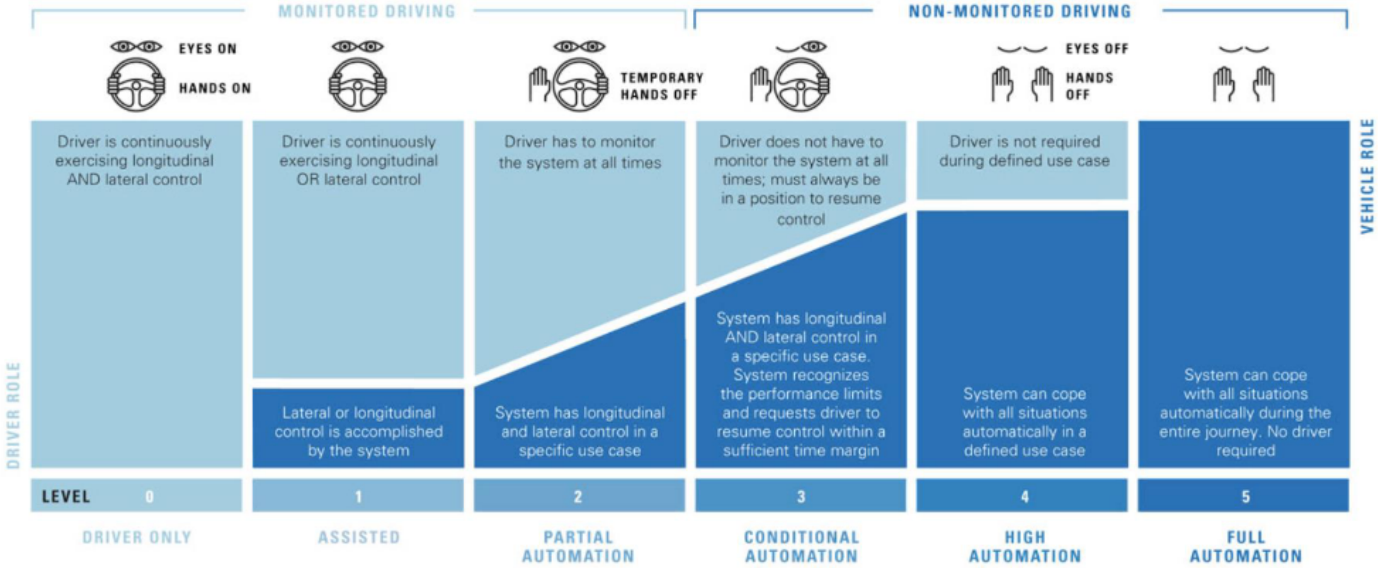
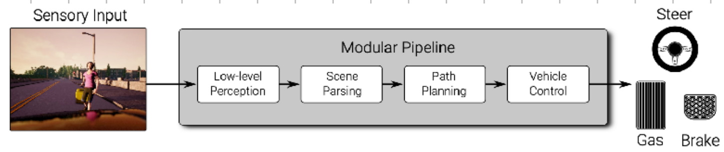
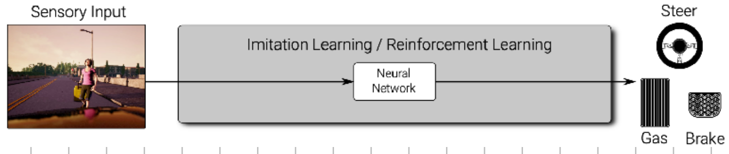
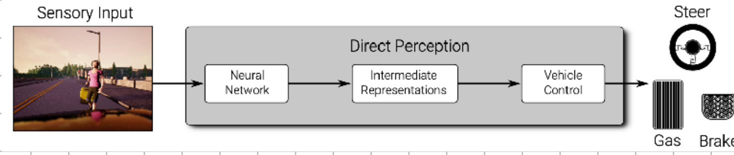




- Small components
- interpretability
- piece-wise-training
- path finding relies on HD images



- End-to-end requires (cheaper) annotation
- lack of interpretability
- issues with generalization



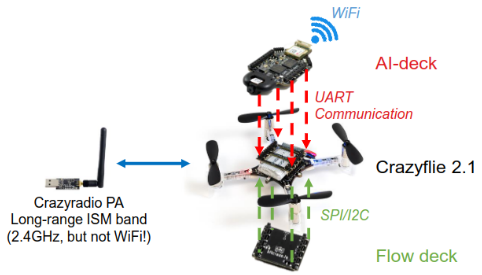
- middle ground
- control is not learnt jointly

- Active
- ultrasonic (5m): blind spot, parking sensor
- radar (300m): adaptive cruise control
- lidar (100m): self driving protocols

- Passive
- camera (50m): get images

GAP8: RISC-V low power 8+1 core / 250 MHz clock / convolutional neural network accelerator

Crazyflie 2.1 nanodrone
- Main processor: ARM Cortex-M4 @168MHz, 192kb SRAM, 1Mb
- Radio: ARM Cortex-M0, 32Mhz, 16kb SRAM, 128kb flash
  - 2.4GHz ISM band radio
- Micro-USB connector

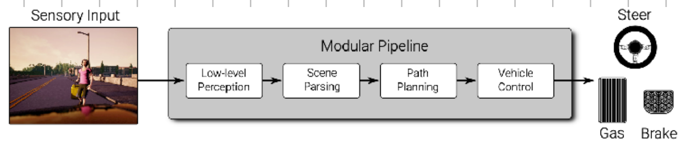


imitation learning: learning from experts
Bad for unexpected situations

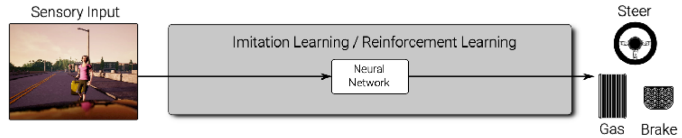
- PilotNet: E2E learning for self driving car
- DroneNet: using img to feed ResNet

Supervized learning: expert demonstration / computing the diff between our actions and experts action / compute loss on affordance indicators

Reinforcement learning (RL): minimize time to target and collision
Q-learning: algo trying to maximise total rewards over n steps
↳ bad scalability $\xrightarrow{\text{fix}}$ Deep Q-learning or multiagent RL



+ Modular + Interpretable - Expert decisions - Piece-wise training

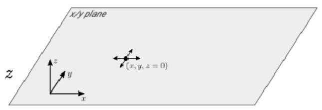


+ End-to-end + Simple - Generalization - Interpretable - Data

affordances: centerline distance / relative angle to road
distance to car / traffic sign / hazards

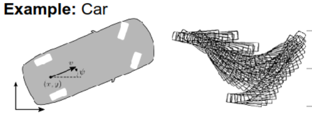
Holonomic constraints
→ Constraints position (config) space
→ Can freely move in any direction
→ Controllable degrees of freedom equal the total degrees of freedom
→ A constraint is defined as f(x,y,z)=0

Example: a 3D-particle in which z=0



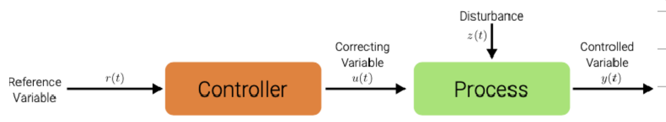
Non-holonomic constraints
→ Constraints velocity space (i.e. the derivative of position)
→ Cannot freely move in any direction
→ Controllable degrees of freedom less than the total degrees of freedom
→ Constraint cannot be defined as f(x,y,z)=0

Example: Car



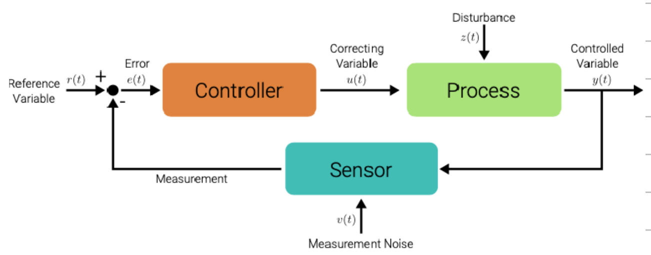
Kinematics: geometric description of motion in space
Kinetics: describes the laws of the causes of motion
inertial Frame: fixed to earth
veichle Frame: fixed to veichle

Open-loop control



- requires precise knowledge of the plan and influence factors
- no feedback about the control variable
- cannot handle unknown disturbances, causes drift

Closed-loop control



- exploits feedback to optimize the error between ref and measure

PID (Black-Box)

- Proportional: alone leads to overshooting/oscillation
- Integral: correct residual errors by integrating past errors
- Derivative: reduces oscillation by introducing dampening

Closed loop-control:

1. black box: dont require knoledge about the process.
2. Geometric: exploits relationship between car and path resulting in compact controls laws for path planning
3. Optimal: uses knowledge of the system and minimize an objective function over future steps

Geometric: - Pure Pursuit control
- Stanley control (DARPA)

Optimal control: - Model predictive control (MPC)