

An Efficient Generalizable Framework for Visuomotor Policies via Control-aware Augmentation and Privilege-guided Distillation

Yinuo Zhao^{1,*}, Kun Wu^{2,*}, Tianjiao Yi¹, Zhiyuan Xu³, Xiaozhu Ju³, Zhengping Che³,
Qinru Qiu², Chi Harold Liu¹, Jian Tang³

Abstract

Visuomotor policies, which learn control mechanisms directly from high-dimensional visual observations, confront challenges in adapting to new environments with intricate visual variations. Data augmentation emerges as a promising method for bridging these generalization gaps by enriching data variety. However, straightforwardly augmenting the entire observation shall impose excessive burdens on policy learning and may even result in performance degradation. In this paper, we propose to improve the generalization ability of visuomotor policies as well as preserve training stability from two aspects: 1) We learn a control-aware mask through a self-supervised reconstruction task with three auxiliary losses and then apply strong augmentation only to those control-irrelevant regions based on the mask to reduce the generalization gaps. 2) To address training instability issues prevalent in visual reinforcement learning (RL), we distill the knowledge from a pretrained RL expert processing low-level environment states, to the student visuomotor policy. The policy is subsequently deployed to unseen environments without any further fine-tuning. We conducted comparison and ablation studies across various benchmarks: the DMControl Generalization Benchmark (DMC-GB), the enhanced Robot Manipulation Distraction Benchmark (RMDDB), and a specialized long-horizontal drawer-opening robotic task. The extensive experimental results well demonstrate the effectiveness of our method, e.g., showing a 17% improvement over previous methods in the video-hard setting of DMC-GB.

1. Introduction

Visuomotor policy is designed to predict precise actions and complete tasks based on high-dimensional pixel input in an end-to-end manner. It has received significant attention

and been successfully applied in the field of robot learning and embodied AI, including robot manipulation [4, 26], autonomous navigation [1], and locomotion tasks [52, 55]. Although visual reinforcement learning (VRL) has shown promising results for learning such a visuomotor policy, recent studies [7, 12, 15, 45] indicated the fragility of VRL agents in the training process, not to mention generalizing to new environments.

One promising solution for addressing the aforementioned challenges is Data Augmentation (DA) [11, 20, 44, 52, 53]. In contrast to weak augmentations, such as random cropping, rotation, and flipping, which provide only limited generalization improvement, strong augmentations like *random conv* [34] and *random overlay* [19] have demonstrated the potential to significantly diversify data and enhance the generalization capabilities of models. Followed by these techniques, many algorithms [5, 11, 20] were proposed to learn consistent visual features or control values between the augmented observations and the original ones. However, strong augmentations change not only the control-irrelevant context but also the control-relevant information, potentially destroying the inherent structures and dynamics of the environment contained in the observations. This usually increases the complexity of the training model and affects stability in both learning and the testing results. Recent advancements aimed at augmenting more specific regions within the observation space. However, these approaches typically depend on the availability of informative reward functions [2, 13] or are limited to identifying only dynamic objects [51]. Identifying control-related pixels in intricate tasks remains a significant challenge, especially in long-horizontal tasks with uninformative rewards.

To improve the generalization ability while maintaining stability, in this paper, we propose an efficient **GE**neralizab**le** fra**ME**work for visu**OM**otor policies (GEMO) to identify the control-related information and enable zero-shot deployment to unseen environments. More specifically, GEMO comprises two jointly optimized modules: 1) a control-aware augmentation module adaptively learns a mask highlighting the control-related pixels with spatial-temporal data from the distillation mod-

¹Beijing Institute of Technology {ynzhao, tjyi}@bit.edu.cn, liuchi02@gmail.com

²Syracuse University {kwu102, qiqiu}@syr.edu

³Midea Group {xuzy70, juxz, chezp, tangjian22}@midea.com

*These authors contributed equally.

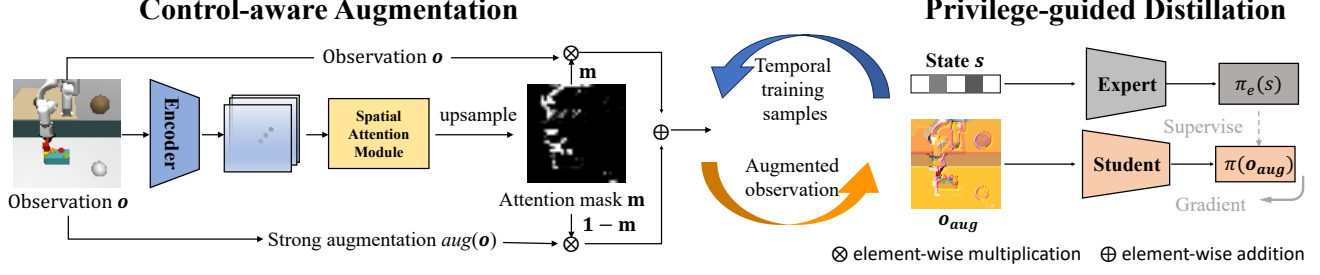


Figure 1. Overview of our method.

ule. We construct a self-supervised reconstruction task in this module with three auxiliary losses for efficient training; 2) A privilege-guided distillation module takes control-aware augmented observations and distills insightful control knowledge from a reliable expert agent to the visuomotor policy. The expert agent is pre-trained through standard deep reinforcement learning algorithm and receives low-level environment states as privileged input. It is worth noting that this privileged information is readily obtainable [3, 14, 33, 39, 43] in a robotic simulated environment, and would only be accessed during training.

To evaluate the proposed method, we conducted experiments on three generalization benchmarks, including the commonly used DMControl Generalization Benchmark [19] and our enhanced Robot Manipulation Distraction Benchmark based on [24] with visual changes on the control-irrelevant background and distractors. To further demonstrate the generalization on long-horizontal robotic tasks, we build a challenging drawer-operation generalization platform on NVIDIA Isaac Sim [37]. Extensive experimental results show that our method advances the state-of-the-art in terms of generalization ability.

We summarize our contributions as follows: 1) We propose GEMO : an efficient **GE**neralizable fra**ME**work for visu**OM**otor policies to achieve zero-shot generalization to unseen environments. 2) We propose to obtain a control-aware mask with the self-supervised reconstruction structure and an auxiliary control prediction loss without the need for extra labels or reward signals. 3) To reduce the variance, we propose to learn the visuomotor policies by distilling from a privileged expert pre-trained with a low-level environment state. 4) Extensive comparison and ablation results on three benchmarks demonstrate the effectiveness of GEMO.

2. Method

In this section, we propose an efficient **GE**neralizable fra**ME**work for visu**OM**otor policies (GEMO). The overall objective of GEMO is to learn robust representations to enhance zero-shot generalization capabilities. As shown

in Fig 1, GEMO comprises two simultaneously optimized modules: a control-aware augmentation module and a privilege-guided distillation module. Retrieving temporal data from the replay buffer, the former involves a self-supervised reconstruction task, along with three auxiliary losses, to identify control-relevant pixels. Following the specified expanded the observation input, the latter module distills knowledge from a pretrained DRL expert (processes only environment states) to the visuomotor student network (processes only image observations). After training, the visuomotor policy network can be reliably deployed to more complex environments with significant visual changes, requiring neither fine-tuning nor additional supervision.

2.1. Control-aware data augmentation

The control-aware data augmentation module aims to learn an attention mask that can efficiently distinguish the control-related regions from other control-irrelevant ones (such as static background and task-unrelated objects). In this way, we can boldly use strong augmentation without fear of corrupting essential information for policy learning, thus enhancing generalization ability. To achieve this, we develop a lightweight mask-generating model based on Convolutional Block Attention [48] (CBA) and learn a clear attention mask in a self-supervised manner.

As shown in Figure 2, the control-aware data augmentation module consists of: 1) an image encoder $f_e(\cdot)$ takes the observation o as input and outputs a feature map z , 2) a spatial attention block $f_a(\cdot)$ takes the feature map z as input and outputs the attention mask m , 3) a decoder $f_d(\cdot)$ takes the feature map z as input and outputs the reconstructed image \hat{o} , 4) a control prediction module $f_{ctl}(\cdot)$ takes the control-relevant feature maps z as input and outputs the predicted control \hat{a} . In the following, we describe how to get the control-related and control-irrelevant features. Then, we introduce a self-supervised reconstruction task with three auxiliary losses to identify the vital control-related pixels.

Firstly, we randomly sample a source image o_t from the replay buffer and obtain its next transition o_{t+1} as the target image. By utilizing the image encoder $f_e(\cdot)$ and spatial block $f_a(\cdot)$, we derive attention masks $m_t = f_a(f_e(o_t))$

and $m_{t+1} = f_a(f_e(o_{t+1}))$ for the source and target images, respectively. $z_t \otimes m_t$ and $z_{t+1} \otimes m_{t+1}$ specifically represent the control-related features for the source and target images. Similar to [51], we synthesize the latent feature \hat{z}_{t+1} for target image by

$$\hat{z}_{t+1} = z_{t+1} \otimes m_{t+1} + z_t \otimes (1 - m_t) \otimes (1 - m_{t+1}), \quad (1)$$

where the operator \otimes denotes element-wise multiplication across the channel. By multiplying the $1 - m_{t+1}$ in the second term, we can reduce the interference from the intersection part of $1 - m_t$ and m_{t+1} , thus constructing a more accurate latent feature for the target image. Then we synthesize the target image \hat{o}_{t+1} by feeding the latent feature \hat{z}_{t+1} into the decoder $f_d(\cdot)$. The reconstruction loss is computed as follows:

$$\mathcal{L}_{rec}(o_t, o_{t+1}) = \|f_d(\hat{z}_{t+1}) - o_{t+1}\|_2^2. \quad (2)$$

The vanilla reconstruction task is insufficient to learn a clear control-aware mask [51]. Therefore, we introduce three auxiliary losses. First, we add an auto-encoder loss \mathcal{L}_{ae} by directly reconstructing from the target feature z_{t+1} to capture more accurate latent information. The auto-encoder loss is computed as follows:

$$\mathcal{L}_{ae}(o_{t+1}) = \|f_d(z_{t+1}) - o_{t+1}\|_2^2. \quad (3)$$

Second, to extract essential control-related regions, we construct a control prediction loss \mathcal{L}_{ctl} to predict the control from the control-related features from the source and target images. $\hat{a}_t = f_{ctl}(o_t \otimes m_t, o_{t+1} \otimes m_{t+1})$.

$$\mathcal{L}_{ctl}(o_t, o_{t+1}) = \|\hat{a}_t - a_t\|_2^2. \quad (4)$$

Last, we add a sparsity penalty loss \mathcal{L}_{sps} to control the generated attention mask flexibly. $\mathcal{L}_{sps} = \|m_j\|_1$. The overall loss function is defined as follows:

$$\mathcal{L}_{att} = \mathbb{E}[\mathcal{L}_{rec} + \mathcal{L}_{ae} + \beta \mathcal{L}_{ctl} + \lambda \mathcal{L}_{sps}]. \quad (5)$$

The image encoder $f_e(\cdot)$, spatial attention block $f_a(\cdot)$, decoder $f_d(\cdot)$, and control prediction model f_{ctl} are optimized simultaneously. Note that to stabilize the training, we stop the gradient propagation along the branch of the source image o_t , which is shown as slashes in Fig 2. To apply observation-level control-aware attention mask m , we directly up-sample it to the observation scale and only augment the control-irrelevant regions. The augmented image o_{aug} is obtained as follows:

$$o_{aug} = o \otimes m + aug(o) \otimes (1 - m), \quad (6)$$

where $aug(\cdot)$ is the strong augmentation operation like *random conv* [34] and *random overlay* [19]. In this way, the control-irrelevant pixels can be augmented, while control-related pixels can be preserved for decision making.

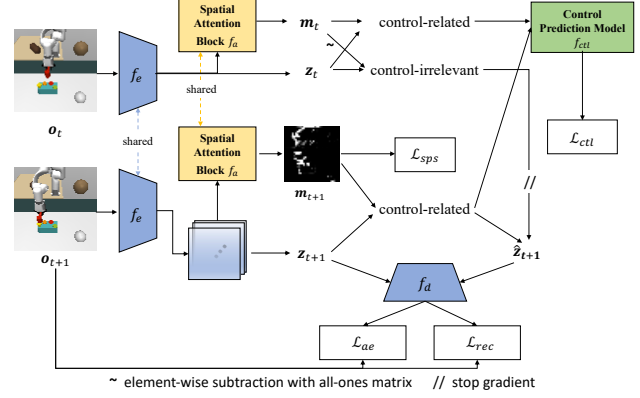


Figure 2. Attention-based self-supervised model.

2.2. Privilege-guided Policy Distillation

While strong augmentation introduces diverse inductive biases to reduce the generalization gaps, it also makes policy optimization much more difficult [11]. For instance, the agent must learn to output the same action for two completely different augmented observations derived from the original observation. This requirement causes a high variance in value function estimations for any standard RL algorithm and makes the training process unstable. In the control-related policy distillation module, the goal is to distill the control-related knowledge from the expert $\pi_e(a|s)$ to the visual policy $\pi(a|o)$ in an imitation learning manner, thus stabilizing the training process.

We first train a control-related expert policy $\pi_e(a|s)$ directly using the state information s . Note that using privileged control-related information s is reasonable [3, 33] because the training environments are usually built in a controllable simulator, and this privileged control-related information is available. For instance, in LBC [3], the authors distill privileged information like object positions, road features, and traffic lights to a sensorimotor agent for autonomous driving tasks. The control-related expert policy $\pi_e(a|s)$ is trained in a standard RL paradigm and can be any RL algorithm. In practice, we build our implementation on the top of DrQv2 [52] due to its broad adoption in continuous control tasks [18, 49, 58].

To distill the control-related knowledge from the expert to the visual policy and eliminate the high variance of value function estimations in RL algorithms, we directly train the visual policy in an imitation learning manner as follows:

$$\mathcal{L}_\pi = \mathbb{E}_{(o,s) \sim \mathcal{D}}[(\pi(o_{aug}) - \pi_e(s))^2]. \quad (7)$$

Similar to SECANT [11], we alternatively choose the π_e and π to collect samples from the environment and store them into a replay buffer \mathcal{D} . At each iteration, we simultaneously 1) update the context-aware data augmentation module using Equation 5, and 2) update the visual policy.

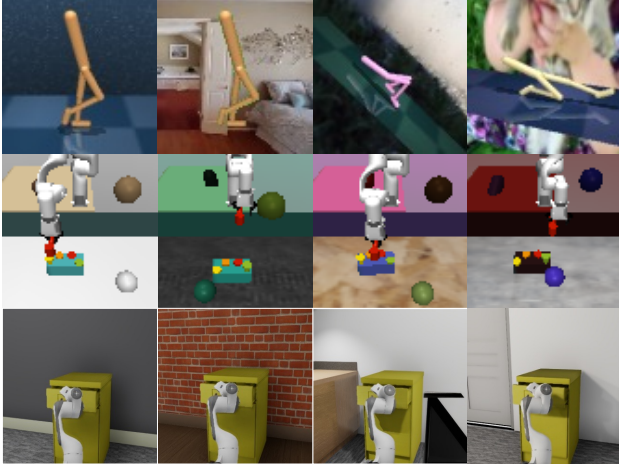


Figure 3. Observation examples from three benchmarks. Top row: Walker-walk task from DMC-GB (first: training, second: video hard setting, last two: distraction setting). Medium row: Hammer task from our enhanced Robot Manipulation Distraction Benchmark (first: training, last three: testing). Bottom row: Our self-designed Drawer Opening Generalization Benchmark (first: training, second: different backgrounds, last two: different locations).

3. Experiments

Experiment setup. To demonstrate the zero-shot generalization capability of GEMO, we conducted experiments in three domains, as shown in Fig. 3: A standard benchmark named DMControl Generalization Benchmark (DMC-GB) [19] requires the agent trained in a simple background to generalize to a more complex environment. We use three settings for evaluation: video-easy, video-hard, and distraction. In the video-hard setting, the background is substituted with real-world natural videos that deviate significantly from the original. In the distraction setting, variations in difficulty level involve changes to the background image, foreground color, and camera view. All methods are trained with 500k iterations; only visual input is given in testing environments. 2) Our enhanced Robot Manipulation Distraction Benchmark (RMDB). The original Robot Manipulation Benchmark [2, 24] has four practical robot tasks - Push, Reach, Hammer, and Pegbox. We added distracting objects with static color in training but various colors/textures in five testing environments. RMDB is more challenging compared to the original one because the robot is required to learn a robust visuomotor policy against visual changes on all task-irrelevant areas (i.e., the static background, the dynamic distraction objects, and part of the manipulated object, as shown in the medium row of Fig. 3, the box is changed to purple and brown). 3) A Drawer Opening Generalization Benchmark (DOGB) based on high-fidelity NVIDIA Isaac Sim. The goal is to perform long-horizontal drawer-opening tasks against large-scale background variance and different locations (where unseen environment

structures are shown in the background). If the opening distance of the drawer is longer than 0.29 meters within 500 time steps, the task ends with a success status. Compared with RMDB, the tasks in DOGB face the challenge of more fine-grained control (of the 7-DoF robot arm and the 2-DoF grasp) and a significantly diverse background change (up to 400 with the change of colors, patterns, and materials of the wallpaper and the carpet).

Baselines. We compared GEMO to the several SOTA algorithms [2, 19, 20, 31, 51–53, 56] in terms of generalization ability. Apart from these SOTA methods, we developed a strong baseline SAM+E that combined the recent large vision model SAM [28] with our privilege expert. A detailed description of our settings for SAM+E and its segmentation results on DMC-GB can be found in Appendix 6.2.

3.1. Comparisons

Evaluation on DMC-GB. We conducted experiments on 7 tasks on DMC-GB following SOTA methods [2, 51] including Ball in cup, Cartpole swingup (denoted as Cartpole), Walker walk, Walker stand, Finger spin, Cheetah run and Finger turn easy (Denoted as Finger turn). We reported our results in terms of the returns over 5 random seeds in Table 1. We highlight the best methods for each task in bold and underline the second best. Overall, GEMO achieves a significant generalization performance improvement on video-hard settings, which is 17.4% higher than previous SOTA method SGQN. We also notice that with powerful vision capabilities, SAM+E also demonstrates notable generalization ability on some tasks in video-hard, such as Finger spin and Finger turn. However, in other tasks, such as Walker walk, the performance in video-hard experiences a decline of 56.6% compared to video-easy. This drop can be attributed to the Walker environment, where the moving robot might be misclassified as background pixels due to predefined point prompts, leading to the destruction of task-related pixels with strong augmentation. This underscores our intuition of applying data augmentation specifically to task-irrelevant pixels. In contrast to SAM+E, our method, equipped with a self-supervised control-aware augmentation mask, consistently achieves high performance in both video-hard and video-easy settings. The generalization performance on video hard during training can be found in Fig. 4. Clearly, GEMO achieves a higher generalization performance compared with other methods within the same update iterations of the visuomotor policy.

In addition, we evaluate GEMO on a more challenging distracting settings [47] with variations of the foreground color, background video, and the camera view in DMC-GB. As shown in Fig. 5, GEMO consistently outperforms other strong baselines in all tasks. Particularly, in Cheetah run, GEMO achieves an average returns of 301 under intensity of 0.1, which is 68.9% higher than SAM+E (with a score of

Table 1. **DMC-GB Generalization Performance.** We report the episode returns over 5 random seeds. Δ denotes GEMO’s improvement upon the second best results. *Italicized numbers* indicate reporting average results with one official pre-released model.

Setting	Easy Task	DrQ	DrQ-v2	RAD	SODA	SVEA	TLDA	VAI	SAM+E	SGQN	GEMO	Δ
Video easy	Ball in cup	380 \pm 188	401 \pm 67	363 \pm 158	939 \pm 10	928 \pm 43	892 \pm 68	909 \pm 44	925	889 \pm 87	973 \pm 9	+34
	Cartpole	459 \pm 81	267 \pm 26	473 \pm 54	742 \pm 73	772 \pm 46	671 \pm 57	729 \pm 19	<i>777</i>	770 \pm 56	861 \pm 6	+84
	Walker walk	747 \pm 21	196 \pm 52	608 \pm 92	771 \pm 66	839 \pm 29	873 \pm 34	871 \pm 42	<i>823</i>	<i>881</i> \pm 18	929 \pm 17	+48
	Walker stand	926 \pm 30	487 \pm 83	879 \pm 64	905 \pm 7	947 \pm 11	<i>973</i> \pm 6	948 \pm 12	<i>910</i>	950 \pm 10	978 \pm 12	+5
	Finger spin	599 \pm 62	491 \pm 35	516 \pm 113	783 \pm 51	737 \pm 93	744 \pm 18	932 \pm 2	<i>859</i>	947 \pm 16	<i>934</i> \pm 7	-13
	Cheetah run	154 \pm 22	79 \pm 8	104 \pm 5	190 \pm 48	275 \pm 3	336 \pm 57	322 \pm 35	<i>444</i>	257 \pm 49	463 \pm 13	+19
	Finger turn	230 \pm 29	358 \pm 68	172 \pm 63	150 \pm 50	197 \pm 98	208 \pm 35	445 \pm 36	<i>645</i>	547 \pm 267	721 \pm 13	+76
	Average	499	326	445	640	671	671	703	769	749	837	+36(4.3%)
Video hard	Ball in cup	100 \pm 40	83 \pm 20	98 \pm 40	381 \pm 163	492 \pm 100	257 \pm 57	<i>524</i>	725	<i>857</i> \pm 29	944 \pm 7	+87
	Cartpole	136 \pm 29	137 \pm 20	152 \pm 29	452 \pm 45	401 \pm 38	286 \pm 47	<i>378</i>	<i>337</i>	<i>537</i> \pm 33	623 \pm 22	+86
	Walker walk	121 \pm 52	87 \pm 5	80 \pm 10	312 \pm 32	521 \pm 68	271 \pm 55	<i>823</i>	<i>357</i>	719 \pm 33	883 \pm 9	+60
	Walker stand	252 \pm 57	234 \pm 50	229 \pm 45	736 \pm 132	840 \pm 16	602 \pm 51	<i>931</i>	<i>612</i>	842 \pm 36	941 \pm 11	+10
	Finger spin	38 \pm 13	31 \pm 8	39 \pm 20	309 \pm 49	361 \pm 25	241 \pm 29	<i>752</i>	<i>772</i>	767 \pm 11	798 \pm 5	+26
	Cheetah run	74 \pm 24	34 \pm 6	98 \pm 19	155 \pm 16	157 \pm 14	90 \pm 27	<i>303</i>	<i>264</i>	238 \pm 37	435 \pm 2	+132
	Finger turn	236 \pm 47	159 \pm 8	155 \pm 9	155 \pm 16	237 \pm 15	104 \pm 18	<i>362</i>	<i>563</i>	461 \pm 64	565 \pm 10	+2
	Average	137	109	122	357	429	264	581	519	<i>631</i>	741	+110(17.4%)

195) and two times higher than SGQN (with a score of 149). This is because with a control-aware augmentation mask, GEMO augments all the task-irrelevant regions including some parts of foreground to improve the generalization ability. Therefore, GEMO demonstrates a robust generalization ability against visual changes in the testing environments. We also notice that although SAM+E achieves high performance in some tasks on video-hard, it is difficult to generalize against the foreground and camera views with a pure visual segmentation.

Evaluation on RMDB. Table. 2 shows the training and evaluation results of GEMO and other five strong baselines on RMDB. Clearly, GEMO achieves high performance both on the training and testing environments. For example in PegBox training task, GEMO achieves a significant improvement (more than 5 times) in average returns compared to SGQN. Although SAM+E achieves high performance in training, its generalization ability is lower than GEMO. Specifically, it achieves 121.6 in testing environment in PegBox, which is 20% lower than that achieved by GEMO (which is 155.3). With the reliable guidance of privilege Expert and a clear visual mask, it is easier for SAM+E to learn a stable visuomotor control policy compared with other VRL methods. However, the visual mask is not enough to handle the visual variations of distracting objects, which is a usual case in many robotic tasks. In contrast, the control-aware mask in GEMO augments the task-irrelevant regions in the observation space and preserve the control-related information for decision making.

In Fig. 6, we visualize the masks obtained by GEMO and SAM for task Hammer and Push. In Hammer, SAM identifies the whole robot arm, the hammer, the toy box and two distracting balls as the foreground and other regions as background. In contrast, in GEMO, only parts of essential joint/link in robot arm, the hammer and the button in the

toy box are captured with the mask and other regions are augmented with *random conv*, as shown in the second and the fourth picture in the top row of Fig. 6. This is because with the sparsity constraint in Equ. 5, only the most control-related areas could be preserved for control prediction. And this control-aware mask in turn helps to improve the generalization ability against visual changes of all task-irrelevant regions. We need to point out GEMO could capture all control-related regions, no matter dynamic or static. For example, in the bottom row in Fig. 6, the robot is required to push the green box towards the red target, which remains static across one episode. However, GEMO could also capture it to minimize the control prediction loss. Thus, with the goal information carefully preserved, the training and testing performance of GEMO are both high compared with other SOTA methods. More visualization result on RMDB and our detailed settings of SAM could be found in the Appendix 8 and 6.2.

3.2. Ablation studies

We further study how each part in GEMO contributes to the generalization and stability performance. In this subsection, we first conduct ablation studies of the control-aware augmentation module and the privilege-guided distillation module on DMC-GB and DOGB. Then, we investigate how much improvement would the privilege Expert brought to other SOTA methods on RMDB tasks, which are challenging for those VRL methods. Last, we perform ablations on the loss function in Equ. 5 on DMC-GB.

Ablation on module structure. We conducted extensive ablation studies on GEMO for evaluating the effectiveness of the control-aware data augmentation module and the privilege-guided distillation module. As shown in Table 3, we designed four ablation methods for comparison. 1) **Q-only** is the vanilla DrQ-v2 algorithm with random shift

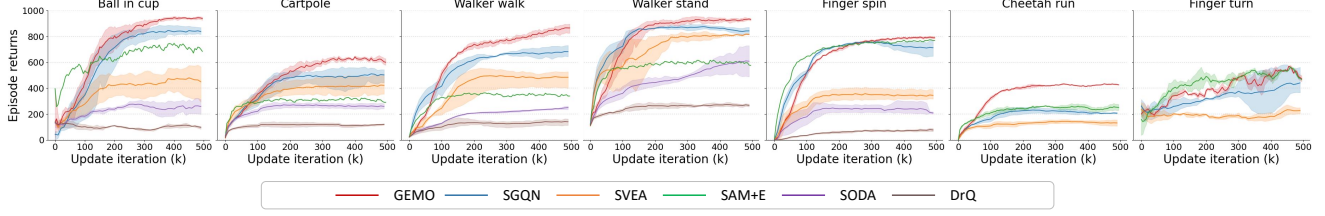


Figure 4. Generalization performance of all methods on DMC-GB video-hard.

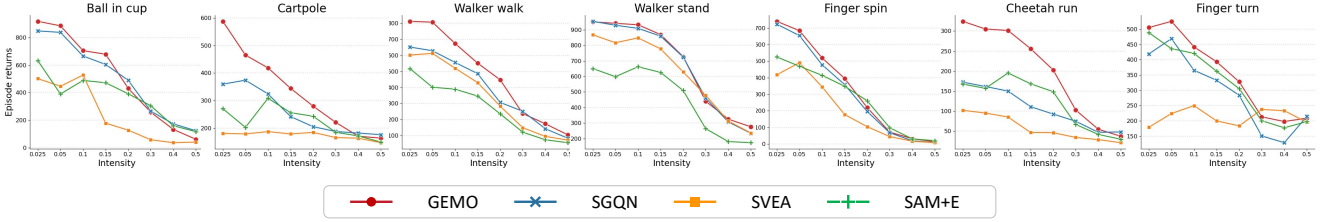


Figure 5. Generalization performance of all methods on DMC-GB distraction setting.

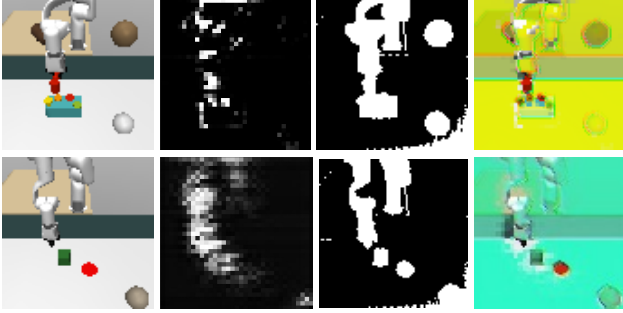


Figure 6. Illustration of the GEMO's mask and SAM's mask on Hammer and Push task. The first column: original observations. The second column: GEMO's mask. The third column: SAM's mask. Last column: control-aware augmented observations.

augmentation. 2) **Q+Aug** is the method applying *random overlay* augmentation to **Q-only**. 3) **Q+Mask** is the method adding control-aware attention mask to **Q+Aug** 4) **E+Aug** is the method using privilege-guided policy distillation and applying *random overlay* to the input image.

In Table 3, we observe that directly applying strong augmentation (method Q+Aug) to the DRL baseline (Q-only) may cause policy performance degradation even in the training environment. For example, for Walker walk task, Q+Aug only achieved a score of 435.6 in the training environment, which is 7% lower than Q-only. In contrast, as shown in the results of Q+Mask and E+Aug, both control-aware attention mask and privilege expert can take advantage of overlay augmentation to improve the training and generalization performance on DMC-GB. For example, in Walker walk, Q+Mask achieved an average increase of

Table 2. **Robot Manipulation Generalization Performance.** Test Average denotes the average returns across 5 different test environments.

Method	Hammer		Pegbox	
	Train	Test	Train	Test
SODA	-17.7 ± 3	-22.6 ± 6	-59.5 ± 22	-49.6 ± 3
SVEA	-10.6 ± 1	-16.8 ± 4	47.3 ± 65	-31.4 ± 27
TLDA	-10.5 ± 2	-20.7 ± 10	28.9 ± 31	-53.7 ± 18
VAI	-13.2 ± 2	-24.9 ± 7	30.3 ± 56	6.8 ± 65
SAM+E	-10.9 ± 1	-21.9 ± 10	169.9 ± 18	121.6 ± 69
SGQN	-11.0 ± 1	-18.7 ± 8	32.2 ± 23	22.5 ± 65
GEMO	-9.9 ± 1	-14.1 ± 4	172.5 ± 16	155.3 ± 8
	Push		Reach	
	Train	Test	Train	Test
SODA	-3.1 ± 4	-2.2 ± 1	10.5 ± 3	1.3 ± 8
SVEA	-4.9 ± 6	-5.7 ± 1	28.9 ± 3	-18.3 ± 6
TLDA	-0.9 ± 2	-0.2 ± 1	28.6 ± 5	15.9 ± 16
VAI	-0.9 ± 3	-2.2 ± 4	32.9 ± 1	26.2 ± 5
SAM+E	8.2 ± 2	0.0 ± 9	31.4 ± 0	-18.0 ± 15
SGQN	-6.4	-6.8 ± 4	30.1 ± 1	16.5 ± 21
GEMO	10.2	1.7 ± 3	33.2 ± 1	30.8 ± 3

44% in the training environment and an average increase of 239% in the video easy environment. E+Aug also gained average increases of 99% and 323% performance in the training and video-easy settings respectively.

It is worth noting that on most tasks under video-hard setting, the control-aware augmentation module contributes more to the generalization performance, while the privilege expert module could decrease the performance variance. For example, in Walker walk, Q+Mask achieved an increase of 453% compared to the vanilla Q-only, but with a large variance of 136. While for E+Aug, it increased the gen-

Table 3. **Ablation Study of our control-aware attention module (denoted as Att.) , privilege-guided distillation module (denoted as Exp.) and the random overlay augmentation (denoted as Aug.) on DMC-GB.** We designed four ablation methods for comparison. 1) Q-only, 2) Q+Aug, 3) Q+Mask and 4) E+Aug.

Setting		Q.	Aug.	Att.	Exp.	Ball in cup	Cartpole	Walker walk	Walker stand	Finger spin	Finger turn	Cheetah run	Average
Train	Q-only	✓				678.1 ± 387	835.0 ± 40	467.7 ± 120	872.7 ± 92.1	839.3 ± 51	454.2 ± 300	457.3 ± 293.8	657.8
	Q+Aug	✓	✓			766.8 ± 203(+13%)	270.4 ± 7(-68%)	435.6 ± 29(-7%)	473.2 ± 331(-46%)	658.4 ± 29(-22%)	246.8 ± 41(-45.7%)	227.6 ± 92(-50.2%)	439.8
	Q+Mask	✓		✓		801.1 ± 120(+18%)	845.5 ± 10(1%)	673.1 ± 185(+44%)	965.6 ± 7(+11%)	844.6 ± 22(0%)	679.2 ± 16(49%)	256.8 ± 12(-44%)	723.3
	E+Aug				✓	970.2 ± 2(+43%)	836.8 ± 10(+0%)	931.4 ± 34(+99%)	907.4 ± 25(+4%)	843.5 ± 23(+1%)	914.6 ± 55(+101.4%)	393.9 ± 11(-13.9%)	828.3
	GEMO	✓	✓	✓	✓	968.2 ± 8(+43%)	869.2 ± 5(+4%)	946.4 ± 5(+102%)	952.1 ± 15(+9%)	839.3 ± 8(+0%)	788.0 ± 47(+73%)	838.8 ± 15(+83%)	886.0
Video easy	Q-only	✓				401.3 ± 67	267.0 ± 26	195.7 ± 52	487.0 ± 83	490.6 ± 35	258.3 ± 68	78.5 ± 8	311.2
	Q+Aug	✓	✓			722.9 ± 169.3(+80%)	248.6 ± 22(-7%)	369.7 ± 12(+89%)	478.6 ± 329.3(-2%)	586.5 ± 20(+20%)	212.9 ± 85(-18%)	212.6 ± 62(+171%)	404.6
	Q+Mask	✓		✓		724.7 ± 107(+81%)	606.6 ± 43(+127%)	633.2 ± 174(+239%)	961.3 ± 2(+239%)	767.9 ± 19(+239%)	298.2 ± 18(16%)	296.4 ± 53(+278%)	612.1
	E+Aug				✓	954.8 ± 12(+138%)	670.9 ± 18(+151%)	828.1 ± 59(+323%)	886.2 ± 41(+82%)	741.0 ± 8(+51%)	577.8 ± 13(+124%)	325.4 ± 11(+314%)	712.0
	GEMO	✓	✓	✓	✓	973.4 ± 9(+143%)	861.0 ± 6(+222%)	929.3 ± 17(+375%)	978.5 ± 12(+101%)	934.1 ± 7(+70%)	721.3 ± 13(+179%)	463.2 ± 13(+490%)	834.7
Video hard	Q-only	✓				83.4 ± 20	136.9 ± 20	87.4 ± 5	233.5 ± 50	30.7 ± 8	158.9 ± 8	34.1 ± 6	109.3
	Q+Aug	✓	✓			313.1 ± 108(+271%)	186.6 ± 22(+36%)	233.3 ± 34(+167%)	382.9 ± 245(+64%)	276.6 ± 15(+801%)	225.2 ± 62(+42%)	113.0 ± 10(+231%)	247.2
	Q+Mask	✓		✓		459.5 ± 31(+451%)	468.2 ± 31(+242%)	483.4 ± 136(+453%)	910.4 ± 17(+290%)	724.9 ± 10(+2216%)	294.2 ± 31(+85%)	183.2 ± 11(+437%)	503.4
	E+Aug				✓	657.4 ± 28(+688%)	301.9 ± 8(+121%)	457.8 ± 44(+424%)	666.4 ± 13(+185%)	363.9 ± 9(+1085%)	394.5 ± 48(+149%)	104.9 ± 12(+208%)	420.7
	GEMO	✓	✓	✓	✓	944.0 ± 7(+1032%)	622.7 ± 22(+355%)	882.5 ± 75(+908%)	941.3 ± 11(+303%)	797.9 ± 5(+2499%)	565.3 ± 10(+256%)	435.1 ± 2(+1179%)	741.5

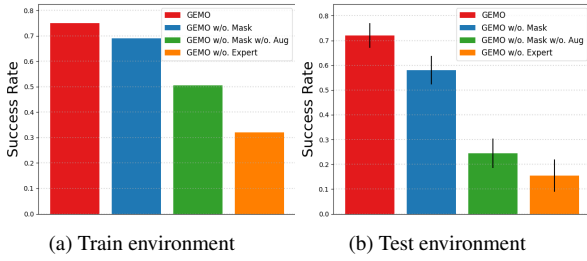


Figure 7. Ablation performance on DOGB

eralization performance with a much lower variance of 44. Therefore, by combining the control-aware attention mask and privilege-guided distillation module, GEMO achieves a significant improvement of 908% in Walker walk. This illustrates the joint influence of these two modules on enhancing the efficient generalization of visuomotor policies.

We conducted ablation studies to validate our method on a long-horizontal drawer opening task in DOGB. As shown in 7, facing with unseen backgrounds, GEMO consistently achieved an average success rate of 72% across large-scale testing environments. GEMO w/o. Expert showed a low performance both on the training and the testing environment with 32% and 16% success rate respectively. This demonstrates that the privilege Expert plays an important role for more challenging tasks. We can also observe that although GEMO w/o. Mask achieved a comparable result with GEMO on the training environment, it showed weak generalization ability without the attention mask. This demonstrates the contribution of our attention mask on the long-horizontal robot manipulation task.

Combine Expert module with other SOTA methods. Although we have combined the privilege Expert with a powerful large vision model, we still want to investigate if the Expert module could consistently improve the generalization performance on other SOTA methods. We combined Expert with SODA, SVEA, VAI and SGQN on RMDb considering of the challenging tasks for VRL methods on this

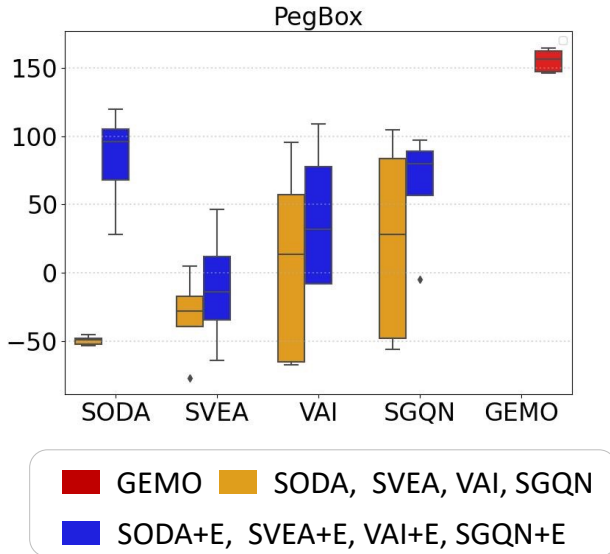


Figure 8. Improvements on strong baselines with privilege expert.

benchmark. We show the results on PegBox task in Fig. 8. Clearly, Expert module could also notably improve the generalization ability of other SOTAs. For example, SGQN+E achieved almost two times improvement from 22 to 63 after applying the Expert module, and experienced 44.6% variance reduction. This indicates that it is crucial to leverage the privilege information in the training environment when facing challenging robotic tasks. We also notice that after combining with Expert, GEMO still outperformed other methods. This contributes to the control-aware augmentation module in GEMO which applies the augmentation only on task-irrelevant regions and thus consistently improves the generalization performance during training.

Ablation on the loss function. In Equ. 5, we formulate a self-supervised reconstruction task along with three auxiliary losses to obtain a control-aware mask for augmentation.

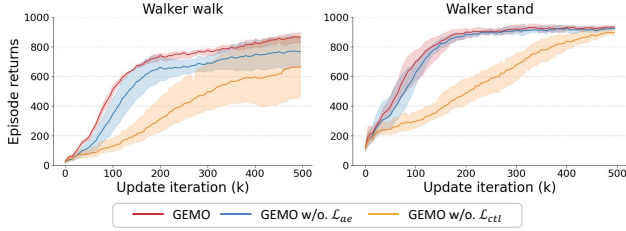


Figure 9. Ablation study of our control prediction loss and the auto-encoder loss in our augmentation module.

We conduct ablation studies of \mathcal{L}_{ctl} and l_{ae} on DMC-GB. In Fig. 9, we show the convergence results on video-hard settings of Walker walk and walk stand, while others could be found in the appendix. Obviously, the control prediction loss \mathcal{L}_{ctl} has a significant impact on the sample efficiency. For example in Walker stand, GEMO quickly converged to a generalizable visuomotor policy before 200 training iterations. While GEMO w/o. \mathcal{L}_{ctl} only converged to the same level at 500 update iterations and with a higher variance. This is because \mathcal{L}_{ctl} could efficiently help the model to find the most essential regions towards decision-making under the restriction of limited masked region. Otherwise, the mask model would focus on other control-irrelevant regions and influence the training efficiency. We also notice that the auto-encoder loss l_{ae} has different influence on different task. In walker walk, the training process without an auto-encoder loss could have large variance and result in an unstable generalization performance. While in walker stand, its influence on stability is much lower. More detailed experimental results on the ablation study of the loss function can be found in Appendix 7.1.

4. Related works

Generalization in Deep RL The poor generalization capability, a core challenge of Deep RL, has been extensively explored in a plethora of prior work [6, 7, 25, 36, 38, 40, 42, 50, 54, 59], but still has not been fully overcome. A common approach is to use regularization techniques like dropout [46], entropy regularization [17, 61] and batch normalization [22]. However, these methods are directly derived from supervised learning [6, 35] and thus can only provide limited improvements and even reduce sampling efficiency [6, 53]. Recently, more works propose to use auxiliary tasks or re-design training objects to learn invariant and robust representations, including information bottleneck (MIB) [10], bisimulation metrics [27, 60] and pre-trained backbone [57]. For more discussions about the generalization ability of DRL, please refer to the comprehensive survey [29]. Our method, GEMO, belongs to the branch of data augmentation technologies that is orthogonal compared to the above methods.

Data augmentation. Data augmentation [19, 20, 30–32, 41, 52] is a promising approach in acquiring generalization ability for visuomotor policies. Weak augmentations, including geometric transformations (e.g., random cropping) and photometric transformations (e.g., grayscale operator), only provide limited generalization improvement. While strong augmentations [2, 11, 56] can reduce generalization gaps by introducing highly diverse samples, but suffer from a diverged action distribution [56]. [34] uses a random convolutions layer to remove the visual bias in images and modify the texture information. VAI [51] proposes a three-step method to learn an invariant mask against visual variance for VRL. EXPAND [16] leverages a human saliency map to augment only the task-irrelevant regions in the images. SQGN [2] use a saliency-guided strong augmentation for Q-networks.

Policy distillation. There are a number of studies that use policy distillation [21] for different scenarios and complex tasks [3, 8, 33]. SECANT [11] leverages a combination of strong augmentation techniques in a two-stage distillation process. Our approach also distills knowledge to remain stability, but from an more reliable expert. Thus, GEMO is capable of handling more intricate tasks, especially in long-horizontal ones with uninformative reward. TLDA [56] augments task-irrelevant pixels in the images using the Lipschitz constants. LBC [3] trains a student policy with limited sensor input under the supervision of an expert with privileged information in the CARLA simulator [9] for autonomous driving tasks. [62] builds a task distillation framework to transfer navigation policies between different simulators. ITER [23] transfers repeated knowledge into a new initialized network to reduce the non-stationary effects. In this work, we use policy distillation in robot control tasks, which differs from the previous works.

5. Conclusion

In this paper, we propose an efficient **GE**neralizable **fr**amewor**K** for visu**Om**otor policies (GEMO) to identify the control-relevant information and achieve zero-shot generalization ability to unseen environments. GEMO consists of two jointly optimized modules: a control-aware augmentation module and a privilege-guided policy distillation module. In the former module, through formulating a self-supervised reconstruction task with three auxiliary losses, we learn the control-aware attention mask to distinguish the task-irrelevant pixels and then apply strong augmentations to reduce the generalization gaps. In the privilege-guided policy distillation module, we distill the knowledge from a pretrained privilege expert to the visuomotor policies. We conduct extensive comparison and ablation studies on three challenging benchmarks to evaluate GEMO. The experimental results well justify the effectiveness of GEMO.

References

- [1] Andrea Banino, Caswell Barry, Benigno Uria, Charles Blundell, Timothy Lillicrap, Piotr Mirowski, Alexander Pritzel, Martin J Chadwick, Thomas Degris, Joseph Modayil, et al. Vector-based navigation using grid-like representations in artificial agents. *Nature*, 557(7705):429–433, 2018. 1
- [2] David Bertoin, Adil Zouitine, Mehdi Zouitine, and Emmanuel Rachelson. Look where you look! saliency-guided q-networks for visual rl tasks. *Advances in neural information processing systems*, 2022. 1, 4, 8, 3
- [3] Dian Chen, Brady Zhou, Vladlen Koltun, and Philipp Krähenbühl. Learning by cheating. In *Conference on Robot Learning*, pages 66–75. PMLR, 2020. 2, 3, 8
- [4] Xi Chen, Ali Ghadirzadeh, Mårten Björkman, and Patric Jensfelt. Adversarial feature training for generalizable robotic visuomotor control. In *2020 IEEE International Conference on Robotics and Automation (ICRA)*, pages 1142–1148. IEEE, 2020. 1
- [5] Hyesong Choi, Hunsang Lee, Seongwon Jeong, and Dongbo Min. Environment agnostic representation for visual reinforcement learning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 263–273, 2023. 1
- [6] Karl Cobbe, Oleg Klimov, Chris Hesse, Taehoon Kim, and John Schulman. Quantifying generalization in reinforcement learning. In *International Conference on Machine Learning*, pages 1282–1289. PMLR, 2019. 8
- [7] Karl Cobbe, Chris Hesse, Jacob Hilton, and John Schulman. Leveraging procedural generation to benchmark reinforcement learning. In *International conference on machine learning*, pages 2048–2056. PMLR, 2020. 1, 8
- [8] Wojciech M Czarnecki, Razvan Pascanu, Simon Osindero, Siddhant Jayakumar, Grzegorz Swirszcz, and Max Jaderberg. Distilling policy distillation. In *The 22nd international conference on artificial intelligence and statistics*, pages 1331–1340. PMLR, 2019. 8
- [9] Alexey Dosovitskiy, German Ros, Felipe Codevilla, Antonio Lopez, and Vladlen Koltun. Carla: An open urban driving simulator. In *Conference on robot learning*, pages 1–16. PMLR, 2017. 8
- [10] Jiameng Fan and Wenchao Li. Dribo: Robust deep reinforcement learning via multi-view information bottleneck. In *International Conference on Machine Learning*, pages 6074–6102. PMLR, 2022. 8
- [11] Linxi Fan, Guanzhi Wang, De-An Huang, Zhiding Yu, Li Fei-Fei, Yuke Zhu, and Animashree Anandkumar. Secant: Self-expert cloning for zero-shot generalization of visual policies. In *International Conference on Machine Learning*, pages 3088–3099. PMLR, 2021. 1, 3, 8
- [12] Jesse Farebrother, Marlos C Machado, and Michael Bowling. Generalization and regularization in dqn. *arXiv preprint arXiv:1810.00123*, 2018. 1
- [13] Xiang Fu, Ge Yang, Pulkit Agrawal, and Tommi Jaakkola. Learning task informed abstractions. In *International Conference on Machine Learning*, pages 3480–3491. PMLR, 2021. 1
- [14] Alexandre Galashov, Josh S Merel, and Nicolas Heess. Data augmentation for efficient learning from parametric experts. *Advances in Neural Information Processing Systems*, 35: 31484–31496, 2022. 2
- [15] Shani Gamrian and Yoav Goldberg. Transfer learning for related reinforcement learning tasks via image-to-image translation. In *International conference on machine learning*, pages 2063–2072. PMLR, 2019. 1
- [16] Lin Guan, Mudit Verma, Suna Sihang Guo, Ruohan Zhang, and Subbarao Kambhampati. Widening the pipeline in human-guided reinforcement learning with explanation and context-aware data augmentation. *Advances in Neural Information Processing Systems*, 34:21885–21897, 2021. 8
- [17] Tuomas Haarnoja, Aurick Zhou, Pieter Abbeel, and Sergey Levine. Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor. In *International conference on machine learning*, pages 1861–1870. PMLR, 2018. 8
- [18] Siddhant Haldar, Vaibhav Mathur, Denis Yarats, and Lerrel Pinto. Watch and match: Supercharging imitation with regularized optimal transport. In *Conference on Robot Learning*, pages 32–43. PMLR, 2023. 3, 1
- [19] Nicklas Hansen and Xiaolong Wang. Generalization in reinforcement learning by soft data augmentation. In *2021 IEEE International Conference on Robotics and Automation (ICRA)*, pages 13611–13617. IEEE, 2021. 1, 2, 3, 4, 8
- [20] Nicklas Hansen, Hao Su, and Xiaolong Wang. Stabilizing deep q-learning with convnets and vision transformers under data augmentation. *Advances in neural information processing systems*, 34:3680–3693, 2021. 1, 4, 8
- [21] Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2015. 8
- [22] Maximilian Igl, Kamil Ciosek, Yingzhen Li, Sebastian Tschitschek, Cheng Zhang, Sam Devlin, and Katja Hofmann. Generalization in reinforcement learning with selective noise injection and information bottleneck. *Advances in neural information processing systems*, 32, 2019. 8
- [23] Maximilian Igl, Gregory Farquhar, Jelena Luketina, Wendelin Boehmer, and Shimon Whiteson. The impact of non-stationarity on generalisation in deep reinforcement learning. *arXiv preprint arXiv:2006.05826*, 2020. 8
- [24] Rishabh Jangir, Nicklas Hansen, Sambaran Ghosal, Mohit Jain, and Xiaolong Wang. Look closer: Bridging egocentric and third-person views with transformers for robotic manipulation. *IEEE Robotics and Automation Letters*, 7(2):3046–3053, 2022. 2, 4
- [25] Niels Justesen, Ruben Rodriguez Torrado, Philip Bontrager, Ahmed Khalifa, Julian Togelius, and Sebastian Risi. Illuminating generalization in deep reinforcement learning through procedural level generation. *NeurIPS Workshop on Deep Reinforcement Learning Workshop*, 2018. 8
- [26] Dmitry Kalashnikov, Alex Irpan, Peter Pastor, Julian Ibarz, Alexander Herzog, Eric Jang, Deirdre Quillen, Ethan Holly, Mrinal Kalakrishnan, Vincent Vanhoucke, et al. Scalable deep reinforcement learning for vision-based robotic manipulation. In *Conference on Robot Learning*, pages 651–673. PMLR, 2018. 1

- [27] Mete Kemertas and Tristan Aumentado-Armstrong. Towards robust bisimulation metric learning. *Advances in Neural Information Processing Systems*, 34:4764–4777, 2021. 8
- [28] Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C Berg, Wan-Yen Lo, et al. Segment anything. *arXiv preprint arXiv:2304.02643*, 2023. 4, 1
- [29] Robert Kirk, Amy Zhang, Edward Grefenstette, and Tim Rocktäschel. A survey of zero-shot generalisation in deep reinforcement learning. *Journal of Artificial Intelligence Research*, 76:201–264, 2023. 8
- [30] Ilya Kostrikov, Denis Yarats, and Rob Fergus. Image augmentation is all you need: Regularizing deep reinforcement learning from pixels. *International Conference on Learning Representations*, 2020. 8
- [31] Misha Laskin, Kimin Lee, Adam Stooke, Lerrel Pinto, Pieter Abbeel, and Aravind Srinivas. Reinforcement learning with augmented data. *Advances in neural information processing systems*, 33:19884–19895, 2020. 4
- [32] Michael Laskin, Aravind Srinivas, and Pieter Abbeel. Curl: Contrastive unsupervised representations for reinforcement learning. In *International Conference on Machine Learning*, pages 5639–5650. PMLR, 2020. 8
- [33] Joonho Lee, Jemin Hwangbo, Lorenz Wellhausen, Vladlen Koltun, and Marco Hutter. Learning quadrupedal locomotion over challenging terrain. *Science robotics*, 5(47):eabc5986, 2020. 2, 3, 8
- [34] Kimin Lee, Kibok Lee, Jinwoo Shin, and Honglak Lee. Network randomization: A simple technique for generalization in deep reinforcement learning. *International Conference on Learning Representations*, 2019. 1, 3, 8
- [35] Zhuang Liu, Xuanlin Li, Bingyi Kang, and Trevor Darrell. Regularization matters in policy optimization - an empirical study on continuous control. In *International Conference on Learning Representations*, 2021. 8
- [36] Marlos C Machado, Marc G Bellemare, Erik Talvitie, Joel Veness, Matthew Hausknecht, and Michael Bowling. Revisiting the arcade learning environment: Evaluation protocols and open problems for general agents. *Journal of Artificial Intelligence Research*, 61:523–562, 2018. 8
- [37] Viktor Makoviychuk, Lukasz Wawrzyniak, Yunrong Guo, Michelle Lu, Kier Storey, Miles Macklin, David Hoeller, Nikita Rudin, Arthur Allshire, Ankur Handa, et al. Isaac gym: High performance gpu based physics simulation for robot learning. In *Thirty-fifth Conference on Neural Information Processing Systems Datasets and Benchmarks Track*, 2021. 2
- [38] Charles Packer, Katelyn Gao, Jernej Kos, Philipp Krähenbühl, Vladlen Koltun, and Dawn Song. Assessing generalization in deep reinforcement learning. *arXiv preprint arXiv:1810.12282*, 2018. 8
- [39] Lerrel Pinto, Marcin Andrychowicz, Peter Welinder, Wojciech Zaremba, and Pieter Abbeel. Asymmetric actor critic for image-based robot learning. *arXiv preprint arXiv:1710.06542*, 2017. 2
- [40] Roberta Raileanu and Tim Rocktäschel. Ride: Rewarding impact-driven exploration for procedurally-generated environments. *International Conference on Learning Representations*, 2020. 8
- [41] Roberta Raileanu, Maxwell Goldstein, Denis Yarats, Ilya Kostrikov, and Rob Fergus. Automatic data augmentation for generalization in reinforcement learning. *Advances in Neural Information Processing Systems*, 34:5402–5415, 2021. 8
- [42] Aravind Rajeswaran, Kendall Lowrey, Emanuel V Todorov, and Sham M Kakade. Towards generalization and simplicity in continuous control. *Advances in Neural Information Processing Systems*, 30, 2017. 8
- [43] Sasha Salter, Dushyant Rao, Markus Wulfmeier, Raia Hadsell, and Ingmar Posner. Attention-privileged reinforcement learning. In *Conference on Robot Learning*, pages 394–408. PMLR, 2021. 2
- [44] Max Schwarzer, Ankesh Anand, Rishab Goel, R Devon Hjelm, Aaron Courville, and Philip Bachman. Data-efficient reinforcement learning with self-predictive representations. *International Conference on Learning Representations*, 2020. 1
- [45] Xingyou Song, Yiding Jiang, Stephen Tu, Yilun Du, and Behnam Neyshabur. Observational overfitting in reinforcement learning. *International Conference on Learning Representations*, 2019. 1
- [46] Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. Dropout: a simple way to prevent neural networks from overfitting. *The journal of machine learning research*, 15(1):1929–1958, 2014. 8
- [47] Austin Stone, Oscar Ramirez, Kurt Konolige, and Rico Jonschkowski. The distracting control suite—a challenging benchmark for reinforcement learning from pixels. *arXiv preprint arXiv:2101.02722*, 2021. 4
- [48] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017. 2
- [49] Che Wang, Xufang Luo, Keith Ross, and Dongsheng Li. Vrl3: A data-driven framework for visual deep reinforcement learning. In *Conference on Neural Information Processing Systems*, 2022. 3, 1
- [50] Huan Wang, Stephan Zheng, Caiming Xiong, and Richard Socher. On the generalization gap in reparameterizable reinforcement learning. In *International Conference on Machine Learning*, pages 6648–6658. PMLR, 2019. 8
- [51] Xudong Wang, Long Lian, and Stella X Yu. Unsupervised visual attention and invariance for reinforcement learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6677–6687, 2021. 1, 3, 4, 8
- [52] Denis Yarats, Rob Fergus, Alessandro Lazaric, and Lerrel Pinto. Mastering visual continuous control: Improved data-augmented reinforcement learning. In *Deep RL Workshop NeurIPS 2021*, 2021. 1, 3, 8
- [53] Denis Yarats, Ilya Kostrikov, and Rob Fergus. Image augmentation is all you need: Regularizing deep reinforcement learning from pixels. In *International Conference on Learning Representations*, 2021. 1, 4, 8
- [54] Denis Yarats, Amy Zhang, Ilya Kostrikov, Brandon Amos, Joelle Pineau, and Rob Fergus. Improving sample efficiency

- in model-free reinforcement learning from images. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 10674–10681, 2021. [8](#)
- [55] Wenhao Yu, Deepali Jain, Alejandro Escontrela, Atil Iscen, Peng Xu, Erwin Coumans, Sehoon Ha, Jie Tan, and Tingnan Zhang. Visual-locomotion: Learning to walk on complex terrains with vision. In *5th Annual Conference on Robot Learning*, 2021. [1](#)
- [56] Zhecheng Yuan, Guozheng Ma, Yao Mu, Bo Xia, Bo Yuan, Xueqian Wang, Ping Luo, and Huazhe Xu. Don’t touch what matters: Task-aware lipschitz data augmentation for visual reinforcement learning. *arXiv preprint arXiv:2202.09982*, 2022. [4](#), [8](#), [3](#)
- [57] Zhecheng Yuan, Zhengrong Xue, Bo Yuan, Xueqian Wang, Yi Wu, Yang Gao, and Huazhe Xu. Pre-trained image encoder for generalizable visual reinforcement learning. In *Advances in Neural Information Processing Systems*, 2022. [8](#)
- [58] Zhecheng Yuan, Zhengrong Xue, Bo Yuan, Xueqian Wang, Yi Wu, Yang Gao, and Huazhe Xu. Pre-trained image encoder for generalizable visual reinforcement learning. In *Advances in Neural Information Processing Systems*, 2022. [3](#), [1](#)
- [59] Amy Zhang, Nicolas Ballas, and Joelle Pineau. A dissection of overfitting and generalization in continuous reinforcement learning. *arXiv preprint arXiv:1806.07937*, 2018. [8](#)
- [60] Amy Zhang, Rowan McAllister, Roberto Calandra, Yarin Gal, and Sergey Levine. Learning invariant representations for reinforcement learning without reconstruction. *arXiv preprint arXiv:2006.10742*, 2020. [8](#)
- [61] Chiyuan Zhang, Oriol Vinyals, Remi Munos, and Samy Bengio. A study on overfitting in deep reinforcement learning. *arXiv preprint arXiv:1804.06893*, 2018. [8](#)
- [62] Brady Zhou, Nimit Kalra, and Philipp Krähenbühl. Domain adaptation through task distillation. In *Computer Vision—ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XXVI 16*, pages 664–680. Springer, 2020. [8](#)

An Efficient Generalizable Framework for Visuomotor Policies via Control-aware Augmentation and Privilege-guided Distillation

Supplementary Material

6. Implementation Details

In this section, we first provide the implementation details for GEMO, and then we provide the implementation details for a strong baseline SAM+E we utilized in our experiments.

6.1. Implementations of GEMO

Algorithm details. We present the pseudocode of GEMO in Algorithm 1. We first train the privileged expert given only the state information (i.e., the state s) using DrQv2 [52] (from line 5 to line 12). Note that any other standard RL algorithm can be applied here to replace DrQv2. In the second part of GEMO (from line 13 to line 24), we train the control-aware data augmentation module and privilege-guided policy distillation module **simultaneously**. For the former module, we build our implementation on top of the convolutional block attention module (CBAM) which consists of a channel attention block and a spatial attention block. The channel attention block comprises an averaging operation module (an average-pooling function followed by two MLP layers) and a maxing operation module (a max-pooling function followed by two MLP layers). A sigmoid activation function is applied on the end to produce a channel mask. The spatial attention block comprises a channel pool function (to extract the maximum and average features across channels), convolutional layers (with a kernel size of 7, stride of 1, and padding of 3), and a batchnorm layer sequentially. More detailed network structures and parameter settings can be found in our open-source code.

Hyperparameter settings. In practice, we build our implementation on top of DrQv2 [52] due to its broad adoption in continuous control tasks [18, 49, 58] as well as a clean and efficient codebase. Therefore, GEMO follows the hyperparameter settings from DrQ-v2, as listed in Tab. 4. Specifically, GEMO introduces only one new hyperparameter λ to control the sparsity of the attention mask. After conducting ablation studies on the sparsity loss in Supplementary 7, we set λ to 0.001 for DMC-GB/DOGB and 0.01 for RMDB. For data augmentation methods, we use random overlay with Places365 for DMC-GB/DOGB and random conv for RMDB.

6.2. Implementations of SAM+E

SAM [28] is one of the most popular state-of-the-art (SOTA) large vision models that is capable of getting the segmentation results automatically with proper point prompts over various domains. Therefore, we use SAM

Algorithm 1 GEMO

```

1:  $\pi_e, Q_1, Q_2$ : randomly initialized policy network and
   two value network for privileged expert
2:  $f_e(\cdot), f_d(\cdot), f_a(\cdot), f_{ctl}(\cdot, \cdot), :$  randomly initialized en-
   coder, decoder, attention network and control predic-
   tion network in control-aware augmentation module
3:  $\pi$ : random initialized student policy network
4: replay buffer  $\mathcal{D}$ , mini-batch size  $b$ 
5: for Each interaction timestep  $i$  do
6:   Rollout  $\pi_e$  for one timestep and add  $(s_i, a_i, r_i)$  to
     dataset  $\mathcal{D} \leftarrow \mathcal{D} \cup (s_i, a_i, r_i)$ 
7:   if  $|\mathcal{D}| \geq b$  then
8:     Sample experiences  $(s, a, r) \sim \mathcal{D}$ 
9:     Update  $\pi_e$ 
10:    Update  $Q_1, Q_2$ 
11:   end if
12: end for
13: Empty  $\mathcal{D} = \emptyset$ 
14: for Each interaction timestep  $t$  do
15:   Choose  $\pi_{roll}$  from  $\pi_e$  or  $\pi$ 
16:   Rollout  $\pi_{roll}$  for one timestep and add  $(o_t, s_t)$  to
     dataset  $\mathcal{D} \leftarrow \mathcal{D} \cup (o_t, s_t)$ 
17:   if  $|\mathcal{D}| > b$  then
18:     Sample experiences  $(o, s, o') \sim \mathcal{D}$ 
19:     Update control-aware data augmentation mod-
       ular according to Equation 5
20:     Obtain control-aware augmented observation
        $o_{aug}$  according to Equation 6
21:     Update  $\pi$  according to Equation 7
22:   end if
23: end for

```

to construct a strong baseline SAM+E. Considering the mask inference time, we use the ViT-B SAM model checkpoint. To get proper background prompts, we first upload an example image on Segment-Anything website and randomly sample points until we get a clear segmentation result. Then, we use an Image Position Coordinate Tool to get these prompts' positions. After configuring the point prompts, we check the segmentation results again with Weights&Biases.

The prompts we used in DMC-GB are (2, 80), (2, 57), (3, 7). The corresponding segmentation results we got with SAM are shown in Fig. 11. The prompts we used in RMDB are (2, 5), (2, 24), (2, 37), (2, 67). The corresponding segmentation results are shown in Fig. 12.

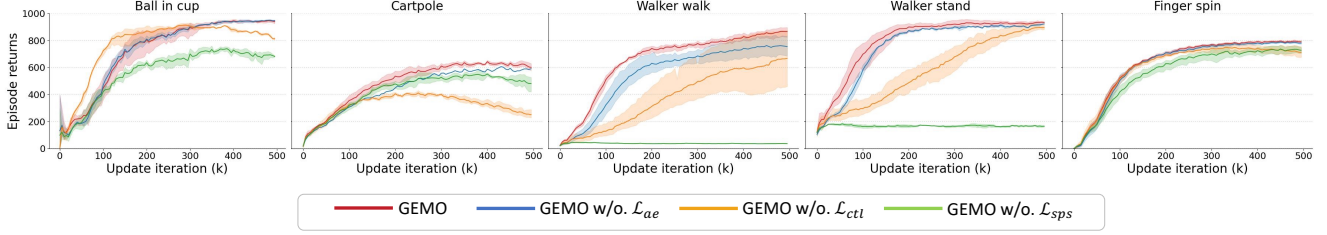


Figure 10. Ablation study on augmentation loss under DMC-GB video-hard.

Table 4. GEMO Hyperparameters.

	Hyperparameter	Value
Teacher	Learning rate for all net	1e-4
	Target update rate	0.01
	Optimizer	Adam
	Batch size	256
	Discount factor	0.99
	n-step	1 for Walker walk & Walker stand, 3 for others
	Update interval	2
Student	Replay buffer size	500k
	Observation	84 × 84 for DMC-GB/RMDB, 128 × 128 for DOGB
	Learning rate for all net	1e-4
	Optimizer	Adam
	Batch size	256
	Frame stack	3 for DMC-GB, 1 for RMDB/DOGB
	Update interval	2
	Replay buffer size	500k for DMC-GB/RMDB, 100k for DOGB
	λ	0.001 for DMC-GB, 0.01 for RMDB/DOGB
	β	0.5
	α in <i>random overlay</i>	linear schedule from 0.4 to 0.9

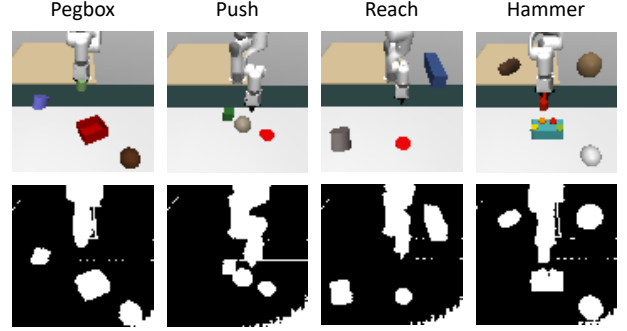


Figure 12. Segmentation results in RMDB with SAM.

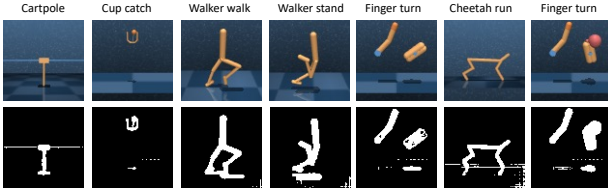


Figure 11. Segmentation results in DMC-GB with SAM.

7. Additional results on DMC-GB

In this section, we first provide more ablation results on DMC-GB. Then we provide more visualization results with GEMO in DMC-GB.

7.1. Ablation study on the loss function

The original manuscript presented ablation results on the loss function for the Walker-walk and Walker-stand tasks. In this section, we extend the ablation analysis to five tasks in DMC-GB, focusing on the loss terms \mathcal{L}_{rec} , \mathcal{L}_{ctl} , and \mathcal{L}_{sps} in Equ. 5.

Impact of \mathcal{L}_{ctl} . Apparently, the control prediction loss \mathcal{L}_{ctl} (represented by the orange line in Fig. 10) is crucial for the stability performance of GEMO. Absence of \mathcal{L}_{ctl} may result in significant variance in policy generalization performance (as observed in Walker walk and Walker stand tasks) or lead to a performance decline during training (as

shown in Ball in cup, Cartpole, and Finger spin tasks). This is due to the sparsity constraint applied in our method. This constraint may cause the augmentation module to distribute low values evenly across the entire dynamic foreground in GEMO w/o. \mathcal{L}_{ctl} . Consequently, with a linear growing schedule in *random overlay*, some control-related parts (e.g., the tip of the cartpole) may be disrupted by the augmentation process, thus influencing the overall stability and performance of the model.

Impact of \mathcal{L}_{sps} . The sparsity loss \mathcal{L}_{sps} (represented by the green line in Fig. 10) is also crucial for generalization performance. In the absence of \mathcal{L}_{sps} , the model’s generalization ability may converge to a lower level or even experience policy degradation. This occurs because, in the absence of the sparsity constraint, the generated mask may focus on a control-irrelevant background or even the entire observation space (in the Walker environment). Without fully utilizing the augmentation technique, this, in turn, leads to an apparent performance decline in video-hard settings.

Impact of \mathcal{L}_{ae} . The effect of \mathcal{L}_{ae} (represented by the blue line in Fig. 10) is different across five tasks. Overall, by adding the auto-encoder loss, we could learn a stable and generalizable policy in various tasks.

7.2. Visualization of GEMO on DMC-GB

In Fig. 13, we visualize the original observation o , the vanilla augmented observation $aug(o)$ with *random overlay* ($\alpha=0.8$), the control-aware mask m and the control-

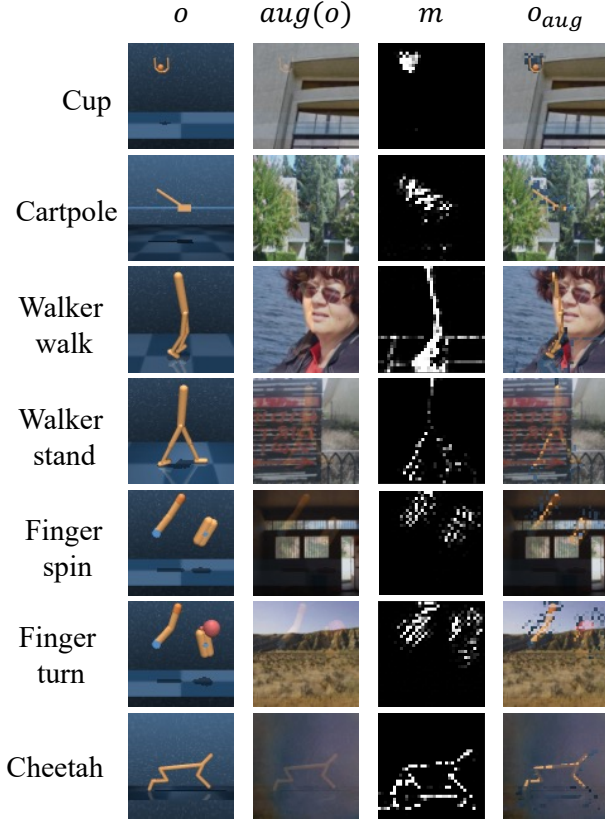


Figure 13. Visualization results of GEMO in DMC-GB.

aware augmented observation o_{aug} . We clearly observed that, with our control-aware mask, GEMO augments only the control-irrelevant regions while preserving the control-related parts. For example, when the walker is standing (in Walker stand), we preserve the some region of the "leg" and "head" of the robot. While the walker is walking (in Walker walk), we mainly concentrate on the "leg" and the "upper-body". With this control-aware augmentation module, we preserve the most control-related features for decision-making while augmenting other areas to enhance generalization.

8. Additional results on RMDB

8.1. Detail description of RMDB

Compared with the original Robot Manipulation benchmark, we made the following modifications in RMDB:

- For each task, we introduced 1-3 distractors on the front desk and/or background floor. The color of these distractors varies in each testing environment but remains consistent during training. We selected this configuration to assess the policy's generalization ability when facing visual appearance changes of these distractors.

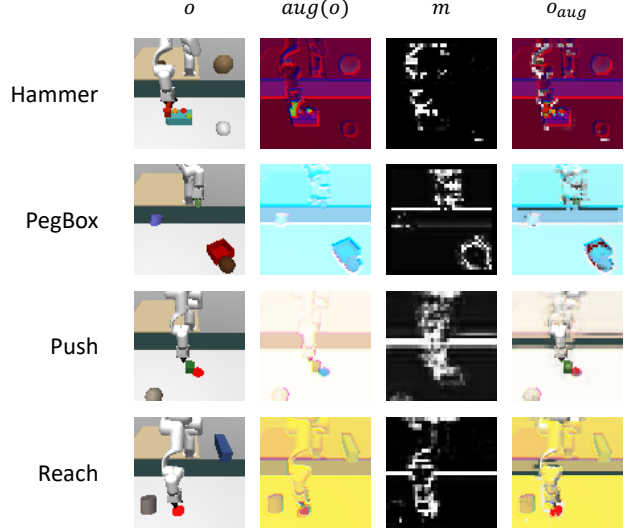


Figure 14. Visualization results of GEMO in RMDB.

- We revised the reward function in the Push task. The original function only considered control magnitude and the distance between the object and its target. We introduced an additional reward term to incentivize the robot arm to approach the object, facilitating learning.

Despite the aforementioned modifications, we follow the original Robot Manipulation benchmark to validate the policy across five testing environments, each with different visual changes to the front desk, background wall, and operating objects. The training environment examples can be found in Fig. 12. The testing environment examples can be found in Fig. 3.

8.2. Visualization of GEMO on RMDB

In Fig. 14, we demonstrate the original observation o , the vanilla augmented observation $aug(o)$ with *random conv*, the control-aware mask m and the control-aware augmented observation o_{aug} in RMDB. More detailed analysis and the comparison with the mask obtained by SAM can be found in Section. 3.1 in the manuscript.

9. Efficiency comparison

Mask generating time efficiency. We evaluated the efficiency of GEMO compared to four other methods in terms of the time required for mask generation in augmentation or consistency learning. Specifically, VAI [51] employs a reconstruction structure similar to GEMO but generates the mask through a decoder network with a predefined threshold. SGQN [2] produces sharp saliency maps by computing policy gradients and uses them in a consistency loss. TLDA [56] generates the mask by computing the Lipschitz constant via policy changes before and after image

Table 5. **Efficiency comparison of GEMO vs. other methods regarding mask generation.**

Method	GEMO	VAI	SGQN	TLDA	SAM
time (ms) per image ↓	1.028	4.989	8.692	10.668	329.941

Table 6. **Training Efficiency comparison of GEMO vs. other SOTAs.**

Method	Expert	GEMO	SVEA	SODA	SGQN	SVEA+E	TLDA
training time (h) ↓	0.9	8.3	9.0	8.0	16.3	22.8	61.3

perturbation. SAM [28] generates masks with given point prompts. While all these methods compute masks only during training, the inference time of the mask can impact training efficiency due to the large observation space. We computed the average mask inference time (ms) for each image over 15,000 iterations. All methods were evaluated on a Ubuntu 18.04 server with an 8GB NVIDIA GTX 1180 graphics card and a 12-core AuthenticAMD CPU.

As indicated in Table 5, GEMO demonstrates time efficiency in mask generation, thanks to a lightweight encoder and the upper-sample operation.

Training efficiency. We evaluate the training efficiency of GEMO compared to other SOTA methods, as detailed in Table 6. The average time efficiency across seven tasks in DMC-GB is provided. **Expert** denotes the privileged expert utilized in GEMO. Notably, GEMO exhibits high training efficiency, characterized by both low training time and high generalization performance. For example, while SGQN was the previous SOTA method in DMC-GB, it requires twice the training time of GEMO. GEMO requires a comparable training time to SVEA and SODA, yet achieves a remarkable average reward of 741 in DMC-GB video-hard settings, which is 72.4% and 107.6% higher than that achieved by SVEA and SODA, as shown in Table. 1.