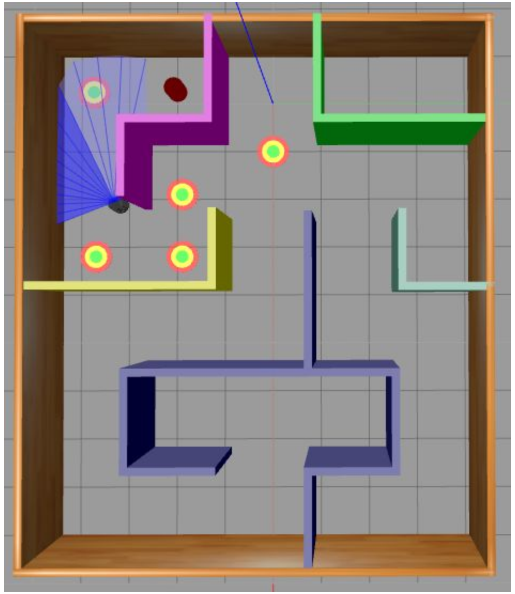
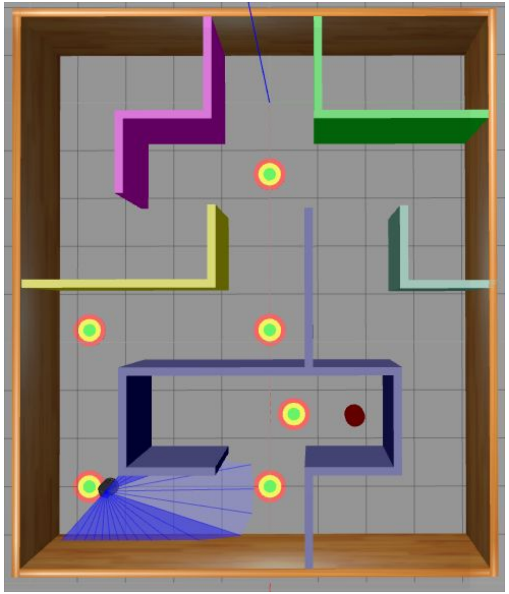
Robot Navigation in Maze with Hints using Reinforcement Learning

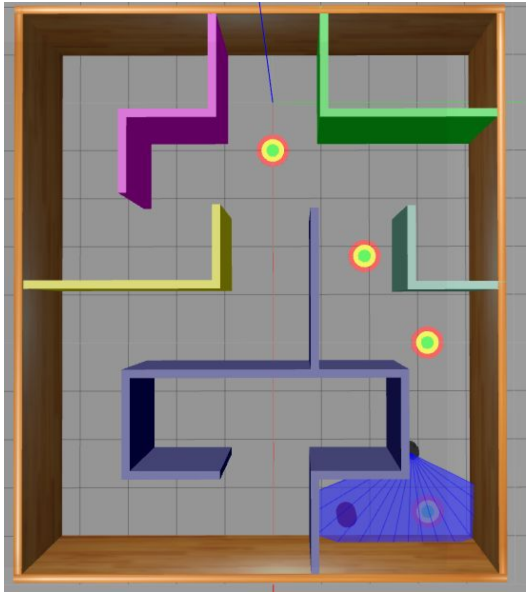
Le Ngoc Tuan Khang
Nguyen Minh Tuan
Nho Minh Tu

1. Problem

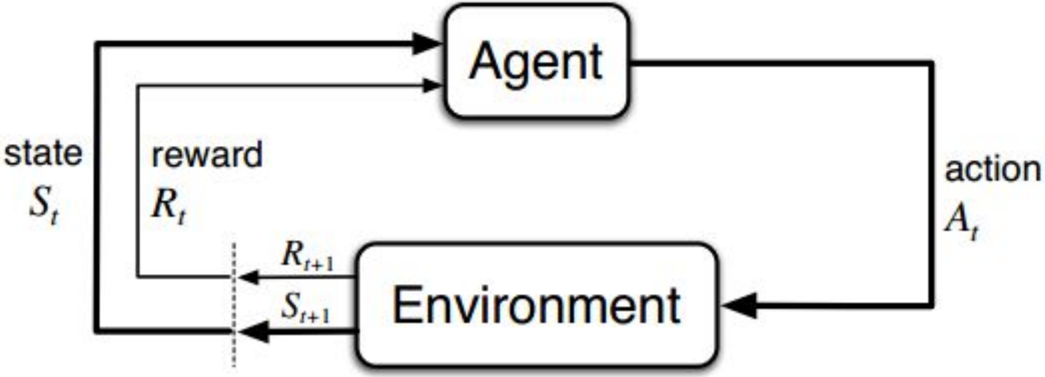
- Robot autonomously finds the way to the target in a maze using hints.





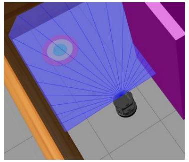


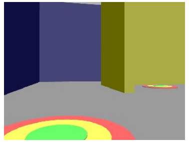
Reinforment Learning



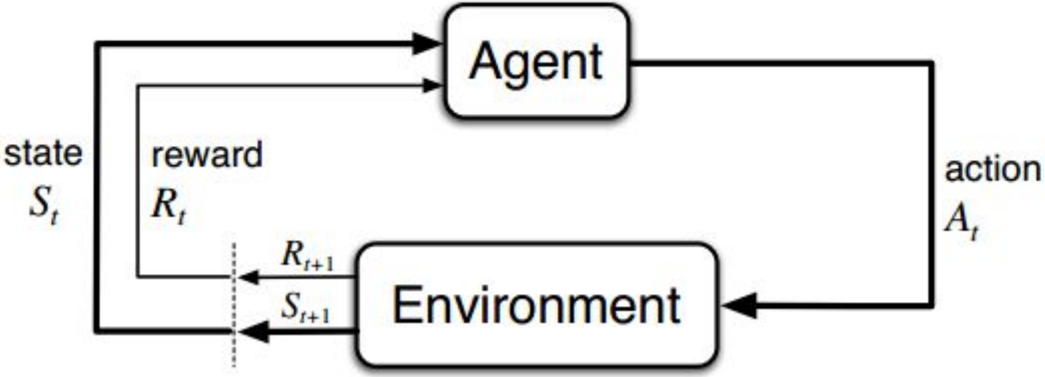
Reinforment Learning

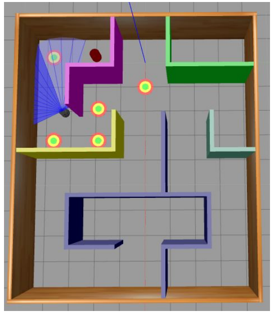
laser

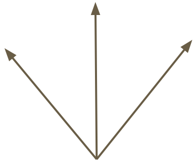




image





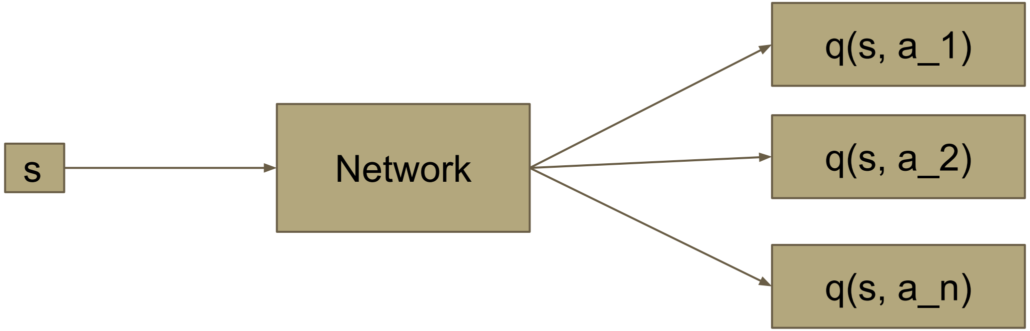


2. Proposed Approach

- Reinforcement Learning Algorithm: Deep Q-network
- DQN for local navigation
- Image Processing for hints
- DQN for final decision

DQN: Details

- q(s, a) is the expected total reward if the agent takes action a in state s.
- But if the state space is too large or continuous, we can't store all q-values.
- Value Function Approximation:

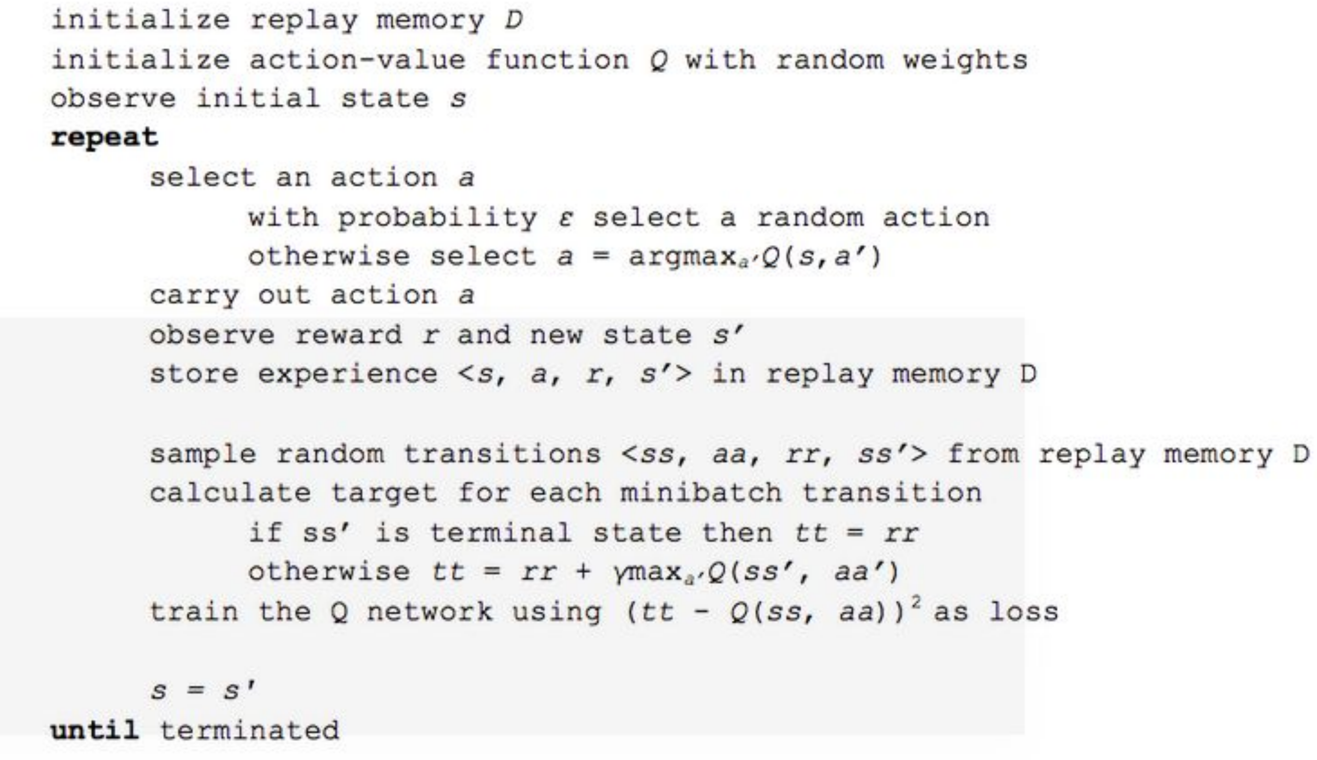


DQN: Details

Update rule:

$$Q(s_t, a_t) \leftarrow (1 - \alpha) \cdot \underbrace{Q(s_t, a_t)}_{\text{old value}} + \underbrace{\alpha}_{\text{learning rate}} \cdot \overbrace{\left(\underbrace{r_t}_{\text{reward}} + \underbrace{\gamma}_{\text{discount factor}} \cdot \underbrace{\max_a Q(s_{t+1}, a)}_{\text{estimate of optimal future value}} \right)}^{\text{learned value}}$$

DQN: Algorithm



```
initialize replay memory $D$
initialize action-value function $Q$ with random weights
observe initial state $s$
repeat
     select an action $a$
          with probability $\varepsilon$ select a random action
          otherwise select $a = \text{argmax}_{a'} Q(s,a')$
     carry out action $a$
     observe reward $r$ and new state $s'$
     store experience $<s, a, r, s'>$ in replay memory D

     sample random transitions $<ss, aa, rr, ss'>$ from replay memory D
     calculate target for each minibatch transition
          if ss' is terminal state then $tt = rr$
          otherwise $tt = rr + \gamma \max_{a'} Q(ss', aa')$
     train the Q network using $(tt - Q(ss, aa))^2$ as loss

     $s = s'$
until terminated
```

Deep Q-network for Local Navigation

- Robot learns to avoid hitting walls in a similar but simpler environment.

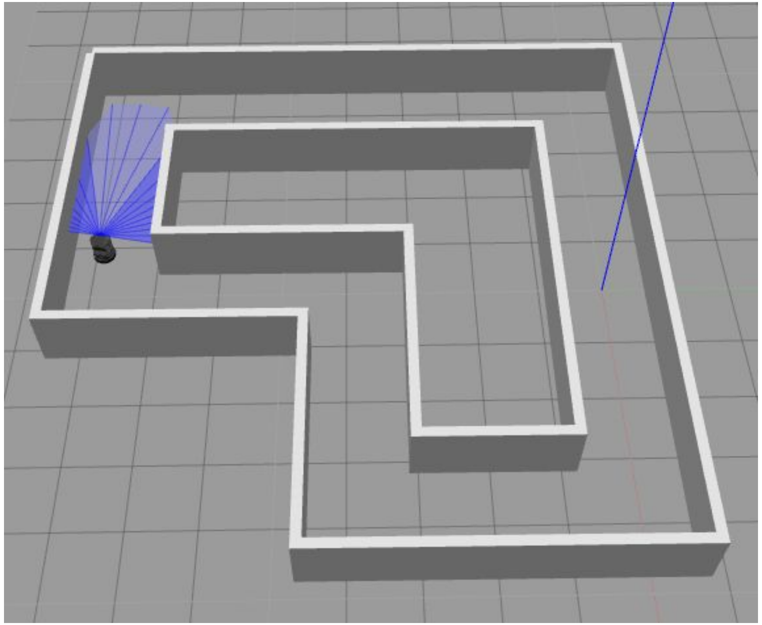
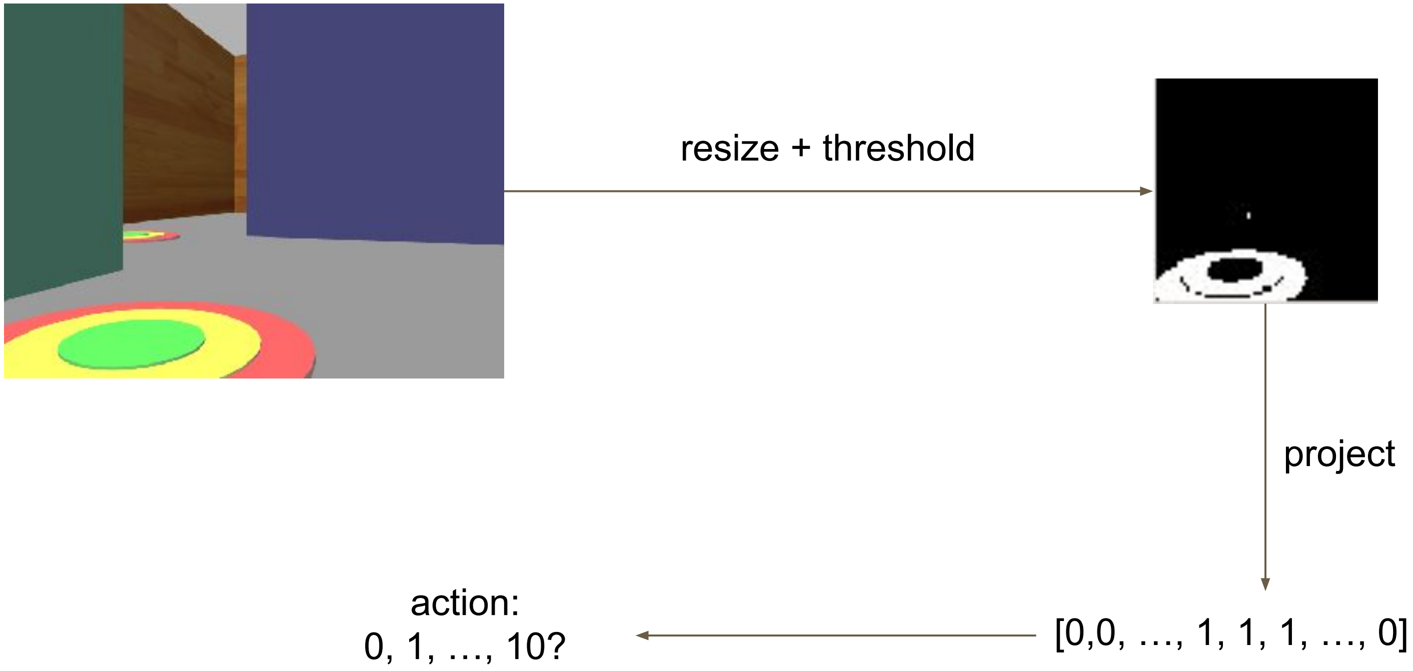


Image Processing for Hint Detection



resize + threshold

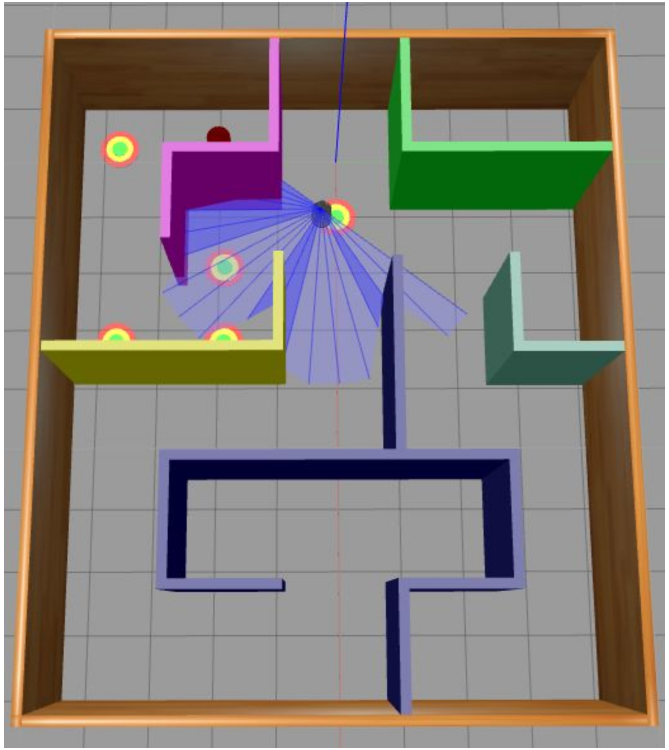
project

[0,0, …, 1, 1, 1, …, 0]

action:
0, 1, …, 10?

Deep Q-network for Action Selection

- 2 actions are available:
    - “local navigation” action and “hint” action
- Which action to select at a specific state?
    - e.g: “local navigation” when being close to wall and “hint” when facing many turns
- Let robot learn that by reinforcement learning.



3. Result and Demo

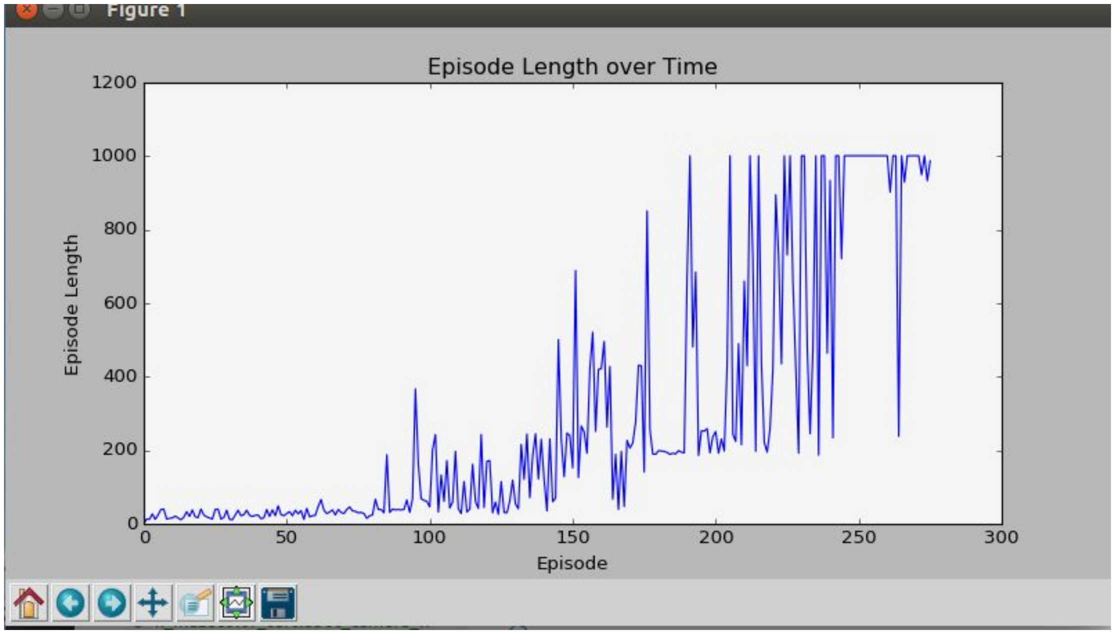


Figure 1
Episode Length over Time
Episode Length
Episode