Navigation

Saurabh Gupta

Tasks

- Locomotion
  - Wheeled ground vehicle
  - Drone
  - Quadruped
  - Off-road terrain

Tasks

- Locomotion
  - Platforms:
    - Wheeled ground vehicle, Drone, Quadrupeds, Off-road terrain
  - Move safely, and gracefully. Avoid collisions.
    - Follow high-level commands
    - Visual Servoing, follow an object
- Goal-directed behavior
  - Goals known / unknown
  - Environment seen / novel

Axes of variation

Supervision Signal

Expert Trajectories ← → Reward Signals ← → Ecological Supervision ← → No supervision (SIFT + Geometry)

Environment Modeling

No Modeling (Reactive Policies) ← → Implicit Modeling (via LSTMs) ← → Explicit modeling of space (metric / topological maps)

Task

Locomotion (move safely) ← → Path following ← → Long range goals (points / semantics / visual) ← → Navigation for actions

Generalization

Extensive training on test environments ← → Drop in novel environment

Axes of variation

Abstraction

State space discretization (eg. grid world) ⟷ Discrete Actions ⟷ Continuous actions

Environment Properties

Random Mazes (limited semantics) ⟷ Real world environments

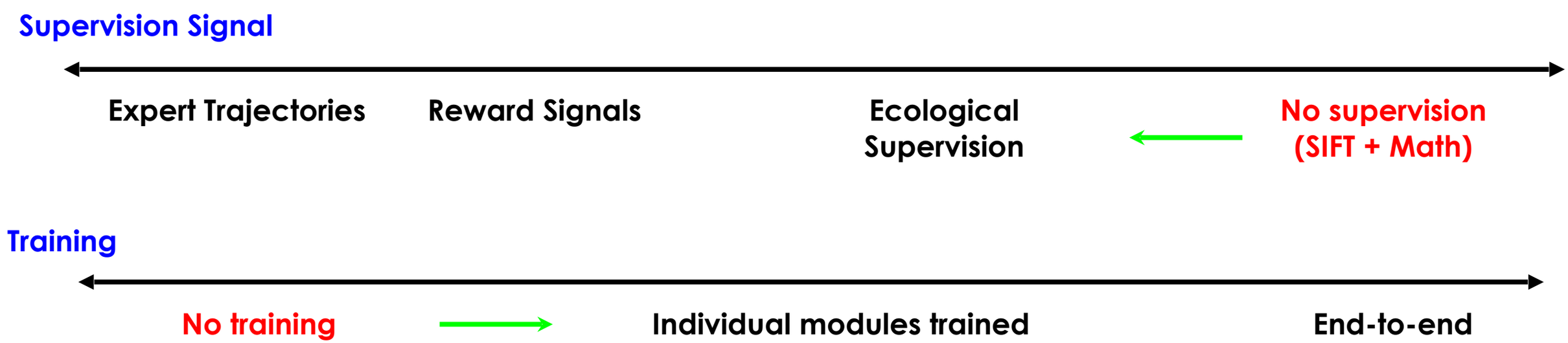
Experimental setup

Simulation ⟷ Simulation (based of real world environments) ⟷ Real world

Training

No training ⟷ Individual modules trained ⟷ End-to-end

CNNs for SLAM



Classical SLAM

- **Localization:**
  - **Semantic Visual Localization** J. Schonberger, M. Pollefeys, A. Geiger, T. Sattler arXiv 17 pdf
  - **Self-Supervised Place Recognition in Mobile Robots** S. Pillai, J. Leonard. IROS17 pdf
  - **Active neural localization** D. Chaplot arXiv17 pdf
- **Visual Odometry**
  - **Towards Visual Ego-motion Learning in Robots.** S. Pillai and J. Leonard IROS17 pdf
  - **Unsupervised Learning of Depth and Ego-motion from Video** Zhou et al. CVPR17 pdf
- **Reconstruction**
  - **Learned Stereo Machines.** Kar et al. NIPS17 pdf
  - **Differentiable ray consistency.** Tulsiani et al. CVPR17 pdf

Learned Locomotion

Environment Modeling

No Modeling (Reactive Policies) ⟷ Implicit Modeling (via LSTMs) ⟷ Explicit modeling of space (metric / topological maps)

Task

Locomotion (move safely) ⟷ Path following ⟷ Long range goals (points / semantics / visual) ⟷ Navigation for actions

Generalization

Extensive training on test environments ⟷ Drop in novel environment

Abstraction

State space discretization (eg. grid world) ⟷ Discrete Actions ⟷ Continuous actions

- **Learning Agile and Dynamic Motor Skills for Legged Robots** J. Hwango et al. Science Robotics 18 pdf, video
- **Agile Autonomous Driving** Y. Pan et al. R:SS18 pdf, video
- **Drone Racing** E. Kaufmann et al. ICRA19 pdf, video
- **Self-supervised Deep Reinforcement Learning with Generalized Computation Graphs for Robot Navigation** G. Kahn, A. Villaflor, B. Ding, P. Abbeel, S. Levine. ICRA18 pdf, video
- **Learning to Fly by Crashing** D. Gandhi, L. Pinto, A. Gupta. IROS17 pdf, video

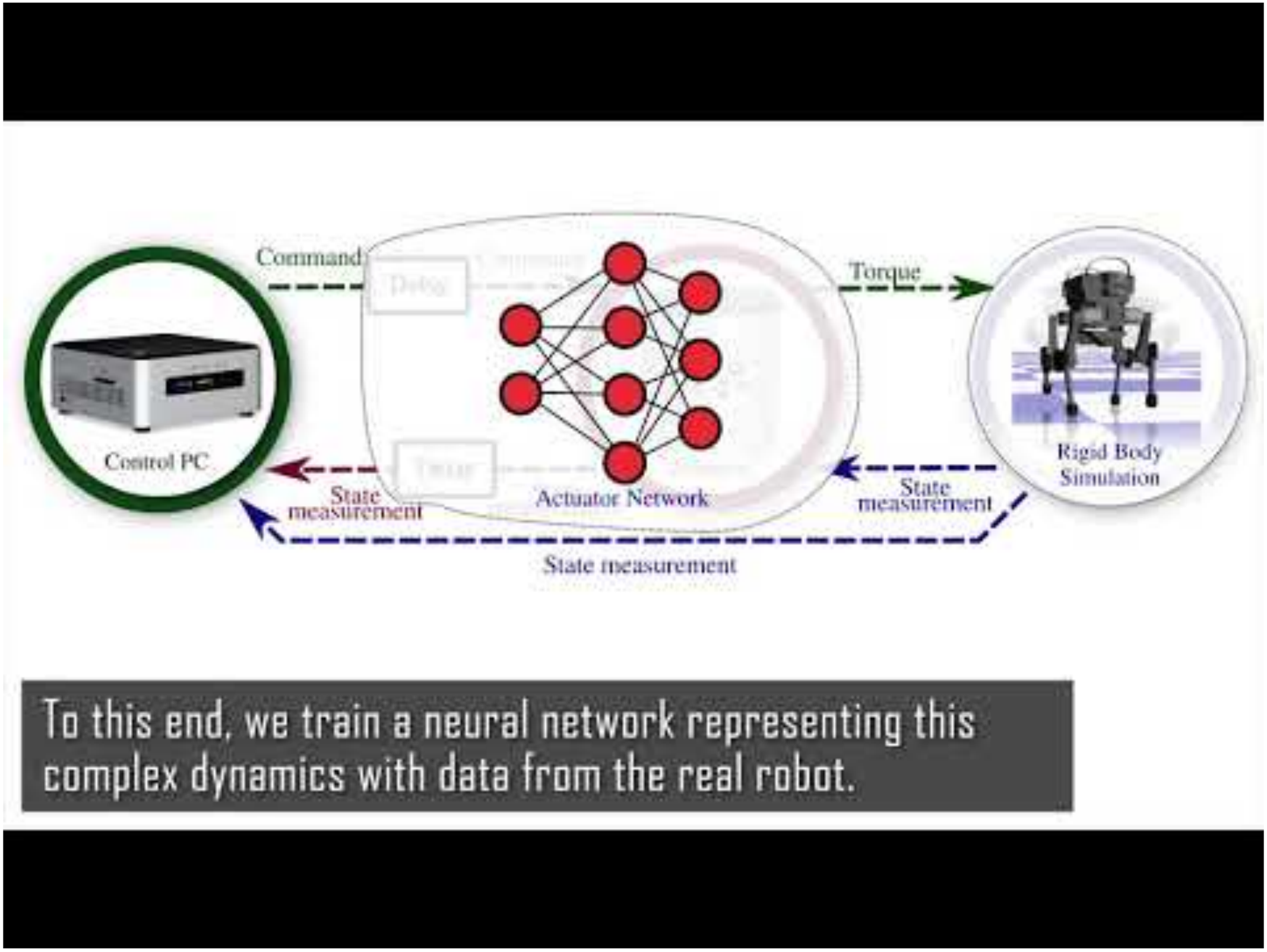
Rewards Signals, Simulation

Imitation Learning

Modularized Learning

Ecological Supervision

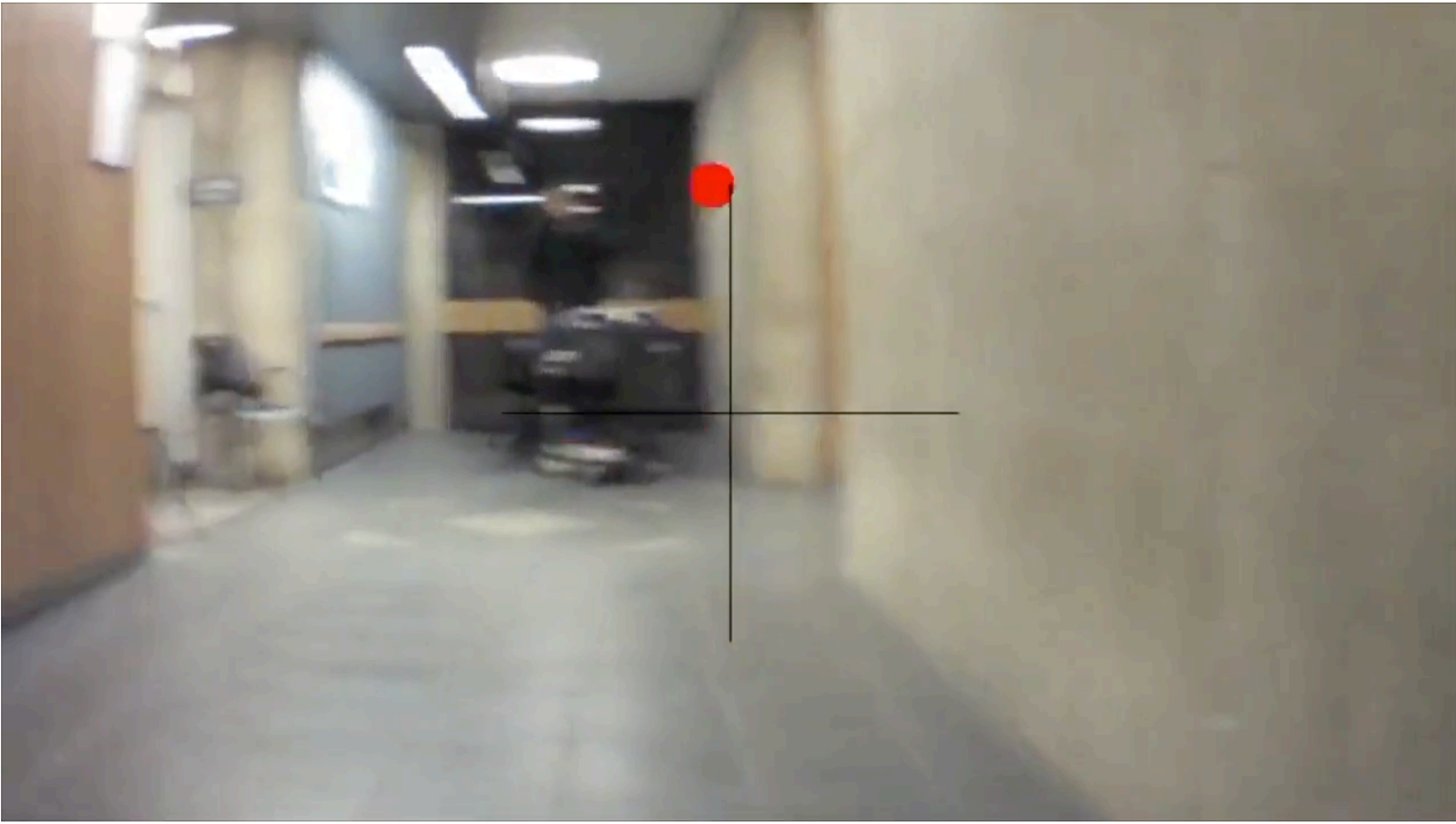
- **Learning Agile and Dynamic Motor Skills for Legged Robots** J. Hwango et al. Science Robotics 18 pdf, video



- **Agile Autonomous Driving** Y. Pan et al. R:SS18 pdf, video

- **Beauty and the Beast: Optimal Methods Meet Learning for Drone Racing** E. Kaufmann et al. ICRA19 pdf, video

Learning to Fly by Crashing D. Gandhi, L. Pinto, A. Gupta. IROS17. pdf, video



Self-supervised Deep Reinforcement Learning with Generalized Computation Graphs for Robot Navigation G. Kahn S.Levine. ICRA18. pdf, video

Self-supervised Deep Reinforcement Learning with Generalized Computation Graphs for Robot Navigation

Gregory Kahn, Adam Villaflor, Bosen Ding, Pieter Abbeel, Sergey Levine

BAIR

BERKELEY ARTIFICIAL INTELLIGENCE RESEARCH

Goal directed navigation

- Goal-directed behavior
  - Goals known / unknown
  - Environment seen / novel
- Central question is that of how to represent space?
  - Reason about what you have seen and where / what you have not seen
  - Spatial reasoning
  - Robustness to actuation noise
- For unknown environments, we may additionally need:
  - Semantic reasoning
  - Easy to build

Representations for Space

- Metric maps
  - Occupancy maps
  - Occupancy maps flavored with semantic information
- Topological maps
  - Collection of images, and connectivity information

Learned Task driven End-to-end navigation

Supervision Signal

Expert Trajectories | **Reward Signals** | **Ecological Supervision** | **No supervision (SIFT +Math)**

Task

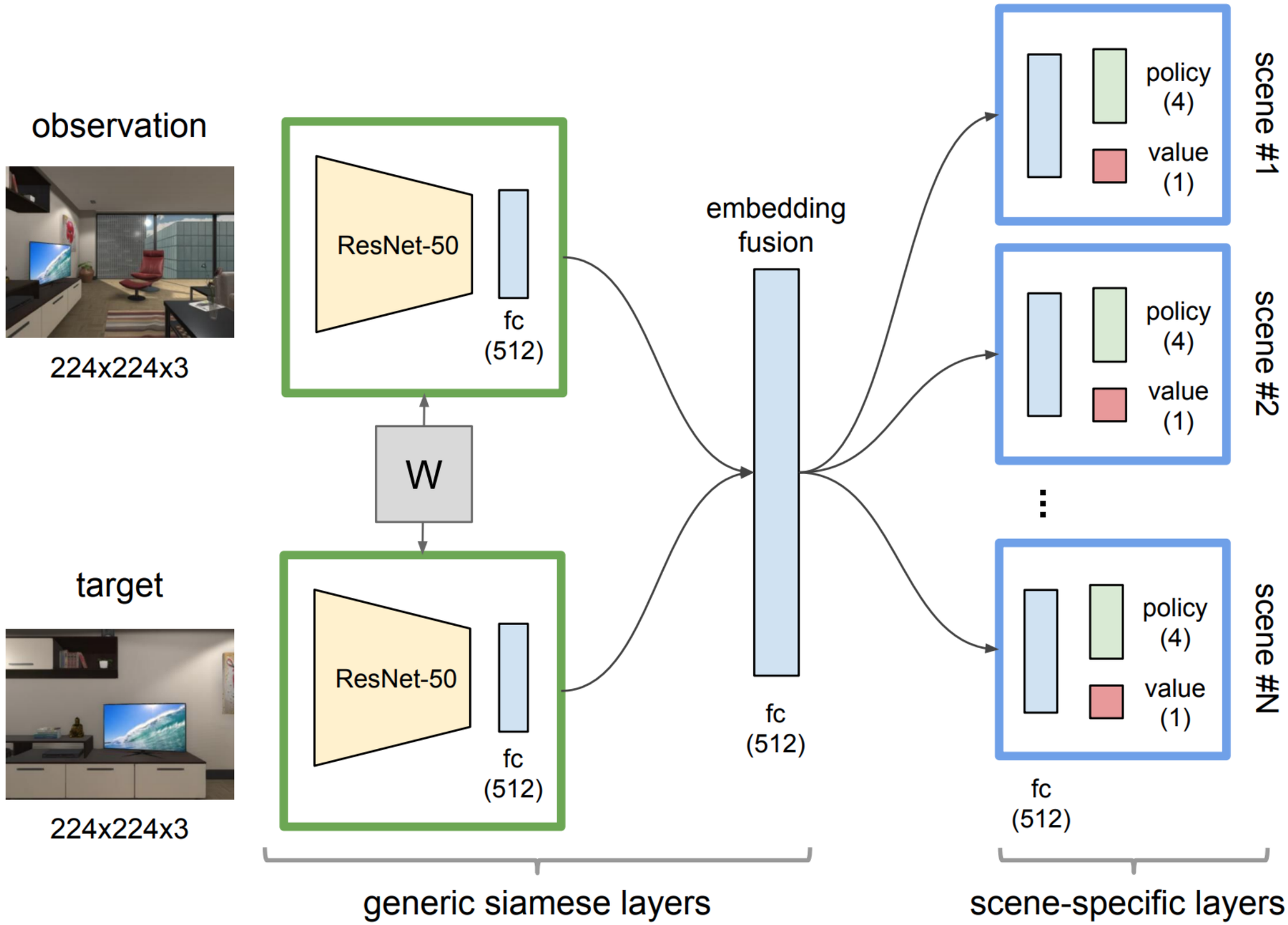
Locomotion (move safely) | **Path following** | **Long range goals (points / semantics / visual)** | **Navigation for actions**

- **Target driven visual navigation in indoor scenes using deep reinforcement learning.** Zhu, … Farhadi ICRA17 pdf, video
- **Learning to navigate in complex environments** P. Mirowski, … R. Hadsell ICLR17 pdf, video
- **Cognitive Mapping and Planning** CVPR17 pdf, video
- **Neural Topological SLAM** CVPR 20 pdf, video

Environment Modeling	Generalization
No Modeling (Reactive Policies)	Extensive training on test environments
Implicit Modeling (via LSTMs)	
Explicit modeling of space (metric / topological maps)	Drop in novel environment

Target driven visual navigation in indoor scenes using deep reinforcement learning.

Y. Zhu, …, A. Gupta, …, A. Farhadi. ICRA17. pdf, video



Simulation Environments

Abstraction

State space discretization (eg. grid world) ⟷ **Discrete Actions** ⟷ **Continuous actions**

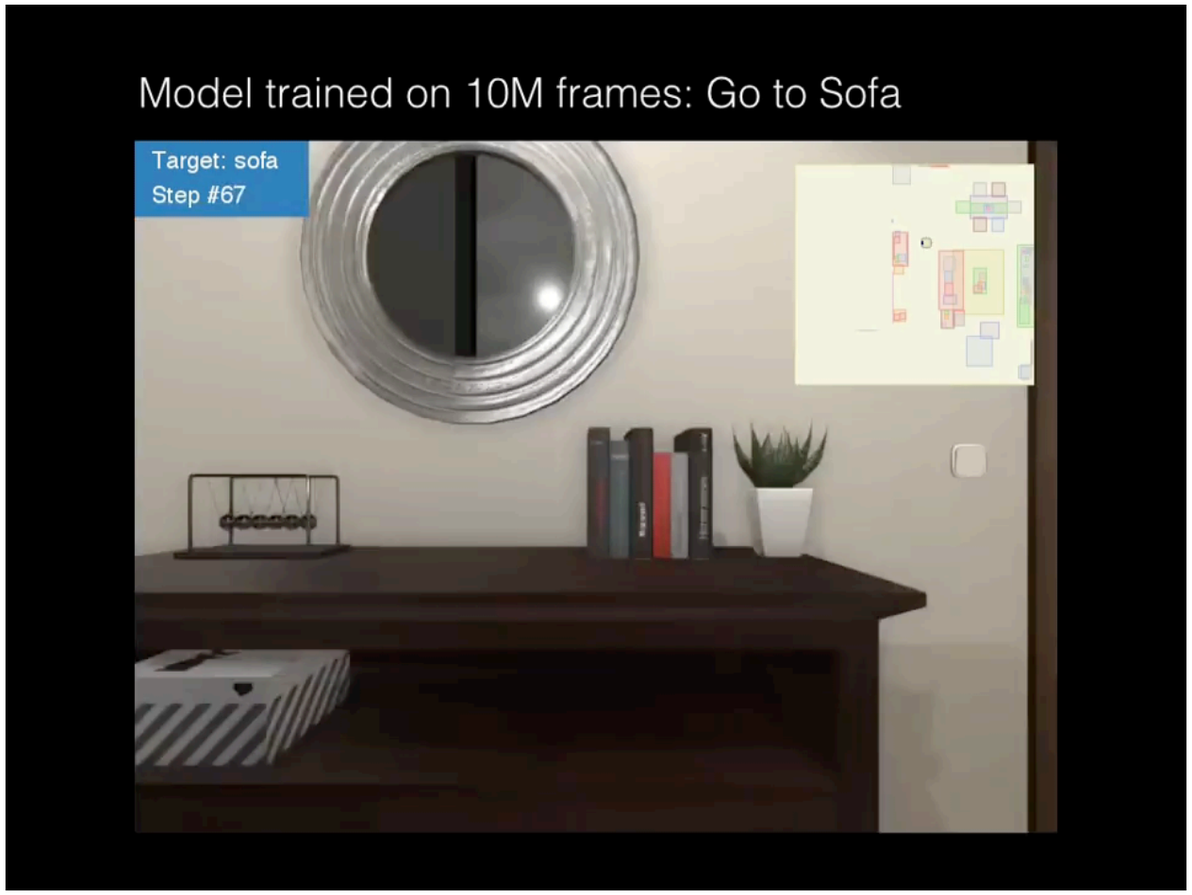
Environment Properties

Random Mazes (limited semantics) ⟷ **Real world environments**

Experimental setup

Simulation ⟷ **Simulation (based of real world environments)** ⟷ **Real world**

Simulators based off games / CG (eg: AI2Thor, VizDoom, CARLA)



Simulators based off real world scans (eg: CMPSim, Habitat, Gibson)

Visual Servoing

Environment Modeling

No Modeling (Reactive Policies) — **Implicit Modeling (via LSTMs)** — **Explicit modeling of space (metric / topological maps)**

Task

Locomotion (move safely) — **Path following** — **Long range goals (points / semantics / visual)** — **Navigation for actions**

Supervision Signal

- **Expert Trajectories**
- **Reward Signals**
- **Ecological Supervision**
- **No supervision (SIFT + Geometry)**

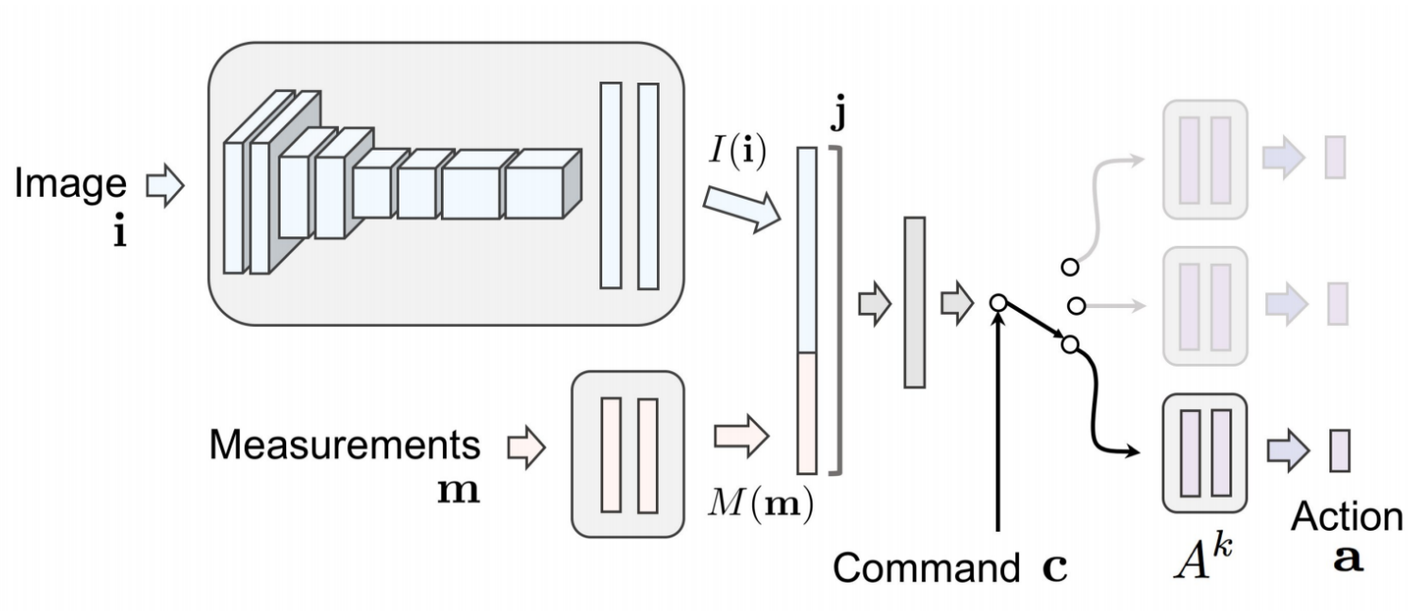
- **Visual Memory for Robust Path Following**
  Kumar, Gupta, ... Malik
- **End-to-end Driving via Conditional Imitation Learning** F. Codevilla, … V. Koltun ICRA18? pdf, video
- **Learning Visual Servoing with Deep Features and Fitted Q-Iteration**, Alex X. Lee, Sergey Levine, Pieter Abbeel. ICLR17. pdf
- **Zero-Shot Visual Imitation** D. Pathak, …, T. Darrell pdf, video ICLR18? pdf, video
- **Classical Visual Servoing**

Learning Visual Servoing with Deep Features and Fitted Q-Iteration

Alex X. Lee, Sergey Levine, Pieter Abbeel. ICLR17. pdf

End-to-end Driving via Conditional Imitation Learning

F. Codevilla, … V. Koltun ICRA18? pdf, video



Other Unexplored Axes?

Environment Scale

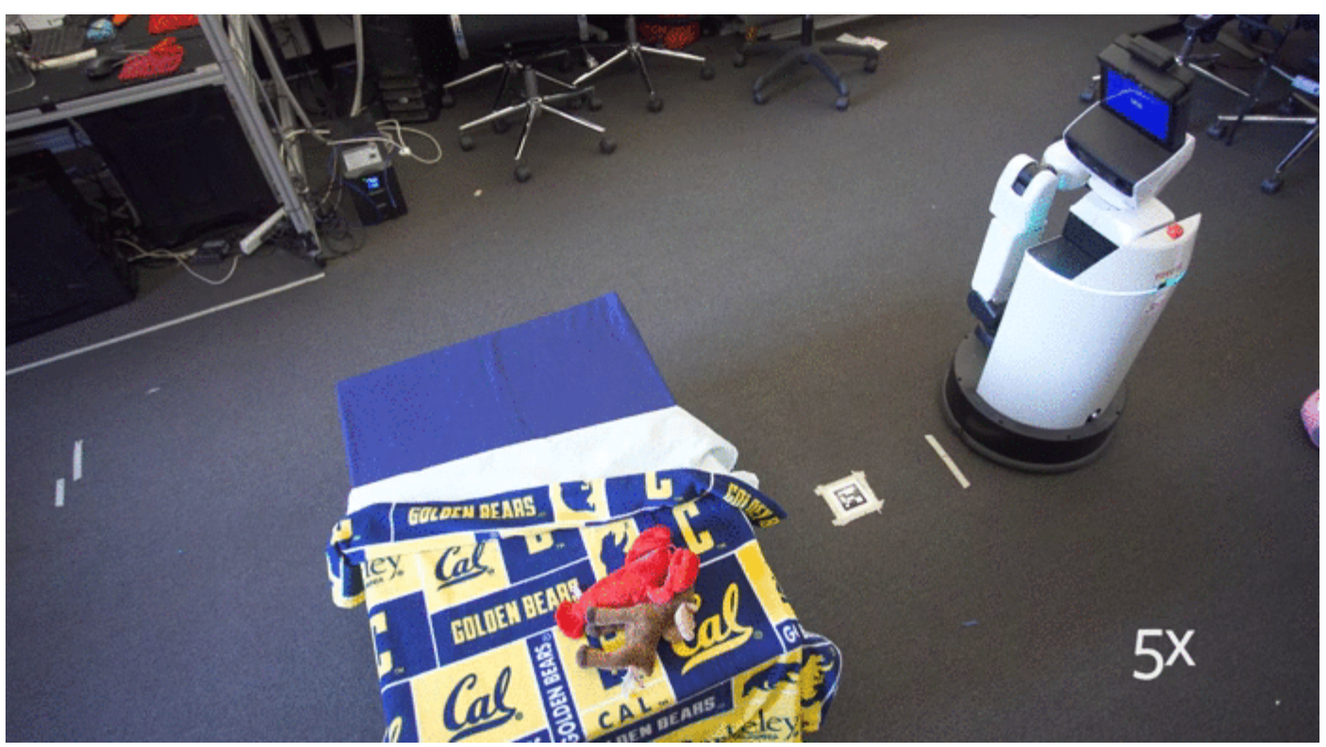
Small mazes — Houses / Offices — City Scale Environments?

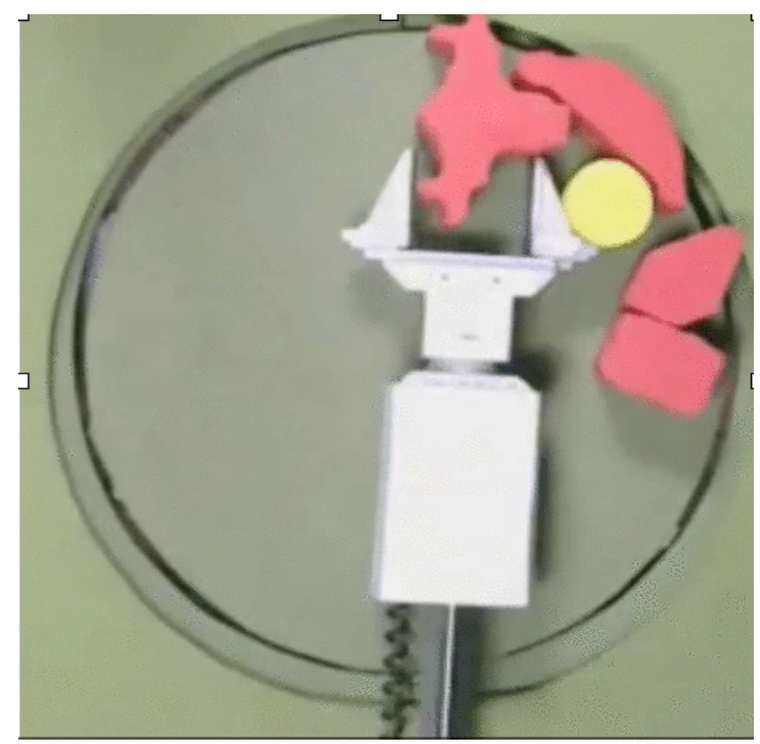
Task

Locomotion (move safely) — Path following — Long range goals (points / semantics / visual) — Navigation for actions?

Dynamic Environments

World is static — World has objects that need to be moved around? — Reason about other agents?





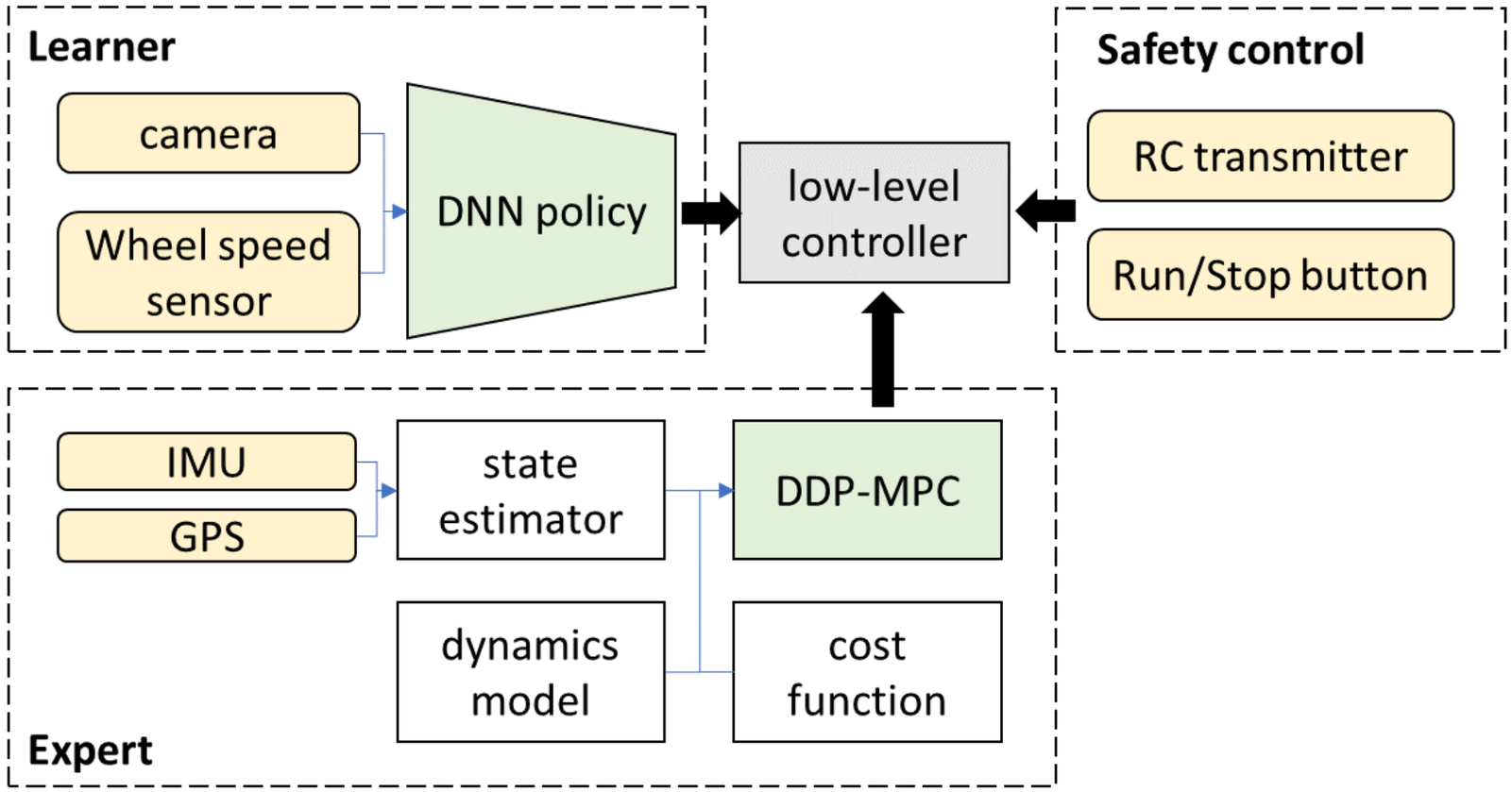
Discussion

- *What if any is the role of learning for solving these different navigation tasks?*

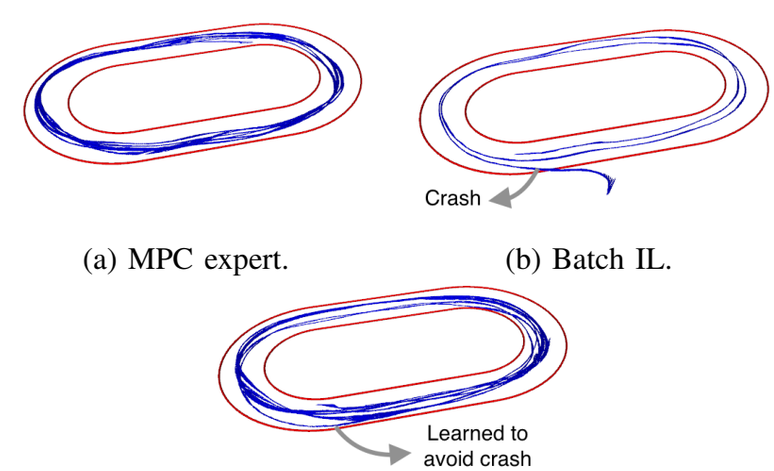
Agile Autonomous Driving using End-to-End Deep Imitation Learning

Yunpeng Pan*‡, Ching-An Cheng*, Kamil Saigol*, Keuntaek Lee†, Xinyan Yan*,
Evangelos A. Theodorou*, and Byron Boots*
*Institute for Robotics and Intelligent Machines, †School of Electrical and Computer Engineering
Georgia Institute of Technology, Atlanta, Georgia 30332–0250
‡JD.com American Technologies Corporation, Mountain View, California 94043
{ypan37,cacheng,kamilsaigol,keuntaek.lee,xyan43}@gatech.edu
evangelos.theodorou@gatech.edu, bboots@cc.gatech.edu

Overview

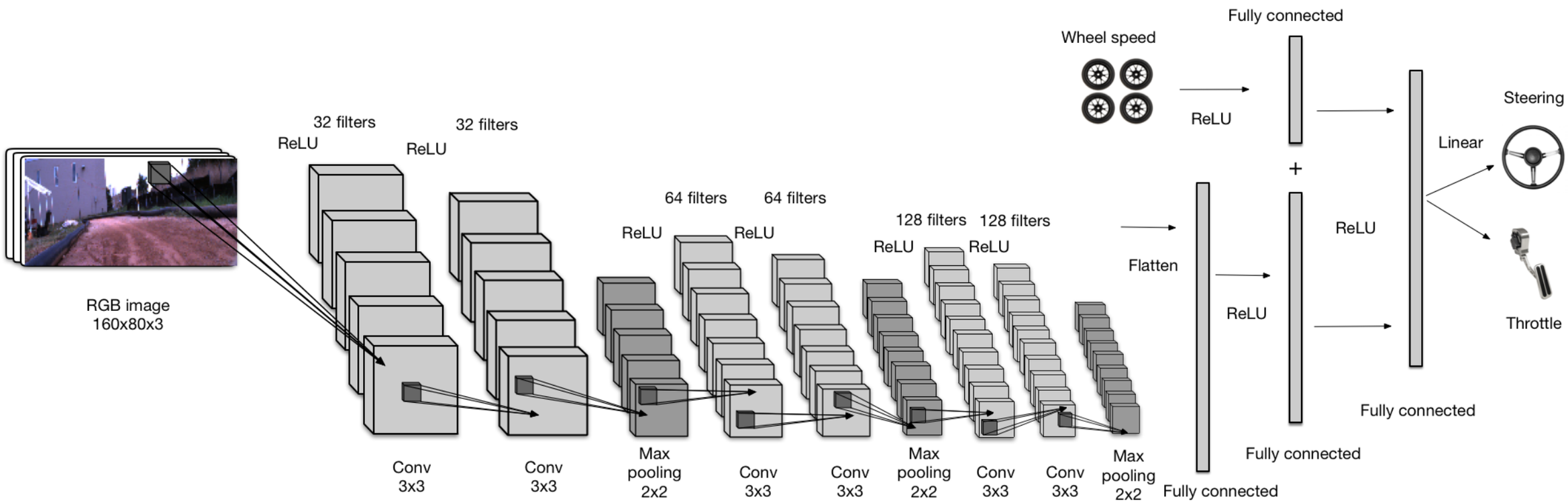


- Algorithmic expert uses MPC with expensive state-estimators
- Learner predicts controls output by expert
- Use DAgger, instead of vanilla behavior cloning



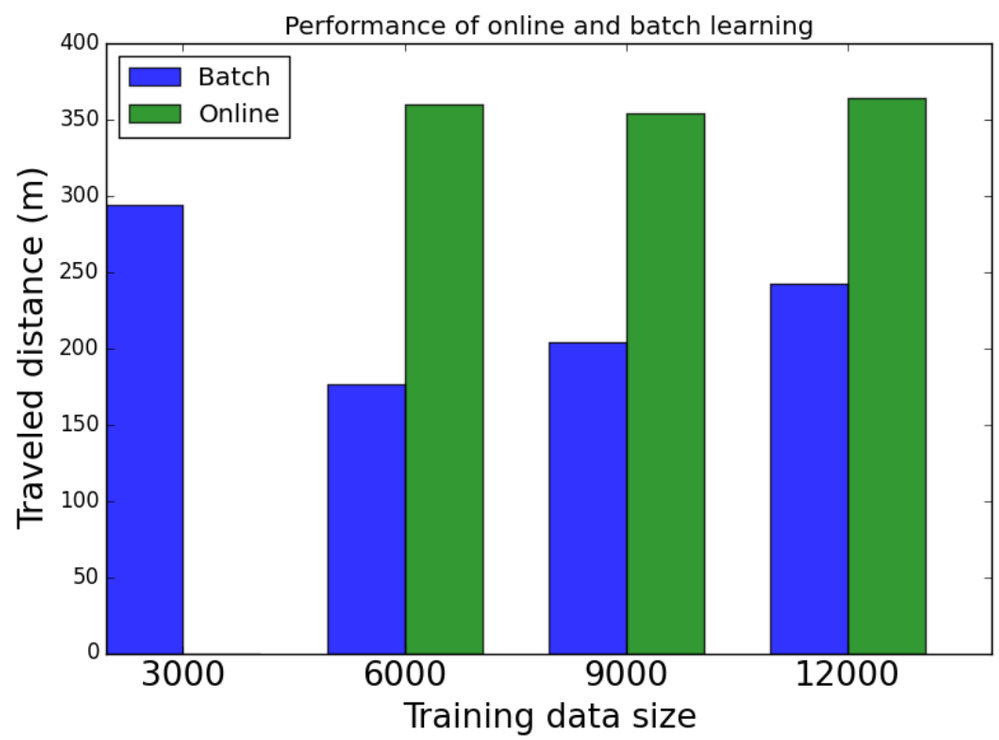
(a) MPC expert. (b) Batch IL. (c) Online IL.

Policy

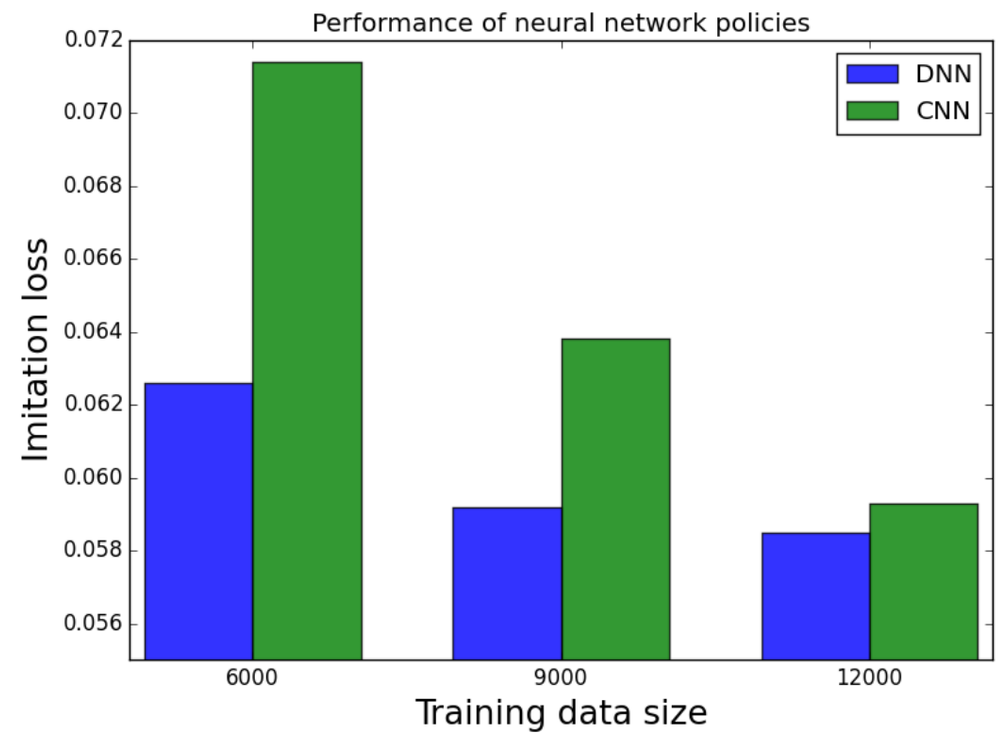

RGB image
160x80x3
Conv
3x3
ReLU
32 filters
Conv
3x3
ReLU
32 filters
Max
pooling
2x2
Conv
3x3
ReLU
64 filters
Conv
3x3
ReLU
64 filters
Max
pooling
2x2
Conv
3x3
ReLU
128 filters
Conv
3x3
ReLU
128 filters
Max
pooling
2x2
Flatten
Fully connected
Wheel speed
ReLU
Fully connected
+
ReLU
Fully connected
ReLU
Linear
Steering
Throttle

Results

Policy	Avg. speed	Top speed	Training data	Completion ratio	Total loss	Steering/Throttle loss
Expert	6.05 m/s	8.14 m/s	N/A	100 %	0	0
Batch	4.97 m/s	5.51 m/s	3000	100 %	0.108	0.092/0.124
Batch	6.02 m/s	8.18 m/s	6000	51 %	0108	0.162/0.055
Batch	5.79 m/s	7.78 m/s	9000	53 %	0.123	0.193/0.071
Batch	5.95 m/s	8.01 m/s	12000	69 %	0.105	0.125/0.083
Online (1 iter)	6.02 m/s	7.88 m/s	6000	100 %	0.090	0.112/0.067
Online (2 iter)	5.89 m/s	8.02 m/s	9000	100 %	0.075	0.095/0.055
Online (3 iter)	6.07 m/s	8.06 m/s	12000	100 %	0.064	0.073/0.055

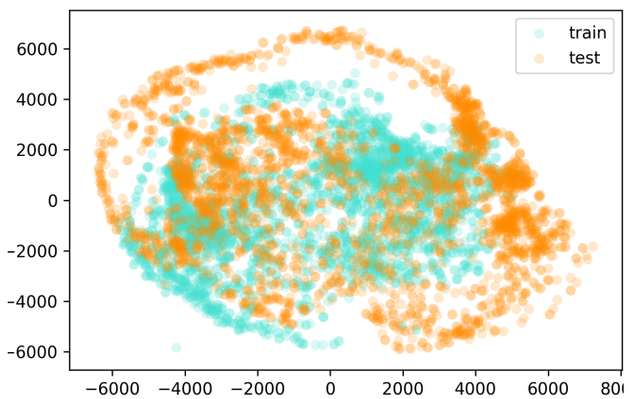


Online IL does better than Batch IL

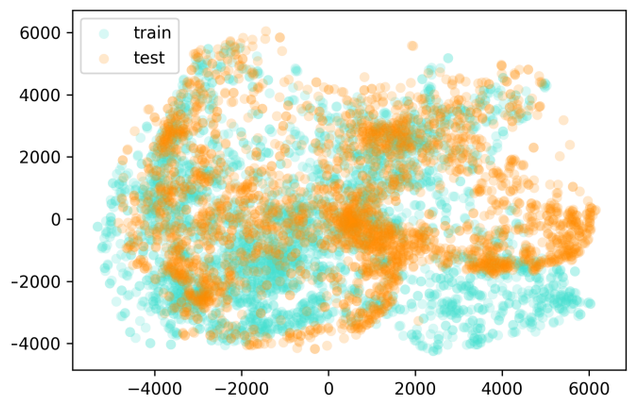


Speed as input to network helps

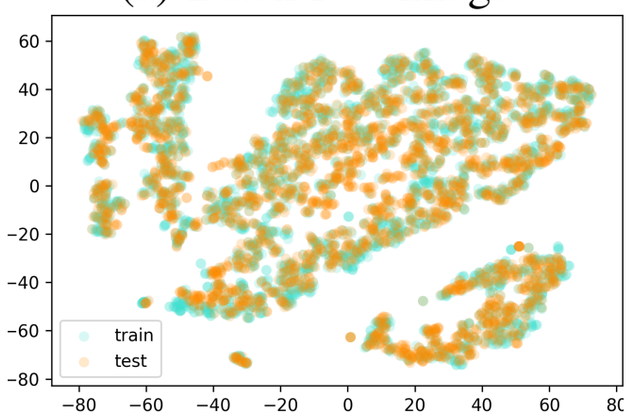
Visualizations



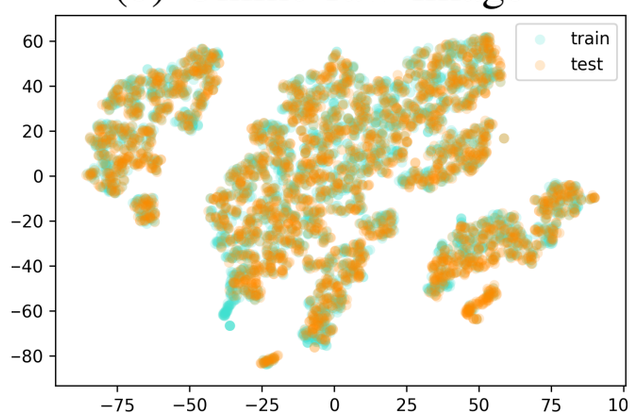
(a) Batch raw image



(b) Online raw image



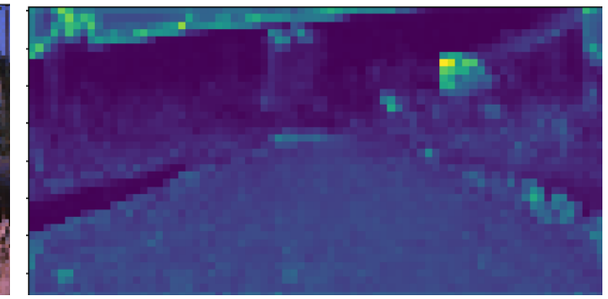
(c) Batch wheel speed



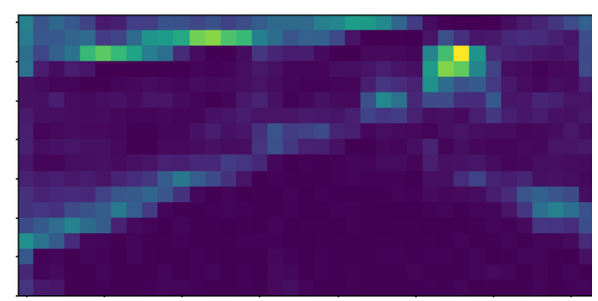
(d) Online wheel speed

Train and test distributions match better for online IL vs batch IL.

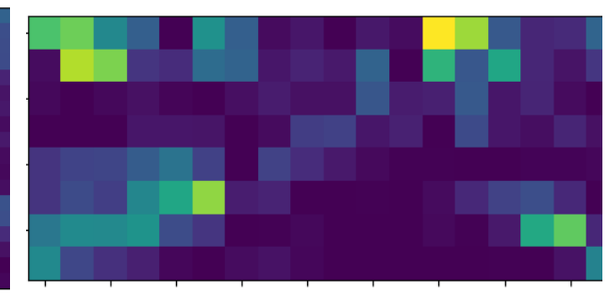
(a) raw image



(b) max-pooling1



(c) max-pooling2



(d) max-pooling3

Network seems to attend to places it should attend to.

Thank you