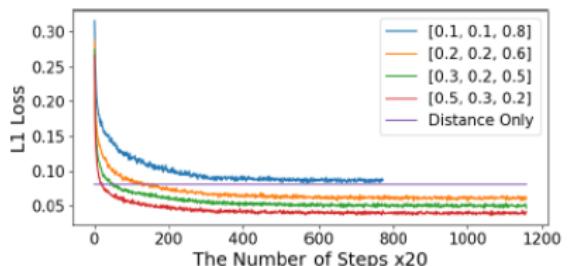
基于深度强化学习的决策控制

环境感知结果

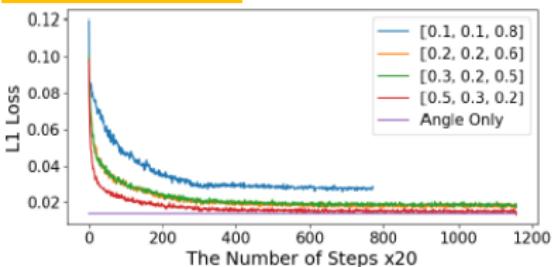
训练阶段

数据集：包含**9个驾驶场景**，训练集约9.9万张图片

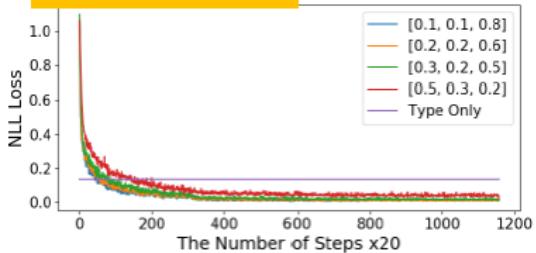
距离损失函数



偏航角损失函数



道路朝向损失函数



基于深度强化学习的决策控制

环境感知结果 测试阶段

数据集：包含**9个驾驶场景**，测试集约1.7万张图片。

■ 测试集准确率与误差

	Task Loss Coefficient α			Distance	Angle	Type
	Distance	Angle	Type	Loss l_1	Loss l_2	Accuracy
Distance	1	0	0	0.06771	-	-
Angle	0	1	0	-	0.01304	-
Type	0	0	1	-	-	99.24%
MTL 1	0.1	0.1	0.2	0.03841	0.01326	99.14%
MTL 2	0.2	0.2	0.6	0.02655	0.01233	99.15%
MTL 3	0.3	0.2	0.5	0.02253	0.0115	98.78%
MTL 4	0.5	0.3	0.2	0.01739	0.01838	98.03%

与单任务学习基线模型对比

多提方法更加稳定。

平均距离预测误差: 0.071m

平均偏航角预测误差: 0.003rad

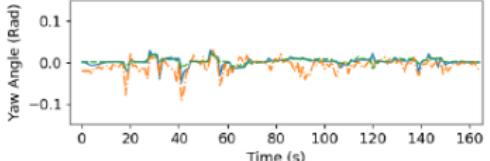
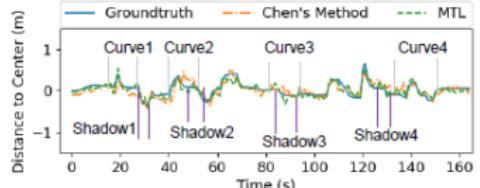
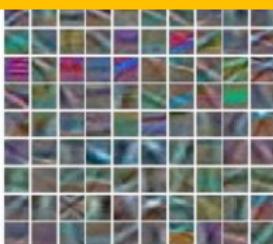
未知赛道上的泛化性

预测性能鲁棒:

平均距离预测误差: 0.107m

平均偏航角预测误差: 0.004rad

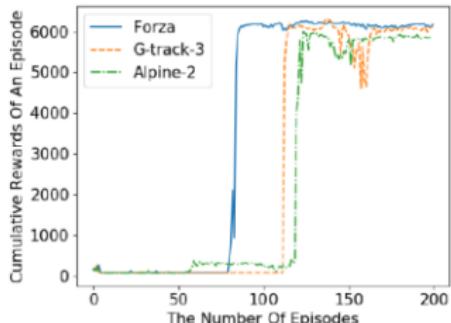
第一层卷积核可视化



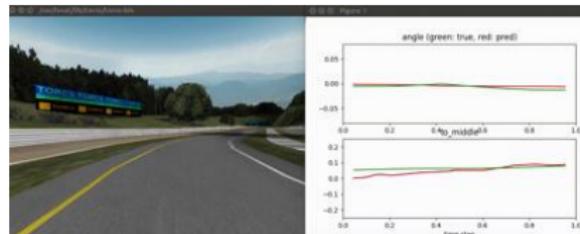
基于深度强化学习的决策控制

横向控制结果

不同难度赛道训练曲线



视觉横向控制



未知赛道控制：单车道



未知赛道控制：三车道



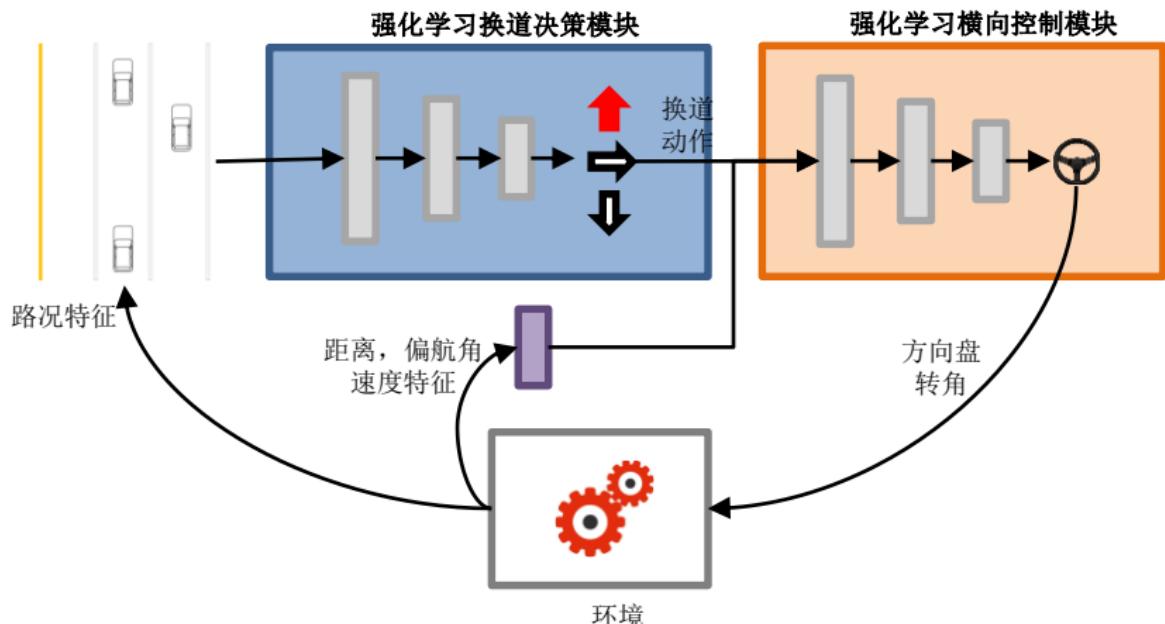
横向控制结果

■ 强化学习与LQR和MPC车道保持控制性能对比

Tracks	LQR Setup					H_p	LQR	MPC	Scores	
	q_1	q_2	q_3	q_4	ρ				RL	
Forza	2.0	1.0	2.0	0.2	0.05	8	6333.1	6348.1	6372.3	
Forza	2.0	0.2	2.0	0.1	0.01		6335.5	6346.3		6375.1
Forza	1.0	0.2	1.0	0.1	0.01		6335.9	6344.7		6372.9
Alpine-2	2.0	1.0	2.0	0	0.05	8	4400.0	4411.6	4415.6	
Alpine-2	2.0	0.3	2.0	0	0.01		4364.0	4405.4		4419.4
Alpine-2	2.0	0.5	1.0	0	0.01		4400.1	4401.2		4415.9
Eroad	3.0	0.2	1.5	0	0.03	8	3585.9	3603.6	3592.8	
Eroad	1.0	0.8	2.5	0	0.01		3591.9	3589.0		3593.9
Eroad	1.5	0.5	1.5	0.03	0.05		3520.9	3557.4		3592.9
G-track-3	2.0	1.0	2.0	1.0	0.05	8	3110.3	3210.5	3213.5	
G-track-3	2.0	0.2	2.0	0.1	0.01		3209.4	3206.3		3212.4
G-track-3	1.0	0.2	1.0	0.1	0.01		3184.6	3187.1		3215.3

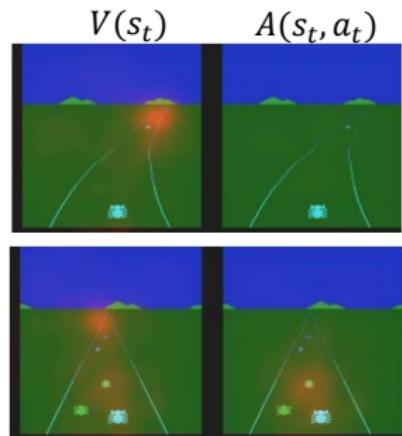
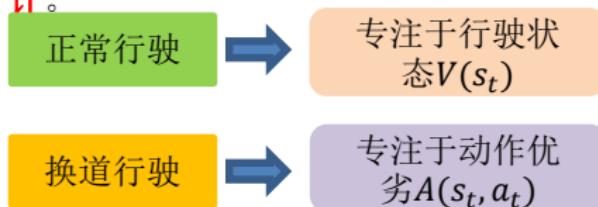
换道决策系统框图

结合**上层决策模块与下层横向控制模块**，实现换道超车功能。



竞争架构深度Q网络算法

分别估计状态值函数 $V(s_t)$ 和优势函数 $A(s_t, a_t)$, 实现**状态与动作的分离估计**。



□ 值函数

$$Q(s_t, a_t) = \mathbb{E}_{s, a \sim \pi, r} \left[\sum_{k=t}^T \gamma^{k-t} r_k \right], \quad V(s_t) = \mathbb{E}_{a \sim \pi} [Q(s_t, a; \theta_Q)]$$

□ 优势函数

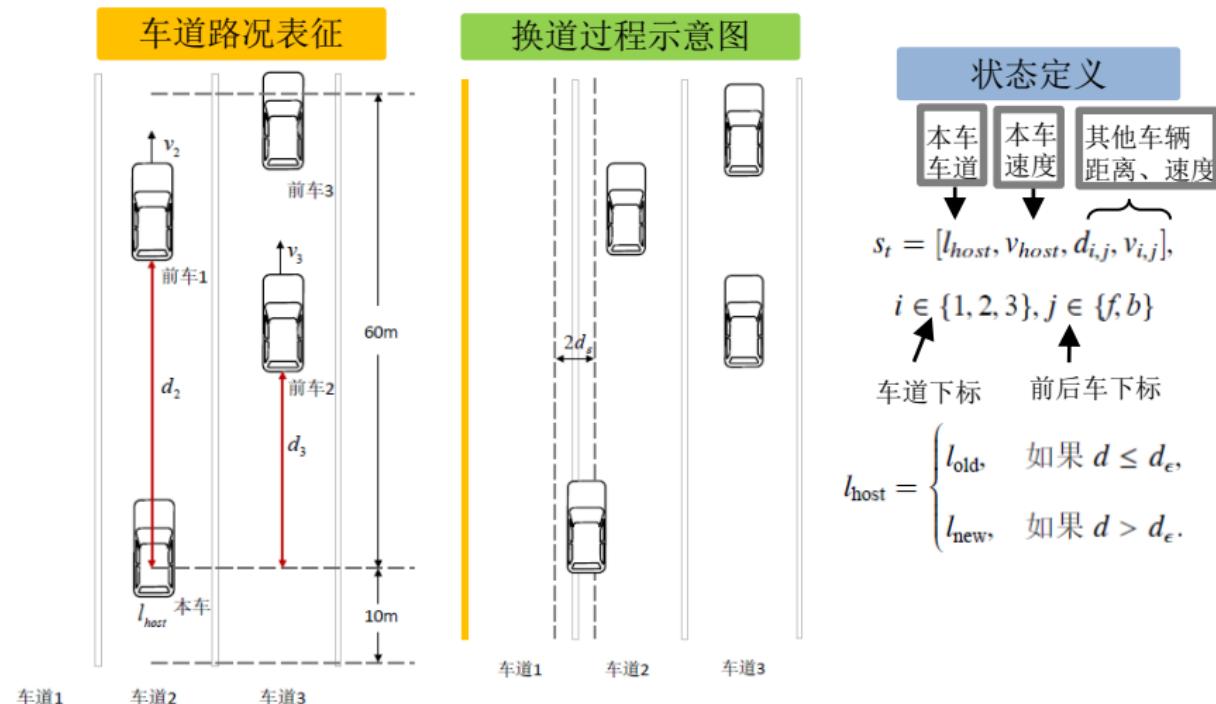
$$A(s_t, a_t) = Q(s_t, a_t) - b(s_t) \quad \xrightarrow{\text{代入}} V(s_t)$$

□ 非二义性V.A函数还原

$$Q(s_t, a_t; \theta_Q) = V(s_t; \theta_V) + A(s_t, a_t; \theta_A) - \frac{1}{|\mathcal{A}|} \sum_a A(s_t, a; \theta_A)$$

强化学习决策模块系统组成

□ 状态：描述当前车道路况



基于深度强化学习的决策控制

强化学习决策模块系统组成

□ 动作: $a_t = \{-1, 0, 1\}$

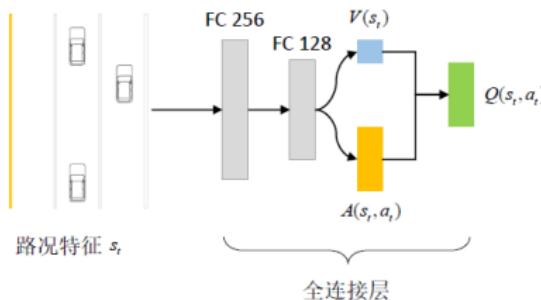
- 向左侧车道换道(-1)
- 保持当前车道不变(0)
- 向右侧车道换道(1)

$$a_t = \begin{cases} \arg \max_{a \in \mathcal{A}} Q(s_t, a; \theta_Q), & \text{如果 } p > \epsilon, \\ \text{random_select}(-1, 0, 1), & \text{其他.} \end{cases}$$

□ 奖赏函数

- **安全性:** 合理换道, 防止与其他车辆碰撞, $r_{\text{col}} = -10.0$ 。
- **有效性:** 本车在最两侧车道时, 不应再向外侧换道, $r_{\text{invalid}} = -1.0$ 。
- **合理性与舒适性:** 避免无意义的频繁换道, 换道时 $r_{\text{comft}} = -0.1$,
不换道时 $r_{\text{comft}} = 0$ 。
- 鼓励**尽可能远**的无碰撞行驶, 每一步给其奖励 $r_{\text{run}} = 0.3$ 。

□ 网络结构



换道决策结果

TORCS平台，双车道与三车道场景

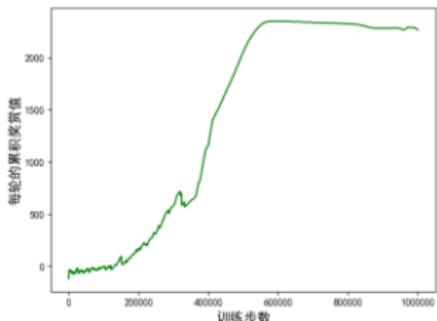
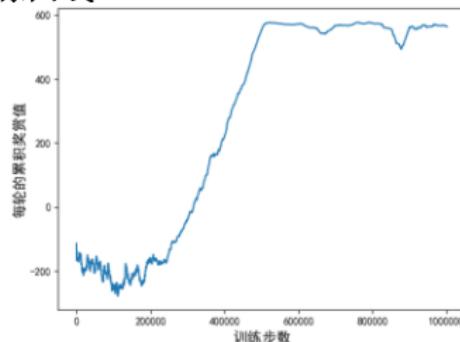


G-track-3双车道，6辆其他车



Brondehach三车道，9辆其他车

训练曲线



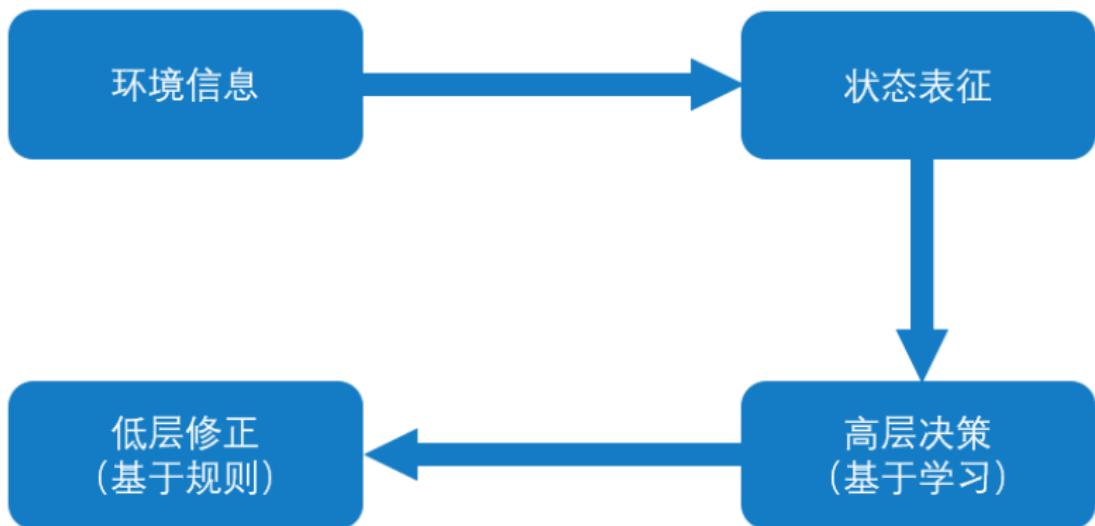
换道决策结果

方法	平均速度	成功率
竞争架构策略	60.1	100%
DQN策略	56.5	60%

控制效果视频

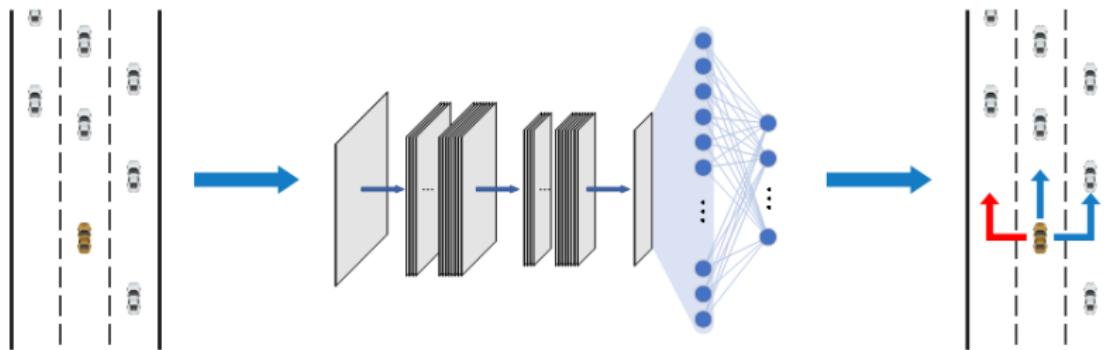


基于深度强化学习的决策控制



基于深度强化学习的决策控制

高层决策(DQN):



- Markov decision process(MDP):

$$M = \langle S, A, P_{sa}, R \rangle$$

- Q-learning:

$$Q^\pi(s, a) = \mathbb{E}_\pi \left[\sum_{k=0}^{\infty} \gamma^k r_{t+k} \mid s_t = s, a_t = a \right]$$

$$Q^*(s, a) = \max_\pi Q^\pi(s, a)$$

$$Q^*(s, a) = \mathbb{E} \left[r + \gamma \max_{a' \in A} Q^*(s', a') \mid s, a \right]$$

- DQN:

A neural network denoted by $Q(s, a; w)$

Loss function

$$L_i(w_i) = \mathbb{E}_{s \sim \pi} [(y_i - Q(s, a; w_i))^2]$$

Where

$$y_i = \mathbb{E}_{s' \sim E} [r + \gamma \max_{a'} Q(s', a'; w_i) \mid s, a]$$

[Mnih, et al. 2013]

基于深度强化学习的决策控制



动作:

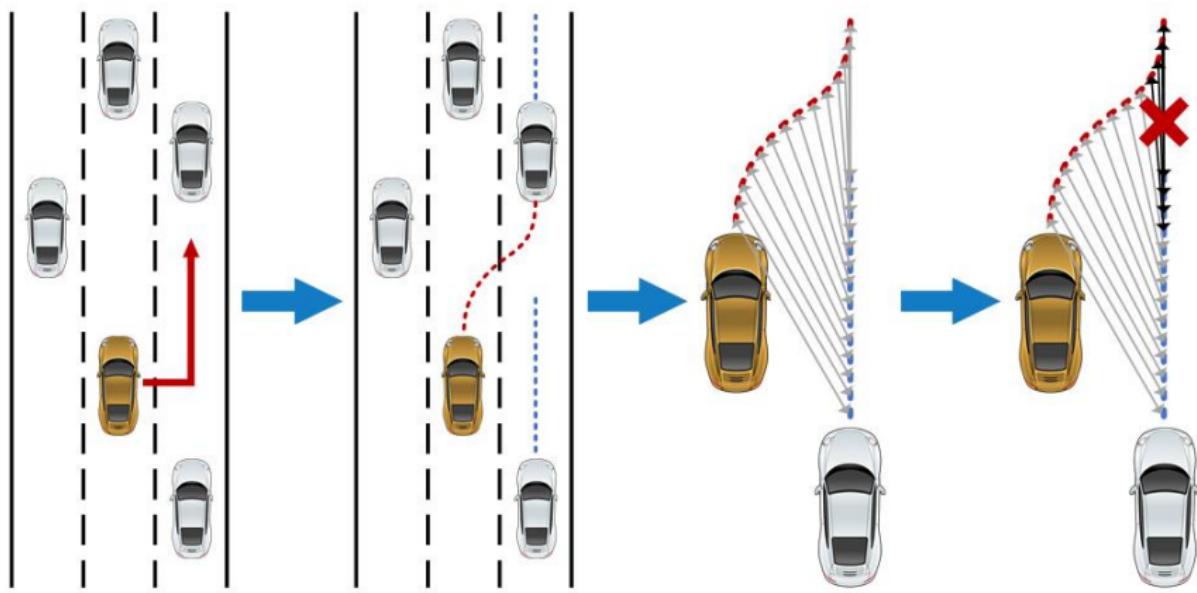
决策	动作
a_0	保持当前车道
a_1	向左换道
a_2	向右换道

奖赏:

$$r = \begin{cases} r_{co} & \text{发生碰撞} \\ r_{ch1} & \text{非法换道} \\ r_{ch2} & \text{无效换道} \\ \lambda \cdot (v - v_{ref}) + r_{ch3} & \text{合法换道} \\ \lambda \cdot (v - v_{ref}) & \text{正常行驶} \end{cases}$$

基于深度强化学习的决策控制

低层修正(规则):



高层决策结果

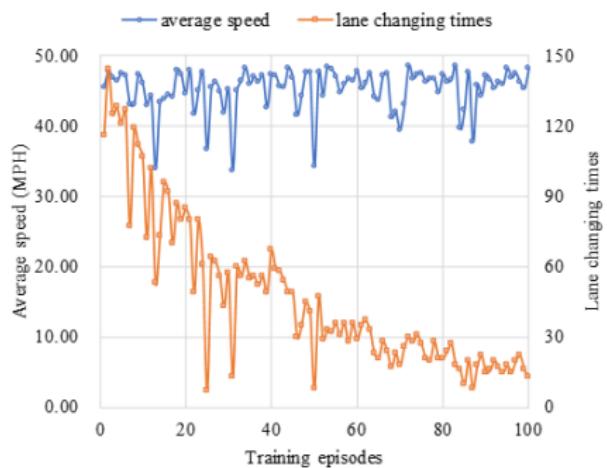
路径规划、预测

距离计算

动作修正

基于深度强化学习的决策控制

训练曲线：



比较：

	avg speed (MPH)	avg lane change times	safety rate
random-action policy	44.59	152.60	0.6
rule-based policy	45.22	8.40	0.6
DQN-based policy	46.16	37.40	0.2
rule-based DQN policy	46.99	8.80	0.8

- 所有四种方法使用同样的纵向控制器
- 随机策略、基于规则的策略和结合规则的DQN策略使用相同的低层校正规则，纯粹的DQN策略没有低层校正

- ◆ 自动驾驶简介
- ◆ 自动驾驶软件架构
- ◆ 深度强化学习与自动驾驶应用
 - ◆ 视觉输入端到端控制
 - ◆ 基于深度强化学习的决策控制
- ◆ 总结

◆ 总结



自动驾驶是一个高度复杂且集成化的系统，目前的DRL决策方法还处于探索阶段，但相对肯定的是利用DRL会成为交互决策的有效技术

如何建立大量训练/测试场景是当前L4级无人驾驶中DRL决策部分面临的关键问题

Safe RL以及算法实车迁移性和可解释性是DRL在自动驾驶领域应用需要重点关注的方向

虚拟仿真器与实际路测道路之间的GAP仍有待解决。