RLLAB
http://rllab.snu.ac.kr

Introduction to Visual Navigation

Presenter: Nuri Kim
Robot Learning Laboratory

Who am I?

RLLAB
http://rllab.snu.ac.kr

Education

Seoul National University
Ph.D. Candidate, Electrical and Computer Engineering
2016 - 2022

Korea University
Bachelor's degree, Electrical Engineering
2012 - 2016

Grade: 4.2 / 4.5

Activities and societies: Board game club 'SulKUs'
Study club 'HandS'

Graduated with Great Honor in Electrical engineering, Korea University. Received Honors Scholarship, fall, 2012 and fall 2013. Received National Science and Engineering Scholarship, 2014, 2015. Received Creative Challenger Scholarship, fall, 2015.

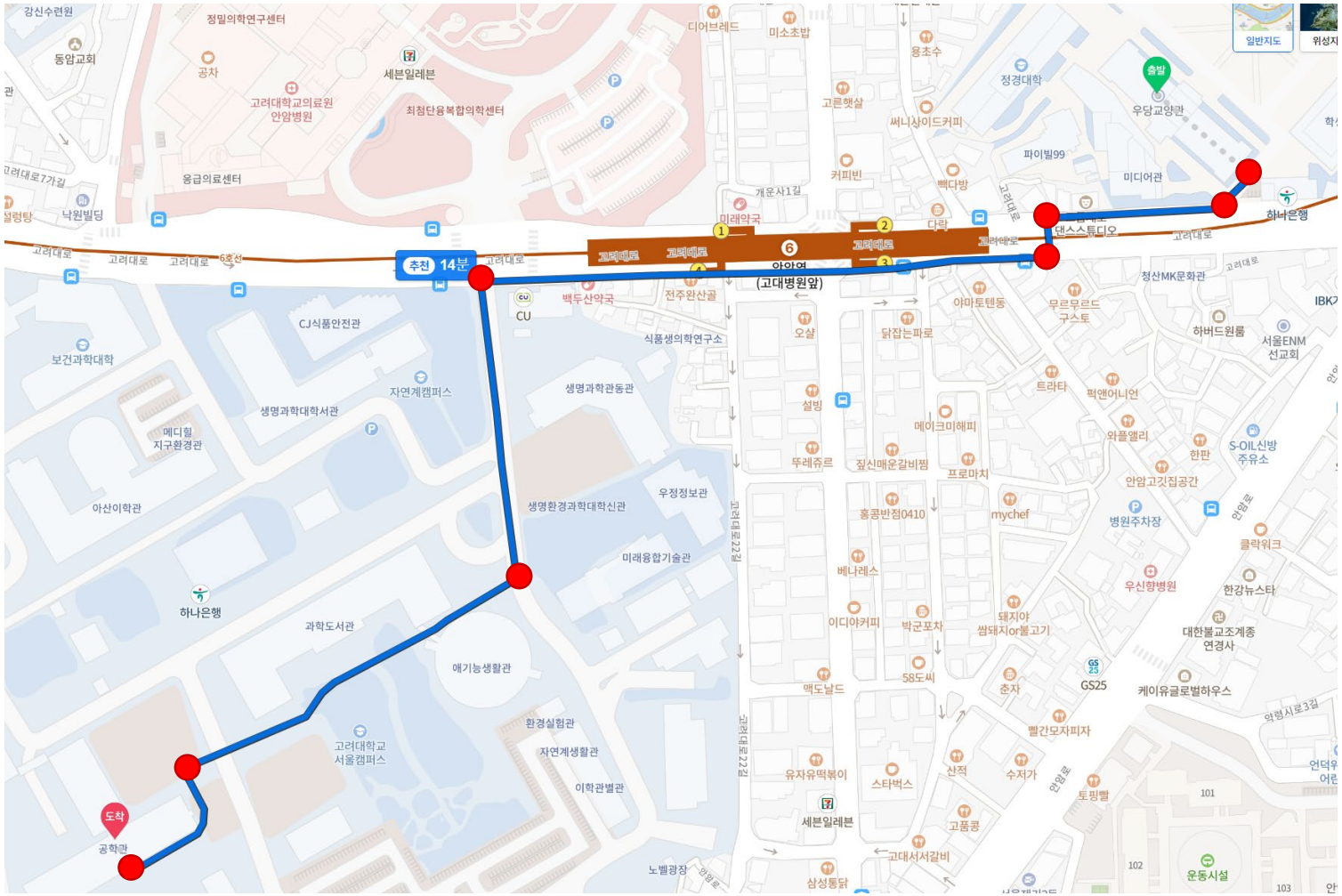
The Australian National University
Electrical and Electronics Engineering
2014 - 2014

Overseas Studies Program: semester at Australian National University, focus on Electrical engineering.

Introduction: Why *Visual?*

RLLAB
http://rllab.snu.ac.kr

- Map is **provided** → Easy 😇



Introduction: Why *Visual?*

RLLAB
http://rllab.snu.ac.kr

- Map is **not provided** → Hard 😱
  - Getting exact map and location is hard.
    - Sensing **obstacles**
    - Estimating **current location**

  😭 The problem is **NOISE**
    - **Sensor noise**: Error in lidar sensor
    - **Actuation noise**: Uncertainty in robot pose

  😎 Vision-based methods
    1. Cheap camera sensor
    2. Semantic navigation -> Human-like navigation
    3. Better performance than rule-based method

Contents

RLLAB
http://rllab.snu.ac.kr

- Resources: Simulator & Dataset
- Memory Structures
  - Metric map
  - Topological map
- Visual Navigation Tasks
  - Exploration
  - Path Following
  - Active Vision
  - Target-driven navigation
    - Point Goal Navigation
    - Image Goal Navigation
    - Object Goal Navigation
  - Vision and Language Navigation
- Research Tips

Contents

RLLAB
http://rllab.snu.ac.kr

- **Resources: Simulator & Dataset**
- Visual Navigation Tasks
  - Exploration
  - Path Following
  - Active Vision
  - Target-driven navigation
    - Point Goal Navigation
    - Image Goal Navigation
    - Object Goal Navigation
  - Vision and Language Navigation
- Memory Structures
  - Metric map
  - Topological map
- Research Tips

Resources

RLLAB
http://rllab.snu.ac.kr

- Visual Navigation

Simulator

Habitat
iGibson
AI2-Thor
Matterport3D
TDW
VirtualHome
VizDoom

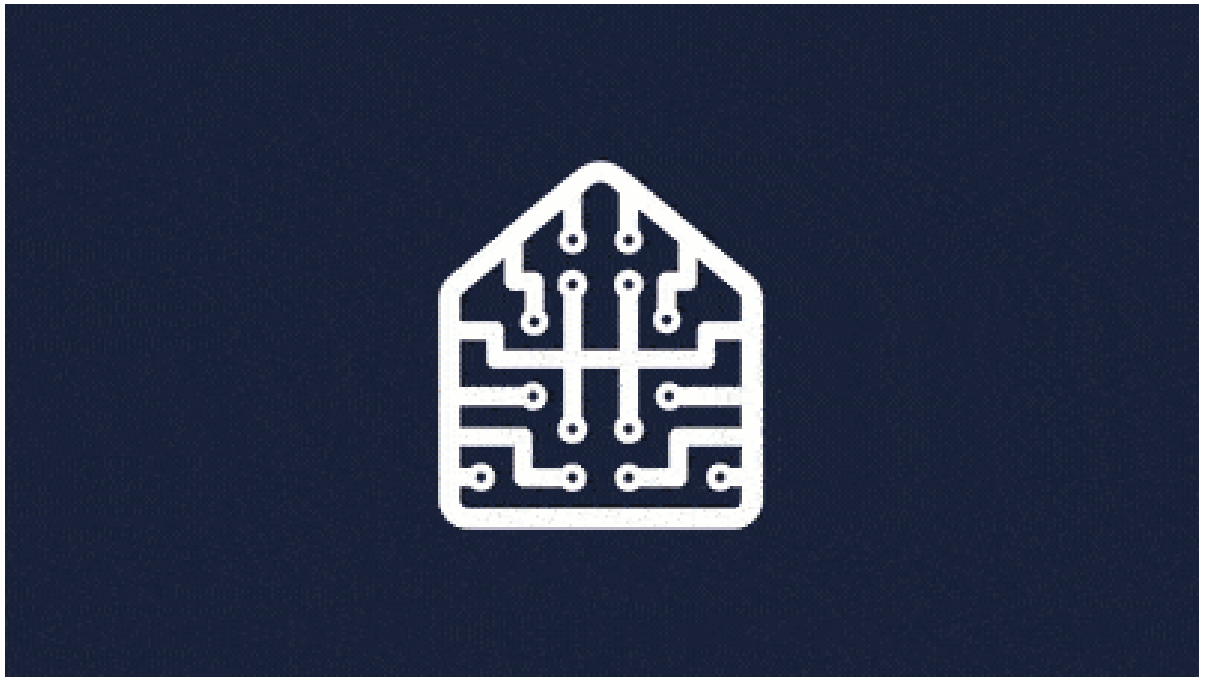
Dataset

Matterport3D
Gibson
Room-to-Room (R2R)
RealEstate 10k
Replica

Habitat

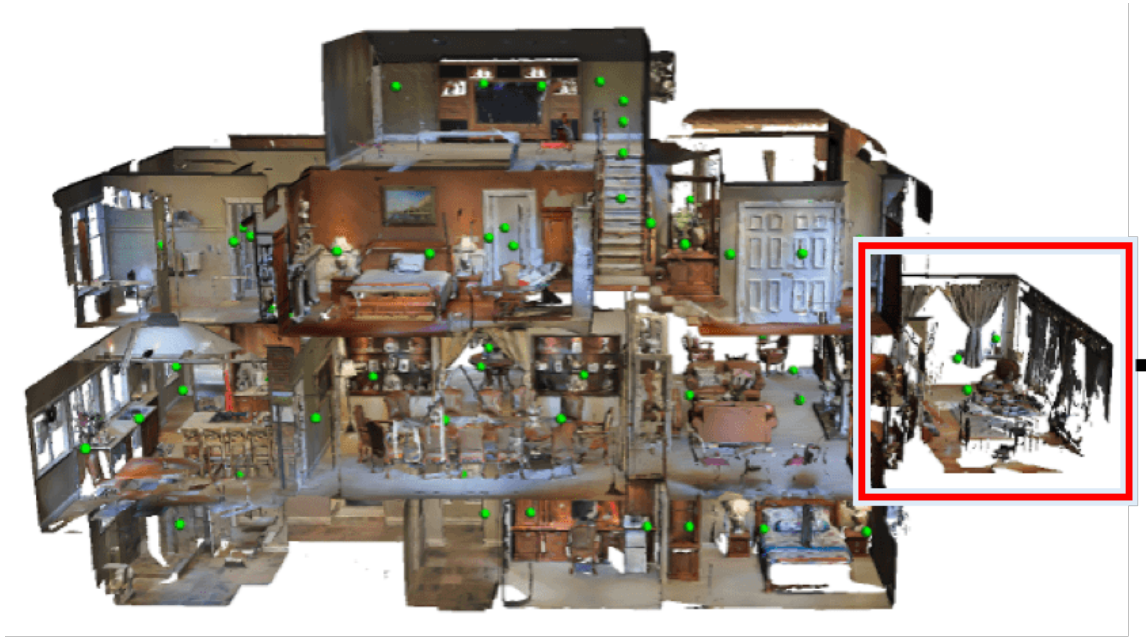
RLLAB
http://rllab.snu.ac.kr

- A high-performance physics-enabled 3D simulator with support for:
  - 3D scans of indoor/outdoor spaces (with built-in support for HM3D, MatterPort3D, Gibson, Replica, and other datasets)
  - CAD models of spaces and piecewise-rigid objects (e.g. ReplicaCAD, YCB, Google Scanned Objects),
  - Configurable sensors (RGB-D cameras, egomotion sensing)

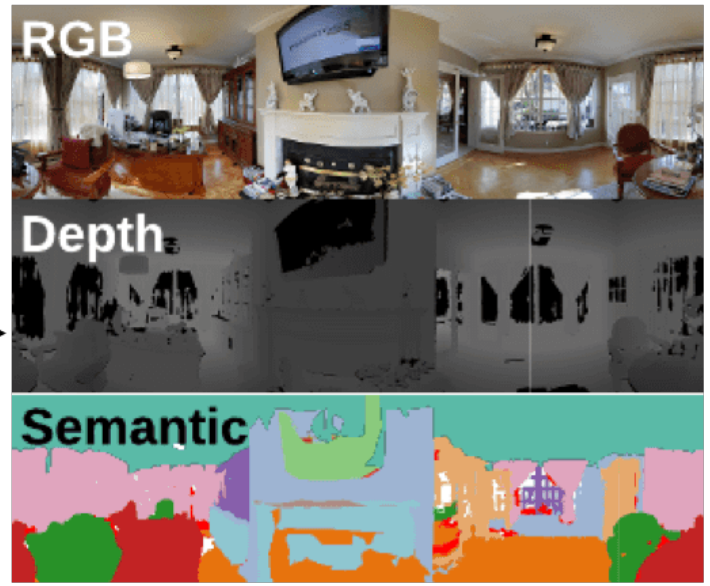


Matterport 3D dataset

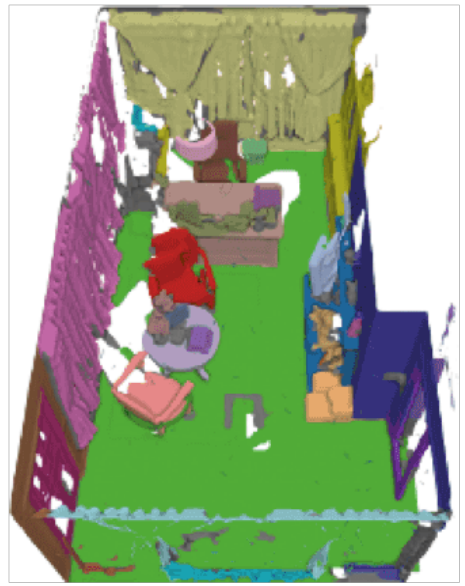
RLLAB
http://rllab.snu.ac.kr



Textured 3D Mesh



Panoramas



Object Instances

Matterport Camera

RLLAB
http://rllab.snu.ac.kr

Coex B1 (Collected by RLLAB Navi Team)

iGibson

RLLAB
http://rllab.snu.ac.kr

- iGibson is a simulation environment providing fast visual rendering and physics simulation based on Bullet.
- iGibson is equipped with fifteen fully interactive high quality scenes, hundreds of large 3D scenes reconstructed from real homes and offices, and compatibility with datasets like CubiCasa5K and 3D-Front, providing 12000+ additional interactive scenes.

Physical Interaction with Articulated Objects

More than 500 object models

Sourced from open source datasets and cleaned up

Articulated objects can be operated by agents

Domain Randomization for Endless Variations

Domain randomization for:
1. Visual textures
2. Dynamics
3. Object instance

AI2-Thor

RLLAB
http://rllab.snu.ac.kr

Contents

RLLAB
http://rllab.snu.ac.kr

- Resources: Simulator & Dataset
- **Memory Structures**
  - Metric map
  - Topological map
- Visual Navigation Tasks
  - Exploration
  - Path Following
  - Active Vision
  - Target-driven navigation
    - Point Goal Navigation
    - Image Goal Navigation
    - Object Goal Navigation
  - Vision and Language Navigation
- Research Tips

RLLAB
http://rllab.snu.ac.kr

Memory Structures

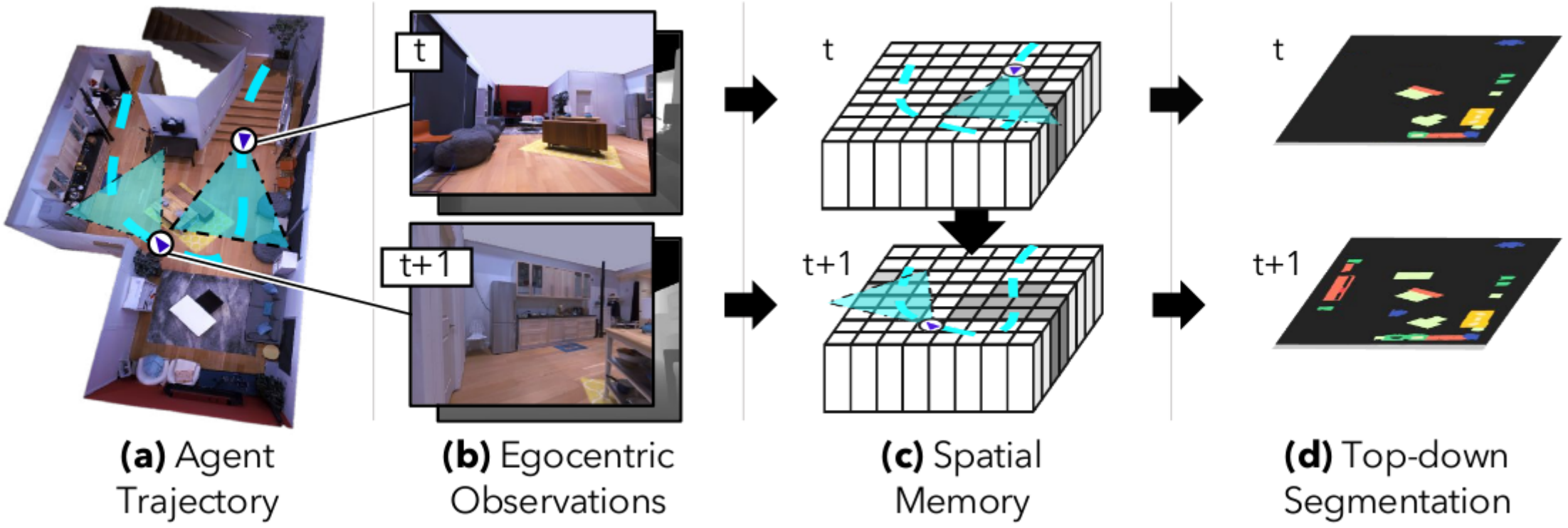
Metric Map

Topological Map

Metric Map

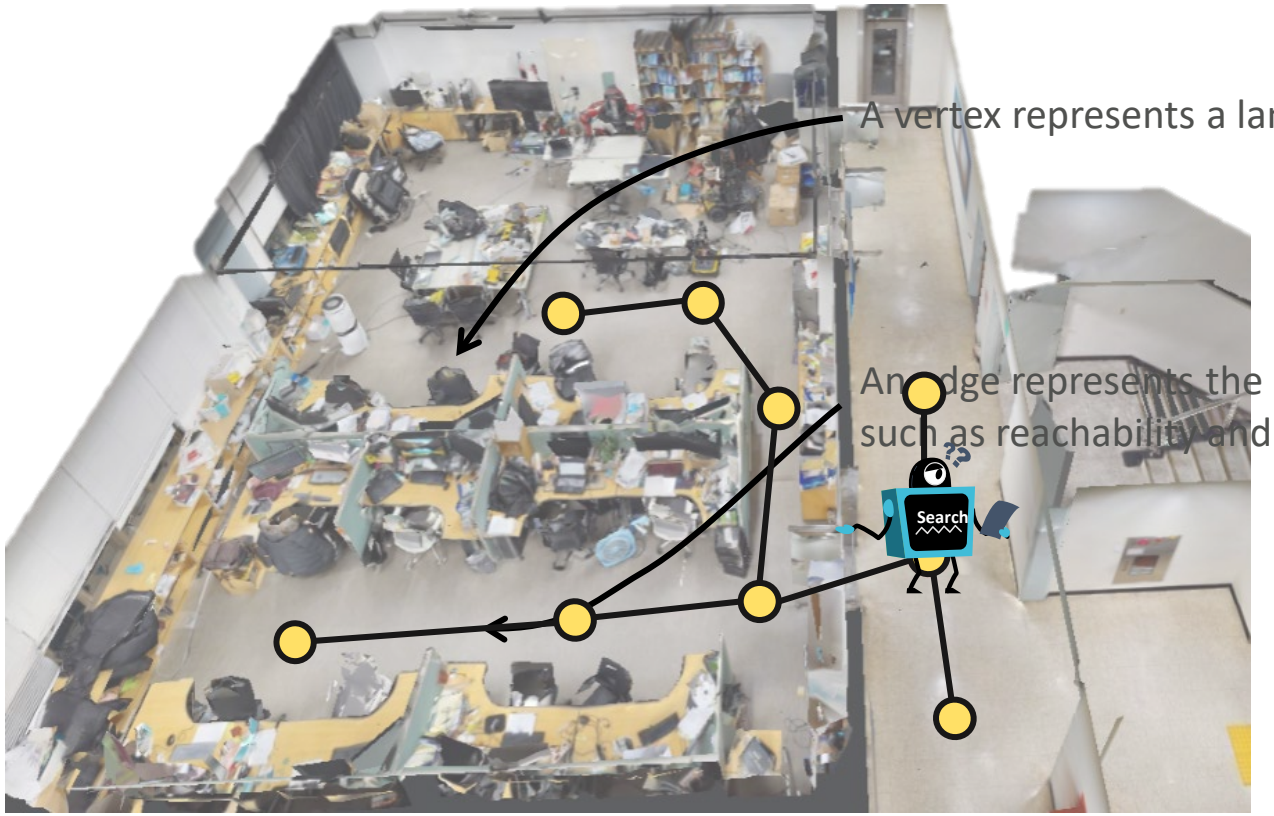
RLLAB
http://rllab.snu.ac.kr

- "Semantic MapNet: Building Allocentric Semantic Maps and Representations from Egocentric Views," 2021 AAAI
  - It introduces a semantic metric map memory architecture for navigation.



Topology Map

RLLAB
http://rllab.snu.ac.kr

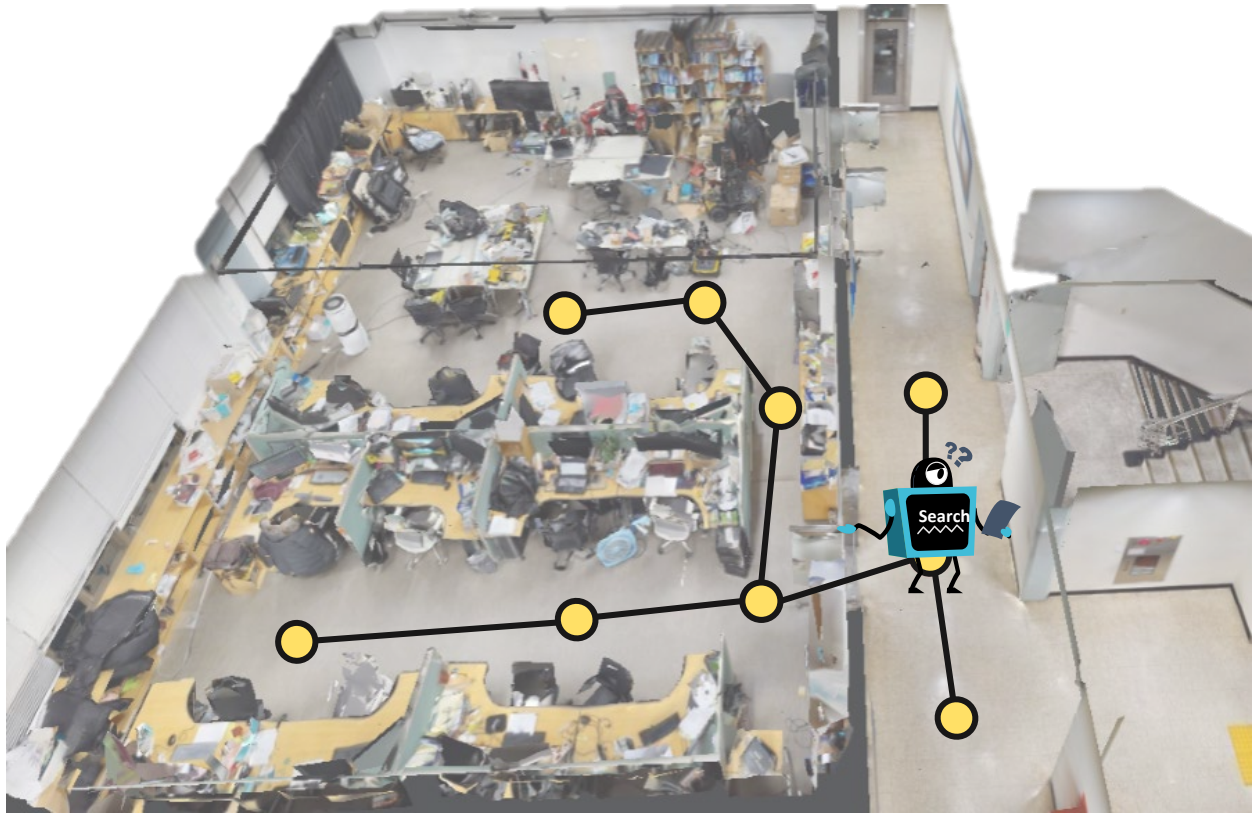


A vertex represents a landmark in the environment

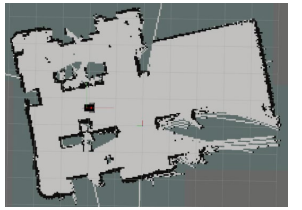
An edge represents the relationship between two vertices, such as reachability and proximity

Topology Map

RLLAB
http://rllab.snu.ac.kr



- **Pros:**
  - concise and sparse
  - accurate geometric information is not required
- **Cons:**
  - can be less accurate than SLAM-based metric map



Contents

RLLAB
http://rllab.snu.ac.kr

- Resources: Simulator & Dataset
- Memory Structures
  - Metric map
  - Topological map
- **Visual Navigation Tasks**
  - Exploration
  - Path Following
  - Active Vision
  - Point Goal Navigation
  - Image Goal Navigation
  - Object Goal Navigation
  - Vision and Language Navigation
- Research Tips

RLLAB
http://rllab.snu.ac.kr

Exploration

Learning To Explore Using Active Neural SLAM, ICLR 2020

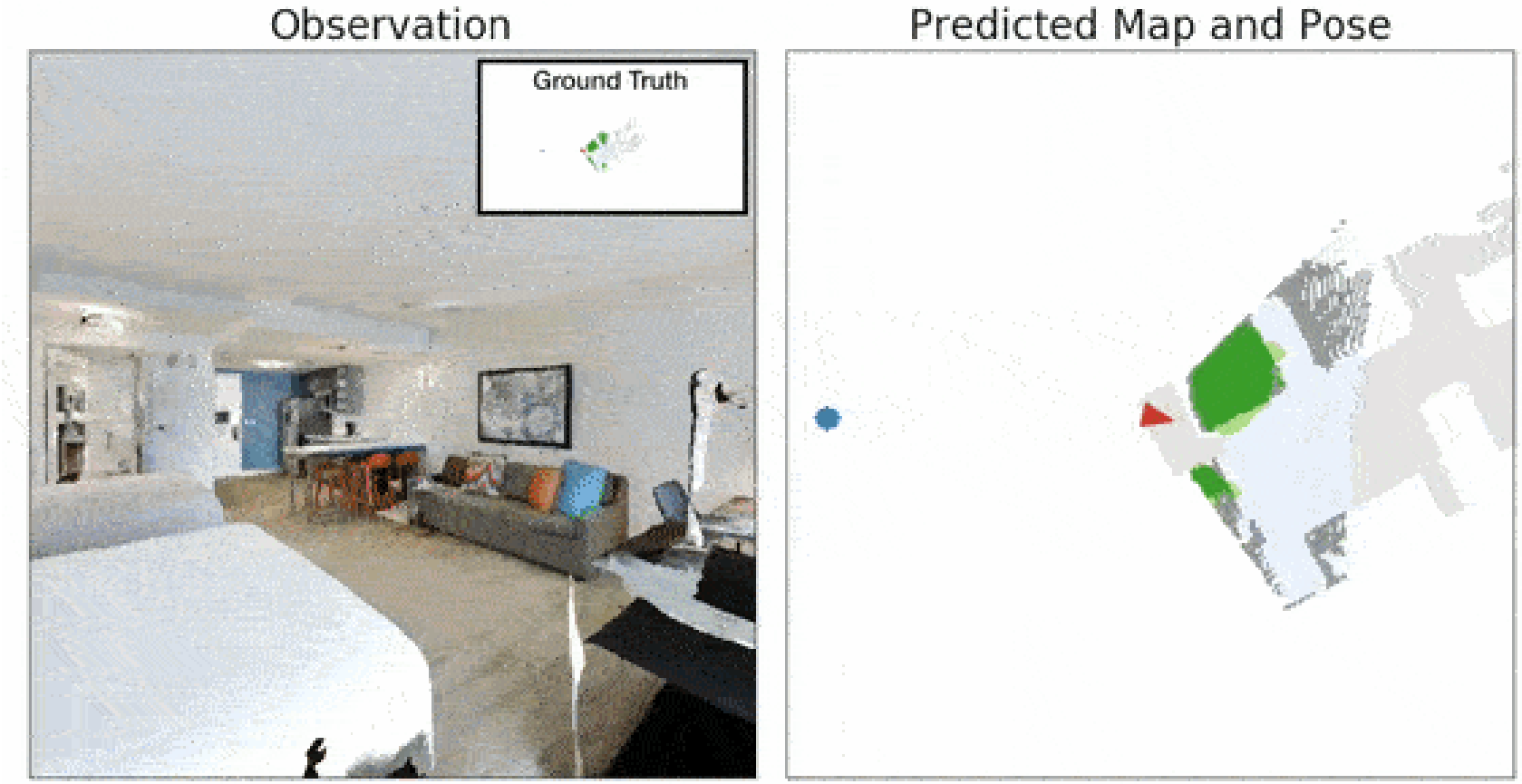
Exploration

RLLAB
http://rllab.snu.ac.kr

- "Learning To Explore Using Active Neural SLAM", ICLR 2020
  - **Exploration**: how to efficiently visit as much of the environment. It is useful for maximizing the coverage.
  - Learning about mapping, state-estimation, and path-planning purely from data in an **end-to-end manner** can be ***expensive***.
  - Combining advantages of rule-based method and learning method

Active Neural SLAM

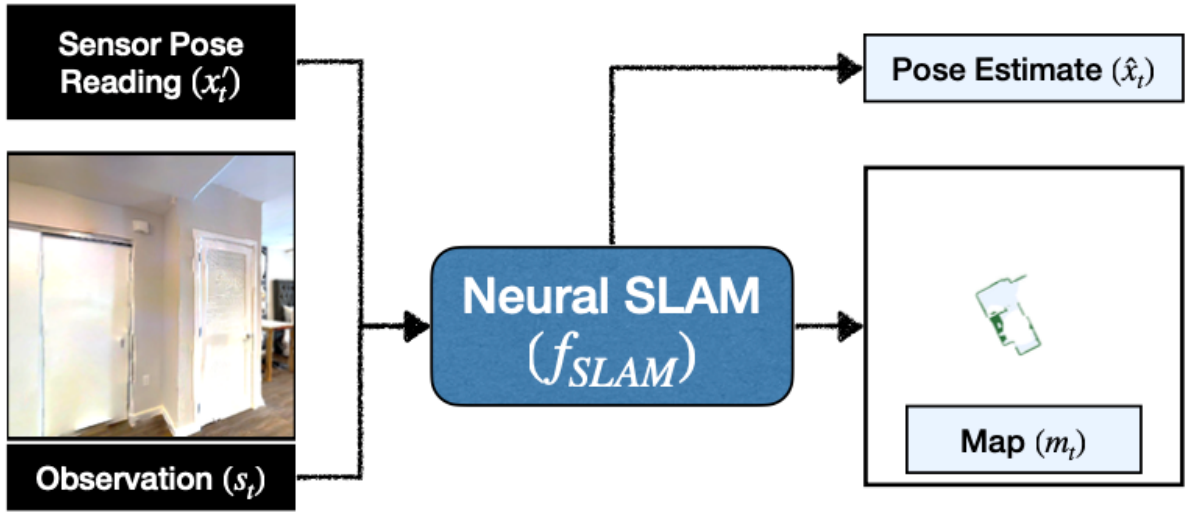
RLLAB
http://rllab.snu.ac.kr



Devendra Singh Chaplot, Dhiraj Gandhi, Saurabh Gupta, Abhinav Gupta, Ruslan Salakhutdinov, "**Learning To Explore Using Active Neural SLAM.**" *ICLR*, 2020.

Active Neural SLAM

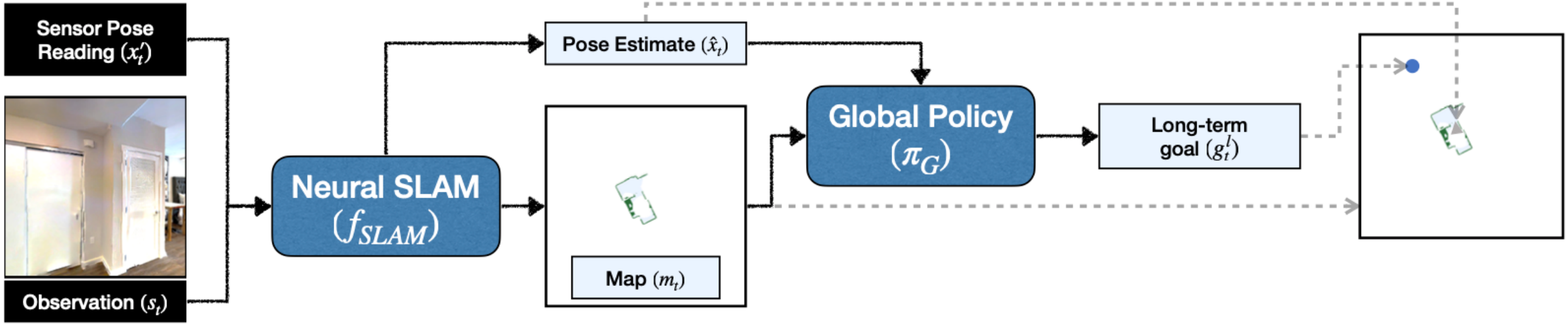
RLLAB
http://rllab.snu.ac.kr



Devendra Singh Chaplot, Dhiraj Gandhi, Saurabh Gupta, Abhinav Gupta, Ruslan Salakhutdinov, "**Learning To Explore Using Active Neural SLAM.**" *ICLR*, 2020.

Active Neural SLAM

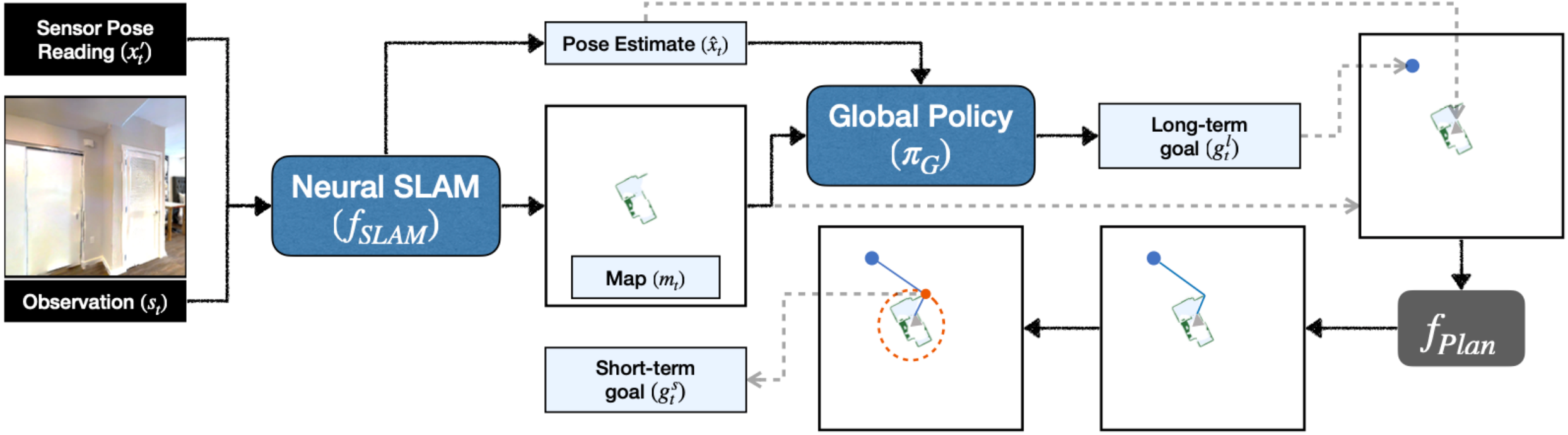
RLLAB
http://rllab.snu.ac.kr



Devendra Singh Chaplot, Dhiraj Gandhi, Saurabh Gupta, Abhinav Gupta, Ruslan Salakhutdinov, "**Learning To Explore Using Active Neural SLAM.**" *ICLR*, 2020.

Active Neural SLAM

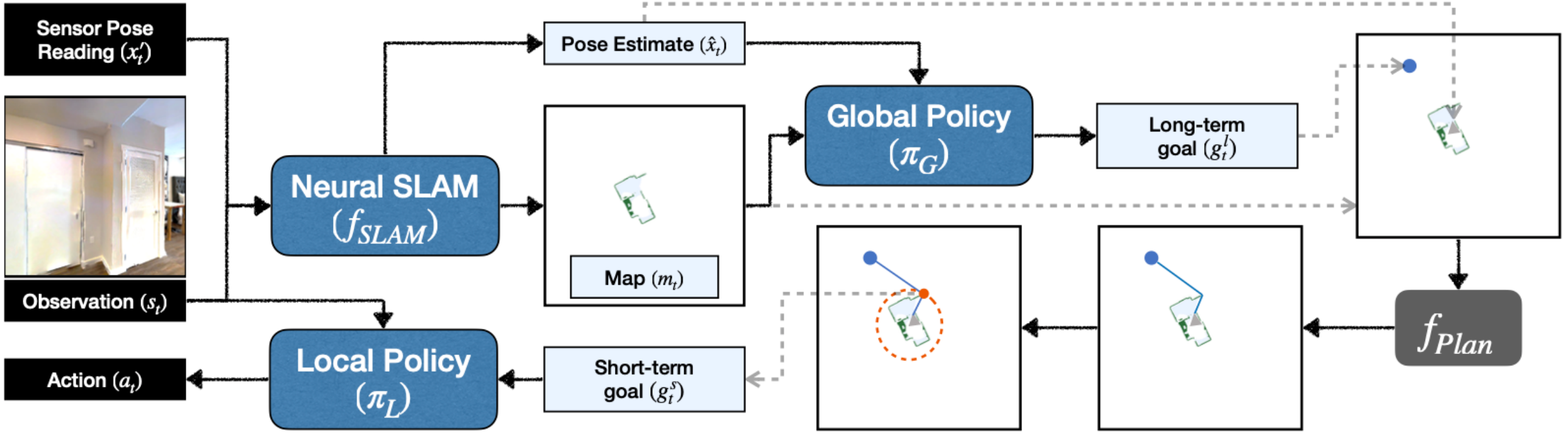
RLLAB
http://rllab.snu.ac.kr



Devendra Singh Chaplot, Dhiraj Gandhi, Saurabh Gupta, Abhinav Gupta, Ruslan Salakhutdinov, "**Learning To Explore Using Active Neural SLAM.**" *ICLR*, 2020.

Active Neural SLAM

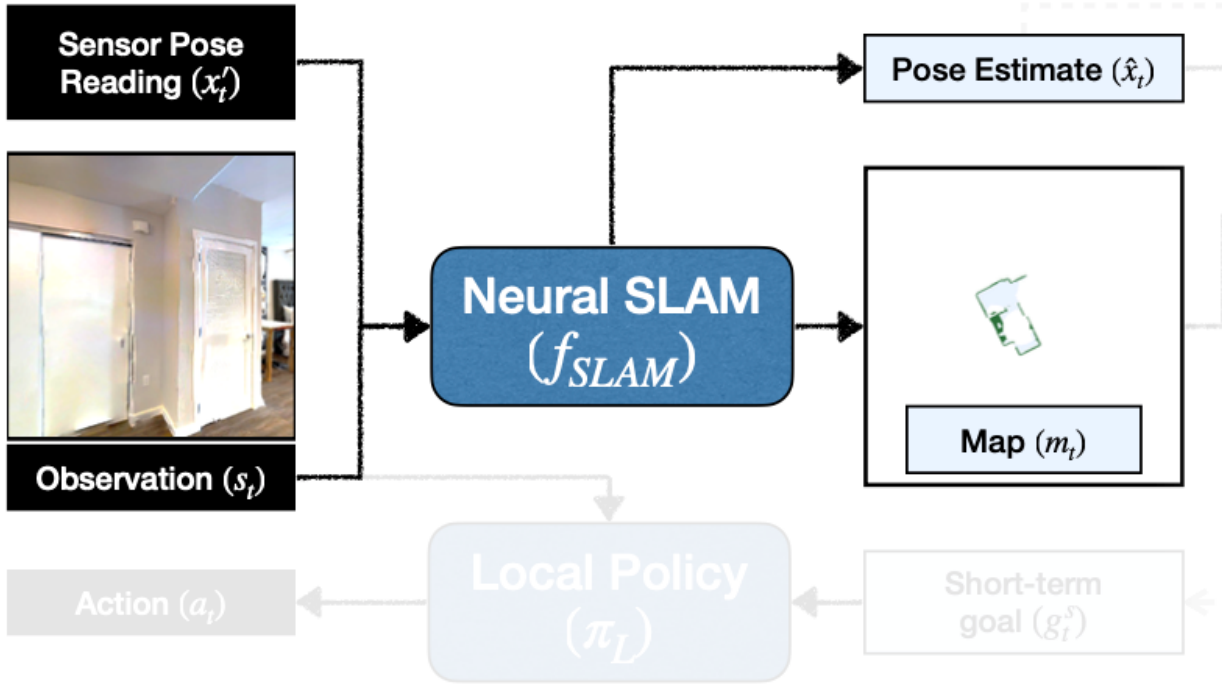
RLLAB
http://rllab.snu.ac.kr



Devendra Singh Chaplot, Dhiraj Gandhi, Saurabh Gupta, Abhinav Gupta, Ruslan Salakhutdinov, "**Learning To Explore Using Active Neural SLAM.**" *ICLR*, 2020.

Active Neural SLAM

RLLAB
http://rllab.snu.ac.kr

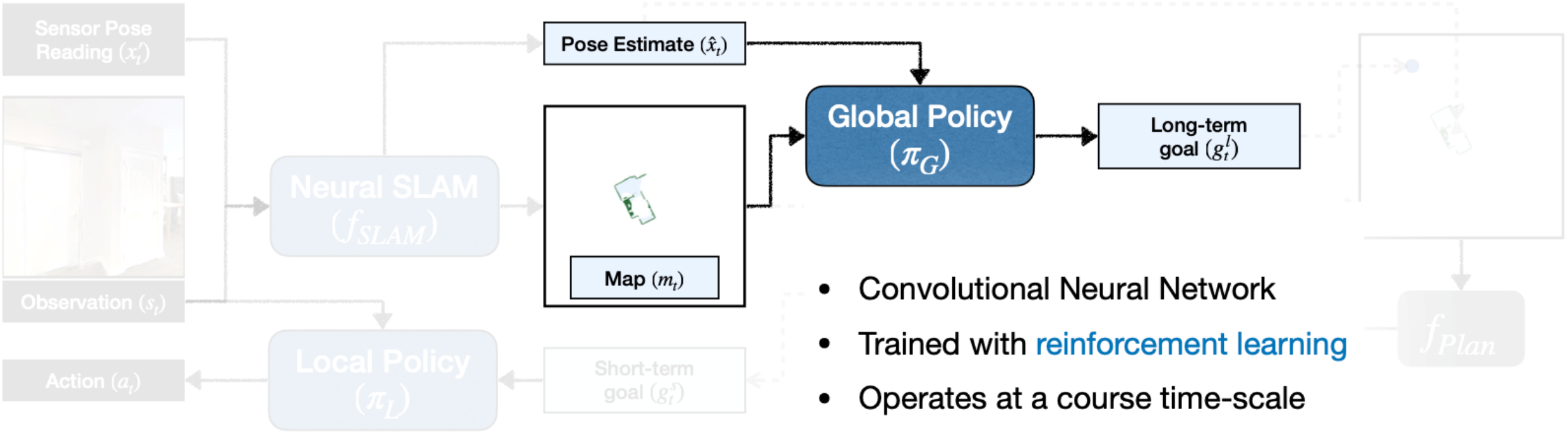


- Conv-Deconv Neural Network
- Trained with supervised learning
- Learns explicit structured map and pose representations

Devendra Singh Chaplot, Dhiraj Gandhi, Saurabh Gupta, Abhinav Gupta, Ruslan Salakhutdinov, "**Learning To Explore Using Active Neural SLAM.**" *ICLR*, 2020.

Active Neural SLAM

RLLAB
http://rllab.snu.ac.kr



- Convolutional Neural Network
- Trained with reinforcement learning
- Operates at a course time-scale

Devendra Singh Chaplot, Dhiraj Gandhi, Saurabh Gupta, Abhinav Gupta, Ruslan Salakhutdinov, "**Learning To Explore Using Active Neural SLAM.**" *ICLR*, 2020.

Active Neural SLAM

RLLAB
http://rllab.snu.ac.kr

Sensor Pose Reading (x_t')

- Convolutional Neural Network
- Trained with imitation learning
- Operates at a fine time-scale

Observation (s_t)

Action (a_t)

Local Policy (π_L)

Short-term goal (g_t^s)

Global Policy (π_G)

Long-term goal (g_t^l)

f_{Plan}

Devendra Singh Chaplot, Dhiraj Gandhi, Saurabh Gupta, Abhinav Gupta, Ruslan Salakhutdinov, "**Learning To Explore Using Active Neural SLAM.**" *ICLR*, 2020.

RLLAB
http://rllab.snu.ac.kr

Path Following

"Visual Memory for Robust Path Following," NeurIPS 2018

Path Following

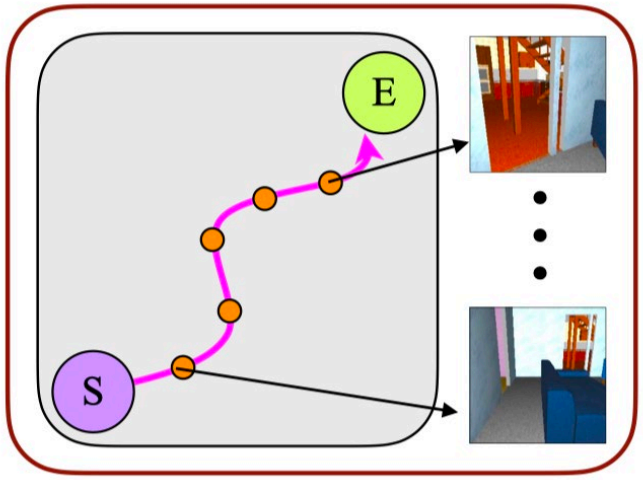
RLLAB
http://rllab.snu.ac.kr

- **Path following: based on human's ability to Retrace**
  - Consider the first morning of a conference in a city you have never been to. Rushing to the first talk, you might follow your phone's direction through a series of twists and turns to reach the venue.
  - When you return later in the day, you can **retrace** your steps to the conference venue relatively robustly, remembering to take a left turn at the bistro and keep straight past the coffee shop.

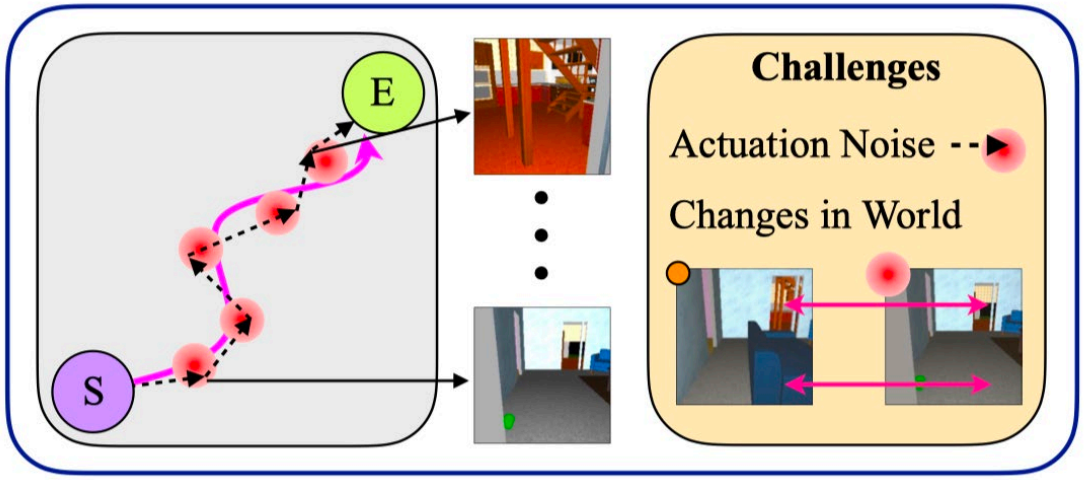
Path Following

RLLAB
http://rllab.snu.ac.kr

- **How to solve?**
  - One classical approach is to build a full 3D model of the world via SLAM.
  - However, for the task of navigation, this might be an overkill.
  - A large number of learning-based approaches have sprung up.
  - The proposed approach only rely on a **single demonstration** in a new environment.



Demonstration



Execution under noisy actions and changes in the world.

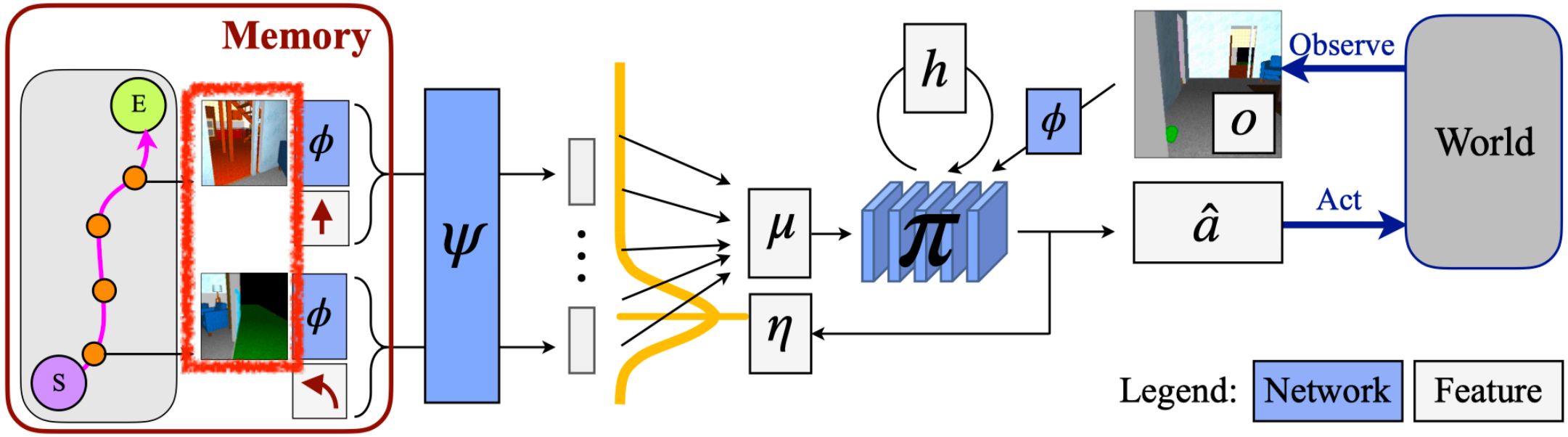
Path Following

RLLAB
http://rllab.snu.ac.kr

- "Visual Memory for Robust Path Following," NeurIPS 2018
  - Given a demonstration of a path, a **first network** generates a **path abstraction**.
  - Equipped with this **abstraction**, a **second network** observes the worlds and decides how to **act to retrace the path** under noisy actuation and a changing environment.
  - Both **networks** are optimized end-to-end at training time.

Path Following

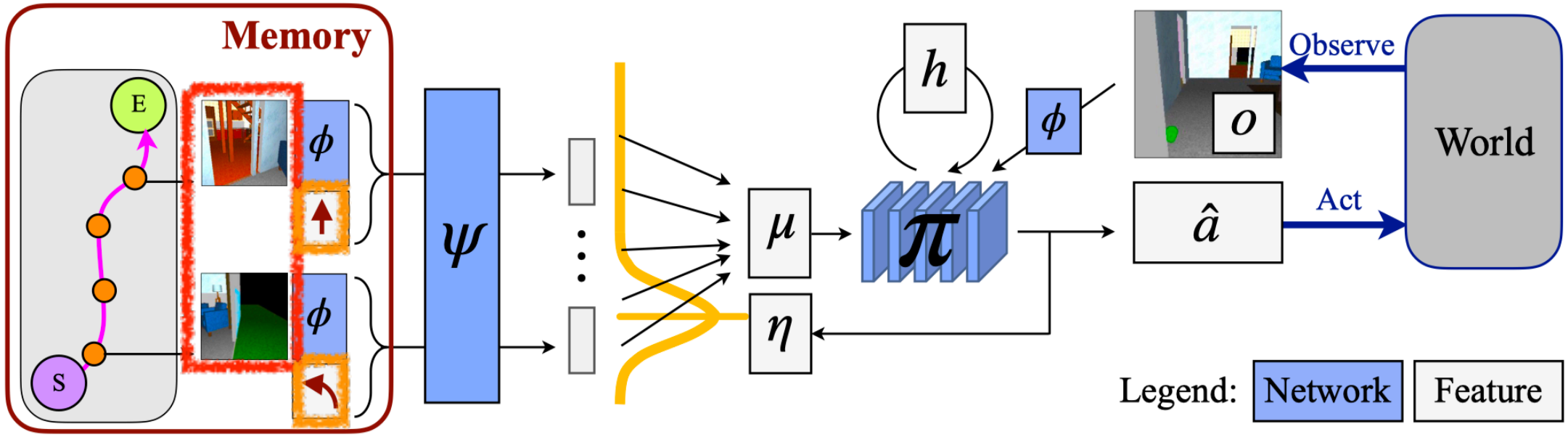
RLLAB
http://rllab.snu.ac.kr



- Given a **sequence of images**

Path Following

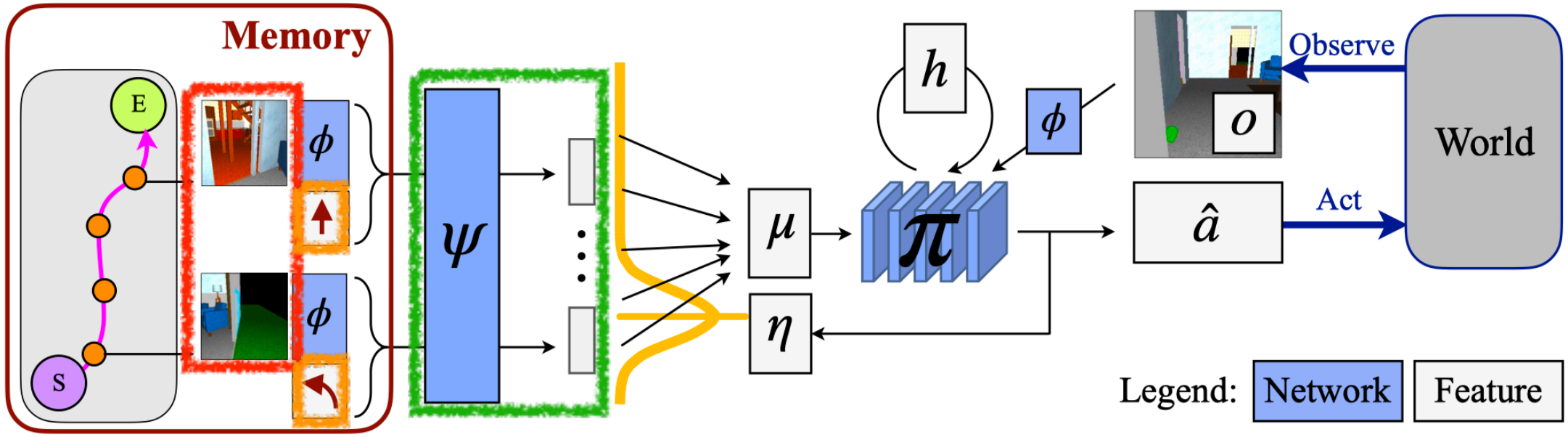
RLLAB
http://rllab.snu.ac.kr



- Given a **sequence of images** and **actions** at these images

Path Following

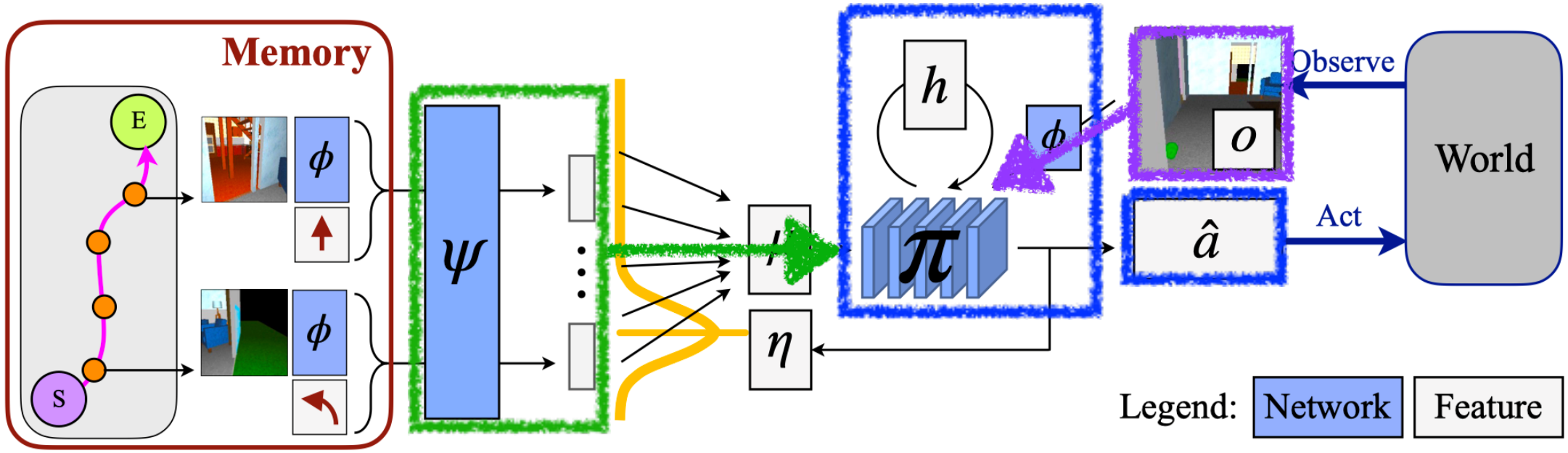
RLLAB
http://rllab.snu.ac.kr



- Given a **sequence of images** and **actions** at these images, it abstracts the s equence into a **sequence of memories**.

Path Following

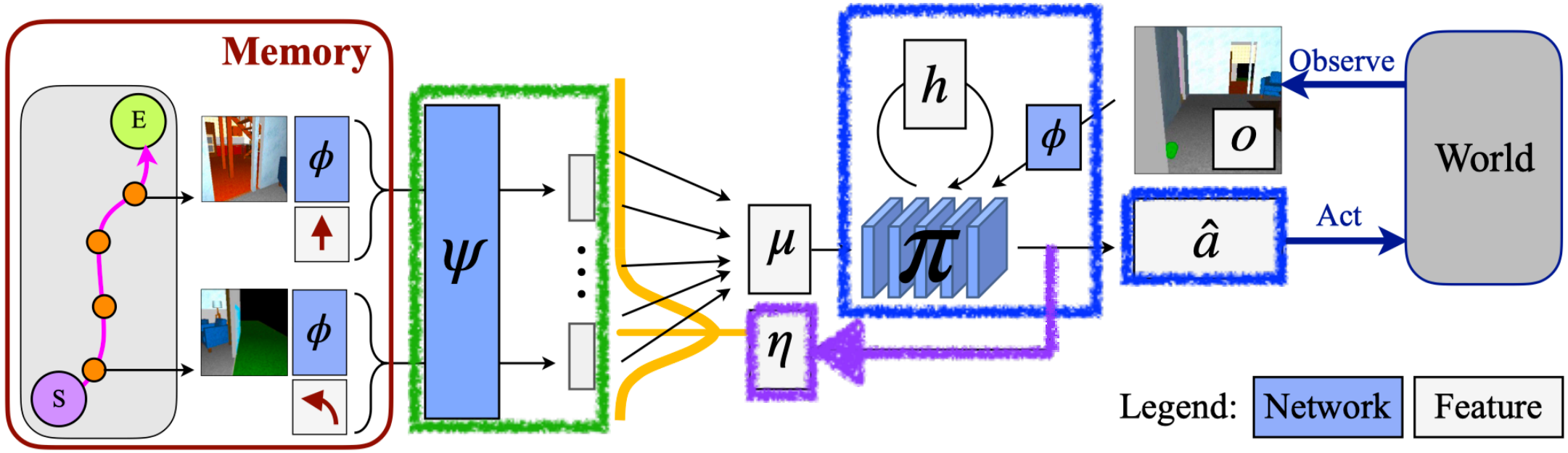
RLLAB
http://rllab.snu.ac.kr



- A second recurrent network π uses **this sequence** and the current **observation** to emit **actions** that **retrace the path**.

Path Following

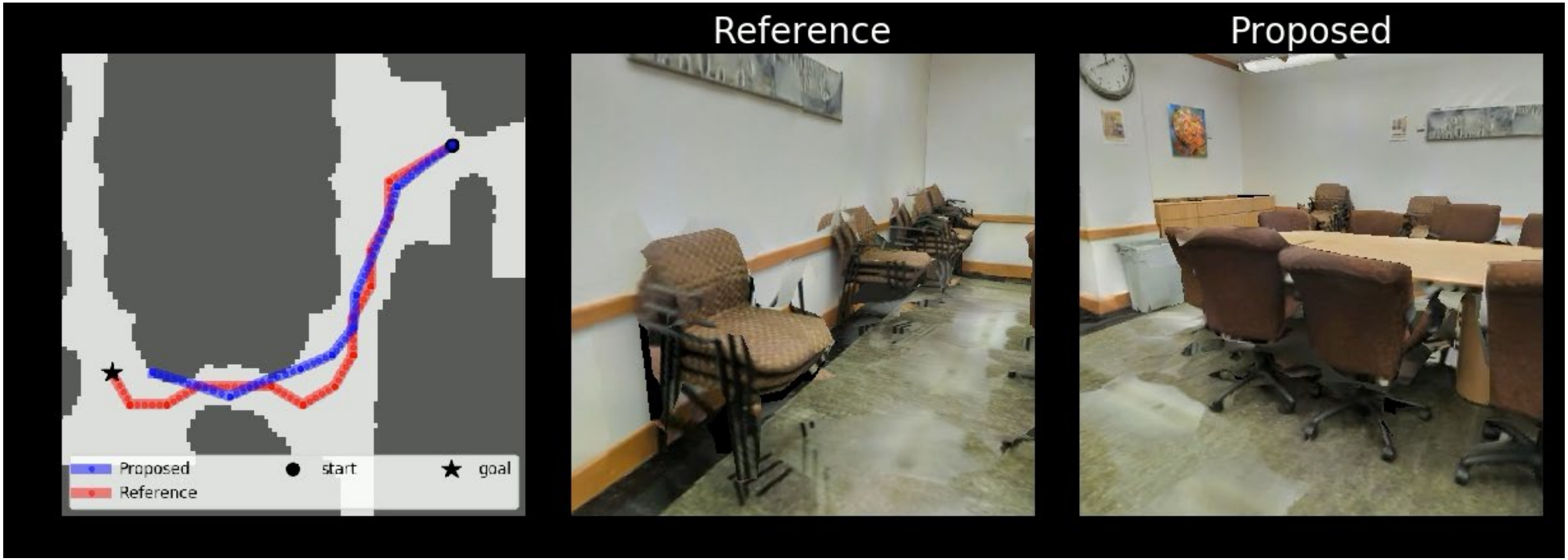
RLLAB
http://rllab.snu.ac.kr



- A second recurrent network π uses **this sequence** to emit **actions** that **retr ace the path**. It also updates the **attention location** η.

Path Following

RLLAB
http://rllab.snu.ac.kr



RLLAB
http://rllab.snu.ac.kr

Active Vision

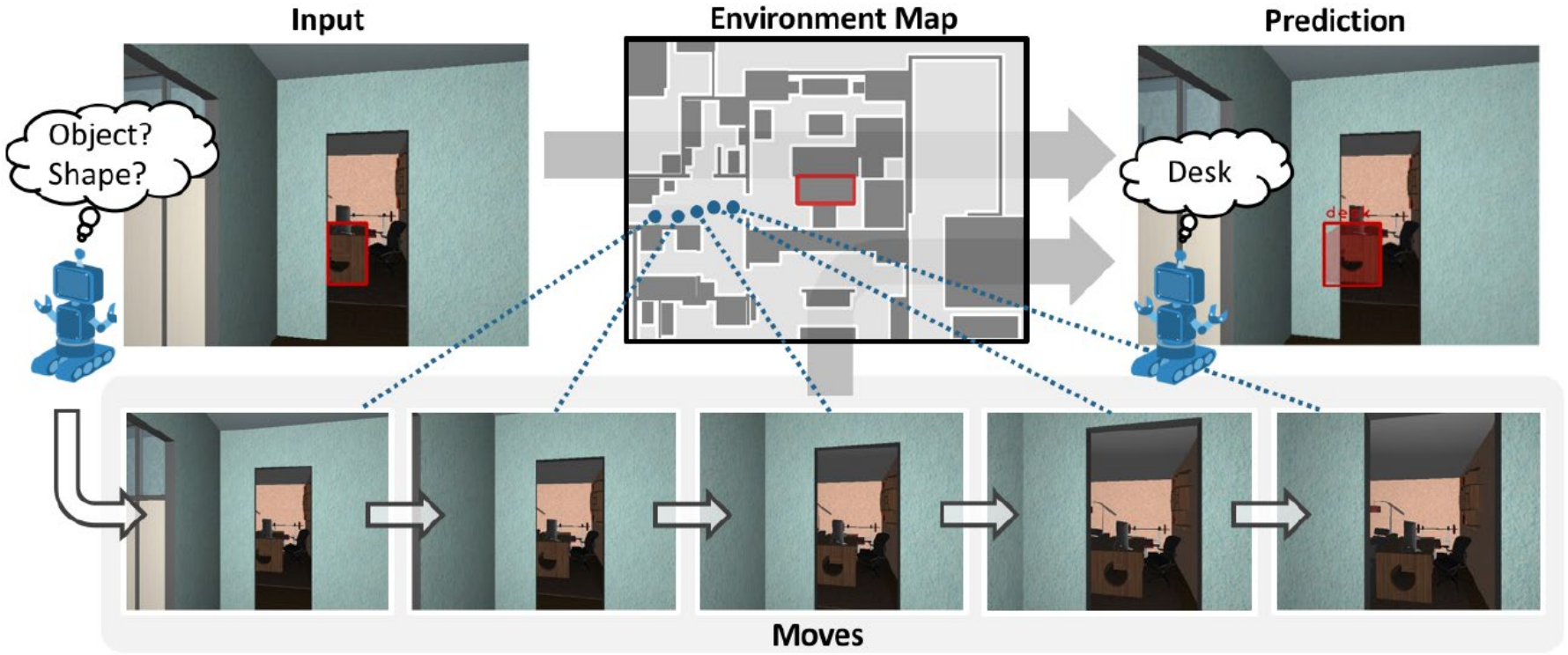
Embodied Visual Recognition, ICCV 2019

SEAL: Self-supervised Embodied Active Learning using Exploration and 3D Consistency, NeurIPS 2021

Embodied Visual Recognition

RLLAB
http://rllab.snu.ac.kr

- Humans have the ability to derive **strategical moves** to gather more information from new viewpoints to further help the visual recognition.
- Toddlers (4-7 months old) are capable of **actively diverting viewpoints** to learn about objects. [1]

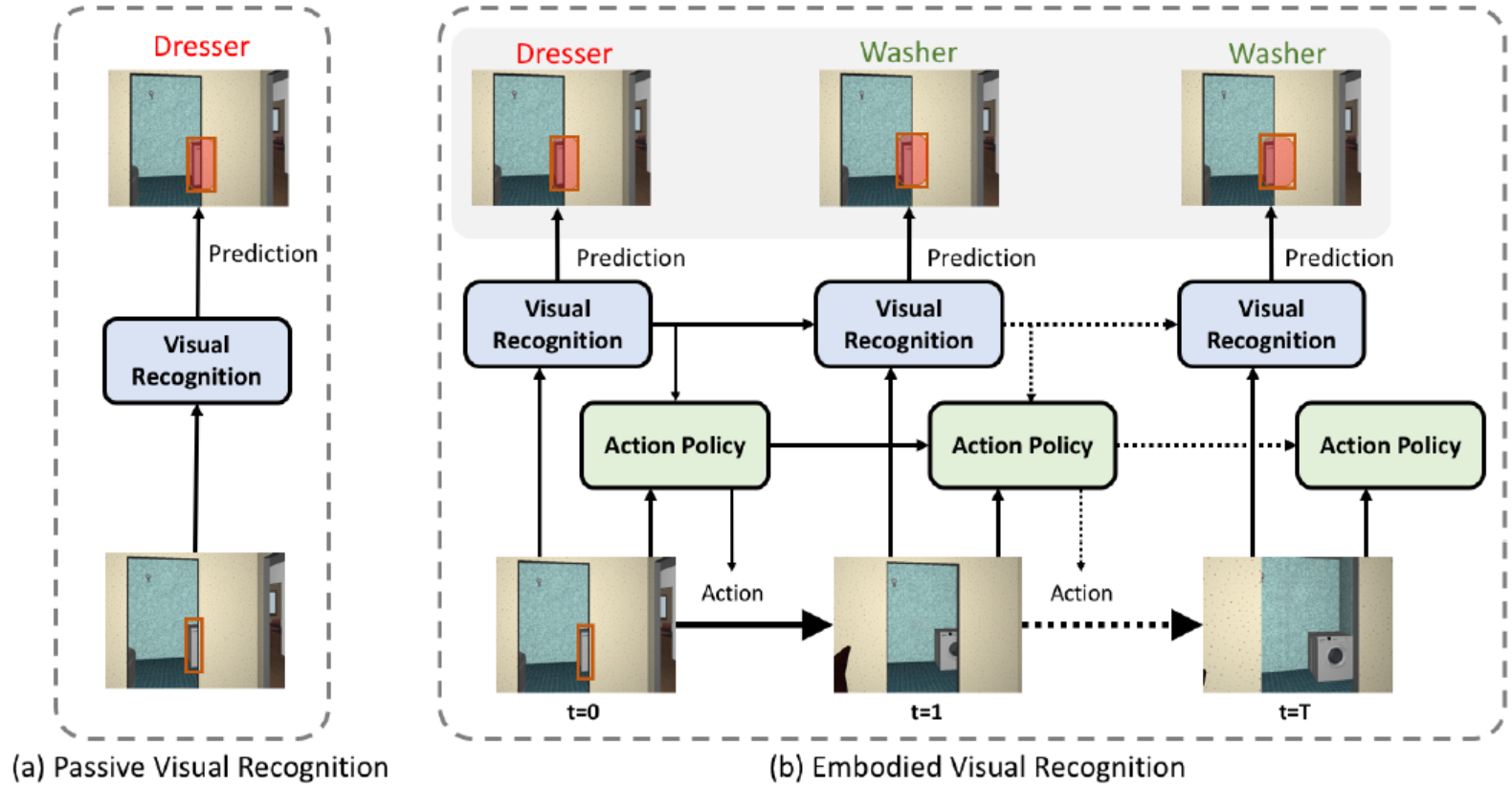


[1] S. Bambach, D. J. Crandall, L. B. Smith, and C. Yu. Toddler inspired visual object learning. In Advances in Neural Information Processing Systems (NIPS), 2018.

Embodied Visual Recognition

RLLAB
http://rllab.snu.ac.kr

- The robot is initialized close to the object (location, gaze direction).
- To perform visual recognition on the occluded object, the **agent learns to move**, rather than standing still and hallucinating.



(a) Passive Visual Recognition

(b) Embodied Visual Recognition

Embodied Visual Recognition

RLLAB
http://rllab.snu.ac.kr

- Based on House3D: a simulator built on top of SUNCG.
  - Filter out atypical 3D rooms in House3D, resulting in 550 houses in total.
  - 640 x 800 images, extend borders of rendered images by 80 pixels on each side (800 x 960 images)
  - Select a subset of object categories: 8 categories out of 80

	bed	chair	desk	dresser	fridge	sofa	table	washer	total
Train	1687	1009	1333	737	900	1742	981	551	8940
Val	197	122	207	82	103	206	144	52	1113
Test	427	210	330	172	207	456	264	104	2170

Embodied Visual Recognition

RLLAB
http://rllab.snu.ac.kr

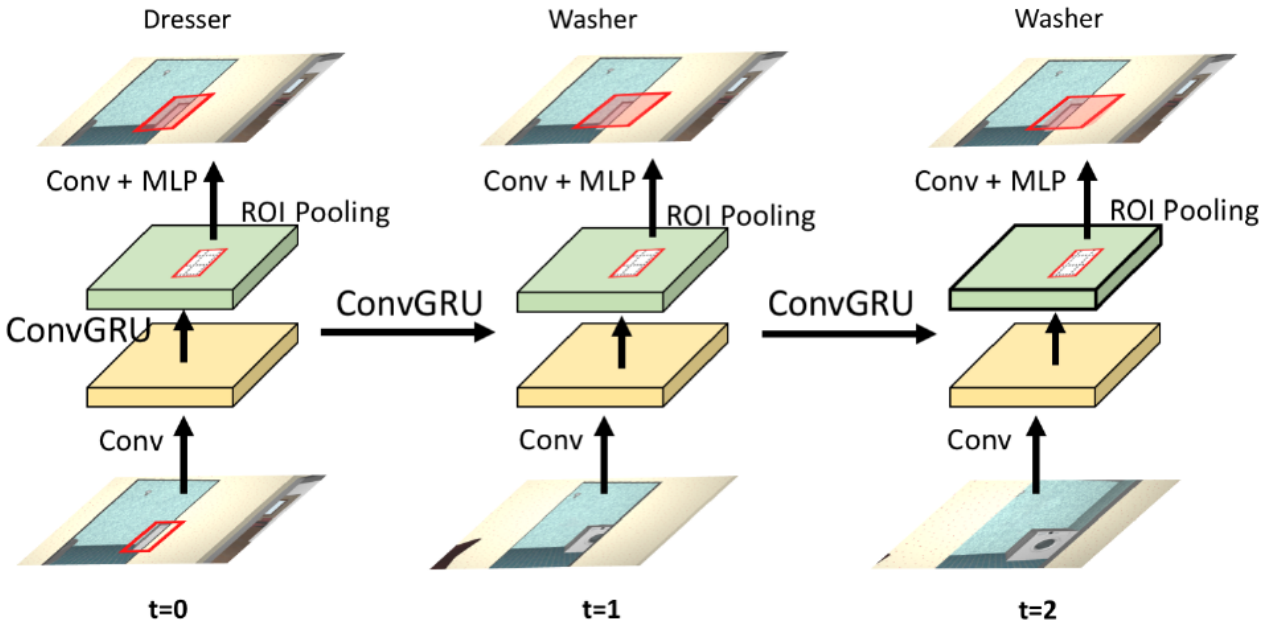
- The visual recognition network $y_t = f(b_o, I_o, I_1, \ldots, I_t)$ consists of three components: f_{base}, f_{fuse}, f_{head}.
  - Firstly, f_{base} uses a CNN to extract feature map $y_t = f(b_o, I_o, I_1, \ldots, I_t)$.

$$x_t = f_{base}(I_t)$$

  - Secondly, f_{fuse} uses ConvGRU to aggregate all the feature map up to t,

$$\hat{x}_t = f_{fuse}(x_0, \ldots, x_t).$$

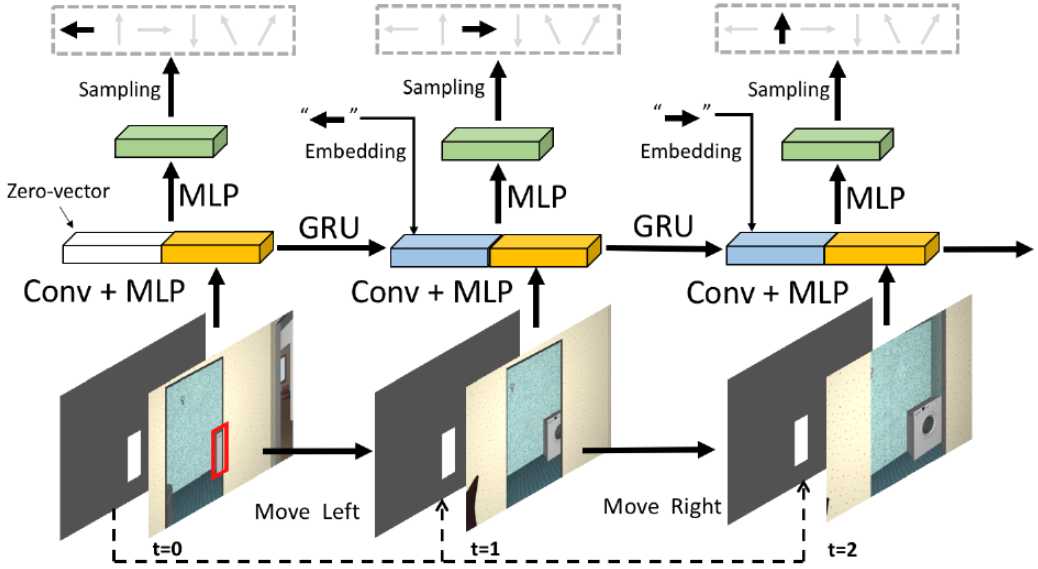
  - Finally, features and the initial bounding box are used to predict **object category**, **bounding box** and **mask**, $y_t = f_{head}(b_o, \hat{x}_t)$.



Embodied Visual Recognition

RLLAB
http://rllab.snu.ac.kr

- The policy network $a_t \sim \pi(b_o, I_0, I_1, \dots, I_t)$ consists of three components: $\{\pi_{imgEnc}, \pi_{actEnc}, \pi_{act}\}$.
  - Firstly, π_{imgEnc} is a conv-bn-relu-pool encoder for image features. The input concatenates the initial input, current input, and the mask for visible bounding box of the target object, $z_t^{img} = \pi_{imgEnc}([I_b, I_o, I_t])$
  - Secondly, π_{actEnc} encodes the last action in each step t, $z_t^{img} = \pi_{actEnc}(a_{t-1})$.
  - Finally, π_{act} is a single-layer GRU, which takes action embedding and image embedding to output the final action value, $a_t \sim \pi_{act}(\left[z_t^{img}, z_t^{act}\right])$.



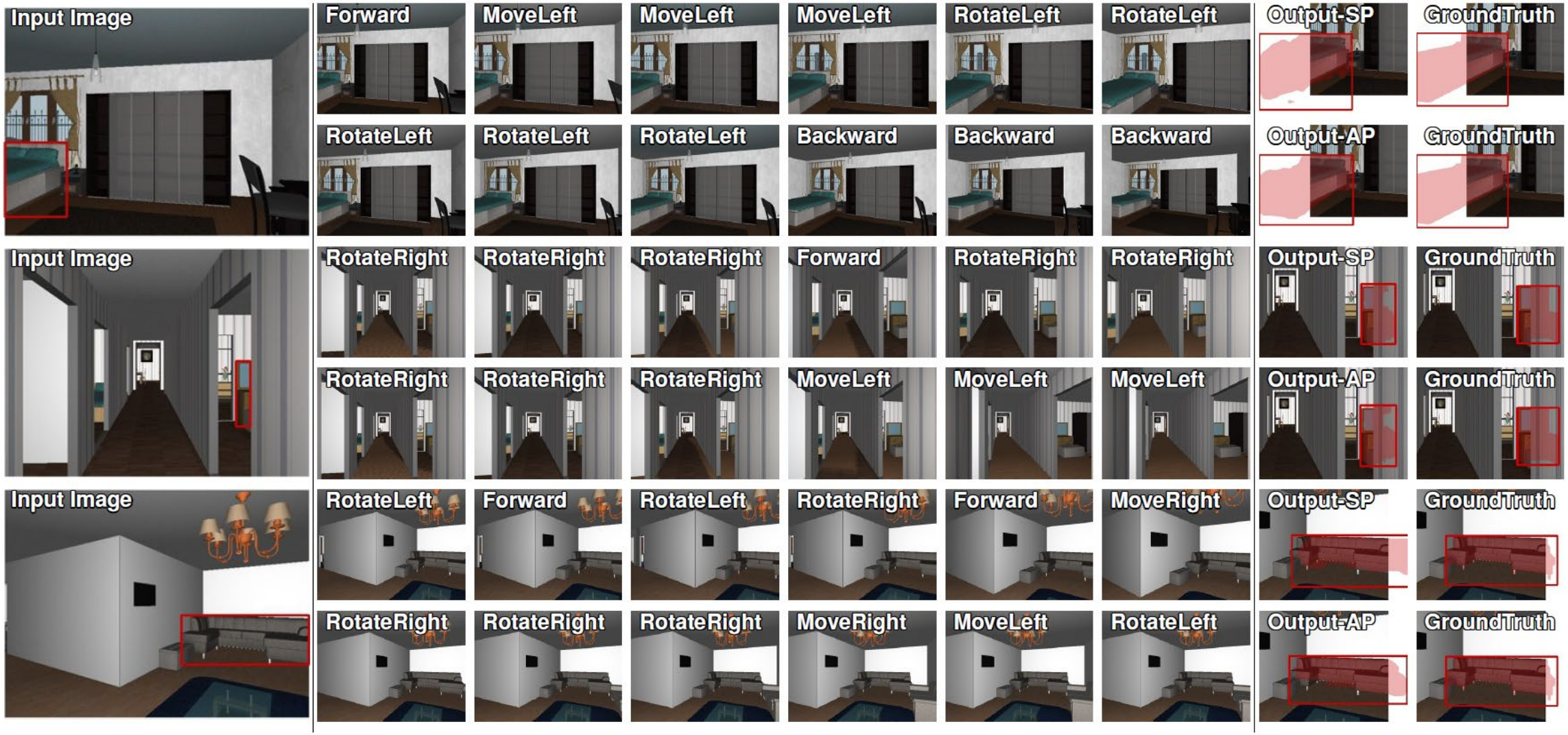
Rewards: classification accuracy, IOU to measure advantage of candidate agent moves.

$$r_t = \lambda_a Acc_t^c + \lambda_b IoU_t^b + \lambda_c IoU_t^m.$$
$$R_t = r_t - r_{t-1}.$$

Used **policy gradient** with REINFORCE to train the policy model.

Embodied Visual Recognition

RLLAB
http://rllab.snu.ac.kr



SEAL

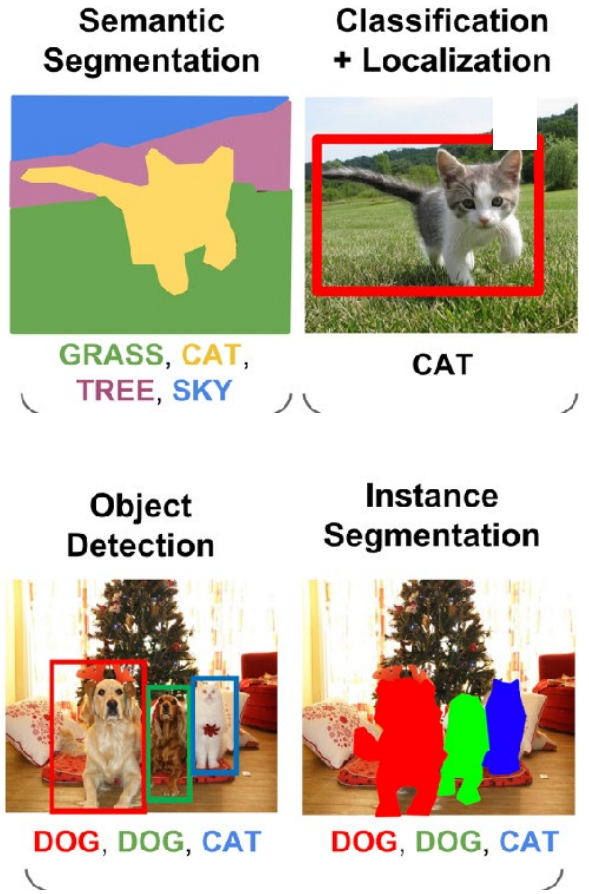
RLLAB
http://rllab.snu.ac.kr

- “SEAL: Self-supervised Embodied Active Learning using Exploration and 3D Consistency”, NeurIPS 2021
  - The agent can actively choose the views it experiences to maximize ***perception performance***.
  - Perceptual models allow the agent to **act in the world and collect data** that improve the perception models.
  - Improved perception models **can improve the agent’s policy** for interacting with the world.

SEAL

RLLAB
http://rllab.snu.ac.kr

- Internet Computer Vision



1 Karpathy. https://cs.stanford.edu/people/karpathy/cnnembed/

2 Li, Johnson, Yeung. http://cs231n.stanford.edu/slides/2017/cs231n_2017_lecture11.pdf

SEAL

RLLAB
http://rllab.snu.ac.kr

- Embodied agent

SEAL

RLLAB
http://rllab.snu.ac.kr

- Internet vs Embodied Data

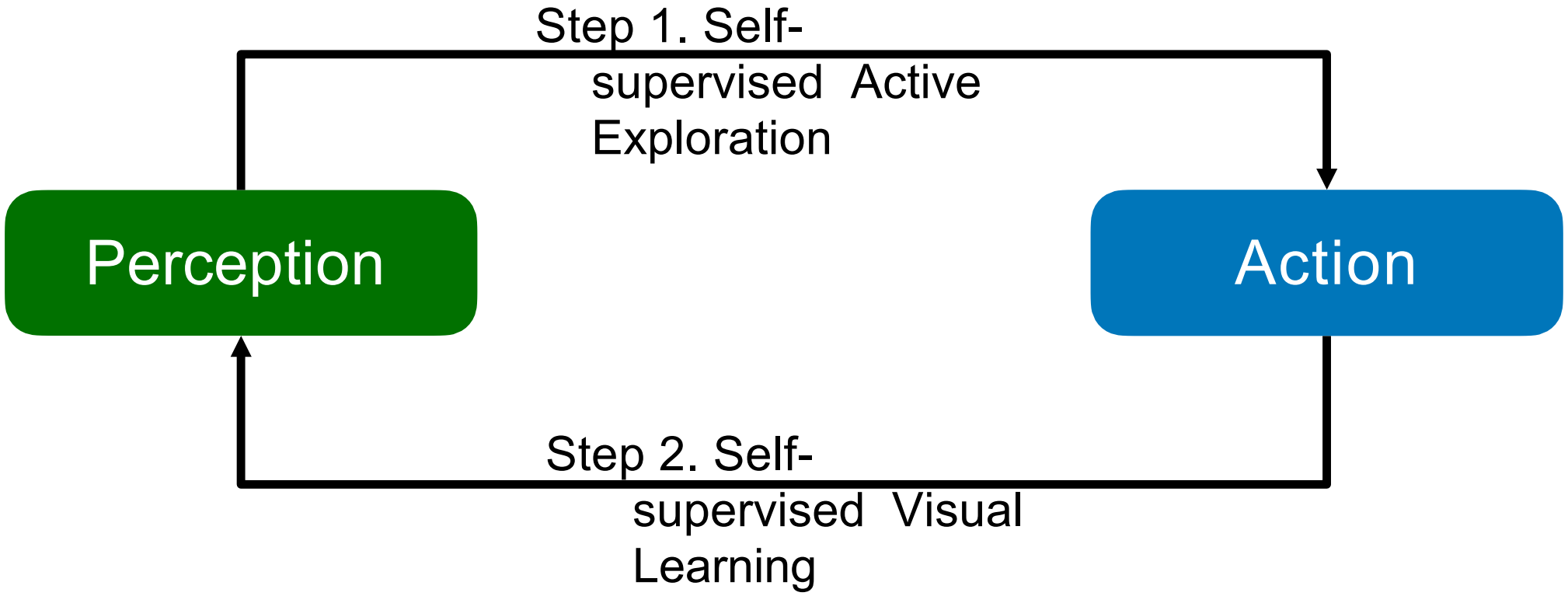
Static Internet data

Active Embodied data

SEAL

RLLAB
http://rllab.snu.ac.kr

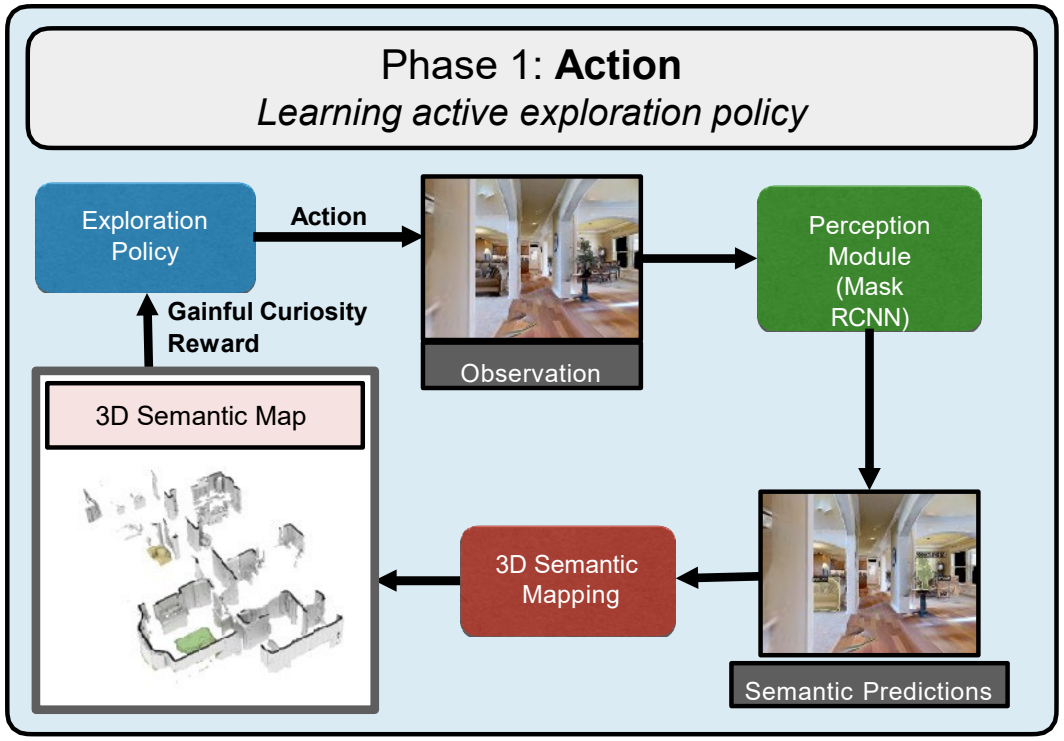
- Perception-Action Loop

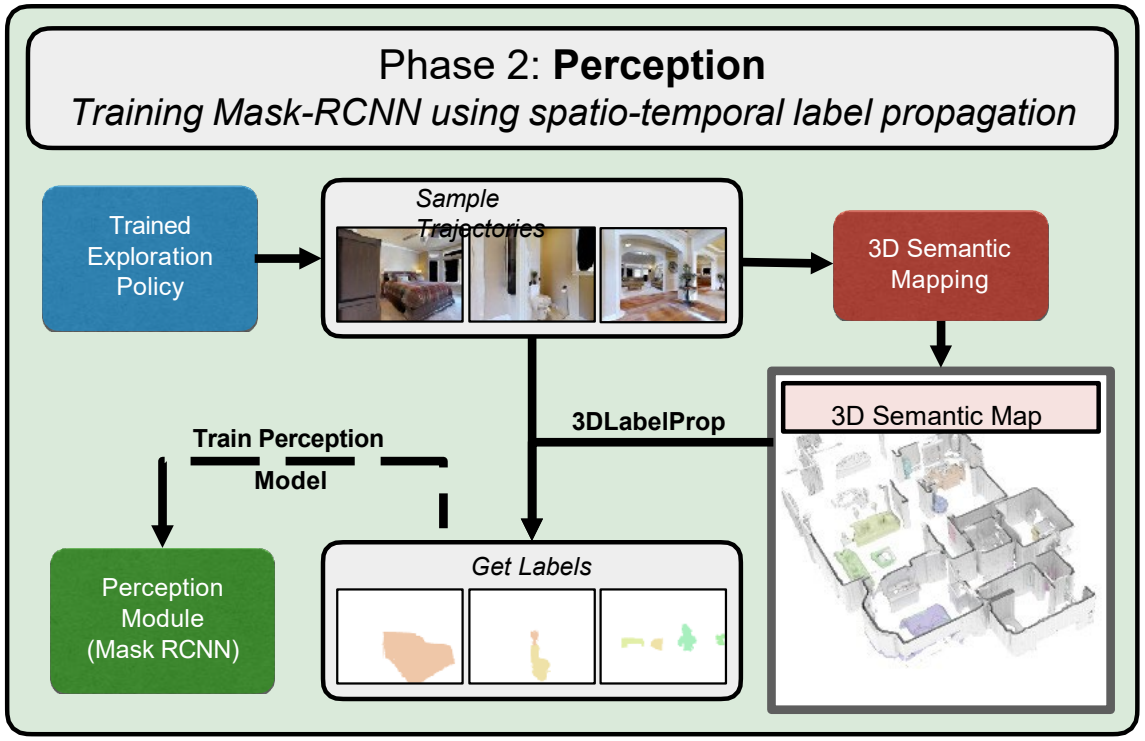


We must perceive in order to move, but we must also move in order to perceive
- Gibson (1979)

SEAL

RLLAB
http://rllab.snu.ac.kr

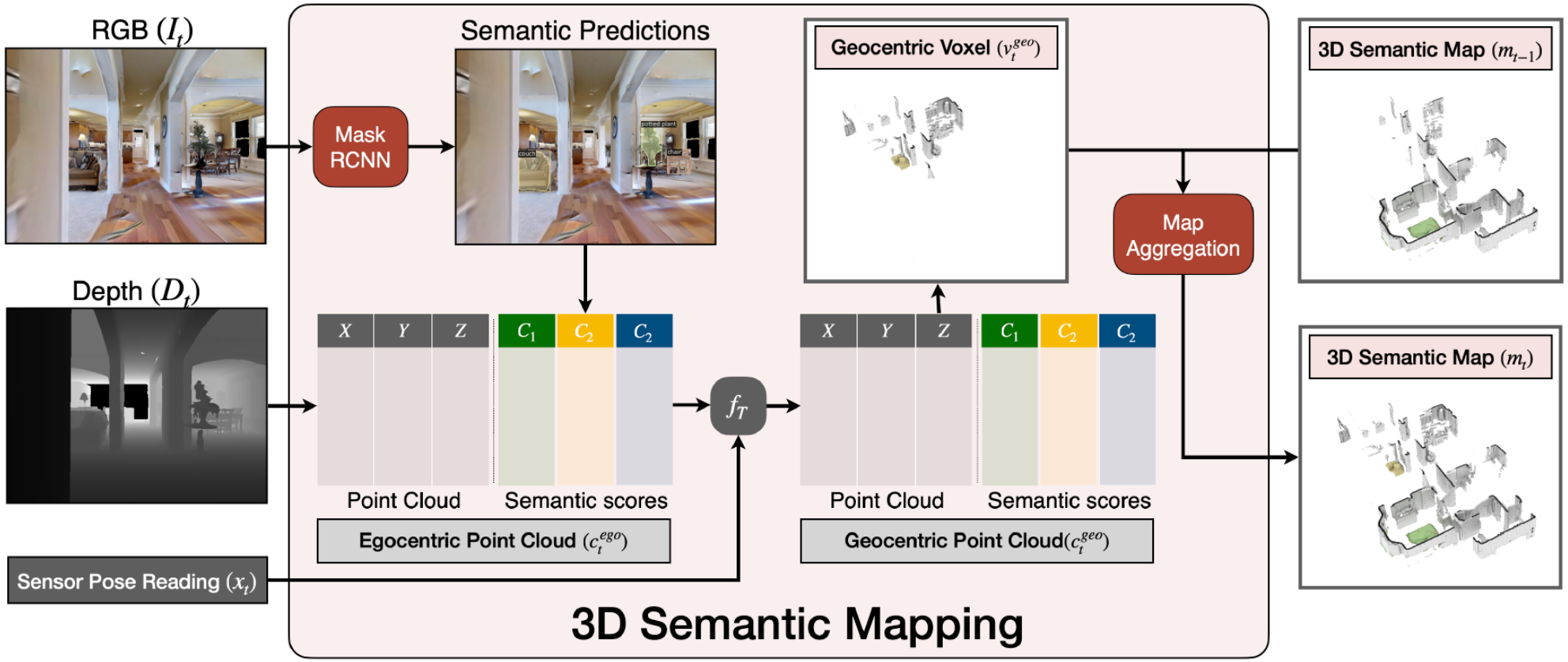




SEAL

RLLAB
http://rllab.snu.ac.kr

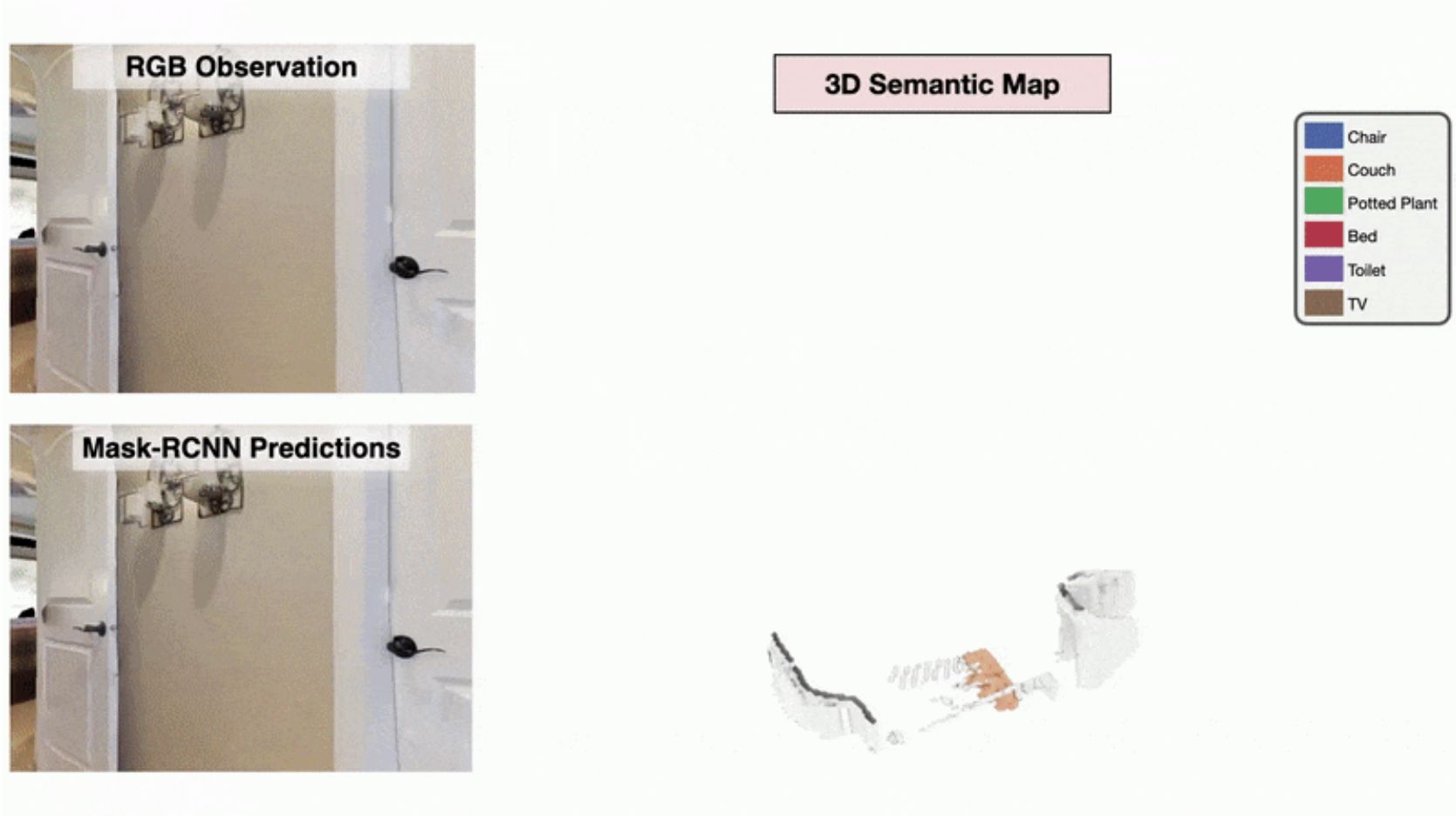
- 3D Semantic Mapping



SEAL

RLLAB
http://rllab.snu.ac.kr

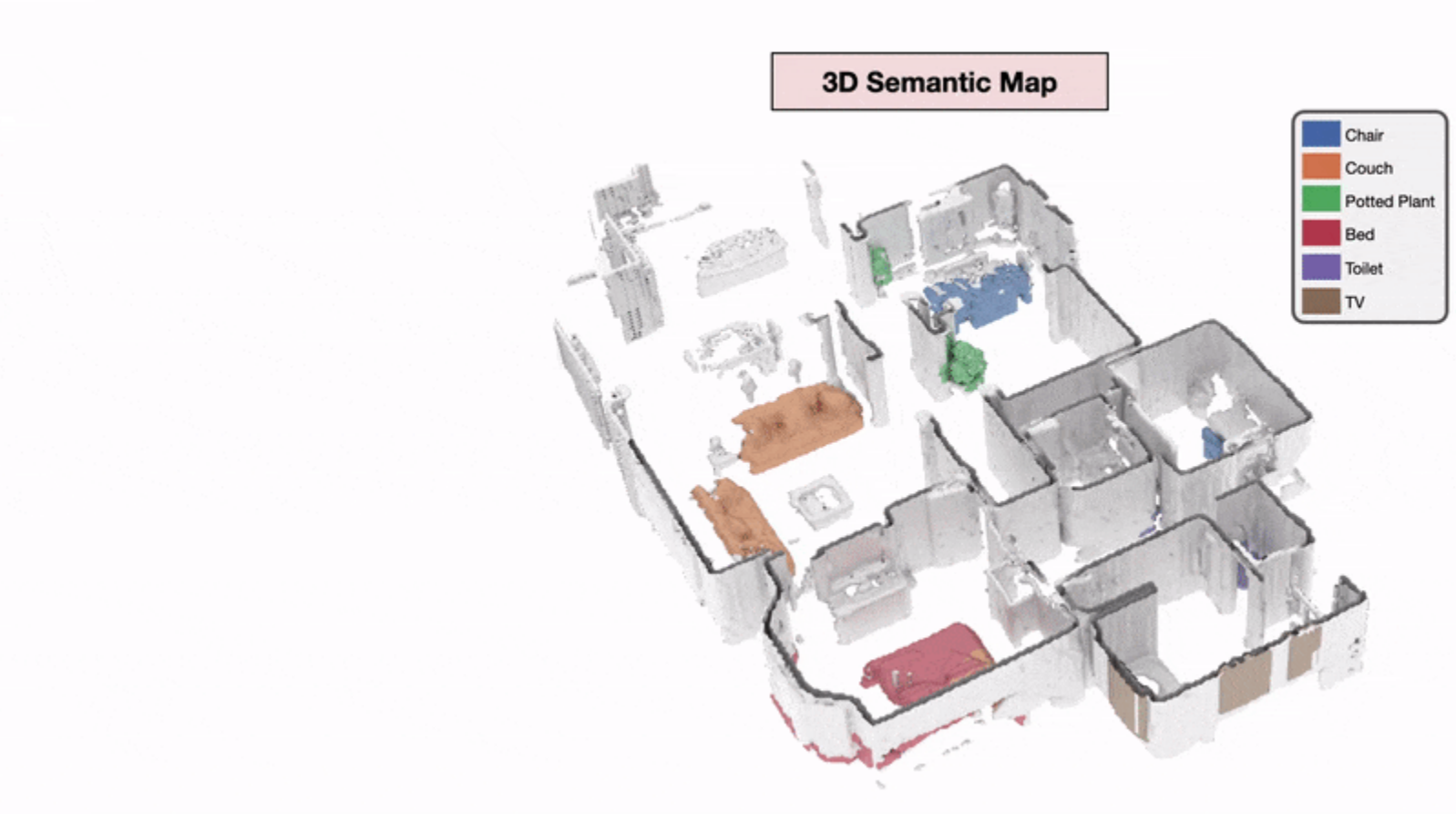
- 3D Semantic Mapping



SEAL

RLLAB
http://rllab.snu.ac.kr

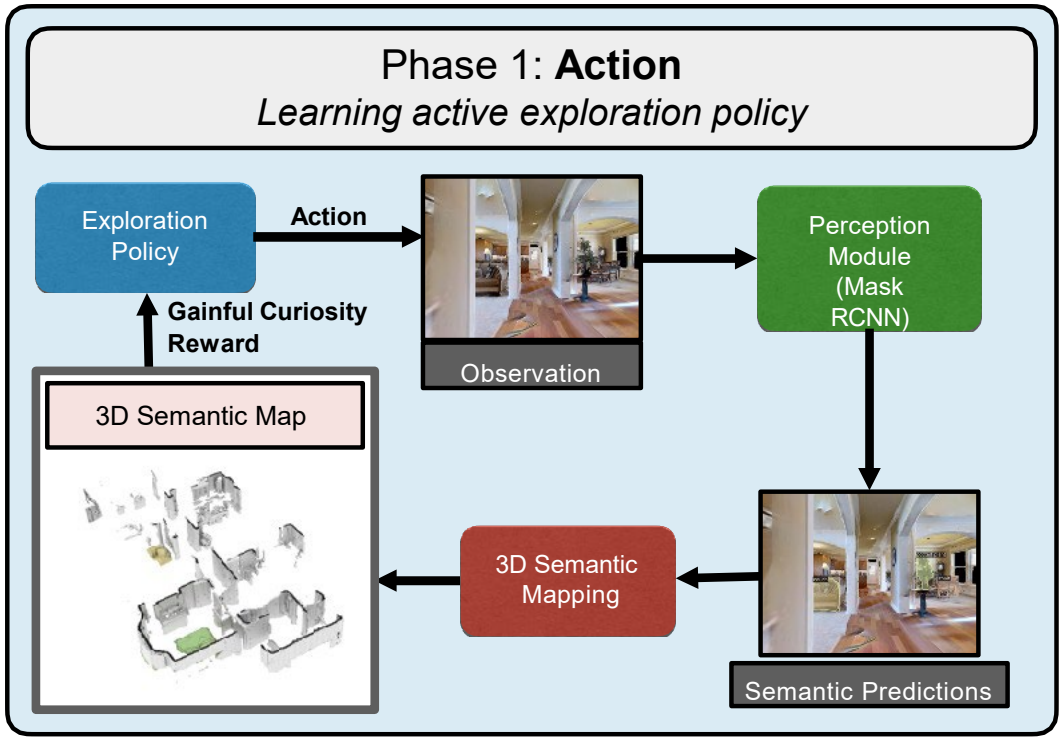
- Phase 1: Learning Action

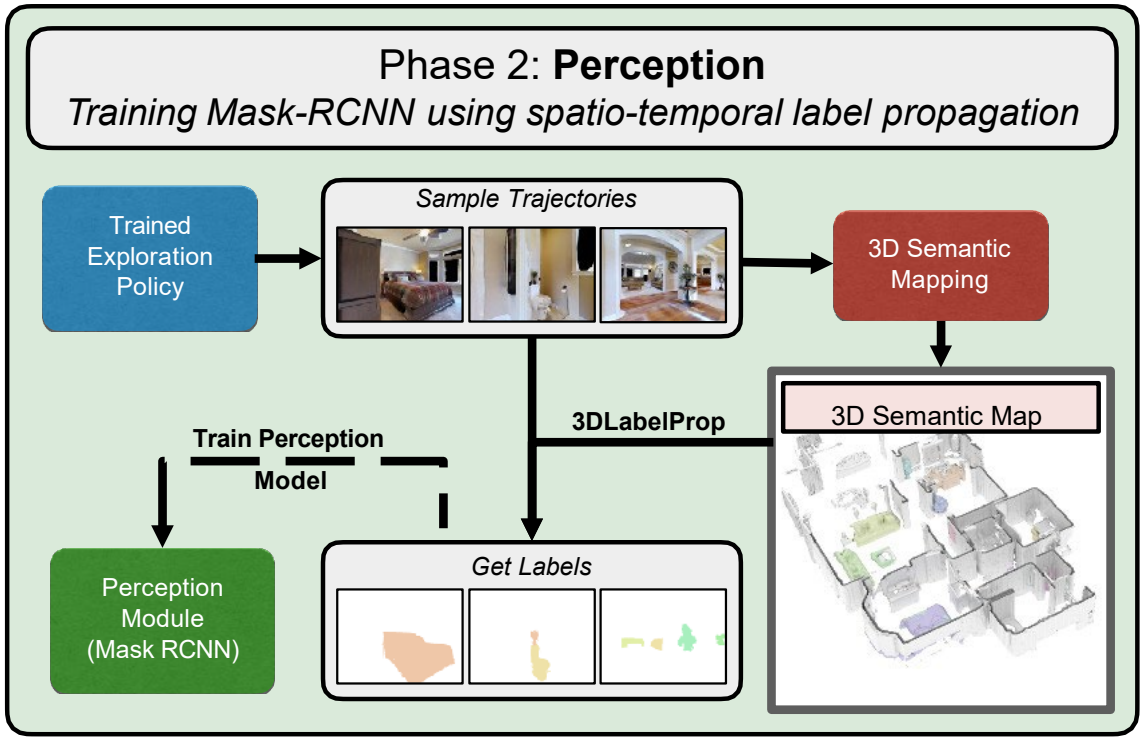


SEAL

RLLAB
http://rllab.snu.ac.kr

- Phase 2: Learning Perception

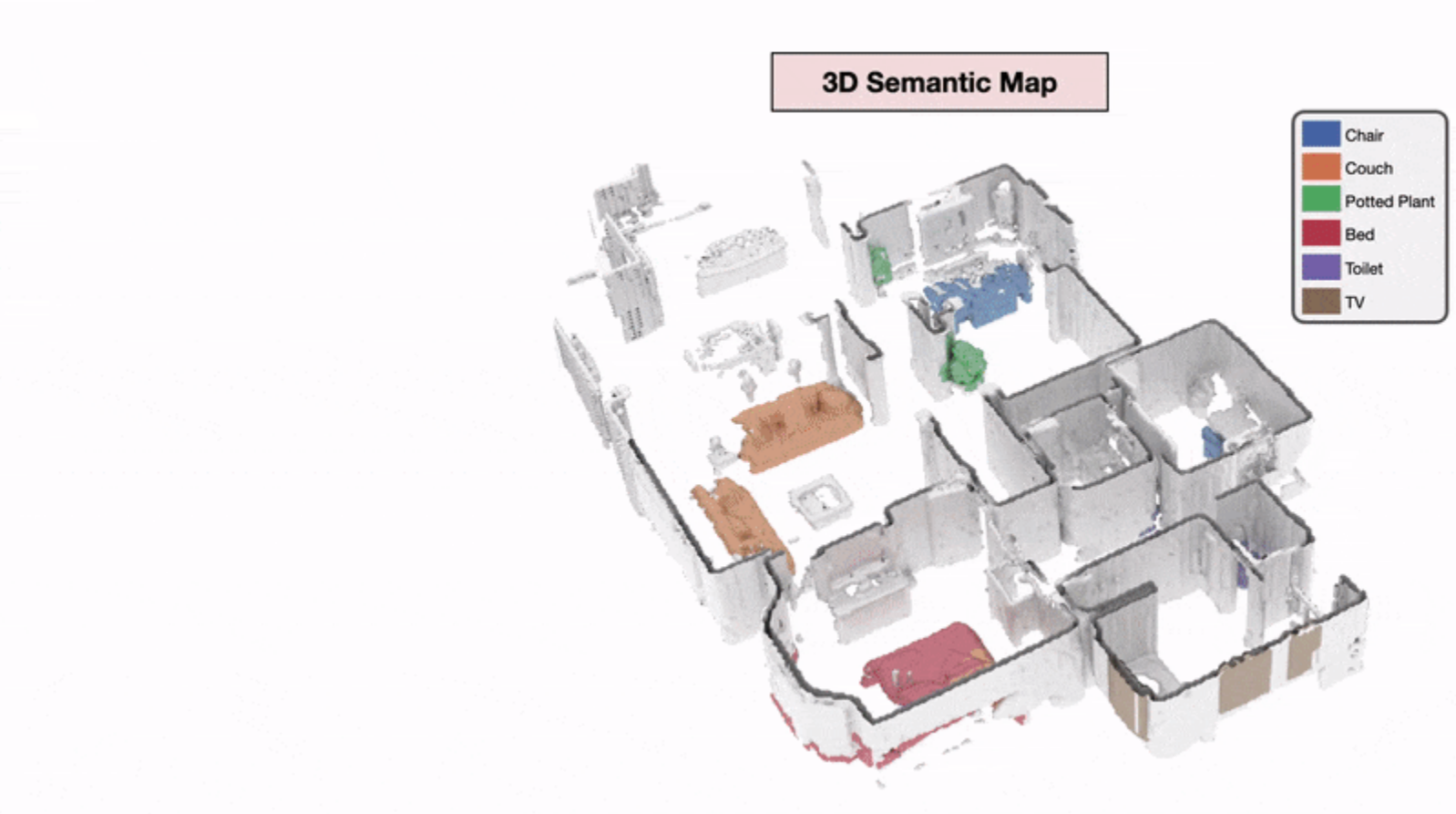




SEAL

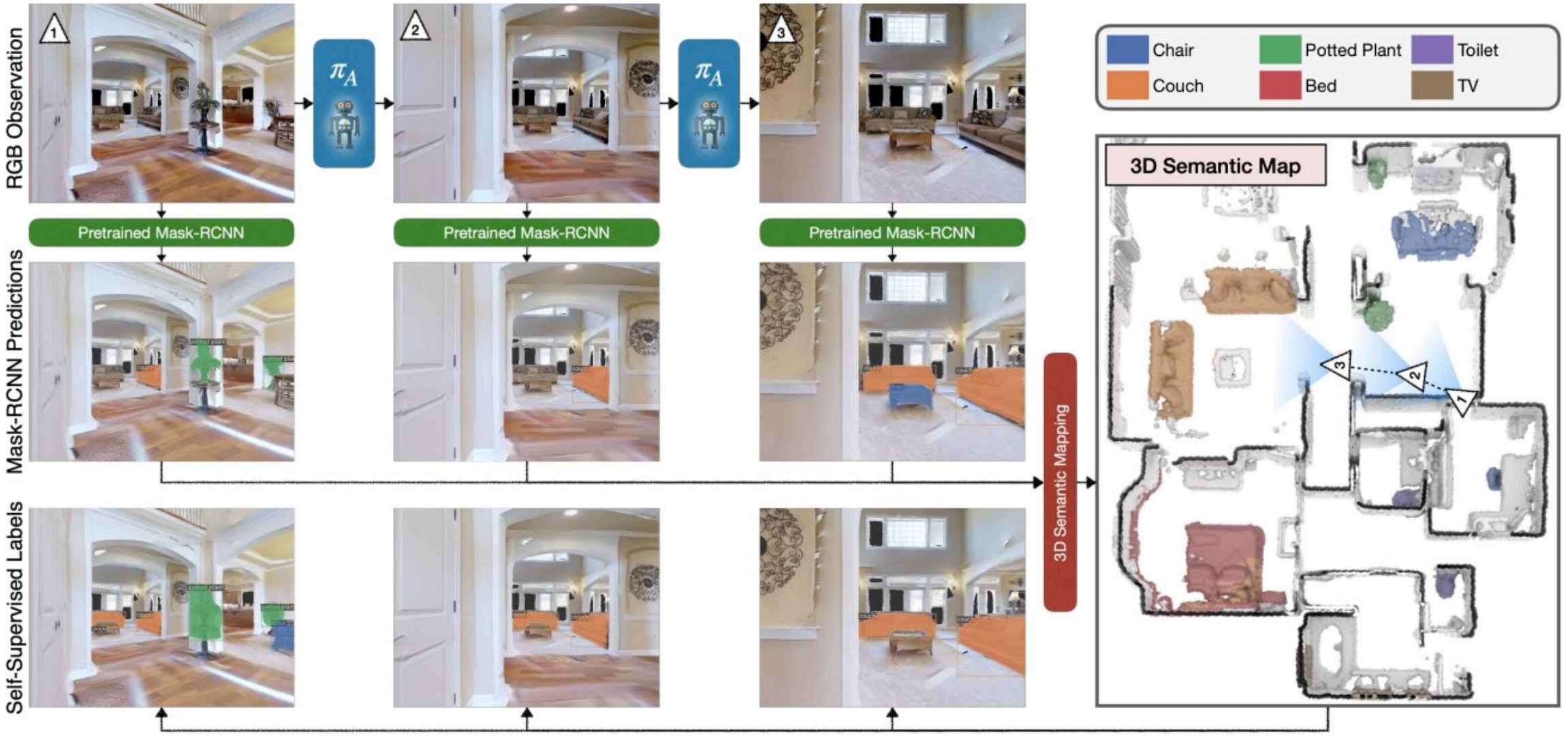
RLLAB
http://rllab.snu.ac.kr

- Phase 2: Learning Perception



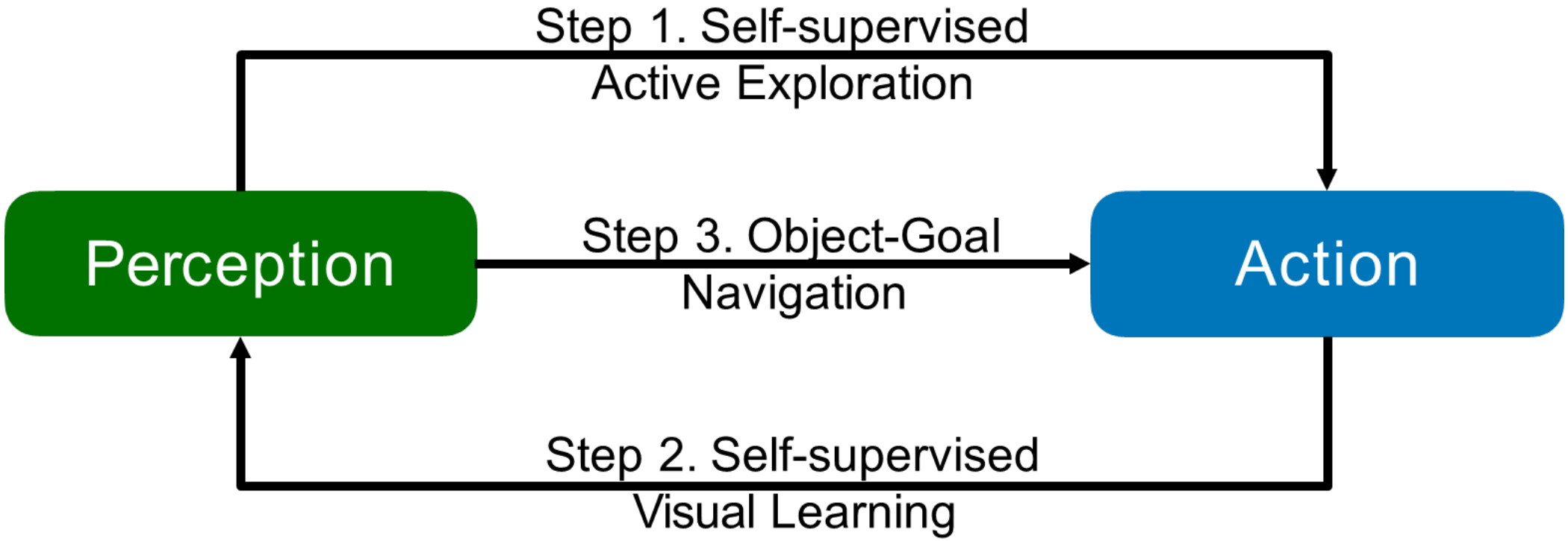
SEAL

RLLAB
http://rllab.snu.ac.kr



SEAL

RLLAB
http://rllab.snu.ac.kr

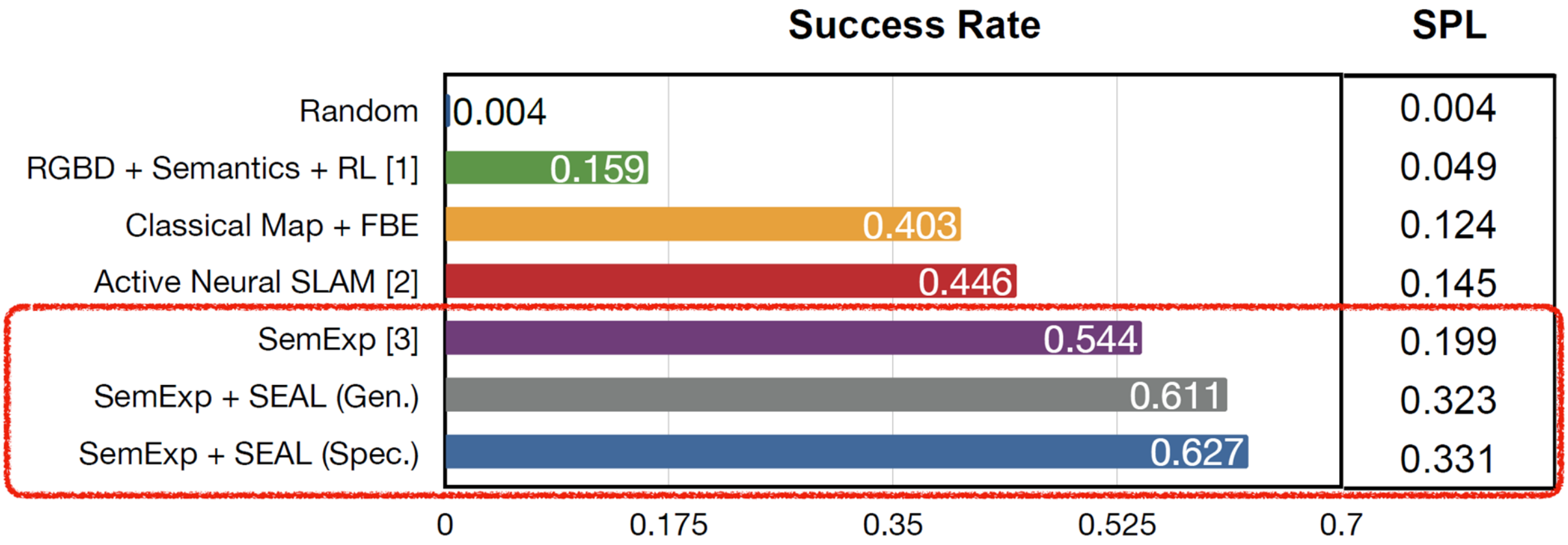


We must perceive in order to move, but we must also move in order to perceive
- Gibson (1979)

SEAL

RLLAB
http://rllab.snu.ac.kr

- Results of Object Goal Navigation



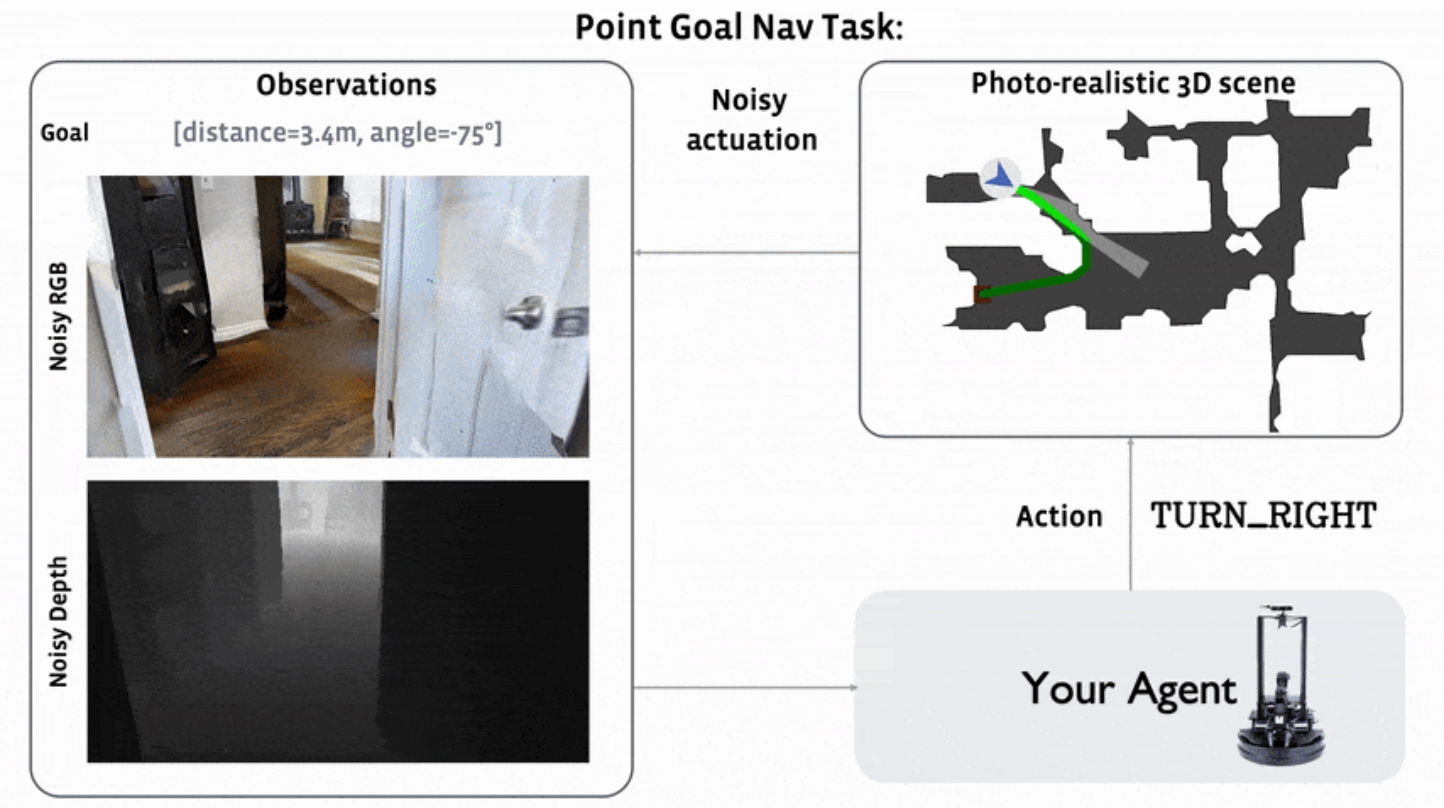
RLLAB
http://rllab.snu.ac.kr

Point Goal Navigation

Point Goal Navigation

RLLAB
http://rllab.snu.ac.kr

- In PointNav, an agent is spawned at a random starting position and orientation in an unseen environment and asked to navigate to **target coordinates specified relative to the agent's start location** (*'Go 5m north, 3m west relative to start'*). No ground-truth map is available and the agent must **only use its sensory input** (an RGB-D camera) to navigate.



RLLAB
http://rllab.snu.ac.kr

Image Goal Navigation

"Semi-Parametric Topological Memory for Navigation," 2018

Image Goal Navigation

RLLAB
http://rllab.snu.ac.kr

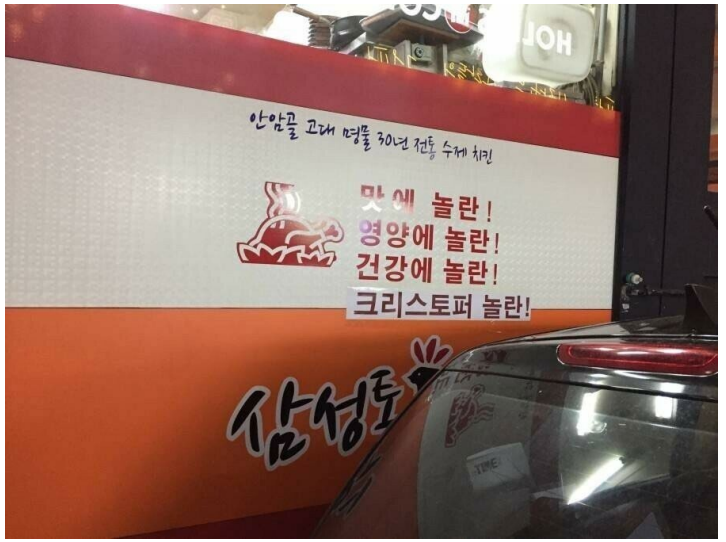


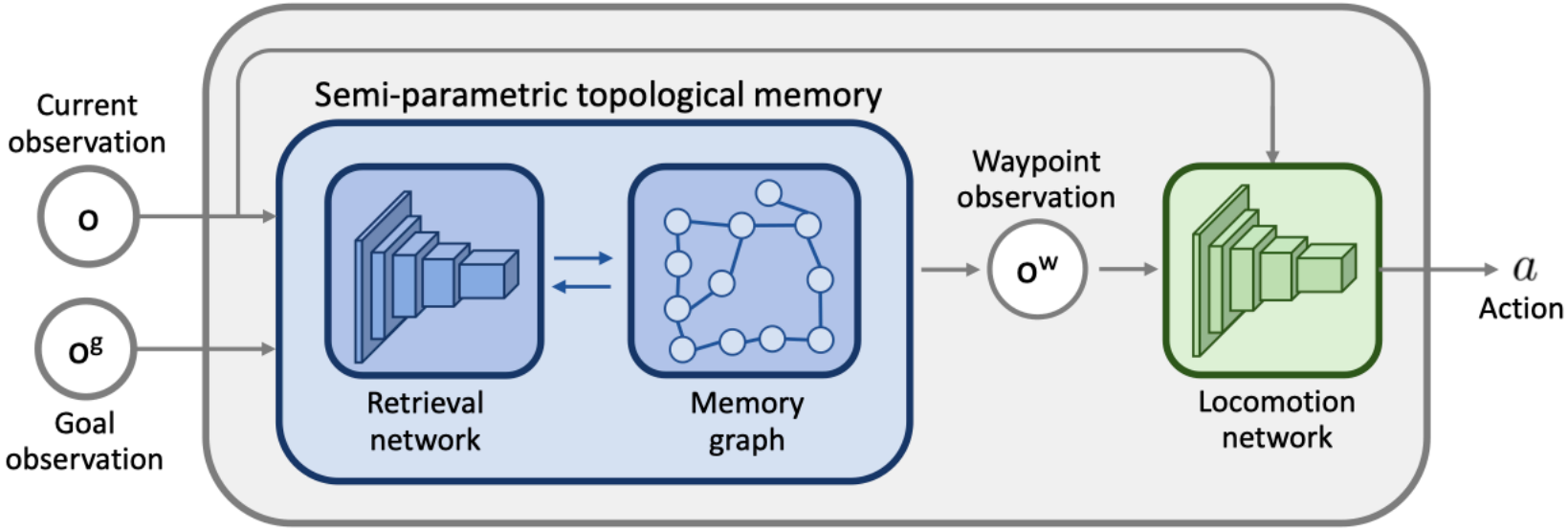
Image Goal Navigation

RLLAB
http://rllab.snu.ac.kr

- "Semi-Parametric Topological Memory for Navigation," 2018
  - It introduces a **memory architecture** for navigation inspired by **landmark-based navigation** in animals.
  - Semi-parametric topological memory (SPTM) consists a **(non-parametric) graph** with nodes corresponding to locations in the environments and a **(parametric) deep network** capable of retrieving nodes from the graph based on observations.
  - The SPTM is used as a **planning module** in a navigation system.
  - It is a two-staged method:
    - **exploration**: records the traversal of the environment and build the internal representation.
    - **goal-directed navigation**: use the internal representation to reach the goal location.

Image Goal Navigation

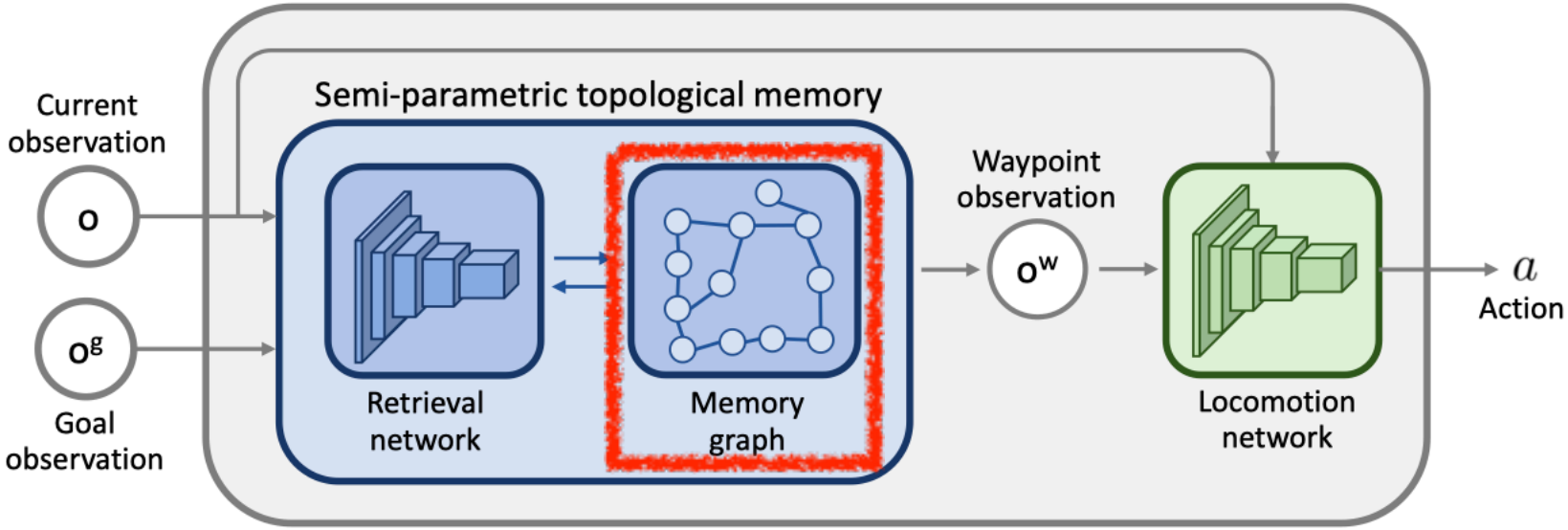
RLLAB
http://rllab.snu.ac.kr



- At each time step t, the agent gets an observation o_t and takes an action a_t.
- The interaction is set up in two stages: **exploration** and **goal-directed navigation**.
- In the above figure, SPTM acts as a **planning module**: given the current and goal observations, it generates a waypoint and the corresponding action.

Image Goal Navigation

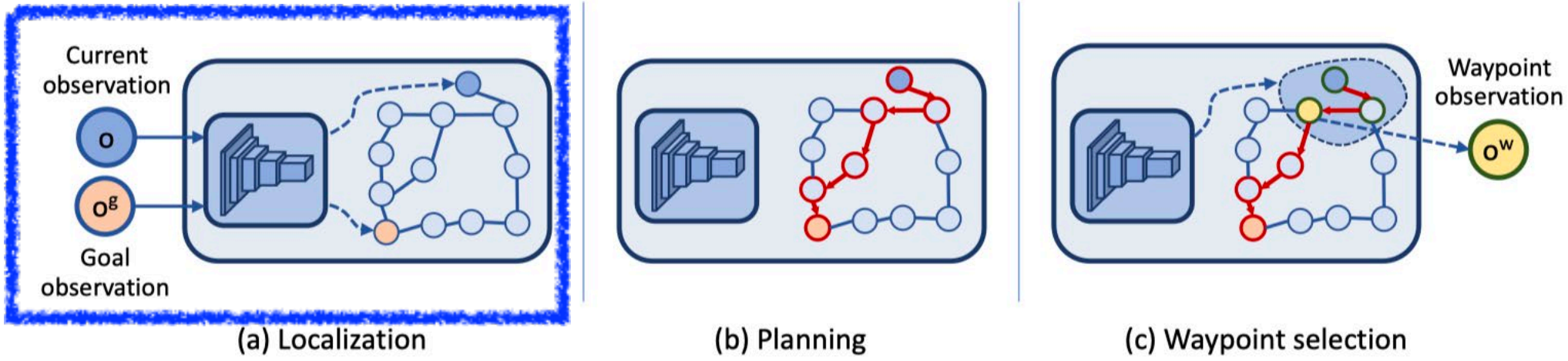
RLLAB
http://rllab.snu.ac.kr



- **Memory graph**
  - The graph is populated based on an **exploration sequence** provided to the agent.
  - Two vertices are connected by an edge in one of two cases: if they correspond to consecutive time steps, or if the observations are very close, as judged by the retrieval network .
  - The network R estimates the similarity of two observations (o_1, o_2) trained on a set of environments in self-supervised manner.

Image Goal Navigation

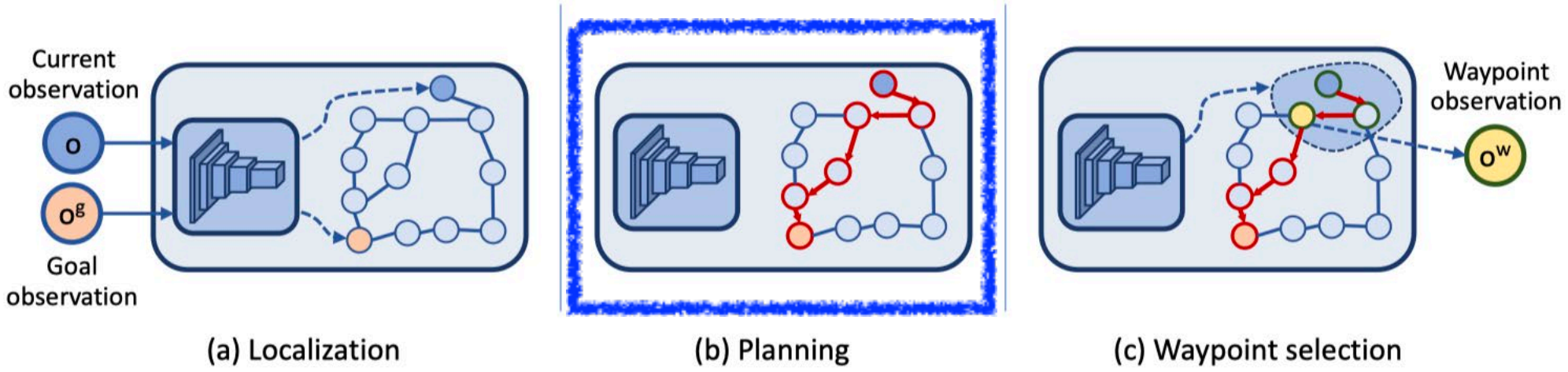
RLLAB
http://rllab.snu.ac.kr



- The retrieval network R **localizes** in the graph the vertices v^a and v^g ,corresponding to the c urrent agent's observation o and the goal observation o^g ,respectively.

Image Goal Navigation

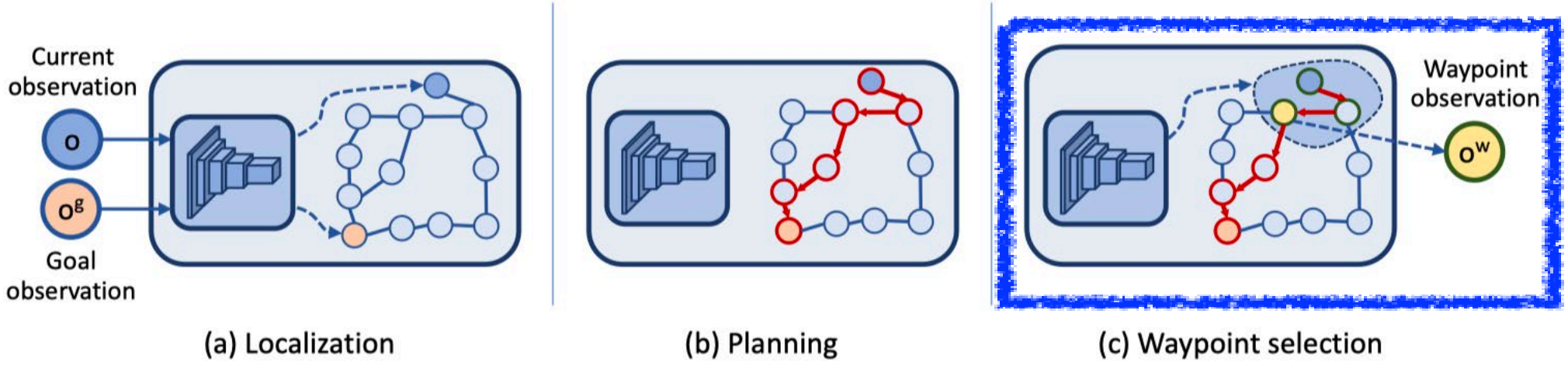
RLLAB
http://rllab.snu.ac.kr



- The shortest path on the graph between these vertices is computed (**red arrows**).
- Dijkstra's algorithm is used in the experiments.

Image Goal Navigation

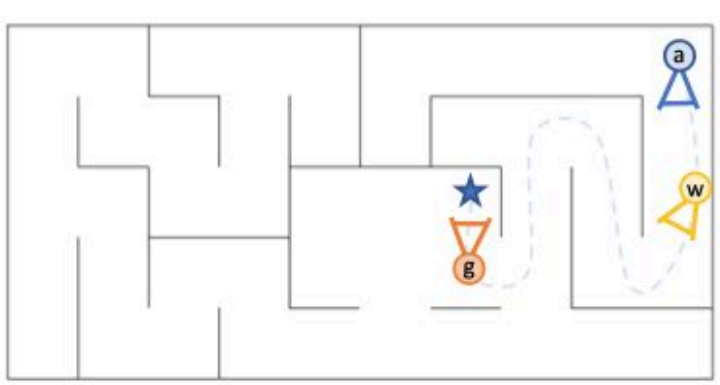
RLLAB
http://rllab.snu.ac.kr



- The waypoint vertex v^w) yellow) is selected as the vertex in the shortest path that is furthest f or the agent's vertex v^a but **can still be confidently reached** by the agent.
- The output of the SPTM is the corresponding waypoint observation $o^w = o_{v^w}$.

Image Goal Navigation

RLLAB
http://rllab.snu.ac.kr



(a) Maze

(b) Agent's observation

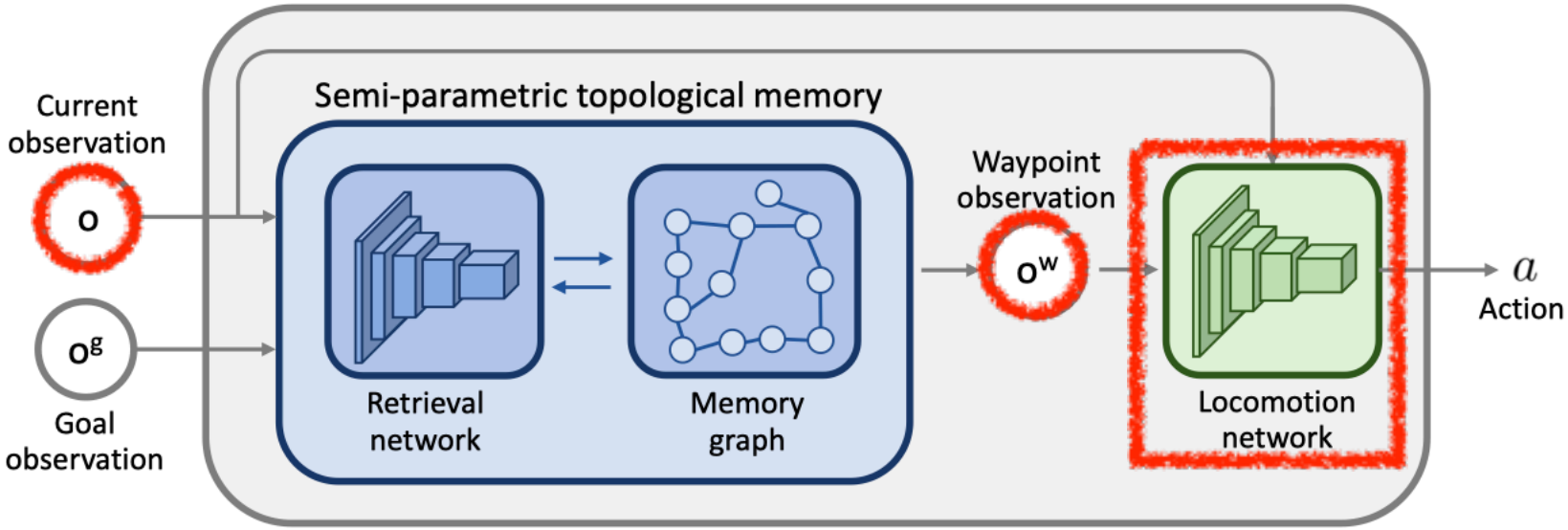
(c) Waypoint observation

(d) Goal observation

- A waypoint observation is produced by SPTM given agent's observation (b) and goal observatio n (d).

Image Goal Navigation

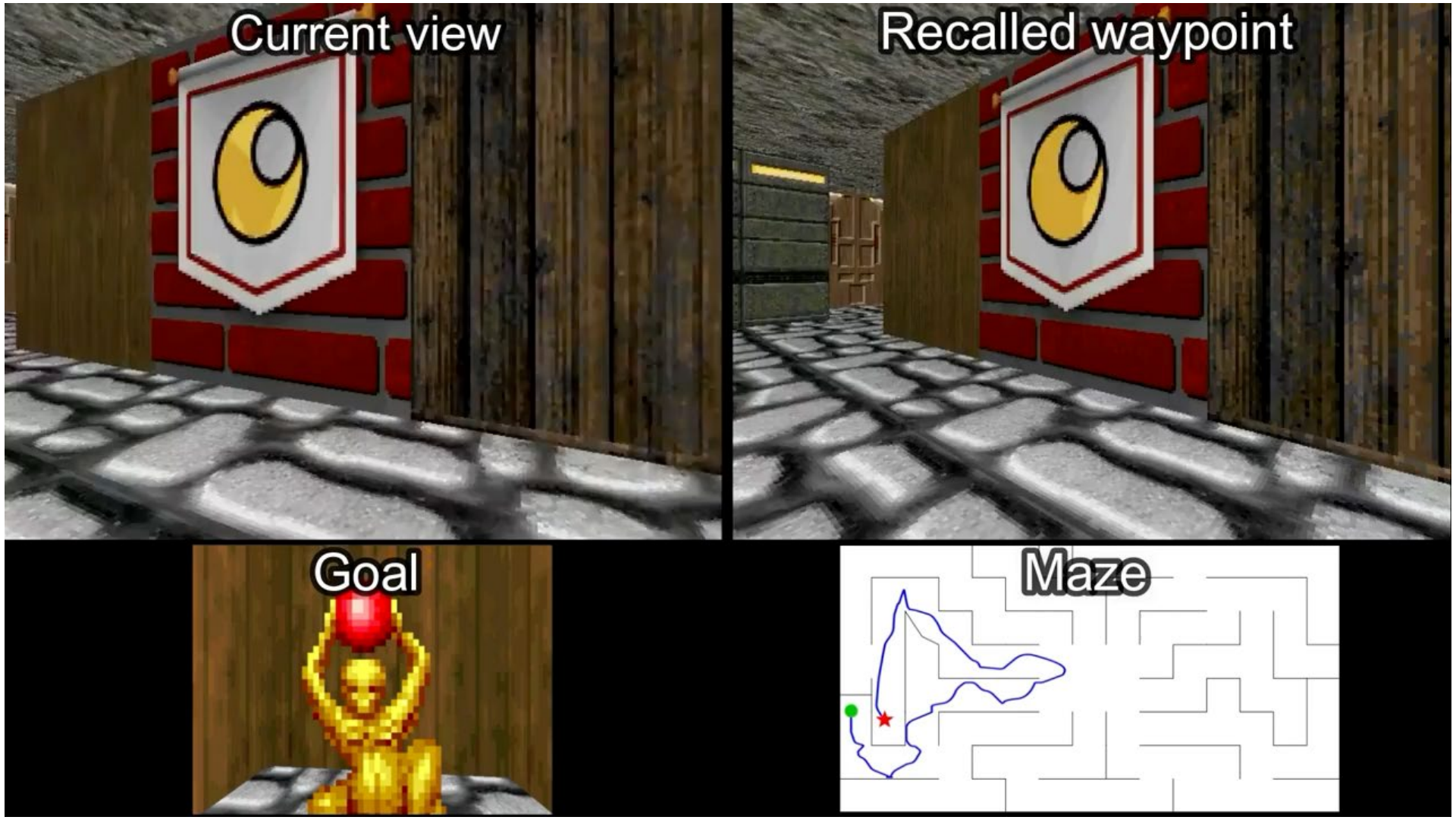
RLLAB
http://rllab.snu.ac.kr



- **Locomotion network** L:
  - The network L is trained to navigate toward **target observations** near the agent.
  - The network maps a pair (o_1, o_2) ,which consists of a current and a goal observations, in to action probabilities.

Image Goal Navigation

RLLAB
http://rllab.snu.ac.kr



RLLAB
http://rllab.snu.ac.kr

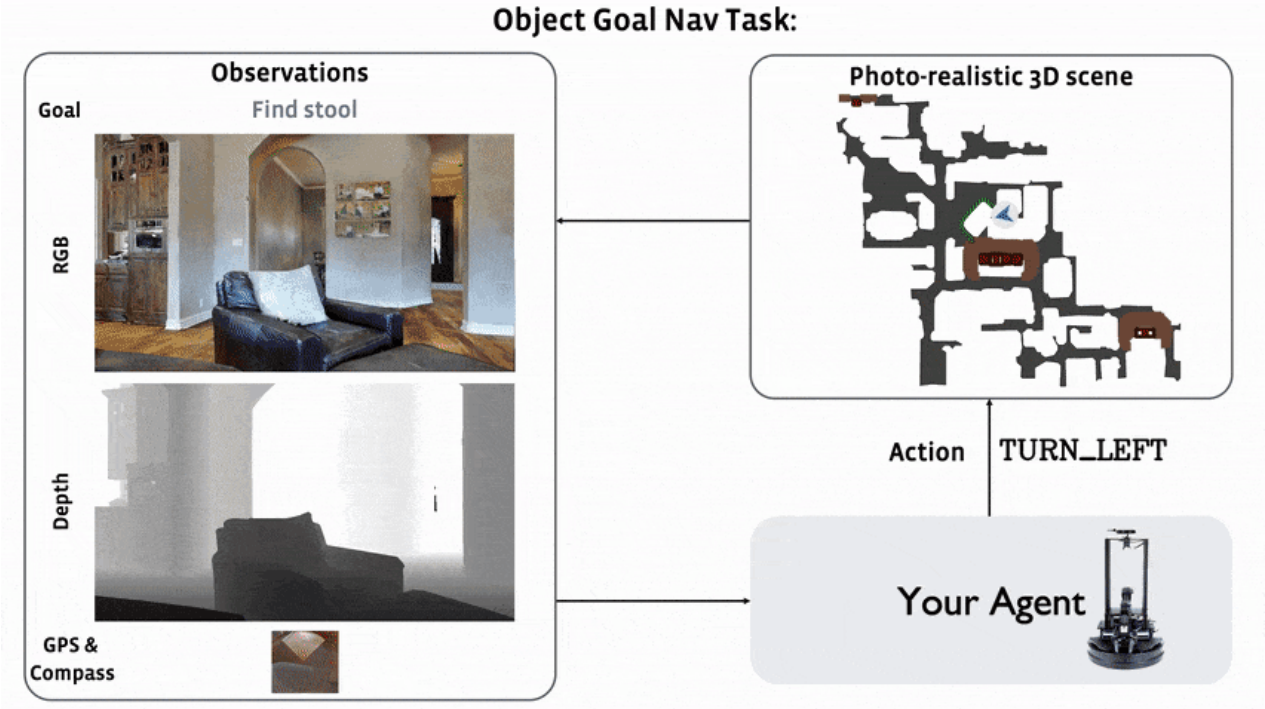
Object Goal Navigation

PONI: Potential Functions for ObjectGoal Navigation with Interaction-free Learning, CVPR 2022

Object Goal Navigation

RLLAB
http://rllab.snu.ac.kr

- In ObjectNav, an agent is initialized at a random starting position and orientation in an unseen environment and asked to find an instance of an **object category** (*'find a chair'*) by navigating to it. A map of the environment is not provided and the agent must only use its sensory input to navigate.
- The agent is equipped with an **RGB-D camera and a (noiseless) GPS+Compass sensor**. GPS+Compass sensor provides the agent's current location and orientation information relative to the start of the episode. We attempt to match the camera specification (field of view, resolution) in simulation to the Azure Kinect camera, but this task does not involve any injected sensing noise.



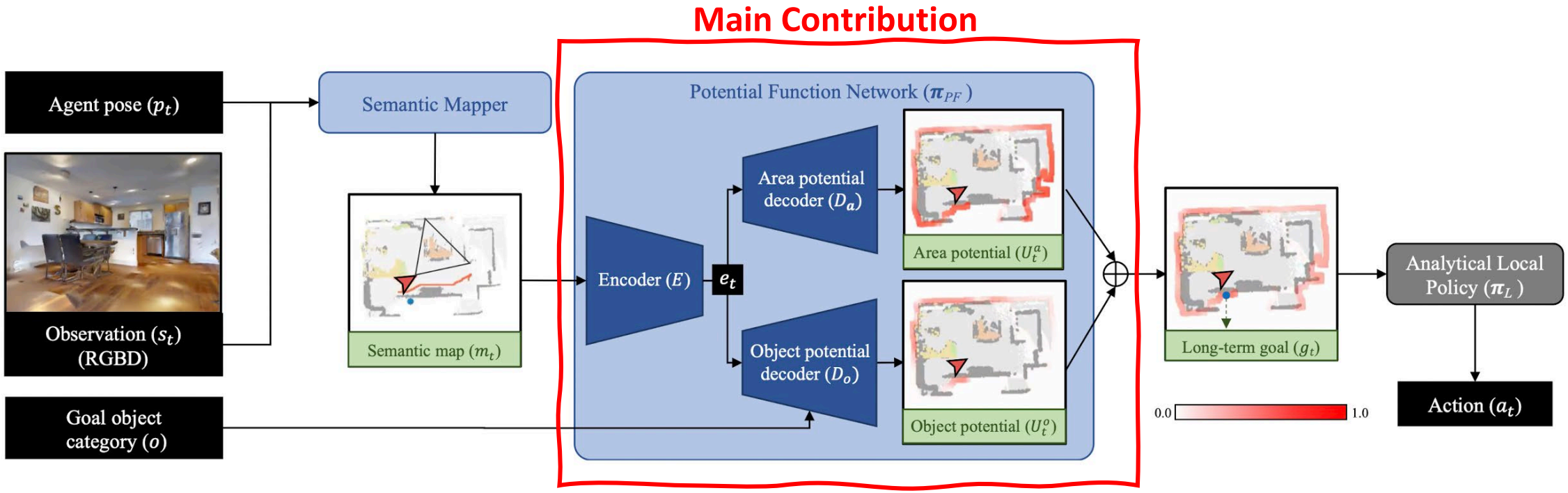
Object Goal Navigation

RLLAB
http://rllab.snu.ac.kr

- "PONI: Potential Functions for ObjectGoal Navigation with Interaction-free Learning", CVPR 2022
  - **Learning to infer *"where to look"* without any *interactions.***
  - Potential Function
    - Defined at the frontiers of a 2D top-down semantic map.
    - Estimated from a partially filled semantic map
  - **Interaction free learning** from dataset
  - Outperforms SOTA on Gibson with 7x lower training cost.

Object Goal Navigation

RLLAB
http://rllab.snu.ac.kr

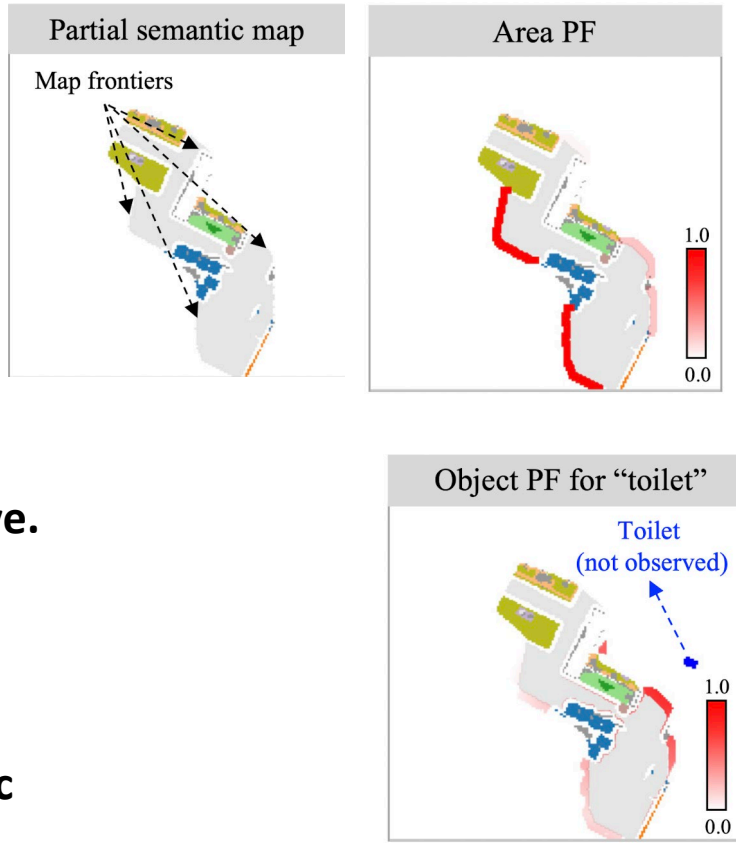


Object Goal Navigation

RLLAB
http://rllab.snu.ac.kr

Potential Function *: Where to look for finding goal object o?*

- The potentials are only defined at the map ***frontiers.***
- **Area Potential Function U_t^a**
  - Area potential $U_t^a(f)$ at a frontier f
    : the amount of free-space left to explore beyond f
  - A guide for efficient exploration, helps find unexplored areas.
  - This function is **critical when the semantic map is not informative.**
- **Object Potential Function U_t^o**
  - A guide for efficient object search, helps find the object o
  - This is **critical to perform semantic reasoning when the semantic is sufficiently informative.**



Object Goal Navigation

RLLAB
http://rllab.snu.ac.kr

- **Area Potential Function**
  - Area potential $U_t^a(f)$ at a frontier f
    : the *amount of free-space left* to explore beyond f
    : *navigable cells which are unexplored* in the partial semantic map.
  - **How to calculate?**
    - Group the unexplored free-space cells into **connected components** $\mathrm{C} = \{c_1, \ldots, c_n\}$ using OpenCV

Object Goal Navigation

RLLAB
http://rllab.snu.ac.kr

- **Area Potential Function**
  - A component c is associated with frontier f only if at least one pixel in c is an 8-connected neighbor of some pixel in f .
  - For each frontier f, area potential $U_t^a(f)$ = sum of areas of connected components associated with f and normalize.

Object Goal Navigation

RLLAB
http://rllab.snu.ac.kr

- **Area Potential Function**
  - A component c is associated with frontier f only if at least one pixel in c is an 8-connected neighbor of some pixel in f.
  - For each frontier f, area potential $U_t^a(f)$ = sum of areas of connected components associated with f and normalize.

Object Goal Navigation

RLLAB
http://rllab.snu.ac.kr

- **Potential Function Network**
  - **Object Potential Function**
    - Geodesic distance between a frontier location x, Object category o_t

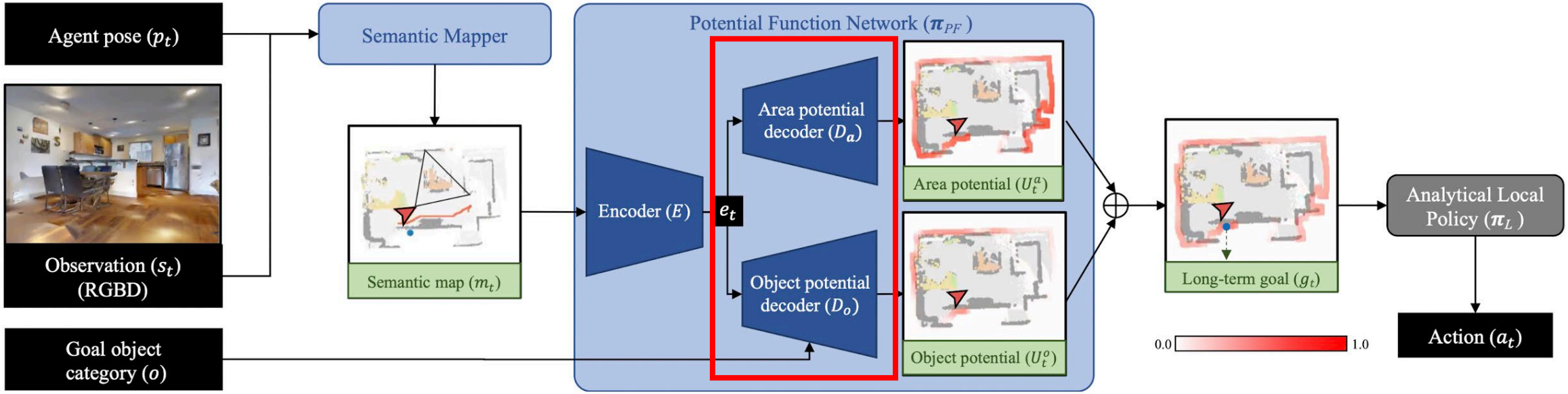
$$U_t^o(o_t, x) = \max(1 - \frac{d_g(o_t, x)}{d_{max}}, 0.0) \qquad d_{max} = 10m$$

If $o_t =$ chair

Object Goal Navigation

RLLAB
http://rllab.snu.ac.kr

- **Potential Function Network**



- Unet Decoder
- Object potential decoder outputs N-channel map : (the number of object categories)

Object Goal Navigation

RLLAB
http://rllab.snu.ac.kr

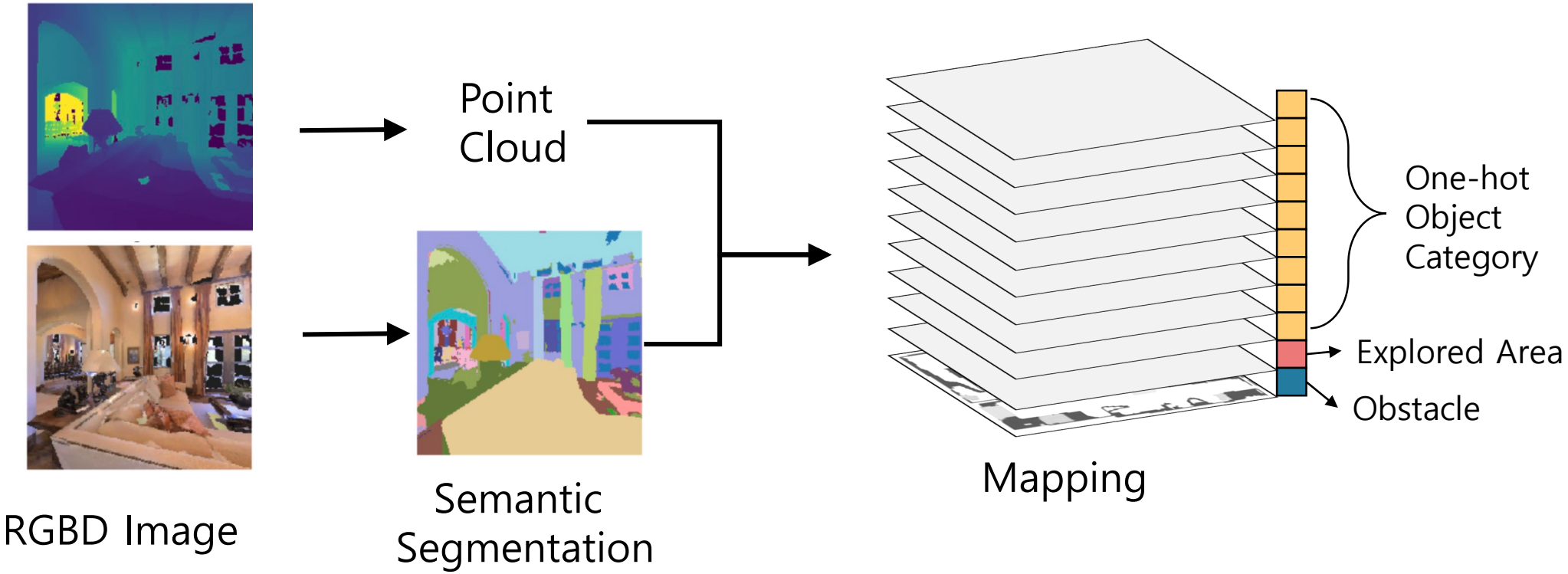
- **Long-term Goal Sampling**
  - Linearly combine the area and object potentials to obtain overall potential.

$$U_t = \alpha U_t^a + (1 - \alpha) U_t^o$$

  - $\alpha = 0.5$ is decided via validation experiments.
  - *Zero-out U_t at all explored map* locations except frontiers.
  - "Since the map frontiers can be noisy during navigation, we retain the predictions from the **unexplored locations**, providing the model some flexibility in deciding the frontier boundaries."

Object Goal Navigation

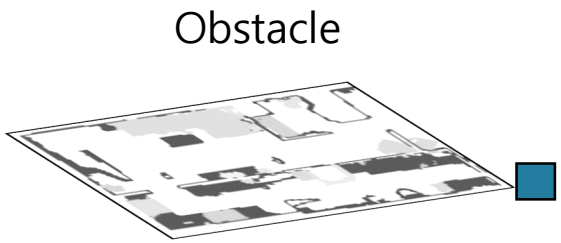
RLLAB
http://rllab.snu.ac.kr



Object Goal Navigation

RLLAB
http://rllab.snu.ac.kr

- **Analytical Local Policy**



- Compute the shortest path using Fast Marching Method
- Takes deterministic actions along the path
- This was found to be as effective as a learned policy (e.g. ANS)

Object Goal Navigation

RLLAB
http://rllab.snu.ac.kr

- Learning without any environmental interactions!
- Potential function network π_{pf}
  - A dataset of semantic maps that are pre-computed
    - Using Semantic MapNet
  - Data Tuple Generation

m^c: complete semantic map

m^p: partial semantic map

Object Goal Navigation

RLLAB
http://rllab.snu.ac.kr

- **Training**
  - Loss function for Potential function network π_{pf}

$$L_a = \frac{1}{|\mathcal{F}|} \sum_{x \in \mathcal{F}} \left|\left| \hat{U}^a(x) - U^a(x) \right|\right|_2^2$$

$$L_c = \frac{1}{|\mathcal{F}|N} \sum_{x \in \mathcal{F}} \sum_{n=1}^{N} \left|\left| \hat{U}^o(x, n) - U^o(x, n) \right|\right|_2^2$$

F : a set of frontier pixels
N : the number of object categories

Object Goal Navigation

RLLAB
http://rllab.snu.ac.kr

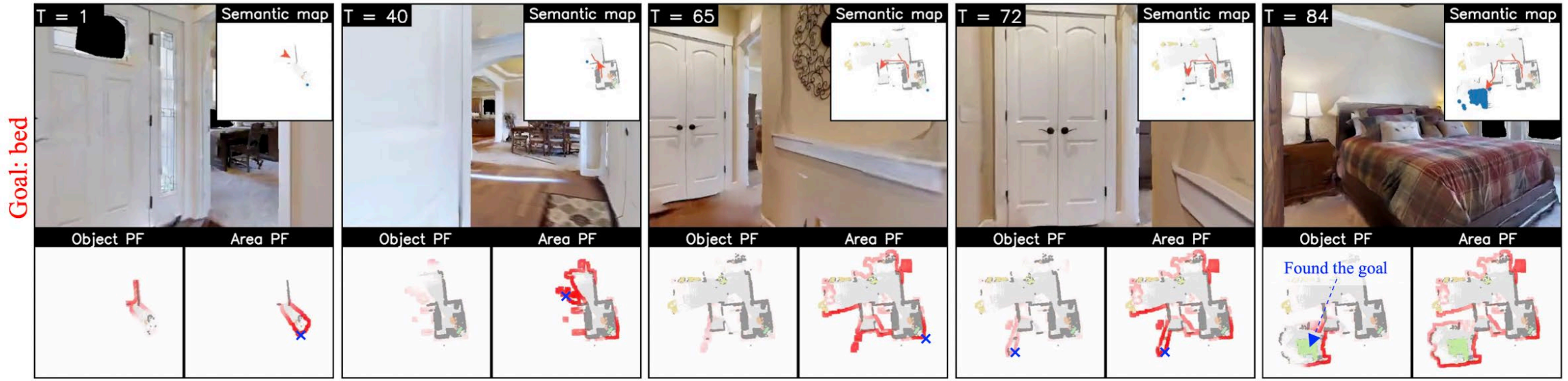


Figure 5. **Qualitative example of navigation using potential functions**. We visualize parts of an ObjectNav episode on Gibson (val), starting from T=1 until the agent finds the goal object (bed). For each step, we show the egocentric RGB view, the predicted semantic map, object and area potential functions. We indicate the maximum location that the agent navigates to using a blue cross on the PF map(s) responsible for the maximum. At the episode start (T=1 to 65), the agent is guided by the area PF which is high near frontiers leading to unexplored areas, allowing it to explore and gather information. The object PF plays a limited role here. After having gathered information, the model predicts higher object PF near the bedroom entrance at T=72, while the area PF remains high at multiple frontiers unrelated to the object location. The agent uses the new signal from the object PF to enter the bedroom and find the bed at T=84. This highlights the value of the two potential functions and how they are combined to perform ObjectNav. Please see supplementary for more examples.

RLLAB
http://rllab.snu.ac.kr

Vision and Language Navigation

Vision and Language Navigation

RLLAB
http://rllab.snu.ac.kr

- It requires an autonomous agent to follow a natural language navigation instruction to navigate to a goal location in a previously unseen real-world building.
- The challenge is situated in the Matterport3D Simulator -- a large-scale reinforcement learning environment based on panoramic images from the Matterport3D dataset.
- The instructions and trajectories comprise the Room-to-Room (R2R) natural language navigation dataset.

Leave the bedroom, and enter the kitchen. Walk forward, and take a left at the couch. Stop in front of the window.

TSGM

RLLAB
http://rllab.snu.ac.kr

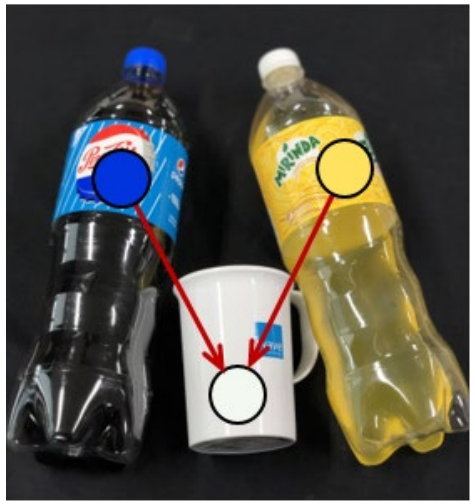
- Visual navigation has two branches
  1. Metric map-based
     - Pros: Exact
     - Cons: Computationally expensive
  2. Topological-memory-based
     - Pros: Robust to noise
     - Cons: Less accurate

- Topoloigical Semantic Graph memory (TSGM)
  - Leverages a semantic navigation like human

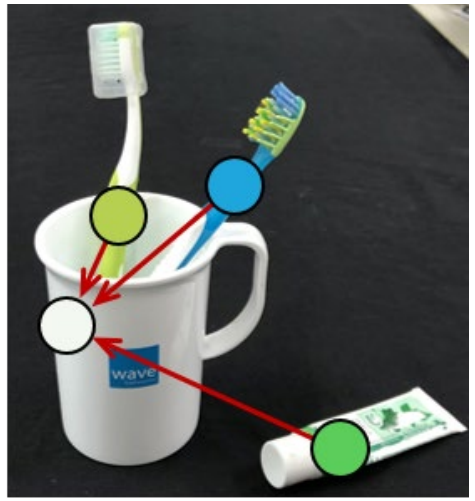
Why *Object?*

RLLAB
http://rllab.snu.ac.kr

- Learning to navigate using **context** of objects
  - Visual navigation methods simply encode the whole image
  - **Object contexts** provide semantic information when there is the same object with different configurations



A cup for drinking

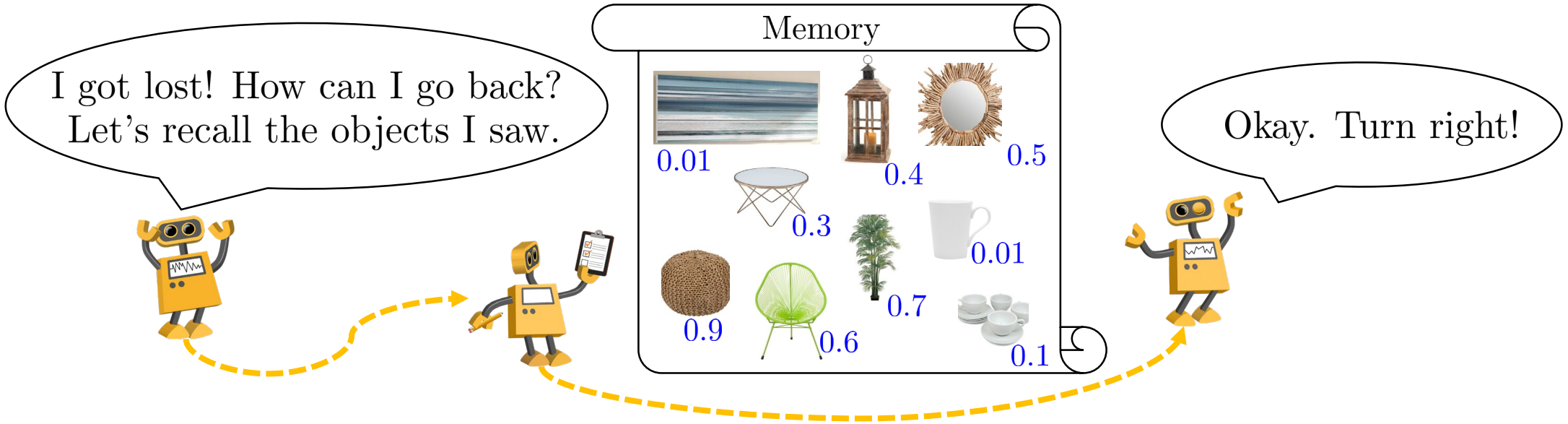


A cup for toothbrushing

  - Rather than image-based memory, utilizing **object-context memory** can help an agent to achieve the target task

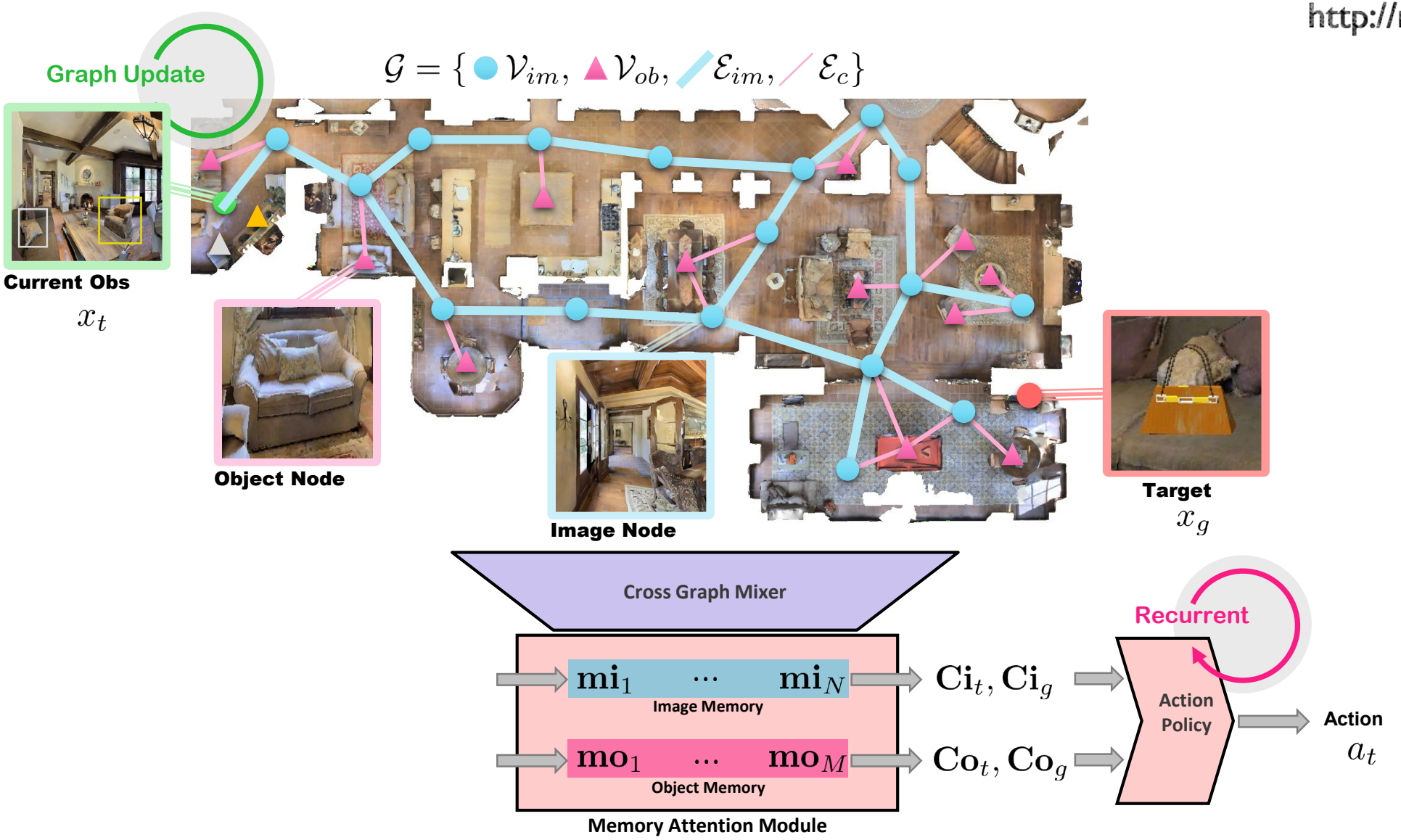
TSGM

RLLAB
http://rllab.snu.ac.kr



TSGM

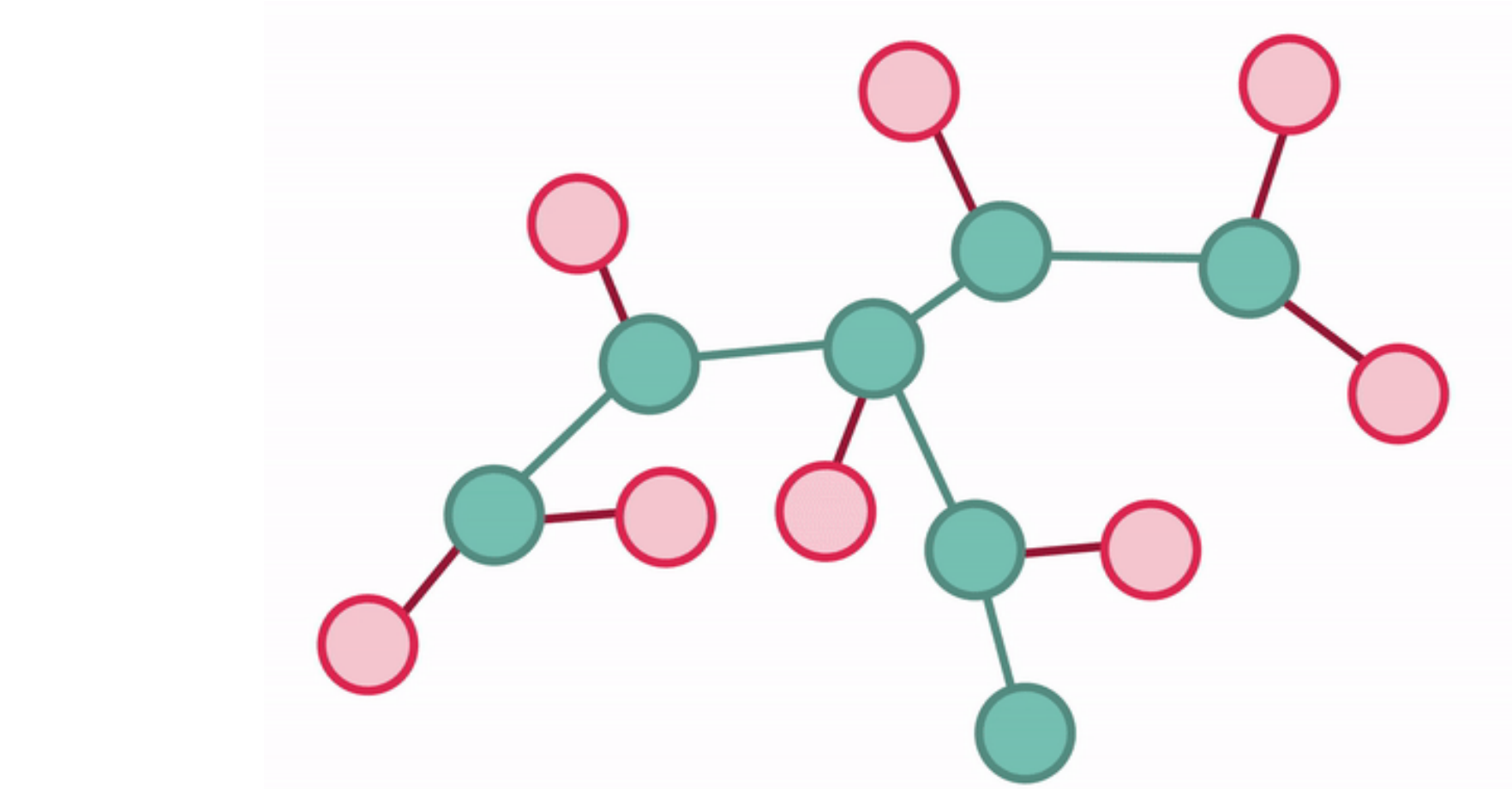
RLLAB
http://rllab.snu.ac.kr



TSGM

RLLAB
http://rllab.snu.ac.kr

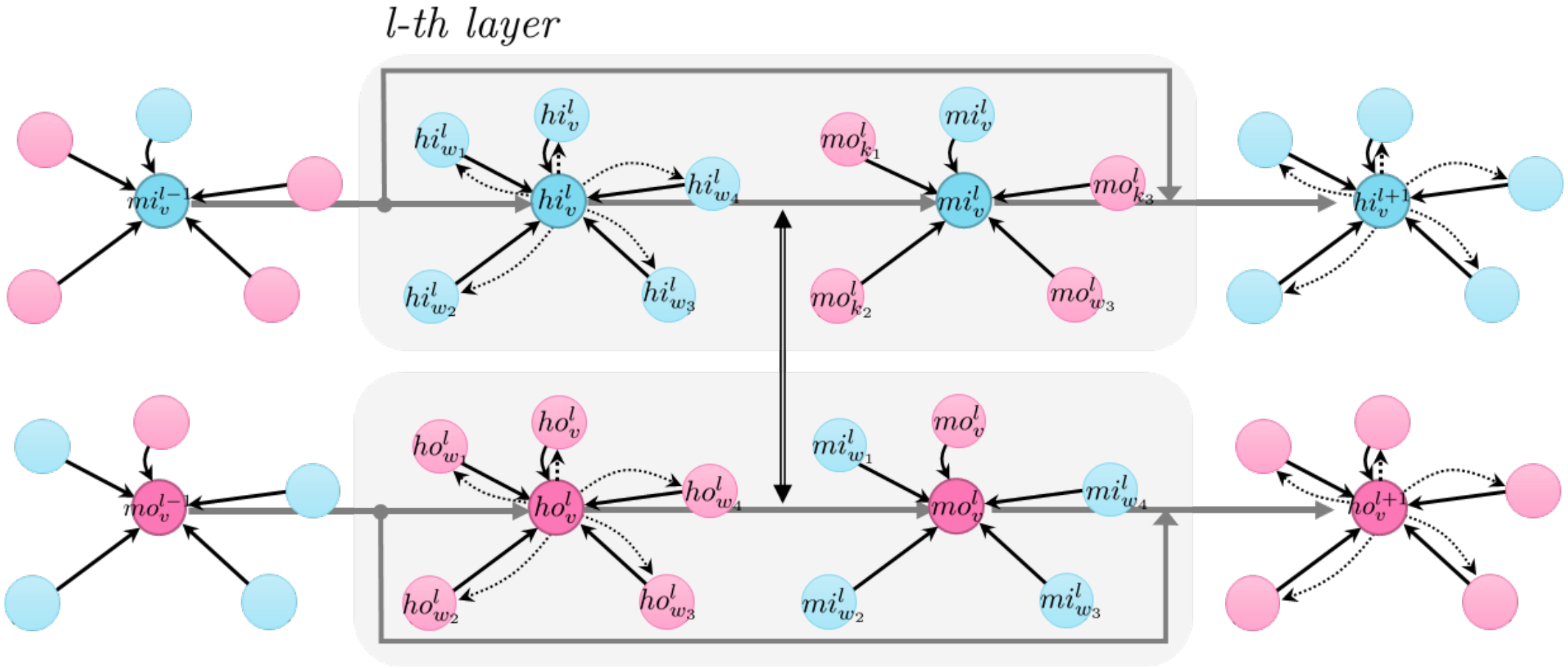
- Graph Connection



TSGM

RLLAB
http://rllab.snu.ac.kr

- Cross Graph Mixer



TSGM

RLLAB
http://rllab.snu.ac.kr

Agent

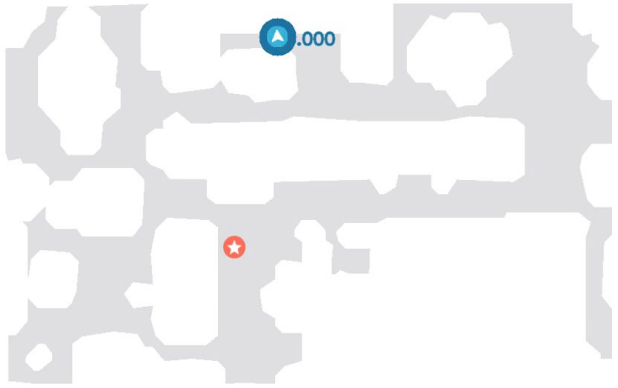
Target

Found

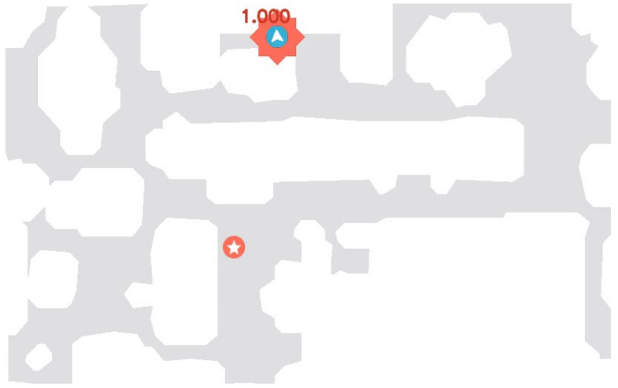
current observation

target observation

Attention scores
about current observation



Attention scores
about target observation



TSGM

RLLAB
http://rllab.snu.ac.kr

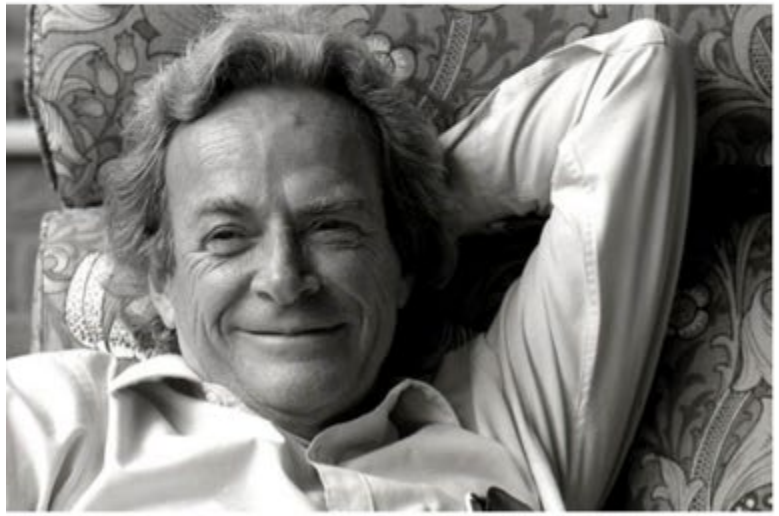
Table 2: Comparison of our model (TSGM) with baselines on Image-Goal Navigation on **Gibson** Test Episodes. We report average Success rate and SPL @ $1m$.

Path Type	Method	Easy		Medium		Hard		Overall	
		Success	SPL	Success	SPL	Success	SPL	Success	SPL
Straight	RL (10M step)	10.5	6.7	18.1	15.1	11.7	10.8	13.4	10.9
	RL (extra data + 100M steps)	43.2	38.5	36.4	34.8	7.4	7.2	29.0	26.8
	BC w/ ResNet + Metric Map	24.8	23.9	11.5	11.2	1.3	1.2	12.5	12.1
	BC w/ ResNet + GRU	34.9	33.4	17.6	17.0	6.1	5.9	19.5	18.8
	NRNS [33]	68.0	61.6	49.1	44.5	23.8	18.2	46.9	41.4
	NRNS [33] (pano)	67.1	57.8	52.4	41.2	32.6	22.4	50.7	40.5
	VGM [20]	81.0	54.4	82.0	69.9	67.3	54.4	76.7	59.6
	TSGM (Ours)	**94.4**	**92.1**	**92.6**	**84.3**	**70.3**	**62.8**	**85.7**	**79.8**
Curved	RL (10M step)	7.5	3.2	9.5	7.1	5.5	4.7	7.5	5.0
	RL (extra data + 100M steps)	22.2	16.5	20.7	18.5	4.2	3.7	15.7	12.9
	BC w/ ResNet + Metric Map	3.1	2.5	0.8	0.7	0.2	0.1	1.3	1.1
	BC w/ ResNet + GRU	3.6	2.8	1.1	0.9	0.5	0.3	1.7	1.3
	NRNS [33]	36.5	18.3	23.9	12.0	12.5	6.8	24.3	12.4
	NRNS [33] (pano)	31.7	13.0	29.0	13.6	19.2	10.4	26.6	12.3
	VGM [20]	81.0	45.5	78.8	59.5	62.2	46.9	74.0	50.6
	TSGM (Ours)	**93.6**	**91.0**	**89.7**	**77.8**	**64.2**	**55.0**	**82.5**	**74.1**

Research Tips

RLLAB
http://rllab.snu.ac.kr

On Richard Feynman's problem solving



- The Feynman problem solving algorithm:

1. Write down the problem
2. Think very, very hard
3. Write down the solution

Research Tips

RLLAB
http://rllab.snu.ac.kr

- Keep up with recent researches
  - Google scholar keyword alerts
  - Paper study with colleagues
- Organize research materials
  - EndNote (paper)
  - Notion (research journal)
  - Slack (experimental results)
  - Github (code)
  - PPT (organize read papers in ppt)
- Visualize your work
  - Wandb / Tensorboard (training)
  - ipython notebook (simple test/visualize)
  - at least plt.show()

Thank you for listening

Papers (Task)

RLLAB
http://rllab.snu.ac.kr

- Exploration
  - Mid-level visual representations improve generalization and sample efficiency for learning active tasks, CoRL 2019
  - SplitNet: Sim2Sim and Task2Task Transfer for Embodied Visual Navigation, ICCV 2019
  - Learning Exploration Policies for Navigation, ICLR 2019
  - Learning To Explore Using Active Neural SLAM, ICLR 2020
- Active Vision
  - Viewpoint Selection for Visual Failure Detection, IROS 2017
  - A dataset for developing and benchmarking active vision, ICRA 2017
  - Geometry-aware recurrent neural networks for active visual recognition, NIPS 2018
  - Learning to look around: Intelligently exploring unseen environments for unknown tasks, CVPR 2018
  - Embodied Visual Recognition, ICCV 2019
  - SEAL: Self-supervised Embodied Active Learning using Exploration and 3D Consistency, NeurIPS 2021

Papers (Task)

RLLAB
http://rllab.snu.ac.kr

- Point Goal Navigation
  - A Behavioral Approach to Visual Navigation with Graph Localization Networks, RSS 2019
  - Learning Exploration Policies for Navigation, ICLR 2019.
  - Sparse Graphical Memory for Robust Planning, arXiv 2020
  - Active Neural Localization, ICLR 2018
  - Active Neural SLAM, ICLR 2020
- Image Goal Navigation
  - Target-driven Visual Navigation in Indoor Scenes using Deep Reinforcement Learning, ICRA 2017
  - Semi-Parametric Topological Memory for Navigation, ICLR 2018
  - Sparse Graphical Memory for Robust Planning, arXiv 2020
- Object Goal Navigation
  - Auxiliary Tasks and Exploration Enable ObjectNav, ICCV 2021
  - Treasure Hunt Data Augmentation for Semantic Navigation, ICCV 2021
  - Object Goal Navigation using Goal-Oriented Semantic Exploration, NeurIPS 2020
  - Learning to Map for Active Semantic Goal Navigation, ICLR 2022
  - PONI: Potential Functions for ObjectGoal Navigation with Interaction-free Learning, CVPR 2022

Papers (Task)

RLLAB
http://rllab.snu.ac.kr

- Visual Language Navigation
  - Hierarchical Cross-Modal Agent for Robotics Vision-and-Language Navigation, ICRA 2021
  - Waypoint Models for Instruction-guided Navigation in Continuous Environments, ICCV 2021

Papers (Memory Structure)

RLLAB
http://rllab.snu.ac.kr

- Metric-Map Memory
  - Learning To Explore Using Active Neural SLAM. ICLR, 2020.
  - Object Goal Navigation using Goal-oriented Semantic Exploration. NeurIPS, 2020.
  - Semantic MapNet: Building Allocentric Semantic Maps and Representations from Egocentric Views. AAAI, 2021.
- Topological Memory
  - Semi-parametric Topological Memory for Navigation, ICLR, 2018.
  - Neural Topological SLAM for Visual Navigation, CVPR, 2020.
  - Hallucinative Topological Memory for Zero-Shot Visual Planning, ICML, 2020.
  - Topological Planning with Transformers for Vision-and-Language Navigation, CVPR, 2021.
  - Pose Invariant Topological Memory for Visual Navigation, ICCV, 2021.
  - Visual Graph Memory With Unsupervised Representation for Visual Navigation, ICCV, 2021.