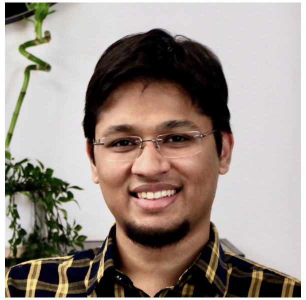
Visual Navigation

Russ Salakhutdinov

Machine Learning Department
Carnegie Mellon University

Visual Navigation



Devendra Chaplot

- Embodied AI: Deep Reinforcement Learning and Control

Learning To Explore Using Active Neural Mapping, Devendra Chaplot, S. Gupta, D. Gandhi, A. Gupta, R. Salakhutdinov ICLR 2020

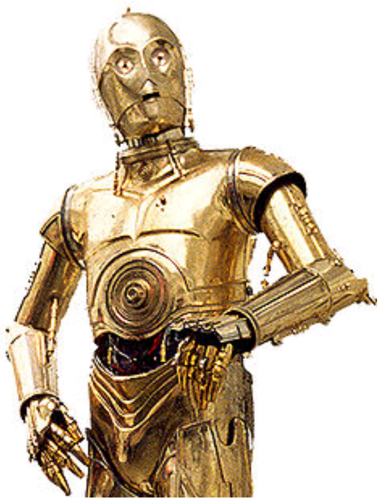
Neural Topological SLAM for Visual Navigation, Devendra Chaplot, R. Salakhutdinov , A. Gupta, S. Gupta, R. Salakhutdinov CVPR 2020

Gated Path Planning Networks, Lisa Lee, E. Parisotto, D. Chaplot, E. Xing, R. Salakhutdinov, ICML 2018

Many of these slides were prepared by Devendra Chaplot

Building AI

Develop computer algorithms that can:

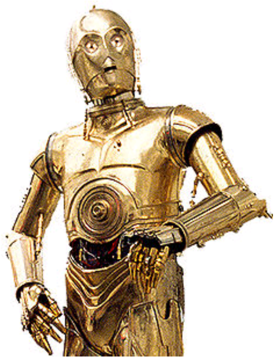


- See and recognize objects around us
- Perceive human speech
- Reason and Understand Natural Language
- Navigate autonomously, Explore, Plan
- Display human like Intelligence

Learning Behaviors

Observation

o_t

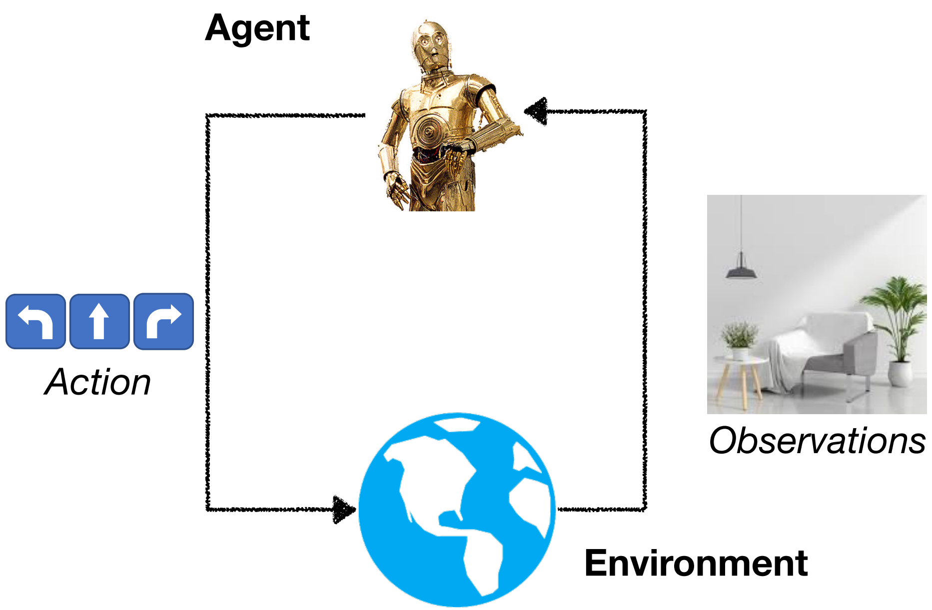


Action

a_t

Learning to map sequences of observations to actions, for a particular goal

Physical Intelligence

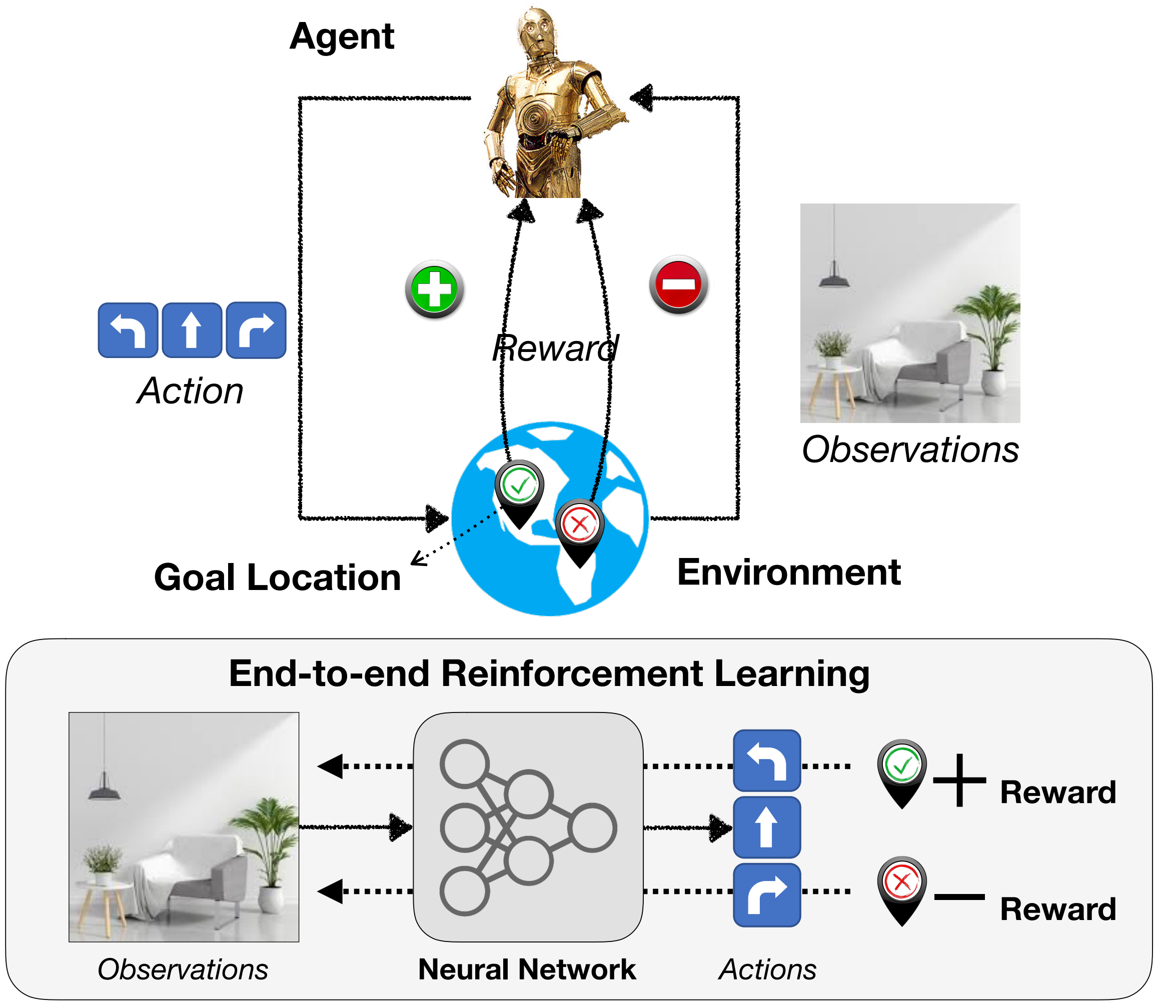


Agent needs to move in the world physically.

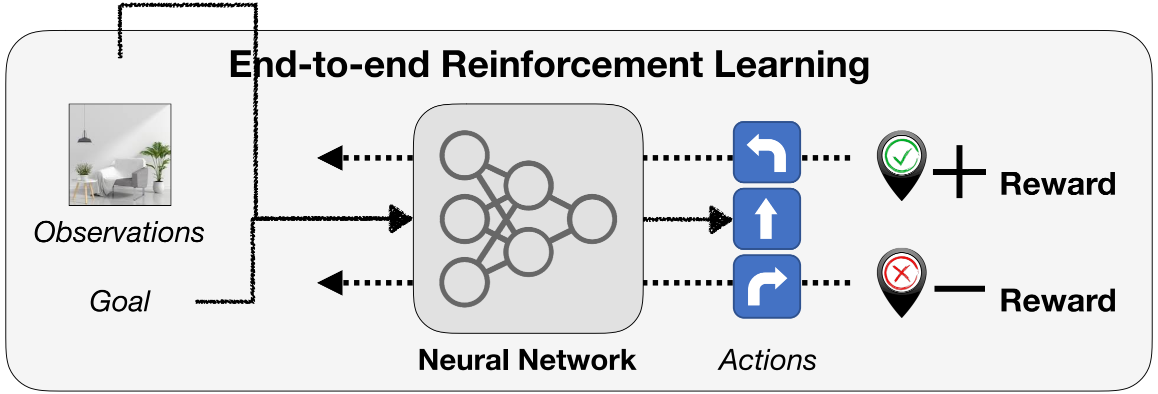
Current actions affect future observations.

Require Spatial and Semantic Understanding.

Navigation


Agent
Reward
Action
Observations
Goal Location
Environment
End-to-end Reinforcement Learning
Observations
Neural Network
Actions
+ Reward
— Reward

Goal-conditioned Navigation



Point Goal

(x, y) *coordinate*

Image Goal

Object Goal

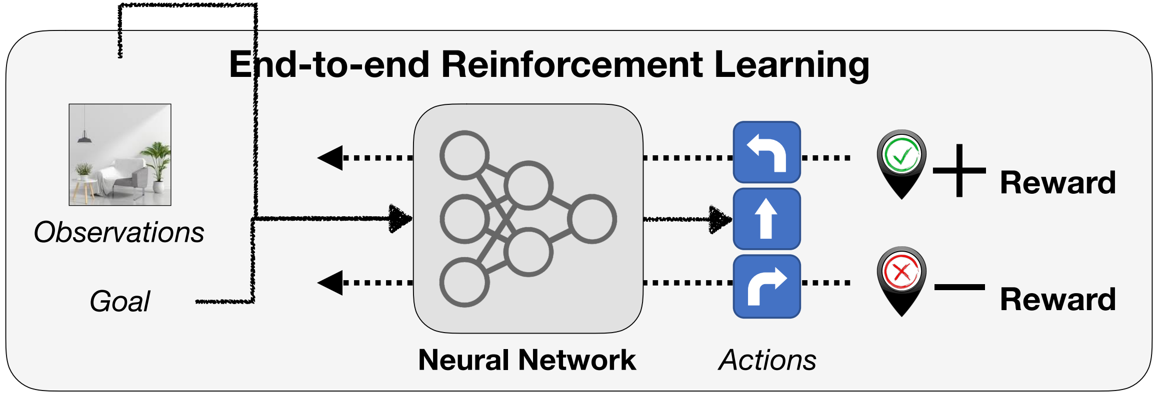
Chair
TV
Sofa

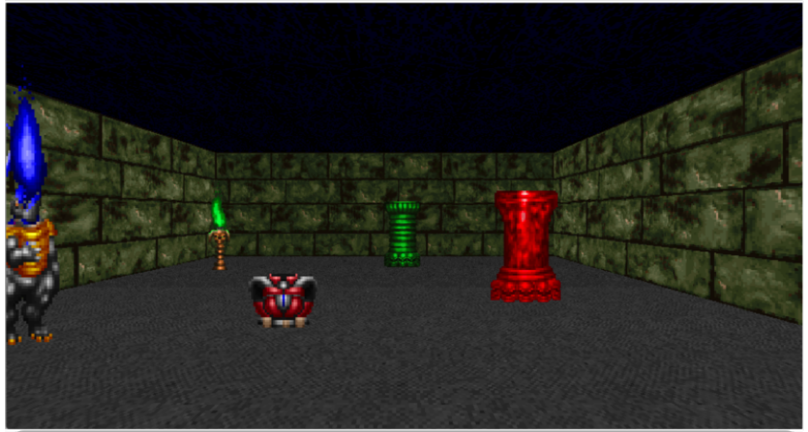
Language Goal

Blue Chair
Largest TV
White Sofa

- Convenient for humans
- Compositionality

Goal-conditioned Navigation





Go to the green torch

Train

Go to the short red torch
Go to the blue keycard
Go to the largest yellow object
Go to the green object

Test

Go to the tall green torch
Go to the red keycard
Go to the smallest blue object

Language Goal

Blue Chair
Largest TV
White Sofa

- Convenient for humans
- Compositionality

Navigation Tasks

Known goal location

- Require efficient navigation to the goal
- Tasks
  - Pointgoal [1, 2, 3]
  - Language Instructions describing path to goal [4]

[1] Anderson et al. *arXiv:1807.06757*, 2018.
[2] Mirowski et al. In *NeurIPS*, 2018.
[3] Savva et al. arXiv:1712.03931, 2017.
[4] Anderson et al. In *CVPR*, 2018.

Navigation Tasks

Known goal location

- Require efficient navigation to the goal
- Tasks
  - Pointgoal [1, 2, 3]
  - Language Instructions describing path to goal [4]

Unknown goal location

- Require exhaustive exploration
- Tasks
  - Exploration: Maximize explored area [5]
  - Object/Area Goal [3, 6, 7]
  - Semantic Goal Navigation [8]
  - Embodied Question Answering [9, 10]

[1] Anderson et al. *arXiv:1807.06757*, 2018.
[2] Mirowski et al. In *NeurIPS*, 2018.
[3] Savva et al. arXiv:1712.03931, 2017.
[4] Anderson et al. In *CVPR*, 2018.
[5] Chen et al. *ICLR*, 2019.
[6] Lample et al. In *AAAI*, 2017.
[7] Mirowski et al. *ICLR*, 2017.
[8] Chaplot et al. *AAAI*, 2018.
[9] Gordon et al. *CVPR*, 2018.
[10] Das et al. *CVPR*, 2018.

Desirable Characteristics of a Navigation model

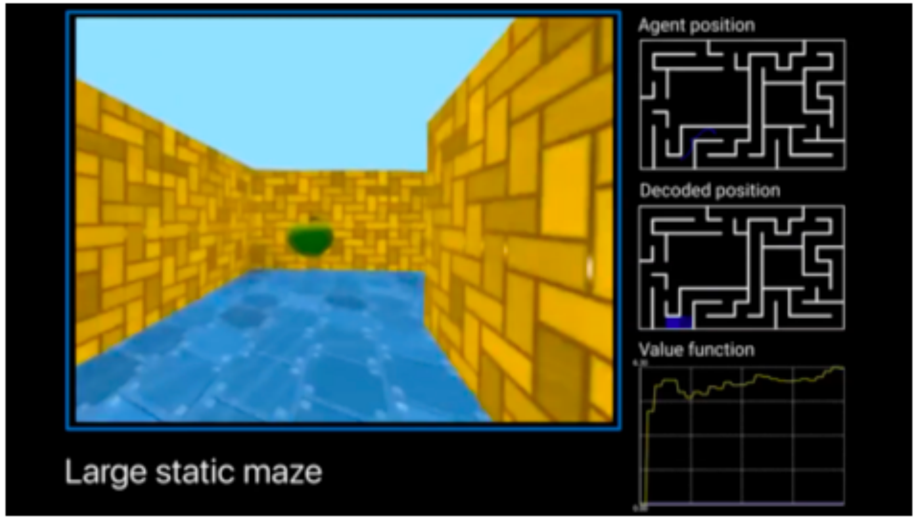
- Effective at both types of Navigation tasks:
  - Known goal location (Pointgoal) and
  - Unknown goal location (Exploration)
- Generalization: domains, task, goals
- Sample efficiency

Limitations of Classical SLAM

- Generalization
  - Robustness to environment conditions [Maddern et al. 2016]
  - Robustness to dynamic objects [Zou and Tan, 2012]
  - Failure cases of keypoint tracking [Cadena et al. 2016]
- Passiveness
  - Unable to decide the actions taken by the agent in order to map the environment or localize as accurately and efficiently as possible.

Deep RL?

[Lample & Chaplot, 2016]



[Mirowski et al. 2017]

Limitations of "end-to-end" Deep RL

- Ineffective at long-term planning
- Sample inefficiency
- Poor transferability

Navigation Tasks

Point Goal

(x, y) *coordinate*

Image Goal

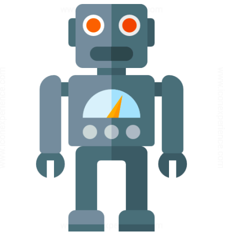
Object Goal

Chair
TV
Sofa

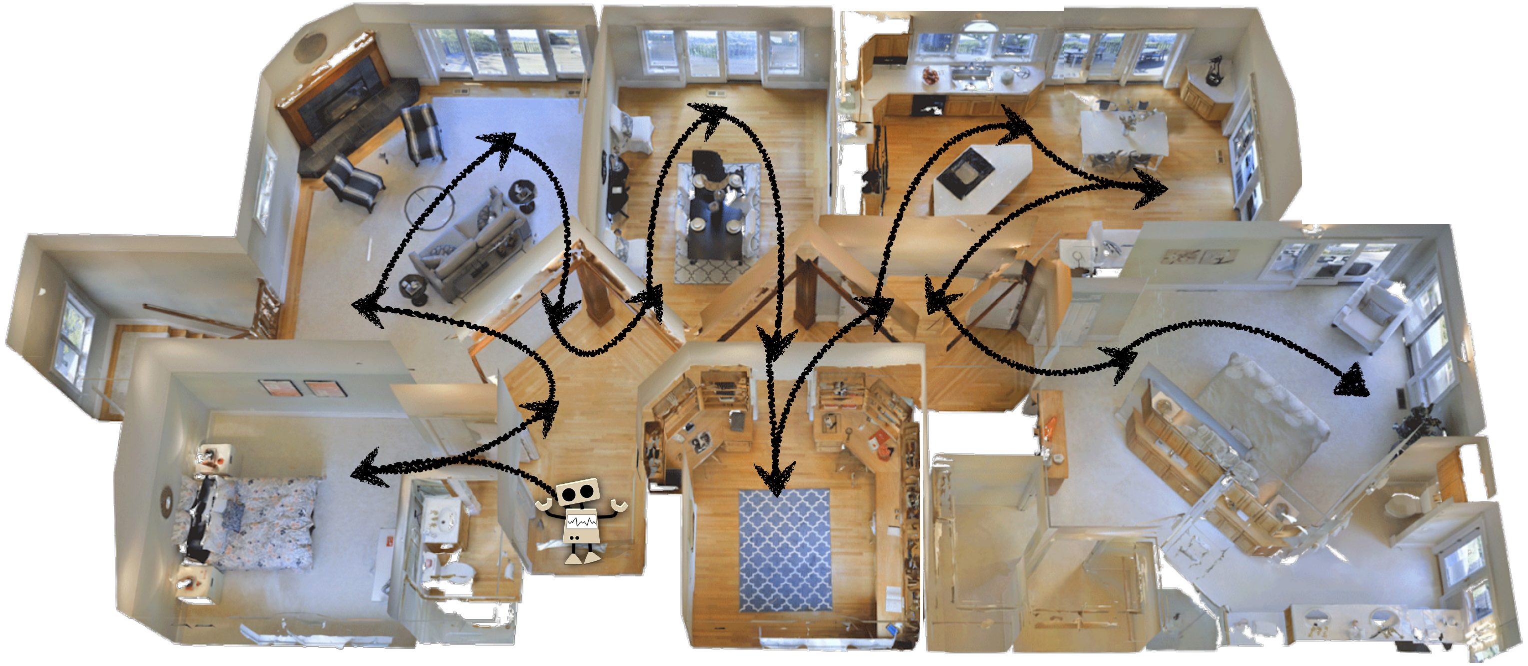
Language Goal

Blue Chair
Largest TV
White Sofa

Require exploring the environment to find the goal

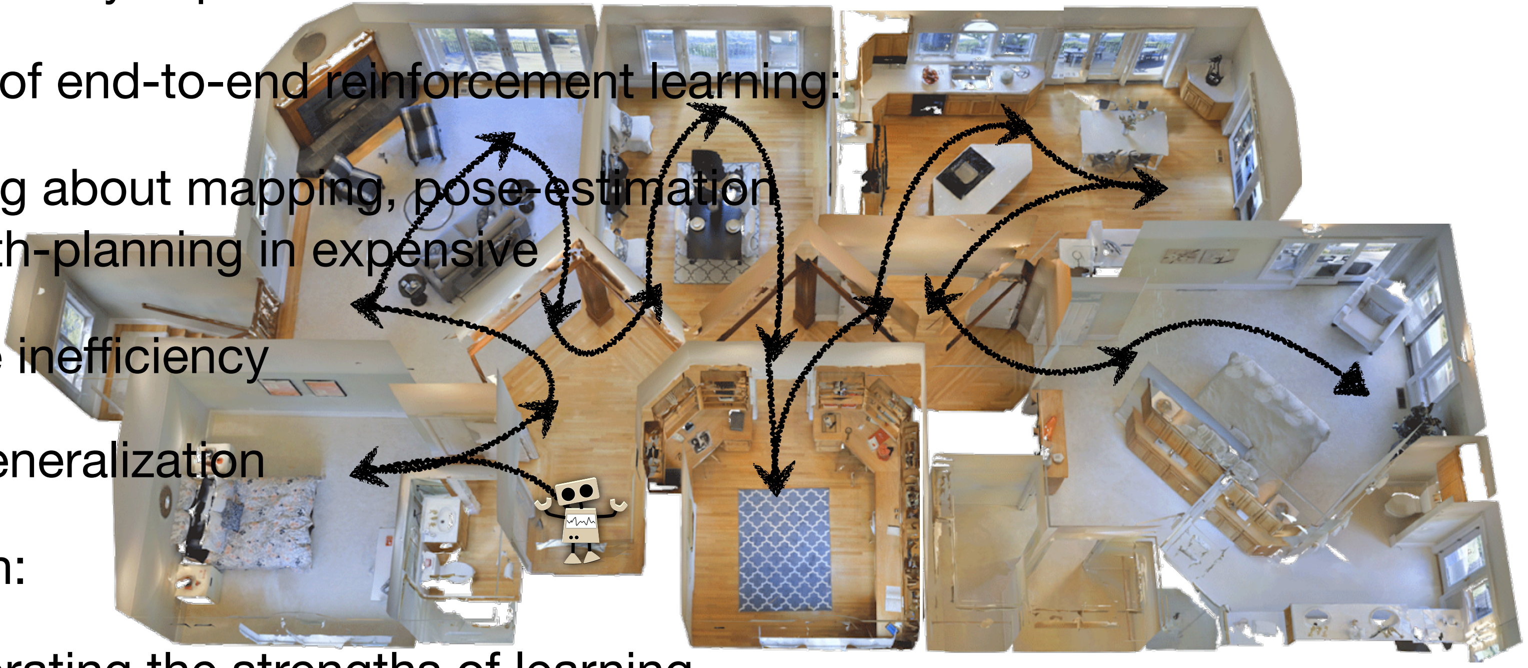


Exploration

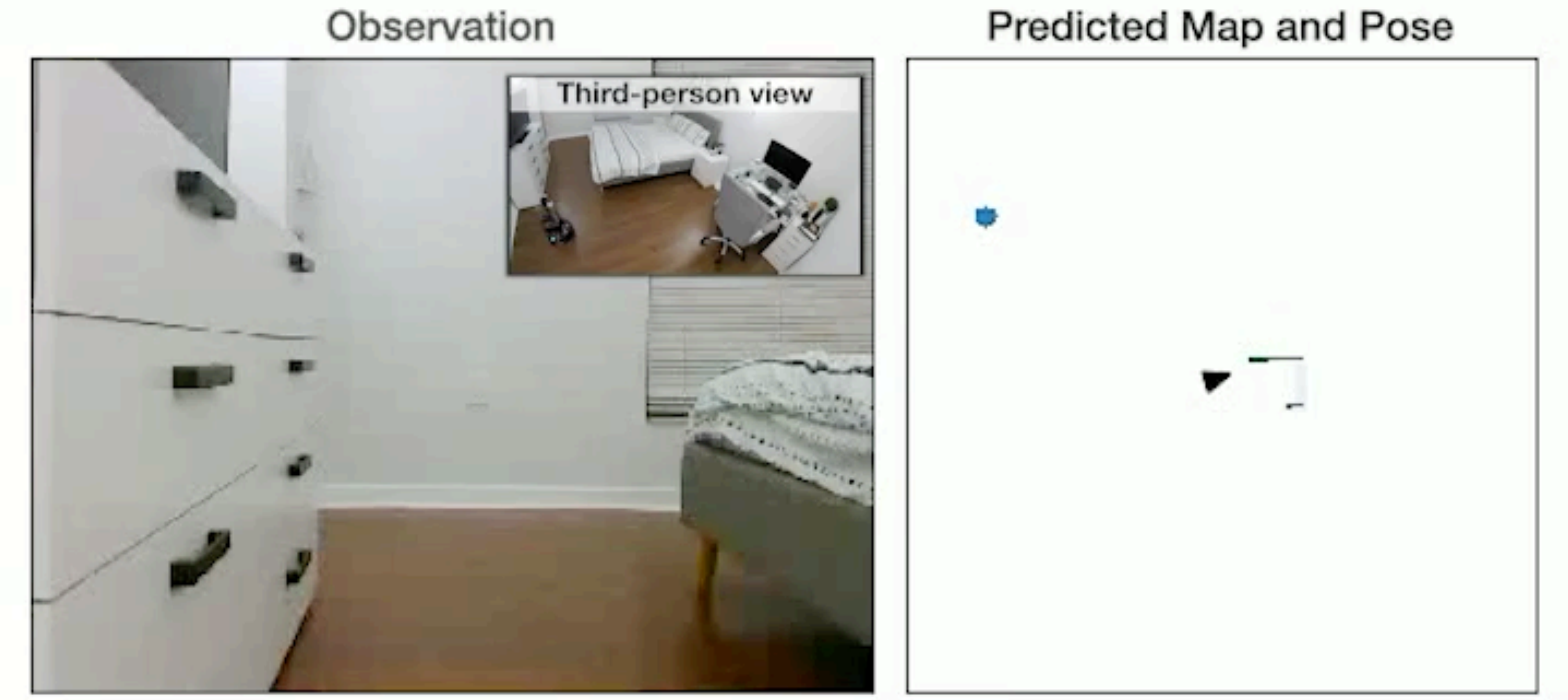


Exploration

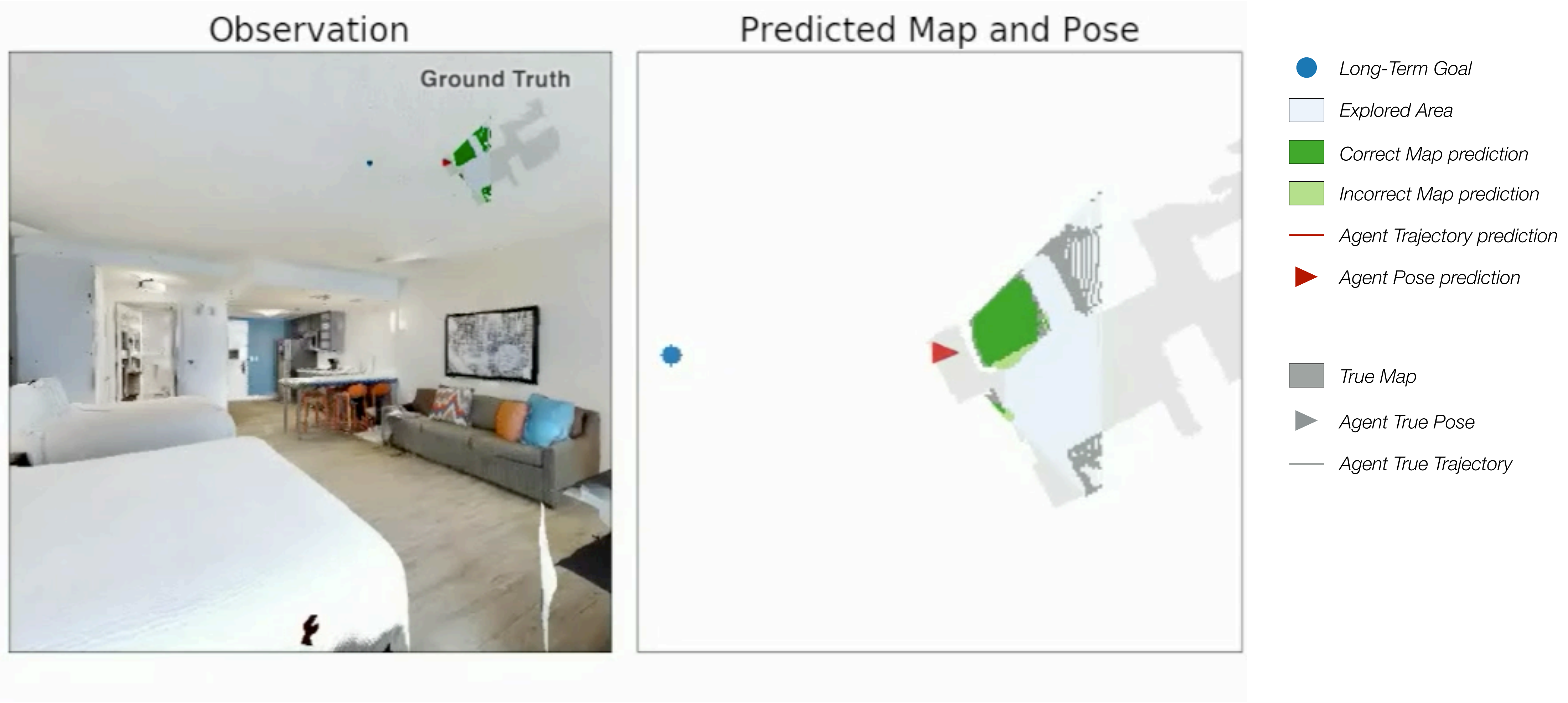
- How to efficiently explore an unseen environment?
- Limitations of end-to-end reinforcement learning:
  - Learning about mapping, pose-estimation and path-planning in expensive
  - Sample inefficiency
  - Poor generalization
- Our solution:
  - Incorporating the strengths of learning
  - Modular and hierarchical system



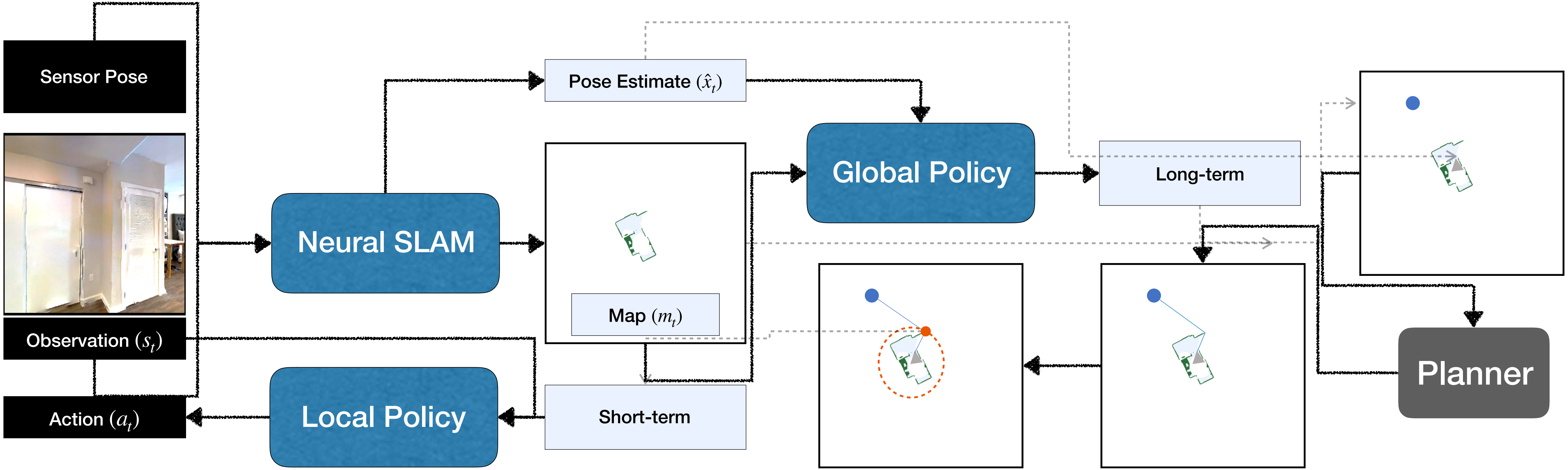
Preview: Visual Navigation in the Real World



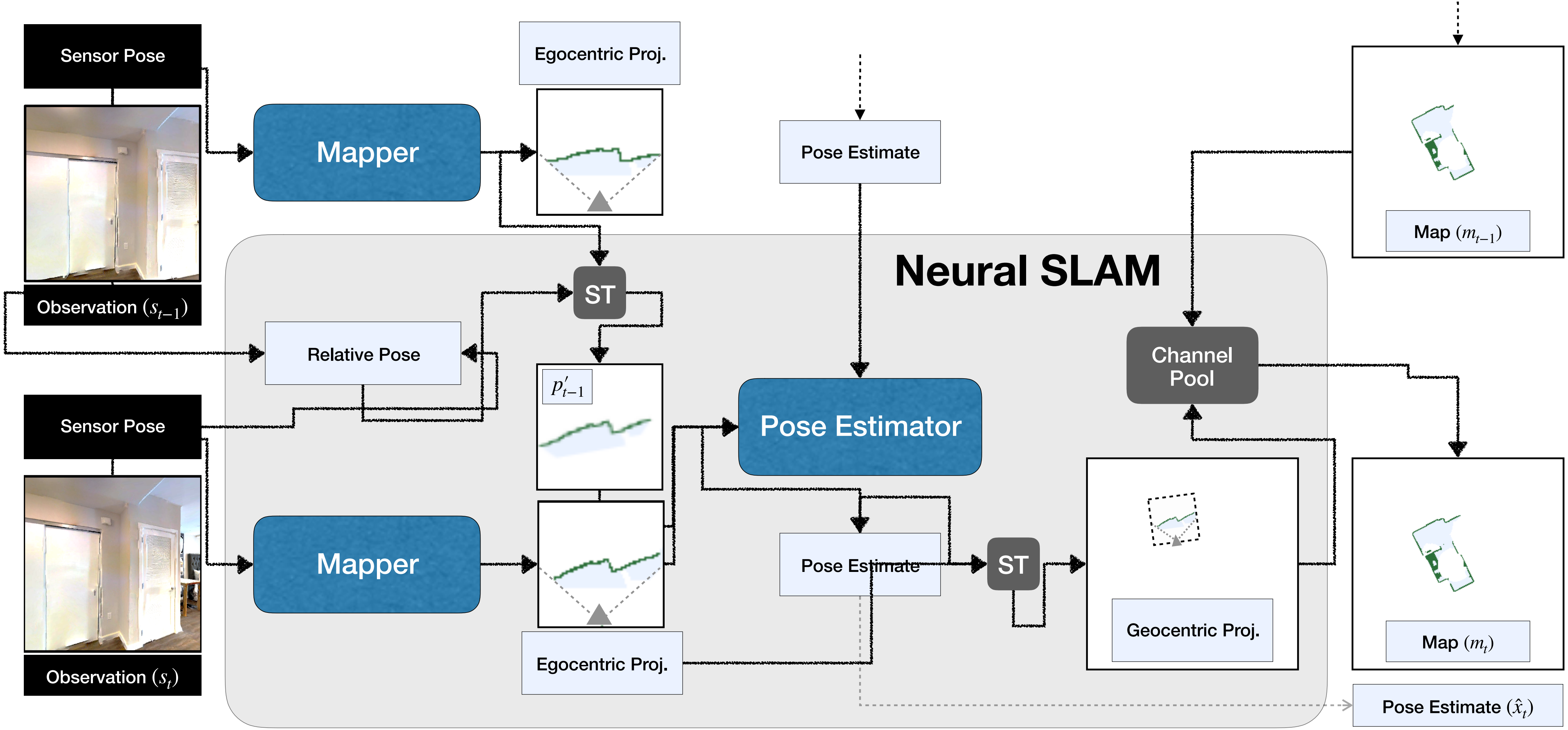
Exploration in Gibson Environment



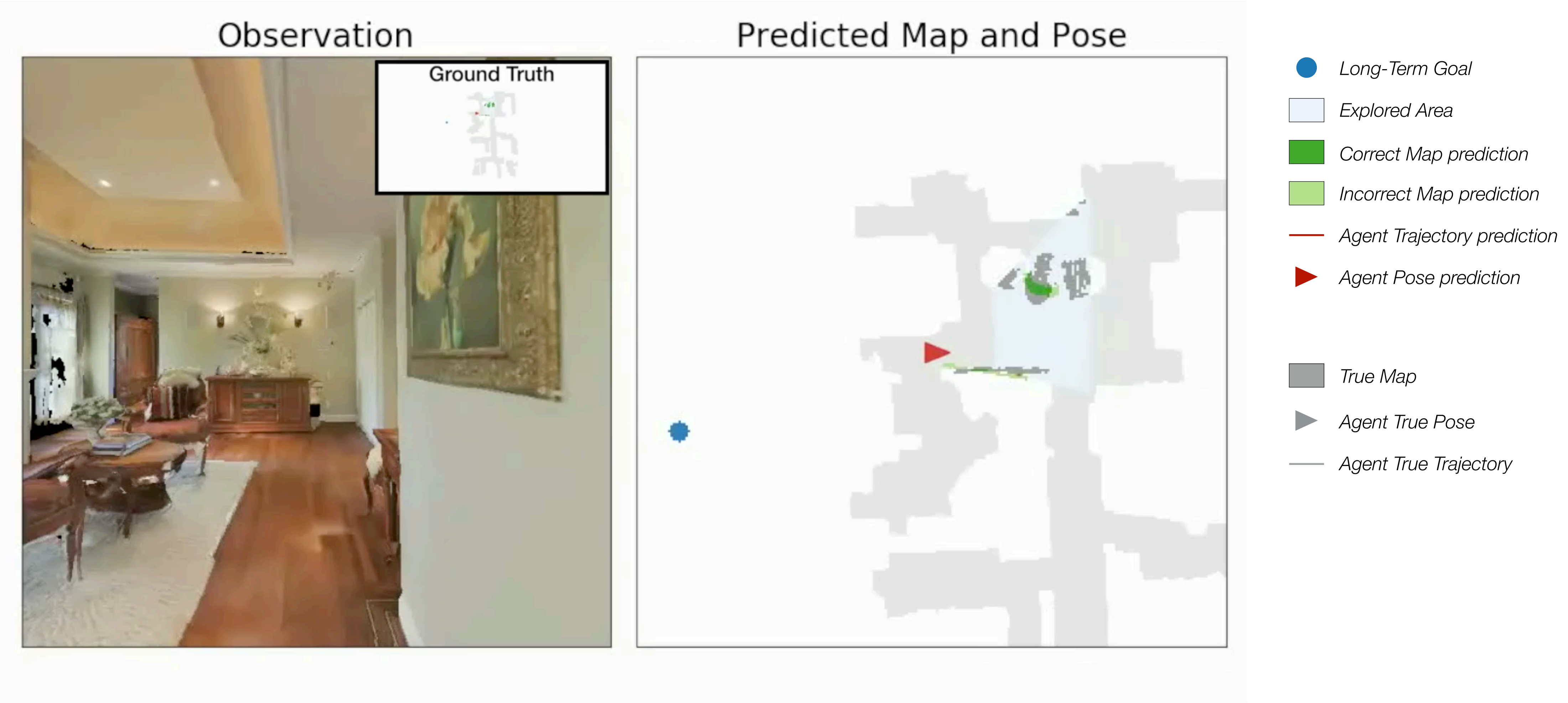
Active Neural SLAM: Overview



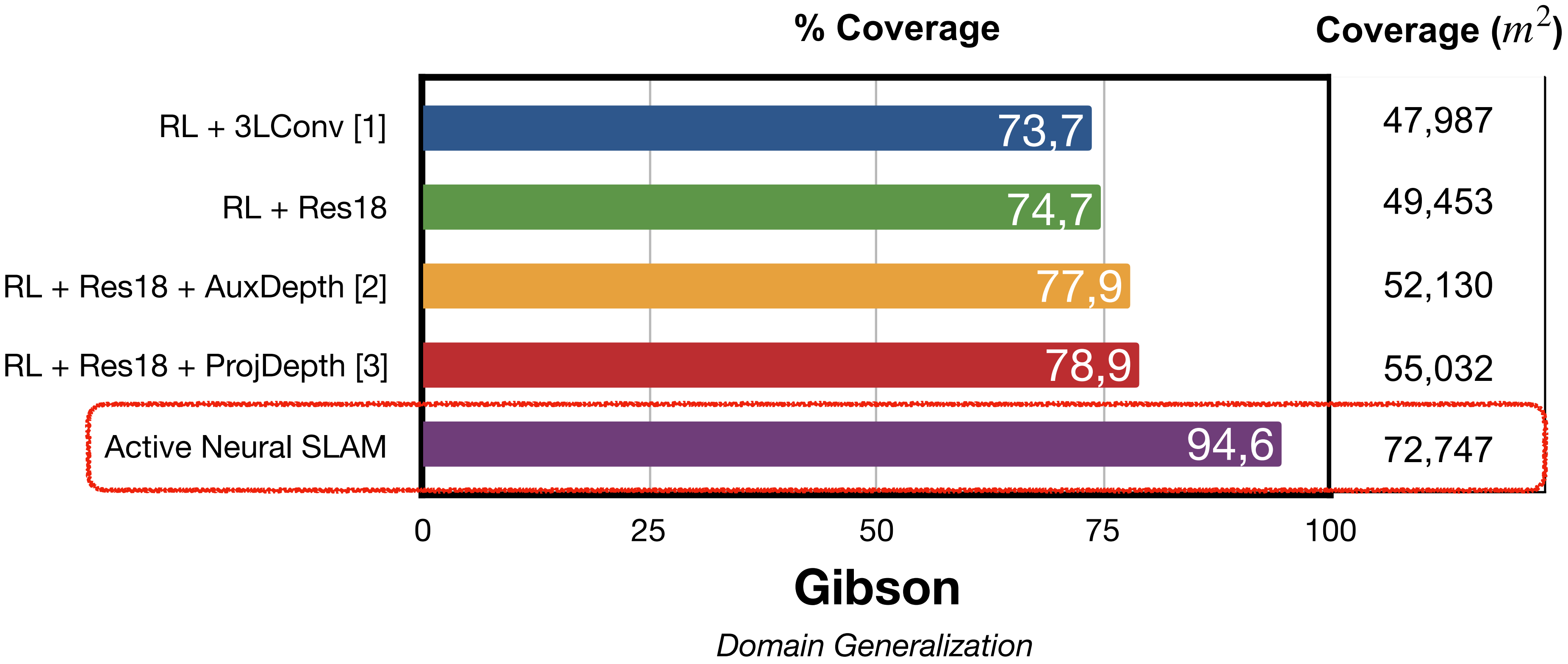
Neural SLAM Module



Domain Generalization: Matterport3D



Exploration Results



*Adapted from [1] Lample & Chaplot. AAAI-17, [2] Mirowski et al. ICLR-17, [3] Chen el al. ICLR-19

Goal-conditioned Navigation

Point Goal

(x, y) *coordinate*

Image Goal

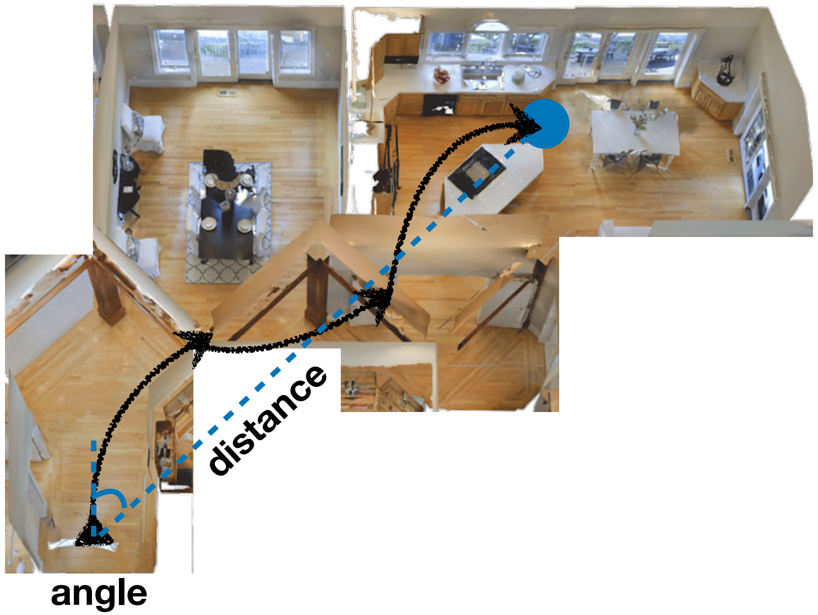
Object Goal

Chair
TV
Sofa

Language Goal

Blue Chair
Largest TV
White Sofa

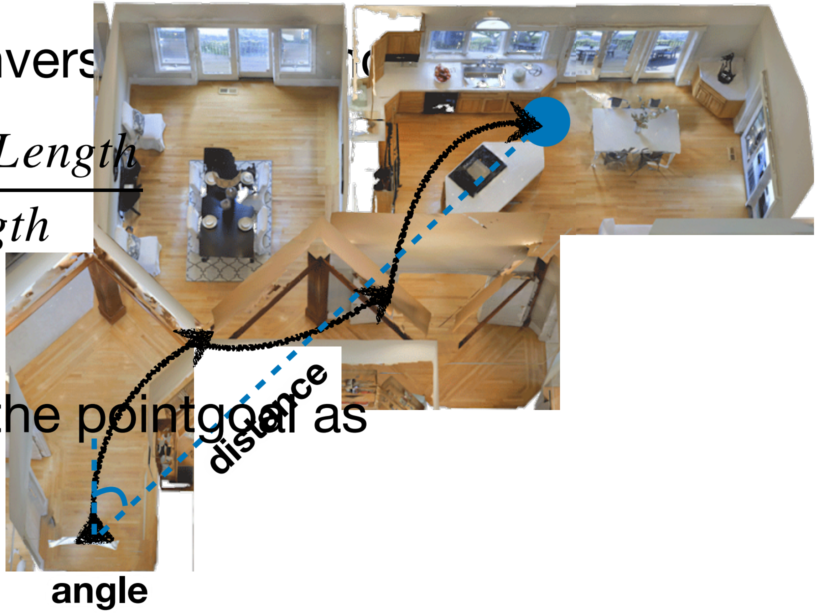
Point-Goal Navigation



Point-Goal Navigation

- Objective: Navigate to goal coordinates
- Metric: Success weighted by invers

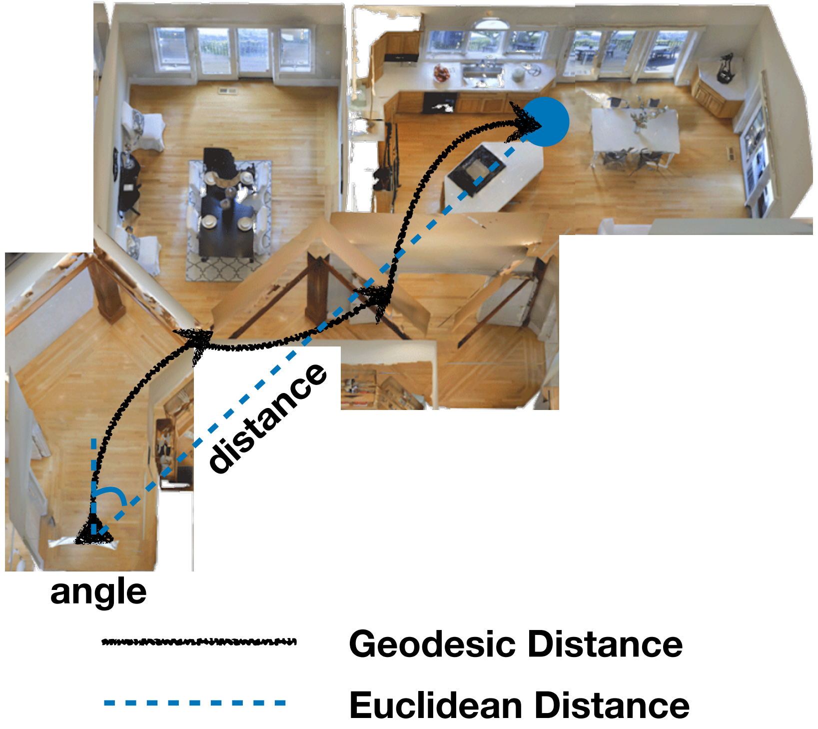
$$\frac{1}{N}\sum_{i=1}^{N} Success * \frac{ShortestPathLength}{PathLength}$$



- Global Policy -> always gives the pointgoal as the long-term goal

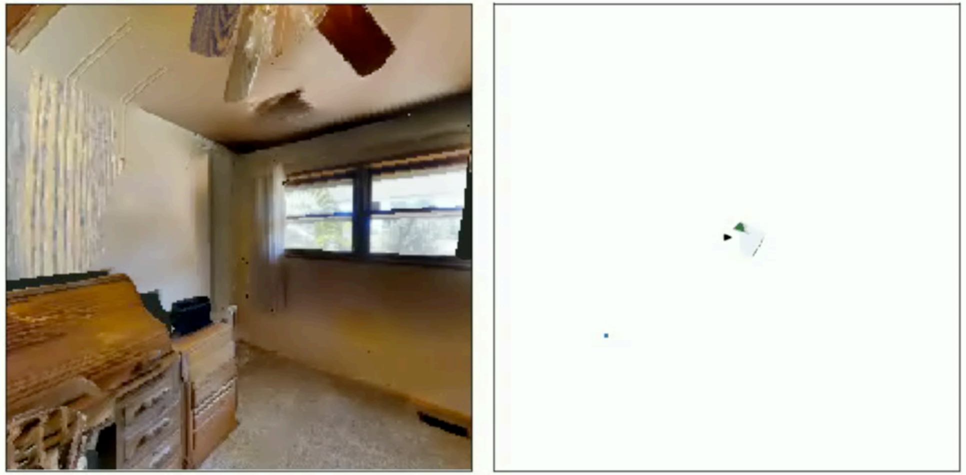
Harder Datasets

- **Hard-GEDR**
  - Higher Geodesic to Euclidean distance ratio (GEDR)
  - Avg GEDR 2.5 vs 1.37, minimum GEDR is 2
- **Hard-Dist**
  - Higher Geodesic distance
  - Avg Dist 13.5m vs 7.0m, minimum Dist is 10m



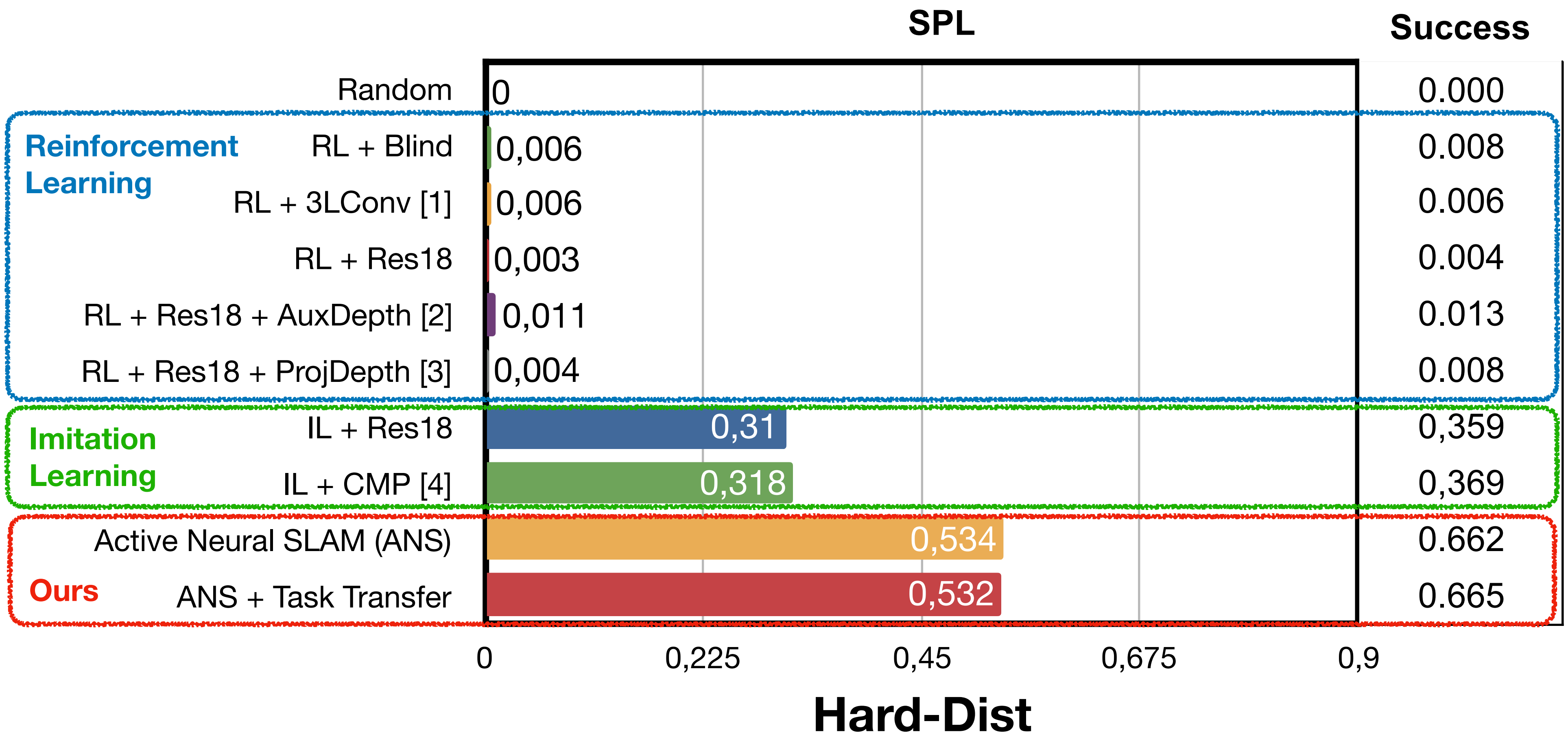
Point-Goal Navigation

Gibson



MP3D

Results



**Adapted from [1] Lample & Chaplot. AAAI-17, [2] Mirowski et al. ICLR-17, [3] Chen el al. ICLR-19, [4] Gupta et al. CVPR-17*

Navigation Tasks

Point Goal

(x, y) *coordinate*

Image Goal

Object Goal

Chair
TV
Sofa

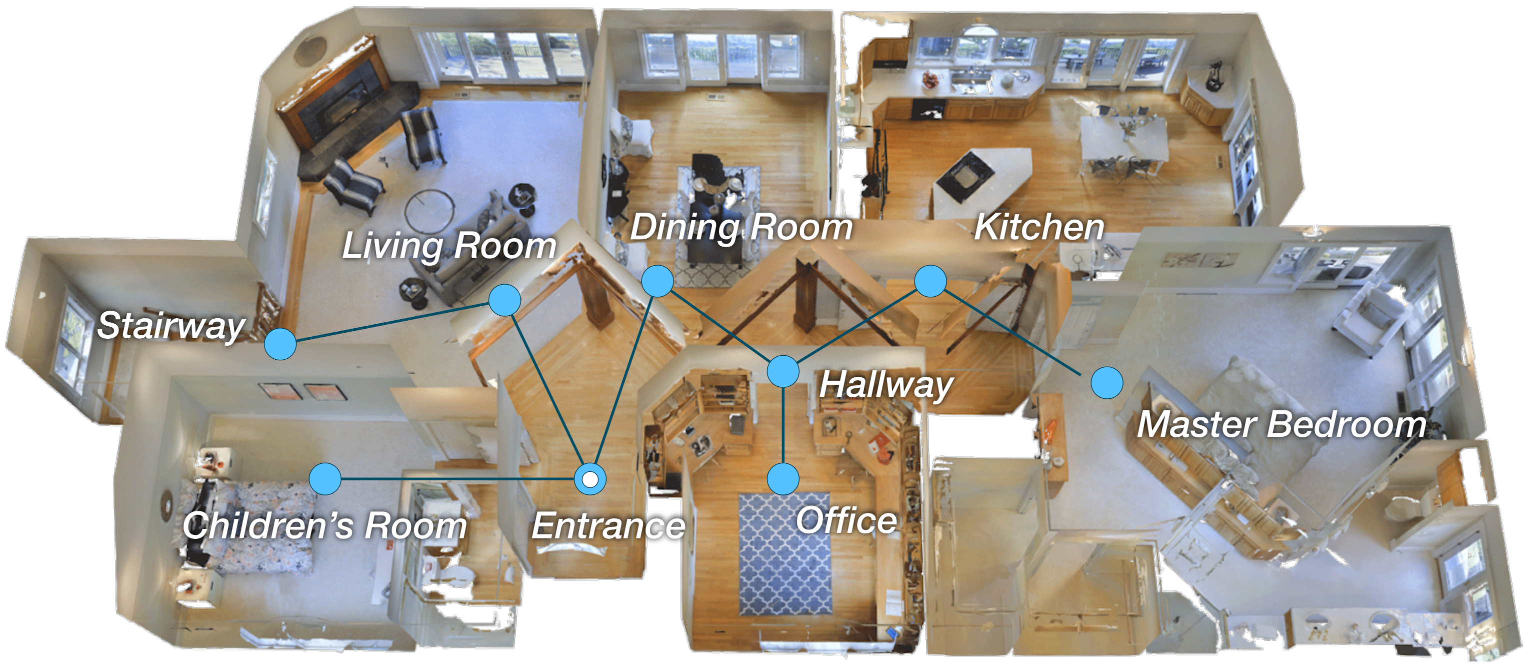
Language Goal

Blue Chair
Largest TV
White Sofa

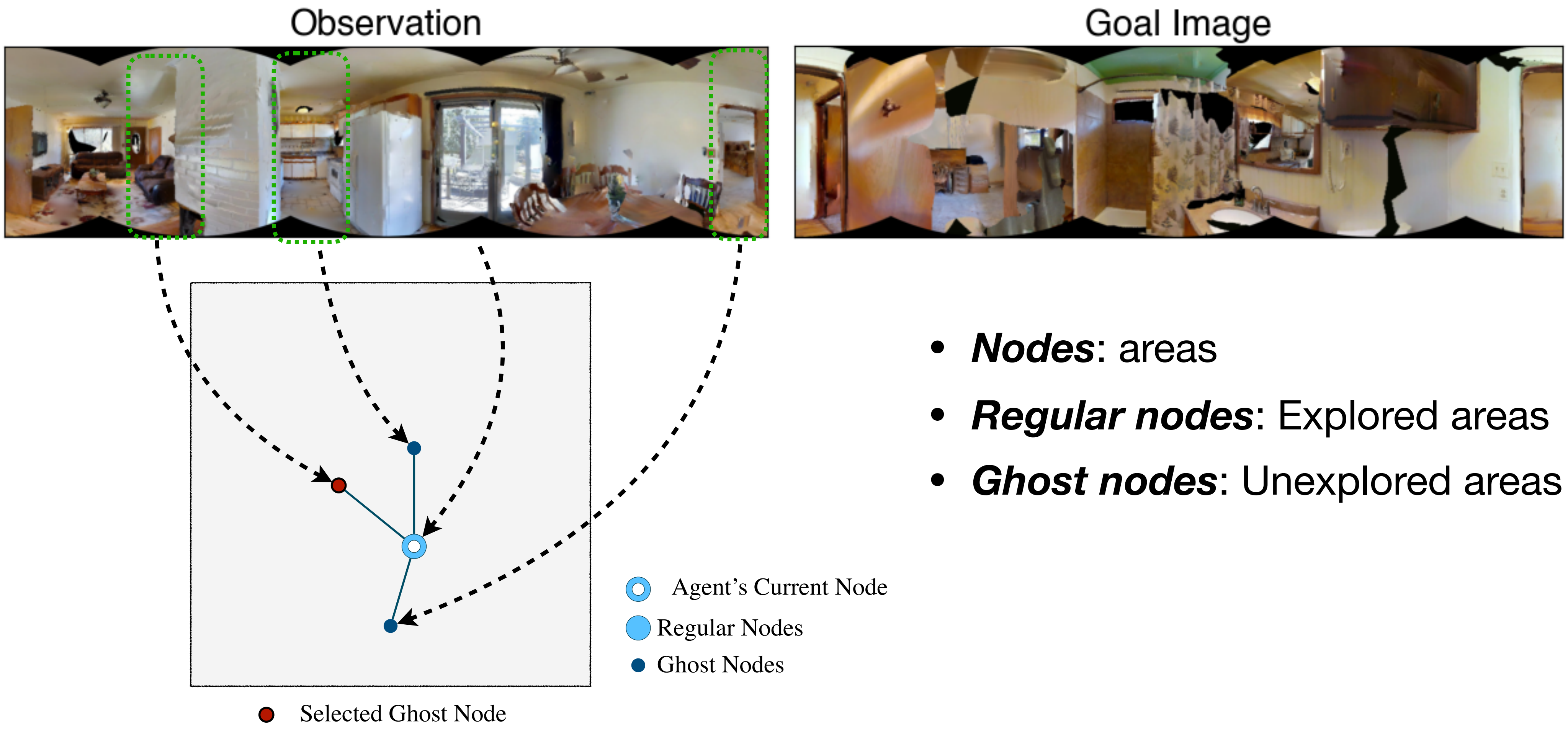
Semantic Priors and Common-Sense

- Humans use semantic priors and common-sense to explore and navigate everyday
- Most navigation algorithms struggle to do so

Topological Maps

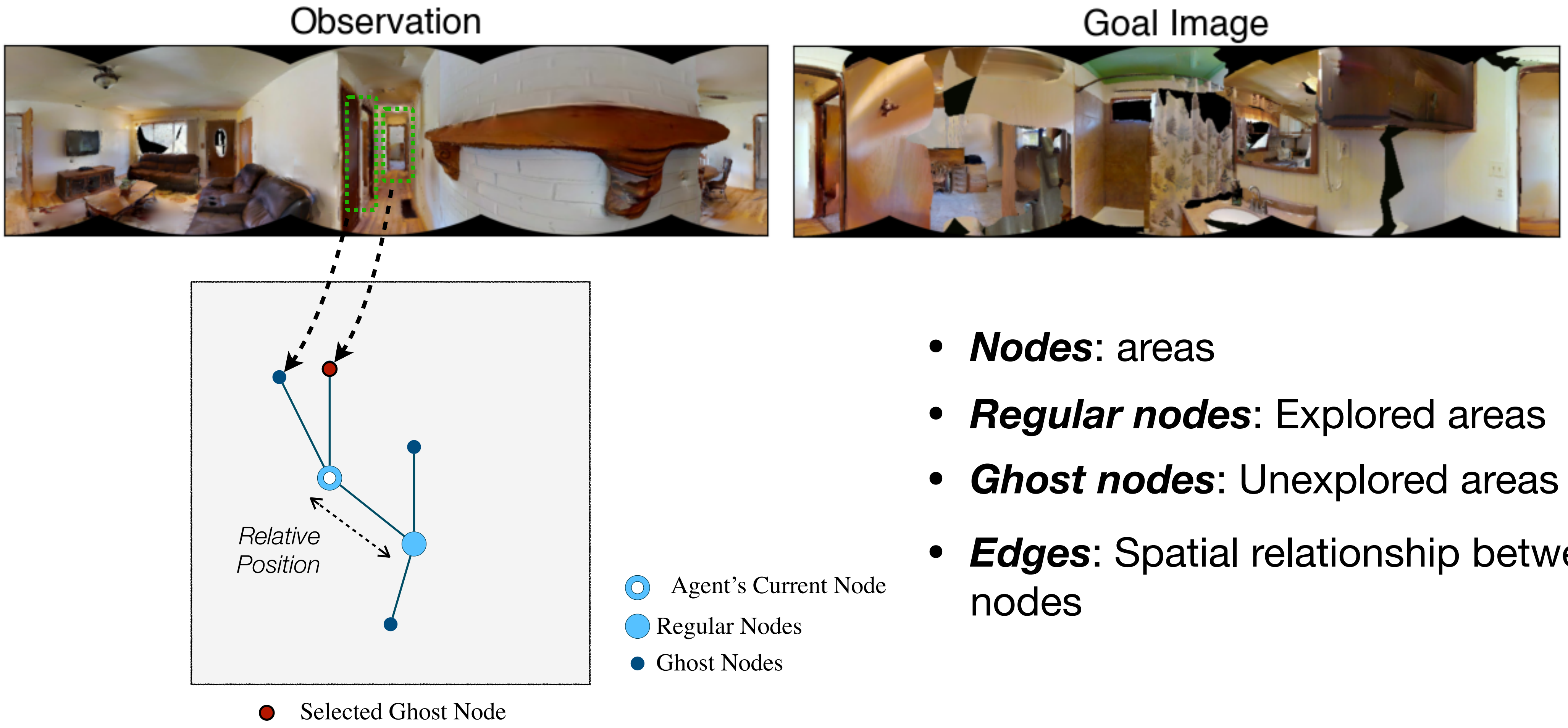


Topological Graph Representation



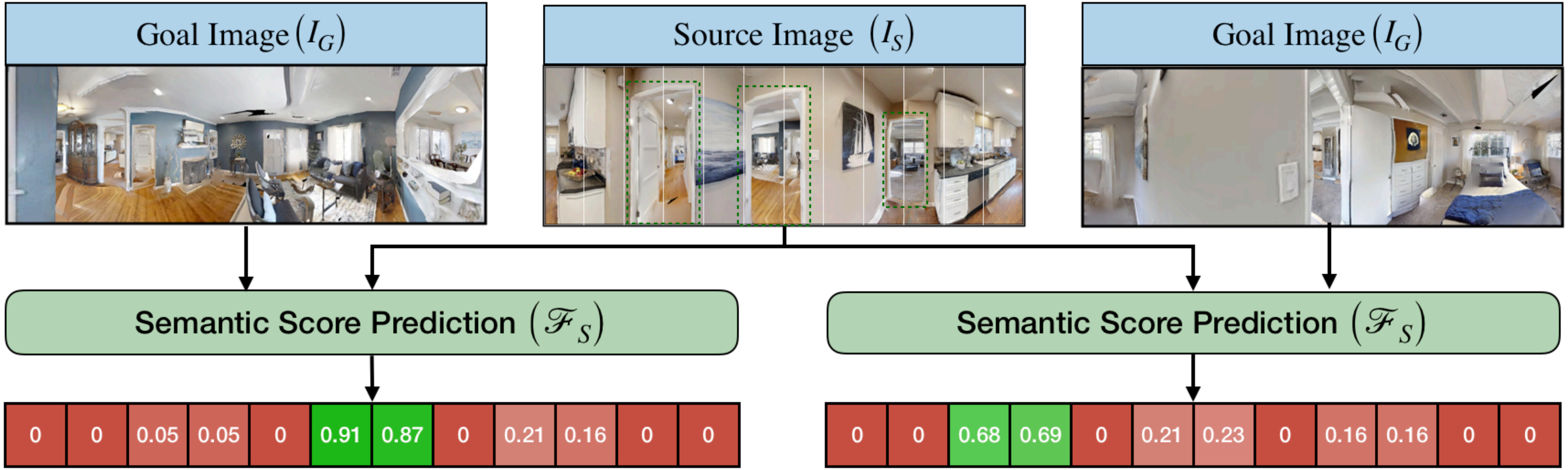
- ***Nodes***: areas
- ***Regular nodes***: Explored areas
- ***Ghost nodes***: Unexplored areas

Topological Graph Representation

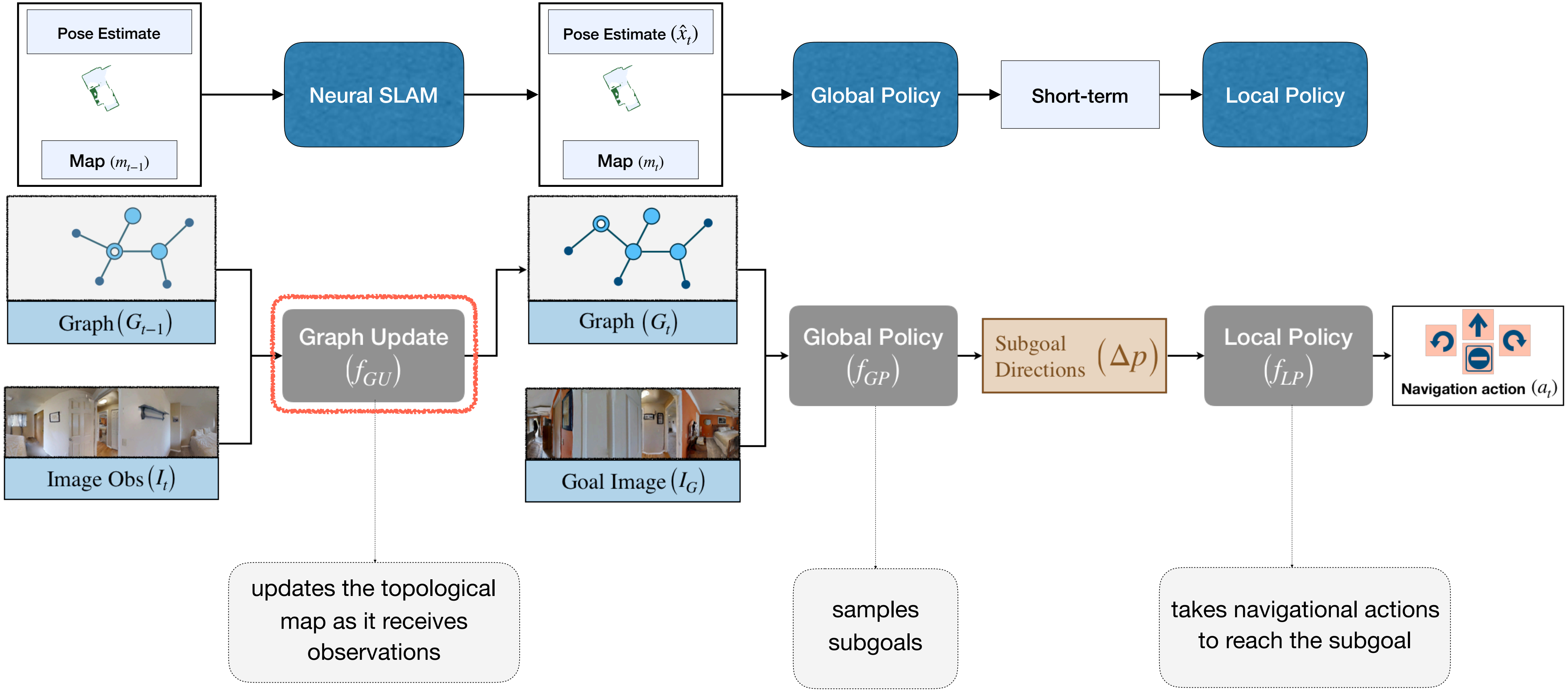


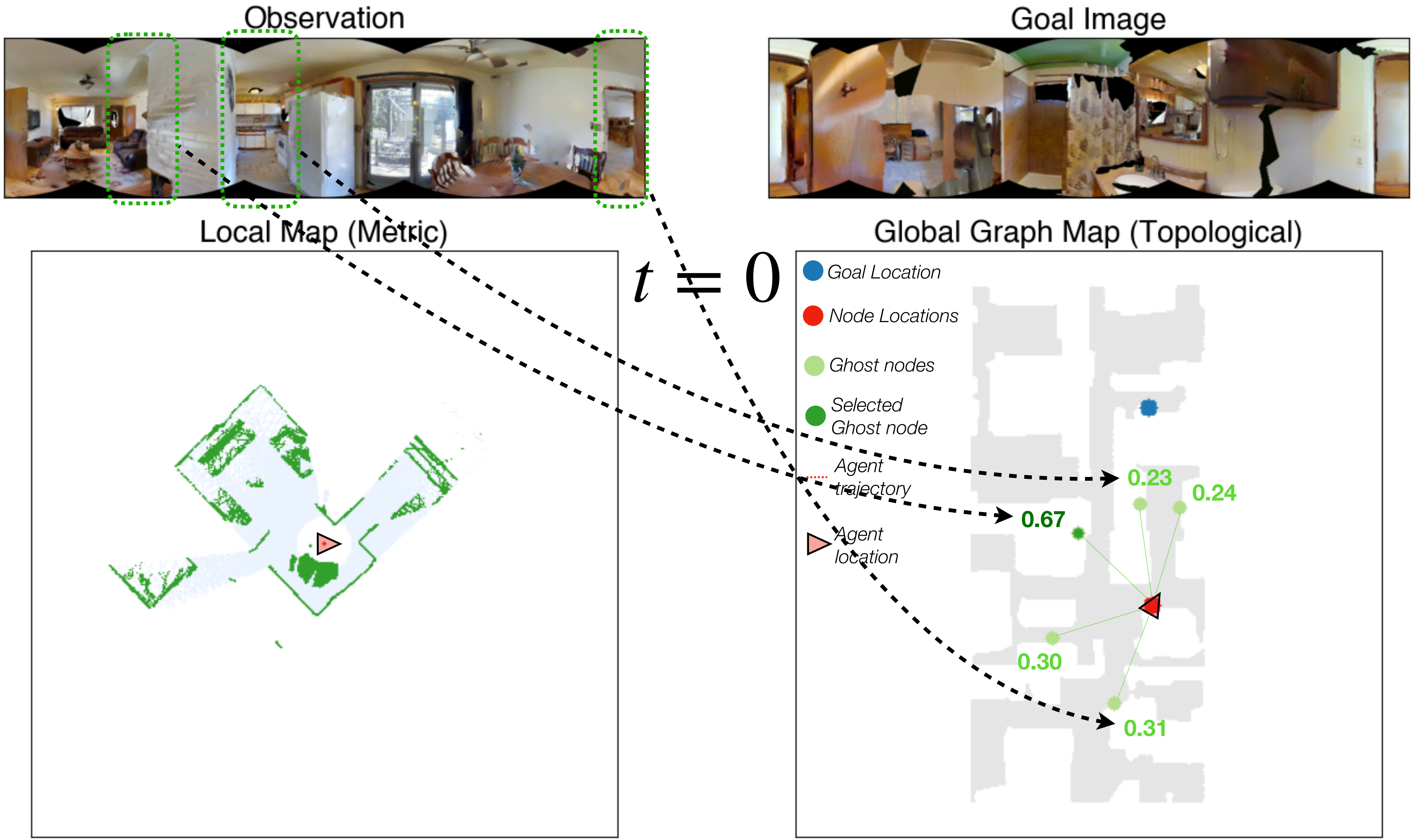
- ***Nodes***: areas
- ***Regular nodes***: Explored areas
- ***Ghost nodes***: Unexplored areas
- ***Edges***: Spatial relationship between nodes

Semantic Prediction



Neural Topological SLAM




Observation
Goal Image
Local Map (Metric)
Global Graph Map (Topological)
$t = 0$
Goal Location
Node Locations
Ghost nodes
Selected Ghost node
Agent trajectory
Agent location
0.23
0.24
0.67
0.30
0.31

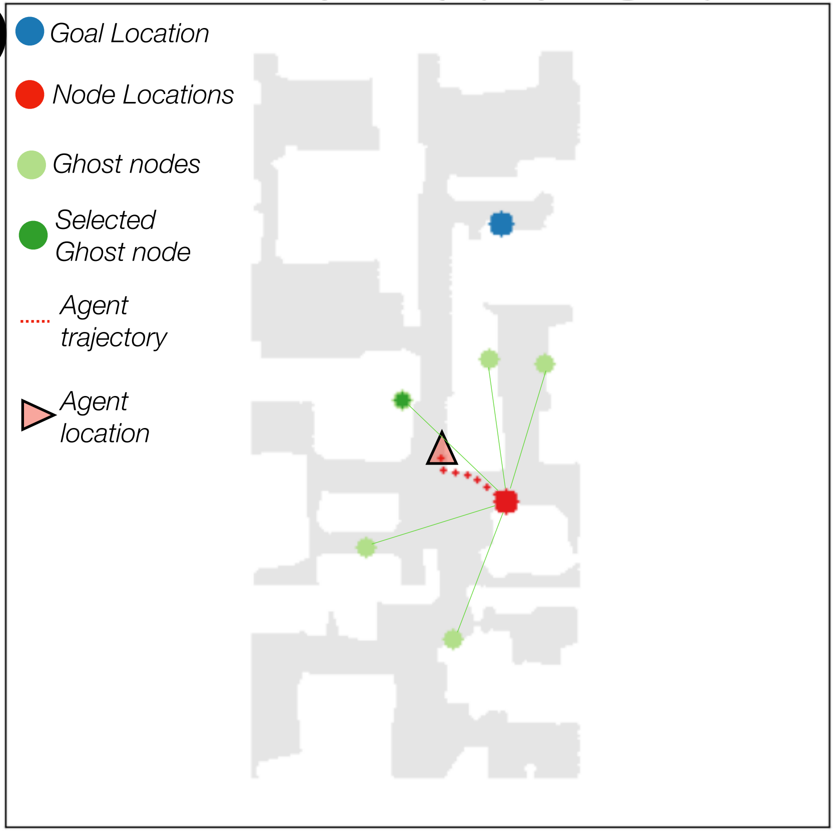
Observation

Goal Image

Local Map (Metric)

$t = 29$

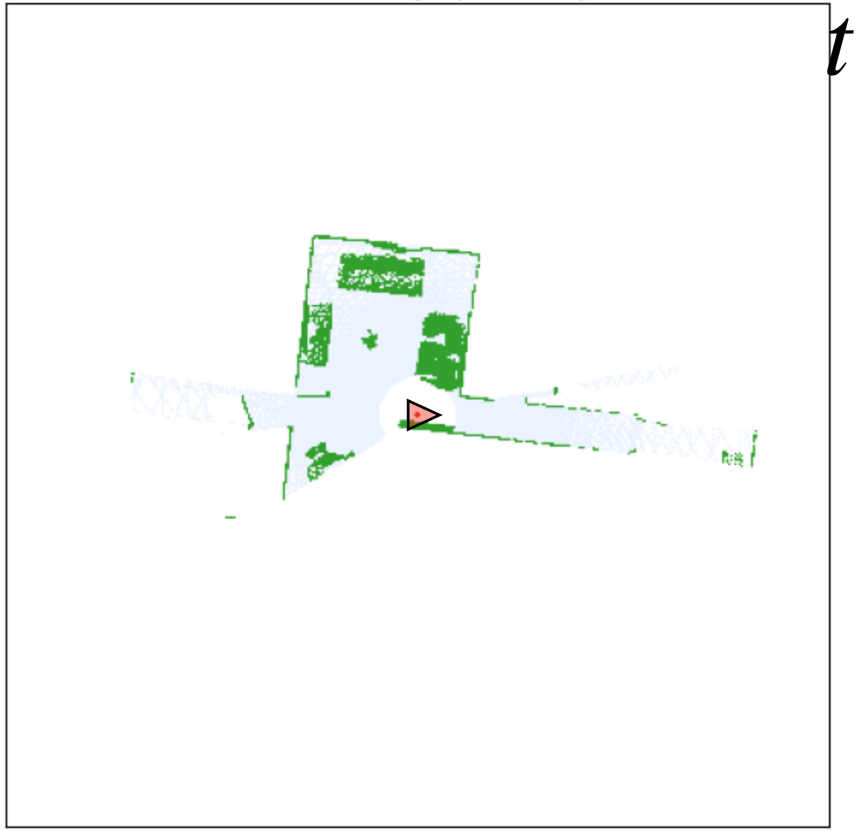
Global Graph Map (Topological)



Observation

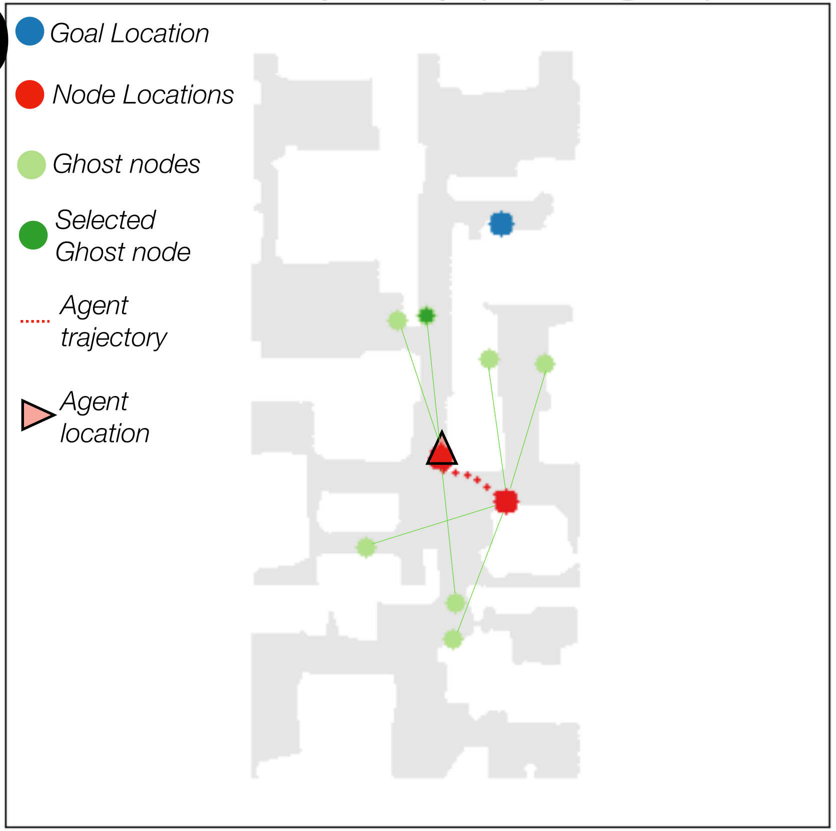
Goal Image

Local Map (Metric)



$t = 30$

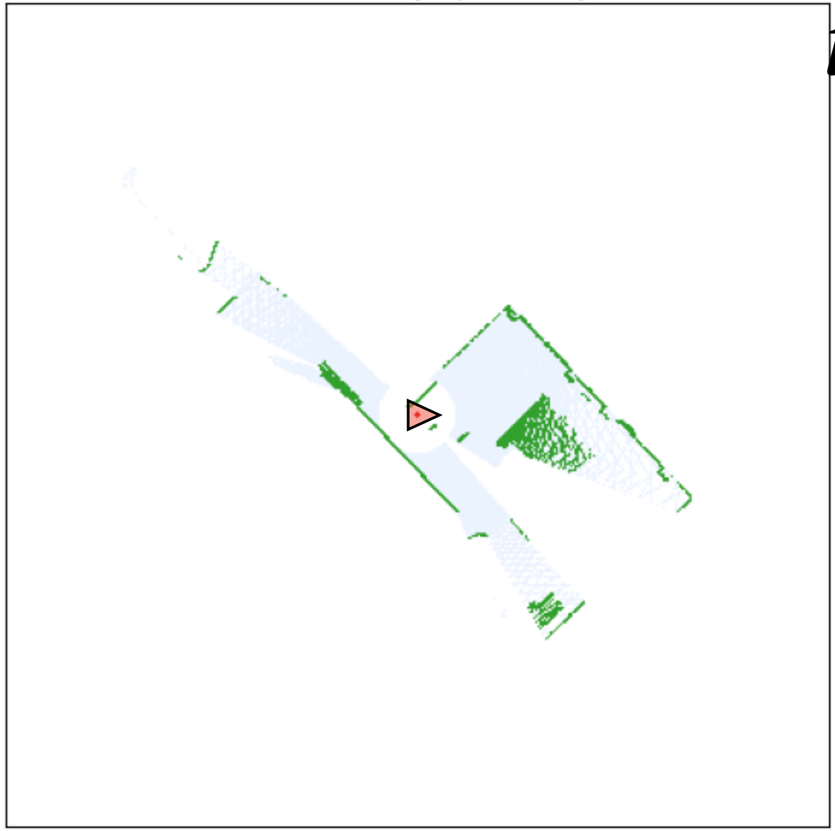
Global Graph Map (Topological)



Observation

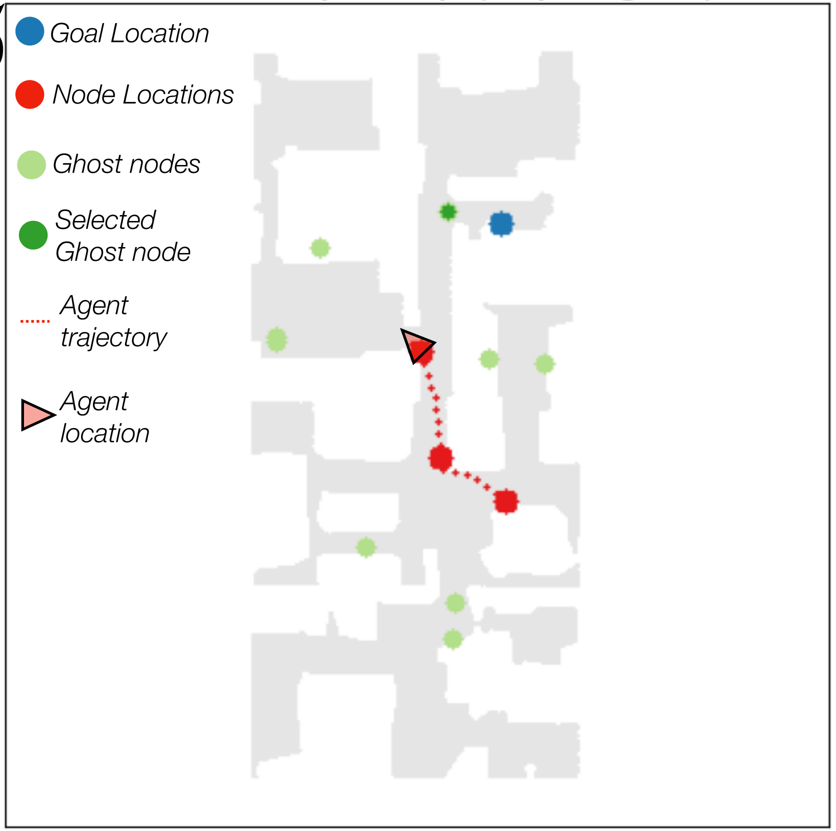
Goal Image

Local Map (Metric)



$t = 45$

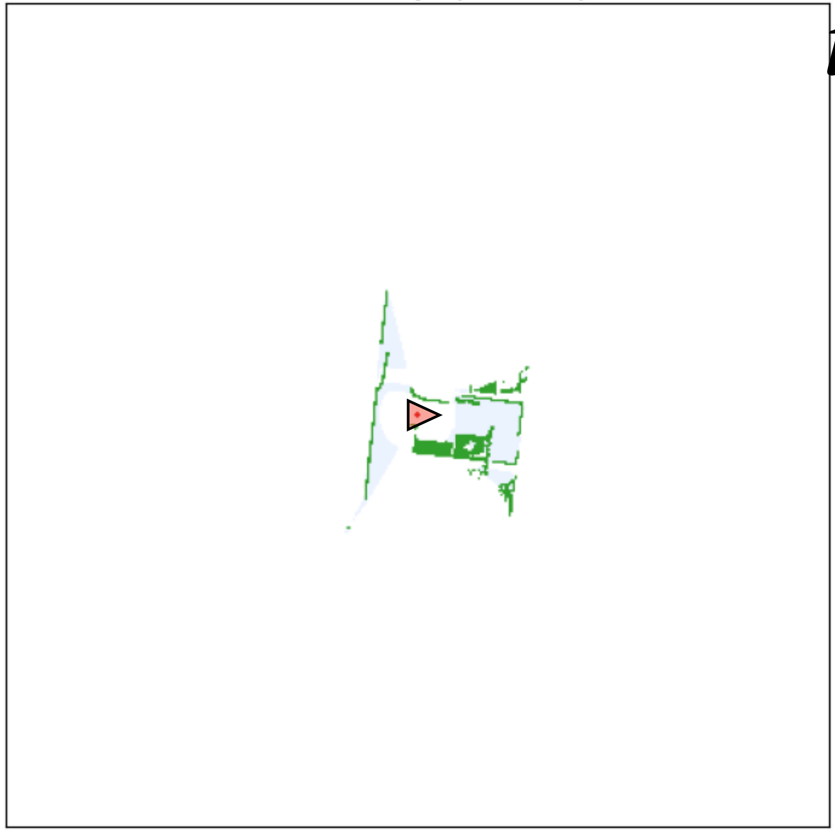
Global Graph Map (Topological)



Observation

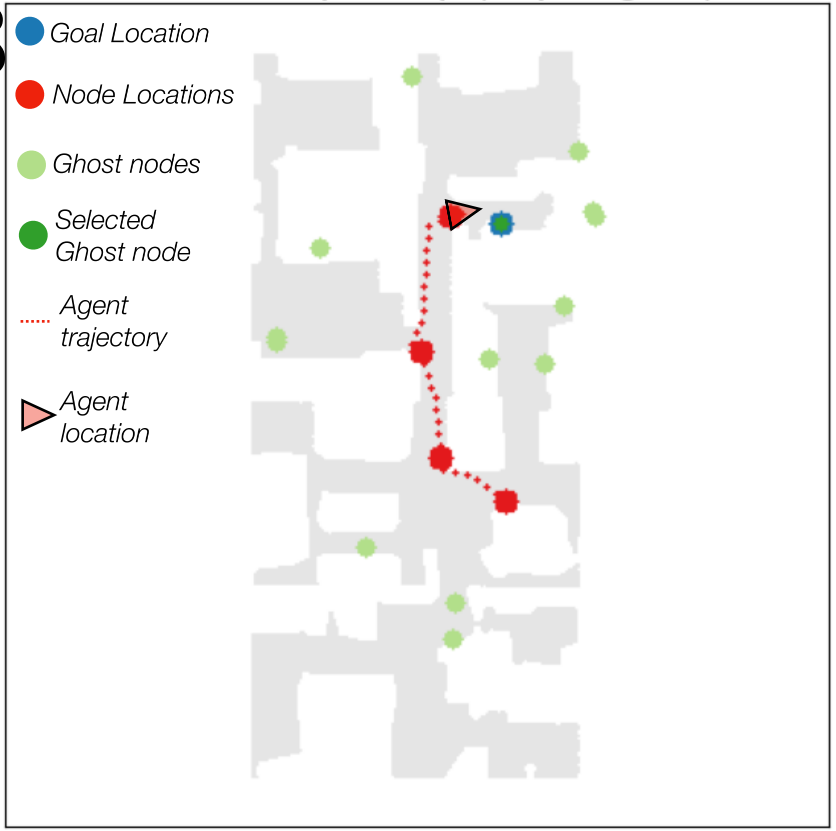
Goal Image

Local Map (Metric)



$t = 78$

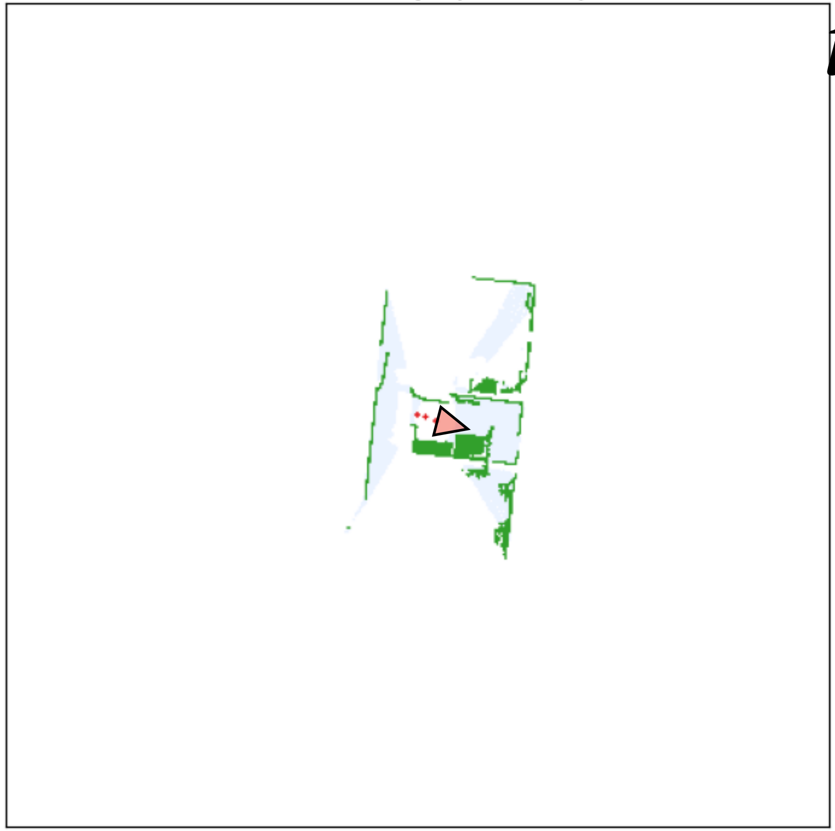
Global Graph Map (Topological)



Observation

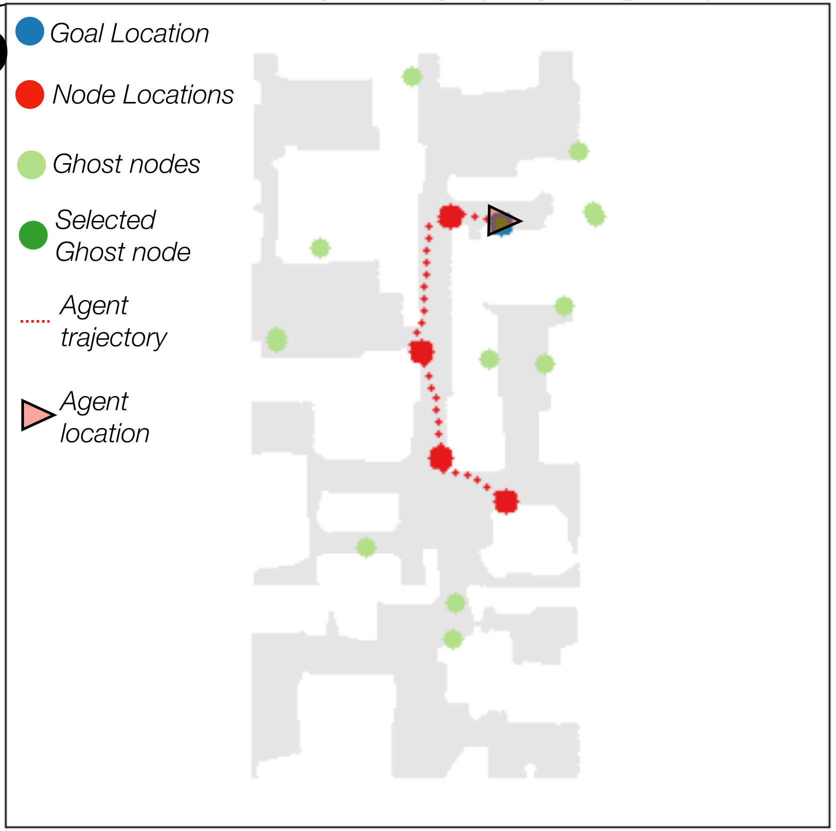
Goal Image

Local Map (Metric)



$t = 86$

Global Graph Map (Topological)



Observation

Goal Image

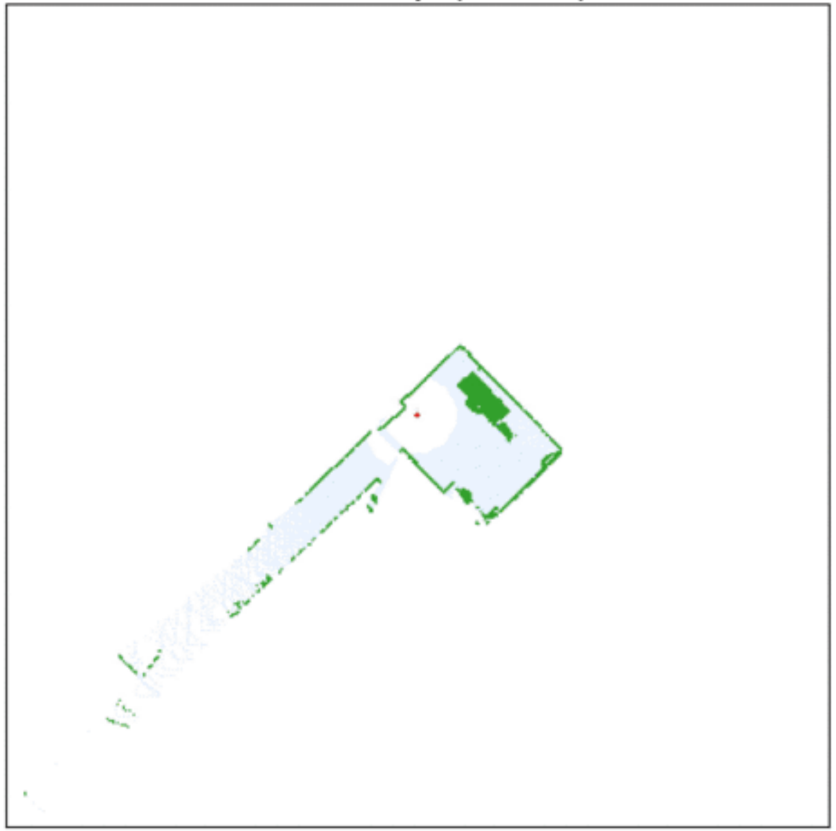
Local Map (Metric)

Global Graph Map (Topological)

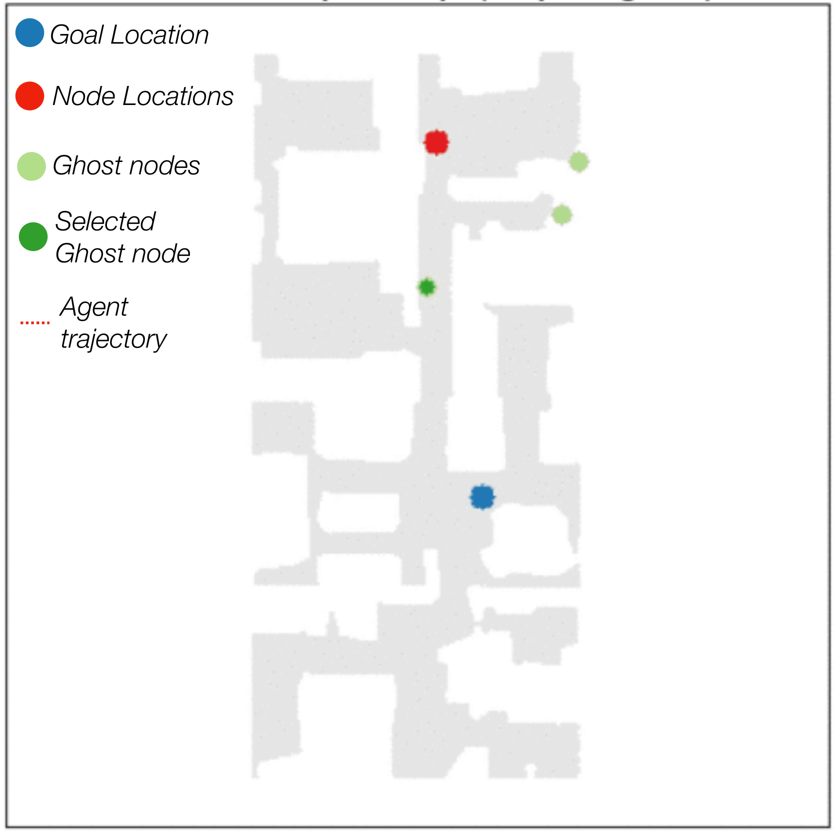
Observation

Goal Image

Local Map (Metric)



Global Graph Map (Topological)



Results

Robustness to Pose Noise

		RGB	RGBD	RGBD (No Noise)	RGBD (No Stop)
End-to-end Learning	LSTM + Imitation	0,10	0,14	0,15	0,18
	LSTM + RL	0,10	0,13	0,14	0,17
Modular Metric Maps	Occupancy Maps + FBE + RL	N/A	0,26	0,31	0,24
	Active Neural SLAM	0,23	0,29	0,35	0,39
Topological Maps	**Neural Topological SLAM**	**0,38**	**0,43**	**0,45**	**0,60**

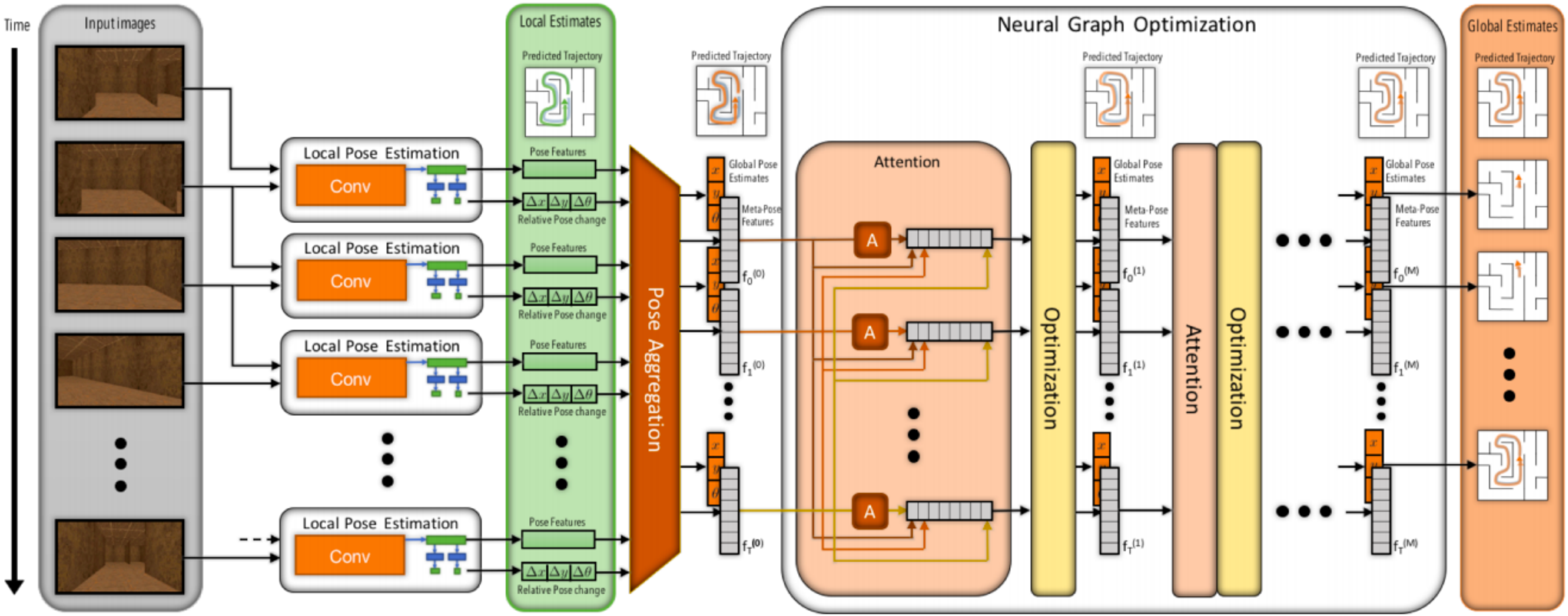
Map based methods are better than vanilla learning methods even in presence of noise

NTS is better than occupancy map models, captures and uses semantic priors.

Pose Estimation: Towards Deep SLAM

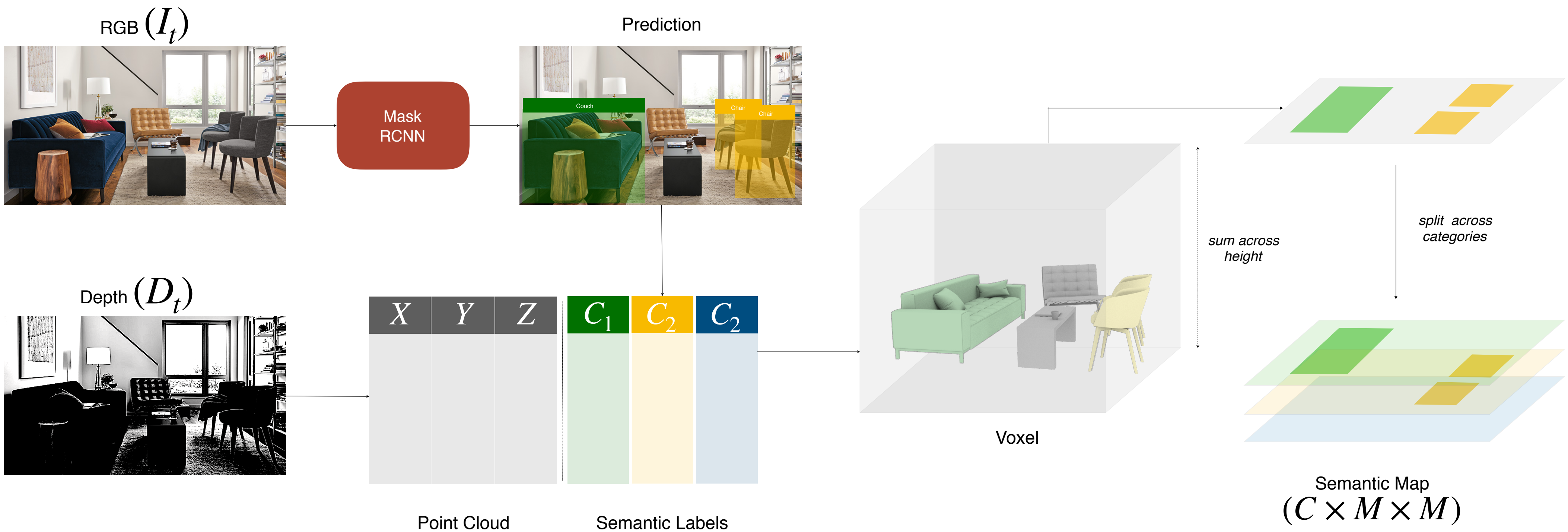
Parisotto et al., CVPR Workshop on Visual SLAM 2018

Pose Estimation: Towards Deep SLAM



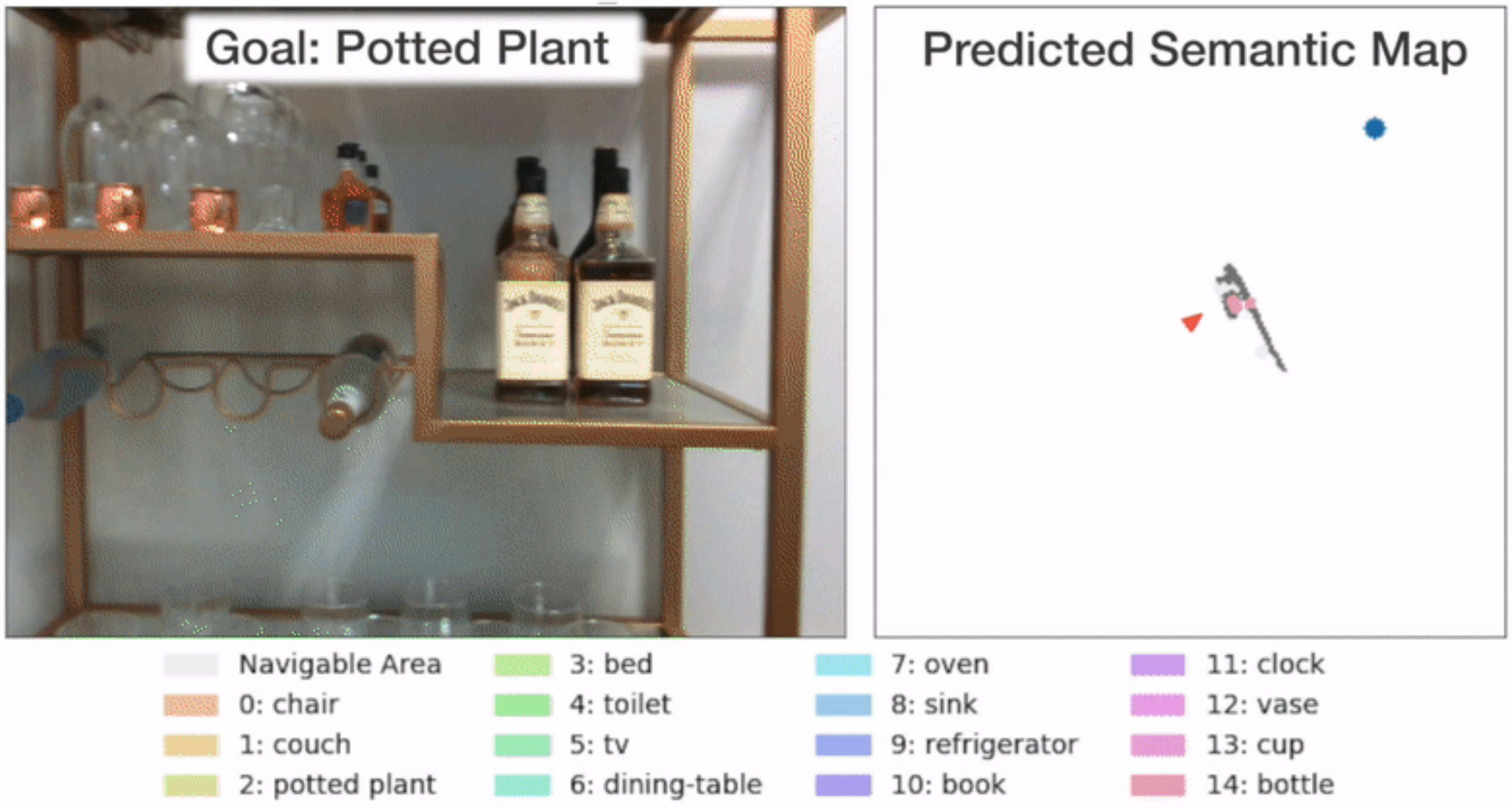
Parisotto et al., CVPR Workshop on Visual SLAM 2018

Explicit Semantic Mapping



Explicit Semantic Mapping

Explicit Semantic Mapping

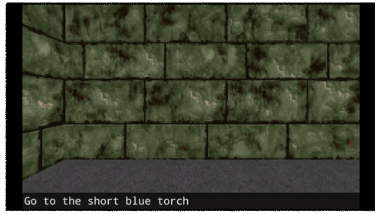


Simulation to Real

Games

ViZDoom

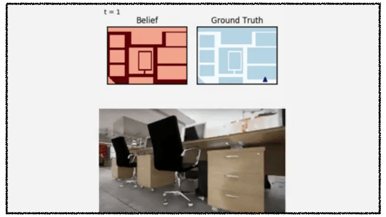
[CL *AAAI-17*]



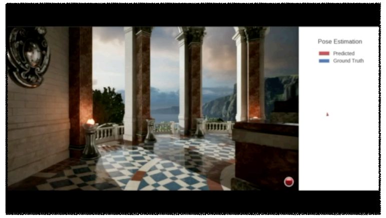
[CMPRS *AAAI-18*]

Photorealistic simulation

Unreal



[CPS *ICLR-18*]

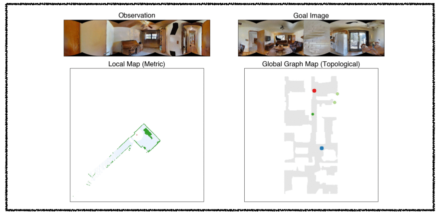


[PCZS *CVPR-18 (w)*]

Reconstructed simulation

Habitat (Gibson, MP3D)

[CGSGG *ICLR-20*]



[CSGG *CVPR-20*]

Physical Domain Gap

Visual Domain Gap

Real-world

?

Simulation to Real

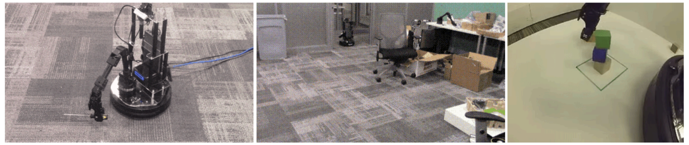
- Physical Domain Gap
  - Actuation noise models
  - Sensor noise models
- Visual Domain Gap
  - Image Translation
  - Policy-based

PyRobot

PyRobot is a light weight, high-level interface which provides hardware independent APIs for robotic manipulation and navigation. This repository also contains the low-level stack for LoCoBot, a low cost mobile manipulator hardware platform.

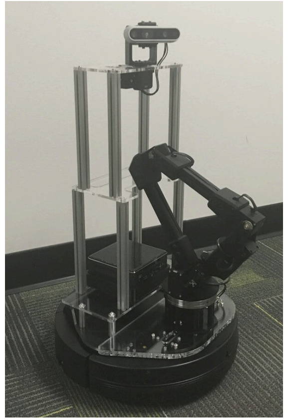
- What can you do with PyRobot?
- Installation
- Getting Started
- The Team
- Citation
- License
- Future features

What can you do with PyRobot?



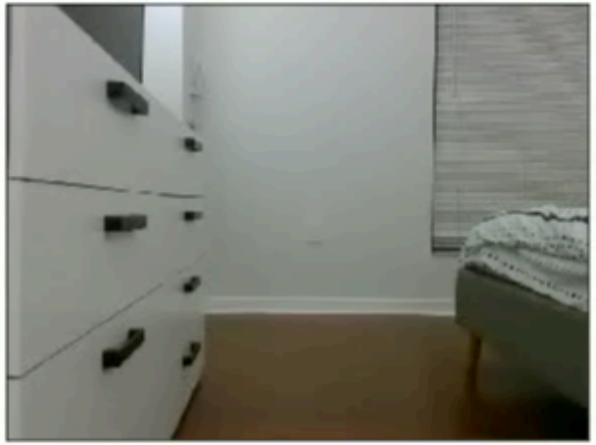
pyrobot.org

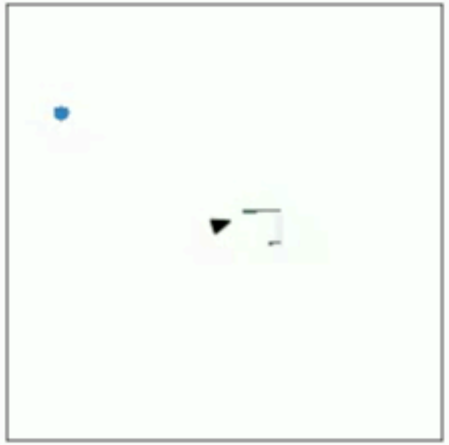
LoCoBot



locobot.org

Simulation to Real





Building Intelligent Agents

- Navigate Autonomously
- Localize and Plan
- Multi-modal Input
- Perceptive Human Speech
- Reason & Understand Language
- Recognize objects