DEVIEW 2019

# Deep Reinforcement Learning of Navigation for AROUND Robots

최진영
NAVER LABS / Robotics
NAVER

# CONTENTS

DEVIEW 2019

1. Introduction
2. Deep Reinforcement Learning
3. Applying Deep Reinforcement Learning for AROUND
4. Future Research Directions

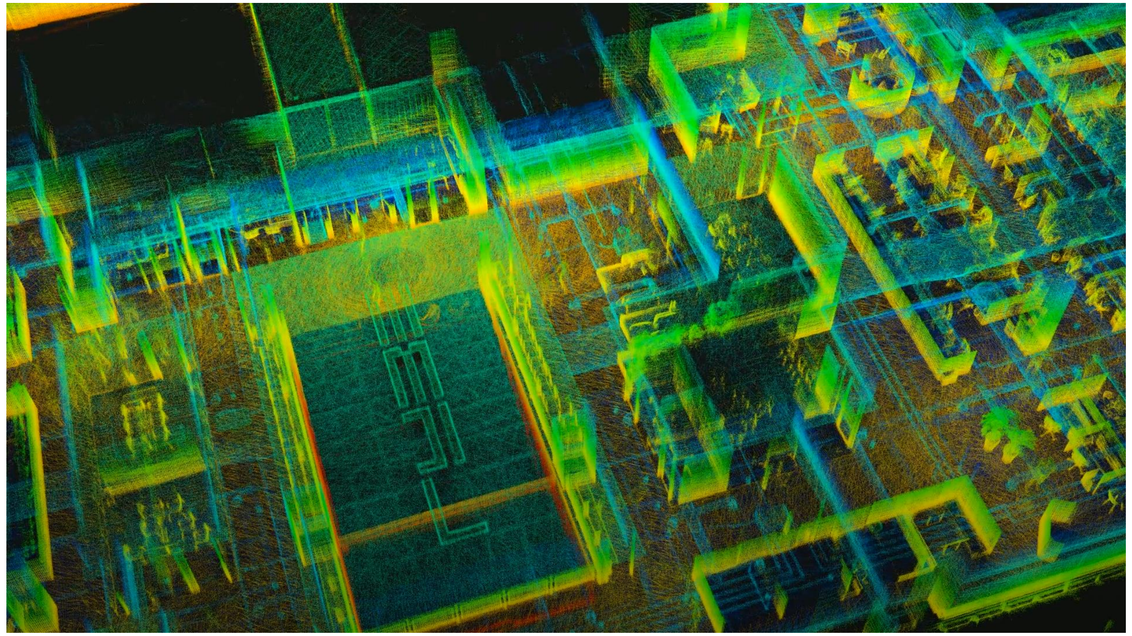
# 1. Introduction

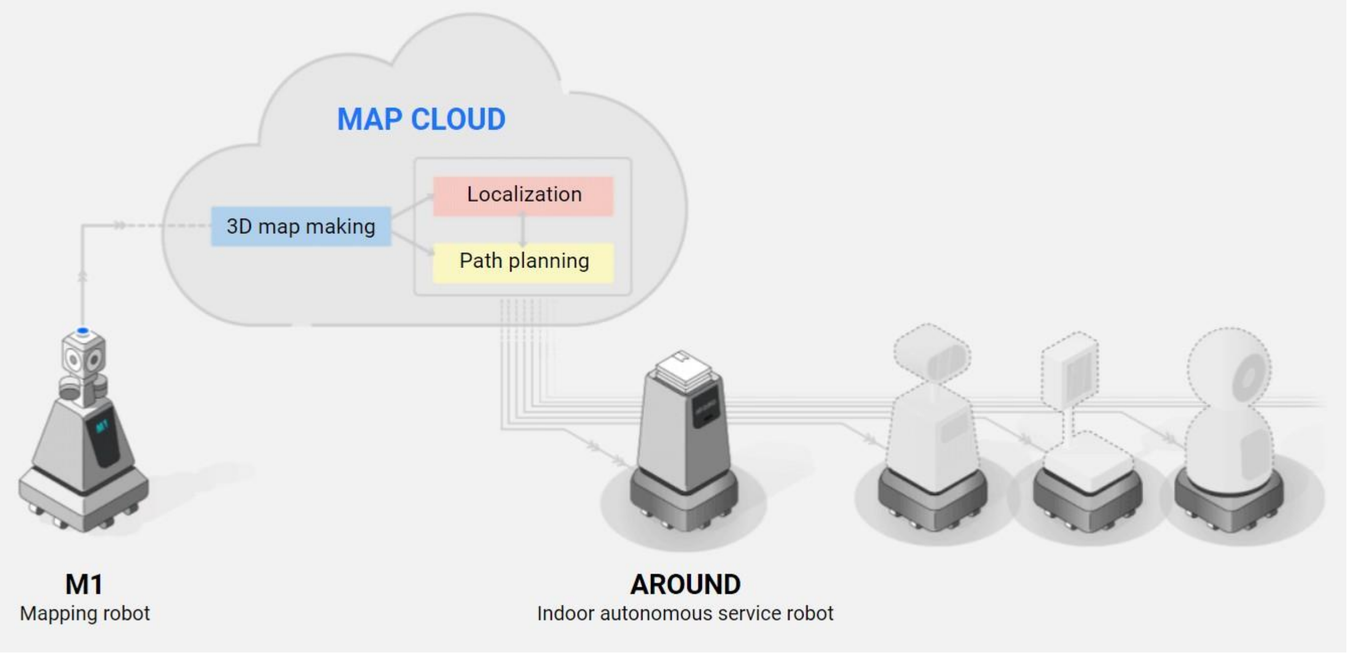
# 1.1 AROUND mobile robots

DEVIEW 2019

## Mobile service platform

- Mapping과 localization을 별도의 로봇과 Cloud에 의존
- 저렴한 센서와 프로세서를 사용한 자율 주행





# 1.1 AROUND mobile robots

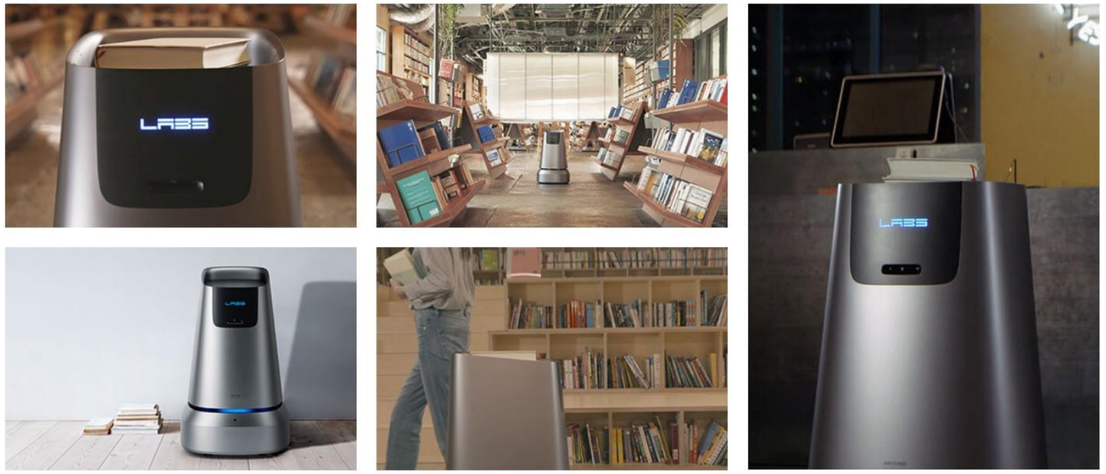
DEVIEW 2019

## Around B (~2017)

- 프로토타입
- 책 회수 (부산 Yes24 매장에서 테스트)

## Around G (2018)

- 가이드 로봇
  - 자연스럽고 끊김 없는 움직임
  - 복잡한 환경에서도 저렴한 센서 만으로 주행
  - 사람 (움직이는 장애물)이 많은 환경에서의 주행
- 원가 절감
  - No Lidar (50만원 ~ 700만원)
  - RGB-D 카메라 (10만원 대)
  - TOF 센서 (1만원 대)

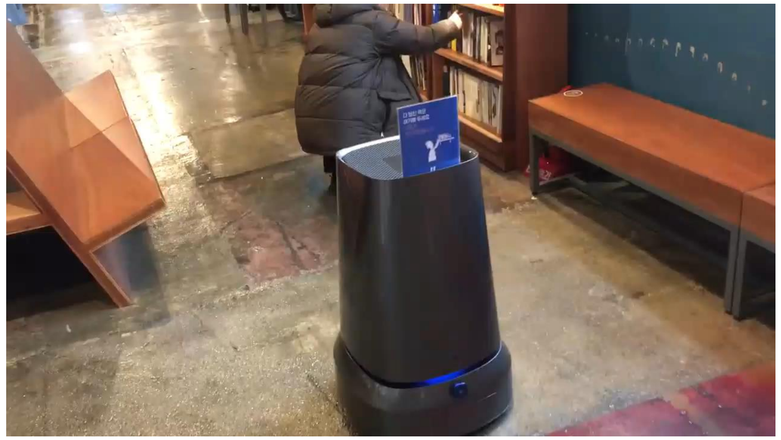


# 1.2 Mobile robot navigation

DEVIEW 2019

## Current paradigm & Problems

- Occupancy grid map based navigation
  - Static map & Obstacle map
  - Localization & planning on map
- Narrow FOV problem
  - 측위, 센서 값 오차 누적
  - RGB-D 카메라, TOF 센서의 낮은 시야 각
  - 잘못 업데이트 된 지도를 보정할 기회 X
- Planning in complex environment
  - 잘못 업데이트 된 지도 = 잘못된 planning
  - 단순한 알고리즘의 성능으로 만족스러운 안내 불가능

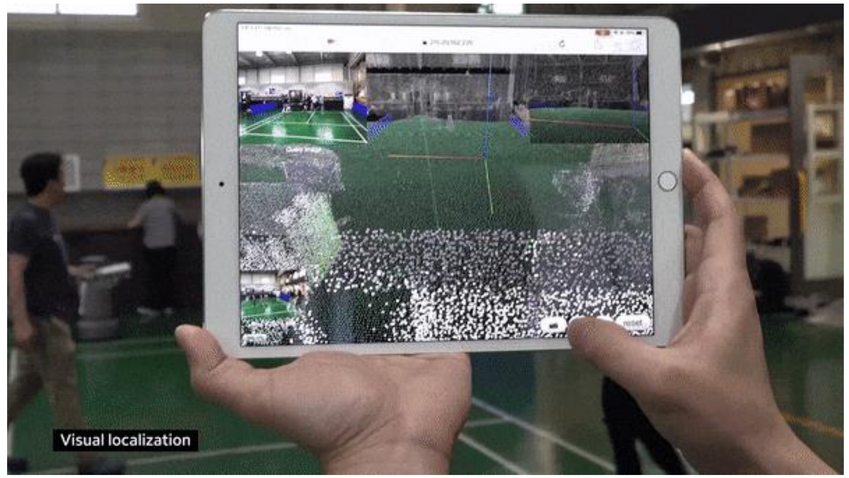


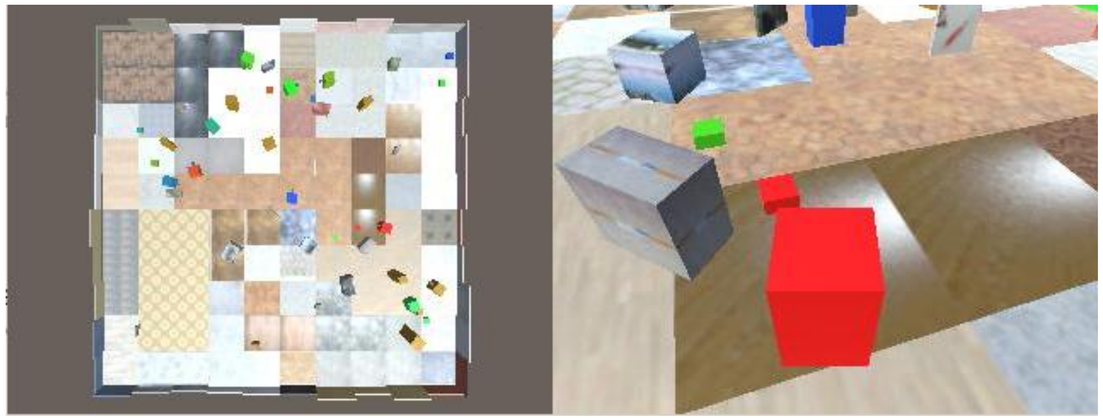
# 1.2 Mobile robot navigation



## Solutions – Navigation without occupancy grid and planning

- Visual Localization
  - RGB 이미지 만으로 로봇의 위치 측위
- Deep reinforcement learning for collision avoidance
  - 많은 task에서 사람보다 뛰어난 성능
  - GPU를 사용한 빠른 inference



# 2. Deep Reinforcement Learning

# 2.1 Reinforcement Learning

DEVIEW 2019

## Reinforcement Learning 이란?

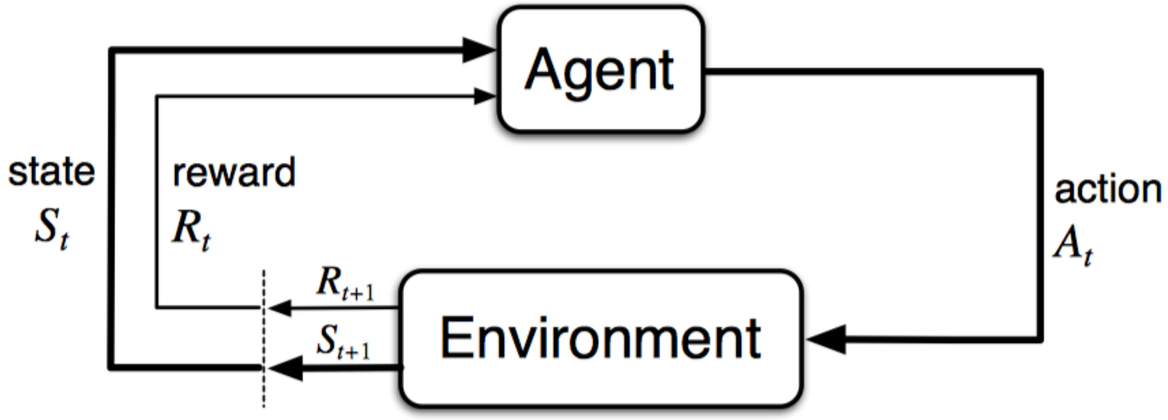
출처 : EBS (https://www.youtube.com/watch?v=kV5dL2Ua96k)

# 2.1 Reinforcement Learning



## Reinforcement Learning 이란?

- 환경과 상호작용 하는 Agent
- State , Action, Reward, Policy
- 누적 reward를 최대화하는 policy $\pi(action|state)$ 를 학습




https://towardsdatascience.com/policy-gradients-in-a-nutshell-8b72f9743c5d

# 2.1 Reinforcement Learning



## Reinforcement Learning 이란?

- 주로 Value function learning 또는 Policy gradient 방식을 사용
- Actor-critic, Deterministic policy gradient 등 여러 알고리즘 존재

| Value function learning | Policy gradient |
|---|---|
| $Q_{st,at} = Q_{st,at} + \alpha * (r_t + \gamma * \max Q(st+1, a) - Q_{st,at})$<br>Learning rate: $\alpha$; Reward: $r_t$; Discount factor: $\gamma$; New value: $Q_{st,at}$; Current value: $Q_{st,at}$; Future value estimate: $\max Q(st+1, a)$ | $Policy\ gradient: E_\pi[\nabla_\theta(log\pi(s,a,\theta))R(\tau)]$<br>Policy function: $\pi(s,a,\theta)$; Score function: $R(\tau)$<br>$Update\ rule: \Delta\theta = \alpha * \nabla_\theta(log\pi(s,a,\theta))R(\tau)$<br>Change in parameters: $\Delta\theta$; Learning rate: $\alpha$ |

https://developer.ibm.com/articles/cc-reinforcement-learning-train-software-agent/

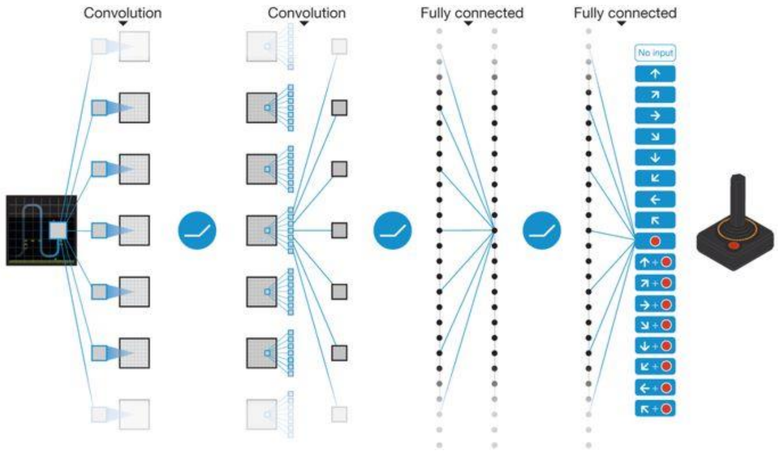
https://www.freecodecamp.org/news/an-introduction-to-policy-gradients-with-cartpole-and-doom-495b5ef2207f/

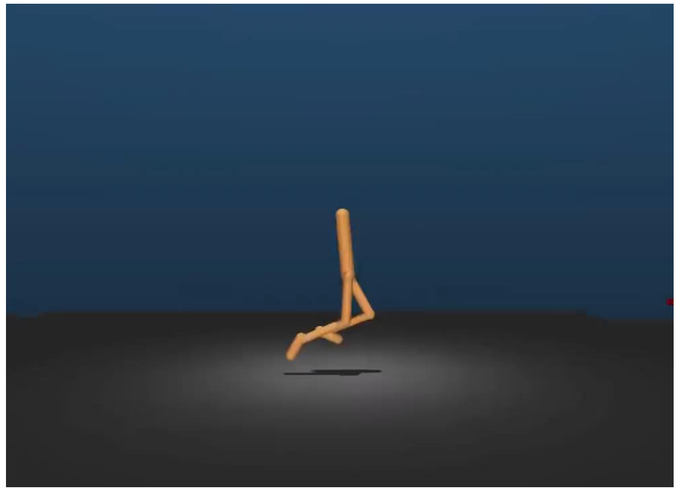
# 2.2 Deep Reinforcement Learning

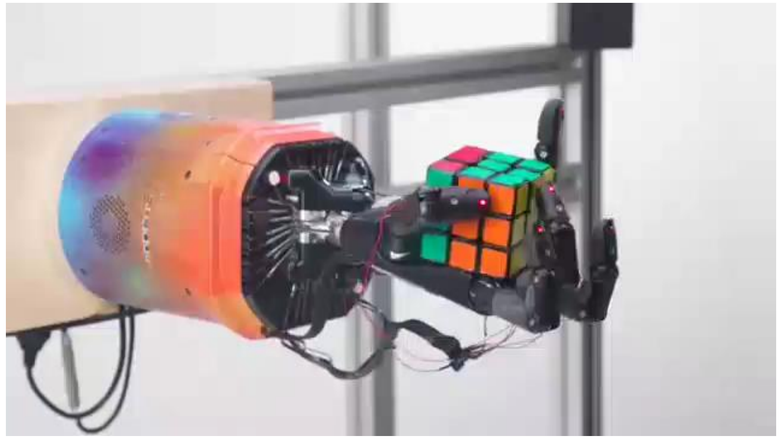
DEVIEW 2019

## ‘Deep’ **Reinforcement Learning** 이란?

- Value function이나 policy를 deep neural network로 학습
- 여러 도메인에서 사람, 사람이 짠 알고리즘 보다 뛰어난 성능
- 게임, 로보틱스 등의 분야에서 활발히 연구 중





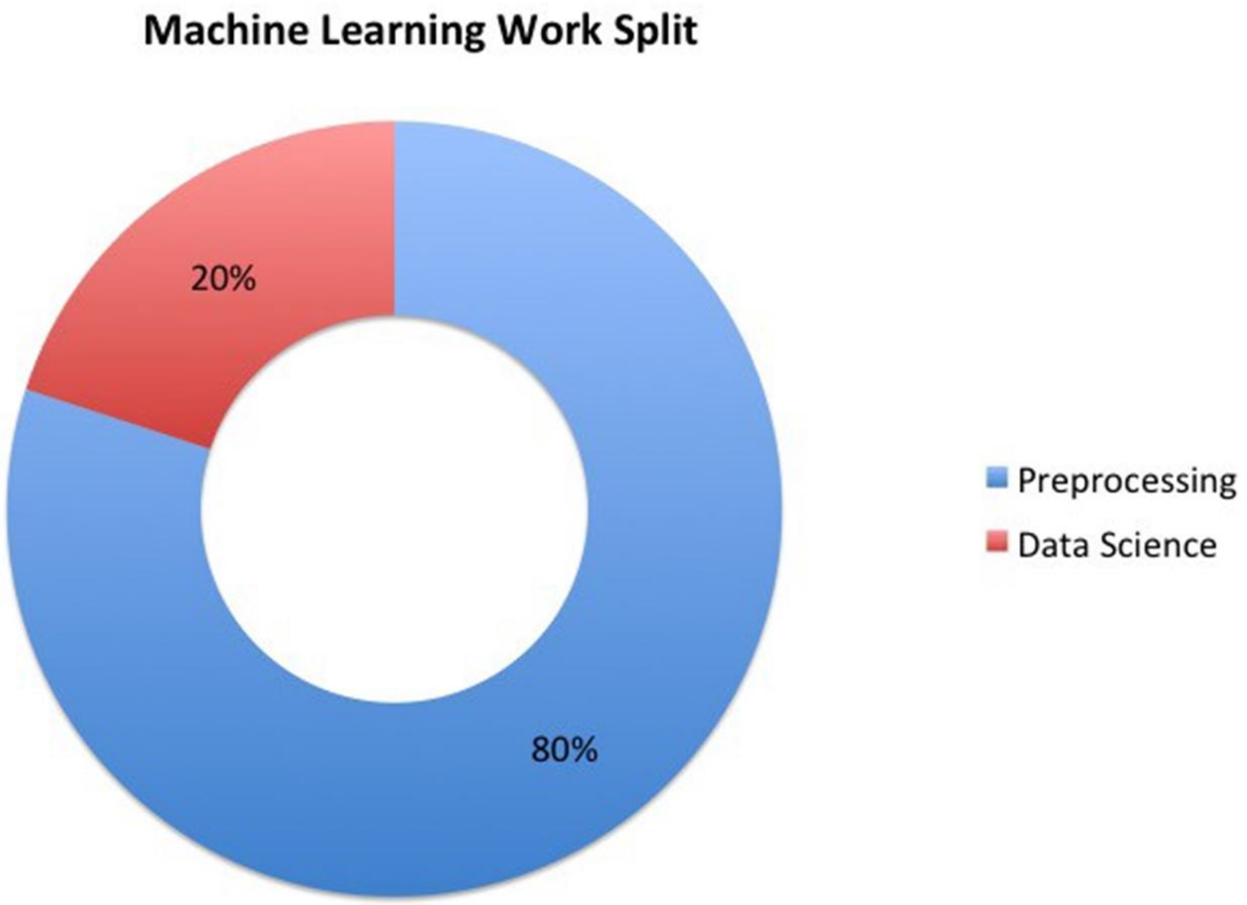
Mnih, Volodymyr, et al. "Human-level control through deep reinforcement learning." *Nature* 518.7540 (2015): 529.
https://www.youtube.com/watch?v=hx_bgoTF7bs , https://www.youtube.com/watch?v=kVmp0uGtShk

# 3. 강화학습 실무에 적용하기

# 3.1 Deep learning in practice

DEVIEW 2019

## 80/20 rule

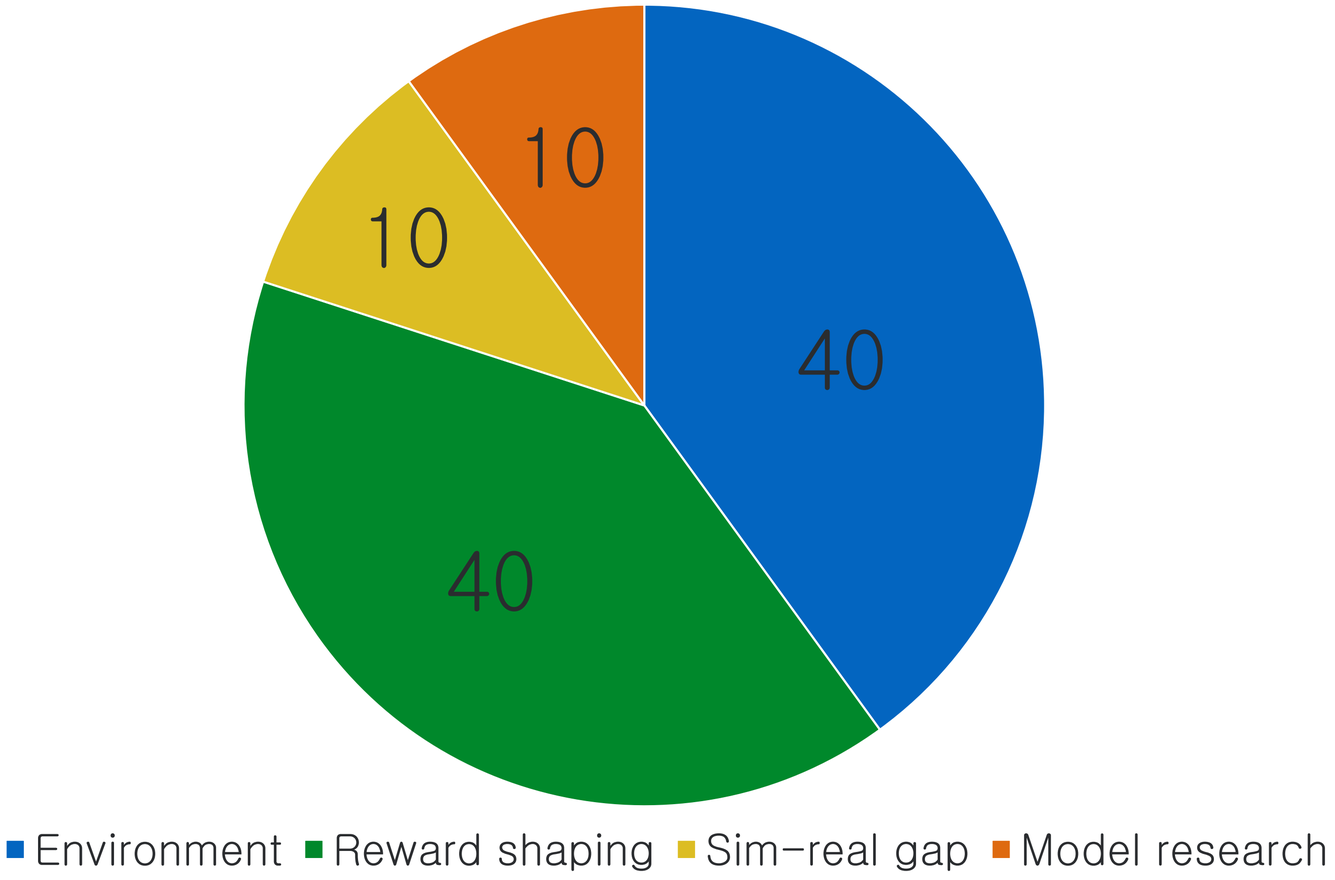



https://towardsdatascience.com/top-5-mistakes-of-greenhorn-data-scientists-90fa26201d51

# 3.1 Deep RL in practice



## 40/40/10/10 rule

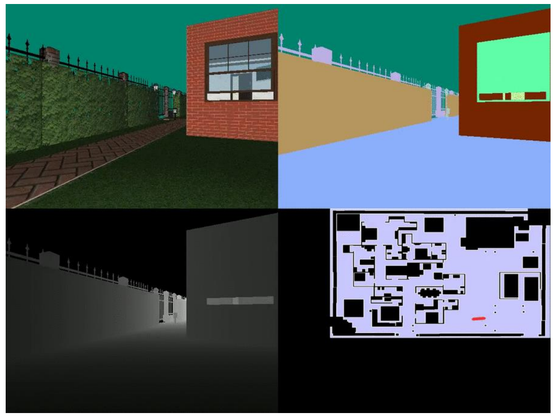


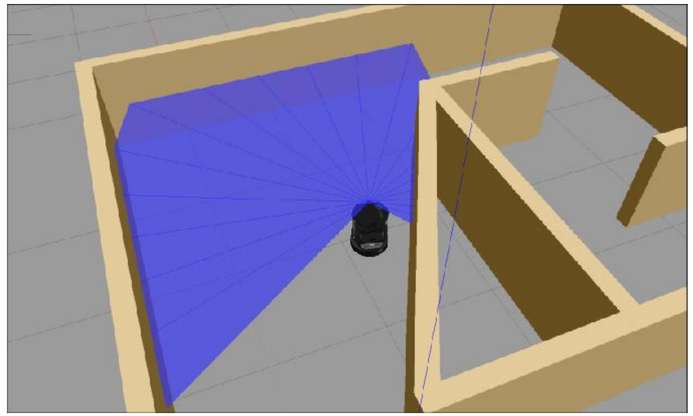
# 3.2 학습 환경 구축하기

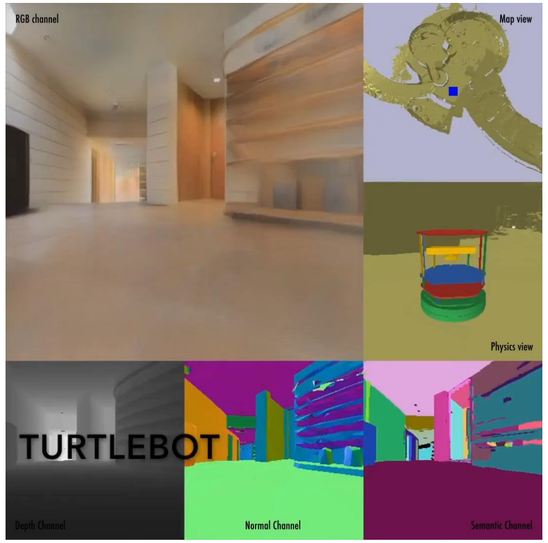
DEVIEW 2019

## Indoor navigation simulator

- Throughput & Parallelizable : 학습에 Million 단위의 sample 필요
- Multi-agent (moving obstacles)
- Diverse environment : structured, unstructured, heights
- Custom robot model for AROUND : TOF센서 등
- 모든 조건을 만족하는 시뮬레이터를 찾기 힘듦








http://gibsonenv.stanford.edu/

Zamora, Iker et al. "Extending the OpenAI Gym for robotics: a toolkit for reinforcement learning using ROS and Gazebo." *ArXiv* abs/1608.05742 (2016): n. pag.

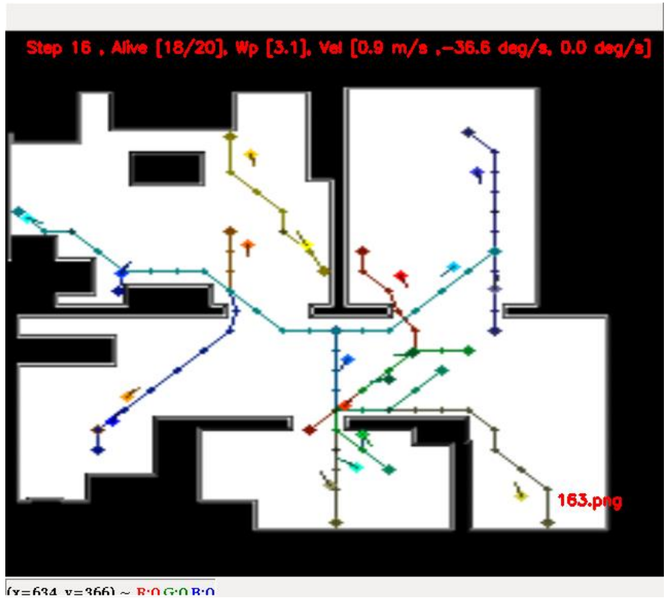
https://github.com/facebookresearch/House3D

# 3.2 학습 환경 구축하기

DEVIEW 2019

## Indoor navigation simulator

- Prototype 시뮬레이터를 자체 제작
- Based on Numpy and OpenCV
- 2D projection of SUNCG dataset ([https://sscnet.cs.princeton.edu/](https://sscnet.cs.princeton.edu/))

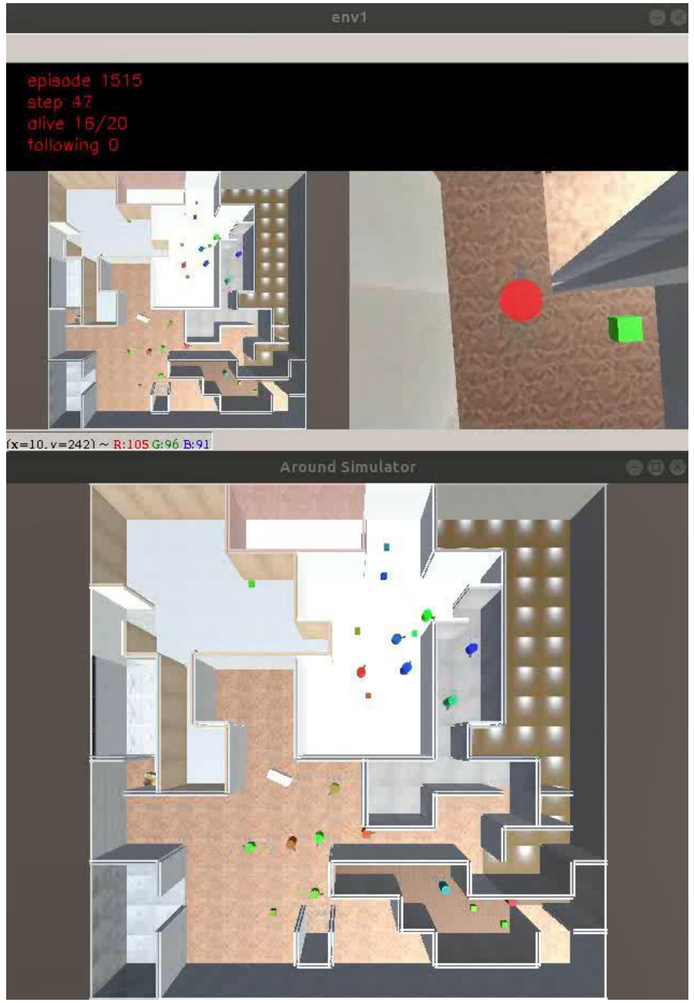


# 3.2 학습 환경 구축하기



## Indoor navigation simulator

- Unity3D based simulator
- Modified ML-Agent
  - 임의의 가변 길이 array를 observation/action 으로 사용 가능
  - Visual observation에 jpg encoding 추가 : 약 30%의 속도 향상
  - C# scripting, 시뮬레이터 빌드의 필요성을 최소화
- Procedural generation of environment
- Up to 40 agents per simulator
- Up to 10 simulators in one desktop PC



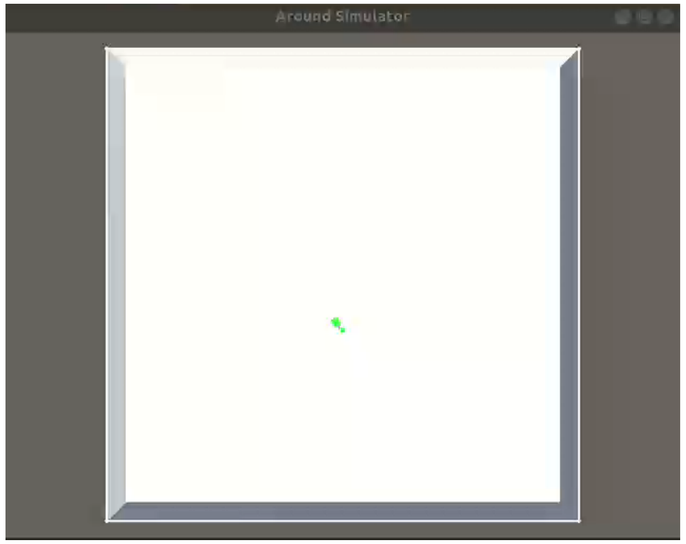

https://github.com/Unity-Technologies/ml-agents

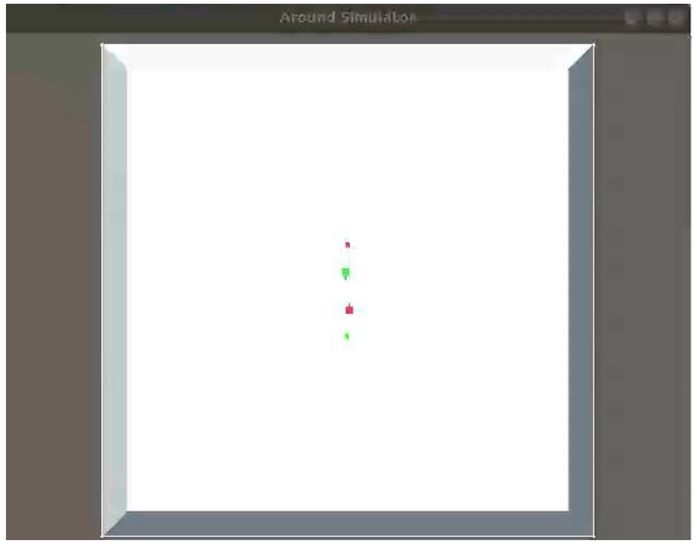
# 3.3 Reward shaping



## Reward shaping이란?

- Reward 함수를 디자인하는 과정
- 에이전트의 행동 방식을 결정
- 환경 세팅과 함께 실용적인 모델 학습을 위해 가장 중요한 요소



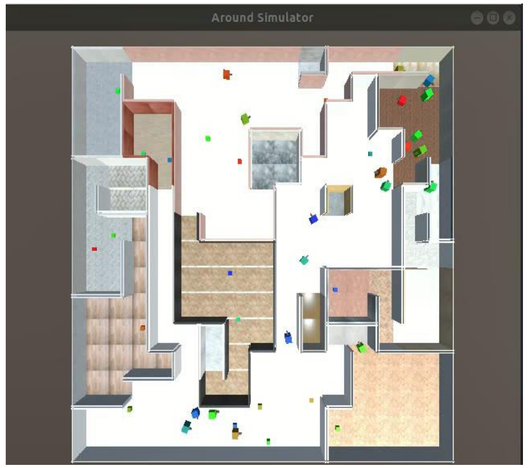



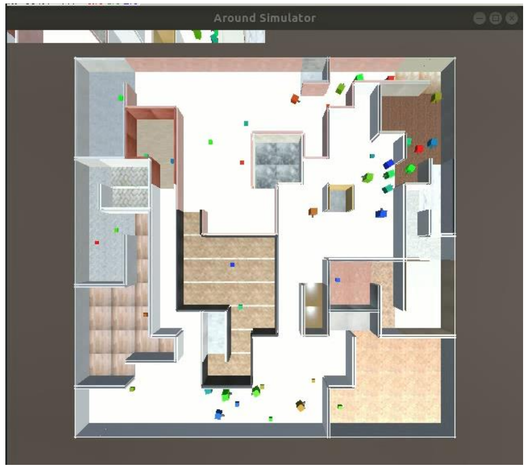
# 3.3 Reward shaping



## Reward shaping의 어려움

- 학습 결과를 예측하기 어려움
- 검증을 위한 학습에 많은 시간과 컴퓨팅 자원 소요
- Use-case(환경, 유저)에 따라 최적 reward 다름



"아냐... 아냐!! 내가 만들고
싶은 건 이런 게 아냐!!"

# 3.3 Reward shaping



## Reward hacking

- Reward 함수 디자인의 허점을 이용하도록 학습
- 높은 Reward를 받지만 원하지 않는 행동
- 실험 visualization과 지속적인 모니터링 필요

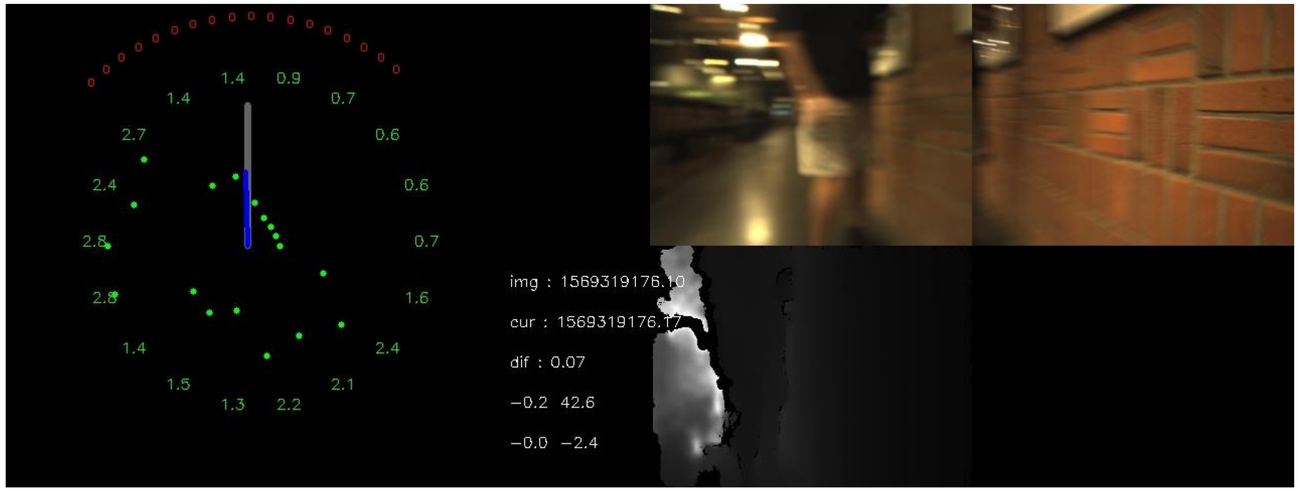
https://openai.com/blog/faulty-reward-functions/

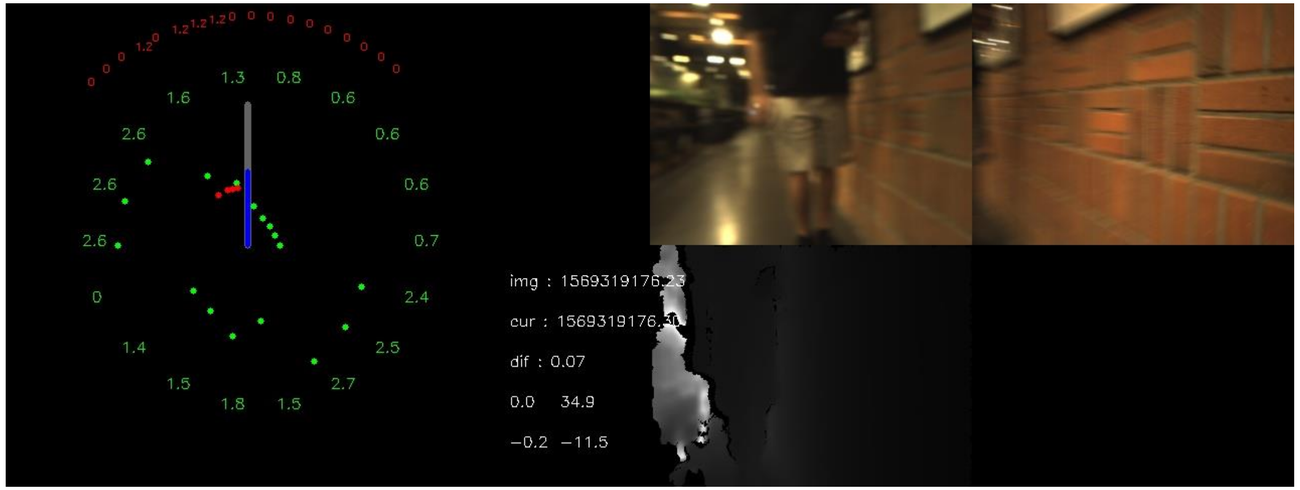
# 3.4 Sim-real gap



## Simulation – Reality gap

- 시뮬레이션과 실제 세계의 차이
- 학습된 에이전트의 성능 저하 원인
- Sensor/actuator noise
- Processing delay
- Hardware problems



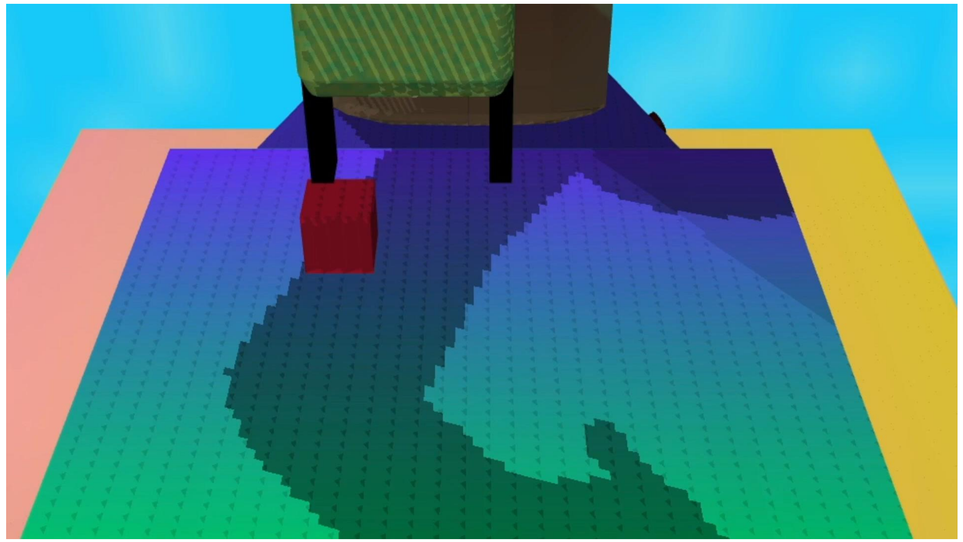



# 3.4 Sim-real gap

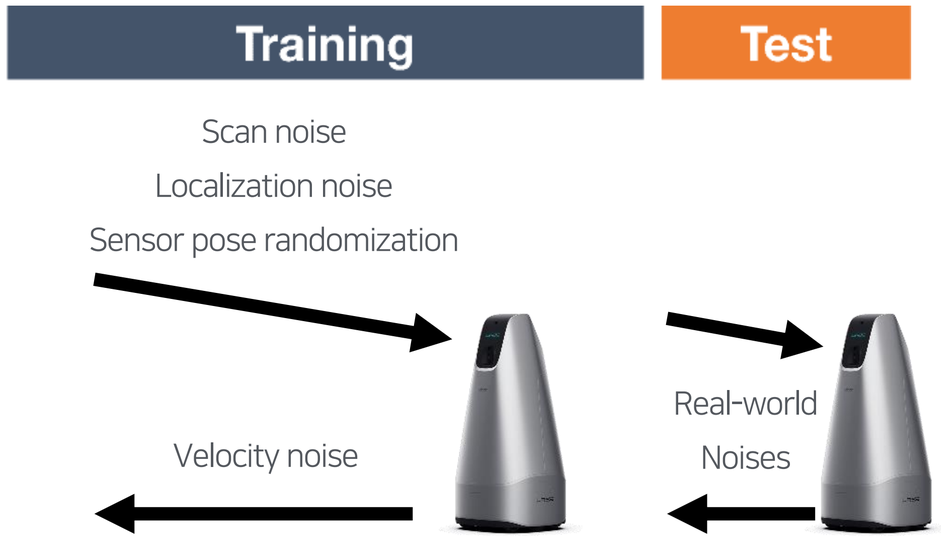


## Simulation – Reality gap

- Domain randomization : 시뮬레이션 내의 dynamics, 환경을 매번 random sampling
- Simulation noises : 실세계의 observation noise를 반영
- System optimization : 최대한 시뮬레이터와 유사하게 시스템 최적화



https://openai.com/blog/generalizing-from-simulation/



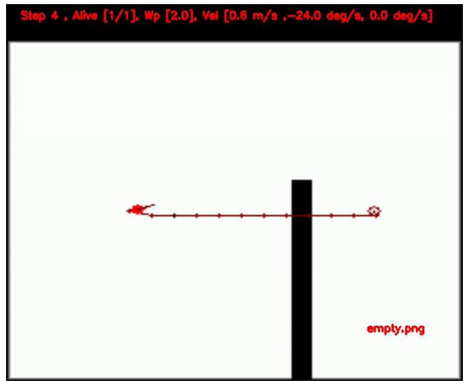
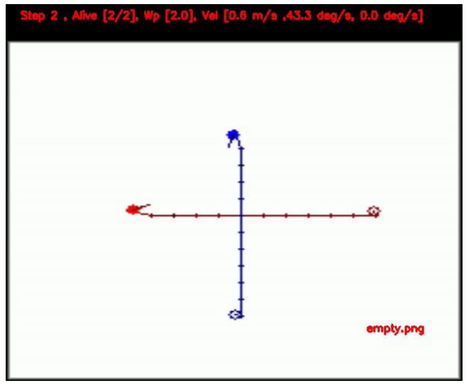
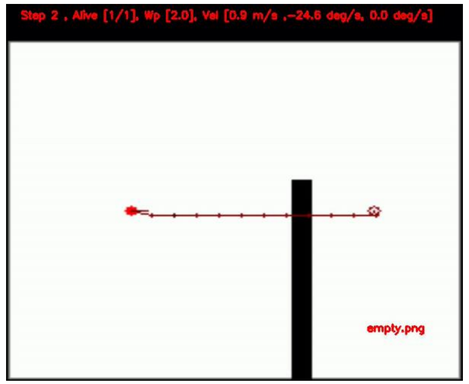
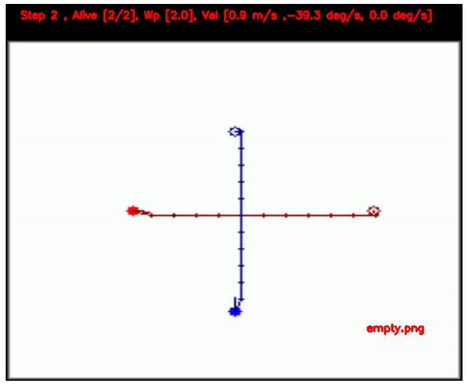
# 3.5 Model research

DEVIEW 2019

## Learning to navigate with narrow FOV

1. Agent with memory

- LSTM / GRU 를 사용
- 2배의 FOV를 가진 에이전트 능가
- 움직이는 장애물 회피에 유리

| | | |
|---|---|---|
| Conv FOV 180 |   |   |
| LSTM FOV 90 |   |   |

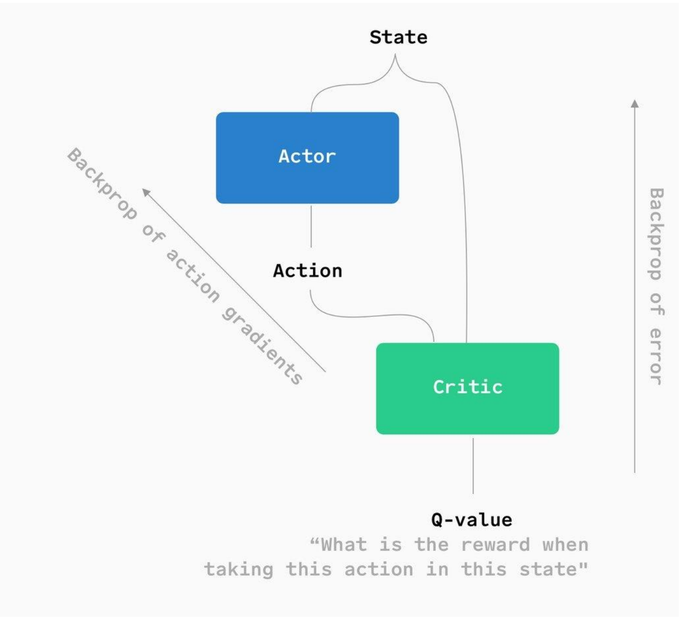
# 3.5 Model research

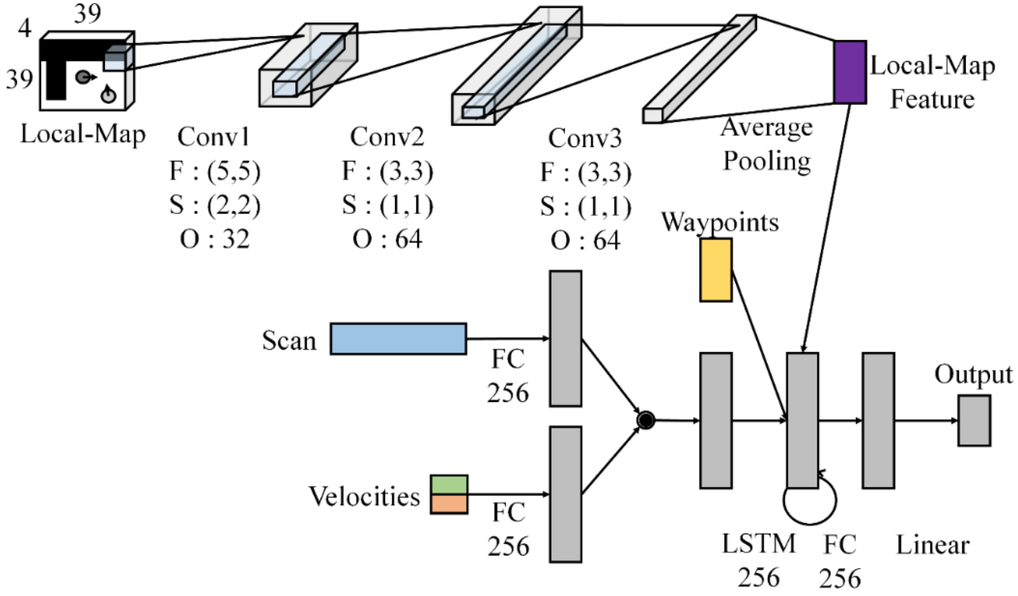
DEVIEW 2019

## Learning to navigate with narrow FOV

### 2. Local map critic

- Partial observability
  -> 부정확한 Q value 학습
  -> sub-optimal한 policy 학습
- Critic network에만 상세한 주변 정보를 추가로 제공
- 정확한 value 학습으로 성능 증가
- 실세계에서 Actor는 추가 정보 없이 동작 가능



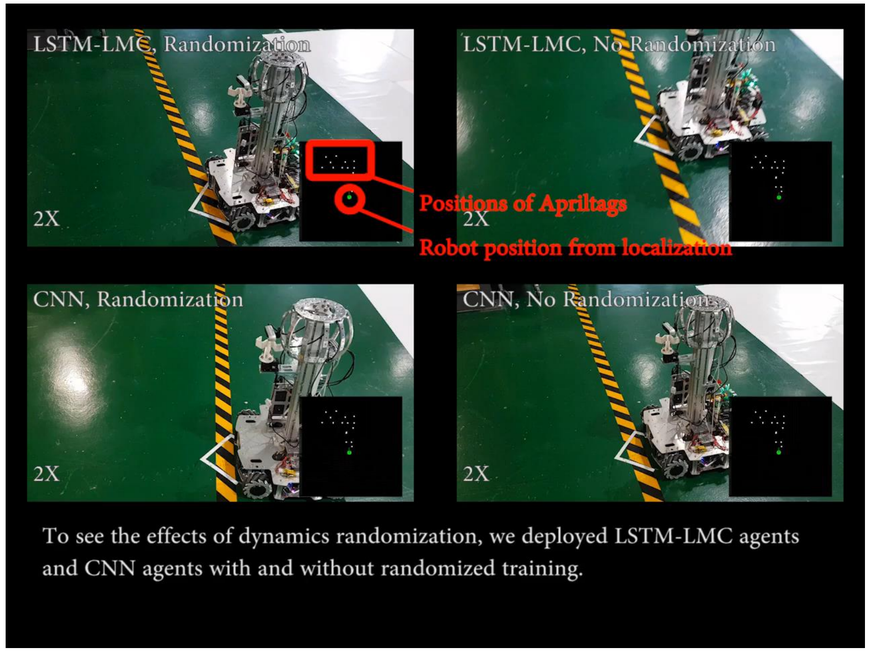



# 3.5 Model research

DEVIEW 2019

## Learning to navigate with narrow FOV

- ICRA 2019에서 공개
- AROUND G에 사용




J. Choi, K. Park, M. Kim and S. Seok, "Deep Reinforcement Learning of Navigation in a Complex and Crowded Environment with a Limited Field of View," *ICRA 2019*

# 3.5 Model research



## Learning to navigate with narrow FOV

- CES 2019 시연

# 3.5 Model research



## Adapting to diverse settings / Finding best setting

- Reward shaping을 할 때 마다 다시 학습 : 시간/리소스 낭비
- 속도, 가속도, 안전거리, reward component weight 등 파라메터 변경 불가
- 모든 agent(사람)이 동일한 policy로 움직이는 것을 가정

# 3.5 Model research



## Adapting to diverse settings / Finding best setting

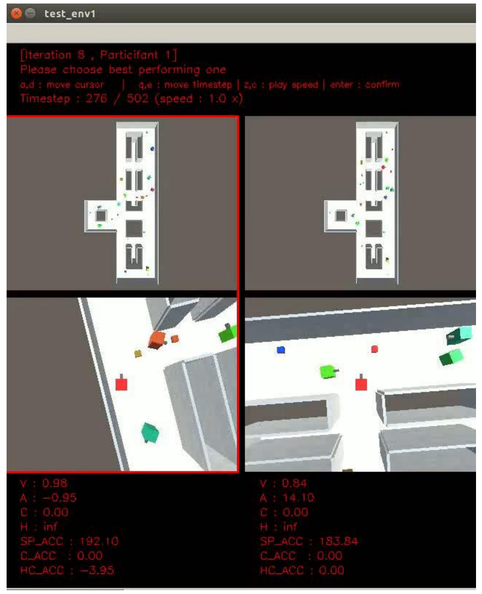
- 로봇 세팅에 대한 파라메터를 추가 input으로 제공
- Multi-agent setting을 활용해 광범위한 파라메터에 대한 policy를 동시에 학습
- 추가 학습 없이 바뀐 파라메터/reward에 즉시 적응
- 여러 세팅의 움직이는 장애물을 상대해 generalization 향상

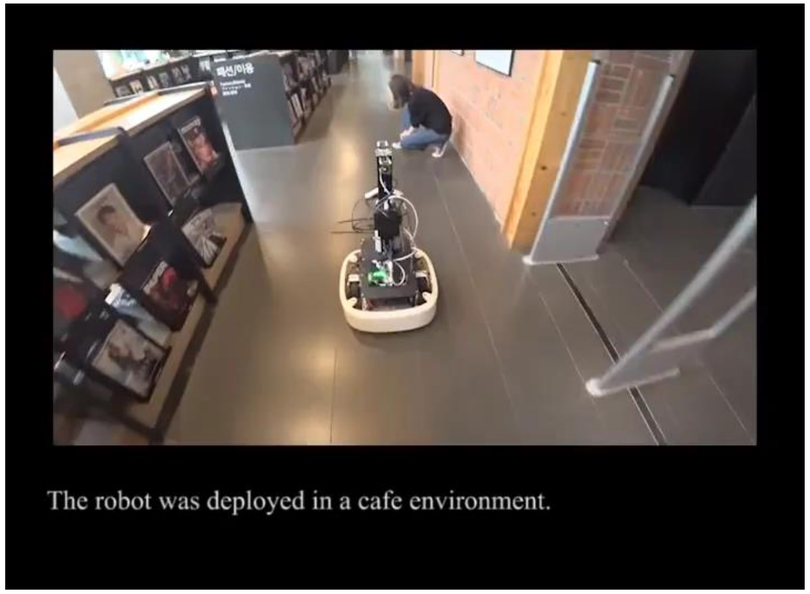
# 3.5 Model research

DEVIEW 2019

## Adapting to diverse settings / Finding best setting

- 사용자의 선호도를 학습해 Use-case에 맞는 세팅을 선택
- Bayesian neural network를 사용한 active learning으로 필요한 선호도 데이터 수 감소
- 실세계 실험에서 선호도 조사 결과 93%의 확률로 최적화된 세팅 선택



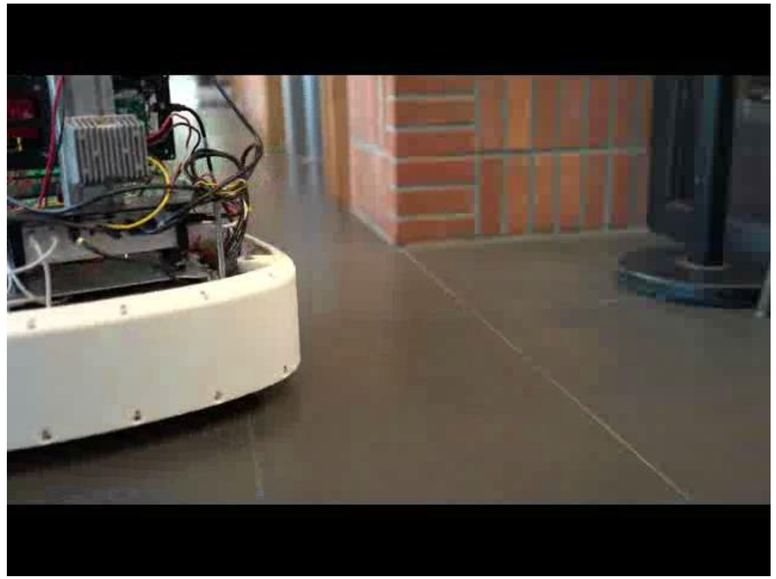



# 3.5 Model research



## Adapting to diverse settings / Finding best setting

- NAVER LABS EUROPE과 공동 연구
- ICRA 2020 submitted
- AROUND C 프로젝트에 적용 : 그린팩토리 1층에서 커피 배달 서비스 예정



# 4. Future Plans

# 4.1 강화학습의 한계



## Sample efficiency

- Policy 학습에 매우 많은 데이터가 필요
- 실세계에서 로봇을 직접 학습 시키기 힘듦
- AROUND G : 30 million samples
- AROUND C : 20 million samples

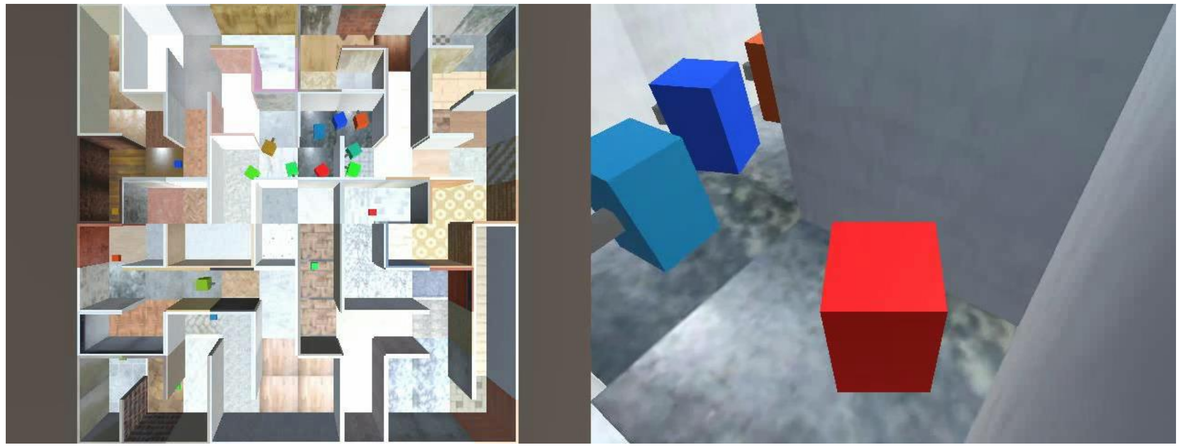
https://www.youtube.com/watch?v=aTDkYFZFWug
https://ai.googleblog.com/2018/06/scalable-deep-reinforcement-learning.html

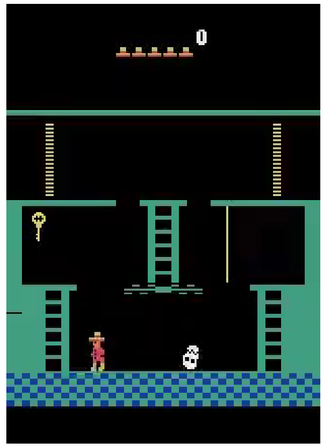
# 4.1 강화학습의 한계



## Exploration

- 부족할 경우 local optima로 수렴
- Sample efficiency가 떨어지는 원인 중 하나
- Reward shaping을 어렵게 하는 원인
- 실세계에서 학습 시 위험한 행동을 반복할 수 없음





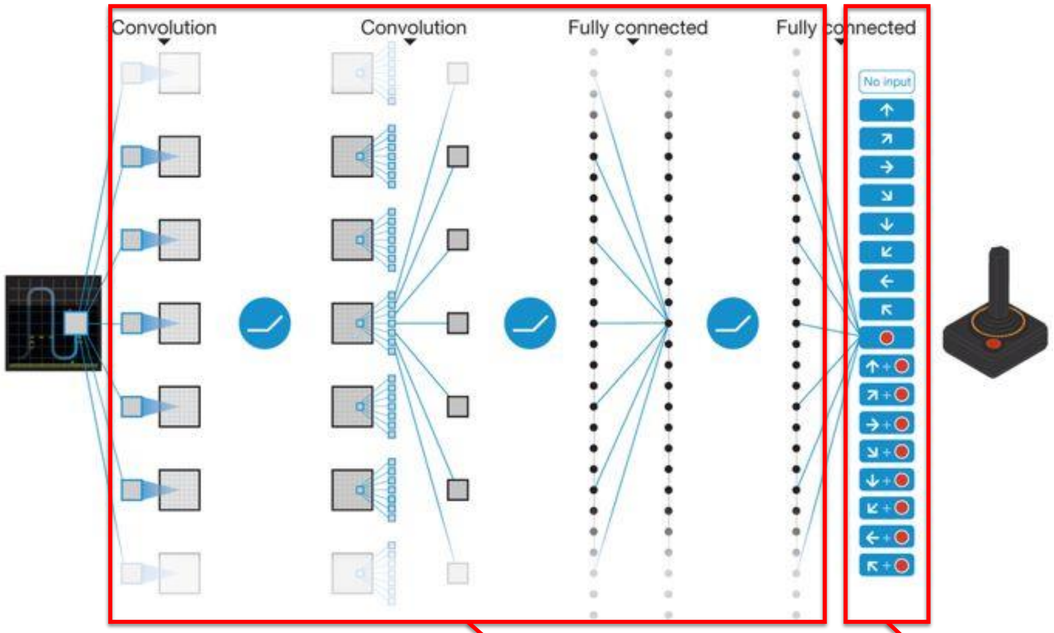
https://openai.com/blog/learning-montezumas-revenge-from-a-single-demonstration/

# 4.1 강화학습의 한계



## Interpretability

- 많은 경우 바로 다음 step의 action만 나오도록 학습
- 어떤 과정을 거쳐서 action을 정하는 지 알 수 없음
- User interaction (gaze, lighting), 모니터링 등에 제약




Black box

Current step only

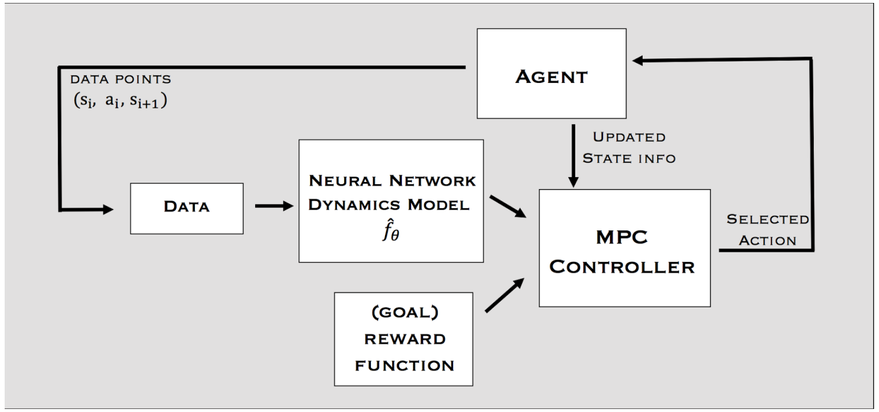
Mnih, Volodymyr, et al. "Human-level control through deep reinforcement learning." *Nature* 518.7540 (2015): 529.

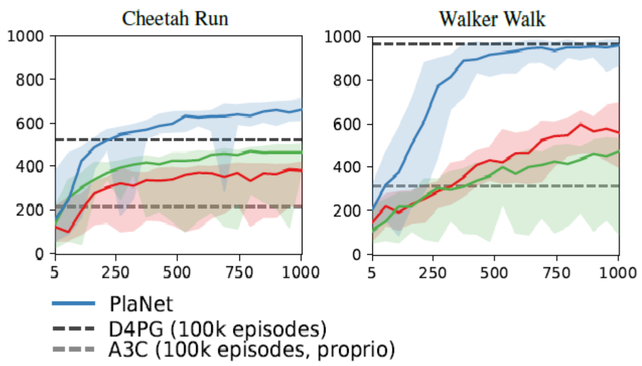
# 4.2 강화학습 연구 동향

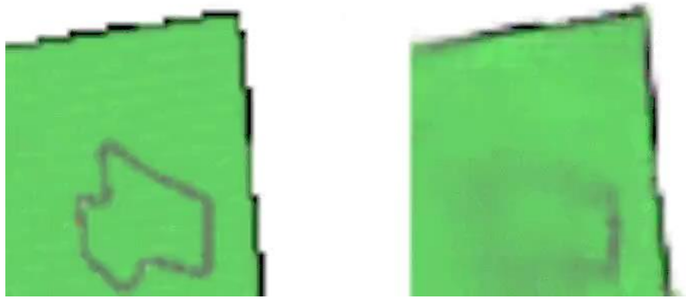
DEVIEW 2019

## Model based RL

- 로봇과 환경의 dynamics를 학습 (environment model)
- 학습된 모델을 사용한 planning
- 학습된 모델을 사용한 policy learning
- Sample efficiency와 interpretability 측면에서 유리



https://bair.berkeley.edu/blog/2017/11/30/model-based-rl/

Hafner, Danijar, et al. "Learning latent dynamics for planning from pixels." arXiv preprint arXiv:1811.04551 (2018).

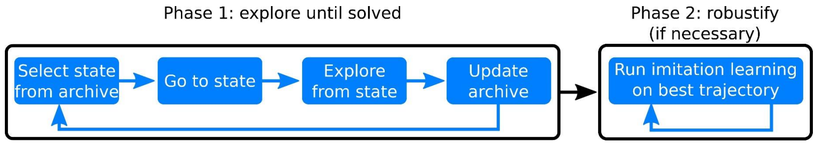
https://worldmodels.github.io/

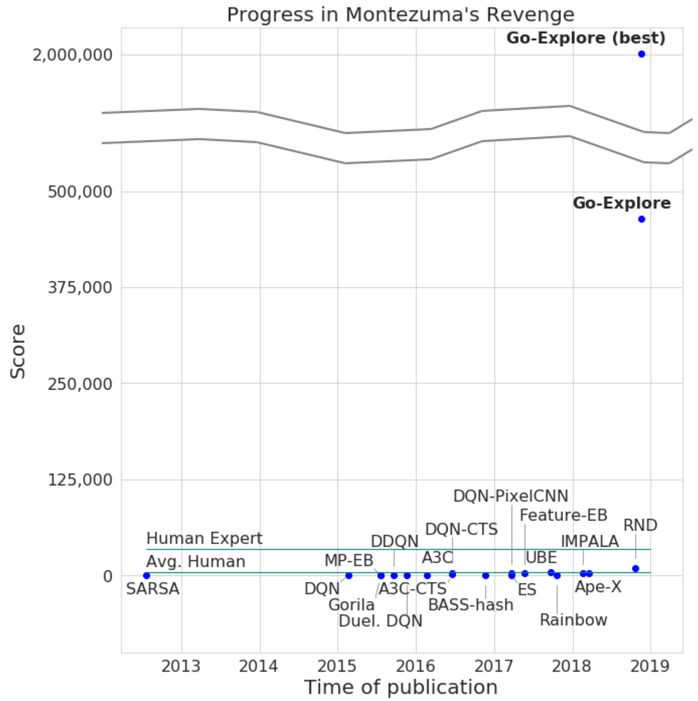
# 4.2 강화학습 연구 동향

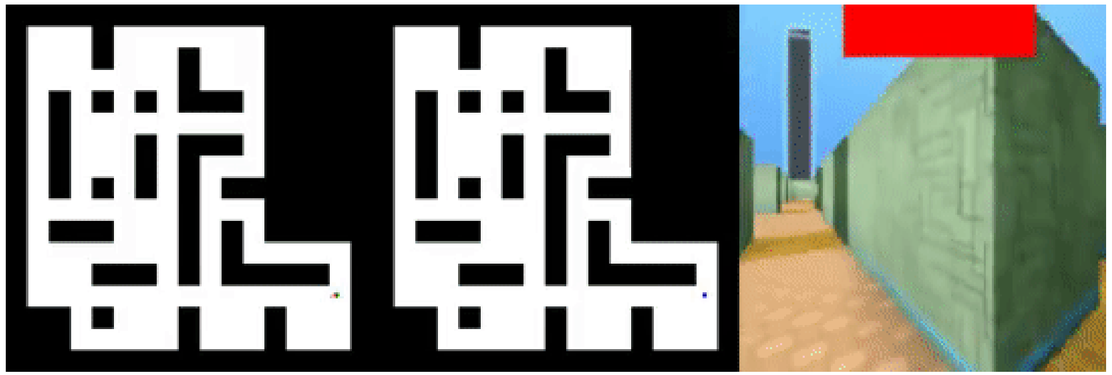
DEVIEW 2019

## Exploration

- Intrinsic motivation
- Memory based exploration
- Heuristics



https://ai.googleblog.com/2018/10/curiosity-and-procrastination-in.html

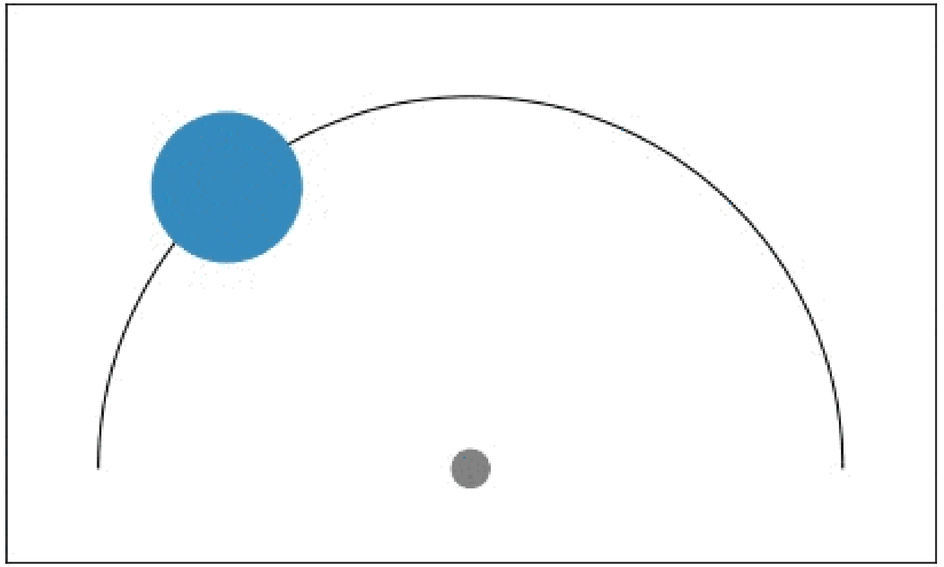
https://eng.uber.com/go-explore/

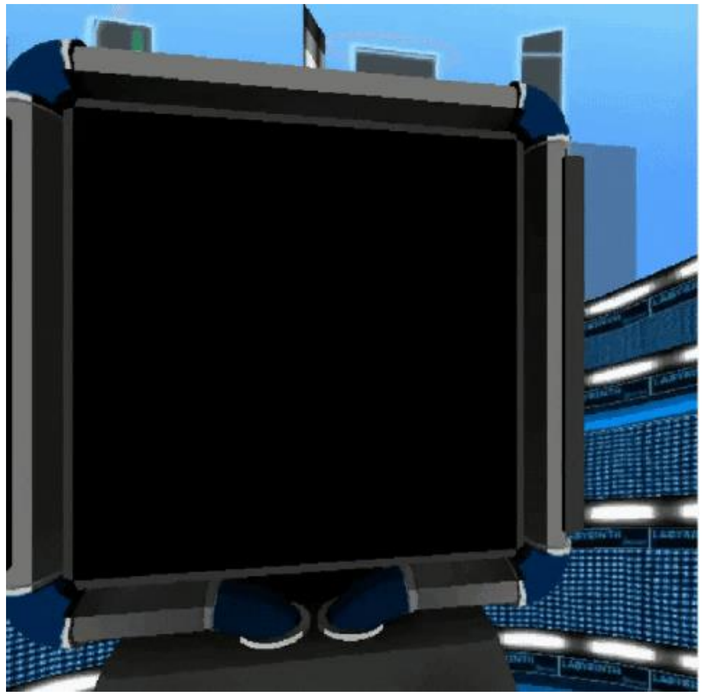
# 4.2 강화학습 연구 동향



## Meta RL

- Learning to learn
- 새로운 task에 빠르게 적응





https://bair.berkeley.edu/blog/2019/06/10/pearl/
https://blog.floydhub.com/meta-rl/

# 4.3 AROUND를 위한 연구 방향



## Fast & safe navigation under infra structure

- 고정밀 지도와 측위 기술에 의존
- 스마트 빌딩의 CCTV, IOT 설비 등을 활용 가능한 에이전트 학습
- 사람보다 빠르고 안전하게
- 상품으로서 통제와 예측이 가능한 알고리즘 개발

## Long range visual exploration & navigation

- 1~2대의 RGB카메라 만으로 주행 가능한 에이전트 학습
- 새로운 장소에 대한 공간 정보를 빠르게 학습 및 업데이트
- End-to-end 방식의 장거리 주행
- Out-of-the-box exploration & navigation

# Q & A

# Thank You

# Appendix1. Deep RL materials



- David Silver lecture : http://www0.cs.ucl.ac.uk/staff/d.silver/web/Teaching.html
- 김성훈 교수님 강의 : https://hunkim.github.io/ml/
- UC Berkeley lecture : http://rail.eecs.berkeley.edu/deeprlcourse-fa17/index.html