CS686: Motion Planning and Applications
Paper Presentation - I

BADGR : An Autonomous Self-Supervised Learning-Based Navigation System

Jeil Jeong
(정제일)

Review

Parallel Monte Carlo Tree Search with Batched Rigid-body Simulations for Speeding up Long-Horizon Episodic Robot Planning

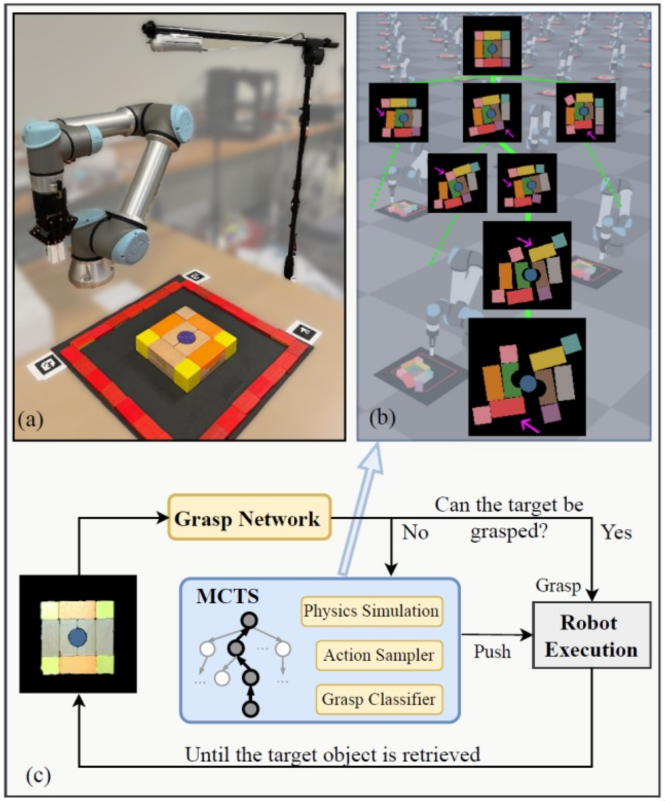
Purpose

- For Long-Horizon, Episodic Robotic Planning Task
  - Accelerating
  - Improve solution quality

Main Idea

- Use Parallel Monte Carlo Tree Search
- Use Batched Simulation

➔ PMBS



3

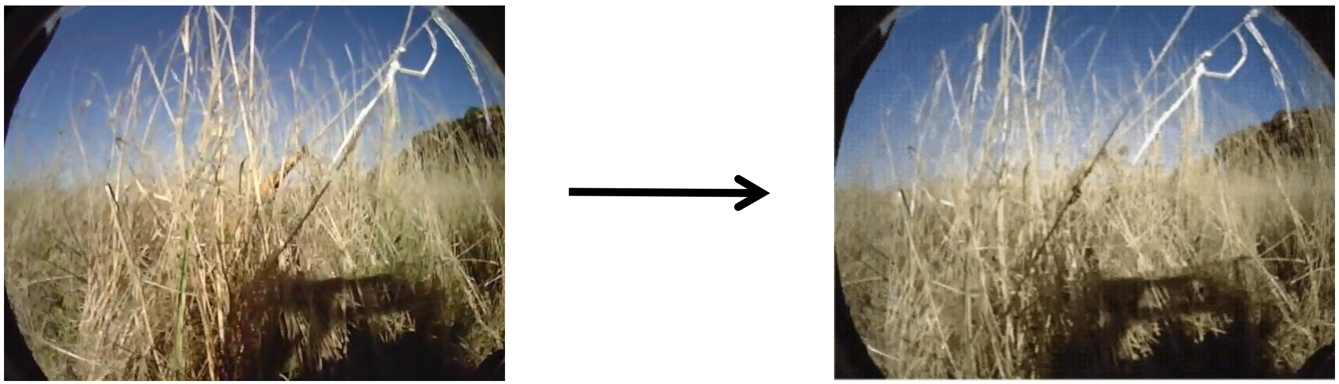
Table of Contents

1. Motivations

Motivations

Purely Geometric view can be insufficient for Navigation

- All tall grass can be considered as an obstacle



- Can not differentiate smooth path and bumpy grass

Motivations

Enabling robots to reason about navigational affordances

It is tall grass
We can traverse this area

There are two possible paths
Paved path is better than bumpy grass in terms of stability.

KAIST

Motivations

This problem has been approached from the standpoint of semantic understanding

- **Requires human – supervision**
- **Only consider traversability(by using labeling)**

The main idea is that using robot's own past experience

- **With self-supervision (without human – supervision)**
- **Not only traversability but many physical attributes (avoiding collision, ground type - vehicle interaction)**

Motivations

- Overview of BADGR (Berkeley Autonomous Driving Ground Robot)

1. Collecting Data
   - Collecting Experience

2. Self – Supervised Data Labelling
   - robot's position, collision, terrain bumpiness etc.

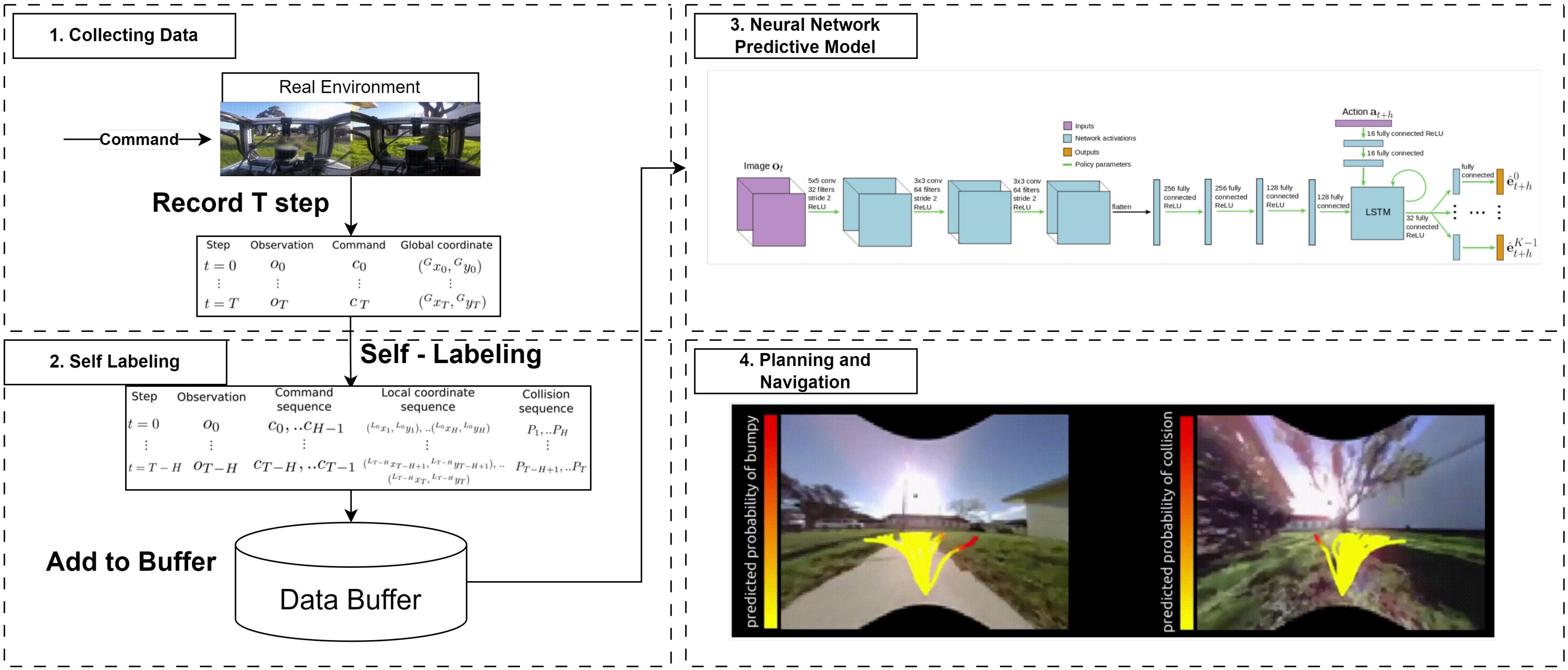
3. Neural Network Predictive Model (Forward Dynamics Model)
   - given input(image, command sequence), predict future events

4. Planning and Navigation

Using Robot's experience to train a predictive model

Motivations

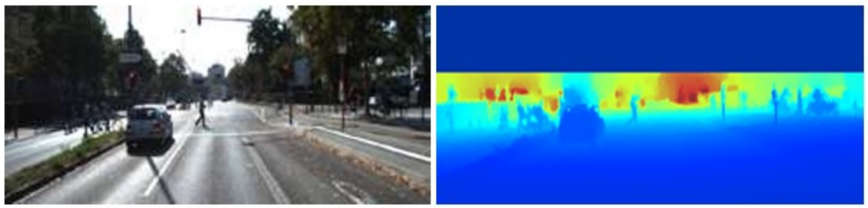
- **Overview of BADGR (Berkeley Autonomous Driving Ground Robot)**



2. Related Work

Related Work

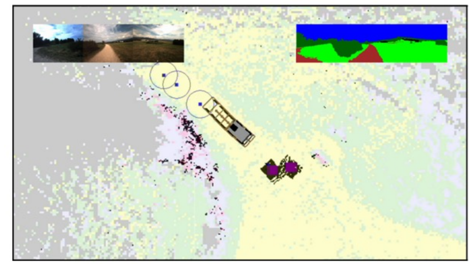
- Geometric view-based navigation



- Estimate Depth & Shape of Obstacle
- Plan a collision free path from estimated scene
- Can not handle semantic information (ex. Tall Grass, puddle)
- Weak at handling textureless scenes (Lidar, Depth sensor limitation)

Wang, Yan, et al. "Pseudo-lidar from visual depth estimation: Bridging the gap in 3d object detection for autonomous driving." *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2019.

- Semantic view-based navigation



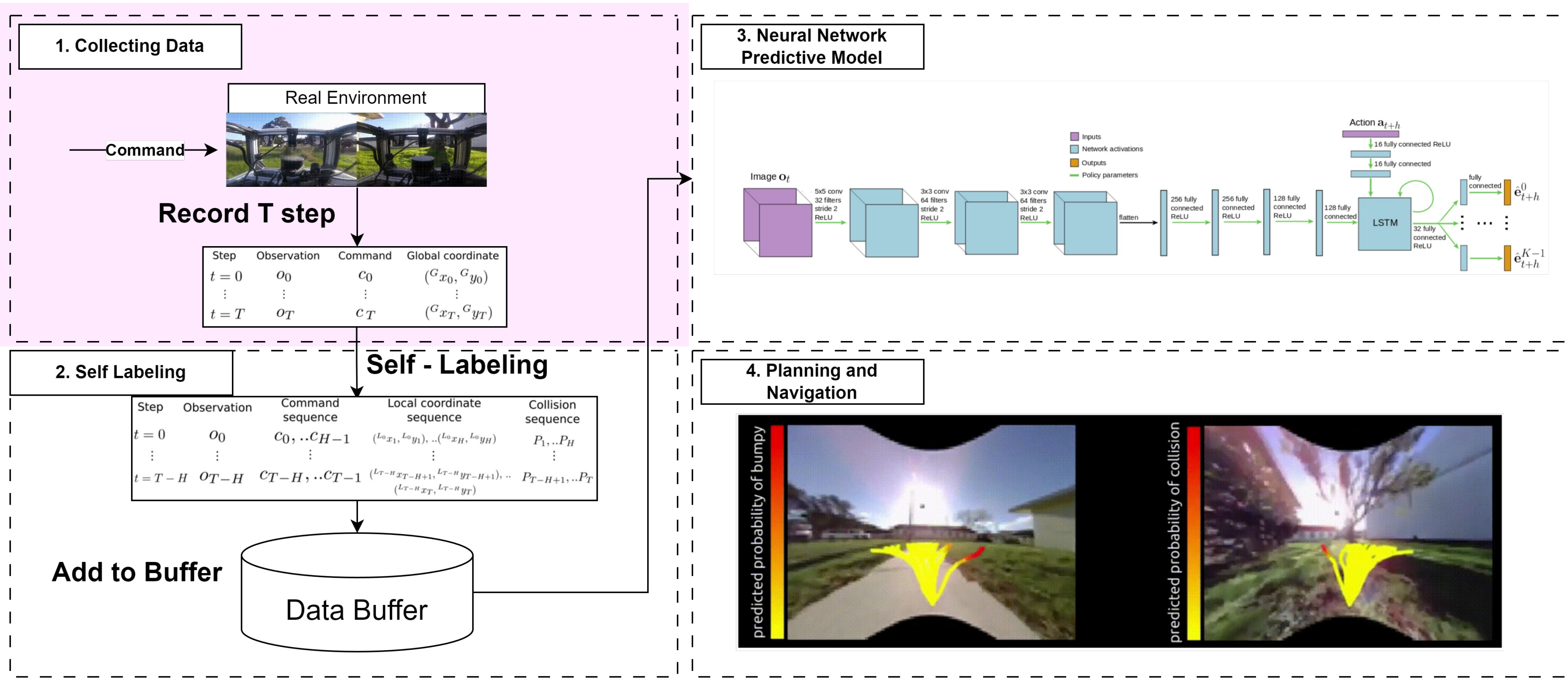
- Give meaningful labels as a semantic information
- Plan a collision free path considering semantic information
- Need exhaustive human - supervision
- Does not consider ground-vehicle interaction (Bumpiness and Stability of Vehicle)

Valada, Abhinav, et al. "Adapnet: Adaptive semantic segmentation in adverse environmental conditions." *2017 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2017.

3. Method

Method - Collecting Data

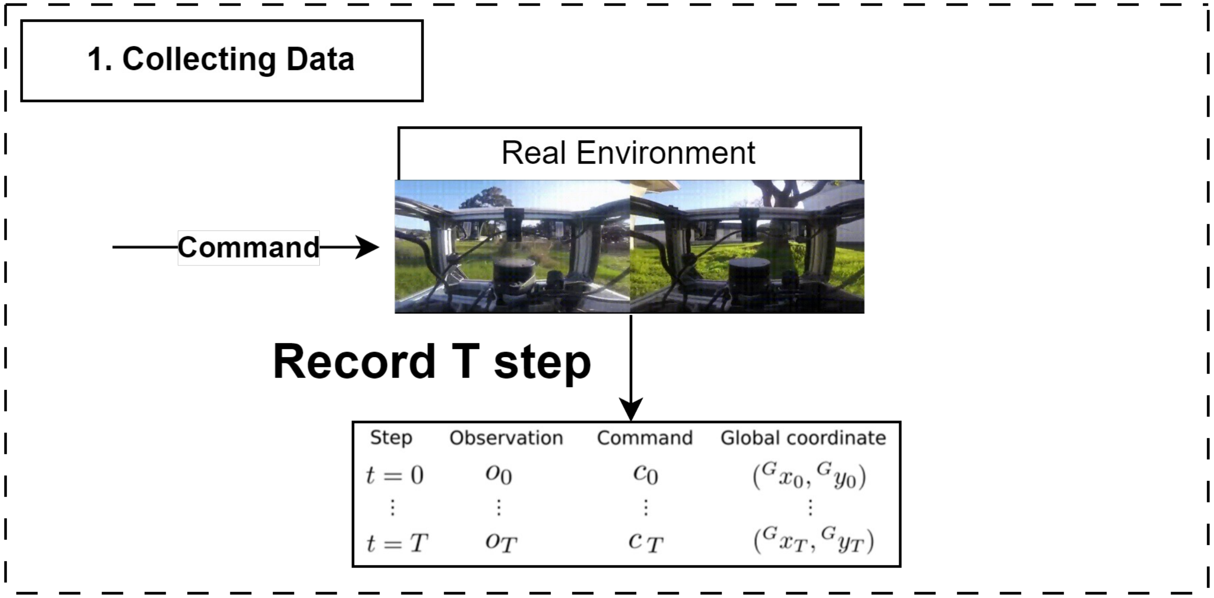
1. Collecting Data



Method - Collecting Data

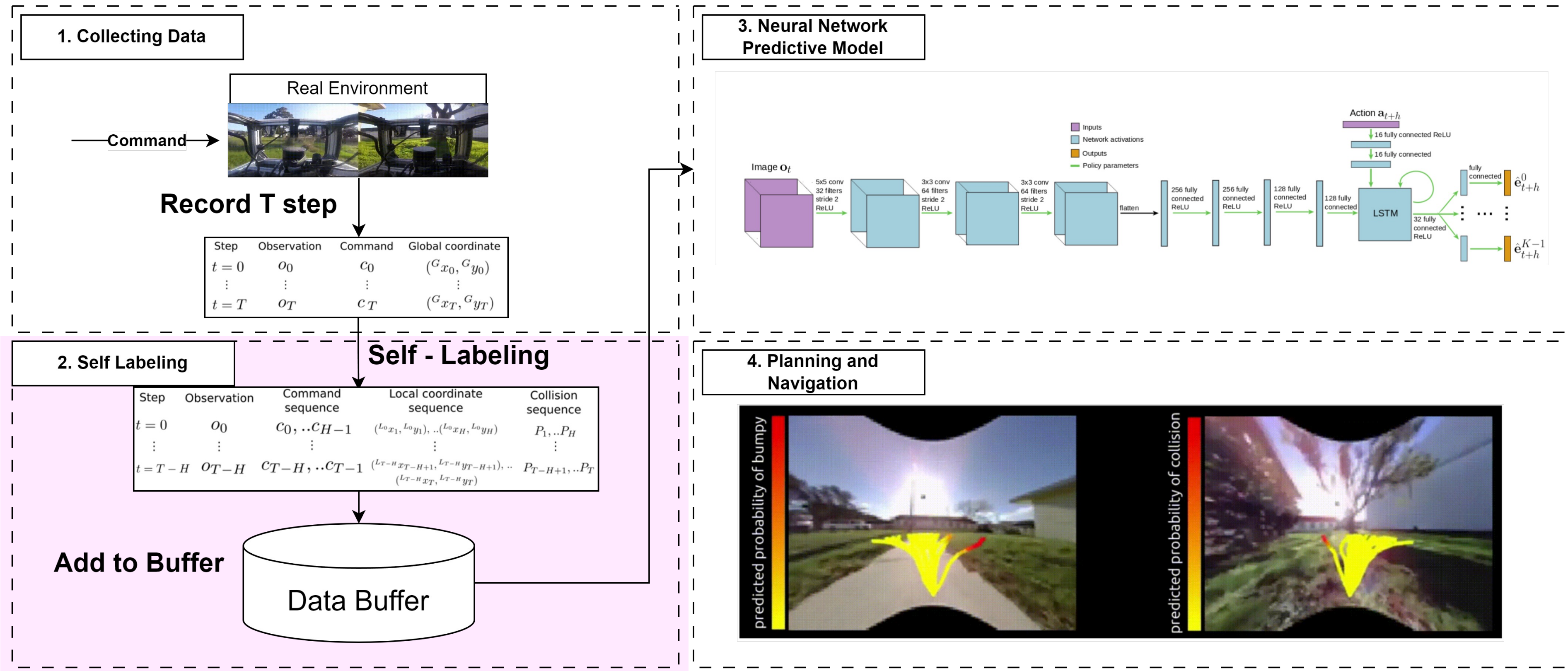
Randomly sample a sequence of commands, and execute commands in real environment

Records every observation(RGB image), command, robot position, IMU data.



Method – Self-Labeling

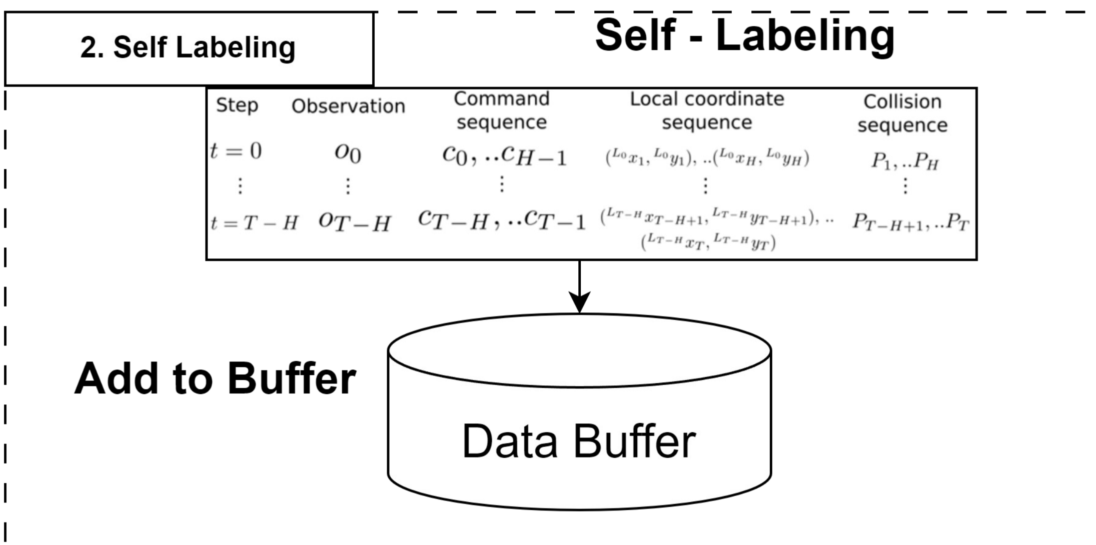
2. Self Labeling



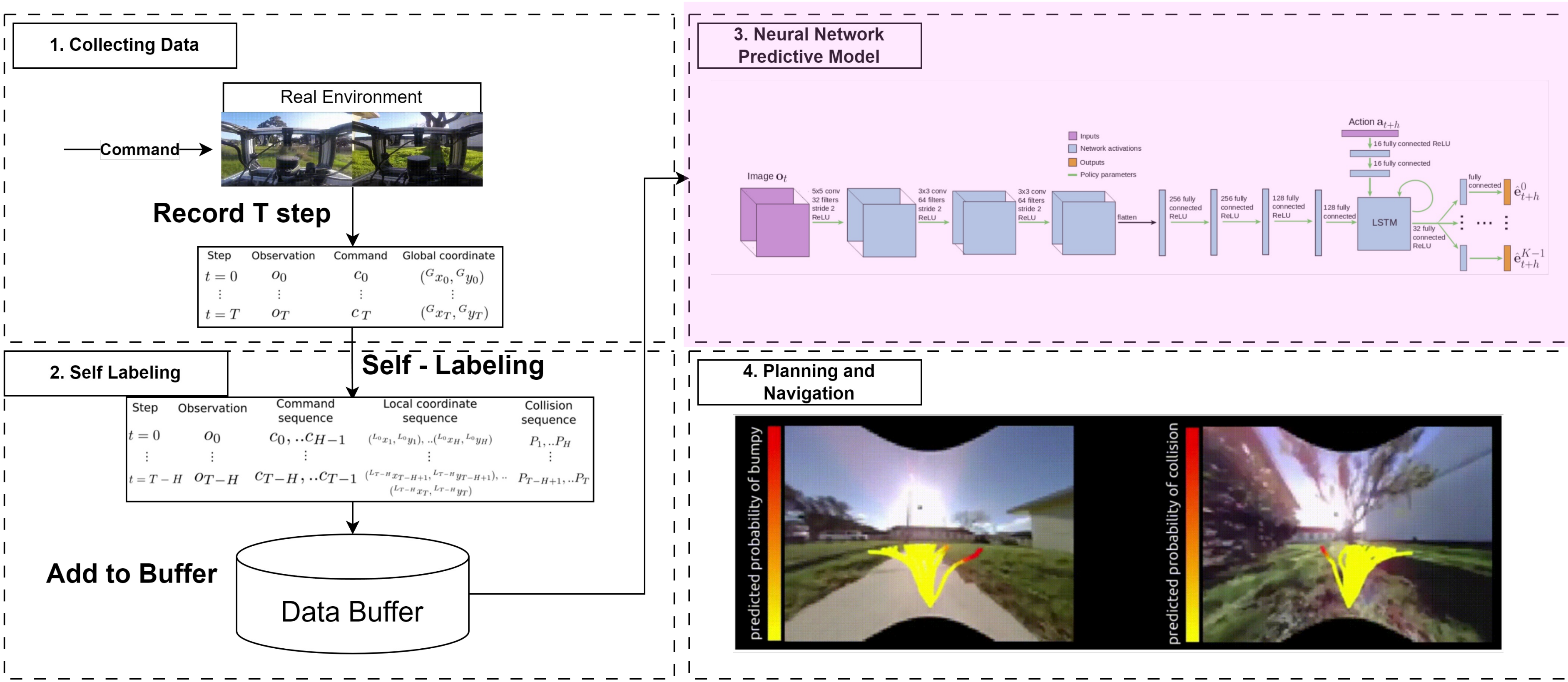
Method – Self-Labeling

After Collecting a Data, BADGR give labels for specific navigational events.
Mapping : Raw sensor data ⇒ Navigation event

Navigational Events	Labeling Criterion
Collision	1. Distance to Obstacle < Certain threshold 2. Change rate of linear acceleration > Certain threshold
Driving Over Bumpy terrain	1. Angular Velocity Magnitude > Certain threshold



Method
Neural Network Predictive Model



Method
Neural Network Predictive Model

BADGR uses collected data to train a deep neural network predictive model

$$\boxed{f_\theta(\mathbf{o}_t, \mathbf{a}_{t:t+H}) \quad \rightarrow \quad \hat{\mathbf{e}}^{0:K}_{t:t+H}}$$

Input

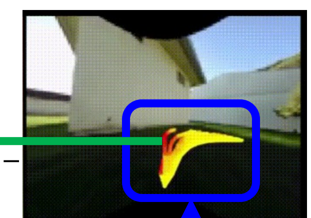
- t- step Obesrvation (RGB Image) ($\mathbf{o}_t$)
- A sequence of H future action (Command)

$$\mathbf{a}_{t:t+H} = (\mathbf{a}_t, \mathbf{a}_{t+1}, ..., \mathbf{a}_{t+H-1})$$

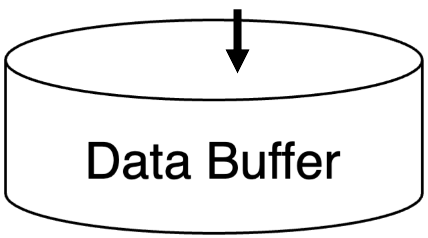
Output

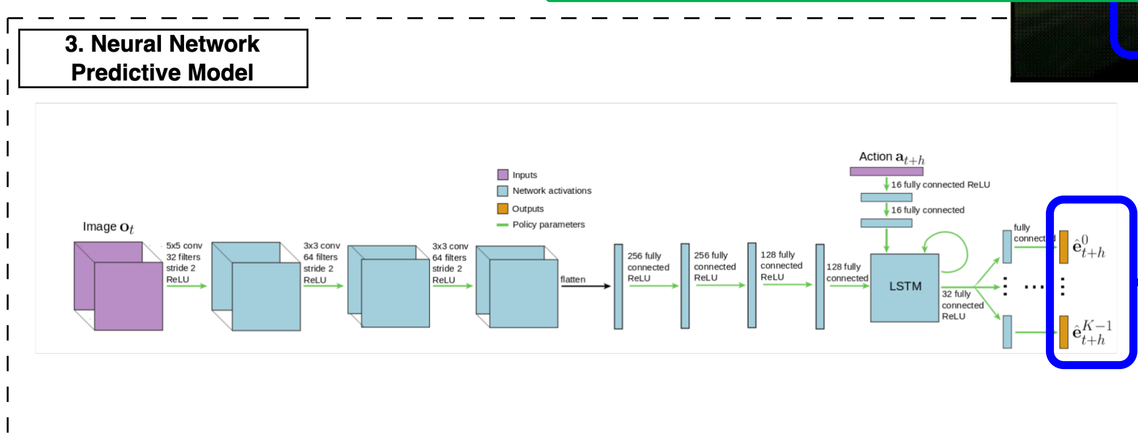
- K different future events

$$\hat{\mathbf{e}}^k_{t:t+H} = (\hat{\mathbf{e}}^k_t, \hat{\mathbf{e}}^k_{t+1}, ..., \hat{\mathbf{e}}^k_{t+H-1})\ \forall k \in \{0, ..., K-1\}$$

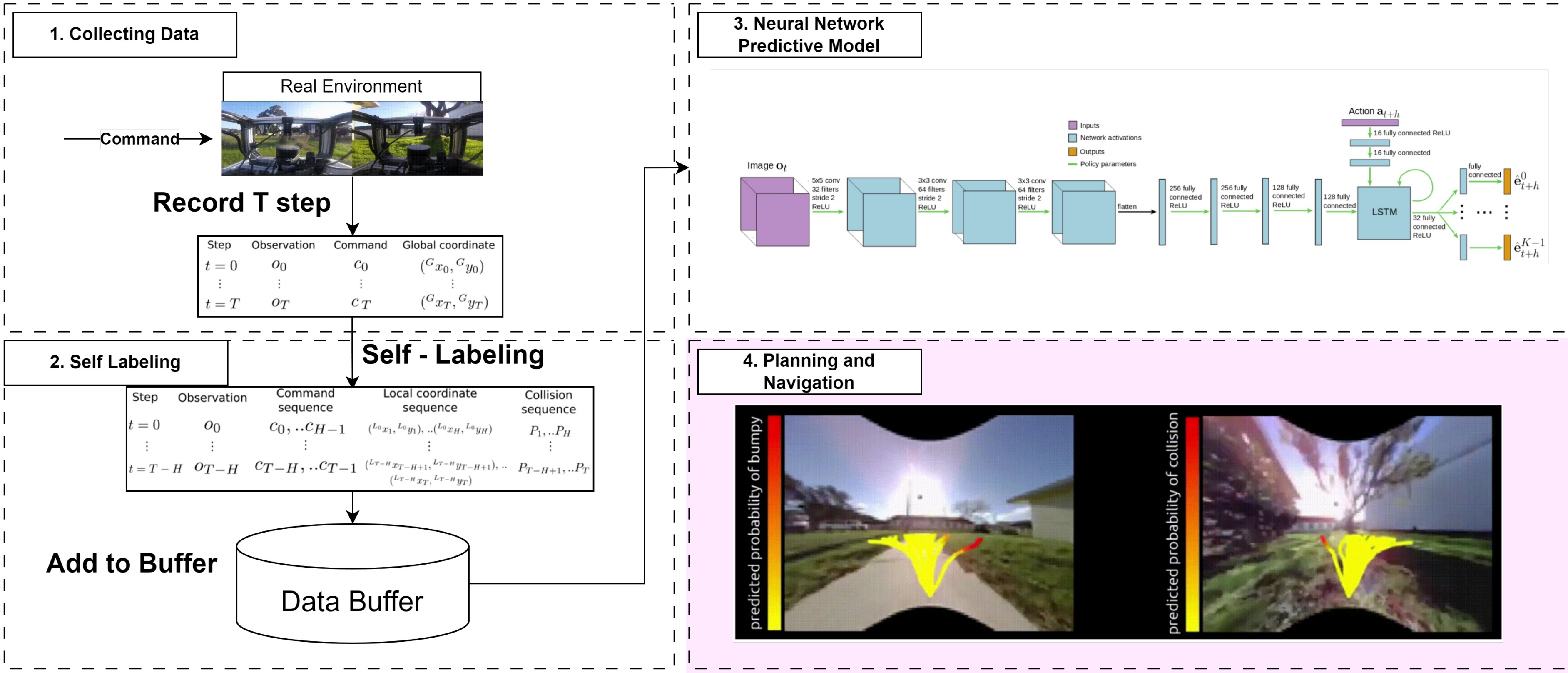


Step	Observation	Command sequence	Local coordinate sequence	Collision sequence
$t=0$	o_0	$c_0, ..c_{H-1}$	$({}^{L_0}x_1, {}^{L_0}y_1), ..({}^{L_0}x_H, {}^{L_0}y_H)$	$P_1, ..P_H$
⋮	⋮	⋮	⋮	⋮
$t=T-H$	o_{T-H}	$c_{T-H}, ..c_{T-1}$	$({}^{L_{T-H}}x_{T-H+1}, {}^{L_{T-H}}y_{T-H+1}), ..({}^{L_{T-H}}x_T, {}^{L_{T-H}}y_T)$	$P_{T-H+1}, ..P_T$





Method – Planning and Navigation



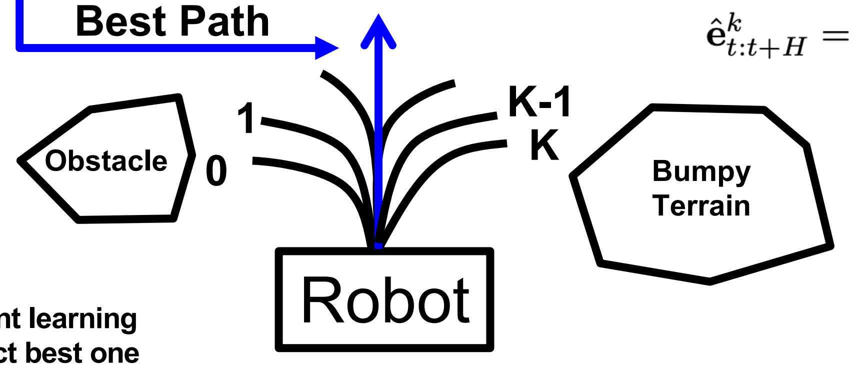
Method – Planning and Navigation

From predicted K different future events, it select best action sequence which maximize task specific rewards

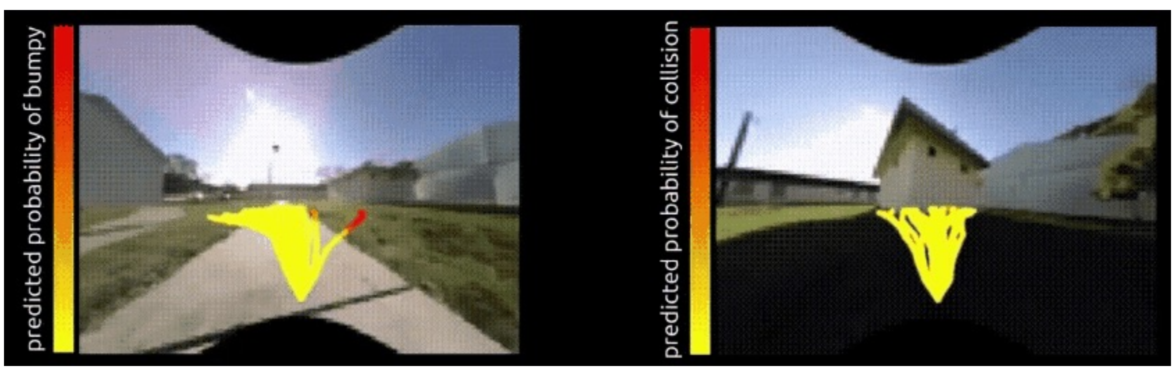
$$\mathbf{a}^*_{t:t+H} = \arg\max_{\mathbf{a}_{t:t+H}} R(f_\theta(\mathbf{o}_t, \mathbf{a}_{t:t+H}))$$

Predict K different future events

$$\hat{\mathbf{e}}^k_{t:t+H} = (\hat{\mathbf{e}}^k_t, \hat{\mathbf{e}}^k_{t+1}, \ldots, \hat{\mathbf{e}}^k_{t+H-1})$$



✓ The reward is not related to reinforcement learning
It is just used to score each path to select best one

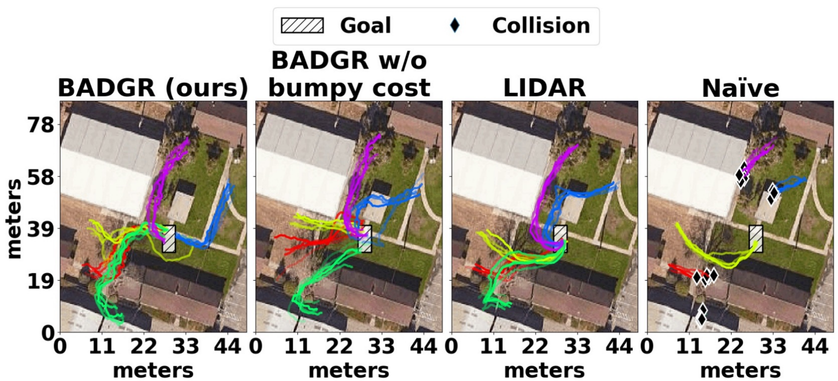


4. Experimental Results

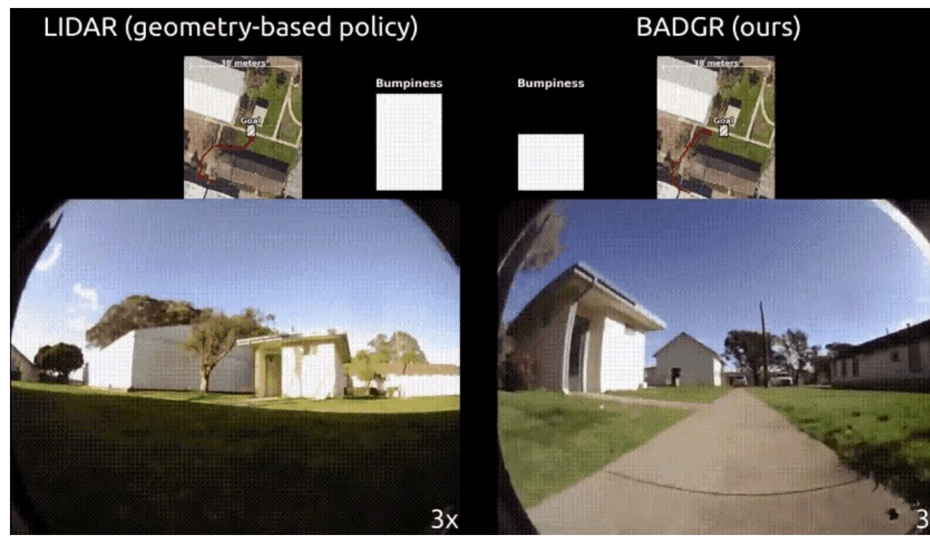
Experimental Results

Task : Reaching a specified goal position in Urban Environment

Method : 5 different start locations
5 runs per each start location
Total 25 trials



Method	Successfully Reached Goal	Avg. Bumpiness
BADGR (ours)	**25/25 (100%)**	**8.7 ± 4.4**
BADGR w/o bumpy cost	25/25 (100%)	15.0 ± 3.4
LIDAR	25/25 (100%)	13.3 ± 2.9
Naïve	5/25 (20%)	N/A



LIDAR : Geomtric View Based Navigation
Naïve : Simply drives straight towards the specified goal

5. Discussion

Discussion

1) BADGR should collect data in real environment,
Can we replace data collecting process from real to simulation ?

2) BADGR only consider two cases 1) collision 2) travel on the bumpy road.
Can we extend BADGR to dense environment ?
1) collision 2) bumpiness 3) density of obstacle

Problems

1) What is the method that isn't accounted for in BADGR ?

   a) Forward Dynamics Model
   b) Human supervision
   c) Model Predictive Path Planning
   d) Self - Labeling

2) How predictive model network can be formulated ?

KAIST