# End-to-end driving from vision with Deep-Learning

Pr. Fabien MOUTARDE
Center for Robotics
MINES ParisTech
PSL Université Paris

Fabien.Moutarde@mines-paristech.fr
http://people.mines-paristech.fr/fabien.moutarde



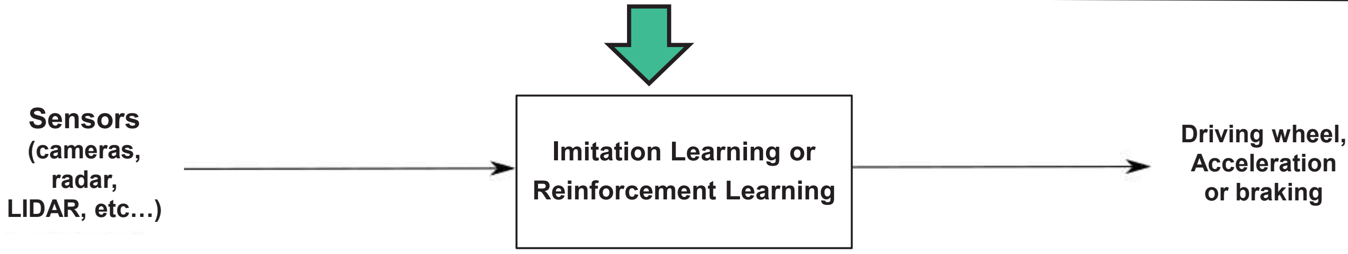
# Outline

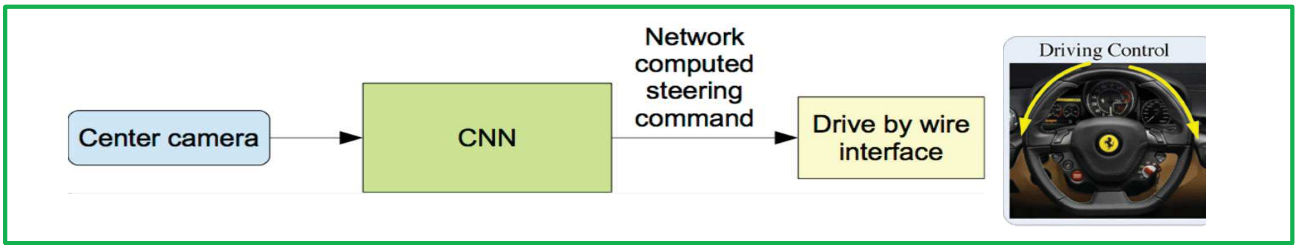
- End-to-end driving & IMITATION learning
- Reinforcement Learning (RL)
- Deep RL (DRL) for Automated Driving

# Principle of "*end-to-end*" driving


## *vs.* HUMAN driving: turn/brake *by just looking in front!*

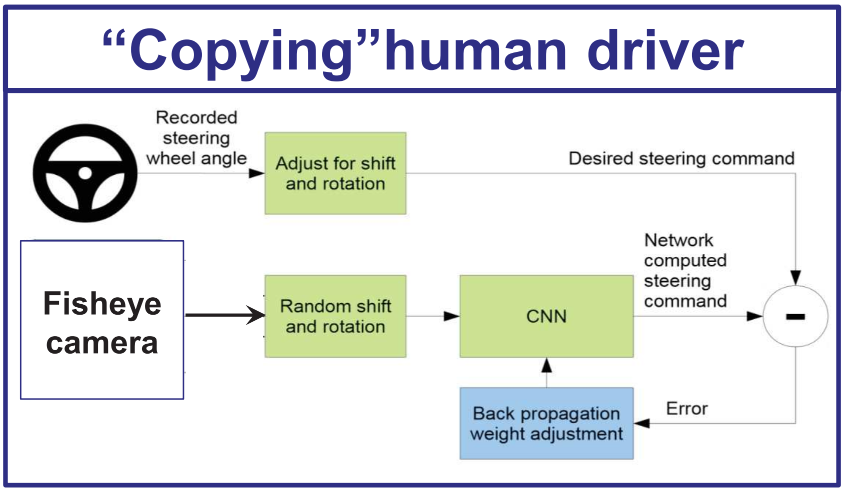
## ≈ *"intelligent" visual servoing*





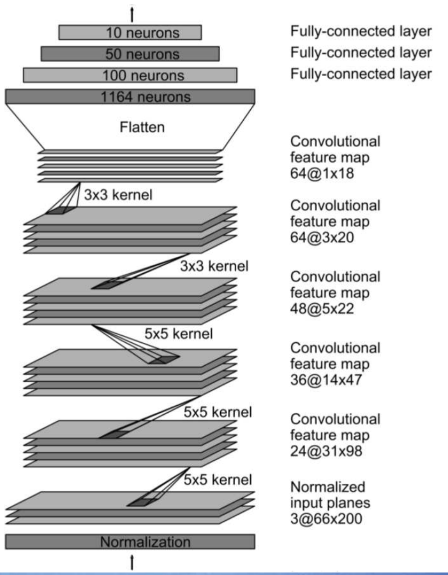

# *Imitation* Learning for end-to-end driving




"End to End Vehicle Lateral Control Using a Single Fisheye Camera", Marin Toromanoff, Emilie Wirbel, Frédéric Wilhelm, Camilo Vejarano, Xavier Perrotton et Fabien Moutarde, 2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2018), Madrid, Spain, 1-5 oct. 2018.

# End-to-end trained convNet

## ConvNet output: steering angle



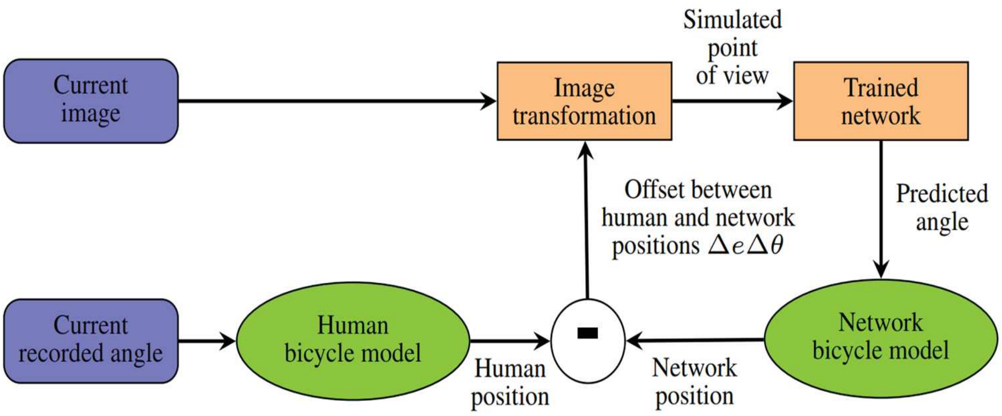

**ConvNet input:
Cylindrical projection of fisheye camera**



# Real data + "simulator" with real images

**Training+testing dataset = 10000 km and 200 hours of human driving in openroad** (highways, urban streets, country roads, etc…) under various weather conditions

**TrainSet = 10 million images, TestSet = 3 million images.**




**"ConvNet in-the-loop" simulator with real images**

***[Work by my Valeo CIFRE PhD student Marin Toromanoff]***

# End-to-end driving: closed loop evaluation

TABLE V: Autonomy (%) and mean absolute distance (MAD, in cm) according to data distribution and validation scenario, the baseline is just going straight.

| Scenario | Urban | | Highways | | Sharp turns | |
|---|---|---|---|---|---|---|
| Metric | Aut. (%) | MAD (cm) | Aut. (%) | MAD (cm) | Aut. (%) | MAD (cm) |
| Original | 99.3 | 16 | **98.7** | **19** | **73.7** | **30** |
| Sel. #1 | 98.9 | **15** | 97.7 | 25 | 83.7 | **27** |
| Sel. #2 | **99.5** | 16 | 97.2 | 24 | **87.5** | 28 |
| Oversamp. | **98** | **18** | **91.8** | **29** | 82.5 | 29 |
| Baseline | 8 | 36 | 14 | 41 | 0 | 35 |

TABLE VI: Comparison of performance between individual networks and bagging

| Scenario | Urban | | Highways | | Sharp turn | |
|---|---|---|---|---|---|---|
| Metric | Aut. (%) | MAD (cm) | Aut. (%) | MAD (cm) | Aut. (%) | MAD (cm) |
| Weights #1 | **99.5** | 16 | 97.2 | 24 | **87.5** | 28 |
| Weights #2 | 98.9 | 15 | 97.7 | 25 | 83.7 | **27** |
| Weights #3 | 99.3 | 16 | **98.7** | **19** | **73.7** | **30** |
| Weights #4 | 98.6 | **18** | **92** | **26** | 85 | 29 |
| Weights #5 | **98.4** | 15 | 96.4 | 21 | 83.7 | 28 |
| Bagging | **99.5** | **13** | **98.7** | **19** | **87.5** | **27** |



# End-to-end driving (learnt purely on real-world data) tested in GTA simulator

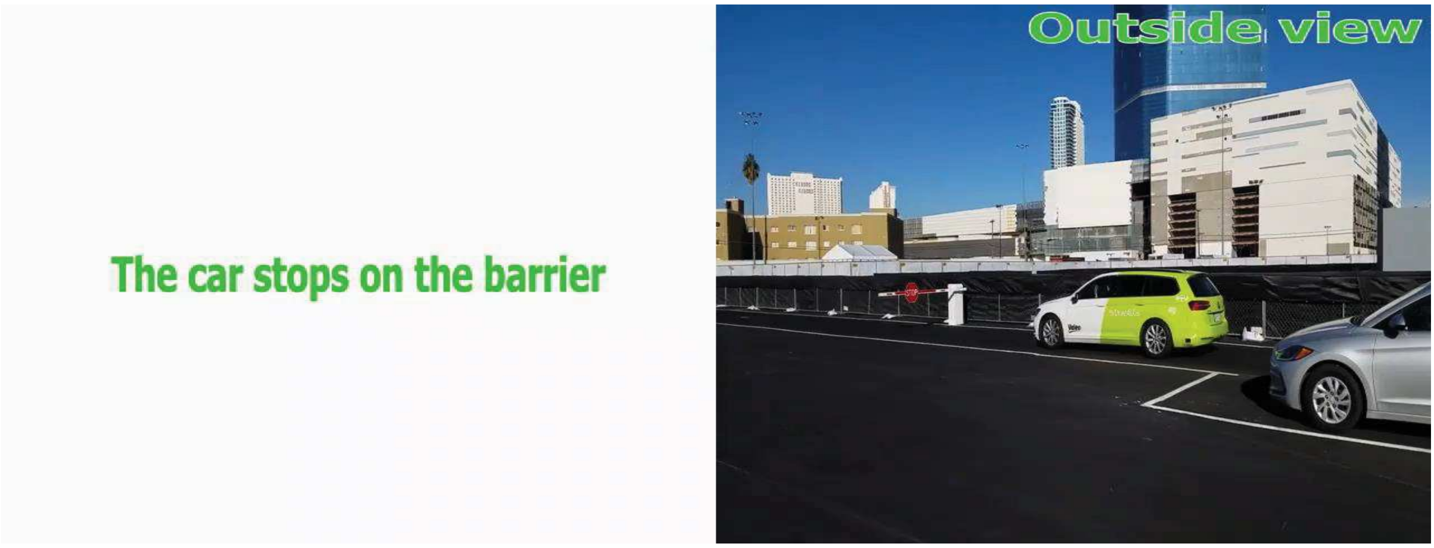
*[Work by my Valeo CIFRE PhD student Marin Toromanoff]*

# Real vehicle end-to-end driving (learnt by imitation)

*[Work by Valeo using ConvNet trained by my CIFRE PhD student Marin Toromanoff]*



# Demonstration at CES'2018




*[Work by Valeo using ConvNet trained by my CIFRE PhD student Marin Toromanoff]*

"End to End Vehicle Lateral Control Using a Single Fisheye Camera", Marin Toromanoff, Emilie Wirbel, Frédéric Wilhelm, Camilo Vejarano, Xavier Perrotton et Fabien Moutarde, 2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2018), Madrid, Spain, 1-5 oct. 2018.

# Conclusion on IMITATION learning for end-to-end driving

**Promising results**

**BUT**

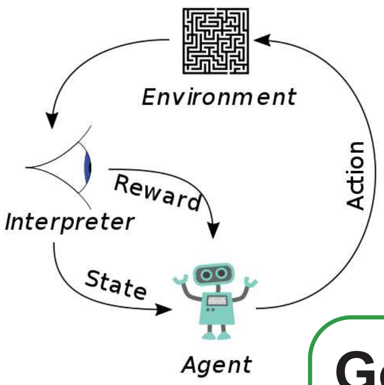
**Difficult to cover large-enough variability of situations/environments with real driving data**

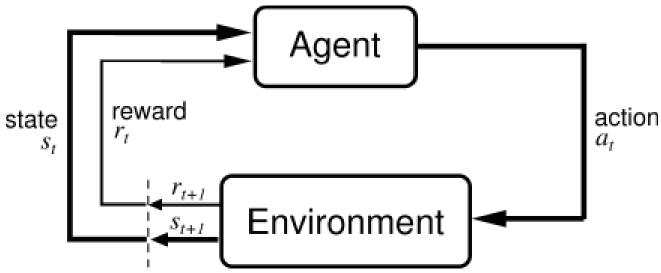


# Outline

- **End-to-end driving & IMITATION learning**
- **<u>Reinforcement Learning (RL)</u>**
- **Deep RL (DRL) for Automated Driving**

# Reinforcement Learning (RL)




**Goal: find a "policy" $a_t=\pi(s_t)$ that**

**Maximizes** $R_t = \sum_{k=0}^{\infty} \gamma^k r_{t+k}, \gamma \in [0, 1[$

**<u>Deep</u> Reinforcement Learning (<u>DRL</u>) if Deep NeuralNet used as model (for policy and/or its "value"): DQN, Actor-Critic A3C**

**End-to-end driving: policy $\pi$ searched as ConvNet(front-image)**



# 3 families of RL algorithms

- **Policy-based** $\pi_\theta \approx \pi^*$
  **optimize a parameterized policy**
- **Value-based** $Q(s, a, \theta) \simeq Q^{\pi^*}(s, a)$
  **find the optimal (parameterized) Q-value**

***Model-free***

- **Model-based**

$$m(s_t, a_t, \theta') \approx s_{t+1}, r_{t+1}$$

## Value and Q-function

- **Value of a policy (from a given state)**

$$V_\pi(s) = \mathbb{E}_\pi[R_t | s_t = s] = \mathbb{E}_\pi[\sum_{k=0}^{T} \gamma^t r_{t+k} | s_t = s]$$

- **Q-function of a policy**

$$Q_\pi(s,a) = \mathbb{E}_\pi[R_t | s_t = s, a_t = a] = \mathbb{E}_\pi[\sum_{k=0}^{T} \gamma^t r_{t+k} | s_t = s, a_t = a]$$

**THERE ALWAYS EXISTS A *DETERMINISTIC* OPTIMAL POLICY $\pi^*$**

$$\forall \pi, \forall s \in S, V_{\pi^*}(s) \geqslant V_\pi(s)$$



## *Deep* Reinforcement Learning with Deep Q Network (DQN)

- **Q-learning:** $Q^{new}(s_t, a_t) \leftarrow (1-\alpha) \cdot \underbrace{Q(s_t, a_t)}_{\text{old value}} + \underbrace{\alpha}_{\text{learning rate}} \cdot \overbrace{\left( \underbrace{r_t}_{\text{reward}} + \underbrace{\gamma}_{\text{discount factor}} \cdot \underbrace{\max_a Q(s_{t+1}, a)}_{\text{estimate of optimal future value}} \right)}^{\text{learned value}}$

- **Optimal policy deduced from optimal Q-value**

$$\pi^*(s) = arg \max_a Q_{\pi^*}(s,a)$$

- **DQN** [1]**: if too many possible states, approximate Q as a neural network, and learn Q* using SGD with loss from Bellman equation**

$$L(s_t, a_t, r_{t+1}, s_{t+1}, \theta) = (\underbrace{r_{t+1} + \gamma \max_a Q(s_{t+1}, a, \theta)}_{target} - Q(s_t, a_t, \theta))^2$$

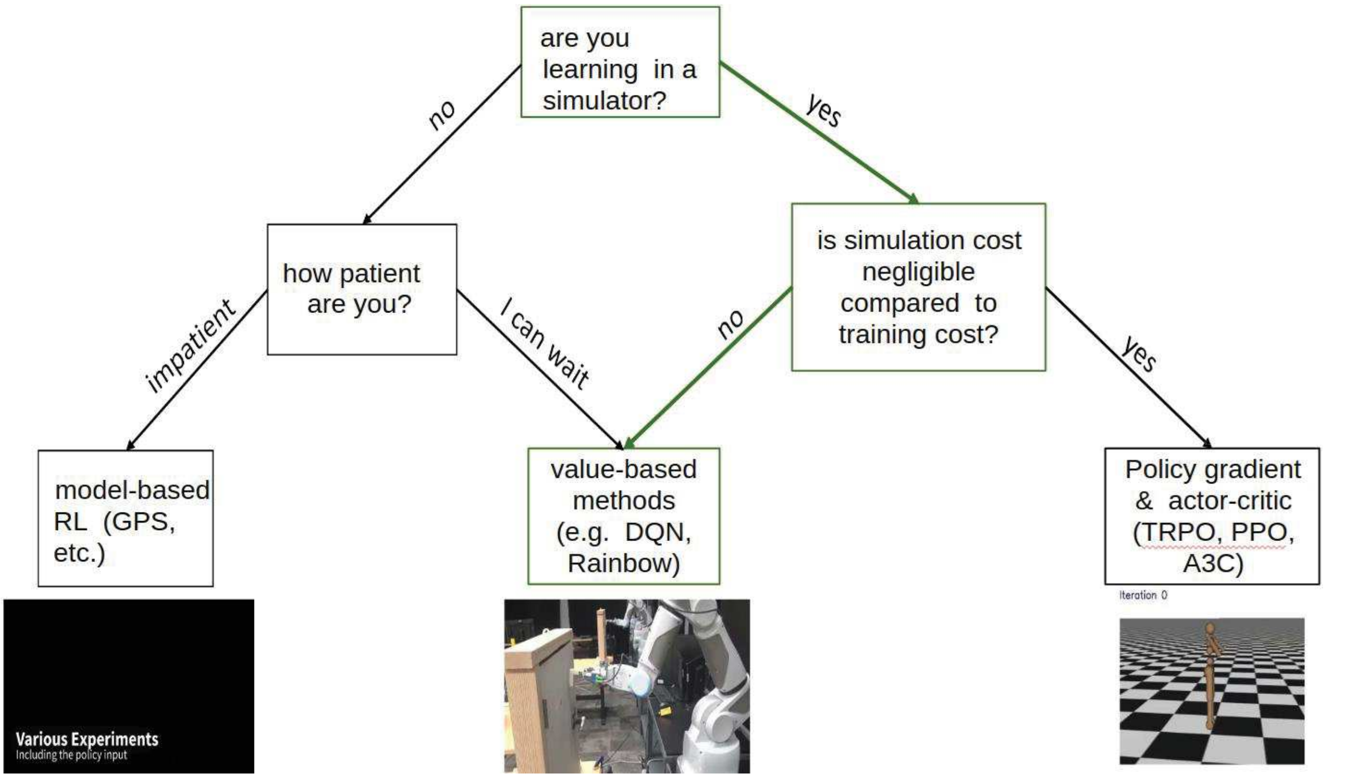
[1] V. Mnih et al : Human-level control through deep reinforcement learning (2015)

# Summary of main RL algorithm types

| Family | Algorithm | On/Off policy | Discrete/continuous output? |
|---|---|---|---|
| Policy based | REINFORCE | On policy | Both |
| | Actor-Critic (A3C) | On policy | Both |
| Value based | SARSA | On policy | Discrete |
| | Q-Learning (DQN) | Off-policy | Discrete |
| Model based | MCTS | Off-policy | Discrete |
| | iLQG | Off-policy | Continuous |



# How to choose RL algorithm?




[1] S. Levine: CS294 *Deep Reinforcement Learning* (2017)

# Outline

- End-to-end driving & IMITATION learning
- Reinforcement Learning (RL)
- <u>Deep RL (DRL) for Automated Driving</u>



# Deep RL for automated driving

- **Until recently, very few published research, and mostly in racing games:**

Asynchronous methods for deep reinforcement learning, V. Mnih, A. P. Badia, M. Mirza, A. Graves, T. P. Lillicrap, T. Harley, D. Silver, and K. Kavukcuoglu, ICML'2016.

End-to-End Race Driving with Deep Reinforcement Learning, Maximilian Jaritz, Raoul De Charette, Marin Toromanoff, Etienne Perot, Fawzi Nashashibi, *ICRA 2018 - IEEE International Conference on Robotics and Automation*, Brisbane, Australia, May 2018.

- **Up to now, only real driving with RL:** "*Learning to Drive in a Day*" (2018, [1])
  - Embed DRL in a real car, and learn « *from scratch* »
  - But *<u>VERY SIMPLE CASE: lane keeping along 250m!</u>*
  - Simulation used before to design architecture and find hyper-parameters

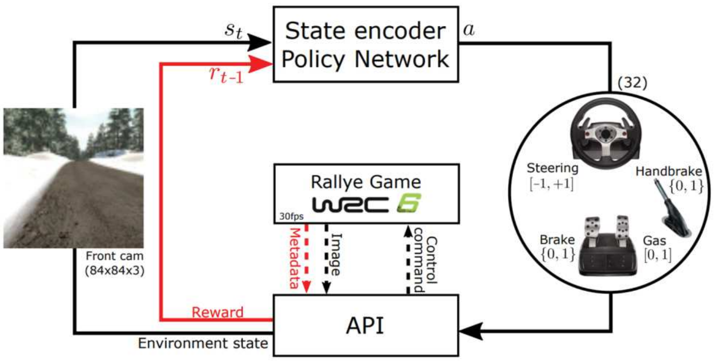
[1] A. Kendall et al.: Learning to Drive in a Day (2018)

# Preliminary DRL experiment for end-to-end driving


*[Work by my Valeo CIFRE PhD student Marin Toromanoff]*

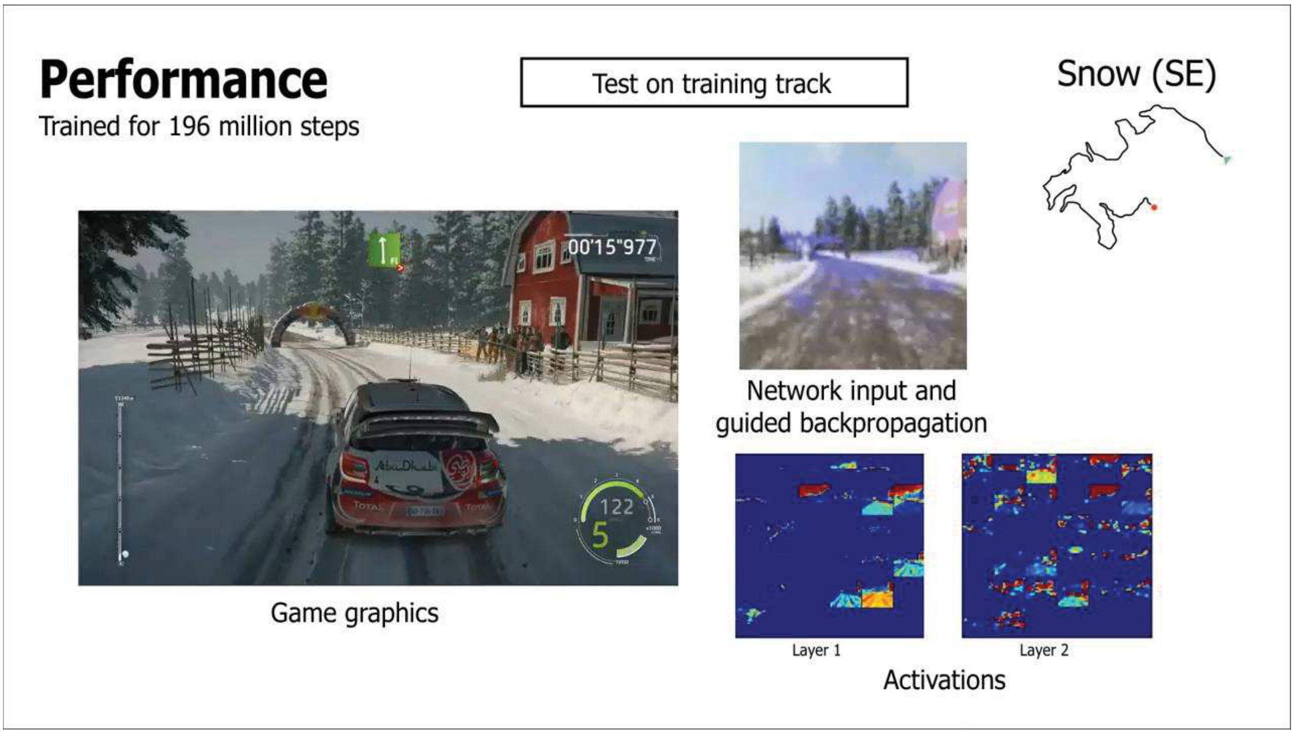


# End-to-end driving learning by RL in racing-car simulator




Etienne Perot, Maximilian Jaritz, Marin Toromanoff, Raoul De Charette. End-to-End Driving in a Realistic Racing Game with Deep Reinforcement Learning, **International conference on Computer Vision and Pattern Recognition - Workshop, Honolulu, United States, Jul. 2017.**

# End-to-end driving learnt by RL in racing-car simulator




End-to-End Race Driving with Deep Reinforcement Learning, **Maximilian Jaritz, Raoul De Charette, Marin Toromanoff, Etienne Perot, Fawzi Nashashibi, *ICRA 2018 - IEEE International Conference on Robotics and Automation*, Brisbane, Australia, May 2018.**



# RL for Automated Driving: why learn in a simulator?

- **RL require huge amount of trial & error, and initial policy = very bad driving!**
  **⇒ *Learn in simulation* (for safety + speed)**
- **Still few driving simulators adapted for DL and RL, and best ones not totally mature**

| Simulateur | GTA | DeepDrive.io | AirSim | CARLA[1] |
|---|---|---|---|---|
| Flexibilité | −− | ++ | ++ | ++ |
| Variété | ++ | −− | − | + |
| Complexité/Réalisme | ++ | −− | − | − |
| Objets mobiles | ++ | −− | −− | + |
| Vitesse exécution | −− | + | + | + |
| Multi-agent | −− | − | − | ++ |

**→ Choice of CARLA**

*[1] A. Dosovitskiy: CARLA: An Open Urban Driving Simulator (2017)*

# CARLA simulator

- Open source, flexible




# CARLA Autonomous Driving challenge

- **Itinerary to be followed in a city** (given by 4 possible orders at intersections: Left, Straight, Right, Follow_Lane)
  BUT must <u>stay on the road</u>, in the lane, respecting Traffic Lights, and no collision with pedestrians and other cars!
- Evaluation metrics = Task completion & Distance between infractions, in an UNSEEN CITY
- Results (and 10.000$ for winners!) on July 1st

## DRL used: Rainbow + IQN + ApeX

- **Rainbow** [1] **= combination of many improvements of DQN** [4] **➔ currently SoA on ATARI benchmark**
- **IQN** [2] **= learning with *probability distributions* rather than just expectation of average**

| | Mean | Median | Human Gap | Seeds |
|---|---|---|---|---|
| DQN | 228% | 79% | 0.334 | 1 |
| PRIOR. | 434% | 124% | 0.178 | 1 |
| C51 | 701% | 178% | 0.152 | 1 |
| RAINBOW | **1189%** | **230%** | 0.144 | 2 |
| QR-DQN | 864% | 193% | 0.165 | 3 |
| IQN | 1019% | 218% | **0.141** | 5 |

- **Ape-X** [3] ***multi-agent* version of DQN allowing massively parallel distributed learning**
  **⇒ Largely better performance, but typically require 22 billions of frames (vs. 200 millions)**

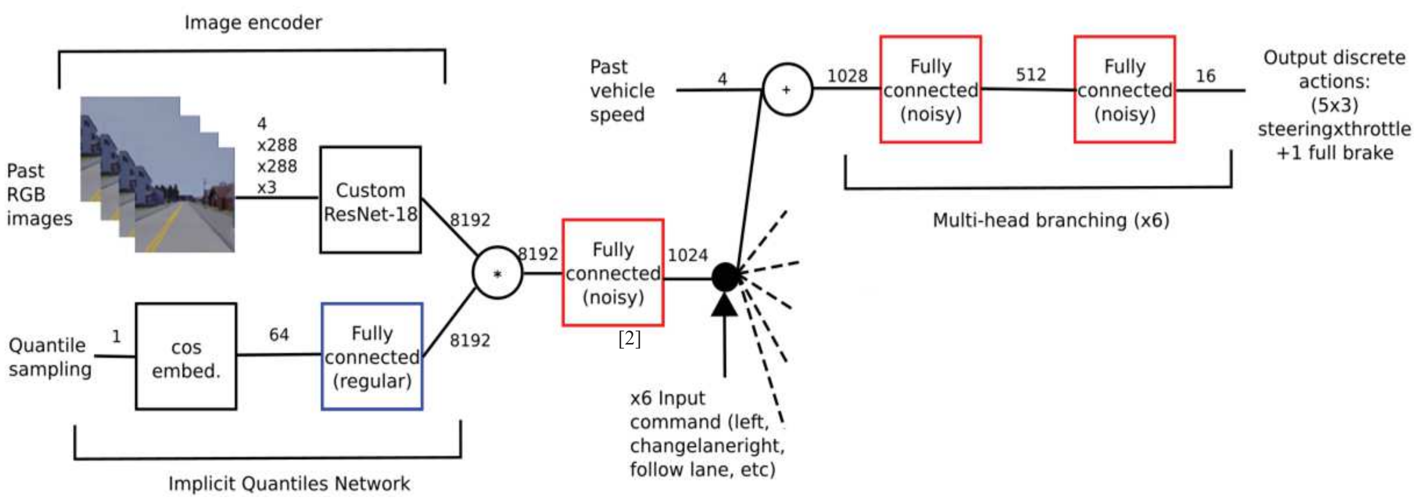
[1] M. Hessel et al : Rainbow: Combining Improvements in Deep Reinforcement Learning Matteo (2017)
[2] D. Silver et al : Implicit Quantile Networks for Distributional Reinforcement Learning (2018)
[3] B. Horgan et al : Distributed Prioritized Experience Replay (2018)
[4] V. Mnih et al : Human-level control through deep reinforcement learning (2015)



## Network architecture

- **U.S. Traffic lights ➔ Need to use *COLOR and high-enough resolution* ⇒ big network, hard and slow to train**
- **Use a resnet-18 (10 times more weight than previously used in *DQN-like* network)**
- **Handle turn-orders (at intersections) with multi-head branching [1]**

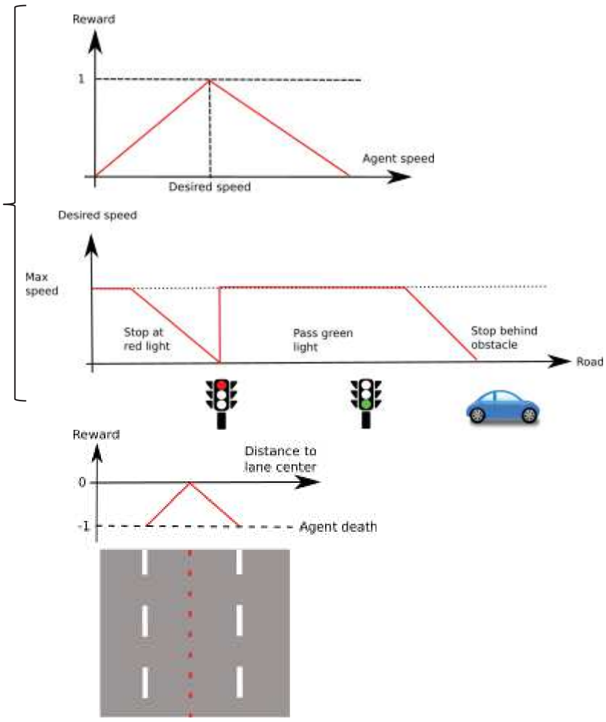
[1] Codevilla et al., *End-to-end driving via Conditional Imitation Learning*, 2017
[2] M. Fortunato et al.. *Noisy Networks for Exploration*, 2017



# Reward shaping

**Rewards scaled in [-1, 1]:**

- **Speed: positive reward to follow speed, depends on obstacles & traffic light**
- **Lateral position: negative reward depending on distance to lane center**
- **Episode terminates on collision, running red traffic light, too far from lane center or stuck (if no reason to stop)**






# Examples of autonomous driving obtained with our DRL

# Conclusions & perspectives

- Very encouraging first results of RL: able to learn a kind of "*Intelligent visual servoing*" for Autonomous Driving
- **Our approach ranked 1st (vision-only track) on CARLA "*Autonomous Driving challenge*" !!**
  ➔ presentation at CVPR'2019 workshop
- Future work:
  - Transferrability to real-world videos
  - Combination of Imitation-Learning and RL?



# QUESTIONS?