Learning Robust Representation for Reinforcement Learning with Distractions by Reward Sequence Prediction (Supplementary Material)

Qi Zhou[1] **Jie Wang**[1,2] **Qiyuan Liu**[1] **Yufei Kuang**[1] **Wengang Zhou**[1,2] **Houqiang Li**[1,2]

[1]1CAS Key Laboratory of Technology in GIPAS, University of Science and Technology of China
[2]Institute of Artificial Intelligence, Hefei Comprehensive National Science Center

1 PROPOSITIONS

1.1 AN UPPER BOUND FOR TASK IRRELEVANT INFORMATION

Proposition 1.1. *Assume that all trajectories are sampled under a fixed stationary policy. Then, we have*

$$I(\mathbf{d}_t; \mathbf{r}_{t+1:t+N} \mid \mathbf{s}_t) \leq \sum_{i=0}^{N-1} I(\mathbf{d}_{t+i}; \mathbf{a}_{t+i} | \mathbf{s}_{t+i}).$$

Proof. As $\mathbf{d}_t$ is independent of $\mathbf{r}_{t+1:t+N}$ conditioned on $(\mathbf{s}_t, \mathbf{a}_{t:t+N-1})$, we have

$$\begin{aligned} I(\mathbf{d}_t; \mathbf{r}_{t+1:t+N} | \mathbf{s}_t) &\leq I(\mathbf{d}_t; \mathbf{a}_{t:t+N-1}, \mathbf{r}_{t+1:t+N} | \mathbf{s}_t) \\ &= I(\mathbf{d}_t; \mathbf{a}_{t:t+N-1} | \mathbf{s}_t) + I(\mathbf{o}_t; \mathbf{r}_{t+1:t+N} | \mathbf{s}_t, \mathbf{a}_{t:t+N-1}) \\ &= I(\mathbf{d}_t; \mathbf{a}_{t:t+N-1} | \mathbf{s}_t) \\ &= I(\mathbf{d}_t; \mathbf{a}_t | \mathbf{s}_t) + \sum_{i=1}^{N-1} I(\mathbf{d}_t; \mathbf{a}_{t+i} | \mathbf{s}_t, \mathbf{a}_{t:t+i-1}). \end{aligned} \tag{1}$$

Then, we have

$$\begin{aligned} I(\mathbf{d}_t; \mathbf{a}_{t+i} | \mathbf{s}_t, \mathbf{a}_{t:t+i-1}) &\leq I(\mathbf{d}_t, \mathbf{d}_{t+i}; \mathbf{a}_{t+i} | \mathbf{s}_t, \mathbf{a}_{t:t+i-1}) \\ &= I(\mathbf{d}_{t+i}; \mathbf{a}_{t+i} | \mathbf{s}_t, \mathbf{a}_{t:t+i-1}) + I(\mathbf{d}_t; \mathbf{a}_{t+i} | \mathbf{s}_t, \mathbf{a}_{t:t+i-1}, \mathbf{d}_{t+i}) \\ &= I(\mathbf{d}_{t+i}; \mathbf{a}_{t+i} | \mathbf{s}_t, \mathbf{a}_{t:t+i-1}) \qquad (2) \\ &\leq I(\mathbf{d}_{t+i}; \mathbf{a}_{t+i}, \mathbf{s}_{t+i} | \mathbf{s}_t, \mathbf{a}_{t:t+i-1}) \\ &= I(\mathbf{d}_{t+i}; \mathbf{s}_{t+i} | \mathbf{s}_t, \mathbf{a}_{t:t+i-1}) + I(\mathbf{d}_{t+i}; \mathbf{a}_{t+i} | \mathbf{s}_t, \mathbf{a}_{t:t+i-1}, \mathbf{s}_{t+i}) \\ &= I(\mathbf{d}_{t+i}; \mathbf{a}_{t+i} | \mathbf{s}_t, \mathbf{a}_{t:t+i-1}, \mathbf{s}_{t+i}) \qquad (3) \\ &= I(\mathbf{s}_t, \mathbf{a}_{t:t+i-1}, \mathbf{d}_{t+i}; \mathbf{a}_{t+i} | \mathbf{s}_{t+i}) - I(\mathbf{s}_t, \mathbf{a}_{t:t+i-1}; \mathbf{a}_{t+i} | \mathbf{s}_{t+i}) \\ &\leq I(\mathbf{s}_t, \mathbf{a}_{t:t+i-1}, \mathbf{d}_{t+i}; \mathbf{a}_{t+i} | \mathbf{s}_{t+i}) \\ &= I(\mathbf{d}_{t+i}; \mathbf{a}_{t+i} | \mathbf{s}_{t+i}) + I(\mathbf{s}_t, \mathbf{a}_{t:t+i-1}; \mathbf{a}_{t+i} | \mathbf{s}_{t+i}, \mathbf{d}_{t+i}) \\ &= I(\mathbf{d}_{t+i}; \mathbf{a}_{t+i} | \mathbf{s}_{t+i}). \qquad (4) \end{aligned}$$

□

Equation (2) holds as $\mathbf{a}_{t+i}$ is independent of $\mathbf{d}_t$ conditioned on $\mathbf{s}_t, \mathbf{a}_{t:t+i-1}, \mathbf{d}_{t+i}$. Equation (3) holds because $\mathbf{s}_{t+i}$ is

Accepted for the 39th Conference on Uncertainty in Artificial Intelligence (UAI 2023).

independent of $\mathbf{d}_{t+i}$ conditioned on $\mathbf{s}_t, \mathbf{a}_{t:t+i-1}$. By combining (1) and (4), we have

$$I(\mathbf{d}_t; \mathbf{r}_{t+1:t+N} \mid \mathbf{s}_t) \leq \sum_{i=\mathbf{0}}^{N-1} I(\mathbf{d}_{t+i}; \mathbf{a}_{t+i} | \mathbf{s}_{t+i}).$$

1.2 CONTRACTION MAPPINGS

Proposition 1.2. *There exists a contraction mapping $\mathcal{T}_{\pi,i}$ such that the following equations holds for $i = 1, 2$*

$$\begin{aligned} Z_{\pi,i}(o,a) &= \left(\mathcal{T}_{\pi,i} Z_{\pi,i}\right)(o,a), \\ \left(\mathcal{T}_{\pi,i} Z_{\pi,i}\right)(o,a) &= W_i R_o(o,a) + \Gamma_i \mathbb{E}_{\mathbf{o}',\mathbf{a}'}\left[Z_{\pi,i}(\mathbf{o}',\mathbf{a}')\right], \end{aligned} \tag{5}$$

where $W_i \in \mathbb{R}^L$, $\Gamma_i \in \mathbb{R}^{L\times L}$, $\mathbf{o}'$ is sampled with probability $P_o(\mathbf{o}'|o,a)$, $\mathbf{a}'$ is sampled with probability $\pi(\mathbf{a}'|\mathbf{o}')$, and all vectors are column vectors.

Proof. First, we provide the forms of W_i and Γ_i for $i = 1, 2$. Then, we prove $\mathcal{T}_{\pi,i}$ is a contraction mapping. Given two functions Z_1 and Z_2, we define the distance by

$$\mathbf{Dist}\left(Z_1, Z_2\right) = \max_{o,a} \max_{0\leq i\leq L-1} \left| \left[Z_1(o,a)\right]_i - \left[Z_2(o,a)\right]_i \right|.$$

(1) For directly predicting reward sequences, we have

$$W_1 = \begin{pmatrix} 1 & 0 & 0 & \cdots & 0 & 0 \end{pmatrix}^T, \qquad \Gamma_1 = \begin{pmatrix} 0 & 0 & \cdot & 0 & 0 & 0 \\ \gamma & 0 & \cdots & 0 & 0 & 0 \\ 0 & \gamma & \cdots & 0 & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ 0 & 0 & \cdots & \gamma & 0 & 0 \\ 0 & 0 & \cdots & 0 & \gamma & 0 \end{pmatrix}.$$

Then, we have

$$\begin{aligned} &\left| \left[\left(\mathcal{T}_{\pi,1} Z_1\right)(o,a)\right]_n - \left[\left(\mathcal{T}_{\pi,1} Z_2\right)(o,a)\right]_n \right| \\ \leq &\left| \left[W_1 R_o(o,a)\right]_n - \left[W_1 R_o(o,a)\right]_n \right| + \left| \gamma \mathbb{E}_{\mathbf{o}',\mathbf{a}'}\left[\left[Z_1(\mathbf{o}',\mathbf{a}')\right]_{n+1} - \left[Z_2(\mathbf{o}',\mathbf{a}')\right]_{n+1}\right] \right| \\ = &\left| \gamma \mathbb{E}_{\mathbf{o}',\mathbf{a}'}\left[\left[Z_1(\mathbf{o}',\mathbf{a}')\right]_{n+1} - \left[Z_2(\mathbf{o}',\mathbf{a}')\right]_{n+1}\right] \right| \\ \leq &\gamma \mathbb{E}_{\mathbf{o}',\mathbf{a}'}\left[\left| \left[Z_1(\mathbf{o}',\mathbf{a}')\right]_{n+1} - \left[Z_2(\mathbf{o}',\mathbf{a}')\right]_{n+1} \right|\right] \\ \leq &\gamma \mathbb{E}_{\mathbf{o}',\mathbf{a}'}\left[\mathbf{Dist}\left(Z_1, Z_2\right)\right] \\ \leq &\gamma \mathbf{Dist}\left(Z_1, Z_2\right). \end{aligned}$$

Therefore, we have

$$\mathbf{Dist}\left(\mathcal{T}_{\pi,1} Z_1, \mathcal{T}_{\pi,1} Z_2\right) \leq \gamma \mathbf{Dist}\left(Z_1, Z_2\right),$$

which implies that $\mathcal{T}_{\pi,1}$ is a contraction mapping.

(2) For predicting the DTFT of reward sequences, we have

$$W_2 = \begin{pmatrix} 1 & 1 & 1 & \cdots & 1 & 1 \end{pmatrix}^T,$$

$$\Gamma_2 = \begin{pmatrix} \gamma & 0 & \cdots & 0 & 0 & 0 \\ 0 & \gamma \exp\left(-\frac{2\pi}{L} j\right) & \cdots & 0 & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ 0 & 0 & \cdots & \gamma \exp\left(-\frac{2(L-3)\pi}{L} j\right) & 0 & 0 \\ 0 & 0 & \cdots & 0 & \gamma \exp\left(-\frac{2(L-2)\pi}{L} j\right) & 0 \\ 0 & 0 & \cdots & 0 & 0 & \gamma \exp\left(-\frac{2(L-1)\pi}{L} j\right) \end{pmatrix}.$$

Then, we have

$$\begin{aligned} & \left| \left[\left(\mathcal{T}_{\pi,2} Z_1 \right)(o,a) \right]_n - \left[\left(\mathcal{T}_{\pi,2} Z_2 \right)(o,a) \right]_n \right| \\ \leq & \left| \left[W_1 R_o(o,a) \right]_n - \left[W_1 R_o(o,a) \right]_n \right| + \left| \gamma \exp\left(-\frac{2n\pi}{L} j \right) \mathbb{E}_{\mathbf{o}',\mathbf{a}'} \left[\left[Z_1(\mathbf{o}',\mathbf{a}') \right]_n - \left[Z_2(\mathbf{o}',\mathbf{a}') \right]_n \right] \right| \\ = & \left| \gamma \exp\left(-\frac{2n\pi}{L} j \right) \mathbb{E}_{\mathbf{o}',\mathbf{a}'} \left[\left[Z_1(\mathbf{o}',\mathbf{a}') \right]_n - \left[Z_2(\mathbf{o}',\mathbf{a}') \right]_n \right] \right| \\ = & \left| \gamma \exp\left(-\frac{2n\pi}{L} j \right) \right| \cdot \left| \mathbb{E}_{\mathbf{o}',\mathbf{a}'} \left[\left[Z_1(\mathbf{o}',\mathbf{a}') \right]_n - \left[Z_2(\mathbf{o}',\mathbf{a}') \right]_n \right] \right| \\ = & \left| \gamma \mathbb{E}_{\mathbf{o}',\mathbf{a}'} \left[\left[Z_1(\mathbf{o}',\mathbf{a}') \right]_n - \left[Z_2(\mathbf{o}',\mathbf{a}') \right]_n \right] \right| \\ \leq & \gamma \mathbb{E}_{\mathbf{o}',\mathbf{a}'} \left[\left| \left[Z_1(\mathbf{o}',\mathbf{a}') \right]_n - \left[Z_2(\mathbf{o}',\mathbf{a}') \right]_n \right| \right] \\ \leq & \gamma \mathbb{E}_{\mathbf{o}',\mathbf{a}'} \left[\mathbf{Dist}\left(Z_1, Z_2 \right) \right] \\ \leq & \gamma \mathbf{Dist}\left(Z_1, Z_2 \right). \end{aligned}$$

Therefore, we have

$$\mathbf{Dist}\left(\mathcal{T}_{\pi,2} Z_2, \mathcal{T}_{\pi,2} Z_2 \right) \leq \gamma \mathbf{Dist}\left(Z_1, Z_2 \right),$$

which implies that $\mathcal{T}_{\pi,2}$ is a contraction mapping. We note that when using $Z_{\pi,2}$ as prediction target, the prediction network Z_θ needs to output $2L$-dimensional vector (L dimensions for the real part and L dimensions for the imaginary part). We actually use W_2 and Γ_2 in the following form

$$W_2 = \begin{pmatrix} 1 & 0 & 1 & 0 & \cdots & 1 & 0 \end{pmatrix}^T,$$

$$\Gamma_2 = \begin{pmatrix} \cos(\frac{0}{L}\pi) & \sin(\frac{0}{L}\pi) & 0 & 0 & \cdots & 0 & 0 \\ -\sin(\frac{0}{L}\pi) & \cos(\frac{0}{L}\pi) & 0 & 0 & \cdots & 0 & 0 \\ 0 & 0 & \cos(\frac{2}{L}\pi) & \sin(\frac{2}{K}\pi) & \cdots & 0 & 0 \\ 0 & 0 & -\sin(\frac{2}{L}\pi) & \cos(\frac{2}{K}\pi) & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & \cdots & \cos(\frac{2K-2}{L}\pi) & \sin(\frac{2K-2}{L}\pi) \\ 0 & 0 & 0 & 0 & \cdots & -\sin(\frac{2K-2}{L}\pi) & \cos(\frac{2K-2}{L}\pi) \end{pmatrix}.$$

□

1.3 LEARNING TRANSFORMS

Proposition 1.3. *For any $W \in \mathbb{R}^L$, if the infinity-norm of $\Gamma \in \mathbb{R}^{L\times L}$ is less than 1, the operator $\mathcal{T}_\pi$ defined by*

$$\left(\mathcal{T}_\pi Z_\pi\right)(o,a) = WR_o(o,a) + \Gamma \mathbb{E}_{\mathbf{o}',\mathbf{a}'}\left[Z_\pi(\mathbf{o}',\mathbf{a}')\right],$$

is a contraction mapping. The prediction target Z_π defined as the fix point of $\mathcal{T}_\pi$ satisfies the equation

$$Z_\pi(o,a) = \sum_{n=0}^{\infty}\left(\frac{\Gamma}{\gamma}\right)^n We_n(o,a;\pi).$$

Proof. Given two functions Z_1 and Z_2, we define the distance by

$$\mathbf{Dist}\left(Z_1,Z_2\right) = \max_{o,a}\max_{0\le i\le L-1}\left|\left[Z_1(o,a)\right]_i - \left[Z_2(o,a)\right]_i\right|.$$

Then, we have

$$\begin{aligned}
&\left\|\left(\mathcal{T}_\pi Z_1\right)(o,a) - \left(\mathcal{T}_\pi Z_2\right)(o,a)\right\|_\infty \\
\le&\left\|WR_o(o,a) - WR_o(o,a)\right\|_\infty + \left\|\Gamma\mathbb{E}_{\mathbf{o}',\mathbf{a}'}\left[Z_1(\mathbf{o}',\mathbf{a}') - Z_2(\mathbf{o}',\mathbf{a}')\right]\right\|_\infty \\
=&\left\|\Gamma\mathbb{E}_{\mathbf{o}',\mathbf{a}'}\left[Z_1(\mathbf{o}',\mathbf{a}') - Z_2(\mathbf{o}',\mathbf{a}')\right]\right\|_\infty \\
\le&\|\Gamma\|_\infty\cdot\left\|\mathbb{E}_{\mathbf{o}',\mathbf{a}'}\left[Z_1(\mathbf{o}',\mathbf{a}') - Z_2(\mathbf{o}',\mathbf{a}')\right]\right\|_\infty \\
\le&\|\Gamma\|_\infty\cdot\mathbf{Dist}\left(Z_1,Z_2\right).
\end{aligned}$$

Therefore, we have

$$\mathbf{Dist}\left(\mathcal{T}_\pi Z_2,\mathcal{T}_\pi Z_2\right) \le \|\Gamma\|_\infty\mathbf{Dist}\left(Z_1,Z_2\right).$$

Because $\|\Gamma\|_\infty$ is less than 1, we have that $\mathcal{T}_\pi$ is a contraction mapping.

$$\left(\mathcal{T}_\pi Z_\pi\right)(o,a) = WR_o(o,a) + \Gamma \mathbb{E}_{\mathbf{o}',\mathbf{a}'}\left[Z_\pi(\mathbf{o}',\mathbf{a}')\right],$$

Then, by the definition of the fix point, we have

$$\begin{aligned}
Z_\pi(o,a) &= WR_o(o,a) + \Gamma\mathbb{E}_{\mathbf{o}',\mathbf{a}'}\left[Z_\pi(\mathbf{o}',\mathbf{a}')\right] \\
&= WR_o(o,a) + \Gamma\mathbb{E}_\pi\left[Z_\pi(\mathbf{o_1},\mathbf{a_1}) \mid \mathbf{o_0}=o,\mathbf{a_0}=a\right] \\
&= WR_o(o,a) + \Gamma\mathbb{E}_\pi\left[WR_o(\mathbf{o_1},\mathbf{a_1}) + \Gamma Z_\pi(\mathbf{o_2},\mathbf{a_2}) \mid \mathbf{o_0}=o,\mathbf{a_0}=a\right] \\
&= WR_o(o,a) + \Gamma W\mathbb{E}_\pi\left[R_o(\mathbf{o_1},\mathbf{a_1}) \mid \mathbf{o_0}=o,\mathbf{a_0}=a\right] + \Gamma\mathbb{E}_\pi\left[Z_\pi(\mathbf{o_2},\mathbf{a_2}) \mid \mathbf{o_0}=o,\mathbf{a_0}=a\right] \\
&\cdots \\
&= \sum_{n=1}^{\infty}\Gamma^n W\mathbb{E}_\pi\left[R_o(\mathbf{o_n},\mathbf{a_n}) \mid \mathbf{o_0}=o,\mathbf{a_0}=a\right] \\
&= \sum_{n=1}^{\infty}\left(\frac{\Gamma}{\gamma}\right)^n We_n(o,a;\pi).
\end{aligned}$$

We note that the infinite sum in RHS converges. The reason is that $\sum_{n=N}^{\infty}\left(\frac{\Gamma}{\gamma}\right)^n We_n(o,a;\pi)$ converges to zero vector, as

$$\left\|\sum_{n=N}^{\infty}\left(\frac{\Gamma}{\gamma}\right)^n We_n(o,a;\pi)\right\|_\infty \le R_{max}\sum_{n=N}^{\infty}\left\|\Gamma^n W\right\|_\infty \le R_{max}\left\|W\right\|_\infty\frac{\left\|\Gamma\right\|_\infty^N}{1-\left\|\Gamma\right\|_\infty}.$$

□

2 DETAILS FOR EXPERIMENTS IN SECTION 4.1

In this section, we provide additional information about the experiments in Section 4.1.

2.1 EXPERIMENTAL SETTING

We evaluate all auxiliary tasks in a modified Cartpole Swingup environment. In each episode, the background images are sampled from two videos and then kept fixed through the whole episode. We label an observation according to the video that its background image is sampled from. We use the InfoNCE objective to estimate the mutual information $I\left(\phi_\theta(\mathbf{o_t}); \mathbf{s_t}\right)$. That is,

$$\mathcal{J}_{NCE} = -\mathbb{E}\left[\log\left(\frac{\|f_w(\phi_\theta(\mathbf{o}_i)) - f'_w(\mathbf{s}_i)\|_2^2}{\sum_{i=k}^{N}\|f_w(\phi_\theta(\mathbf{o}_k)) - f'_w(\mathbf{s}_k)\|_2^2}\right)\right],$$

where f_w and f'_w are two networks, $((\mathbf{o}_1, \mathbf{s}_1), \cdots, (\mathbf{o}_N, \mathbf{s}_N))$ is a batch of samples. We train these two networks via maximizing $\mathcal{J}_{NCE}$. We use the final loss as an estimate for $I\left(\phi_\theta(\mathbf{o_t}); \mathbf{s_t}\right)$. We train a network with a cross-entropy loss to predict background images and use the loss to estimate the mutual information $I\left(\phi_\theta(\mathbf{o_t}); \mathbf{d_t}\right)$. That is,

$$\mathcal{J}_{CE} = -\mathbb{E}\left[\frac{1}{N}\sum_{i=1}^{N}\log q_w(\mathbf{d}_i|\phi_\theta(\mathbf{o}_i))\right],$$

where q_w is the classification network, and $\mathbf{d}_i$ is the label of the background image. We train the network q_θ by minimizing J_{CE}. We use $\log 2 - \mathcal{J}_{CE}$ as an estimate of $I\left(\phi_\theta(\mathbf{o_t}); \mathbf{d_t}\right)$. For all methods, we optimize the policy network and value network via 200K-step online training and then estimate the mutual information using the saved data. Note that all auxiliary tasks are combined with DrQ.

2.2 ADDITIONAL RESULTS

We show the performance during training in Figure 1. Results show RSP significantly outperforms other auxiliary tasks. In our experiments, the VAE-based auxiliary task tends to minimize reconstruction losses by reconstructing the background images as shown in Figure 2.

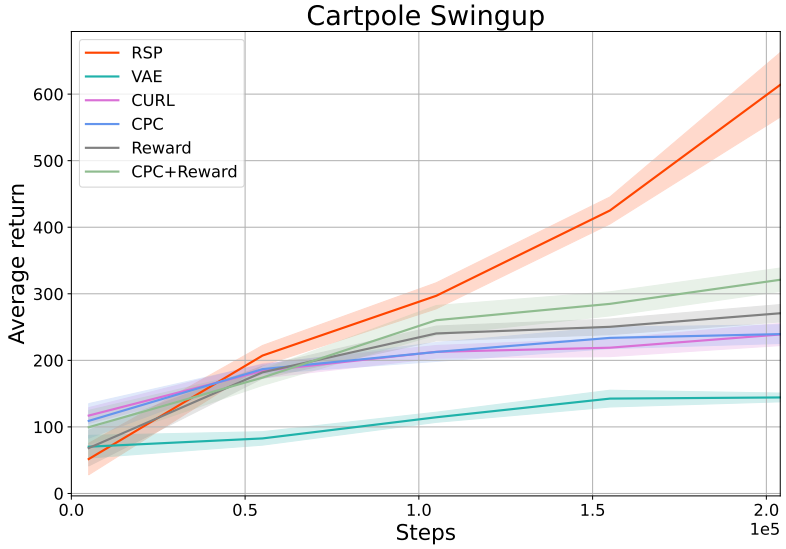


Figure 1: The performance of different auxiliary tasks during training in the Cartpole Swingup task.

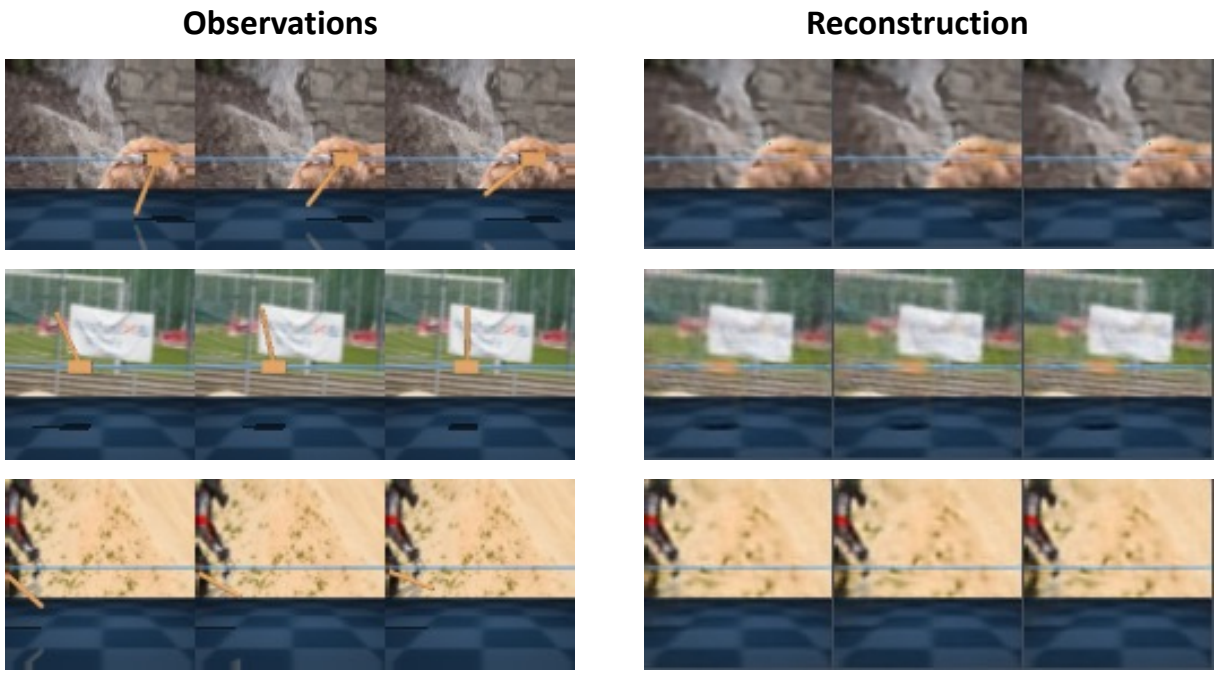


Figure 2: Ground-truth and reconstructed images. Results show that representations learned by VAE mainly encode information about the background images, which is irrelevant to the control task.

	BiC-Catch	C-Swingup	C-Run	F-Spin	R-Easy	W-walk
DrQ (DCS)	138 ± 20	334 ± 29	4 ± 2	378 ± 125	113 ± 22	28 ± 1
DrQ (Our)	99 ± 99	341 ± 52	211 ± 64	543 ± 245	168 ± 63	30 ± 8

Table 1: Comparison between different implementations in multi-distraction environments. Our implementation achieves similar or better performance than that used in DCS.

3 EXPERIMENTS

3.1 MULTI-DISTRACTION SETTING

Figure 3 shows the snapshots of all six environments. In these tasks, robots face multiple distractions at the same. We implement DrQ and DrQv2 using Pytorch. We run all experiments in one GPU, Geforce 2080Ti. Our implementation of DrQ is slightly different from the official implementation of DrQ. (1) Our implementation does not use a target encoder similar to DrQv2. (2) We use a small batchsize 256 instead of 512 for a fair comparison. (3) We use a small learning rate 5e-4 instead of 1e-3. (4) We use a large replay buffer whose size is 500K instead of 100K. The first two modifications are to improve the computational efficiency while the last two are to unify the hyperparameters used in DrQ and DrQv2. In Tabel 1, we provide a comparison between our implementation and that used in DCS. Note that these modifications do not reduce the performance of DrQ, and even improve it in some environments. Our implementation of DrQv2 also uses batchsize 256 and learning rate 5e-4 for a fair comparison. Moreover, for DrQv2, we set the action repeat hyperparameter the same as DrQ. We directly use the official implementation of DBC, TIA, and TPC for experiments in Section 6.1. All hyperparameters of RSP are listed in Table 2. Please note that RSP also predicts one-step rewards and we regularize the outputs of policy networks by l2 norm with a small coefficient 0.01. The action regularization can control the task-irrelevant information used by the early exploration policy. This trick can not improve the performance of DrQ/DrQv2, but can reduce the performance variance of RSP with different random seeds in some tasks. For computation efficiency, we use a smaller batchsize 128 instead of 256 for all ablation studies.

Figure 3: The six environments used in our Section. Agents face the camera distractions, color distractions, and background distractions simoutenously.

Hyperparameter	**Setting**
Input dimension	$3\times84\times84$
Stacked frames	3
Discount factor	0.99
Replay buffer size	500K
Batch size	256
learning rate	5e-4
Random cropping padding	4
Seed steps	4000
Encoder conv layers	4
Encoder conv strides	[2,1,1,1]
Encoder conv channels	32
Encoder feature dim	50
Actor/Critic head MLP layers	3
Actor/Critic head MLP hidden dim	1024
Actor update frequency	2
Critic target update frequency	2
Critic soft-update rate	0.01
DrQv2: noise schedule	linear(1.0, 0.1, 500000)
* RSP network: prediction layers	3
* RSP network: hidden dim	256
* RSP network: output dim	1024
* RSP: share the first linear layer	True
* RSP: stop gradients of RL losses	True
* RSP: prediction target	$Z_{\pi,1}$ in R-Easy, BiC-Catch, W-Walk $Z_{\pi,2}$ in C-Swingup, F-Spin, C-Run

Table 2: Hyperparameters were used in our experiments. The marker * means the extra hyperparameters used in RSP. The noise schedule "linear(1.0, 0.1, 500000)" used for DrQv2 means that the exploration noise decays linearly from 1.0 to 0.1 after 500K environment steps.

3.2 RESULTS IN NO-DISTRACTION ENVIRONMENTS

Many methods considering distraction perform worse than DrQ in standard DMC environments. However, the final scores of DrQ+RSP is comparable with those of DrQ (Table 3).

	BiC-Catch	C-Swingup	C-Run	F-Spin	R-Easy	W-Walk
DrQ	963 ± 9	868 ± 10	660 ± 96	938 ± 103	942 ± 71	921 ± 45
DrQ+RSP	963 ± 7	864 ± 12	642 ± 46	981 ± 3	973 ± 4	950 ± 18

Table 3: 500K step scores in no-distraction environments.

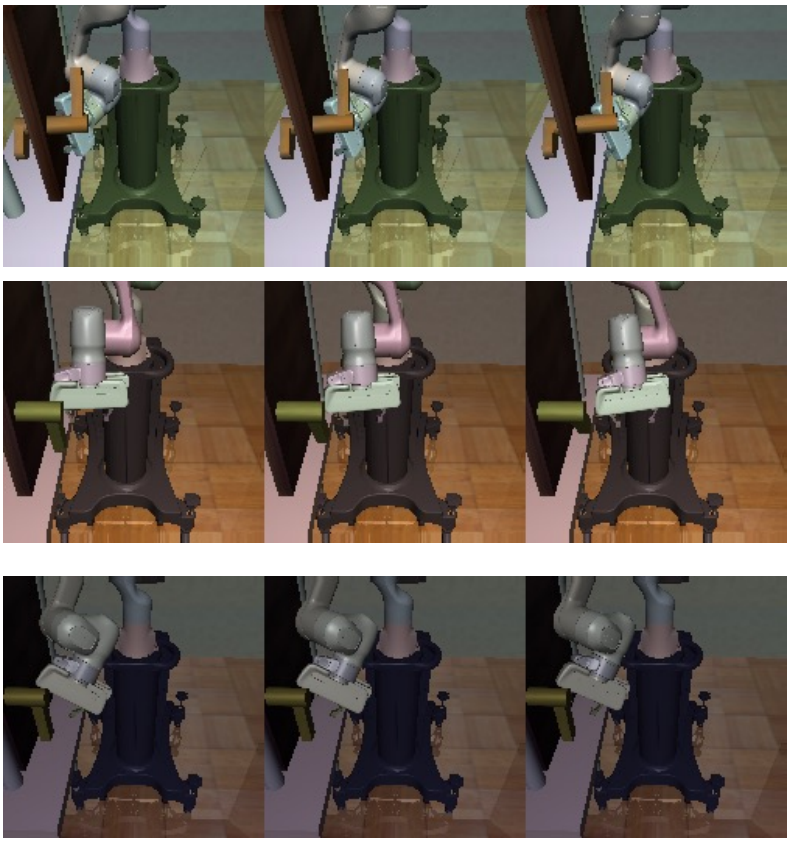


Figure 4: Door Opening in Robosuite benchmarking. We illustrate observations in different episodes.

3.3 COMPARISON IN DOOR OPENING

Here, we compare DrQv2+RSP and DrQv2 in a Robosuite task, Door Opening. Compared with DCS environments, the Door Opening environment simulates a more realistic robotic scenario, where a robot arm learns to turn a handle and then open the door. The dimension of observations ($3 \times 168 \times 168$) in Door Opening is also higher than that ($3 \times 84 \times 84$) in DCS. In our experiments, three kinds of distractions exist during the training phase, including color, light, and camera distractions. We illustrate the environment in Figure 4. We use hyperparameters similar to that of SECANT. The hyperparameters different from Table 2 are shown in Table 4. We report results over five random seeds. Figure 5 shows the performance after 150K environment steps (300 episodes). RSP provides significant performance improvement ($+736\%$) in the Door Opening task.

Hyperparameter	Setting
Input dimension	$3\times168\times168$
Episode length	500
Policy learning rate	1e-3
Critic learning rate	1e-4
Random cropping padding	8
Seed steps	2000
Critic target update frequency	4
Regularization coef	0.05
DrQv2: noise schedule	linear(1.0, 0.1, 100000)
RSP: prediction target	$Z_{\pi,1}$

Table 4: Hyperparameters used in the Door Opening task.

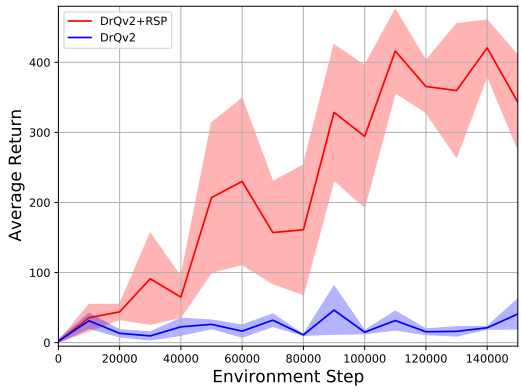


Figure 5: Comparision between DrQv2+RSP and DrQv2 in the Door Opening environment. Results show that RSP significantly improvide the sample efficiency and final performance.

3.4 ABLATION RESULTS AND VISUALIZATION

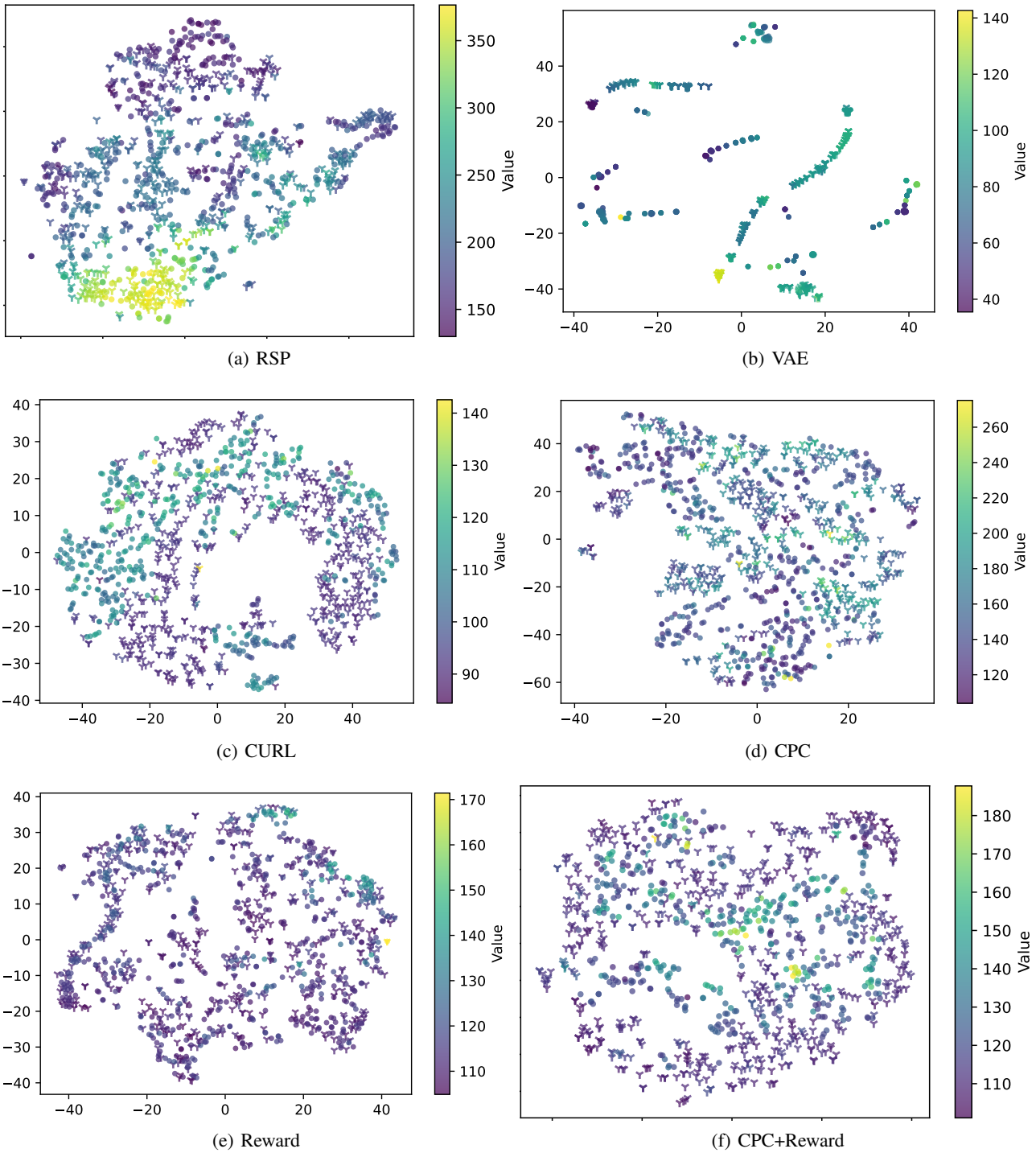
Figure 6 visualize the latent spaces learned by six different auxiliary tasks using the data the same as that used in Section 41. Figure 7 and 8 provide more comparisons between RSP and CPC+Reward.

3.5 COMPARISON BETWEEN VARIANTS OF RSP

In Figure 9 and 10, we provide comparisons between the prediction targets $Z_{\pi,1}$ and $Z_{\pi,2}$, which corresponding to **direct** and **Fourier** respectively. Figure 11-14 further visualize reward sequences and state sequences to understand potential reasons why **Fourier** outperforms **direct** in some tasks. The results show that the variant **Fourier** outperforms **direct** in Cheetah Run, Cartpole Swingup, and Finger Spin environments. We observe the approximate periodicity of reward sequences in Cartpole Swingup and Finger Spin. We do not observe the periodicity of reward sequences in Cheetah Run. However, some dimensions of states are approximate periodic. We hypothesize that **Fourier** outperforms **direct** in Cheetah Run due to the approximate periodicity of states. In the Ball in Cup environment, we do not observe periodicity of reward sequences or state sequences. Therefore, **Fourier** performs worse than **direct** in the Ball in Cup environment.

3.6 COMPLETE RESULTS OF OUR BASELINES

This part provides complete results (15) of our baselines in multi-distraction settings. Results show that the baselines hardly improve the performance compared with DrQ/DrQv2, indicating the difficulty of learning representations when multiple distractions exist.



(a) RSP (b) VAE

(c) CURL (d) CPC

(e) Reward (f) CPC+Reward

Figure 6: Embedding spaces learned by different auxiliary tasks. Results show that RSP can capture information about state values and tends to map observations with different background images to the same region.

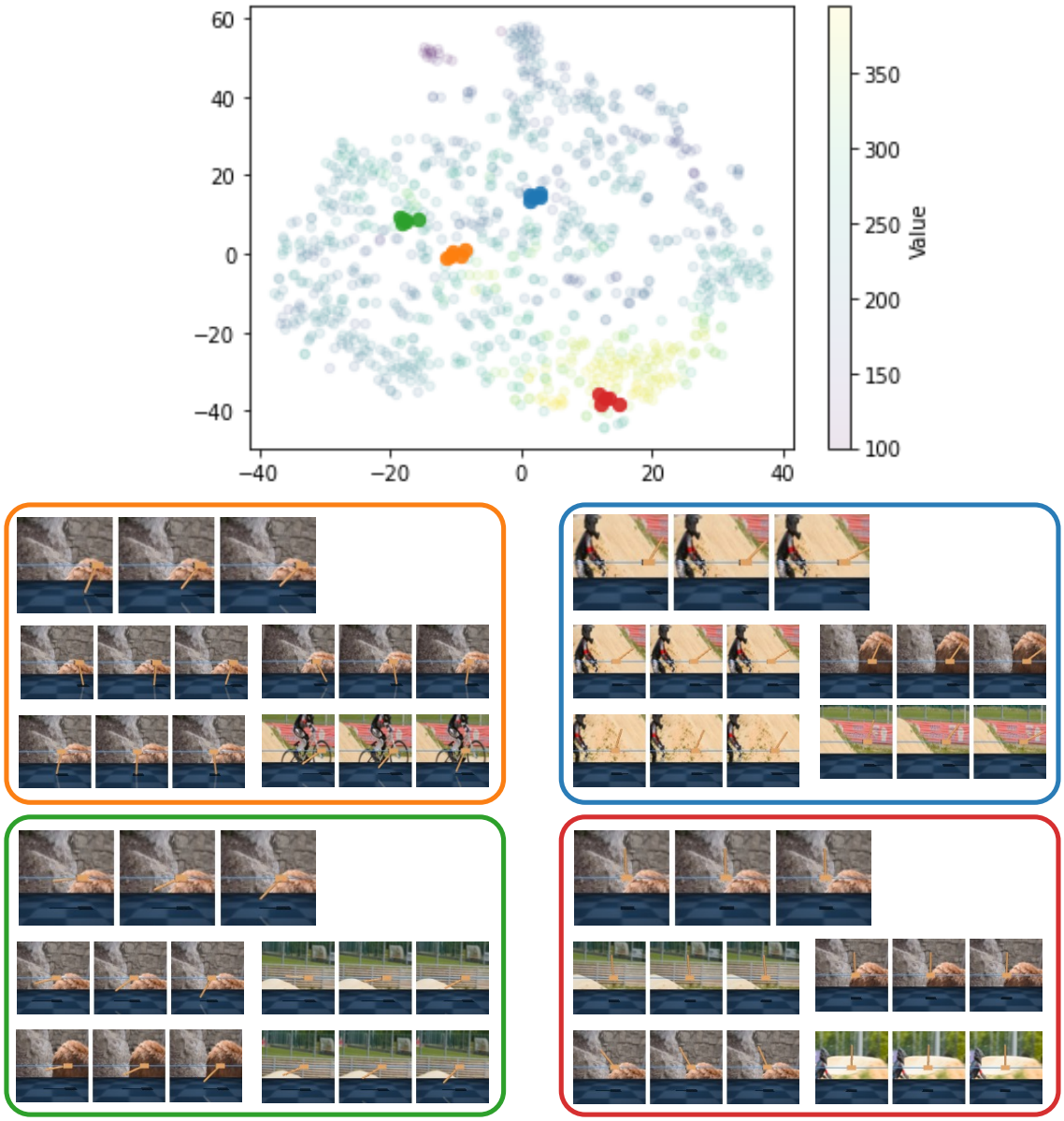


Figure 7: T-SNE of the embedding space learned by RSP. We randomly sample four observations (corresponding to four colors in the T-SNE figure) and match them with their nearest neighbors respectively (shown in the bottom subfigures). The results show that neighboring points in the embedding space learned by RSP metric have similar states.

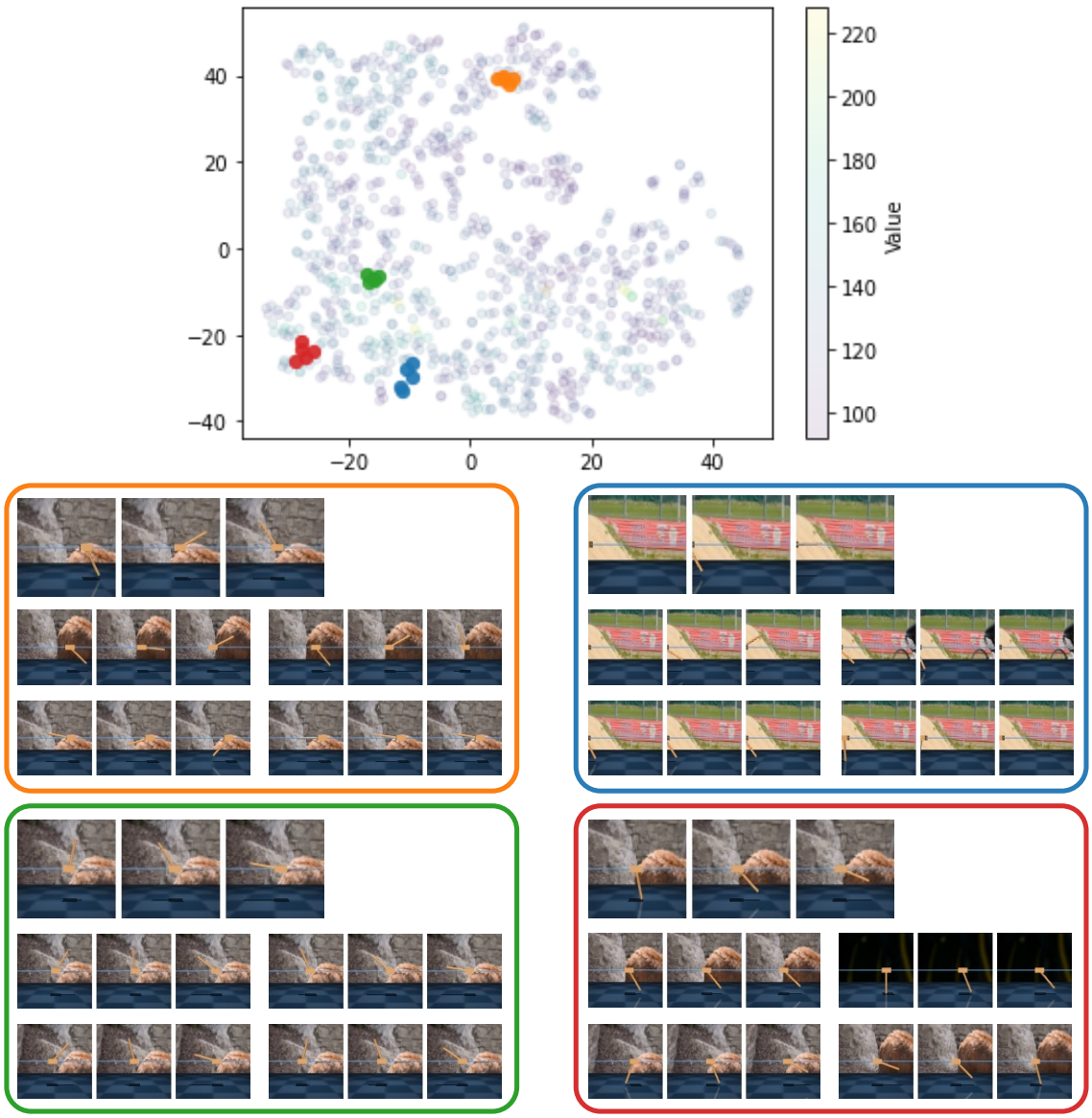


Figure 8: T-SNE of the embedding space learned by CPC+Reward. CPC+Reward tends to map observations with similar background images to neighboring regions, even though those observations may corresponds to dissimilar states.

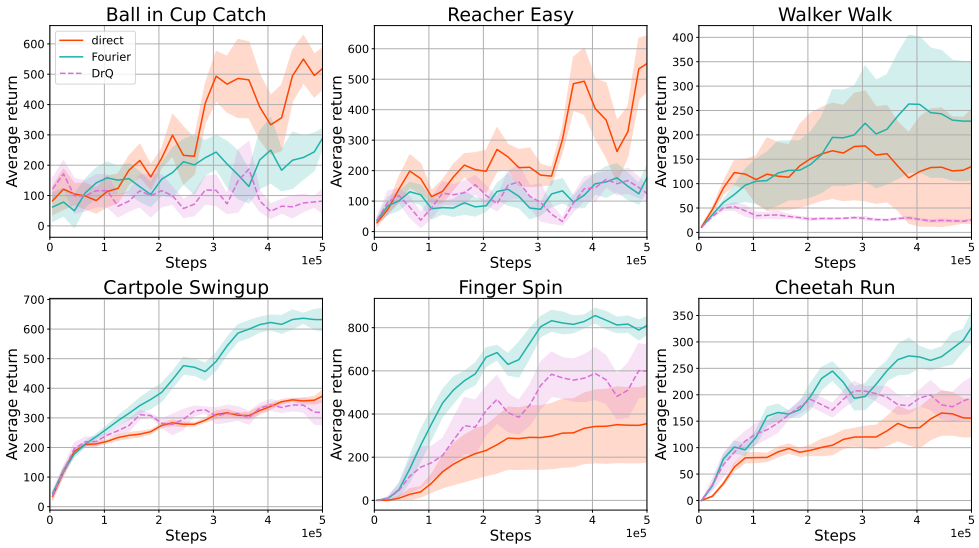


Figure 9: Comparison between the variants **direct** and **Fourier** based on DrQ.

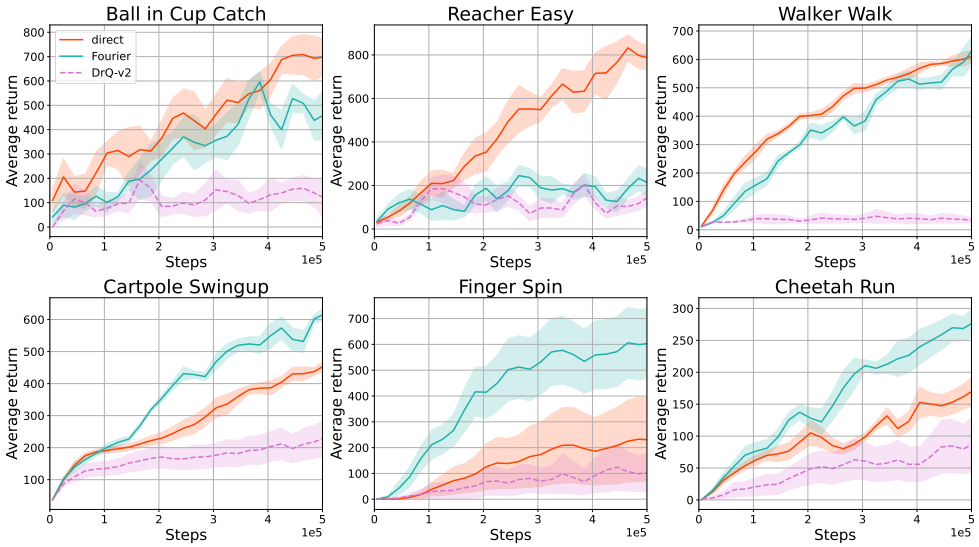


Figure 10: Comparison between the variants **direct** and **Fourier** based on DrQv2.

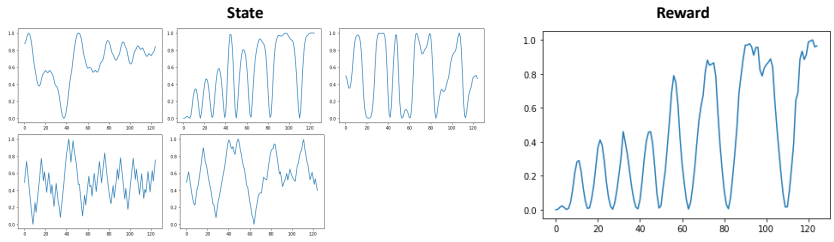


Figure 11: State sequences and the corresponding reward sequence in the Cartpole Swingup task.

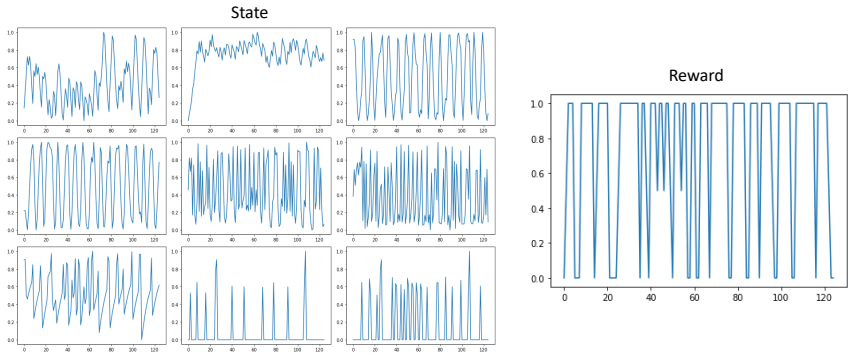


Figure 12: State sequence and the corresponding reward sequence in the Finger Spin task.

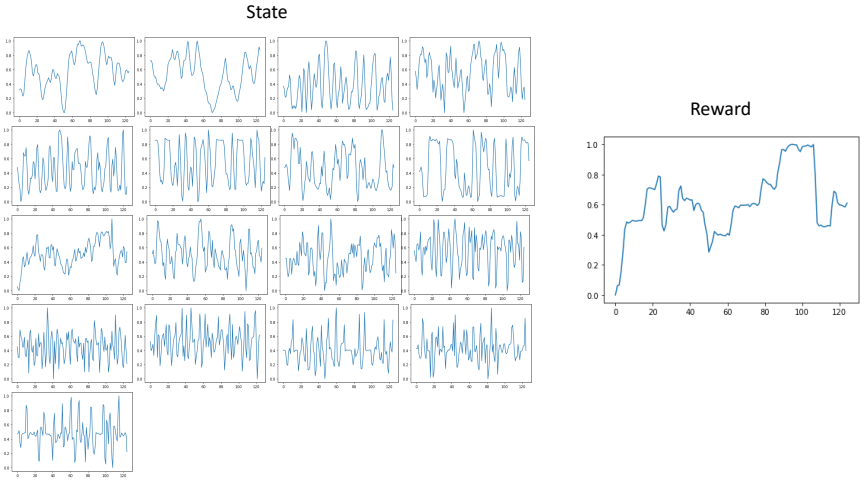


Figure 13: State sequences and the corresponding reward sequence in the Cheetah Run task.

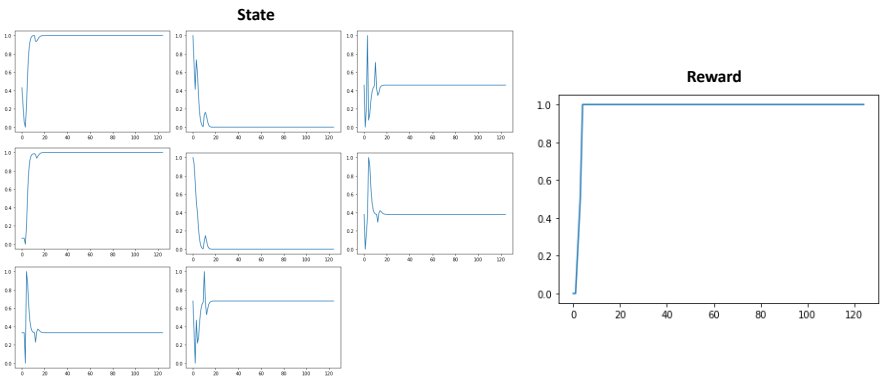


Figure 14: State sequences and the corresponding reward sequence in the Ball in Cup Catch task.

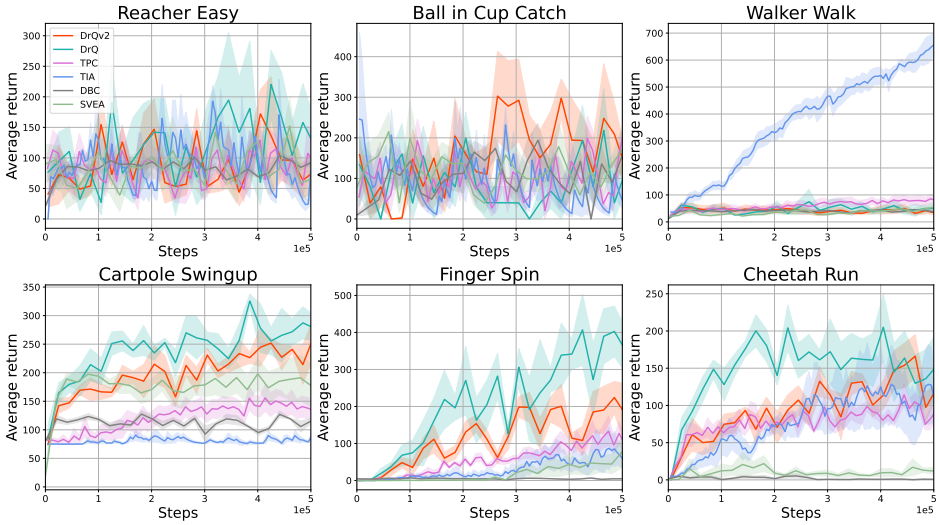


Figure 15: Complete results of our baselines in multi-distraction settings.