Cognitive Mapping and Planning for Visual Navigation

Saurabh Gupta[1,2] James Davidson[2] Sergey Levine[1,2] Rahul Sukthankar[2] Jitendra Malik[1,2]
[1]UC Berkeley [2]Google

Presented by Kent Sommer

Korea Advanced Institute of Science and Technology

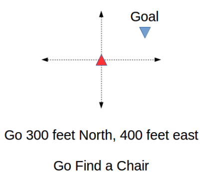
Table of contents

Problem Statement

Problem Statement

Robot equipped with a first person camera

Dropped into a novel environment



Navigate in the environment

Robot Navigation in novel envionments

Motivation: Intelligent Navigation

What does it mean to navigate **intelligently?**

- Navigate through novel environments
- Draw on prior experience or similar conditions
- Reason about free-space, obstacle-space, topology

Motivation: Why Are Humans So Good?

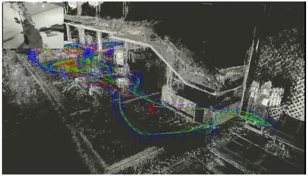
Humans can often reason about their environment while classical agents can at best do uninformed exploration

- Know where we are likely to find a chair
- Know that hallways often lead to other hallways
- Know common building patterns

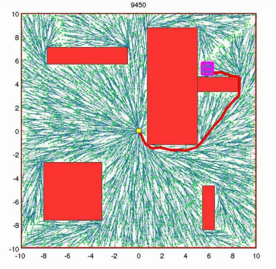
Related Work

Classical Work

- **Over-complete**
  - Precise reconstruction of everything is not necessary
- **Incomplete**
  - Nothing is known till it is explicitly observed, fail to exploit the structure of the world
  - Only geometry, no semantics
- Unnecessarily fragile due to separation between mapping and planning



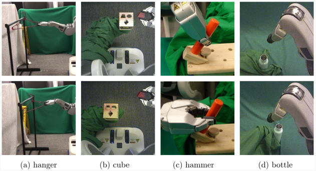
LSD-SLAM



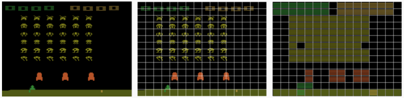
RRT

Contemporary Work

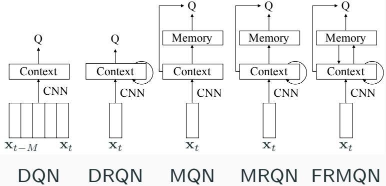
Target-driven Visual Navigation in Indoor Scenes using Deep Reinforcement Learning, Zhu et al., ICRA 2017



End-to-End Training of Deep Visuomotor Polocies, Levine et al., JMLR 2015

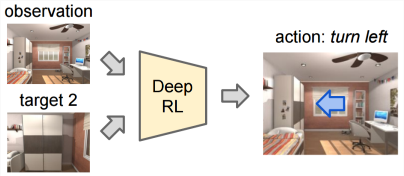


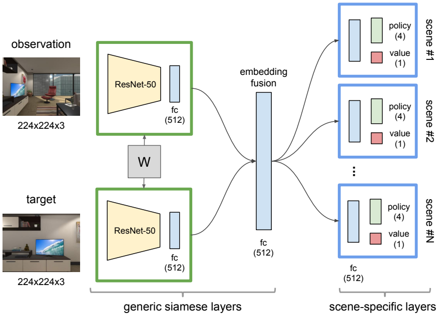
Human-level control through deep reinforcement learning, Mnih et al., Nature 2014



Control of Memory, Active Perception, and Action in Minecraft, Oh et al., IMCL 2016

Contemporary Work





Feed Forward architecture without memory.

- Agent can't systematically explore a new environment or backtrack.
- Agent needs experience with a new environment before it can start navigating successfully.

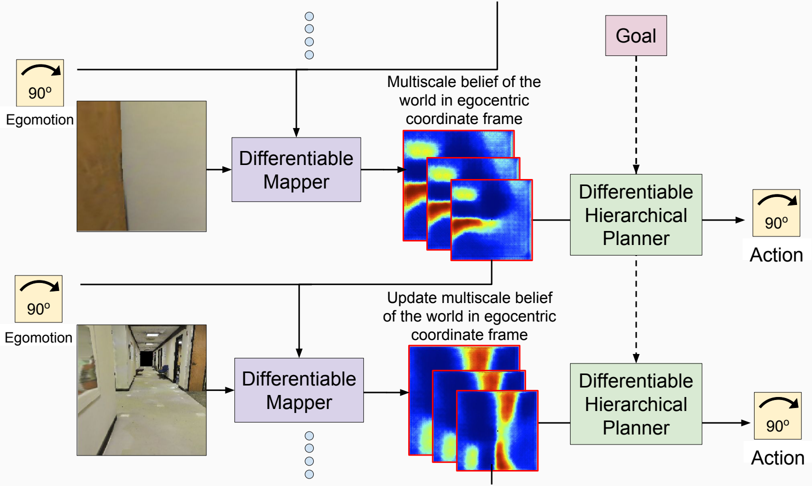
Contribution

Contribution

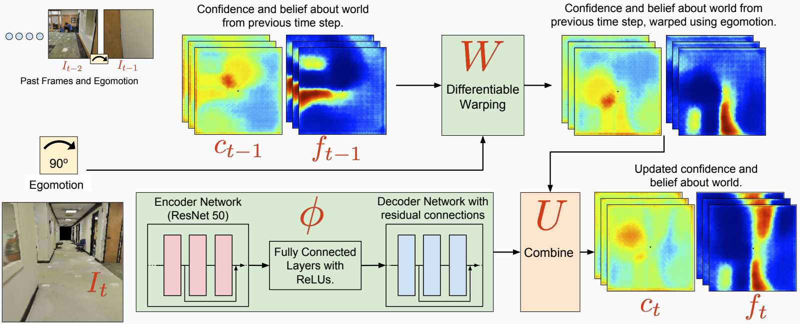
Neural network policy for visual navigation

- Joint architecture for mapping and planning
- Spatial memory with the ability to plan given partial observations
- Is end-to-end trainable

Cognitive Mapping and Planning: System Overview



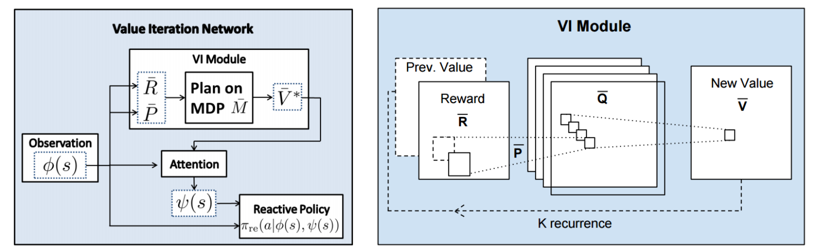
Differentiable Mapper



Differentiable Planner

Value Iteration Network[1]

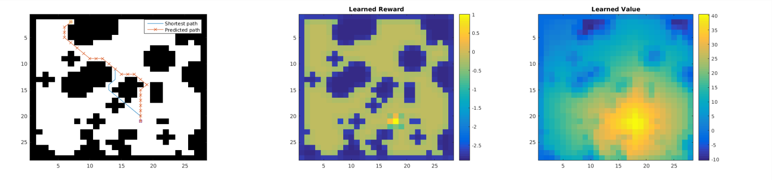
- $Q_n(s,a) = R(s,a) + \gamma \sum_{s'} P(s'|s,a) V_n(s')$
  - Computed as convolutions
- $V_{n+1}(s) = \max_a Q_n(s,a) \quad \forall s$
  - Computed as max pooling over channels



[1] Aviv Tamar et al. "Value iteration networks". In: *Advances in Neural Information Processing Systems*. 2016, pp. 2146–2154.

Differentiable Planner: Value Iteration Network

- $Q_n(s,a) = R(s,a) + \gamma \sum_{s'} P(s'|s,a) V_n(s')$
  - Computed as convolutions
- $V_{n+1}(s) = \max_a Q_n(s,a) \quad \forall s$
  - Computed as max pooling over channels



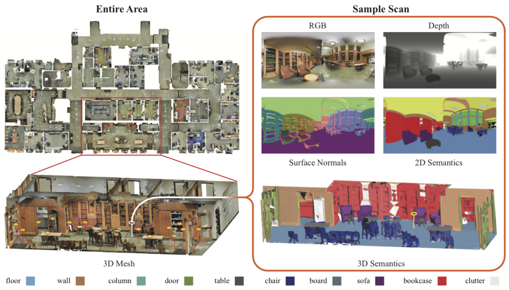
Trainable using simulated data

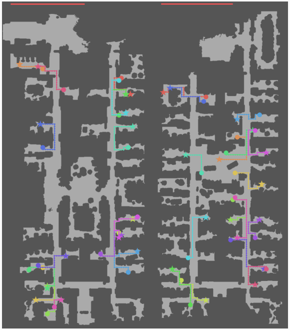
Experimental Setup: Overview

- Trained and tested in static simulated real-world environments
- **Testing environment is different from training environments**
- Robot:
  - Lives in a grid world, and motion is discrete
  - Has 4 macro-actions:
    - Go Forward, Turn left, Turn right, Stay in place
  - Has access to precise egomotion
  - Has RGB and/or Depth Cameras
- All models are trained using DAGGER
- Geometric Task:
  - Goal is sampled to be at most 32 time steps away. Agent is run for 39 time steps.
- Semantic Task:
  - 'Go to a Chair,' agent run for 39 time steps.

Experimental Setup: Dataset

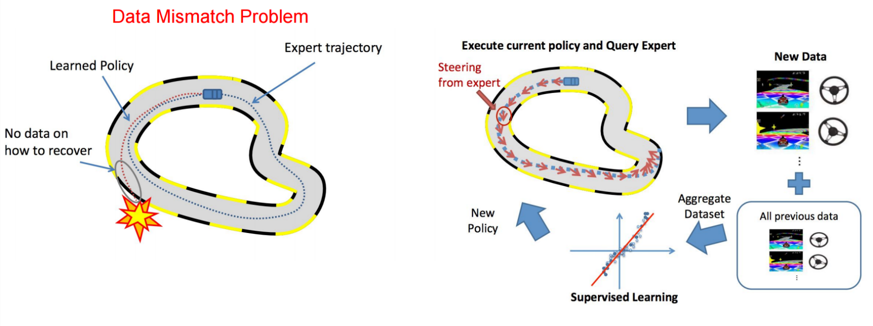
Stanford Building Parser Dataset





Experimental Setup: Policy Training

Use DAGGER[2]



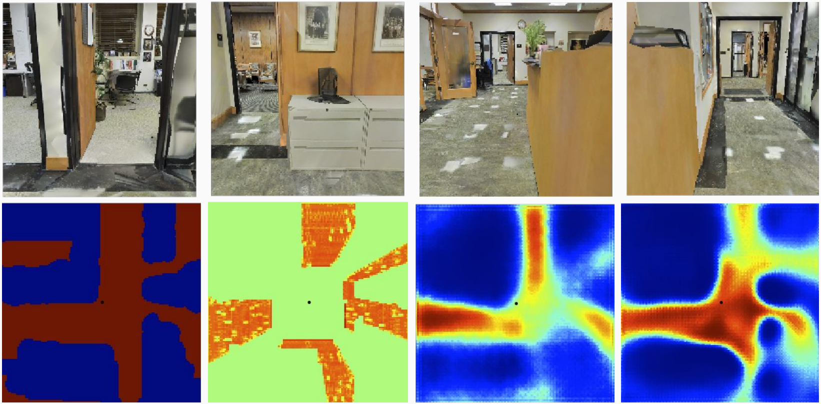
3

[2] Stéphane Ross, Geoffrey J Gordon, and Drew Bagnell. "A Reduction of Imitation Learning and Structured Prediction to No-Regret Online Learning." In: *AISTATS*. vol. 1. 2. 2011, p. 6.

[3] Image from: John Schulmanś Lecture on Reinforcement Learning

Results

Mapper Unit Test



Ground Truth Analytical Project RGB Pred D Pred

Navigation Results: Geometric Task

Method	Mean		75th %ile		Success %age	
	RGB	Depth	RGB	Depth	RGB	Depth
Geometric Task						
Initial	25.3	25.3	30	30	0.7	0.7
No Image LSTM	20.8	20.8	28	28	6.2	6.2
Reactive (1 frame)	20.9	17.0	28	26	8.2	21.9
Reactive (4 frames)	14.4	8.8	25	18	31.4	56.9
LSTM	10.3	5.9	21	5	53.0	71.8
Our (CMP)	**7.7**	**4.8**	**14**	**1**	**62.5**	**78.3**

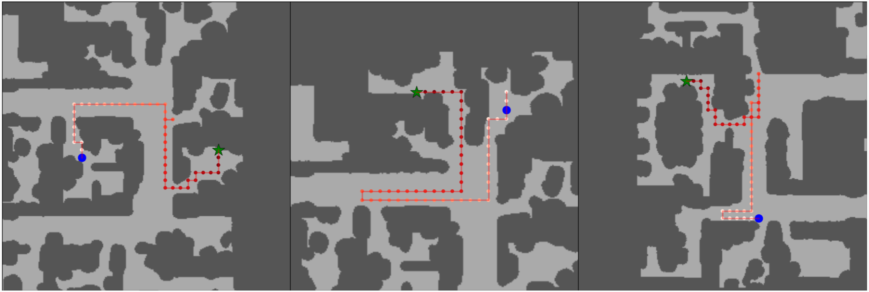
Geometric Results: Mean distance to goal location, 75th percentile distance to goal and success rate after executing the policy for 39 time steps.

Navigation Results: Semantic Task

Method	Mean		75^{th} %ile		Success %age	
	RGB	Depth	RGB	Depth	RGB	Depth
Semantic Task (Aggregate)						
Initial	16.2	16.2	25	25	11.3	11.3
Reactive	14.2	14.2	22	23	23.4	22.3
LSTM	13.5	13.4	20	23	23.5	27.2
Our (CMP)	**11.3**	**11.0**	**18**	**19**	**34.2**	**40.0**

Semantic Results: Mean distance to goal location, 75^{th} percentile distance to goal and success rate after executing the policy for 39 time steps.

Successful Navigations



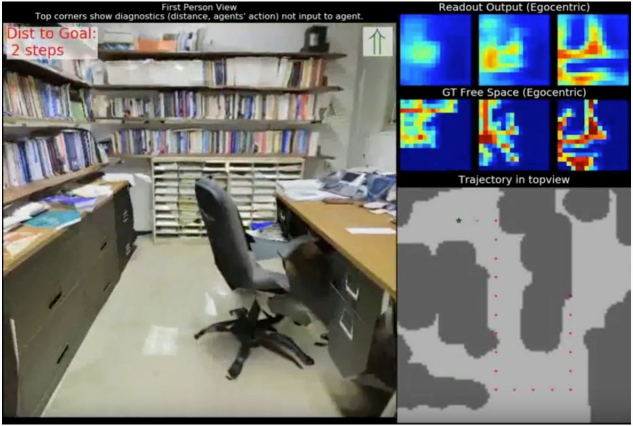
Agents exhibit backtracking behavior!

Failure Cases

Missed Thrashing Tight

Video Demo

Demo



Video Demonstration

Summary

Summary

- Joint fully end-to-end neural network policy for mapping and planning
- Uses mapping module to map from RGB and/or Depth images to a top-down ego-centric belief map
- Uses a Value Iteration Network to plan in the belief map generated by the mapper
- Trains the end-to-end policy using DAGGER

Questions?

Quiz

- Why was DAGGER used to train the models?
  1. Other training methods were not possible
  2. To allow the agent to recover from bad decisions (backtracking)
  3. To minimize crashes in simulation
  4. Because it has a cool name
- The model was trained end-to-end allowing for the mapping module to encode whatever was most useful to the planning module
  1. True
  2. False