1. Dataset

We collected several datasets for our experiments, which we present in this section. The data used for both real robot experiments and simulation experiments is presented here.

1.1. Real robot

Our robot experiments were carried out using 306 expert demonstrations. The training environment is shown in Figure 1. We used 1700 images as weakly labeled data, a small sample from which is available in Figure 2. We evaluated our policies in unseen environments, which can be seen in Figure 3.

1.2. Simulation

Our simulation datasets consists of 400 expert trajectories from a single environment, where each trajectory has 100 timesteps. This results in 40,000 images for each task. For the weakly data, we have 2,000 images from each of the 40 environments, which results in a total of 80,000 images.

Pushing For the pushing task, the images from training and test environments can be seen in Figure 4. The unlabeled images for the pushing task can be seen in Figure 5. This pushing task has now been merged into the OpenAI Gym repository https://github.com/openai/gym/blob/master/gym/envs/mujoco/pusher.py.

Striking For the striking task, the images from training and test environments can be seen in Figure 6. The unlabeled images for the striking task can be seen in Figure 7. This pushing task has now been merged into the OpenAI Gym repository https://github.com/openai/gym/blob/master/gym/envs/mujoco/striker.py.

2. Training Details

All of our models with attention have the same architecture: five convolutional layers with 3x3 filters, and the number of filters are 64, 64, 32, 32, and 16. The stride is equal to 2 for the first conv layer, and 1 for all the subsequent conv layers. The conventional convolutional network...
instead has the same number of filters and layers as the attention model, a stride of 4 in the first layer, and strides of 1 in subsequent convolutional layers, in order to preserve the spatial information that is essential for the task. The first layer of all networks (with and without our attention mechanism) is initialized from the VGG-16 network [2]. The fully connected layers all have 400 units. We use dropout on the output of the spatial attention layer, in order to force some degree of redundancy into the feature points, which increases the robustness of the model. We also use batch normalization between the convolutional layers. Training is done with Adam [1] and a learning rate of 3e-4, and all models are trained for 50K training steps. Input images are 200x175 for the simulation experiments, and 320x240 for the real robot experiments. For the first 5K steps, we only optimize for $L_{\text{task}}$; we optimize for the complete objective in all subsequent steps.
References
