Annual Conference of KIPS 2025

신진학자 워크숍

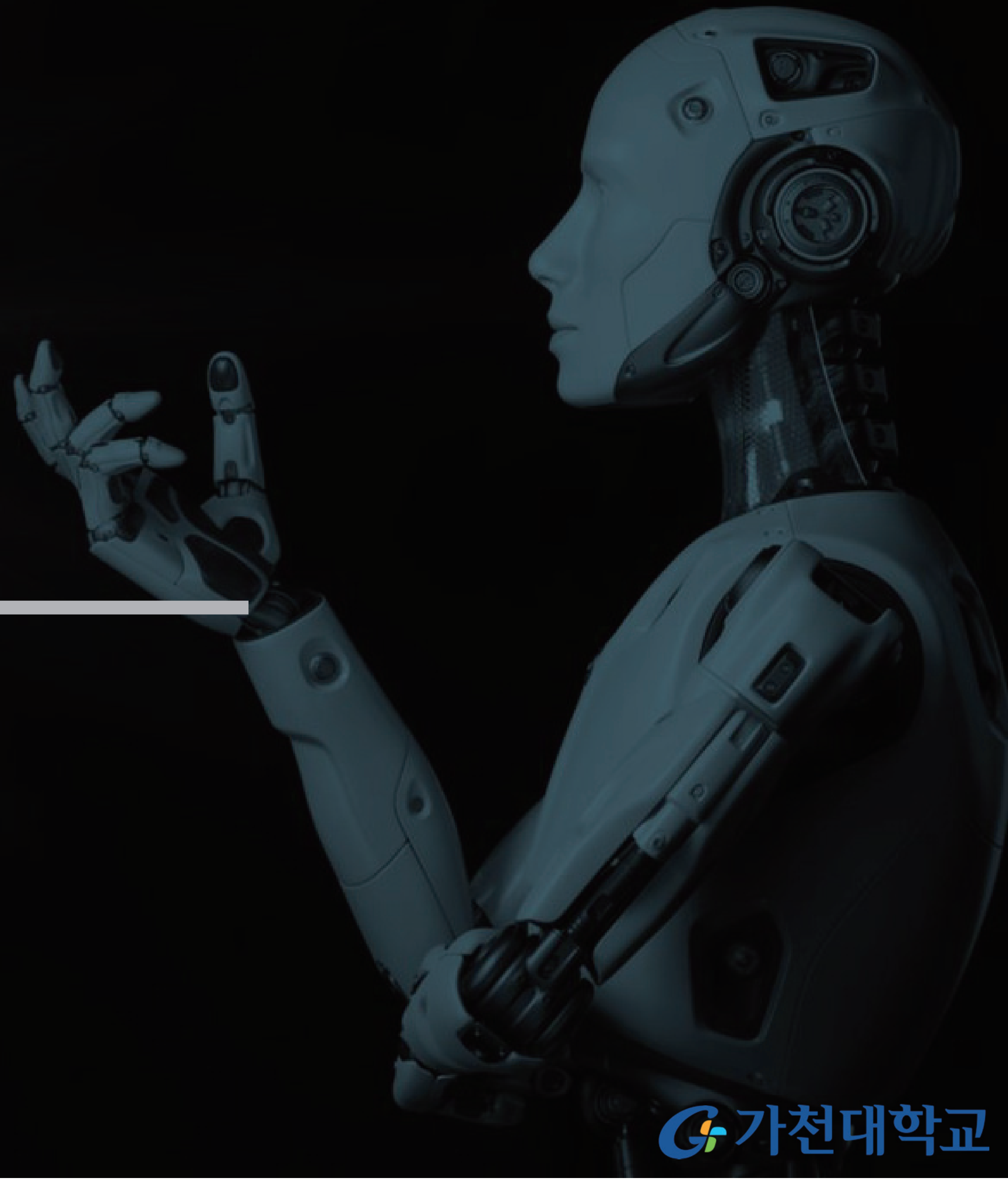
Reinforcement Learning for Real-world Applications

이경현 교수(가천대학교)

Reinforcement Learning for Real-world Applications

Assistant Professor, Gachon Unviersity

Kyunghyun Lee



Introduction

Mechanical Engineering **POSTECH**	Aerospace Engineering **SNU**	UVIFY **CPO (Co-founder)**	Robotics Program **KAIST**	Team Leader **LG AI Research**	Gachon Univ
2013. 03	2015. 02	2018. 08	2023. 08	2025. 03	

Research Goal:
Reinforcement Learning for Real-world Problems

Keywords
- Deep Reinforcement Learning
- Fixed/Rotary-wing Drones
- Computer Vision in Challenging Environment

Classes of Learning Problem

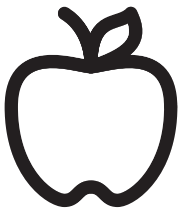
Supervised Learning

Data: (x, y)
x: data, y: label

Goal: Learn function to map
$$x \rightarrow y$$

Example:

This thing is an apple



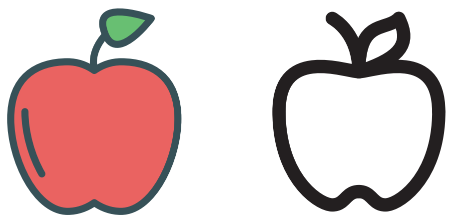
Unsupervised Learning

Data: x
x: data, no labels

Goal: Learn underlying structure

Example:

This thing is like the other thing



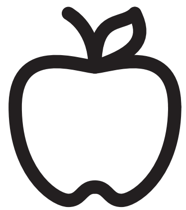
Reinforcement Learning

Data: state-action pairs

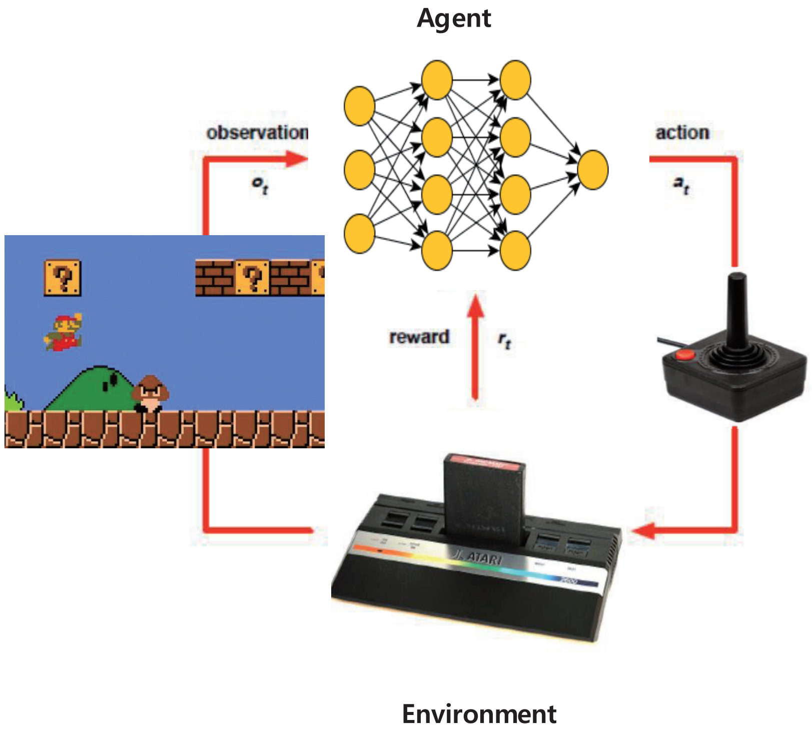
Goal: Maximize future rewards

Example:

Eating this thing is good



Reinforcement Learning (RL) Key Concepts



State s_t, and observation o_t

- $s_t \in S$, state space (discrete or continuous)
- Fully observable: $o_t = s_t$
- Partially observable: $o_t = o(s_t)$
- e.g. raw pixel image, velocity, ...

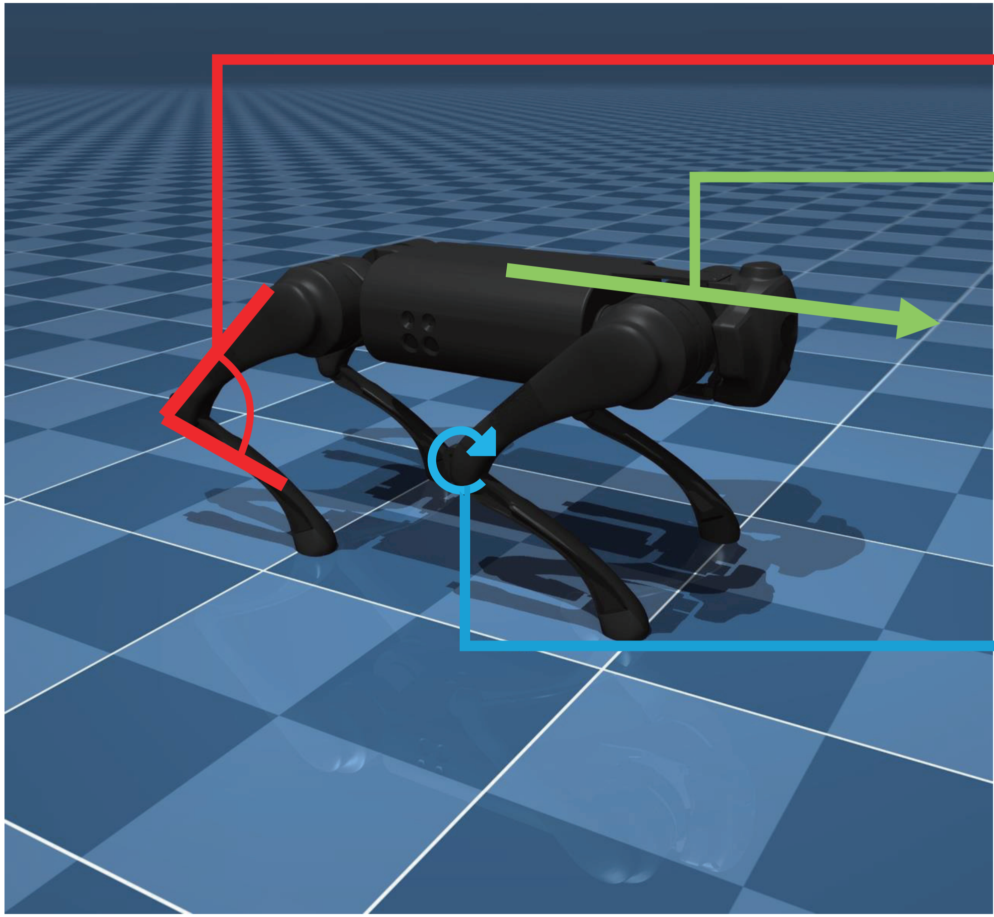
Action a_t

- a_t ∈A, action space (discrete or continuous)
- e.g. going forward/backward, jump, ...

Reward r_t

- $r_t \sim R(s_t, a_t)$, R is reward function.
- e.g. reaching the exit, reaching some goals, penalty for move, …

Reinforcement Learning (RL) Key Concepts



State s_t or observation o_t
: Joint Angle, Angular Velocity, Acceleration

Reward r_t
: Velocity, Pose, Distance…

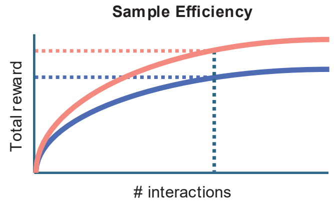
Action a_t
: Motor torque, ….

가천대학교

Challenges in the Real-world RL

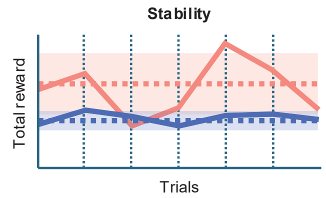
Sample Efficiency Problem

- Number of interactions to learn a good policy
- Data should be collected by trial and error



Stability Problem

- Performance variation by different seeds
- Sensitive to network/environment initialization



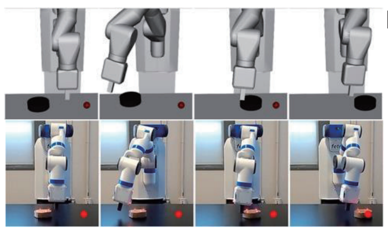
Practical Design Issues

- State, action, reward design sensitivity

[1-3]

Simulation-to-Real Gap

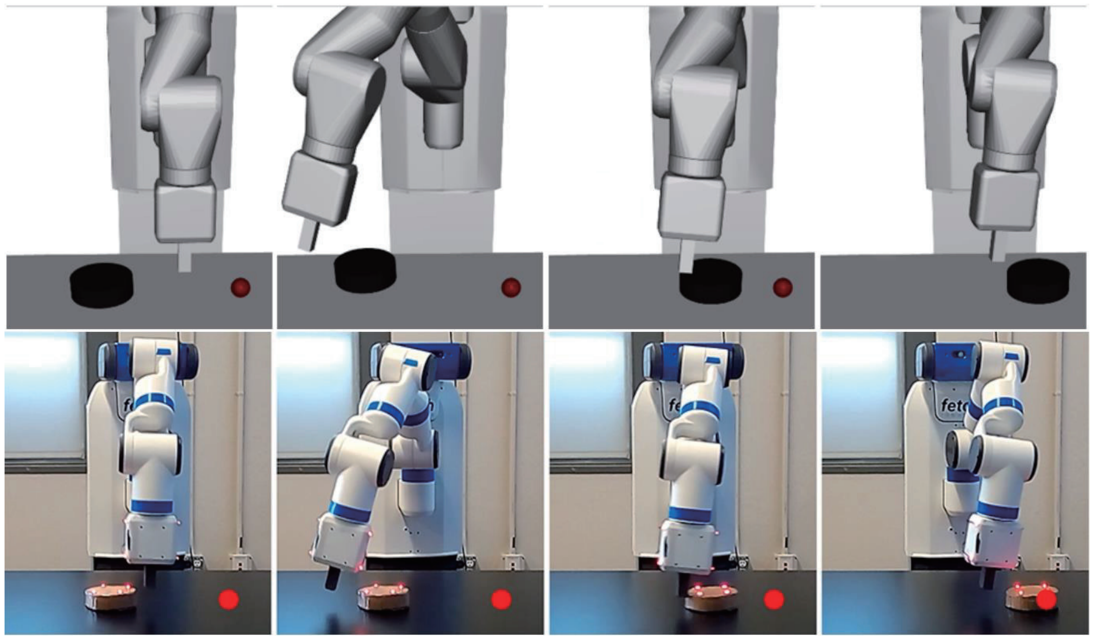
- State, Action distribution are different



[4]

가천대학교

Simulation-to-Real Gap Problem



Simulator

Real-world

Simulation-to-Real Gap Problem

Direct Training

- Pros
  - No Gap
  - True dynamics
  - Sensor/Actuator behavior
- Cons
  - Unsafe
  - Slow, costly
  - Sample inefficient

Domain Randomization

- Pros
  - No real data
  - Robust to unseen variations
  - Fast parallel training
- Cons
  - Unstable learning
  - Sample inefficient
  - Additional tuning for parameter range

System Identification

- Pros
  - Physically interpretable
  - zero-shot deployment
- Cons
  - Complex architecture

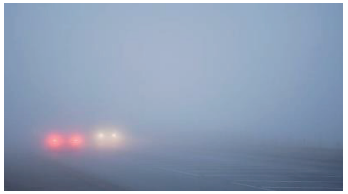
Perception Failure in Real-world Cameras

- Visual applications (Detection, Segmentation, SLAM, …)
  - Assume high quality images

Low Quality Images from Camera

Environmental

Heavy rain



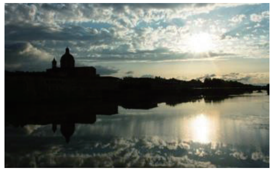
Dense fog

Dust

Image Acquisition

Motion blur

Image noise

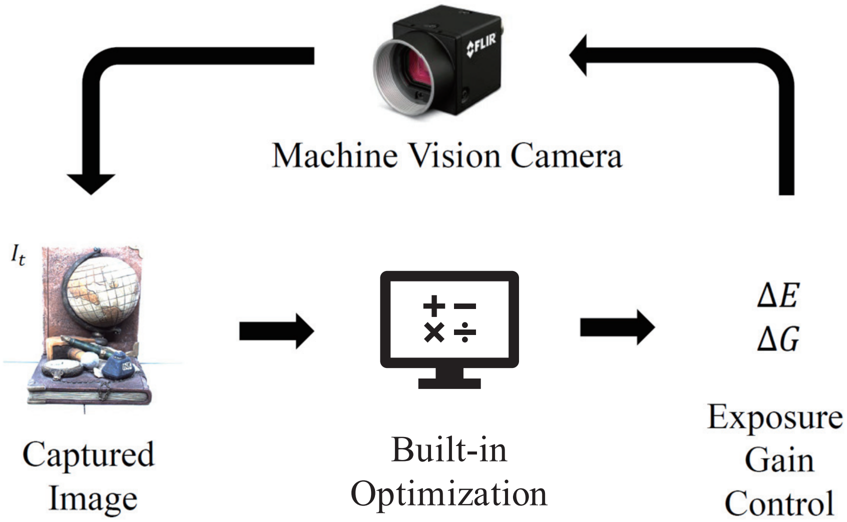


Under exposure

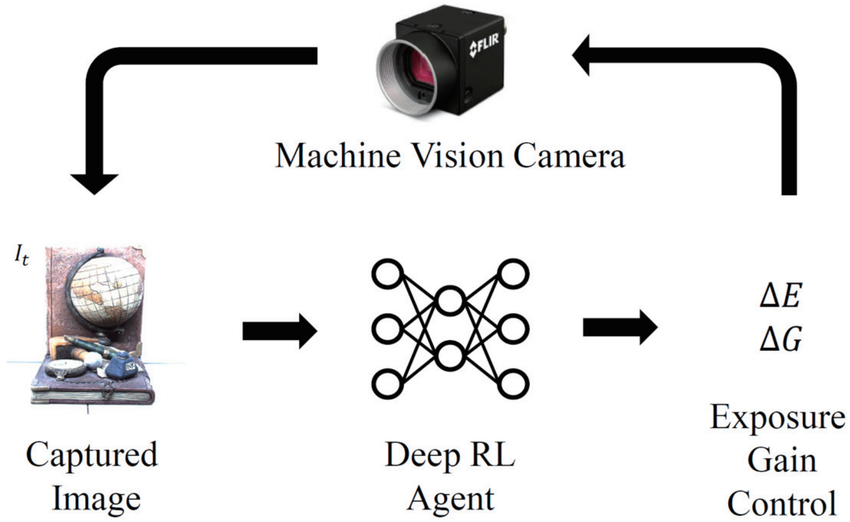
Over exposure

Visual Application

Exposure Control with Reinforcement Learning



Traditional Method



Proposed Method [1]

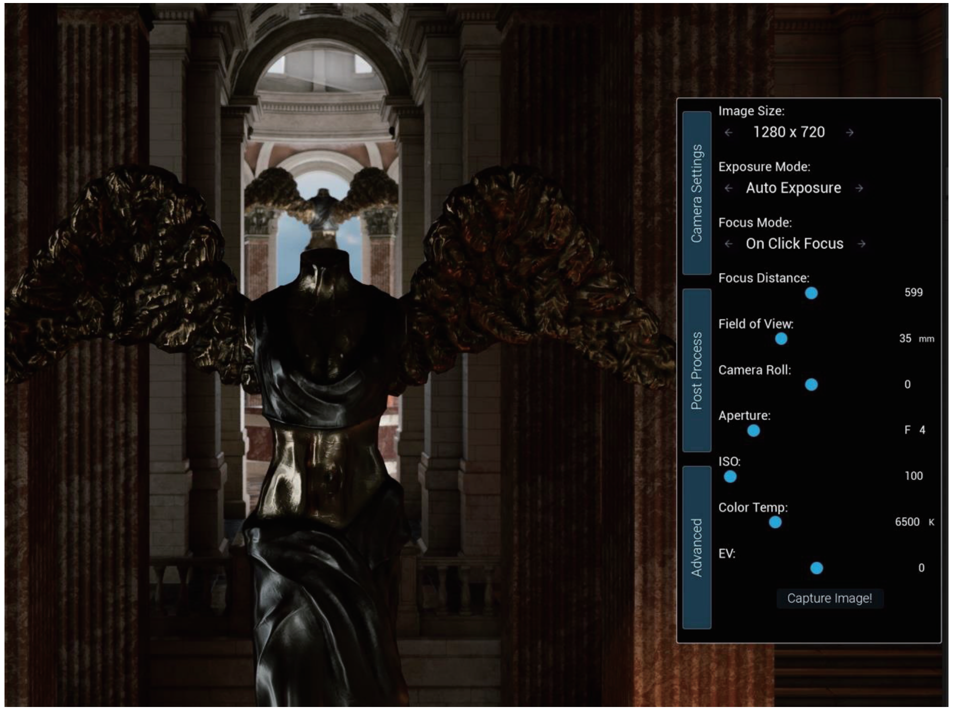
[1] Lee, Kyunghyun, Ukcheol Shin, and Byeong-Uk Lee. "Learning to Control Camera Exposure via Reinforcement Learning." Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2024.

The Darkroom Environment

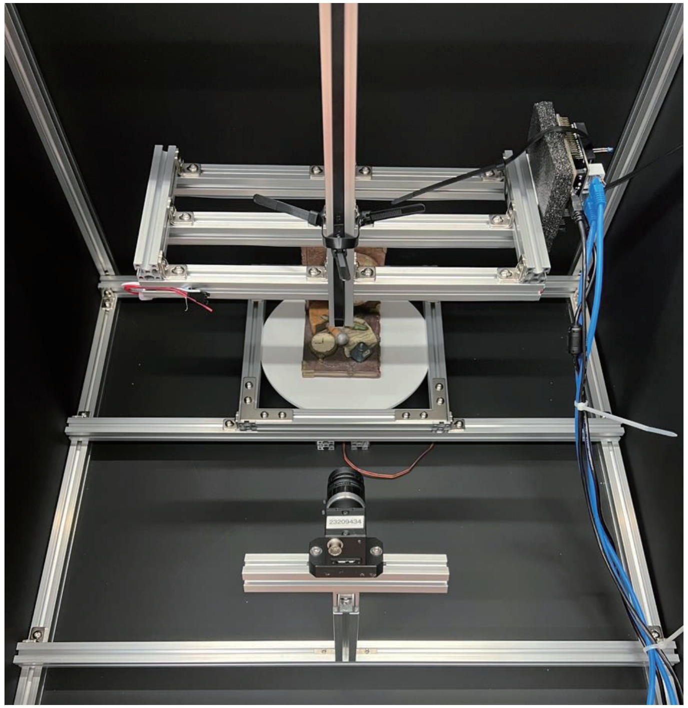
• Simulator is not accurately modeled in extreme conditions

→ Directly train in the light-controlled real-world environment

Simulator Example



The Darkroom Environment



LED bar

Light Controller

Machine Vision Camera

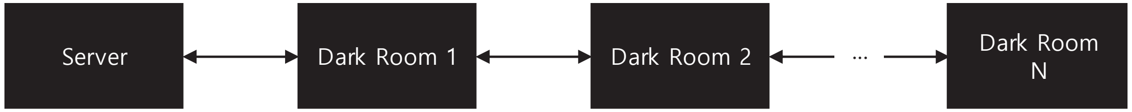
Target Object

[1] Lee, Kyunghyun, Ukcheol Shin, and Byeong-Uk Lee. "Learning to Control Camera Exposure via Reinforcement Learning." Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2024.

Parallel Environments

- Build 4 Parallel “Dark Rooms”

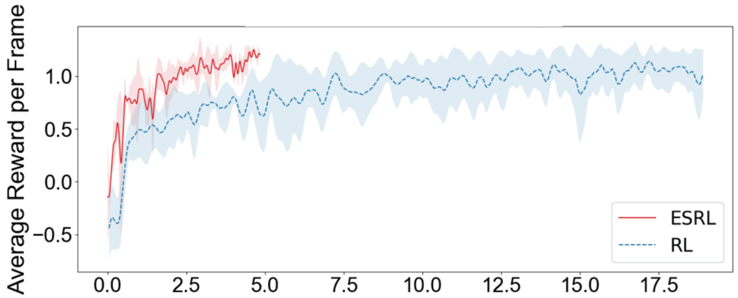
- Scalable Architecture
  - Chain architecture
  - Auto detect posterior dark room



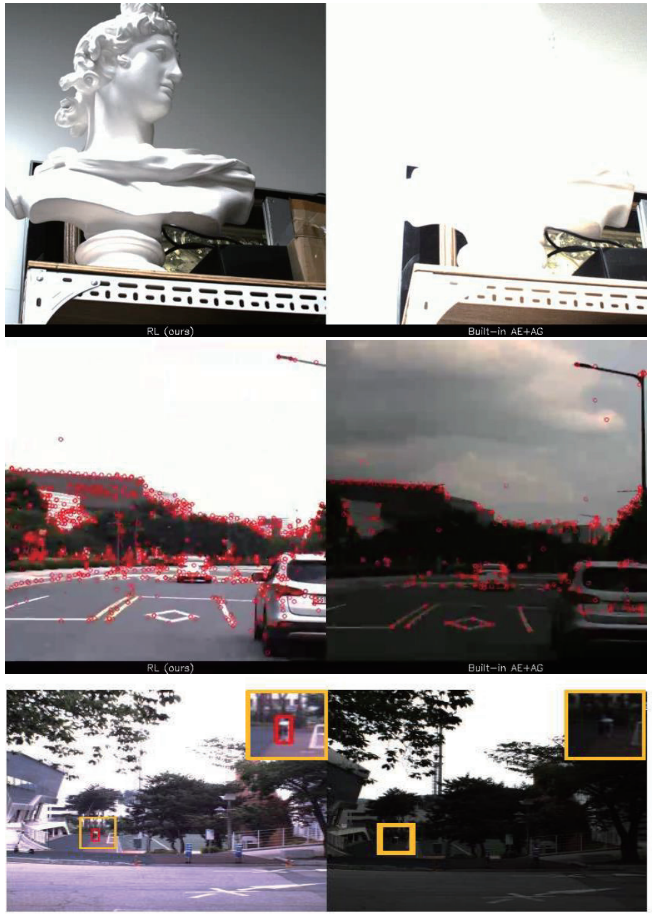
- Aggregate room information
- Broadcast periodically

[1] Lee, Kyunghyun, Ukcheol Shin, and Byeong-Uk Lee. "Learning to Control Camera Exposure via Reinforcement Learning." Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2024.

Experimental Result



- 500k timesteps
- Training time required:
  - RL: 18.1 h
  - ESRL: 4.9h
  - 3.69x speed up with 4 parallel envs
- Final performance:
  - RL: 1.37
  - ESRL: 1.44
  - 5% performance enhanced



Ours **Built-in AE**

- Dramatic Lighting Change
  - Rapid convergence (in 5 frames)
- ORB Feature Detection
  - 52% more features
- No additional Training
  - Zero-shot generalization
- Detection from YOLO-v5
  - Capture object 4 frames earlier

[1] Lee, Kyunghyun, Ukcheol Shin, and Byeong-Uk Lee. "Learning to Control Camera Exposure via Reinforcement Learning." Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2024.

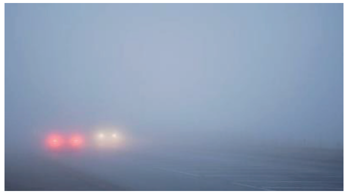
Perception Failure in Real-world Cameras

- Visual applications (Detection, Segmentation, SLAM, …)
  - Assume high quality images

Low Quality Images from Camera

Environmental


Heavy rain

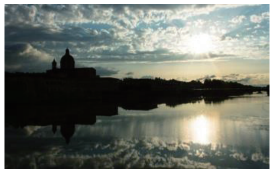

Dense fog


Dust

Image Acquisition


Motion blur


Image noise


Under exposure


Over exposure

Visual Application

가천대학교

Multi-Agent Manipulation via Locomotion using Hierarchical Sim2Real

Goal

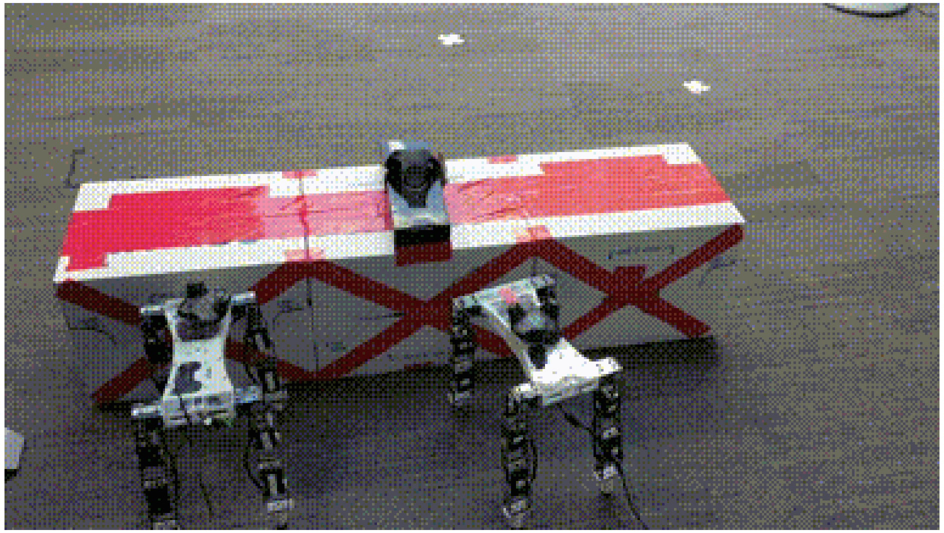
- Move object with cooperative robots in real-world

Hierarchical RL

- High-level policy generates the goal
- Low-level policy generates actions

Domain Randomization

- Physical property randomization



O. Nachum, et. al., “Multi-Agent Manipulation via Locomotion using Hierarchical Sim2Real,” CoRL, 2020, Google AI

Multi-Agent Manipulation via Locomotion using Hierarchical Sim2Real

State

- 3 motors on each leg, total 4 legs = 12 dof
- Total 30 dimensions
- Joint angles(12) + joint angular velocities(12) + global position(3) + global orientation(3)

Action

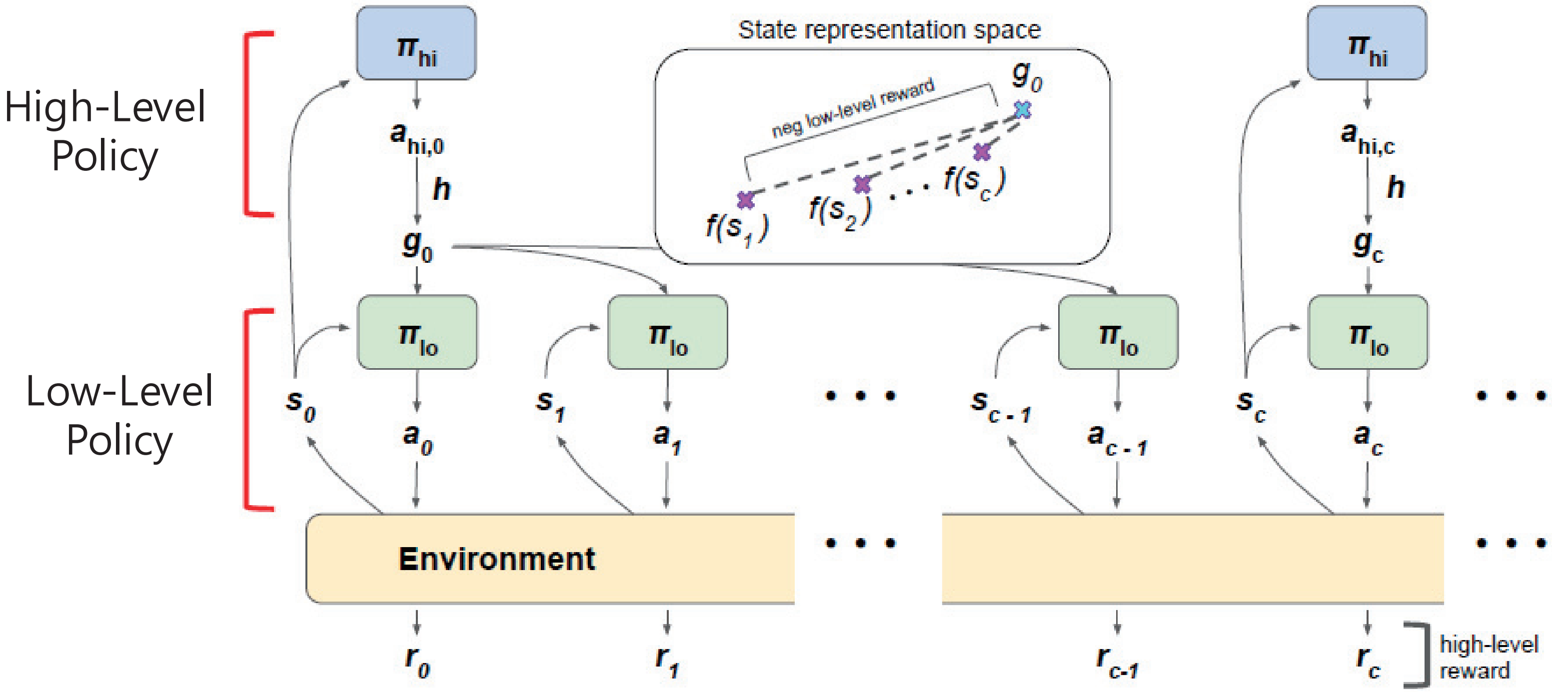
- continuous actions for each motors
- Total 12 dimensions

Coordinate setting for “n” robots

- Multiply state and action by “n”
- Ex) 2 robots: 60 states and 24 actions

O. Nachum, et. al., “Multi-Agent Manipulation via Locomotion using Hierarchical Sim2Real,” CoRL, 2020, Google AI

Multi-Agent Manipulation via Locomotion using Hierarchical Sim2Real

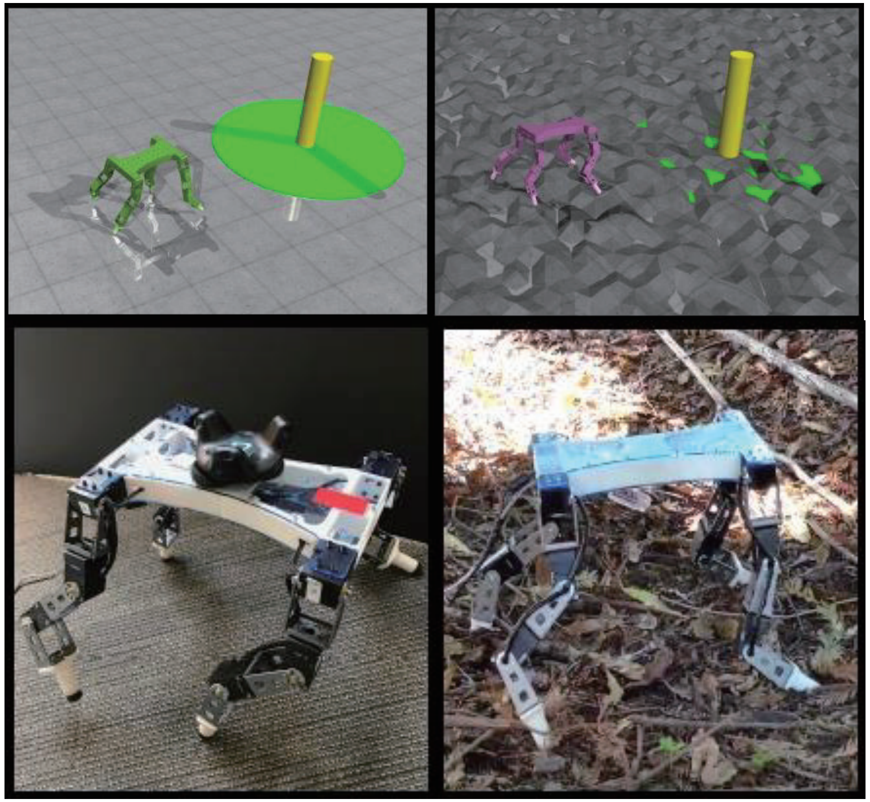


O. Nachum, et. al., "Multi-Agent Manipulation via Locomotion using Hierarchical Sim2Real," CoRL, 2020, Google AI

Multi-Agent Manipulation via Locomotion using Hierarchical Sim2Real

Simulation-to-Real

- Damping ratio
$$\zeta \sim uniform(0.9, 1.1)$$
- Friction
$$\mu \sim uniform(0.001, 0.005)$$
- Mass
$$m \sim uniform(1.6, 2.0)$$
- Actuator gain
$$g_a \sim uniform(4, 6)$$
- Field height
$$h_{field} \sim uniform(0, 0.05)$$



O. Nachum, et. al., “Multi-Agent Manipulation via Locomotion using Hierarchical Sim2Real,” CoRL, 2020, Google AI

가천대학교

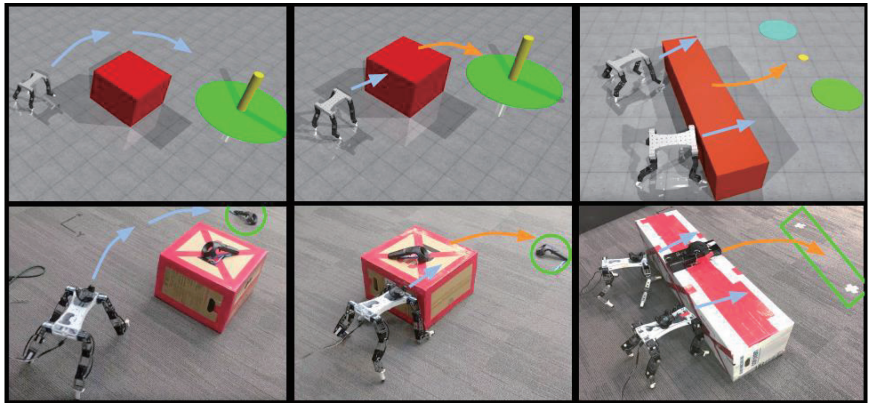
Multi-Agent Manipulation via Locomotion using Hierarchical Sim2Real

Task Specific Reward

- Avoid
    1. Negative distance to the target
    2. A bonus for being within 0.5m of the target location
    3. A penalty for being within 0.5m of the block
    4. Upright bonus

- Push
    1. Negative distance to the target
    2. Negative distance to the block
    3. A bonus for being within 0.5m of the target location
    4. Upright bonus

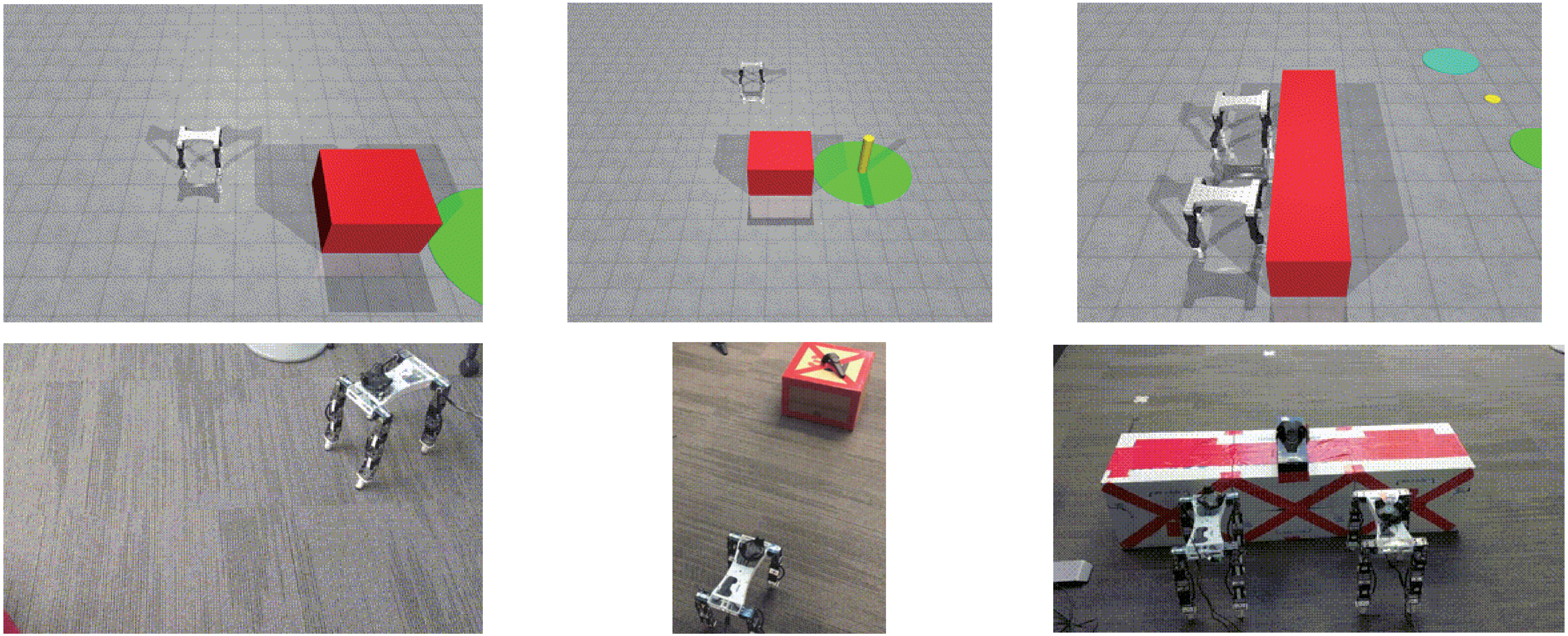
- Coordinate
    1. A sum of negative distance of each end of the block to the desired target location
    2. A bonus for when both ends of the block are within 0.3m of the desired target location
    3. Upright bonus



Avoid Push Coordinate

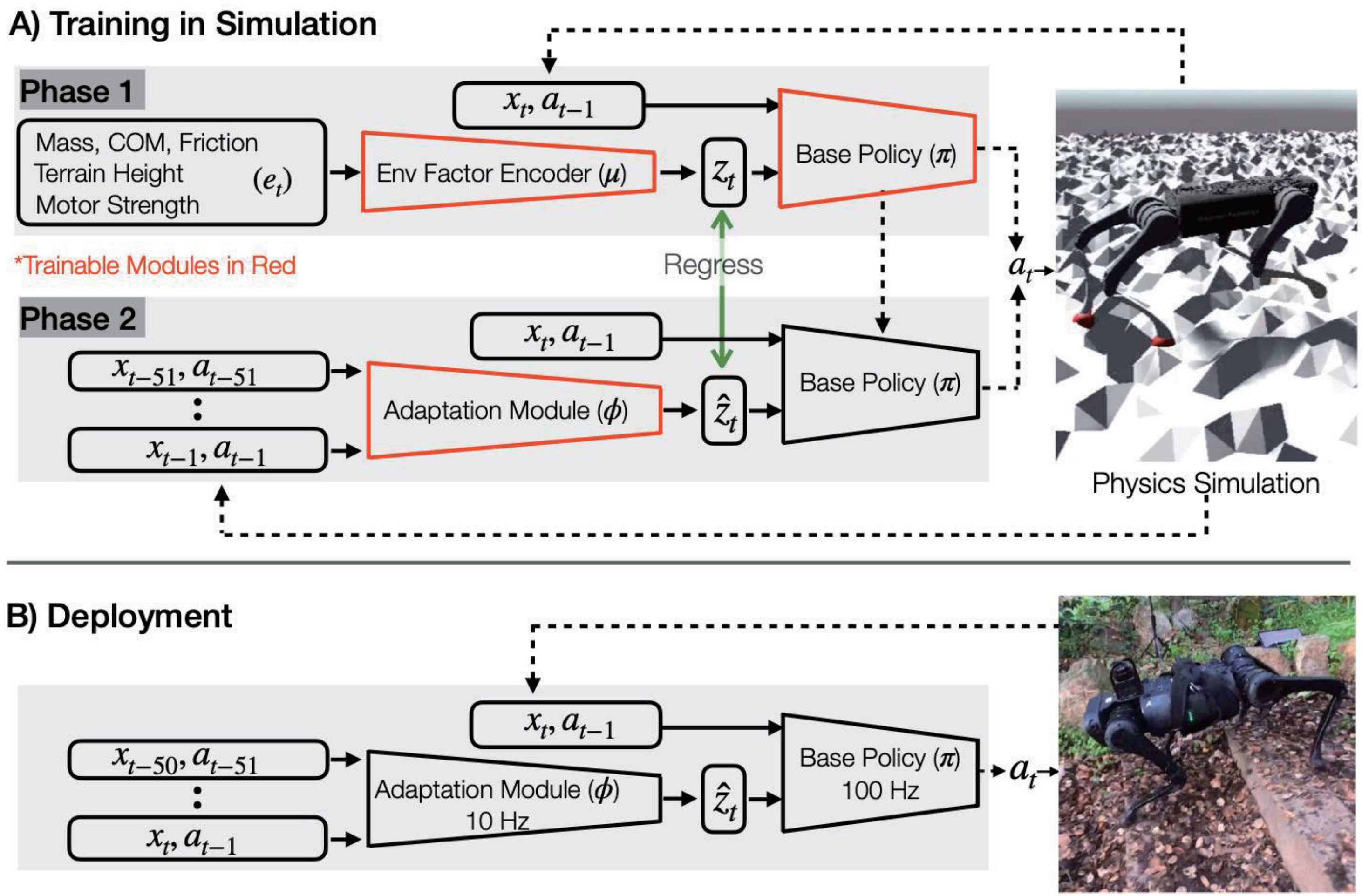
O. Nachum, et. al., "Multi-Agent Manipulation via Locomotion using Hierarchical Sim2Real," CoRL, 2020, Google AI

Multi-Agent Manipulation via Locomotion using Hierarchical Sim2Real



O. Nachum, et. al., "Multi-Agent Manipulation via Locomotion using Hierarchical Sim2Real," CoRL, 2020, Google AI

RMA: Rapid Motor Adaptation for Legged Robots



Kumar, Ashish, et al. "Rma: Rapid motor adaptation for legged robots." *RSS* (2021).

RMA: Rapid Motor Adaptation for Legged Robots

Conclusion

- Each approach has trade-offs
  - Real-world training: highest fidelity, but costly and unsafe
  - Domain Randomization: Strong robustness, but unstable
  - System Identification: Efficient zero-shot transfer, but complex setup

- Combining these approaches yields the most practical real applications

Q&A