Neural.**Orb**

development by Tabor Henderson

Deep Quality Networks and Agents

DeepMind has demonstrated DQN's as game playing AIs on a variety of Atari games including Space Invaders and Pong.

Recently, one of their AI's beat the No. 1 ranked player at the board game Go.

DeepMind’s DQNs

Utilized “experience replay” to avoid overfitting and local minima.

Trained on 10 million frames, epsilon-greedy starting at p = 1, annealed to p = 0.1 over first 1 million frames, held at p = 0.1 thereafter.

DeepMind’s DQN Algorithm

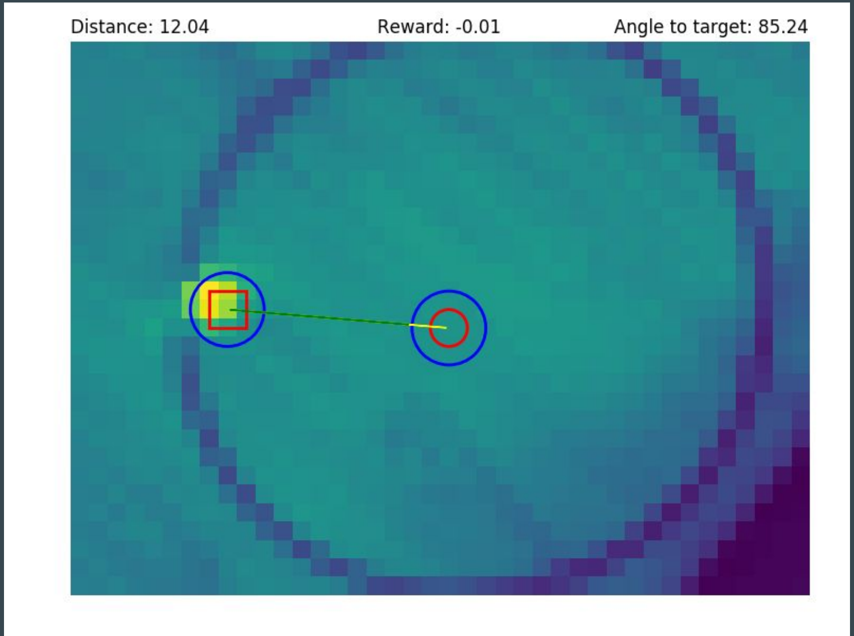
Algorithm 1 Deep Q-learning with Experience Replay

Initialize replay memory $\mathcal{D}$ to capacity N
Initialize action-value function Q with random weights
for episode $= 1, M$ **do**
  Initialise sequence $s_1 = \{x_1\}$ and preprocessed sequenced $\phi_1 = \phi(s_1)$
  for $t = 1, T$ **do**
    With probability ϵ select a random action a_t
    otherwise select $a_t = \max_a Q^*(\phi(s_t), a; \theta)$
    Execute action a_t in emulator and observe reward r_t and image x_{t+1}
    Set $s_{t+1} = s_t, a_t, x_{t+1}$ and preprocess $\phi_{t+1} = \phi(s_{t+1})$
    Store transition $(\phi_t, a_t, r_t, \phi_{t+1})$ in $\mathcal{D}$
    Sample random minibatch of transitions $(\phi_j, a_j, r_j, \phi_{j+1})$ from $\mathcal{D}$
    Set $y_j = \begin{cases} r_j & \text{for terminal } \phi_{j+1} \\ r_j + \gamma \max_{a'} Q(\phi_{j+1}, a'; \theta) & \text{for non-terminal } \phi_{j+1} \end{cases}$
    Perform a gradient descent step on $(y_j - Q(\phi_j, a_j; \theta))^2$ according to equation 3
  end for
end for

Deploying a DQN to a Physical Robot

Using computer vision and commands sent over Bluetooth, I set up a simple training environment for a Sphero robotic ball

Learning was very slow, but showed promise



Training Acceleration

In addition to DeepMind's training acceleration techniques, I explored model guidance and model extension. I demonstrated these techniques in simulation.

Q-learning is, by definition, model-free, but by using a model to guide the DQN in training, I accelerated training substantially.

By integrating a deterministic strategy in the DQN's options, I accelerated training by an order of magnitude.