Improving Generalization in Visual Reinforcement Learning via Conflict-aware Gradient Agreement Augmentation

Siao Liu   Zhaoyu Chen   Yang Liu   Yuzheng Wang   Dingkang Yang   Zhile Zhao
Ziqing Zhou   Xie Yi   Wei Li*   Wenqiang Zhang   Zhongxue Gan*

Academy for Engineering & Technology, Fudan University

{saliu20, zhaoyuchen20, yang_liu20, ganzhongxue}@fudan.edu.cn

Abstract

Learning a policy with great generalization to unseen environments remains challenging but critical in visual reinforcement learning. Despite the success of augmentation combination in the supervised learning generalization, naively applying it to visual RL algorithms may damage the training efficiency, suffering from serve performance degradation. In this paper, we first conduct qualitative analysis and illuminate the main causes: (i) high-variance gradient magnitudes and (ii) gradient conflicts existed in various augmentation methods. To alleviate these issues, we propose a general policy gradient optimization framework, named Conflict-aware Gradient Agreement Augmentation (CG2A), and better integrate augmentation combination into visual RL algorithms to address the generalization bias. In particular, CG2A develops a Gradient Agreement Solver to adaptively balance the varying gradient magnitudes, and introduces a Soft Gradient Surgery strategy to alleviate the gradient conflicts. Extensive experiments demonstrate that CG2A significantly improves the generalization performance and sample efficiency of visual RL algorithms.

1. Introduction

With the development of deep learning in various tasks [28, 26, 25, 27, 7, 6, 38, 40, 39, 24], visual Reinforcement Learning (RL) has achieved impressive success in various fields such as robotic control [11], autonomous driving [17], and game-playing [35]. Previous works usually formulate it as a Partially Observable Markov Decision Process (POMDP) [33], and the agent receives high-dimensional image observations as inputs. As depicted in [15, 14], visual RL generalization refers to the ability of a pretrained RL agent to perform well in unseen environments. Due to the dynamic nature of the real world, even minor perturbations in the environment can result in significant semantic shifts in the visual observations, which makes visual RL generalization challenging.

To improve generalization performance, data augmentation [29] is a widely adopted technique in reinforcement learning. Numerous studies [22, 13] utilize data augmentation methods to generate synthetic data and diversify the training environments, yielding considerable performance improvements. However, recent methods [14, 3, 44] mostly select a single augmentation technique to improve the generalization capability, resulting in a poor performance in the environments with observations varying far from the augmented images. For instance, ColorJitter [23] is the preferred choice for addressing color variations, but agents trained with such augmentation still hard to cope with intricate texture patterns. In other words, the generalization ability heavily relies on the selection of specific data augmentation technique, which is so-called generalization bias.

Compared to single data augmentation, Augmentation Combination (AC) [16] integrates multiple data augmentation methods to enhance the diversity of augmentations and alleviate the generalization bias, which is a more promising pre-processing solution. Unfortunately, there is a dilemma in incorporating AC into visual RL. Although data augmentation combination can effectively improve generalization capability in the supervised visual tasks, RL algorithms are quite sensitive to excessive variations, resulting in performance degradation and training sample inefficiency. Therefore, it is necessary to rethink why visual RL algorithms cannot benefit from AC as much as supervised learning.

From the perspective of gradient optimization, we conduct numerous qualitative analysis to illustrate the causes of performance degradation and training collapse that occur when employing augmentation combinations during training. There are two primary reasons for this phenomenon: **(i)**

---

*Corresponding author

the utilization of diverse data augmentations leads to high gradient magnitude variations, resulting in biased generalization; (ii) the gradient conflicts[1] [42] existed cross multiple augmentation methods hinder the policy optimization. To balance the gradients with high-variance magnitudes, one effective approach is to customize the weights of the loss terms with manually defined hyper-parameters [14]. However, hyper-parameter tuning relies heavily on expert knowledge, which can be inflexible and computationally expensive when dealing with multiple data augmentations. Besides, [30] indicates that the widely employed average-based gradient update strategies tend to converge towards the speed-greedy direction, and are ill-posed to effectively handle complex gradient conflicts, leading to local optima and a decrease in sampling efficiency.

To address these issues, we propose a general policy gradient optimization framework, named Conflict-aware Gradient Agreement Augmentation (CG2A), to integrate augmentation combination into the RL framework and improve its generalization performance. Specifically, the CG2A contains two key components: an adaptive weight assigner called Gradient Agreement Solver (GAS) and a conflict-aware gradient update strategy Soft Gradient Surgery (SGS). To effectively harmonize high-variance magnitudes gradients, we formulate the hyper-parameter tune as a second-order multi-objective optimization problem and use the GAS to obtain a proximal approximate solution with minimal computational cost. Moreover, according to [30], although gradient conflicts slow down convergence speed, these conflicting gradient components may contain more semantic-irrelevant information that can improve invariant learning consistency. Motivated by this hypothesis, we propose SGS to improve the gradient update process, which preserves a small amount of conflicting gradient components to strike a balance between convergence speed and generalization performance. To validate the effectiveness of CG2A, we conduct extensive experiments on DMControl Generalization Benchmark (DMC-GB) and some robotic manipulation tasks. In summary, our contribution encompasses three main manifolds:

- We point out the generalization bias induced by single data augmentation and illustrate the primary causes for performance degradation when naively applying augmentation combination in RL algorithms.

- We propose a general policy gradient optimization framework named Conflict-aware Gradient Agreement Augmentation (CG2A), to efficiently integrate data augmentation combinations into the RL algorithms and significantly improve the generalization performance in various environments.

- We devise a Gradient Agreement Solver (GAS) to harmonize multiple gradients with high-variance magnitudes, and propose a Soft Gradient Surgery (SGS) strategy to alleviate the gradient conflicts existed in various data augmentations.

- Compared to previous state-of-the-art methods, CG2A achieves competitive generalization performance and significantly improves sample efficiency.

2. Related Work

2.1. Data Augmentation in Visual RL

Benefiting from the development in the field of computer vision [45, 4, 5, 37, 8], data augmentation is widely used in the visual RL [22, 14, 15]. As noted by Kirk *et al.* [20], DA force the agents to learning an invariance knowledge through regularising models to have same output or inherent representation for different augmented images. Kostrikov *et al.* [21] adopt simple pixel-level transformations to perturb image observations and regularize the value function and policy, which significantly boost the sample efficiency. Inspired by MixUp [45], Wang *et al.* [36] propose to train agents with a mixture of observations and impose linearity constraints to improve the generalization capability. Meanwhile, Raileanu *et al.* [31] propose a principle to automatically select an effective augmentation from a set of data augmentations for RL tasks. More recently, task-aware data augmentation with Lipschitz constant is devoloped [43], which maintain the sample efficiency and alleviate instability caused by the aggressive data augmentations. Unlike prior work, our method expect to explore the utilization of augmentation combinations in visual RL to improve generalization capability in various unseen environments instead of a single manually or automatically data augmentation.

2.2. Generalization in Visual RL

Numerous studies [21, 15, 14, 44] attempt to enhance the generalization capability of agents through various approaches, such as data augmentation [22, 43, 21], domain randomization [32], and self-learning based methods [15]. Hansen *et al.* [15] build a BYOL [10]-like architecture and use an auxiliary loss to foster the representations to be invariant with the irrelevant perturbations. Hansen *et al.* [14] introduce a regularization term for the Q-function which reduces variance implicitly by linear combining the estimated Q-value between unaugmented and augmented data. Bertoin *et al.* [3] integrate saliency maps into the RL [12] architecture, enabling the agent to guide its focus towards the most salient aspects of the observation images during the decision-making process. Yuan *et al.* [44] utilize a pre-trained model and extract generalizable representations from the early layers of the encoder for enhancing the generalization performance. Unlike previous work, we utilize

---

[1]The gradient conflicts mean the gradient directions point away from each other, *e.g.*, appears a negative cosine similarity.

data augmentation combination to eliminate the generalization bias and develop an effective optimize framework to avoid the performance collapse during the training stage.
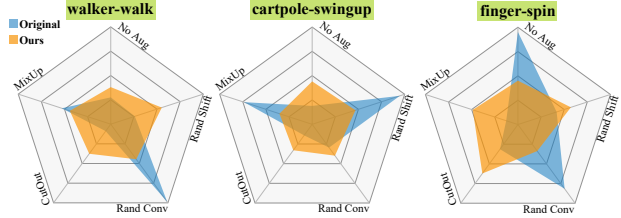
3. Preliminaries

Reinforcement Learning. Considering that the agent cannot directly observe the underlying state of environment from the given images [19], visual RL [2] formulates the interaction between the agent and its environment as a discrete-time Partially Observable Markov Decision Process (POMDP). Formally, a POMDP can be defined as a 6-tuple $\langle \mathcal{O}, \mathcal{S}, \mathcal{A}, \mathcal{P}, r, \gamma \rangle$, where $\mathcal{O}$ is the high-dimensional observation space, $\mathcal{S}$ is the state space, $\mathcal{A}$ is the action space, $\mathcal{P}(s_{t+1}|s_t, a_t)$ is the conditional transition function between states, $r : \mathcal{S} \times \mathcal{A} \rightarrow \mathbb{R}$ is the reward function, and the $\gamma \in [0, 1)$ is the discount factor [33].
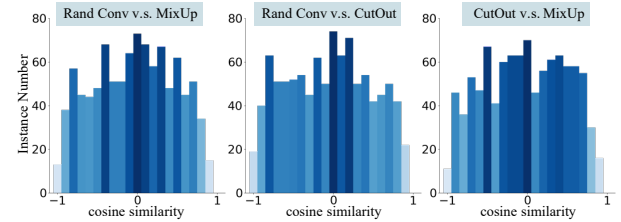
Generalization Definition. Given a set of POMDPs $\mathbb{M} = \{\mathcal{M}_1, \mathcal{M}_2, \ldots, \mathcal{M}_n\}$, each POMDP $\mathcal{M}_i$ has its own observation space $\mathcal{O}_i$, but shares a common underlying state space $\mathcal{S}$ and dynamic conditional transition function $\mathcal{P}$. Our objective is to learn a general policy that can generalize to unseen environments in a zero-shot manner. Specifically, we have access to only one specific POMDP $\mathcal{M}_i \in \mathbb{M}$ and utilize it to learn a general policy π^*. The policy π^* is expected to alleviate the dependency on the individual observation space $\mathcal{O}_i$ and explore the inherent state structure to perform well over the whole set of $\mathbb{M}$.

Optimization Pitfall with Augmentation Combination. Here, we elucidate the main reasons for limited training performance when data augmentation combinations are naively applied in policy gradient optimization. Specifically, we choose a random initialized SAC [12] agent as a base model and collect multiple data augmentation technologies, including random shift [21], random convolution [22], CutOut [8], and MixUp [45], to form a data combination. Then, a SVEA [14] architecture agent is adopted to integrate the data combination and calculate corresponding critic loss items associated with each data augmentation and obtains the corresponding gradient via backpropagation. Based on such framework, we conduct various toy experiments on DMC [34] suite and analyse the composition of the gradient from the perspective of the magnitude and direction. There are two primary reasons. **(i)** *High-variance gradient magnitudes*: we collect the normalized gradients associated with various augmented data over 5000 times for a same agent and compute the arithmetic mean as the gradient magnitude representation for such augmentation. As shown in Figure 1(a), the empirical results show that all agents exhibit high variance in gradient magnitude over the three tasks and that certain gradients prevail in magnitude, implying that the corresponding data augmentations dominate the policy optimization and lead to significant generalization bias. **(ii)** *Gradient conflicts existed in*

(a) (a) *Visualization of normalized gradients' magnitude associated with various augmentations in AC.* For each task, the optimization process can be dominated by specific gradients. Our method effectively suppresses these aggressive gradients.

(b) (b) *Statistics of cosine similarity between paired gradients for various augmented data.* There are a lot of paired gradients with a negative direction, which misleads gradient optimization.

Figure 1: The qualitative analysis of gradient magnitudes and gradient directions for the gradient vectors associated with data equipped with various augmentations.

multiple DAs: we sample 1000 image observations from an unseen environments and record the mean of gradient cosine similarity over 5 times cross paired augmented data. In our context, we define two gradients as conflicting if they have negative cosine similarity, indicating that they are far apart from each other. According to [42], if two gradients conflict with each other, the agent would struggle to find a balance between various optimization. Figure 1(b) indicates that gradient conflicts are a prevalent issue among different data augmentations, leading to slower convergence and performance degradation.

4. Method

In this section, we propose a general policy gradient optimization framework, named Conflict-aware Gradient Agreement Augmentation, to address these pitfalls from the perspective of gradient magnitude and direction.

4.1. Overview

Soft-critic-actor [12] (SAC) is implemented as the basic visual reinforcement learning algorithm. Briefly, SAC simultaneously learns a stochastic policy and a Q-function to maximize the discounted rewards, while incorporating entropy regularization to encourage the policy to explore diverse actions. Sampling a mini-batch of transitions $\zeta = \{o_t, a_t, r(s_t, a_t), o_{t+1}\}$ from the replay buffer $\mathcal{B}$, the critic
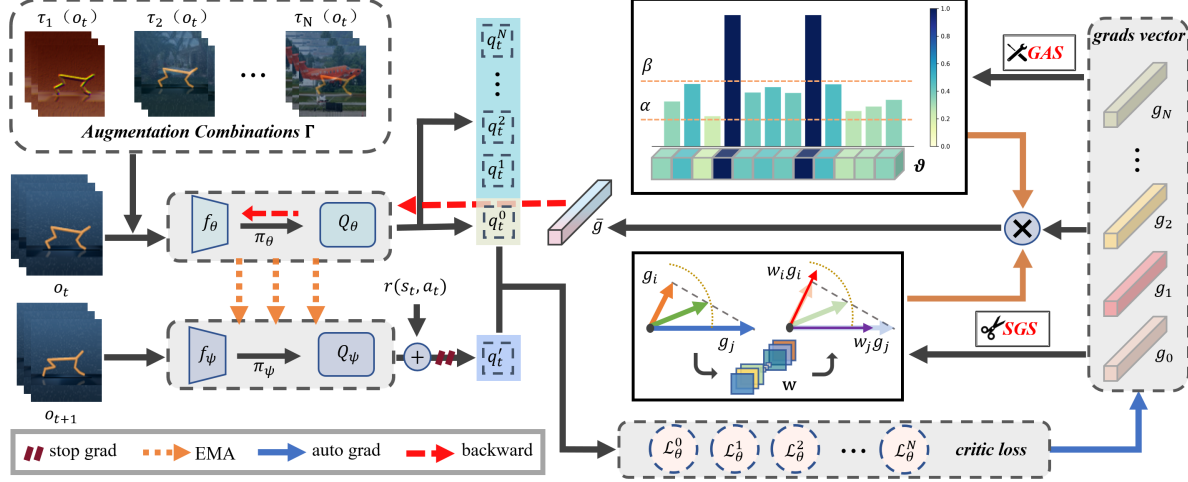
Figure 2: **Overview.** We input the original observation and augmented data and obtain the estimated Q-values q_t^* to calculate the loss items and the corresponding gradients g_i. Given the gradient vectors, CG2A utilizes SGS and GAS to calculate the weight coefficients $\mathbf{w}$ and gradient masks ϑ respectively and update the policy parameters with generated $\hat{g}$.

loss function can be expressed as follows:

$$\mathcal{L}_\theta = \sum_{\zeta \in \mathcal{B}} ||Q_\theta^*(o_t, a_t) - \hat{q}_t||^2, \qquad (1)$$

with the Temporal Difference (TD) target $\hat{q}_t$:

$$\hat{q}_t = r(o_t, a_t) + \max_{\hat{a}} Q_\psi(s_{t+1}, \hat{a}). \qquad (2)$$

To correct the generalization bias caused by single data augmentation, we gather a collection of diverse augmentations, *e.g.*, random shift [21], random conv [22] and MixUp[45], to construct an augmentation combination denoted as $\Gamma = \{\tau_1, \tau_2, \ldots, \tau_N\}$, where N is the number of augmentations. As shown in the Figure 2, we inject the vanilla observation o_t and all augmented data $\{\tau_1(o_t), \ldots, \tau_N(o_t)\}$ into the network π_θ and then obtain respective estimated state-action value $\{q_t^0, q_t^1, \ldots, q_t^N\}$. As per the recommendation in [14], we employ no data augmentation on the successor observations to maintain a deterministic Q-target $\hat{q}_t$.

To mitigate the high-variance cross various gradient magnitudes, we associate an adaptive weight coefficient w_i to each critic loss term and reformulate it as follows:

$$\mathcal{L}_\theta = \sum_{i=0}^N \sum_{\zeta \in \mathcal{B}} w_i ||Q_\theta^*(\tau_i(o_t), a_t) - \hat{q}_t||^2. \qquad (3)$$

Then, we joint the optimization of coefficient $\mathbf{w}$ into the agent policy training and formulate it as a multi-objective optimization. Motivated by [9], we apply proximal approximation using Taylor series and obtain an optimal solution of the weights $\mathbf{w}$. To directly adjust the overall gradient magnitude, we opt to multiply the weight coefficients w_i with the individual gradient vectors g_i in the practical implementation. Next, we utilize the soft gradient surgery to

locate the conflicting gradient components in the gradient vectors and randomly clip a certain proportion of conflict components to rectify the gradient direction. Considering the above process only involves linear or sign operations, the two modules can be computed independently and use linear multiplication to obtain the final update gradients $\overline{g}$. More details are shown in Figure 2.

4.2. Gradient Agreement Solver

To avoid some dominant augmentations to misguide the policy gradient optimization, we devise gradient agreement solver to adaptively assign weight coefficients $\mathbf{w}$ to all loss items, which can directly affect the gradient magnitude. Here, we aim to simultaneously optimize the policy parameters θ and find optimal coefficients $\mathbf{w}$, and thus we model it as a two order multi-objective optimization process. Additionally, we incorporate $L2$ regularization into the objective function to mitigate over-fitting and promote the weights smoothing, defined as follows:

$$\hat{\theta}, \hat{\mathbf{w}} = \arg\min_{\theta, \mathbf{w}} \sum_{i \in T} \mathcal{L}_\theta^i(\theta(\mathbf{w})) + \lambda ||\mathbf{w}||_2^2$$
$$\text{s.t. } \theta(\mathbf{w}) = \arg\min_{\overline{\theta}} w_i \mathcal{L}_\theta^i, (\overline{\theta}), ||\mathbf{w}||_1 = 1, \qquad (4)$$

where λ is a regularization item. Considering that the optimal solution is not influenced by the scale of $\mathbf{w}$, we normalize the magnitude of $\mathbf{w}$ to one by default. A typical method of addressing second order derivative problems is the implicit differentiation [9]. Briefly, implicit differentiation solves second-order optimization problems by differentiating the equation of the objective function, and then solving for the rate of change of the variable being optimized, which is computationally intensive. To reduce com-

putational cost, we assume the objective function have sufficient differentiability and then adopt the Taylor series to derive a proximal approximation. Specifically, we approximate the i-th loss item $\mathcal{L}_\theta^{\tau_i}$ in Eq. (4) using the Taylor series as follows:

$$\mathcal{L}_\theta^i(\theta) \approx \mathcal{L}_\theta^i(\theta^t) + g_i(\theta - \theta^t). \tag{5}$$

By plugging Eq. (5) into Eq. (4) and temporarily ignoring the L_1 normalization constraint, we can obtain the following objective function:

$$\theta^{t+1}, \hat{\mathbf{w}} = \arg\min_{\theta, \mathbf{w}} \sum_{i \in T} [\mathcal{L}_\theta^i(\theta^t) + g_i(\theta(\mathbf{w}) - \theta^t)]$$
$$\text{s.t. } \theta(\mathbf{w}) = \arg\min_{\bar{\theta}} w_i [\mathcal{L}_\theta^i(\theta^t) + g_i(\bar{\theta} - \theta^t)] + \frac{||\bar{\theta} - \theta^t||_2^2}{2\epsilon}, \tag{6}$$

The closed-form solution [1] to the quadratic problem in Eq (6) can be $\theta(\mathbf{w}) = \theta_t - \epsilon \mathbf{w}^T \mathbf{g}$, which is a classical SGD update process, and we compute the derivative:

$$w_i = \lambda \sum_{j \in T} (g_i^T g_j). \tag{7}$$

Finally, we add the L_1 constraint to normalize the weight coefficients $\mathbf{w}$, we obtain w_i as follows:

$$w_i = \frac{\sum_{j=0}^N (g_i^T g_j)}{\sum_{k=0}^N |\sum_{j=0}^N (g_k^T g_j)|}. \tag{8}$$

Intuitively, we can regard the weight w_i as a linear expansion of a dot product between the gradient vector g_i and the average of all gradients. Hence, the GAS entails assigning greater weights to loss items that exhibit well-aligned with the average of all gradients. Consequently, GAS enable to guide the policy optimize towards a direction that exhibits greater agreement among all data augmentations.

4.3. Soft Gradient Surgery

Here, we propose to modify the standard average based gradient descent by incorporating a soft gradient surgery step before updating the neural parameters θ of agent policy. Instead of directly computing similarity between pairwise gradient vectors, we focus on recognizing the element-level gradient conflicting component cross the all gradient items. Our approach aims to adjust the model parameters θ by modifying the gradient updates to point towards conflict-free direction and improve consistency across all data augmentation. Specifically, given a set of gradient vectors (one for each data augmentation), we construct semantic agreement gradients by retaining the components with the same sign and clipping the conflicting components with a damping factor γ to restrain excessive exploration caused by conflicting components. In particular, we introduce the following element gradient mask ϑ as an indicator to determine

---

Algorithm 1: Soft Gradient Surgery.

Input: Hyper-parameters α, β; gradient set
$\qquad \mathbb{G} = \{g_0, g_1, \ldots, g_N\}$
Output: The new gradient set $\overline{\mathbb{G}}$
// flatten each gradients
1 **foreach** g_i *in gradient set* $\mathbb{G}$ **do**
2 $\quad \mid \quad g_i \leftarrow \text{flatten}(g_i)$
3 **end**
4 $M \leftarrow len(g_0)$;
5 $G \leftarrow \text{concat}(g_0, g_1, \ldots, g_N)$;
// obtain the elemental mask
6 **for** $j \leftarrow 1$ **to** M **do**
7 $\quad \mid \quad$ **if** $\sum_{i=0}^N sign(g_i^j) = N + 1$ **then**
8 $\quad \mid \quad \mid \quad \vartheta[j] \leftarrow 1$;
9 $\quad \mid \quad$ **else**
10 $\quad \mid \quad \mid \quad \vartheta[j] \leftarrow 0$;
11 $\quad \mid \quad$ **end**
12 **end**
// randomly clipping the conflicting
gradient components
13 $\gamma \leftarrow$ randomly sample from $\mathbf{U}(\alpha, \beta)$;
14 $G \leftarrow \vartheta \times G + \gamma \times \sim \vartheta \times G$;
15 **for** $i \leftarrow 0$ **to** N **do**
16 $\quad \mid \quad \overline{g_i} \leftarrow \text{reshape}(G[i])$
17 **end**
18 **return** $\{\overline{g_0}, \overline{g_1}, \ldots, \overline{g_N}\}$

---

which component is in conflict:

$$\vartheta[j] = \begin{cases} 1, & \sum_{i=0}^N \text{sign}(g_i^j) = N + 1, \\ 0, & \text{otherwise}, \end{cases} \tag{9}$$

where g_i^j denotes j-th component in the flattened gradient vectors g_i. Note that $\vartheta[\cdot] = 1$ denotes that such gradient component is consistently agree upon all data augmentations, so we preserve the complete gradient information. In contrast, $\vartheta[\cdot] = 0$ indicates the component is in conflict and would be clipped. To precisely constrain the semantic-irrelevant information in conflicting gradient components, we introduce a damping factor γ, sampling from a uniform distribution $\gamma \in \mathbf{U}(\alpha, \beta)$, to control the clipping ratio. Compared to constant clipping, random sampling can effectively enhance the policy exploration ability and prevent falling into local optima. Hence, the gradient update procedure for gradient vector g_i can be written as follows:

$$g_i = \vartheta \times g_i + \gamma \times \sim \vartheta \times g_i, \tag{10}$$

where $\sim \vartheta$ is obtained by applying the bitwise NOT operator to the mask ϑ. The whole update procedure of the SGS algorithm is provided in Algorithm 1. The computational overhead of our SGS is minimal, primarily involving sim-

	DMCGB [15]	SAC [12]	DrQ [21]	DrQv2 [41]	RAD [41]	PAD [13]	SODA [15]	SVEA [14]	TLDA [43]	PIE-G [44]	SGQN [3]	Ours
Random Colors	Walker,Walk	144±19	520±91	168±90	400±61	468±47	692±68	749±61	823±58	884±20	785±57	**902±46**
	Walker,Stand	365±79	770±71	413±61	644±88	797±46	893±12	933±24	947±26	960±15	929±12	**972±23**
	Ball_in_cup,Catch	151±36	365±210	469±99	541±29	563±50	805±28	959±5	932±32	964±7	864±75	**972±10**
	Finger,Spin	316±119	402±208	478±46	667±154	803±72	793±128	912±6	876±45	922±54	905±43	**928±43**
	Cartpole,Swingup	248±24	586±52	277±80	590±53	630±63	949±19	832±23	760±60	749±46	840±13	**856±40**
	Cheetah,Run	76±25	100±27	109±45	121±79	159±28	228±76	273±23	371±51	369±53	162±38	**375±32**

Table 1: **Generalization on random colors environments.** Experiments are conducted on 6 challenging tasks in the DMC-GB. Our CG2A agent perform well over all tasks and exceeds the prior SOTA methods with a significant margin.

ple operations such as sign and addition functions applied to the flatten gradient vectors.

5. Experiments

To evaluate the generalization performance and sample efficiency of our proposed CG2A, we compare it to several state-of-the-art methods on a set of standard tasks from the DMControl Generalization Benchmark (DMC-GB) and two vision based robotic manipulation tasks.

Setup. Following prior works [14, 3], we implement the SAC [12] algorithm with random shift as baseline and adopt the same network architecture and hyper-parameter setup as Hansen *et al.* [15] for all applicable methods. The observation for DMC-GB tasks is a sequence of three consecutive RGB frames with dimensions of $84 \times 84 \times 3$, except for robotic manipulation tasks which use a single frame. Besides, the hyper-parameter α and β in SGS are set as 0.22 and 0.28 respectively. In all experiments, the generalization evaluations are executed in a zero-shot paradigm and we report the average result over 5 times.

Baselines and Data Augmentations. To evaluate the generalization capability of our CG2A, we benchmark CG2A against strong baselines and several state-of-the-art methods: SAC [12], DrQ [21], DrQv2 [41], RAD [22], PAD [13] SODA [15], SVEA [14], TLDA [43], PIE-G [44], and SGQN [3]. For all compared methods, we report the best performance in the available literature as well as in the reproduced results. Considering that most methods use data augmentation in one of their stages, we adopt random overlay [15] as default, which mixup observations and random images from the Places365 dataset [46] $\mathcal{D}$, as follows:

$$\tau_{overlay}(o) = (1 - \mu)o + \epsilon, \epsilon \in \mathcal{D}_{place}, \quad (11)$$

where $\mu \in [0, 1)$ is the interpolation coefficient and default set as 0.5. For μ values smaller than 0.20, we consider the augmentation to be perceptually insensitive and label it as overlay-S for brevity. To reduce computational overhead, we choose three augmentations ($N = 3$) to construct the augmentation combination, including random conv [22], random overlay [15], random overlay-S [15].

5.1. Evaluation on DMC-GB

The DMC-GB contains a set of vision-based continuous control tasks [34], which allows agents to be trained in a fixed environment and evaluate generalization capability on unseen environments with distribution shifts, including *random colors* and *video backgrounds*. For *video backgrounds* setting, *video easy* benchmark modifies solely the background of images with a distracting image, whereas the hard version extends this modification to include the ground and the shadows, which is more challenging. The training process includes 500,000 interaction steps with 4 action repeats as default, and the agents are evaluated with 100 episodes.

Random Colors. The experimental results, as depicted in Table 1, show that CG2A outperforms prior state-of-the-art methods in all tasks, indicating its superior performance. These results demonstrate that integrating augmentation combination can effectively enhance the robustness of agent to color change in unknown environment, which exposing the potential of augmentation combination mechanism for improving generalization in Visual RL.

Video Background. As illustrated in Table 2, our CG2A surpasses the baselines in 11 out of 12 instances in terms of mean cumulative rewards. Notably, CG2A achieves competitive performance with all prior methods in the context of video easy setting. In particular, for tasks such as "Finger, Spin" and "Cartpole, Swingup", CG2A obtains a substantial profit margin of 8.9% and 10.1% respectively, outperforming other state-of-the-art methods. Additionally, CG2A achieves near-perfect scores in such setting on the "Walker, walk" and "Ball in cup, catch" tasks while significantly reducing the empirical variance to an inconsequential level. Moreover, the agent trained with CG2A demonstrates robust policy acquisition in more challenging video hard environments. These results highlight the effectiveness of CG2A in enhancing the agents' performance and generalization ability in complex and dynamic scenarios.

5.2. Evaluation on Robotic Manipulation Tasks

To validate the performance of the agent in realistic scenarios, we follow prior work [14, 3] and incorporate

	DMCGB [15]	SAC [12]	DrQ [21]	DrQv2 [41]	RAD [22]	PAD [13]	SODA [15]	SVEA [14]	TLDA [43]	PIE-G [44]	SGQN [12]	Ours
Video Easy	Walker,Walk	245±165	682±8	175±117	608±92	717±79	771±66	819±71	873±83	871±22	910±24	**918±20**
	Walker,Stand	389±131	873±83	560±48	879±64	935±20	965±7	961±8	946±6	957±12	955±9	**968±6**
	Ball_in_cup,Catch	192±157	318±157	453±60	363±158	436±55	939±10	871±106	892±68	922±20	950±24	**963±28**
	Finger,Spin	152±8	533±119	456±15	334±54	691±80	535±52	808±23	744±18	837±107	609±61	**912±69**
	Cartpole,Swingup	472±26	485±105	267±41	391±66	521±76	678±120	702±80	671±57	587±61	717±35	**788±24**
	Cheetah,Run	87±21	102±30	64±22	43±21	206±34	184±64	249±20	308±57	287±20	269±33	**314±49**
Video Hard	Walker,Walk	122±47	104±22	34±11	80±10	189±54	312±32	385±63	271±55	600±28	**739±21**	687±18
	Walker,Stand	231±57	289±49	151±13	229±45	411±36	736±132	747±43	602±51	852±56	851±24	**895±35**
	Ball_in_cup,Catch	101±37	92±23	97±27	98±40	174±71	381±163	403±174	257±57	786±47	782±57	**806±44**
	Finger,Spin	25±6	71±45	21±4	15±6	144±19	221±48	335±58	241±29	762±59	540±53	**819±38**
	Cartpole,Swingup	153±22	138±9	130±3	117±22	255±60	339±87	393±45	286±47	401±21	428±60	**472±24**
	Cheetah,Run	28±6	32±13	23±5	21±7	35±22	94±75	105±37	90±27	134±17	144±34	**168±16**

Table 2: **Generalization on video backgrounds environments.** Episode return in two kind of dynamic video background environments, *e.g.*, *video easy* (Top) and *video hard* (Bottom). Bold font indicates the best performance among all methods.

two goal-reaching robotic manipulation tasks, "Reach" and "Peg In Box", from the vision-based robotic manipulation simulator outlined in [18]. To provide a comprehensive view, the RGB camera is positioned in front of the entire setup, providing a third-person view with a large field of view encompassing the robot, target objects, and workspace. The "Reach" is required to locate the goal with a red disc on the table and control the robotic gripper move to there. And "Peg in Box" aim to guide the robot insert a peg affixed to its arm into a box, which is more challenging. The position of the gripper and target objects is randomized in all tasks, and there are significant variations in lighting and texture between the training and testing environments. More training hyper-parameters details and environment descriptions are provided in Appendix.

We train all agents for 250, 000 steps with default setting and evaluate its generalization performance in comparison to the agents trained with SAC [12], SODA [15], SVEA [14], and SGQN [3]. Table 3 demonstrates that all agents trained with prior SOTA fail to maintain their performance when evaluated on the three test environments. Instead, our approach outperforms these baselines in all robotic manipulation tasks, achieving advanced mean cumulative rewards by a significant margin. Particularly, our approach achieves remarkable performance in the first environment (Test1), surpassing the previous methods by 99.3% and 919.5% respectively in the Reach and PegInBox tasks.

5.3. Sample Efficiency

To verify the sample efficiency of our proposed CG2A, we compare our method with prior state-of-the-art methods, including DrQ [21], SVEA [14], and SGQN [3], on DM-Control suite [34] and robotic benchmark. Figure 3 demonstrates that our proposed CG2A agent significantly outperforms the other SOTA agents across selected tasks, in terms of asymptotic performance and sample efficiency, on all

Task	Settings	SAC [12]	SODA [15]	SVEA [14]	SGQN [3]	Ours
Reach	train	9.7±2	31.8±1	32.2±5	31.8±1	**39.6±4**
	test1	-20.9±16	-30.9±43	-17.6±10	14.4±14	**28.7±1**
	test2	-21.9±14	-20.2±29	-2.1±39	31.0±3	**36.7±4**
	test3	-43.2±6	-68.4±30	1.4±29	29.2±7	**35.4±4**
PegInBox	train	-46.7±7	180.1±2	177.5±3	183.9±9	**189.9±11**
	test1	-20.9±16	16.9±44	-21.3±10	-72.0±14	**155.4±17**
	test2	-21.9±14	0.7±30	96.8±42	110.7±3	**157.8±22**
	test3	-43.2±6	73.6±31	40.5±28	154.6±7	**174.0±21**

Table 3: **Generalization on robotic manipulation tasks.** Our CG2A significantly outperforms other methods by a large margin in both tasks, with only a slight decrease in performance observed across all testing scenarios.

evaluation settings. Compared with SVEA [14], our method shows better performance and stability with a smaller variance. Notably, SGQN [3] exhibits severe performance collapse on some tasks, which can be attributed to the limitations of saliency-based self-learning. Once the auxiliary task parameters get into a local dilemma, it can cause the performance of the RL agent to crash to maintain gradient balance. Therefore, our CG2A avoids the introduction of any learnable parameters for constructing the auxiliary task to ensure training stability. Besides, the CG2A converges to optimal performance for the 'Ball_in_Cup, Catch' and 'Cartpole, Swingup' tasks at least 100,000 training steps earlier than other methods like SVEA and SGQN. These experimental results demonstrate that the utilization of data augmentation combinations not only enhances the policy's generalization ability but also effectively improves sampling efficiency and training stability.

5.4. Ablation Study

To evaluate the effectiveness of our proposed CG2A, we conduct comprehensive ablation analyses to closely val-
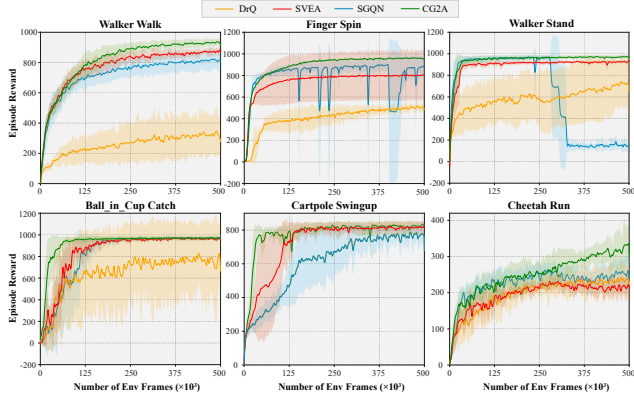
Figure 3: **Training sample efficiency.** Comparison of CG2A (Green Line) with sample-efficient RL algorithms, including DrQ [21] (Yellow Line), SVEA [14] (Red Line) and SGQN [3] (Blue Line). Our method achieve better performance on all tasks.

AC	GAS	SGS	Walker, walk	Walker, stand	ball_in_cup, run	Finger, spin
			144 ±34	289 ±49	92 ±23	71 ±45
✓			274 ±78	557 ±87	418 ±56	473 ±64
✓	✓		424 ±42	618 ±21	563 ±90	474 ±77
✓		✓	619 ±38	805 ±41	724 ±53	757 ±23
✓	✓	✓	**687** ±18	**895** ±35	**806** ±44	**819** ±38

Table 4: Ablation study of individual components.

Figure 4: Ablation study of the damping factor γ.

idate the individual components of the CG2A. All these agents are trained on four standard tasks from DMControl suite [34] with 500, 000 training steps and evaluated on the challenging *video hard* benchmark. More ablation results about additional tasks and various augmentation combinations are provided in Appendix.

Effectiveness of Individual Components. Compared with vanilla SAC algorithm, CG2A gathers multiple data augmentations to construct augmentation combination and enhance the SAC architecture with an adaptive weight solver GAS and a conflict-aware gradient fusion strategy named SGS. We perform an ablation study to investigate the effectiveness of individual components in CG2A and the results are shown in Table 4. Individually, each of these features contributes significantly to the improvement of generalization performance across all environments. The introduction of augmentation combination provides great performance gains over vanilla SAC, which can achieve the comparable performance with SVEA [14]. Notably, agents trained with naive augmentation combination lead to higher performance variances than the vanilla SAC agents. Heavy data augmentations lead to higher performance variances against the vanilla over all tasks, which also leads to the severe gradient conflicts. The SGS strategy provides the most significant performance gains and stabilize the training process through randomly clipping conflicting gradient components. The impact of hyper-parameter γ in the SGS is also illustrate in the next section. Experimental results suggest that the agents trained with the complete CG2A achieve the most superior generalization performance on all tasks.

Impact of Damping Factor γ. In our experiments, the value of the damping factor γ was sampled from a uniform distribution with hyper-parameters α and β, which were ob-

tained through grid search. To assess the sensitivity of the hyper-parameter γ in SGS, we compared it with constant values of $\gamma \in 0, 0.2, 0.3$ and other random distributions. Figure 4 shows that our method is robust to variations in the hyper-parameter γ and achieves superior performance on most tasks. Notably, when γ is set to 0, some gradient components are removed, limiting its performance upper bound.

6. Conclusion

In this paper, we integrate augmentation combination into visual RL to eliminate the generalization bias induced by single data augmentation, and propose Conflict-aware Gradient Agreement Augmentation, which can efficiently harmonize gradients with high-variance magnitudes and significantly mitigates performance degradation caused by gradient conflicts. Experimental results demonstrate that our method achieves state-of-the-art generalization performance with great sample efficiency. In the future, we will further explore the impact of the augmentation combination composition on generalization performance.

References

[1] JS Arora, OA Elwakeil, AI Chahande, and CC Hsieh. Global optimization methods for engineering applications: a review. *Structural optimization*, 9:137–159, 1995.

[2] Kai Arulkumaran, Marc Peter Deisenroth, Miles Brundage, and Anil Anthony Bharath. Deep reinforcement learning: A brief survey. *IEEE Signal Processing Magazine*, 34(6):26–38, 2017.

[3] David Bertoin, Adil Zouitine, Mehdi Zouitine, and Emmanuel Rachelson. Look where you look! saliency-guided q-networks for visual RL tasks. *CoRR*, abs/2209.09203, 2022.

[4] Zhaoyu Chen, Bo Li, Shuang Wu, Shouhong Ding, and Wenqiang Zhang. Query-efficient decision-based black-box patch attack. *arXiv preprint arXiv:2307.00477*, 2023.

[5] Zhaoyu Chen, Bo Li, Shuang Wu, Kaixun Jiang, Shouhong Ding, and Wenqiang Zhang. Content-based unrestricted adversarial attack. *arXiv preprint arXiv:2305.10665*, 2023.

[6] Zhaoyu Chen, Bo Li, Shuang Wu, Jianghe Xu, Shouhong Ding, and Wenqiang Zhang. Shape matters: deformable patch attack. In *Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part IV*, pages 529–548. Springer, 2022.

[7] Zhaoyu Chen, Bo Li, Jianghe Xu, Shuang Wu, Shouhong Ding, and Wenqiang Zhang. Towards practical certifiable patch defense with vision transformer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15148–15158, 2022.

[8] Terrance DeVries and Graham W Taylor. Improved regularization of convolutional neural networks with cutout. *arXiv preprint arXiv:1708.04552*, 2017.

[9] Justin Domke. Generic methods for optimization-based modeling. In *Artificial Intelligence and Statistics*, pages 318–326. PMLR, 2012.

[10] Jean-Bastien Grill, Florian Strub, Florent Altché, Corentin Tallec, Pierre H. Richemond, Elena Buchatskaya, Carl Doersch, Bernardo Ávila Pires, Zhaohan Guo, Mohammad Gheshlaghi Azar, Bilal Piot, Koray Kavukcuoglu, Rémi Munos, and Michal Valko. Bootstrap your own latent - A new approach to self-supervised learning. In *NeurIPS 2020, December 6-12, 2020, virtual*, 2020.

[11] Shixiang Gu, Ethan Holly, Timothy Lillicrap, and Sergey Levine. Deep reinforcement learning for robotic manipulation with asynchronous off-policy updates. In *2017 IEEE international conference on robotics and automation (ICRA)*, pages 3389–3396. IEEE, 2017.

[12] 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.

[13] Nicklas Hansen, Rishabh Jangir, Yu Sun, Guillem Alenyà, Pieter Abbeel, Alexei A Efros, Lerrel Pinto, and Xiaolong Wang. Self-supervised policy adaptation during deployment. *arXiv preprint arXiv:2007.04309*, 2020.

[14] Nicklas Hansen, Hao Su, and Xiaolong Wang. Stabilizing deep q-learning with convnets and vision transformers under data augmentation. In Marc'Aurelio Ranzato, Alina Beygelzimer, Yann N. Dauphin, Percy Liang, and Jennifer Wortman Vaughan, editors, *NeurIPS 2021, December 6-14, 2021, virtual*, pages 3680–3693, 2021.

[15] Nicklas Hansen and Xiaolong Wang. Generalization in reinforcement learning by soft data augmentation. In *IEEE International Conference on Robotics and Automation, ICRA 2021, Xi'an, China, May 30 - June 5, 2021*, pages 13611–13617. IEEE, 2021.

[16] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 9729–9738, 2020.

[17] David Isele, Reza Rahimi, Akansel Cosgun, Kaushik Subramanian, and Kikuo Fujimura. Navigating occluded intersections with autonomous vehicles using deep reinforcement learning. In *2018 IEEE international conference on robotics and automation (ICRA)*, pages 2034–2039. IEEE, 2018.

[18] 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.

[19] Leslie Pack Kaelbling, Michael L. Littman, and Anthony R. Cassandra. Planning and acting in partially observable stochastic domains. *Artif. Intell.*, 101(1-2):99–134, 1998.

[20] Robert Kirk, Amy Zhang, Edward Grefenstette, and Tim Rocktäschel. A survey of zero-shot generalisation in deep reinforcement learning. *J. Artif. Intell. Res.*, 76:201–264, 2021.

[21] Ilya Kostrikov, Denis Yarats, and Rob Fergus. Image augmentation is all you need: Regularizing

deep reinforcement learning from pixels. *ArXiv*, abs/2004.13649, 2020.

[22] 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.

[23] 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.

[24] Siao Liu, Zhaoyu Chen, Wei Li, Jiwei Zhu, Jiafeng Wang, Wenqiang Zhang, and Zhongxue Gan. Efficient universal shuffle attack for visual object tracking. In *ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 2739–2743. IEEE, 2022.

[25] Yang Liu, Jing Liu, Kun Yang, Bobo Ju, Siao Liu, Yuzheng Wang, Dingkang Yang, Peng Sun, and Liang Song. Amp-net: Appearance-motion prototype network assisted automatic video anomaly detection system. *IEEE Transactions on Industrial Informatics*, pages 1–13, 2023.

[26] Yang Liu, Jing Liu, Mengyang Zhao, Shuang Li, and Liang Song. Collaborative normality learning framework for weakly supervised video anomaly detection. *IEEE Transactions on Circuits and Systems II: Express Briefs*, 69(5):2508–2512, 2022.

[27] Yang Liu, Jing Liu, Xiaoguang Zhu, Donglai Wei, Xiaohong Huang, and Liang Song. Learning task-specific representation for video anomaly detection with spatial-temporal attention. In *ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 2190–2194. IEEE, 2022.

[28] Yang Liu, Dingkang Yang, Yan Wang, Jing Liu, and Liang Song. Generalized video anomaly event detection: Systematic taxonomy and comparison of deep models. *arXiv preprint arXiv:2302.05087*, 2023.

[29] Samuel G Müller and Frank Hutter. Trivialaugment: Tuning-free yet state-of-the-art data augmentation. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 774–782, 2021.

[30] Giambattista Parascandolo, Alexander Neitz, Antonio Orvieto, Luigi Gresele, and Bernhard Schölkopf. Learning explanations that are hard to vary. *arXiv preprint arXiv:2009.00329*, 2020.

[31] Roberta Raileanu, Maxwell Goldstein, Denis Yarats, Ilya Kostrikov, and Rob Fergus. Automatic data aug-mentation for generalization in reinforcement learning. In *Neural Information Processing Systems*, 2021.

[32] Reda Bahi Slaoui, William R Clements, Jakob N Foerster, and Sébastien Toth. Robust domain randomization for reinforcement learning. 2019.

[33] Matthijs TJ Spaan. Partially observable markov decision processes. *Reinforcement learning: State-of-the-art*, pages 387–414, 2012.

[34] Yuval Tassa, Yotam Doron, Alistair Muldal, Tom Erez, Yazhe Li, Diego de Las Casas, David Budden, Abbas Abdolmaleki, Josh Merel, Andrew Lefrancq, et al. Deepmind control suite. *arXiv preprint arXiv:1801.00690*, 2018.

[35] Fei-Yue Wang, Jun Jason Zhang, Xinhu Zheng, Xiao Wang, Yong Yuan, Xiaoxiao Dai, Jie Zhang, and Liuqing Yang. Where does alphago go: From church-turing thesis to alphago thesis and beyond. *IEEE/CAA Journal of Automatica Sinica*, 3(2):113–120, 2016.

[36] Kaixin Wang, Bingyi Kang, Jie Shao, and Jiashi Feng. Improving generalization in reinforcement learning with mixture regularization. In Hugo Larochelle, Marc'Aurelio Ranzato, Raia Hadsell, Maria-Florina Balcan, and Hsuan-Tien Lin, editors, *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*, 2020.

[37] Yuzheng Wang, Zhaoyu Chen, Dingkang Yang, Yang Liu, Siao Liu, Wenqiang Zhang, and Lizhe Qi. Adversarial contrastive distillation with adaptive denoising. In *ICASSP 2023-2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1–5. IEEE, 2023.

[38] Dingkang Yang, Zhaoyu Chen, Yuzheng Wang, Shunli Wang, Mingcheng Li, Siao Liu, Xiao Zhao, Shuai Huang, Zhiyan Dong, Peng Zhai, and Lihua Zhang. Context de-confounded emotion recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 19005–19015, June 2023.

[39] Dingkang Yang, Shuai Huang, Haopeng Kuang, Yangtao Du, and Lihua Zhang. Disentangled representation learning for multimodal emotion recognition. In *Proceedings of the 30th ACM International Conference on Multimedia (ACM MM)*, pages 1642–1651, 2022.

[40] Dingkang Yang, Shuai Huang, Shunli Wang, Yang Liu, Peng Zhai, Liuzhen Su, Mingcheng Li, and Lihua Zhang. Emotion recognition for multiple context awareness. In *Proceedings of the European Conference on Computer Vision (ECCV)*, volume 13697, pages 144–162, 2022.

[41] Denis Yarats, Rob Fergus, Alessandro Lazaric, and Lerrel Pinto. Mastering visual continuous control: Improved data-augmented reinforcement learning. *arXiv preprint arXiv:2107.09645*, 2021.

[42] Tianhe Yu, Saurabh Kumar, Abhishek Gupta, Sergey Levine, Karol Hausman, and Chelsea Finn. Gradient surgery for multi-task learning. *Advances in Neural Information Processing Systems*, 33:5824–5836, 2020.

[43] 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. In Luc De Raedt, editor, *IJCAI 2022, Vienna, Austria, 23-29 July 2022*, pages 3702–3708. ijcai.org, 2022.

[44] Zhecheng Yuan, Zhengrong Xue, Bo Yuan, Xueqian Wang, Yi Wu, Yang Gao, and Huazhe Xu. Pre-trained image encoder for generalizable visual reinforcement learning. *CoRR*, abs/2212.08860, 2022.

[45] Hongyi Zhang, Moustapha Cisse, Yann N Dauphin, and David Lopez-Paz. mixup: Beyond empirical risk minimization. *arXiv preprint arXiv:1710.09412*, 2017.

[46] Bolei Zhou, Agata Lapedriza, Aditya Khosla, Aude Oliva, and Antonio Torralba. Places: A 10 million image database for scene recognition. *IEEE transactions on pattern analysis and machine intelligence*, 40(6):1452–1464, 2017.