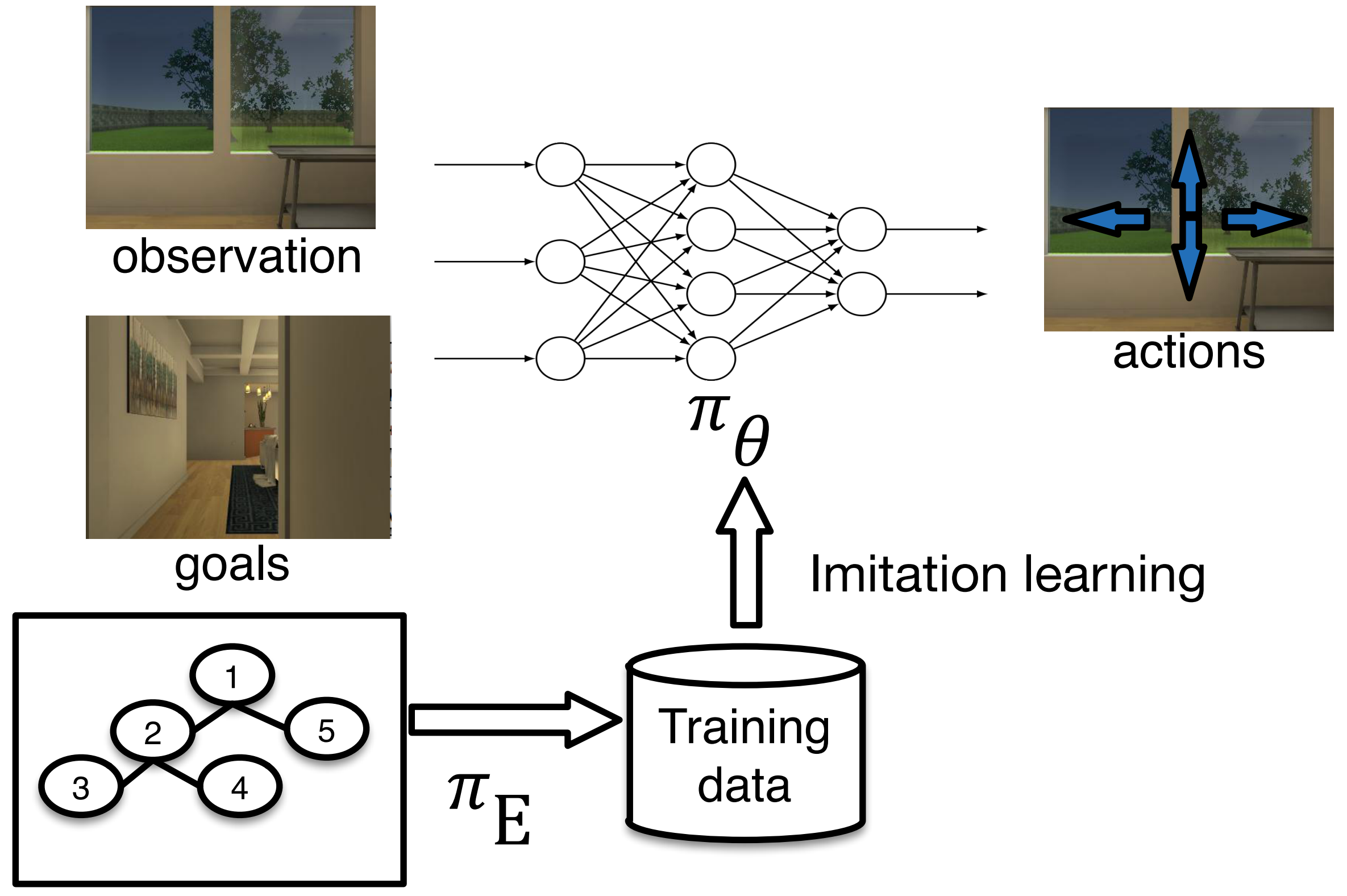
IMITATION LEARNING WITH THOR

ALBERT LIU (ALBERTPL@STANFORD.EDU)

PROBLEM

Visual navigation is the task of determining the shortest path between a starting point and a given goal point with visual inputs, which has been active research topic in vision and control domains for many years. We explore imitation learning approaches with episodic setting.

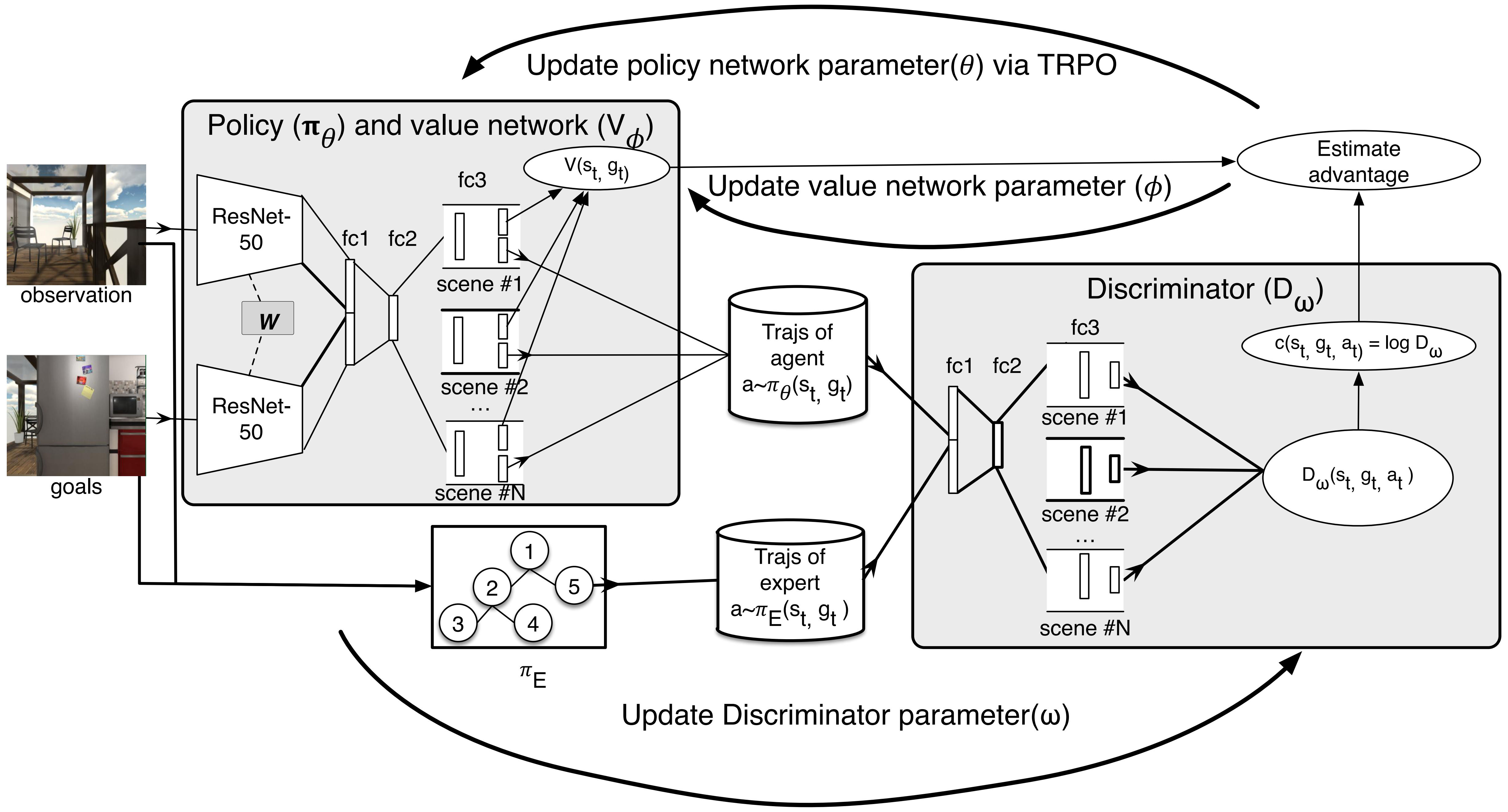


DATASET

The House Of inteRactions (AI2-THOR) framework provides an simulation environment for visual navigation.

1. 20 scenes with 130∼1000 images per scene
2. Each observation is represented by four $84 \times 84 \times 3$ images
3. Pre-trained $2048D$ feature vector for each image is available
4. State space has a dimension of $2 \times 2048 \times 4$
5. Actions space is a discrete set of commands, $i.e. forward, backward, left, right$
6. Transitions between states are deterministic

SYSTEM ARCHITECTURE


Update policy network parameter(θ) via TRPO
Policy ($\boldsymbol{\pi}_\theta$) and value network (V_ϕ)
observation
goals
ResNet-50
W
ResNet-50
fc1
fc2
fc3
$V(s_t, g_t)$
scene #1
scene #2
…
scene #N
Update value network parameter (ϕ)
Estimate advantage
Trajs of agent $a \sim \pi_\theta(s_t, g_t)$
Trajs of expert $a \sim \pi_E(s_t, g_t)$
1
2
3
4
5
π_E
Discriminator (D_ω)
fc1
fc2
fc3
scene #1
scene #2
…
scene #N
$c(s_t, g_t, a_t) = \log D_\omega$
$D_\omega(s_t, g_t, a_t)$
Update Discriminator parameter(ω)

- Expert policy is computed from environment's transition table
- Use label smooth. $K = 4$ and a^* is from deterministic expert policy $\pi_E(a|s) = (1 - \epsilon)\mathbf{1}\{a = a^*\} + \epsilon/K$

GENERATIVE ADVERSARIAL IMITATION LEARNING

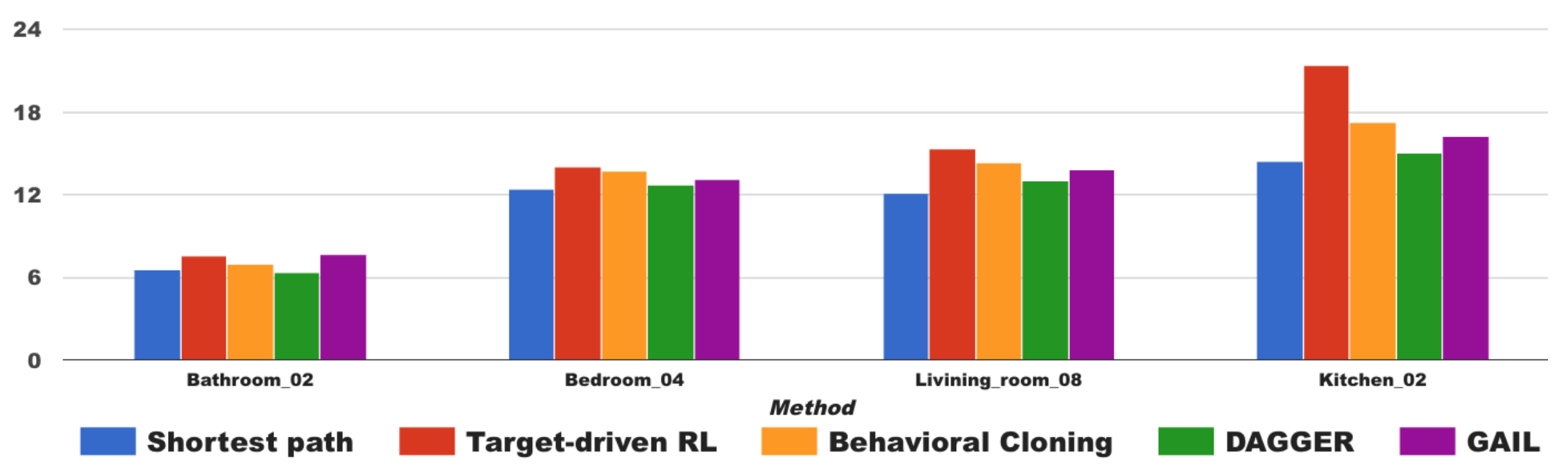
1. Discriminatory classifier (D_ω)
   - Used to fit the cost function from expert policy
   - $\omega = \arg\min_\omega L_\omega(s, g, a)$
   - $L_\omega = \hat{E}_{\tau_i}[\log(D_\omega(s, g, a))] + \hat{E}_{\tau_E}[\log(1 - D_\omega(s, g, a))]$
   - $c(s, g, a) = \log(D_\omega(s, g, a))$
2. Value network (V_ϕ)
   - Used as baseline estimator
   - $\phi = \arg\min_\phi (V(s, g) - \hat{V})^2$
3. Policy network (π_θ)
   - TRPO update
   - $L_{\pi_{old}}(\theta) = E_{s\sim\pi_{old}, a\sim\pi_{old}}[\frac{\pi(a|s,g)}{\pi_{old}(a|s,g)} A^{\pi_{old}}(s, g, a)]$
   - $\theta = \arg\max L_{\pi_{old}}(\pi_\theta)$ subject to $\overline{KL}_{\pi_{old}}(\pi_\theta) < \delta$

DATASET AGGREGATION

```
Sample T step trajectories using $\pi_E^l$ for each task $l$.
Get dataset $D^l = (s^l, g^l, \pi_E^l)$
for $t = 1 \rightarrow N$ do
    Train $\pi_\theta^l$ on $D^l$
    Sample T step trajectories using $\pi_\theta^l$
    Get dataset $D_t^l = (s^l, g^l, a^l \sim \pi_E^l(s^l, g^l))$
    Aggregate datasets: $D^l \leftarrow D^l \cup D_t^l$
end for
```

1. $\theta = \arg\min_\theta H_{s,a\sim D}(\pi_E, \pi_\theta)$
2. The assumption is that we can ask expert to give demonstrations for any given pair of $(observation, target)$ during training.

PRELIMINARY RESULTS



The above results are evaluated over 100 sampled episodes after training of less than 100 thousands steps across 20 targets (5 targets for 4 scenes), except for Target-driven RL.

DISCUSSIONS

1. In general, imitation learning approaches converge faster than baseline method, i.e. Target-driven RL which requires 10 millions steps. This suggests better data efficiency, as expected
2. Also imitation learning approaches achieve less average episode length due to the use of expert demonstrations
3. Behavioral Cloning is easy to converge but it appears to converge to non-optimal local minimum
4. DAGGER perform almost as good as expert with the limitation that it needs to access expert's policy at will during training
5. It turns out that it is very important to bootstrap policy network with Behavioral Cloning for GAIL training, as the gradients are otherwise close to zero and it doesn't learn

FUTURE WORKS

1. It would be interesting to see if memory of experience can improve either data efficiency or performance. One possibility is to replace $fc1$ with LSTM cells.
2. The feature vectors are pre-trained on ImageNet dataset and there may be values in fine-tuning or adding convnets layers in the system.