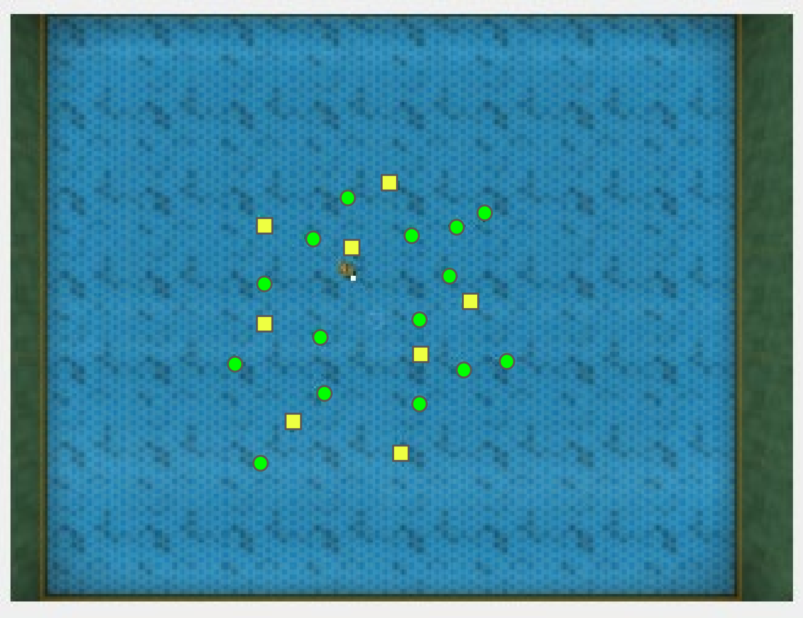
Exploring DQN in DeepMind Lab

Pengkai Lyu, Xiaohu Li

What Problem

- Home Robot vacuuming and mopping in a custom environment
- Pick up garbage (Green +1)
- Avoid knock things down (Yellow -1)
- Find optimal policy according to the environment

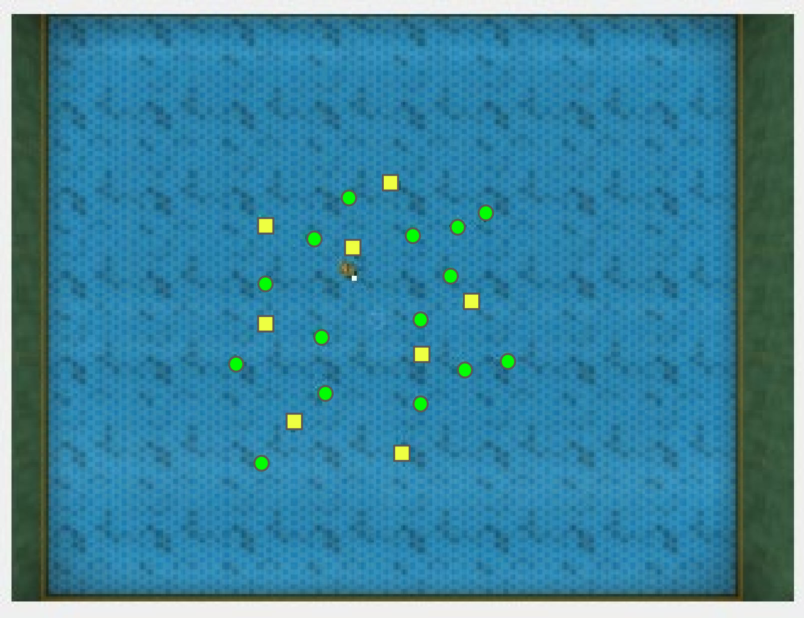


What Method

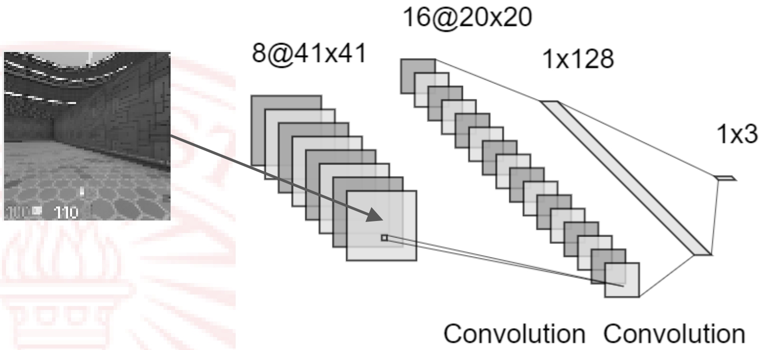
- DeepMind Lab
- ~~Basic Q-Learning~~
- DQN
- Convolutional Layers

State Space: First-person view image 80x80

Action Space: Turn left, turn right, go straight



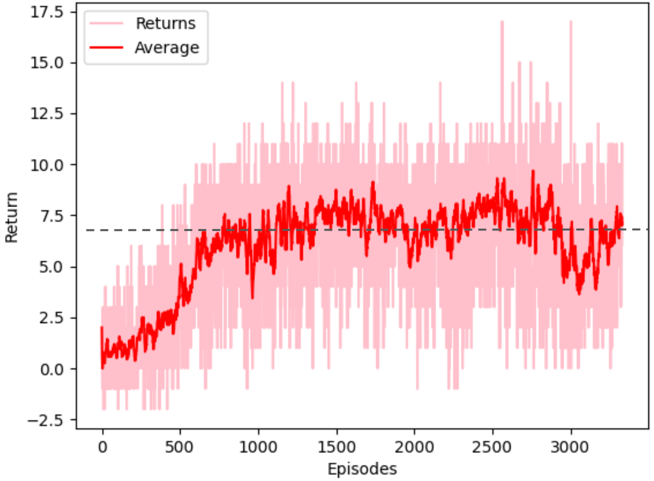
First-Attempt



Work as we expected:

- Pick up Rewards
- The majority of pickup is positive
- RGB color input doesn't help

(x=307, y=170) ~ R:185 G:139 B:27



Problem

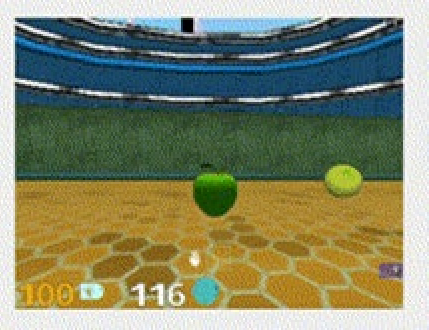
Not moving forward, “shaking its head”

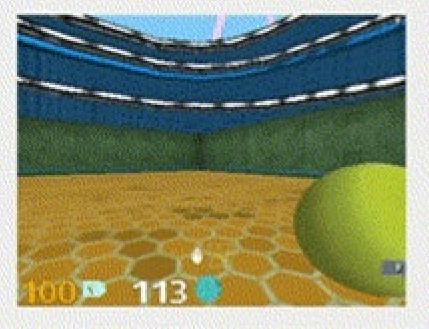
Possible Reasons:

- Rudimentary Network
- Bad Input

Improvement:

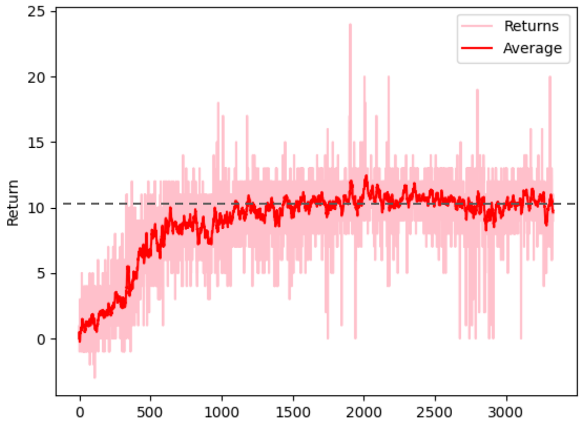
- Build a effective deep network — Add subsampling layers
- Image Queue — Assigned a image buffer for agent that serve as a short memory

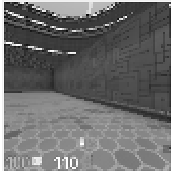


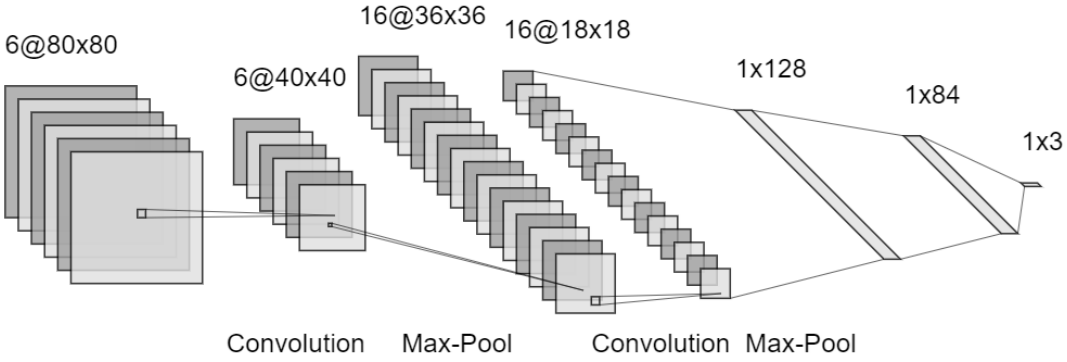


DQN LeNet

- Simple, a pioneering architecture in CNN
- Imitating LeNet-5 architecture, adapte the network by adding two subsampling layers

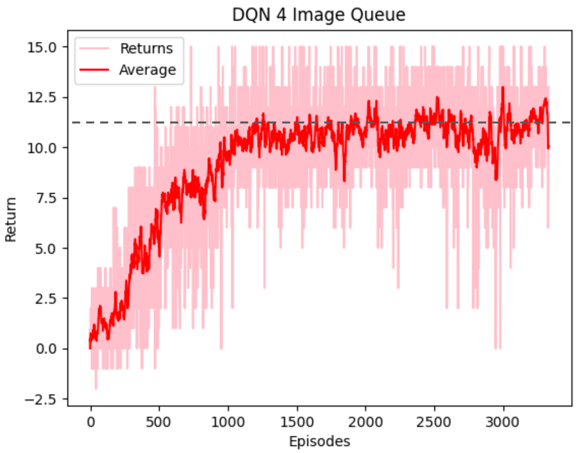


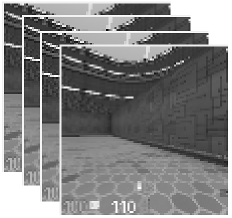


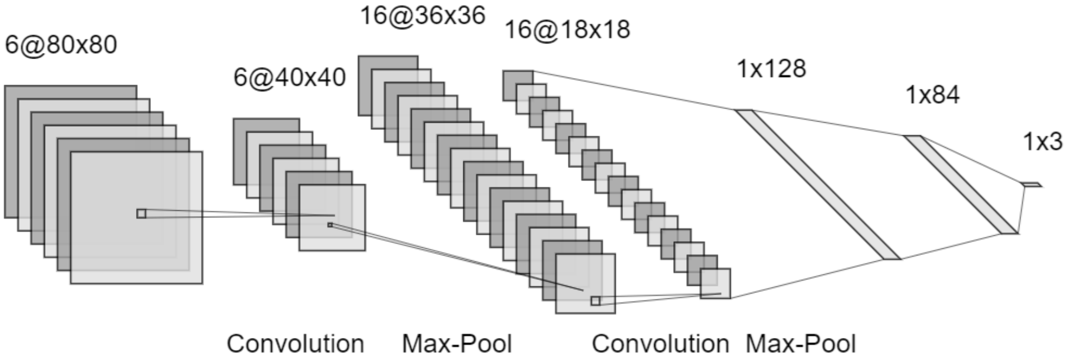


DQN with Images Buffer

- Create a buffer that store a image queue with a length of 4 images
- Enqueue images during the preprocessing stage



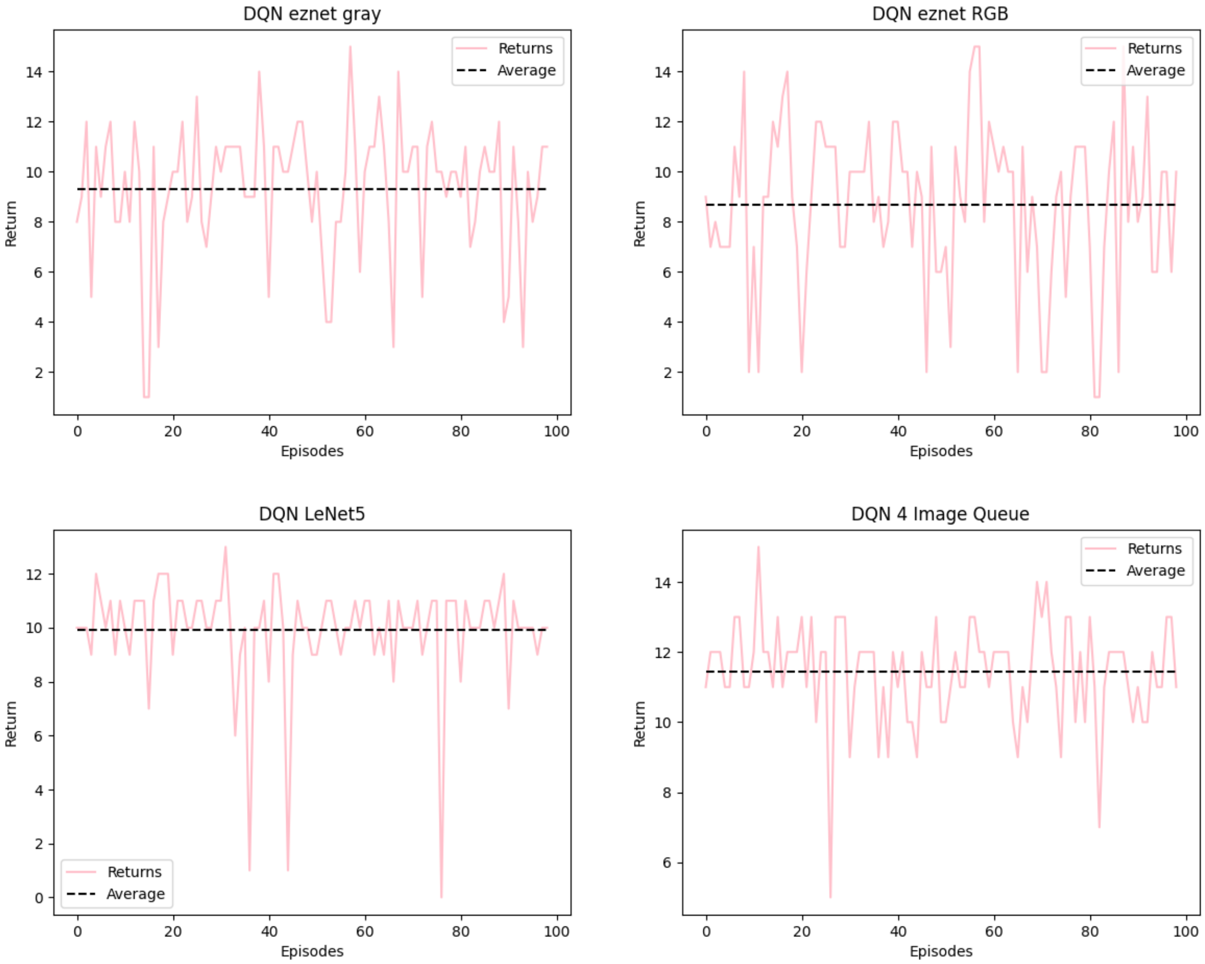




Test Result

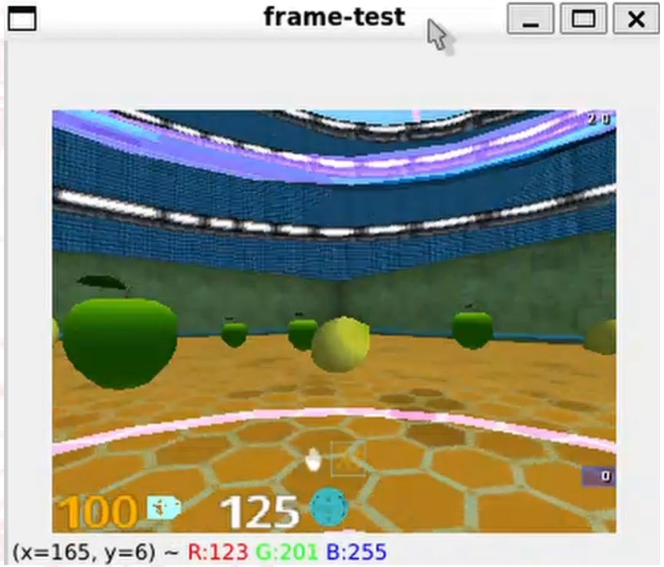
Run 100 episodes for each trained models.

1. Image Buffer
2. Lenet-5
3. Gray scaled
4. RGB input



Result

- Pick up all positive rewards
- Avoid penalty
- Never stuck, keep looking



(x=165, y=6) ~ R:123 G:201 B:255

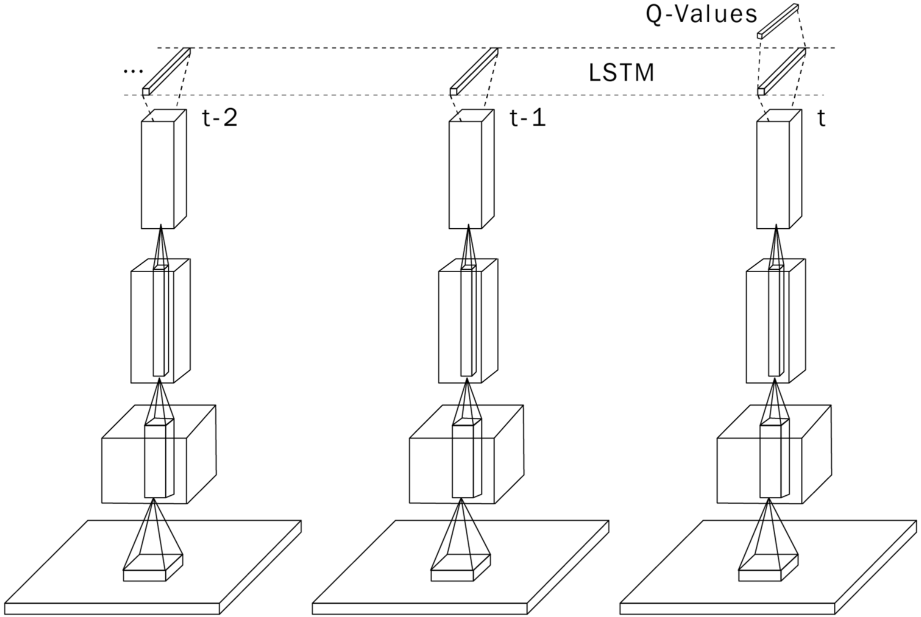
Good Network architecture

A memory Cell

What’s Next

DGN-LSTM

- Handling Sequential Information
- Memory of Past States
- Help in Partially Observable Environments



Hausknecht, M. (2015, July 23). *Deep recurrent Q-Learning for partially observable MDPs*.