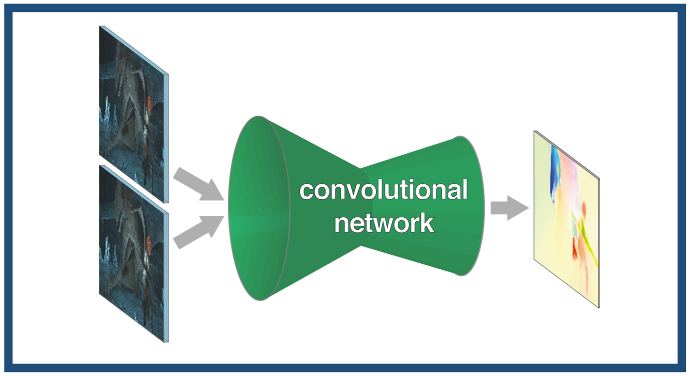
Learning Sensorimotor Control from Experience and Demonstration

Alexey Dosovitskiy
Intel Visual Computing Lab, Munich

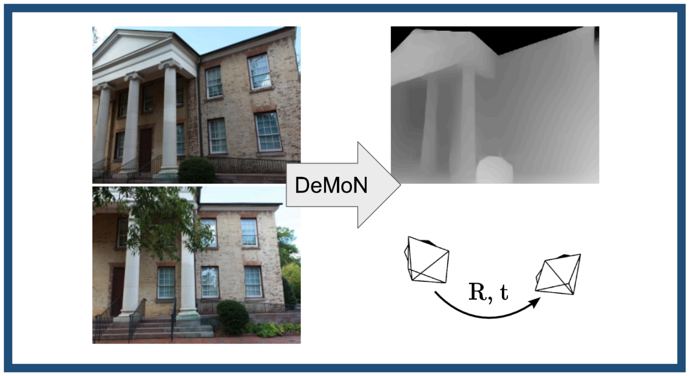
CMP Colloquium
05.10.2017, Prague

COMPUTER VISION University of Freiburg

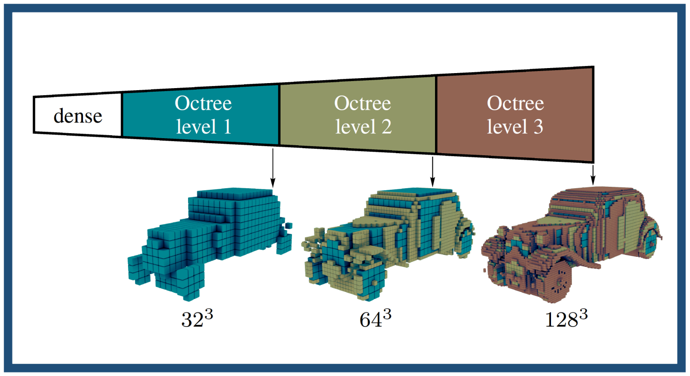
Generating flow, depth and octrees



- FlowNet and FlowNet 2.0



- DeMoN: Depth and Motion Net

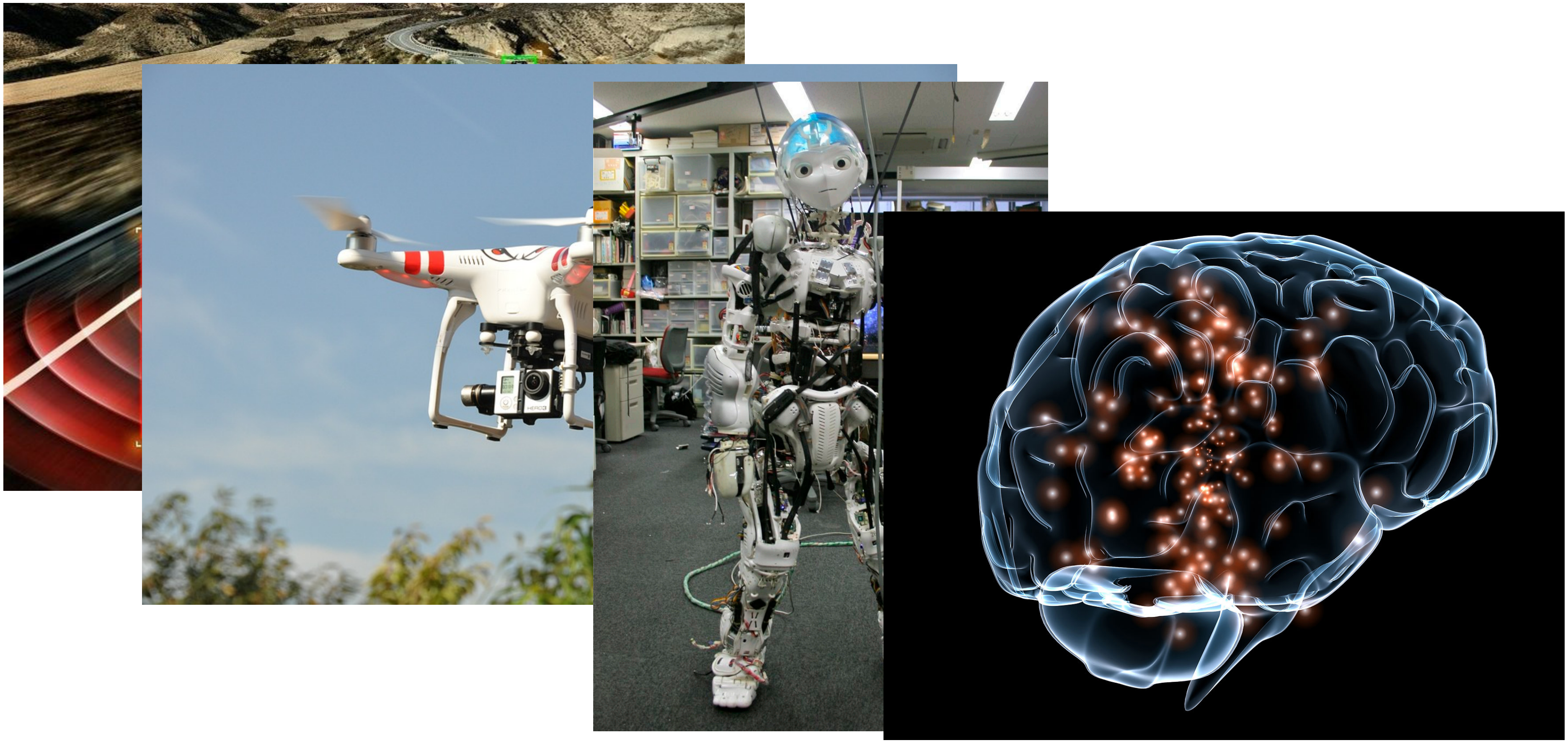


- OGN: Octree Generating Networks

[O] Vision
COMPUTER VISION University of Freiburg

Sensorimotor control

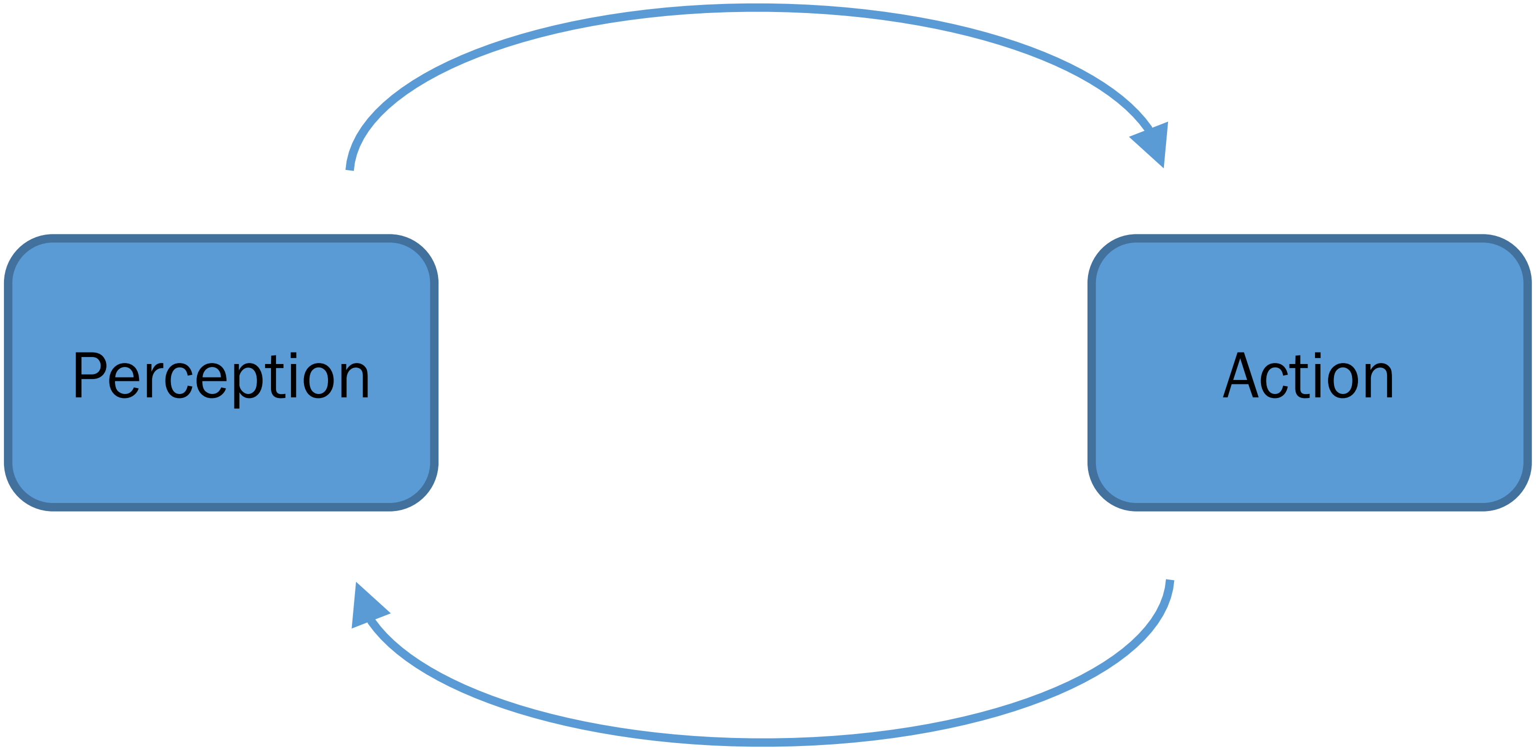
Producing useful motor actions based on sensory inputs



"We have a brain for one reason and one reason only, and that's to produce adaptable and complex movements."

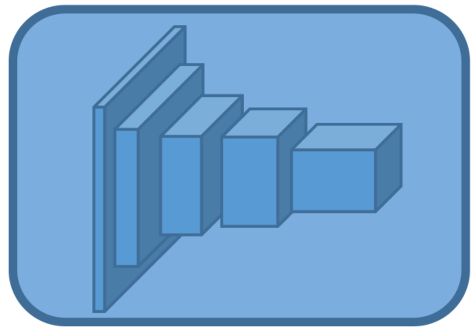
- Daniel Wolpert

Perception and action



End-to-end learning

Predicting controls directly from observations

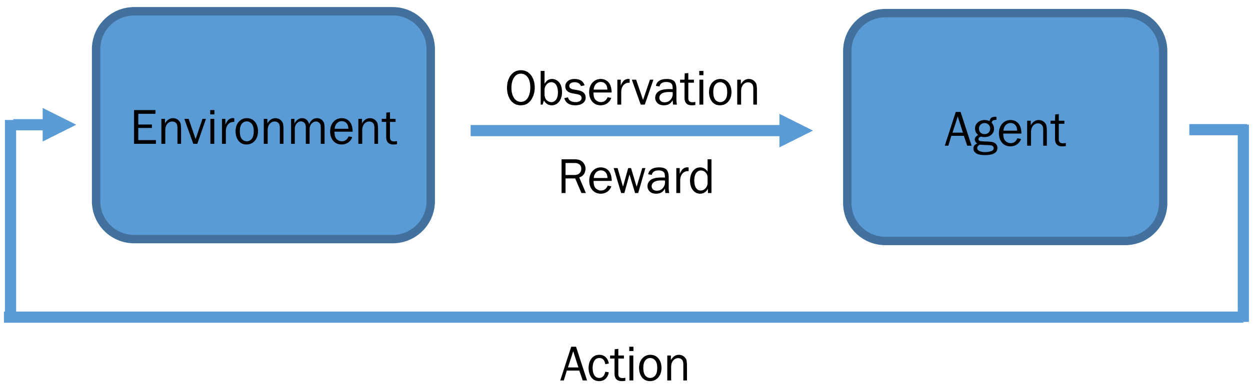


Convolutional network

Control

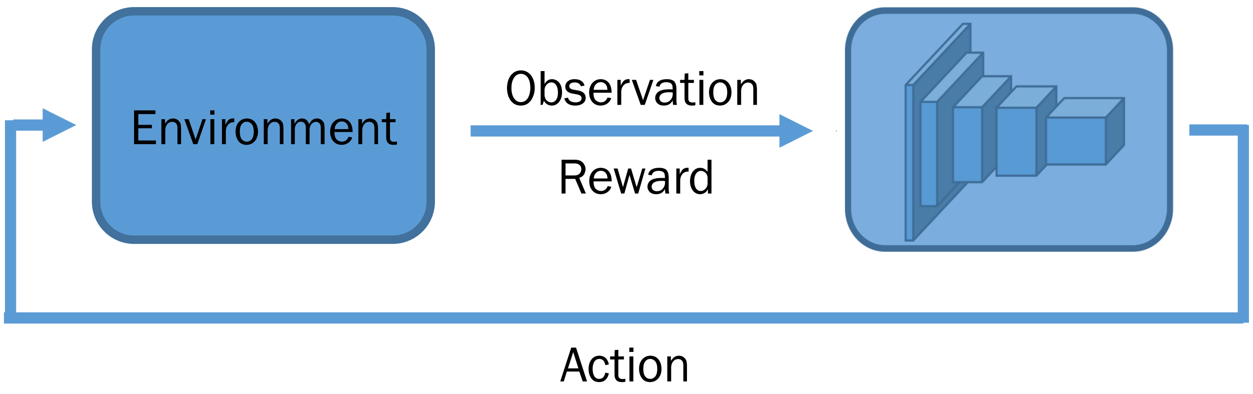
How do we train this?

Reinforcement learning



- Learn from interaction with the environment
- … by maximizing the (discounted) sum of future rewards

End-to-end deep reinforcement learning

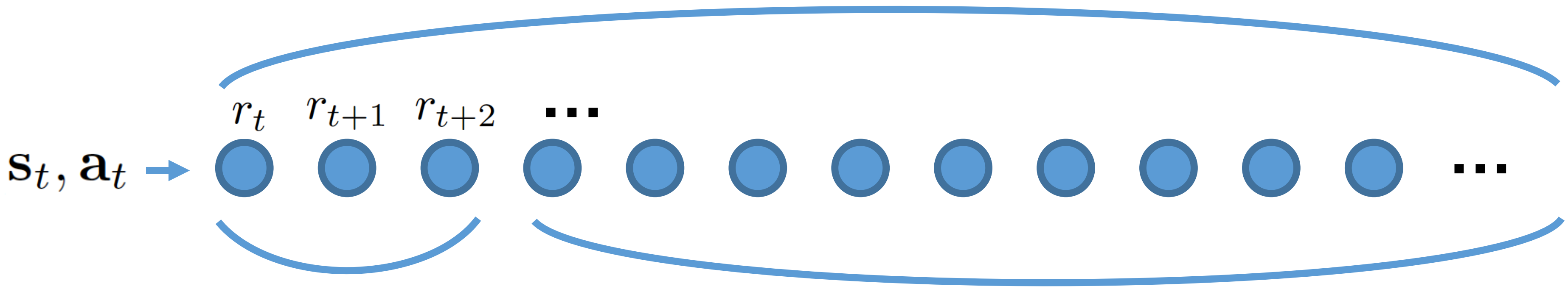


- Learn from interaction with the environment
- … by maximizing the (discounted) sum of future rewards
- Directly map sensory inputs to actions
- … with a deep network

Reinforcement learning 101

$$Q(s_t, a_t; \theta) \approx R_t^{\gamma} = \sum_{i=t}^{T} \gamma^{i-t} r_i = r_t + \gamma r_{t+1} + \gamma^2 r_{t+2} + \ldots$$

(Discounted) return



Rollout

$$\sum_{i=t}^{t+n-1} \gamma^{i-t} r_i$$

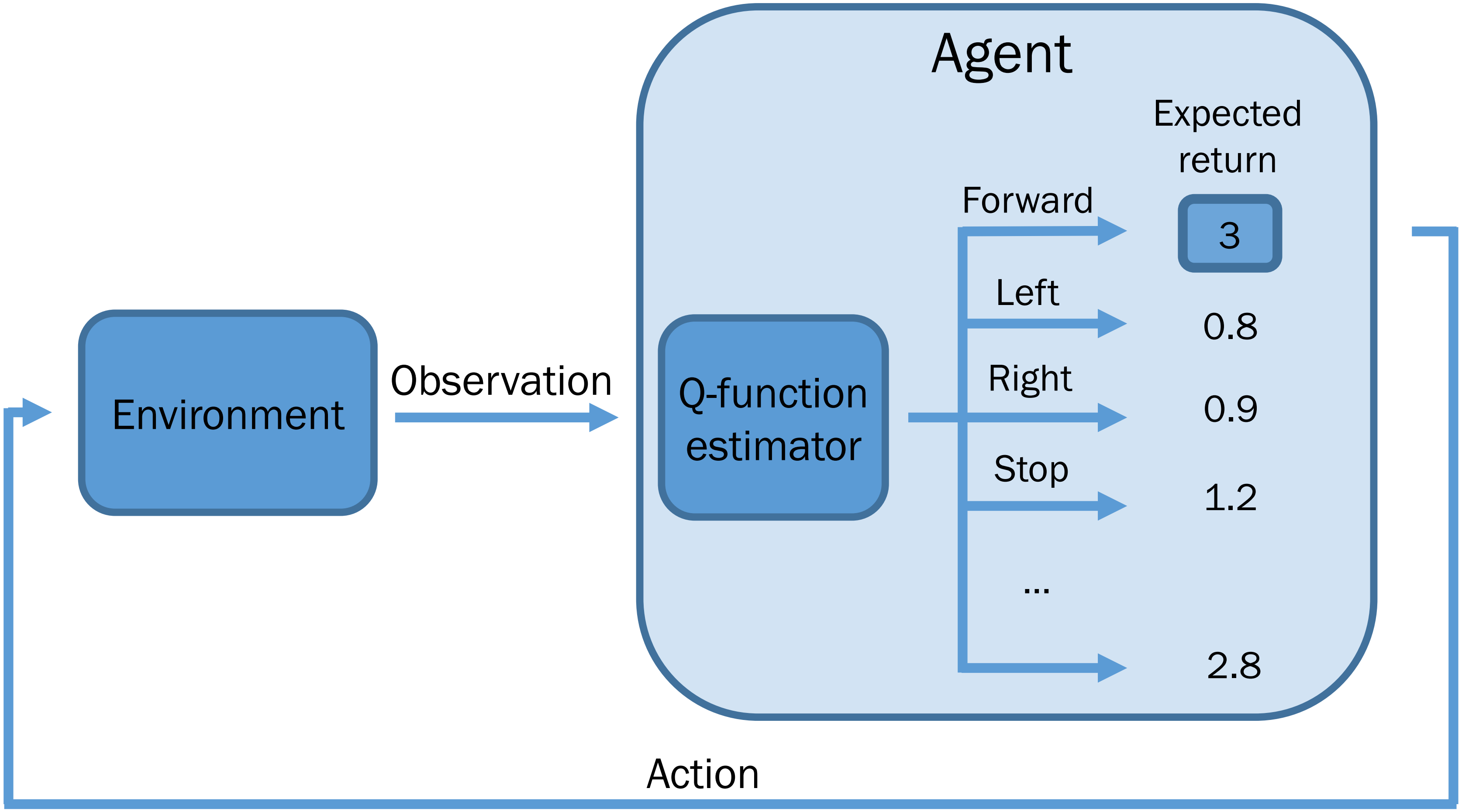
Bootstrapping

$$\approx \gamma^n Q(s_{t+n}, a_{t+n}; \theta)$$

$$\mathcal{L}(\theta) = \left(Q(s_t, a_t; \theta) - Q_{target}\right)^2$$

$$Q_{target} = \boxed{\sum_{i=t}^{t+n-1} \gamma^{i-t} r_i} + \boxed{\gamma^n Q(s_{t+n}, a_{t+n}; \theta)}$$

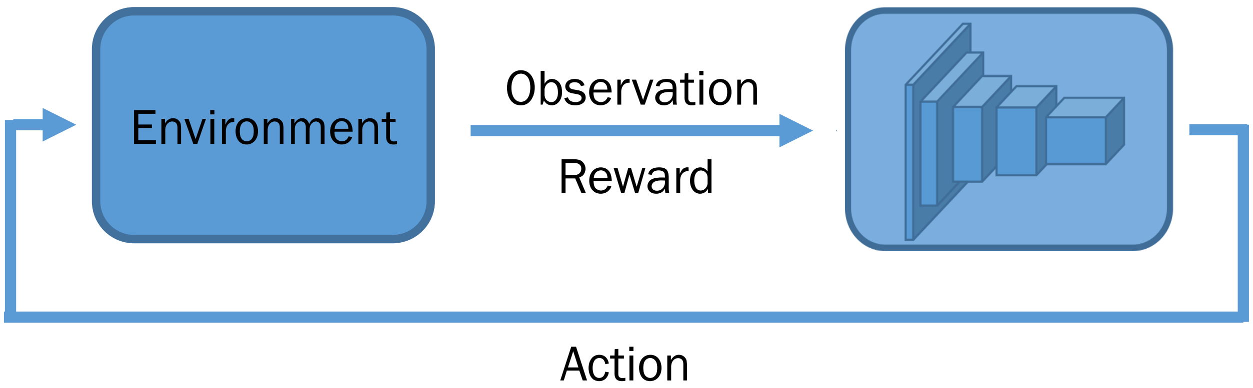
(Deep) Q-learning



Problems with end-to-end deep RL?

- Very brittle, lots of parameter tuning
- Sample-inefficient
- Single goal only
- Does not work on complex tasks

End-to-end deep reinforcement learning



- Learn from interaction with the environment
- … by maximizing the (discounted) sum of future rewards
- Directly map sensory input to returns/actions
- … with a deep network

Learning to Act by Predicting the Future

Published as a conference paper at ICLR 2017

LEARNING TO ACT BY PREDICTING THE FUTURE

Alexey Dosovitskiy
Intel Labs

Vladlen Koltun
Intel Labs

ABSTRACT

We present an approach to sensorimotor control in immersive environments. Our approach utilizes a high-dimensional sensory stream and a lower-dimensional measurement stream. The cotemporal structure of these streams provides a rich supervisory signal, which enables training a sensorimotor control model by interacting with the environment. The model is trained using supervised learning techniques, but without extraneous supervision. It learns to act based on raw sensory input from a complex three-dimensional environment. The presented formulation enables learning without a fixed goal at training time, and pursuing dynamically changing goals at test time. We conduct extensive experiments in three-dimensional simulations based on the classical first-person game Doom. The results demonstrate that the presented approach outperforms sophisticated prior formulations, particularly on challenging tasks. The results also show that trained models successfully generalize across environments and goals. A model trained using the presented approach won the Full Deathmatch track of the Visual Doom AI Competition, which was held in previously unseen environments.

Direct Future Prediction

- Control as "future-supervised" learning
- Instead of learning to maximize returns, learn to predict the future

How to represent the future?

Naïve approach: predict pixels

- Predict the future observation (image)
  - Oh et al. 2015, Finn et al. 2016, Chiappa et al. 2017, …
- Problem: uncertainty!

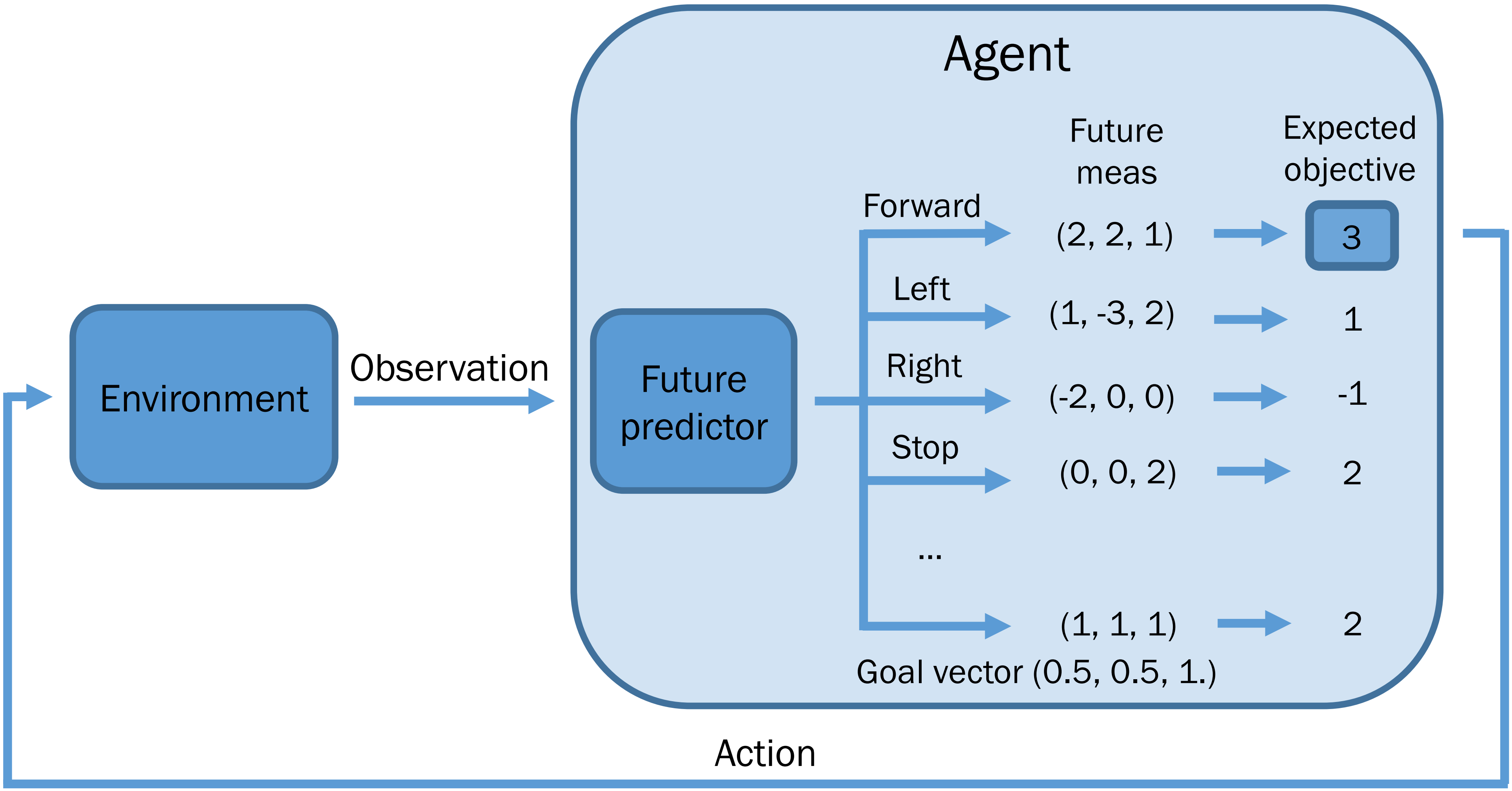
We only need to predict relevant values

Measurements

- Measurements are behaviorally relevant values available to the agent

- Assumption: goals (objective functions) can be expressed as functions of measurements

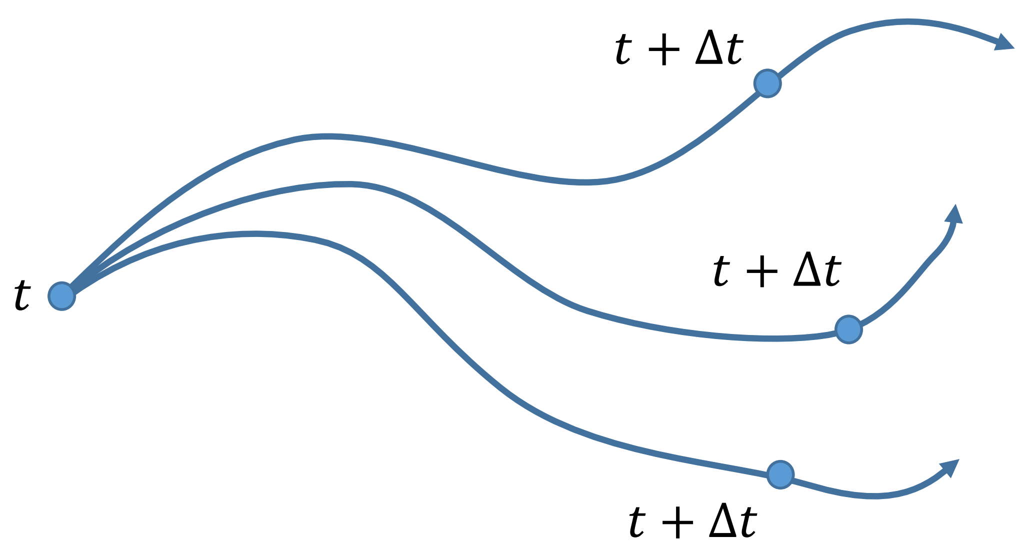
Using future predictions to act



Objective is linear in future measurements

Direct future prediction

- Predict the future measurements for each action
- Simple supervised learning

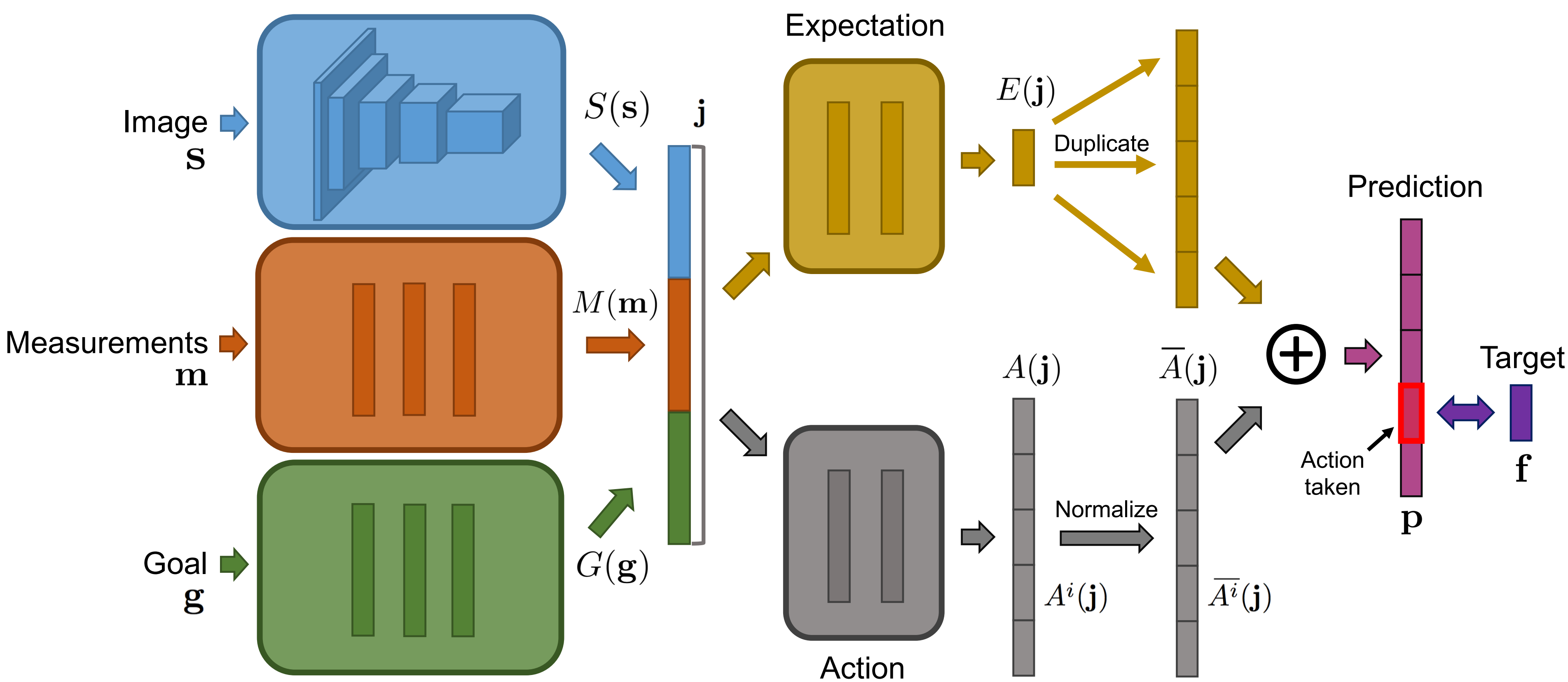


- The future is stochastic
- … and depends on the future actions

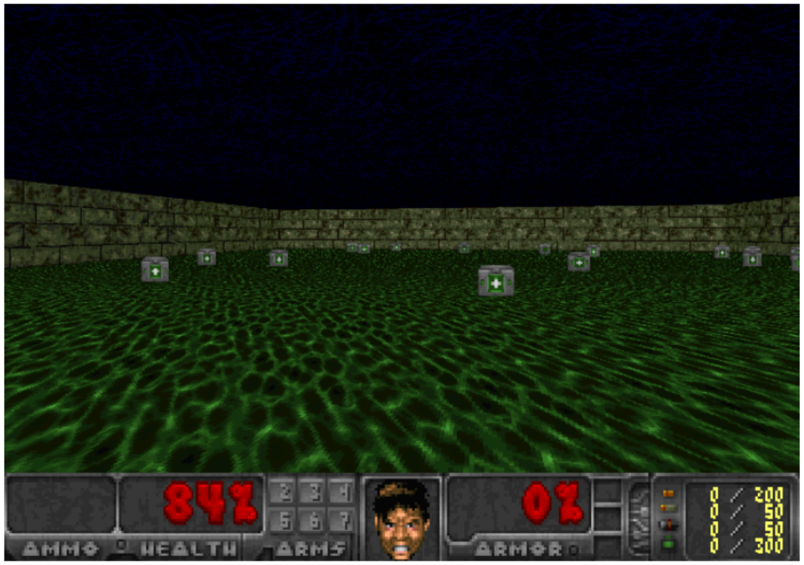
Predict expectations

On-policy

Network architecture



ViZDoom – tasks



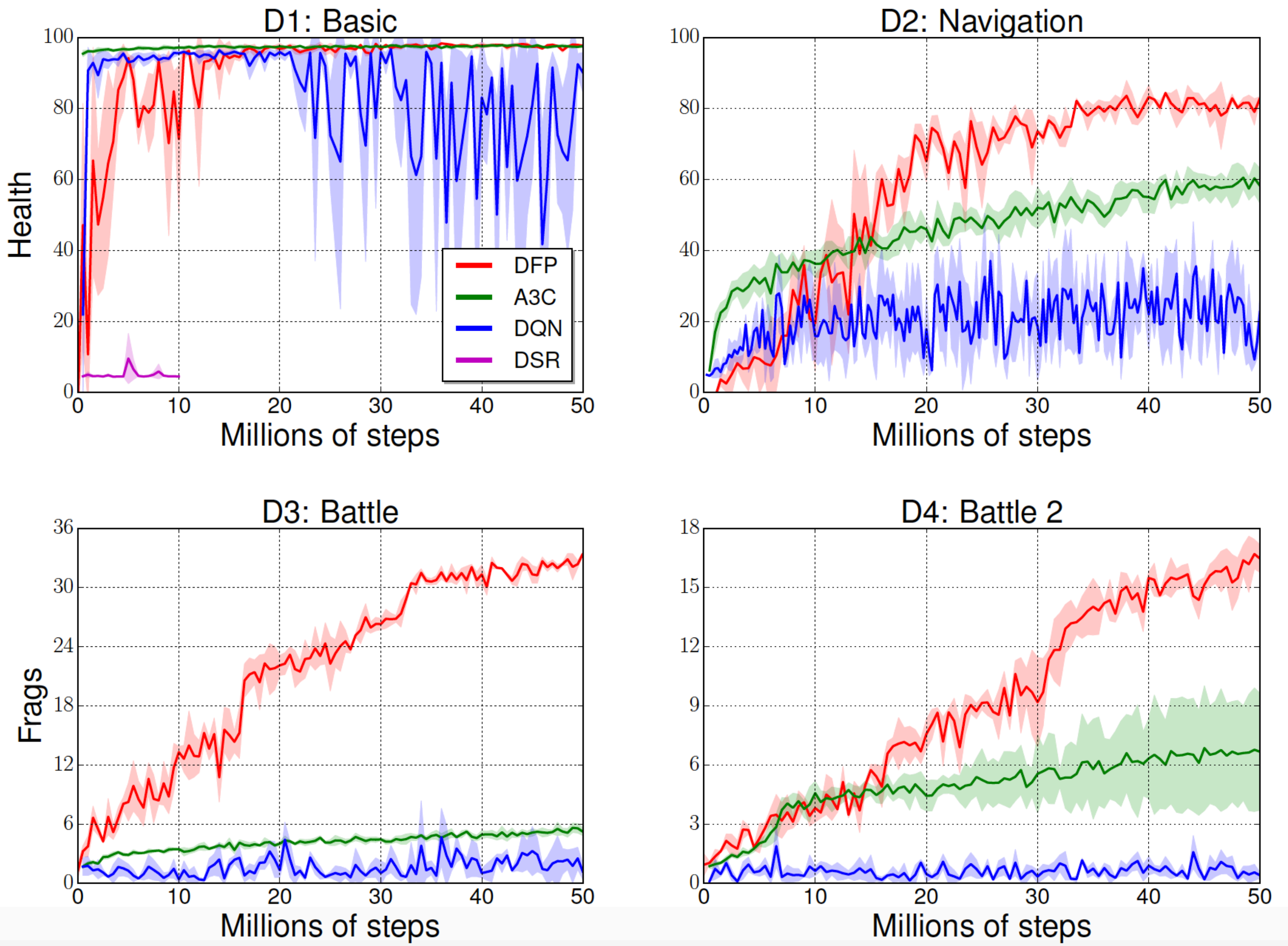
D1: Basic

D2: Navigation

D3: Battle

D4: Battle 2

Comparison to existing methods



Learning to Act by Predicting the Future

Alexey Dosovitskiy Vladlen Koltun

Generalization

This was testing on the training set

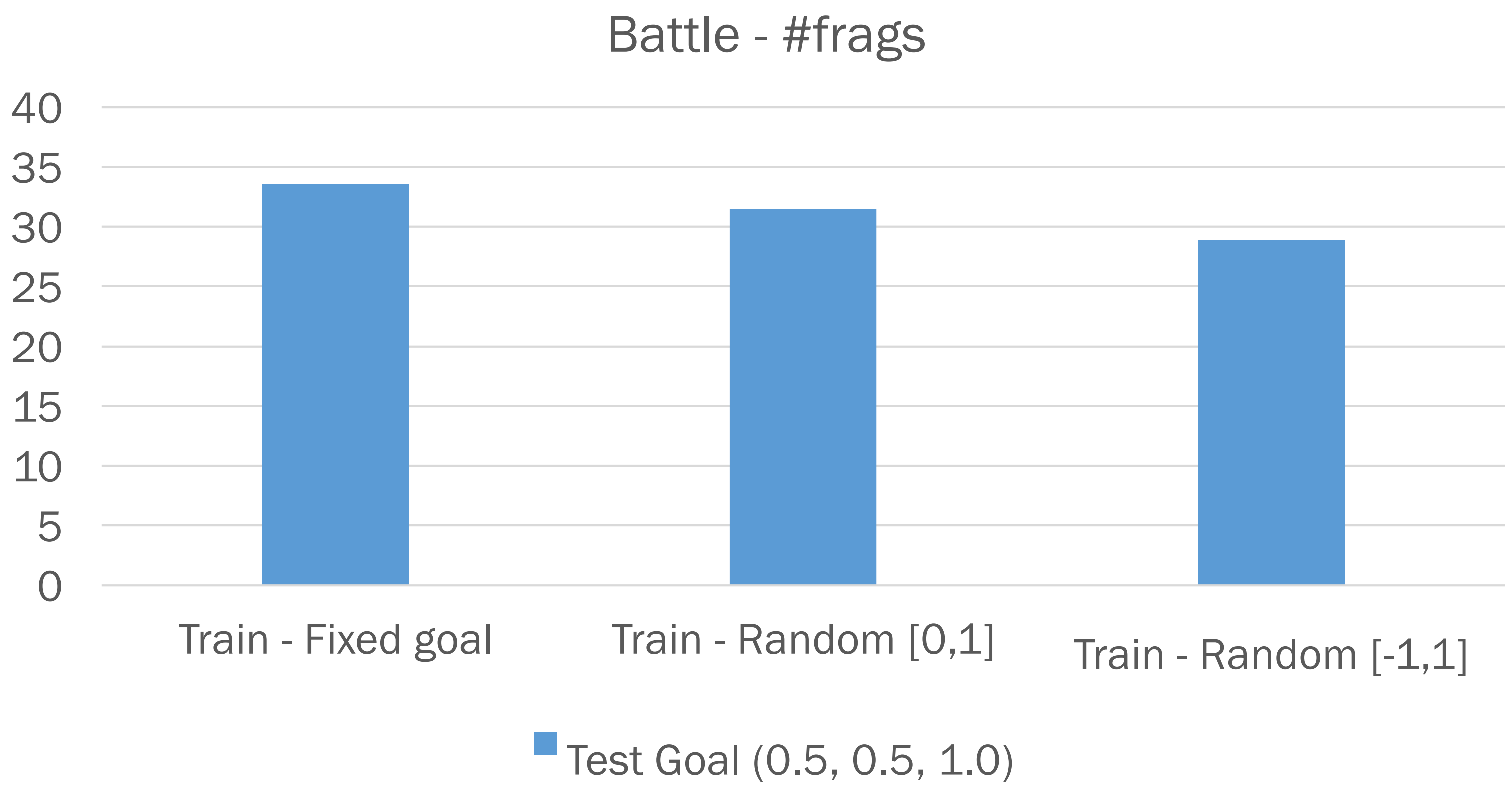
Can we generalize?

Generalization across goals

- Train with a random goal vector in every episode
  - Uniform [0,1]
  - Uniform [-1,1]

- Change the goal vector at test time
  - The end goal does not have to be known at training time!

Generalization across goals

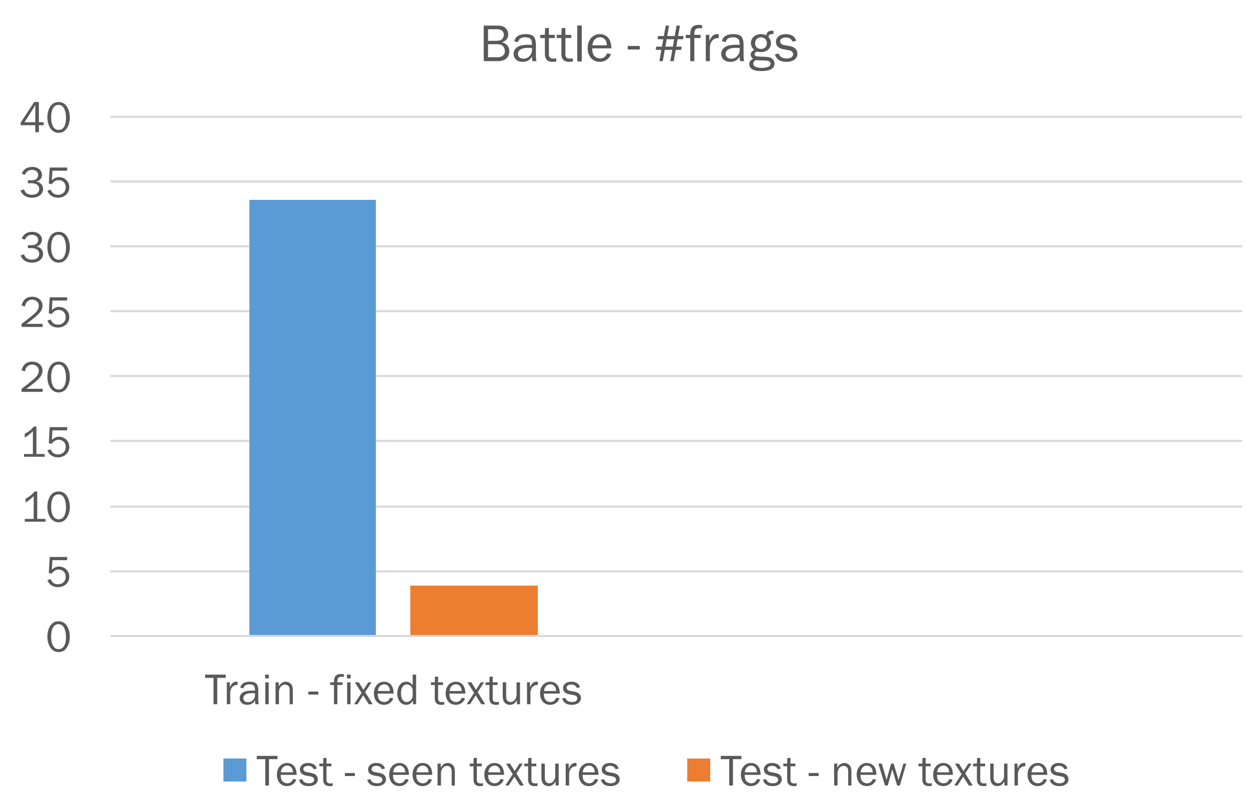


- Goal-agnostic training performs very close to training with a fixed goal
- Generalizes to different goals much better

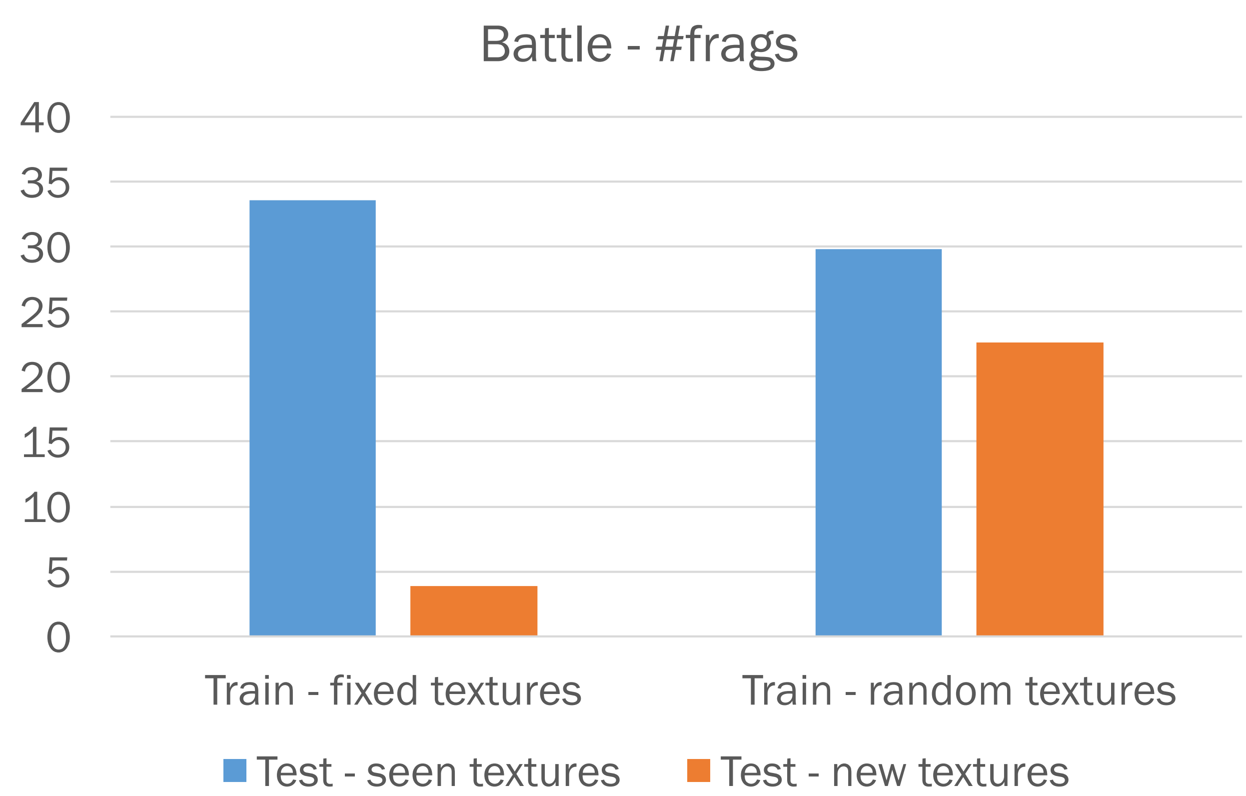
Generalization across environments

- Train with randomized textures

Generalization across environments



Generalization across environments



- Good generalization to previously unseen textures and labyrinth layouts

ViZDoom Competition: Full Deathmatch

Place	Team	1	2	3	4	5	6	7	8	9	10	11	12	Total
1	**IntelAct**	29	21	23	21	6	11	9	6	30	32	33	35	256
2	**The Terminators**	22	17	21	15	13	12	6	5	14	13	13	13	164
3	**TUHO**	8	11	13	12	0	-1	-1	-4	2	2	6	3	51
4	ColbyCS	2	4	0	1	-1	0	-1	0	3	3	4	3	18
5	5vision	3	0	4	2	1	0	1	0	0	-1	1	1	12
6	Ivomi	3	0	1	0	1	-1	-4	-4	1	1	0	0	-2
7	PotatoesArePrettyC	0	0	2	0	-1	-3	-1	0	-2	-1	-1	-2	-9

Summary

- Simple finite-horizon supervised training performs very well on visuomotor control tasks

- Predicting measurements instead of rewards:
  - Better training signal
  - Flexible goal setting, goal-agnostic learning

- Training with random textures leads to good generalization across environments

Are we done?

- We can learn relatively difficult tasks in Doom
  - … but even in Doom algorithms fail in complex scenarios, e.g. requiring reasoning or memory
  - … and Doom is unrealistic and simplistic

?

Better task?

- Requires complex perception, planning, control, memory, mapping, reasoning
- Real problem, there is no way to cheat or hack
- Difficult even for humans

CARLA: An Open Urban Driving Simulator

CARLA: An Open Urban Driving Simulator

Alexey Dosovitskiy German Ros Felipe Codevilla Antonio Lopez Vladlen Koltun

Abstract: We introduce CARLA, an open-source simulator for autonomous driving research. CARLA has been developed from the ground up to support training, prototyping, and validation of autonomous urban driving models, including both perception and control. In addition to open-source code and protocols, CARLA provides open digital assets (urban layouts, buildings, vehicles, pedestrians, etc.) that were created specifically for this purpose and can be used and redistributed freely. The simulation platform supports flexible specification of sensor suites and a wide range of environmental conditions. Using the presented simulation platform and content, we study the performance of two approaches to autonomous urban driving: a classic modular rule-based pipeline and an end-to-end model trained via imitation learning. The approaches are evaluated on a series of controlled scenarios of increasing difficulty, and their performance is examined in detail via metrics provided by the platform, illustrating the platform's utility for research on autonomous urban driving.

To appear soon

CARLA: An Open Urban Driving Simulator

To be released soon

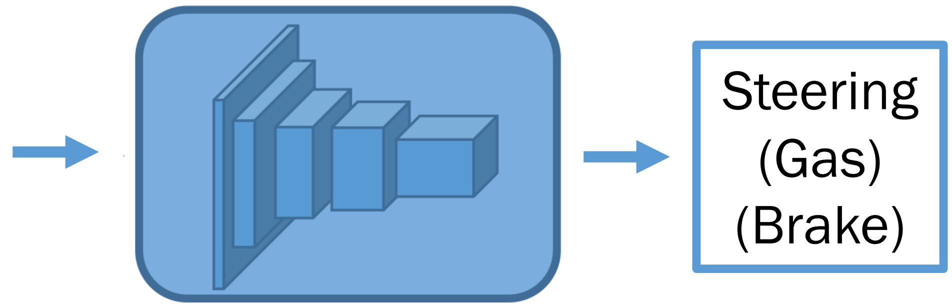
How do we learn driving?

From experience?

Fails so far!

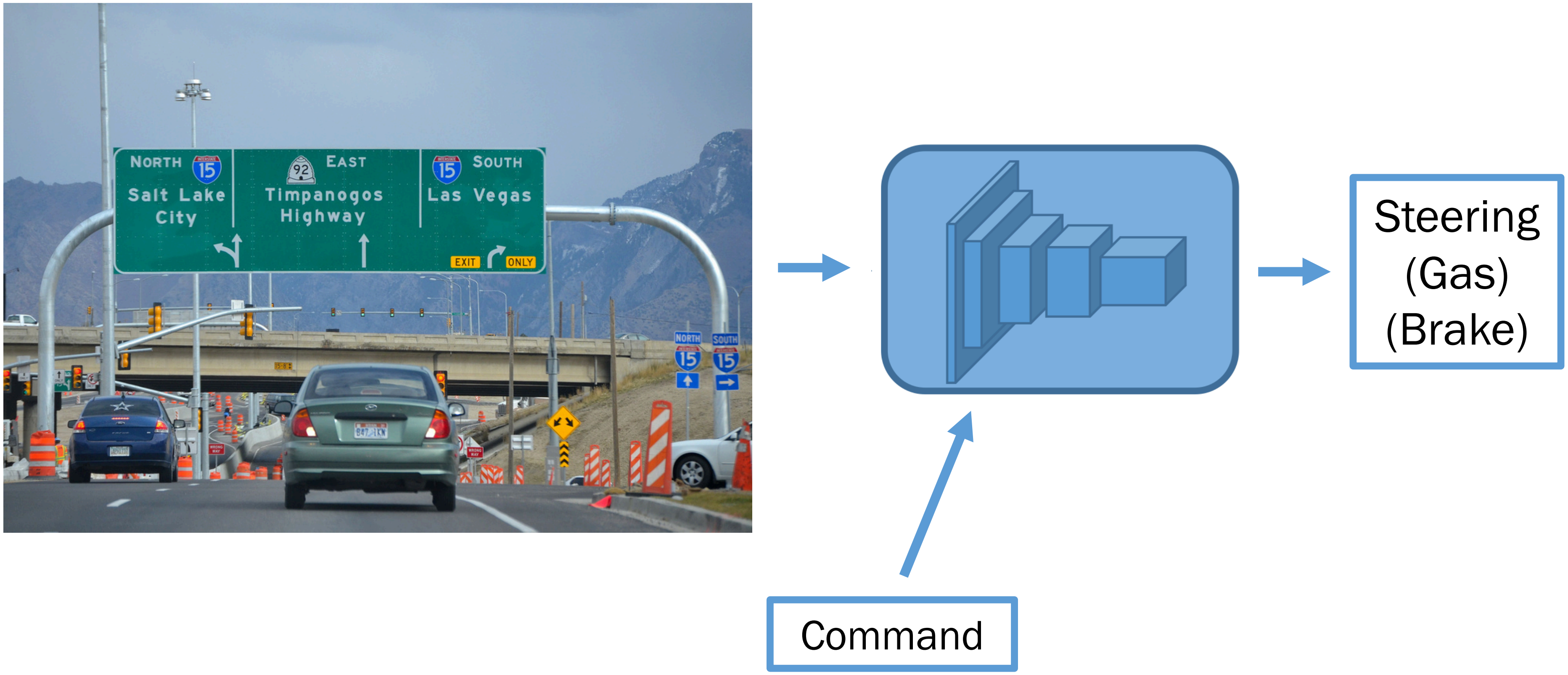
From demonstration!

Imitation learning



- Training data recorded from human drivers
- An old idea:
  - ALVINN - Pomerleau, NIPS 1988
  - DAVE - LeCun et al., NIPS 2005
  - "DAVE 2" - Bojarski et al. arxiv 2016

Conditional imitation learning



- Problem: no way to control the system
- Idea: feed commands to the ConvNet

Conditional imitation learning

End-to-end Driving via Conditional Imitation Learning

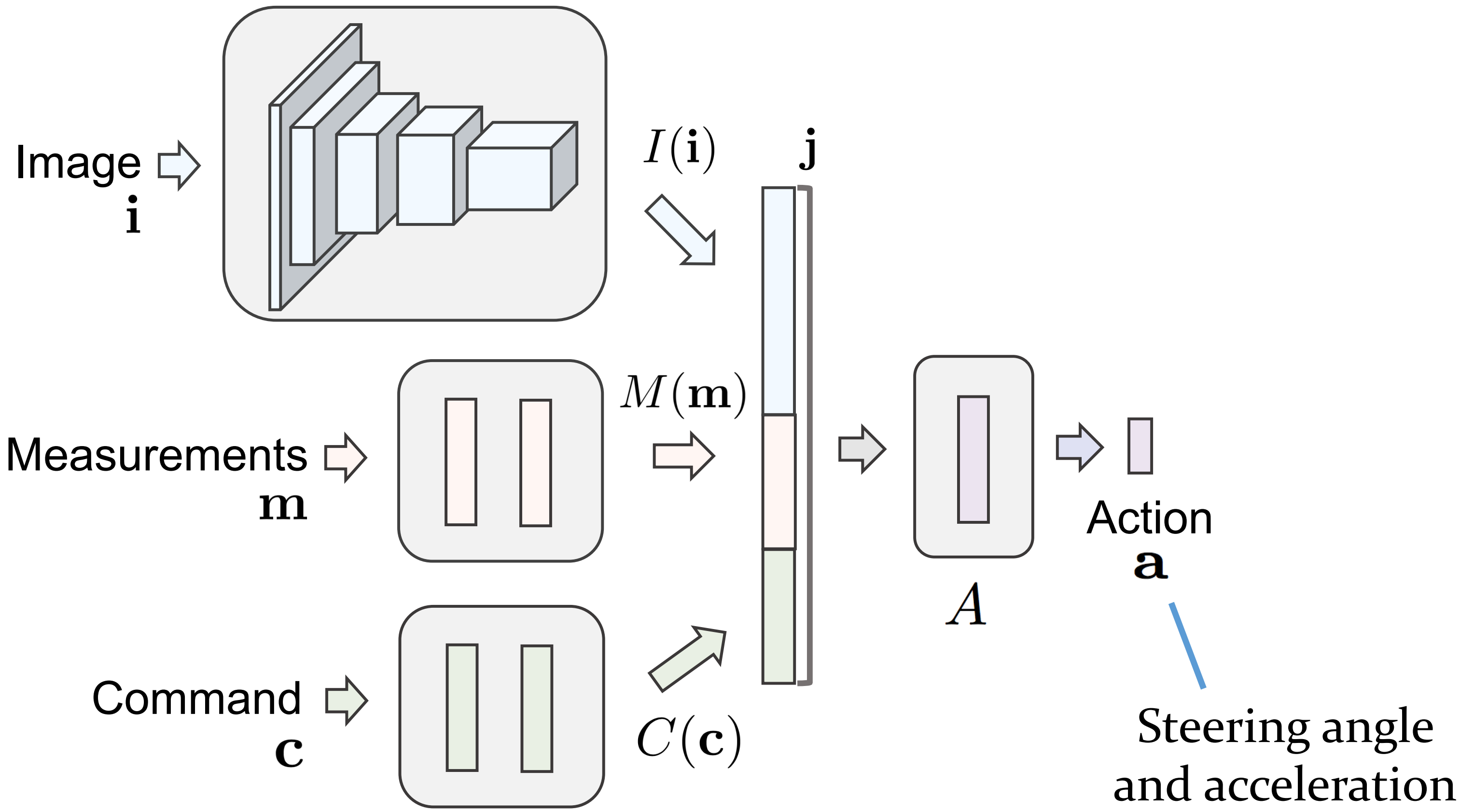
Felipe Codevilla[1,2] Matthias Müller[1,3] Alexey Dosovitskiy[1] Antonio López[2] Vladlen Koltun[1]

(a) Aerial view of test environment (b) Vision-based driving, view from onboard camera (c) Side view of vehicle

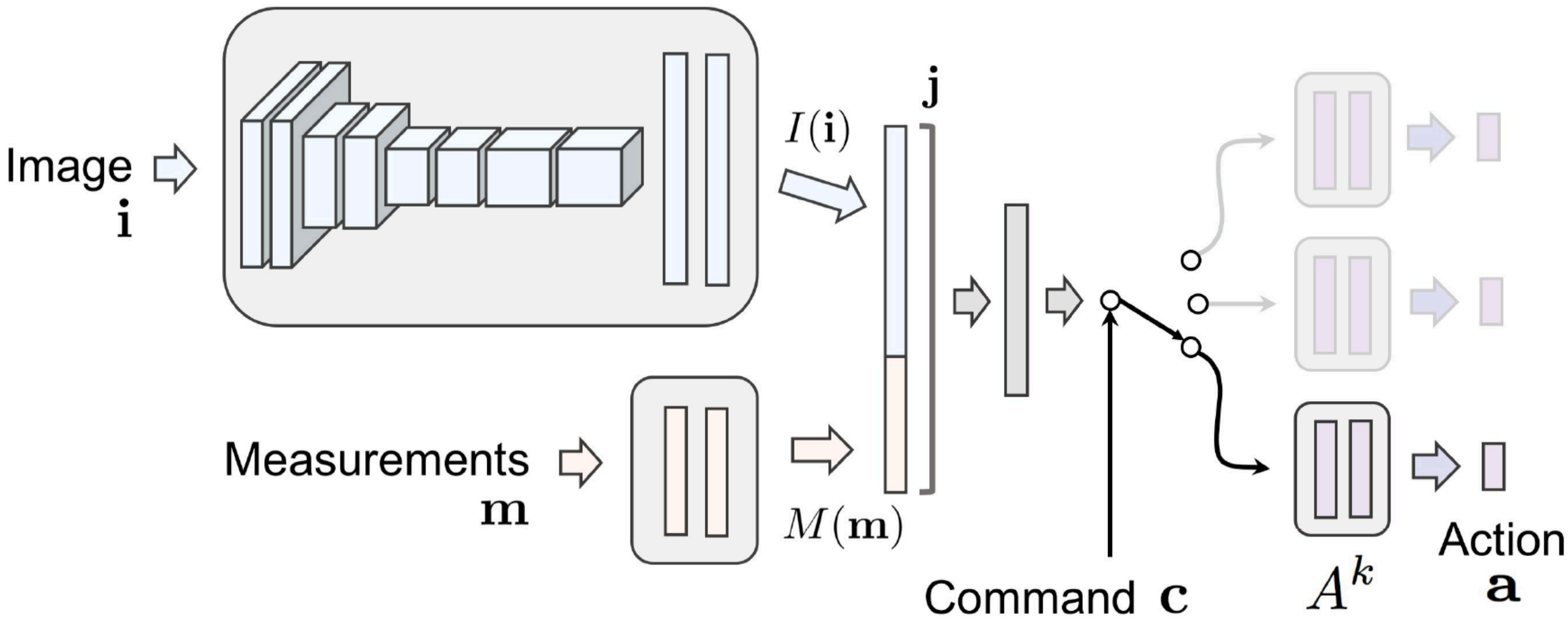
Fig. 1. Conditional imitation learning allows an autonomous vehicle trained end-to-end to be directed by high-level commands. (a) We train and evaluate robotic vehicles in the physical world (top) and in simulated urban environments (bottom). (b) The vehicles drive based on video from a forward-facing onboard camera. At the time these images were taken, the vehicle was given the command "turn right at the next intersection". (c) The trained controller handles sensorimotor coordination (staying on the road, avoiding collisions) and follows the provided commands.

Under submission

Basic architecture



Branched architecture



- 3 commands: “straight”, “left”, “right”
- A “specialist” branch for each command
- Command input acts as a switch

Conditional imitation learning - Video

Autonomous driving solved?

- No!
  - In CARLA reaches ~90% goals and crashes every ~3 km
  - Far from the desired 99.9999% quality

Summary

- Imitation learning works surprisingly well on difficult sensorimotor control tasks
- Command-conditional imitation learning:
  - Controllable by the user
  - If combined with a navigation device, makes a fully autonomous driving system
- Still a very long way to go!