Adapt2Reward: Adapting Video-Language Models to Generalizable Robotic Rewards via Failure Prompts

Yanting Yang[1⋆], Minghao Chen[2⋆†], Qibo Qiu[3,7], Jiahao Wu[4], Wenxiao Wang[1], Binbin Lin[1,5], Ziyu Guan[6], and Xiaofei He[7]

[1]School of Software Technology, Zhejiang University,
[2]School of Computer Sciene and Technology, Hangzhou Dianzi University,
[3] China Mobile (Zhejiang) Research & Innovation Institute,
[4] The Hong Kong Polytechnic University, [5] Zhiyuan Research Institute,
[6] School of Computer Sciene and Technology, Xidian University,
[7] State Key Lab of CAD&CG, Zhejiang University
{yantingyang,qiuqibo_zju,wenxiaowang,binbinlin}@zju.edu.cn,
minghaochen01@gmail.com, jiahao.wu@connect.polyu.hk,
zyguan@xidian.edu.cn, xiaofeihe@cad.zju.edu.cn

Abstract. For a general-purpose robot to operate in reality, executing a broad range of instructions across various environments is imperative. Central to the reinforcement learning and planning for such robotic agents is a generalizable reward function. Recent advances in vision-language models, such as CLIP, have shown remarkable performance in the domain of deep learning, paving the way for open-domain visual recognition. However, collecting data on robots executing various language instructions across multiple environments remains a challenge. This paper aims to transfer video-language models with robust generalization into a generalizable language-conditioned reward function, only utilizing robot video data from a minimal amount of tasks in a singular environment. Unlike common robotic datasets used for training reward functions, human video-language datasets rarely contain trivial failure videos. To enhance the model's ability to distinguish between successful and failed robot executions, we cluster failure video features to enable the model to identify patterns within. For each cluster, we integrate a newly trained failure prompt into the text encoder to represent the corresponding failure mode. Our language-conditioned reward function shows outstanding generalization to new environments and new instructions for robot planning and reinforcement learning.

1 Introduction

Recent research in the field of "generalist robots" [4, 16, 45, 48] aims to equip robots with the ability to perform a wide range of tasks conditioned on natural language instructions. This emerging field is propelled by breakthroughs in

⋆ Equal contribution;† Corresponding author

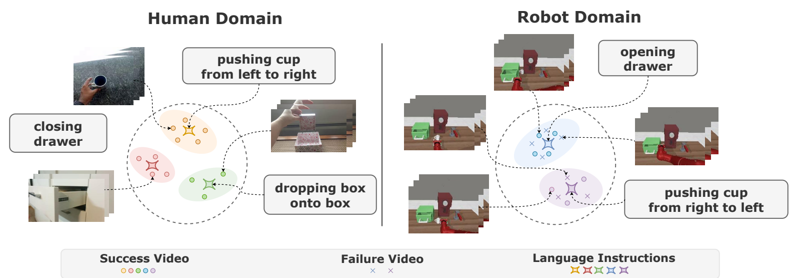


Fig. 1: Human video-language datasets typically lack failure videos. This limitation will result in models that are effective at categorizing tasks but exhibit diminished efficiency in distinguishing between successful and unsuccessful task executions.

multi-modal models, such as CLIP [34], BLIP [21], and DALLE-2 [36], known for their exceptional proficiency in vision-language tasks. Their success stems from training on comprehensive datasets encompassing extensive human data across varied environments. Inspired by these achievements, some notable works in robotics [3,4,16,48] have employed rich and diverse robotic interaction datasets. However, the collection of such comprehensive, high-quality robotic data is considerably more challenging and labor-intensive than collecting datasets for computer vision and natural language processing [34,39,40,53,58].

One promising approach to tackling these challenges is developing a generalized reward function, which can assess the success of robotic behaviors. Traditional methods, e.g., inverse reinforcement learning [1], often result in specialized reward functions limited to a narrow range of tasks. In this paper, we attempt to adapt pre-trained vision-language models as reward functions. Such models potentially offer broad generalization [30,35]. However, applying these models to robotic reward learning is not straightforward. Firstly, due to the scarcity of robotic interaction data, fine-tuning these models on limited robotic datasets risks model overfitting and catastrophic forgetting, undermining their generalization ability. Secondly, there's a significant domain shift and embodiment gap when transitioning from human-centric data to robotic applications. Unlike human videos, which feature diverse environments and perspectives, robot videos typically originate from more controlled, static settings, and the mechanical nature of robotic movements differs markedly from the fluidity of human actions. Previous work, i.e., DVD [5], has attempted to address these challenges by training a discriminator to classify if two videos perform the same task. However, this approach overlooks the need for a binary classification of success and failure in reward functions, underscoring the necessity for data on failed robotic tasks.

In contrast to typical robotic datasets used for training reward functions [29], human video-language datasets rarely include inconsequential or minor failure videos. This discrepancy often leads to models proficient in categorizing tasks but less capable of distinguishing successful from unsuccessful executions. As Fig. 1 shows, while a video-language model can effectively distinguish the "closing a drawer" task from the "opening a drawer" task, it may not identify failure videos in the task, such as incomplete closure or accidental reopening. Consequently, training solely with successful videos is insufficient for the model to recognize and classify diverse failure scenarios accurately. A straightforward solution is to employ Binary Cross-Entropy (BCE) loss in training to enhance the distinction between success and failure samples. However, this approach can be problematic. The model can assign high scores to success samples in the training set and underscore others, leading to poor performance in novel situations and tasks.

To address these challenges, we introduce a novel method integrating learnable failure prompts within the model architecture. This approach is based on the hypothesis that task failures can be grouped into several modes and potentially transferable across similar tasks. We begin by clustering failure videos to identify distinct failure patterns, assigning each cluster a unique identifier for a corresponding failure prompt. This strategy enables the model to develop a nuanced understanding of different failure modes, facilitating knowledge transfer through learnable failure prompts and improving initial performance on unseen similar tasks. Furthermore, we incorporate cross-domain contrastive learning and domain-specific prompt learning to align text and video representations across human and robot domains.

Our contributions can be summarized as follows:

- We highlight the significance of including failed robotic videos in reward learning with human videos, and we identify the gap between human video-language datasets and robotic datasets in reward learning.
- We propose learnable failure prompts to model patterns of robotic failures effectively. The introduction of these prompts significantly enhances the model's adaptability and applicability.
- When combined with Visual Model Predictive Control (VMPC), our approach demonstrates superior generalization in the MetaWorld environment, outperforming previous methods. The effectiveness of our reward model for reinforcement learning is also illustrated in the Concept2Robot environment, where it generalizes to unseen tasks with varied viewpoints and objects.

2 Related Works

2.1 Reward Learning

Reinforcement learning (RL) presents a dynamic framework for automating decision-making and control, though it often entails significant engineering of features and rewards for practical use. To address these challenges, inverse reinforcement learning (IRL) [1] aims to infer experts' reward functions from observed behaviors, as extensively discussed in the literature [10,11,15,54,57,63]. Despite expansions to scenarios involving human-provided outcomes or demonstrations [6, 12, 22, 46, 52, 62], existing research tends to focus on single tasks within confined environments, limiting wider application. Modern research trends are moving towards developing multi-task reward functions using visual inputs [5,29,30,42]. An exemplar is LORel [29], which learns language-driven skills from sub-optimal offline data and crowd-sourced annotations. However, obtaining high-quality robot demonstrations remains resource-intensive. In contrast, our approach leverages extensive human-centric datasets, circumventing the need for robot hardware and capitalizing on the availability of online resources.

2.2 Robotic Learning from Human Video

Extensive research has been devoted to learning robotic behaviors from human videos. A prevalent method involves transforming human trajectories into robotic motions [18,19,31,37,44,50,59], achieved by identifying and tracking hand positions in human videos and subsequently aligning them with corresponding robotic actions or primitives for task execution. Another tactic is pixel-level transfer, where human demonstrations or goals are directly transposed into a robotic framework [24, 43, 47], employing both paired and unpaired datasets. Recent research has refined self-supervised algorithms to produce embeddings that are sensitive to object interactions and postures, while minimizing the impact of non-essential factors such as viewpoint and embodiment [30,33,41,55,62]. Furthermore, recent advancements have recognized platforms like YouTube as substantial sources of "in-the-wild" visual data, featuring diverse human interactions, such as Something-Something-v2 [13]. Robots that can assimilate and learn reward functions from this extensive array of data have the potential for broad generalization. For instance, Concept2Robot [42] utilizes a pre-trained video classifier to infer robot reward functions. Similarly, DVD [5] proposes a domain-agnostic video discriminator, aimed at facilitating generalizable reward learning. LIV [26] inducing a cross-modal embedding with temporal coherence and semantic alignment for language-image reward.

2.3 Language-Conditioned Robotic Learning

Recently, CLIP [34] aligns vision and language features from millions of image-caption pairs sourced from the internet, which is a robust foundation established for grounding semantic concepts prevalent across tasks. Therefore, prior research has predominantly concentrated on end-to-end learning of intricate robot manipulation, leveraging the multi-environment and multi-task robotic datasets, such as BC-Z [16], CLIPort [45], and RT-1 [4]. Recent studies [7, 9, 17, 28] have also advocated for leveraging pretrained foundation models to generate reward. Diverging from these approaches, most recent studies have utilized Large Language Models (LLMs) for automated generation of dense [27, 56] or sparse [61] reward coding for policy learning or action synthesis in robotics. These methods, however, require extremely Large Language Models, i.e., GPT-4 [32], to generate accurate code for each task and are more suited for complex tasks requiring deep semantic understanding. Our work deviates from previous efforts by circumventing the challenges of collecting diverse robotic data and the requirements of super-large models.

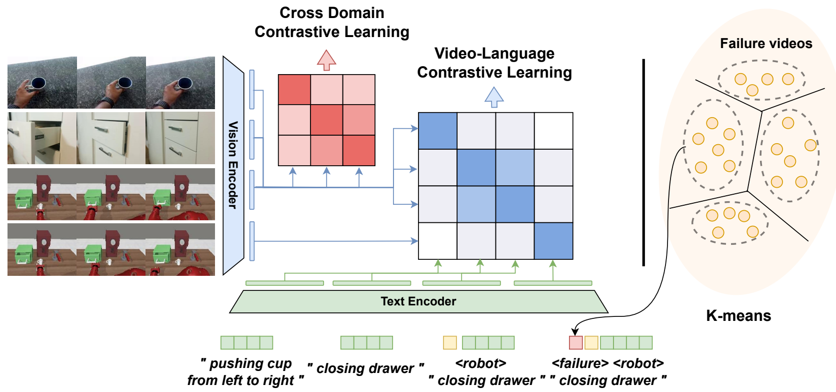


Fig. 2: Adapt2Reward Architecture. We propose Adapt2Reward which incorporates learnable failure prompts into the model's architecture. Our approach starts with clustering failure videos to discern specific patterns. Each identified cluster is then associated with a unique failure prompt. Additionally, we employ cross-domain contrastive learning and domain-specific prompt learning to align text and video representations between human and robot domains.

3 Methods

3.1 Preliminaries

Problem Statement. In the setting of our problem, we consider a robotic agent to accomplish tasks from a task distribution, $\mathcal{T} \sim \mathcal{D}_{\mathcal{T}}$. Each task has some underlying reward function R and can be expressed in natural language l. Consequently, for a specified task $\mathcal{T}$, our robotic agent operates in a fixed horizon Markov decision process: $(\mathcal{S}, \mathcal{A}, p, \mathcal{R}, T)$, where $\mathcal{S}$ denoted as the state space (image or short video clip in our case), $\mathcal{A}$ is the action space of the robot, $p(s_{t+1}|s_t, a_t)$ represents the stochastic dynamics of robotic environments, $\mathcal{R}$ displays the reward for task $\mathcal{T}$, and T is the episode horizon. Our target is to learn a parametric model $\mathcal{R}_\theta$ that estimates the underlying reward function $\mathcal{R}$ for each task $\mathcal{T}$, conditioned on the natural language expression l. With such a reward function, we can utilize open-loop planners or reinforcement learning to optimize the reward for each task.

In order to learn a broadly generalized reward function $\mathcal{R}_\theta$, we have access to human dataset $\{(x_i^h, l_i^h)\}_{i=1}^N$, consisting of N videos of human demonstration for various tasks, where l_i^h is a sentence expressing a task $\mathcal{T}$ sampled from the human task distribution $\mathcal{T} \sim \mathcal{D}_{\mathcal{T}}^h$. We are also given a finite robot dataset $\{(x_j^r, l_j^r, r_j)\}_{j=1}^M$ of videos of the robot doing tasks $\mathcal{T} \sim \mathcal{D}_{\mathcal{T}}^r$, in which the robot can complete the task, $r_j = 1$, or the robot failed to finish the task, $r_j = 0$. Although human data is widely available, robot data consists of only a limited number of tasks in a handful of environments, $\mathcal{D}_{\mathcal{T}}^r \subset \mathcal{D}_{\mathcal{T}}$. Therefore, we have many more diverse human video demonstrations than robot video demonstrations per task and many more tasks that have human videos but not robot videos, in which case $\mathcal{D}_{\mathcal{T}}^{r} \subset \mathcal{D}_{\mathcal{T}} \subset \mathcal{D}_{\mathcal{T}}^{h}$. We only use visual observations to determine whether a task is completed or not, and do not use low-dimensional states or actions. What is clear is that we will also face a large domain shift in the human and robot domains. By learning such a generalizable and robust reward model, we expect it to be able to complete these tasks in new environments and to generalize to unseen tasks.

VLM as reward model. Learning multi-modal representations using large-scale video-text pretraining has proved to be effective for a wide range of uni-modal and multi-modal applications that allows us to train multi-task success detectors by directly leveraging powerful pretrained Vision Language Model (VLM), such as CLIP4Clip [25] and Singularity [20]. In our work, the VLM is given a visual input representing the state of the world (a short video clip $x_i \in \mathbb{R}^{L\times H\times W\times C}$ with L frames), and a corresponding text description l_i. The vision encoder encodes the T frames as a batch of images independently and feeds these frame-level features to a temporal vision encoder to obtain fine-grained temporal video representation $v_i \in \mathbb{R}^D$. For the text describing the desired behavior or task, we denote the representation as $t_i \in \mathbb{R}^D$. The video and text features are input to the multimodal encoder to get a video-level prediction score, which can serve as a reward.

3.2 Human-Robot Contrastive learning

Cross Domain Contrastive Learning. We present our methodology for leveraging contrastive learning to learn domain-invariant features by forming pairs across domains. Specifically, we hope that samples within the same category, irrespective of their domain origin, are positioned closely in feature space, while those from distinct classes are separated regardless of domain. More formally, we consider l_2-normalized features v_i^r from the i-th sample (x_i^r, l_i^r) in the robot domain as an anchor, and it forms a positive pair with the sample having the same expression from the human domain and robot domain, whose features are denoted as v_p. We formulate the cross-domain contrastive loss (CDC) as:

$$\mathcal{L}_{CDC}^{r,i} = -\frac{1}{|P(l_i^r)|} \sum_{p\in P(l_i^r)} \log \frac{\exp(\boldsymbol{v}_i^{r\top} \boldsymbol{v}_p/\tau)}{\sum_{j=1}^{B} \exp(\boldsymbol{v}_i^{r\top} \boldsymbol{v}_j/\tau)} \tag{1}$$

where $P(l_i^r) = \{k|l_k^h = l_i^r, k \in \{1,..,B_h\}\} \cup \{k|l_k^r = l_i^r, k \in \{1,..,B_r\}\}$ indicates the set of positive samples from the cross-domain that share the same label with the anchor x_i^r and $B = B_h + B_r$ denotes the batch size. In each batch, we sample B_h human samples from the human dataset, and B_r successful robot samples $(r_j = 1)$ from the robot dataset. In Eqn. 1, we consider samples from the robot domain as anchors. Alternatively, we can use human samples as anchors and compute $\mathcal{L}_{CDC}^{h,i}$ Then, we combine $\mathcal{L}_{CDC}^{r,i}$ with $\mathcal{L}_{CDC}^{h,i}$ to derive the cross-domain contrastive loss as follows:

$$\mathcal{L}_{CDC} = \sum_{i=1}^{B_h} \mathcal{L}_{CDC}^{h,i} + \sum_{i=1}^{B_r} \mathcal{L}_{CDC}^{r,i} \tag{2}$$

Video-Language Contrastive Learning. To promote the model's ability to capture semantically pertinent features across human and robot domains, we employ a video-language contrastive loss. This approach, distinct from conventional video-language alignment, aims to minimize discrepancies in both domains. It not only aligns temporal dynamics with task semantics but also enhances the adaptation of video features by leveraging the shared action semantics across both domains. Formally, we denote the video-text paired features (v_i, t_i) from the human or robot domain, where $v_i \in \left\{v_i^r, v_i^h\right\}$ and $t_i \in \left\{t_i^r, t_i^h\right\}$. The modified video-language contrastive loss (VLC) is defined as:

$$\mathcal{L}_{VLC}^{v_i,t_i} = \frac{\exp\left(v_i^\top t_i/\tau\right)}{\sum_{j=1}^{B} \exp\left(v_i^\top t_j/\tau\right)} \tag{3}$$

where τ is the temperature. In particular, we minimize the sum of two multimodal contrastive losses:

$$\mathcal{L}_{VLC} = \sum_{i=1}^{B} \mathcal{L}_{VLC}^{v_i,t_i} + \sum_{i=1}^{B} \mathcal{L}_{VLC}^{t_i,v_i} \tag{4}$$

3.3 Learning from failure

In human cognition, task acquisition often hinges on introspective analysis of failures, leading to insights into causal missteps. Consequently, we posit that the integration of robot failure data into reward learning can enrich the model's capability to distinguish between efficacious and errant actions.

Binary Cross-Entropy Loss. To distinguish between successful and failed videos, we adopt Binary Cross-Entropy (BCE) loss, a methodologically straightforward yet effective approach. Concretely, for each training batch, apart from B_h human samples and B_r successful robot samples, we sample B_f failed robot samples $(r_i = 0)$ from the robot dataset. We denote l_2-normalized features (v_i^r, t_i^r) from a sample (x_i^r, l_i^r, r_i) in successful or failed robot samples. We then minimize the binary cross-entropy loss:

$$\mathcal{L}_{BCE} = -\sum_{i=1}^{B_r+B_f} \left[r_i \log(p_i) + (1-r_i)(1-\log(p_i)))\right] \tag{5}$$

where $p_i = \sigma(\boldsymbol{v}_i^{r\top} t_i)$ and σ is the sigmoid function. While BCE loss enhances the model's capability to differentiate between successes and failures in the training dataset, it risks fostering overconfidence in predictions due to the limited size of the robot dataset and insufficient feature granularity. This overconfidence may result in the model disproportionately favoring successful videos from the training set, while marginalizing others. Consequently, this approach could hinder the model's generalization capabilities in novel scenarios or tasks beyond the training dataset.

Failure Prompts. We aim to achieve a deeper understanding and identification of failure patterns and their root causes, rather than merely dismissing all unseen states. Our robot failure dataset comprises videos of failures, each accompanied by corresponding task description. By leveraging the distinct context of each failure, we seek to capture the unique precursors leading to each specific failure instance. Acknowledging the varied reasons for failures across different tasks, we propose the creation of a "failure prompts pool" to achieve this. This pool allows for flexible grouping and integration as input into the model, facilitating a nuanced and task-specific approach to understanding failures. For each task $\mathcal{T} \in \mathcal{D}^r_{\mathcal{T}}$, whose expression is $l_{\mathcal{T}}$, the task-specific prompt pool consists of K learnable prompts:

$$\mathbf{P}^{\mathbf{f}}_{\mathcal{T}} = \{P^f_{\mathcal{T},1}, P^f_{\mathcal{T},2}, ..., P^f_{\mathcal{T},K}\}$$

where $P^f_{\mathcal{T},k} \in \mathbb{R}^{L^f_p \times \mathcal{D}}$ is a prompt with token length L^f_p and the same embedding size D as $y_{\mathcal{T}}$, where $y_{\mathcal{T}} = f^t_e(l_{\mathcal{T}})$ is the embedding features of $l_{\mathcal{T}}$.

We dynamically select suitable prompts for various videos depicting robotic task failures. For each task's failure videos, we utilize spherical K-means clustering to iteratively update the clustering centers at the end of each training epoch. This process enables the assignment of new pseudo-labels to the failure videos, effectively uncovering distinct failure themes specific to each task. Formally, for the task $\mathcal{T}$, we denote $\{v_i\}_{i=1}^{M_{\mathcal{T}}}$ as failure video features encoded by the vision encoder of the current epoch, where $M_{\mathcal{T}}$ devotes the number of videos in this task. The $\mathbf{i}$-th video's pseudo-label $\mathbf{q_i} \in \mathbb{R}^{K \times 1}$ and cluster centers $\mathbf{C}$ are obtained by minimizing the following problem:

$$\min_{\mathbf{C} \in \mathbb{R}^{d \times K}} \frac{1}{M_{\mathcal{T}}} \sum_{i=1}^{M_{\mathcal{T}}} \min_{\mathbf{q_i}} -v_i^{\top} \mathbf{C} \mathbf{q_i} \tag{6}$$

To ensure label stability, aligning clustering results across consecutive epochs is imperative. The assigned pseudo-labels are interpreted as indicators of the respective failure causes. We assume that these pseudo-labels, derived from clustering, succinctly encapsulate the semantic essence of each failed video, thereby elucidating the underlying reasons for failures. Consequently, we select failure prompts based on their corresponding pseudo-label k, leveraging this alignment to foster understanding of failure dynamics. For each task $\mathcal{T} \in \mathcal{D}^r_{\mathcal{T}}$, the input text embeddings of robot failure context are defined as follows:

$$y^f_{\mathcal{T},k} = [P^f_{\mathcal{T},k}; y_{\mathcal{T}}]$$

where ; denotes concatenation along the token length dimension and $k \in \{1...K\}$. By learning K failure prompts for each task, we aim for the model to identify K failure causes per task. Finally, for each epoch, we encode each failure context and get the feature set $t^f_{\mathcal{T}} = \{t^f_{\mathcal{T},1}, t^f_{\mathcal{T},2}, ..., t^f_{\mathcal{T},K}\}$.

In the previous section 3.2, semantically related video and text features are brought closer, and irrelevant or opposite semantic features are pushed away. Beyond that, we think a series of failure texts should also stay away from success videos. Hence, if we denote (v_i, t_i) as a video and language feature pair that completes task $\mathcal{T}$, its corresponding failure text features $t^f_{\mathcal{T}}$ should also be used as a negative sample. We modify video-language contrastive loss for each video-text pair (v_i, t_i) across human and robot domains as below:

$$\mathcal{L}^{v_i,t_i}_{VLC} = \frac{\exp\left(v_i^\top t_i/\tau\right)}{\sum_{j=1}^{B} \exp(v_i^\top t_j/\tau) + \sum_{k=1}^{K} \exp(v_i^\top t^f_{\mathcal{T},k}/\tau)} \tag{7}$$

Considering that different failure texts in the same failure text pool indicate distinct reasons for task failure, the failure video-text correspondences of different categories should be separated. Therefore, we define the failure's video-language contrastive loss for (v^r_i, t^r_i) from the failure robot sample $(x^r_i, l^r_i, r_i = 0)$ as follows:

$$\mathcal{L}^{v^r_i,t^r_i}_{fVLC} = \frac{\exp(v^{r\top}_i t^f_{\mathcal{T},k^*}/\tau)}{\exp(v^{r\top}_i t^r_i/\tau) + \sum_{k=1}^{K} \exp(v^{r\top}_i t^f_{\mathcal{T},k}/\tau)} \tag{8}$$

where k^* represents the index of the failure prompt (cluster) that the sample $(x^r_i, l^r_i, r_i = 0)$ belongs to.

4 Experiments

In our experiments, we aim to transfer video-language models with robust generalization into generalizable language-conditioned reward functions, utilizing robot video data from a minimal amount of tasks in a singular environment. We focus on studying how effectively our method Adapt2Reward can leverage a few successful and failed robot executions and to what extent doing so enables generalization to unseen environments and tasks and enhances the model's ability to discriminate between successful and failed robot executions. Concretely, we study the following questions:

1) Is Adapt2Reward able to generalize to new environments more effectively?

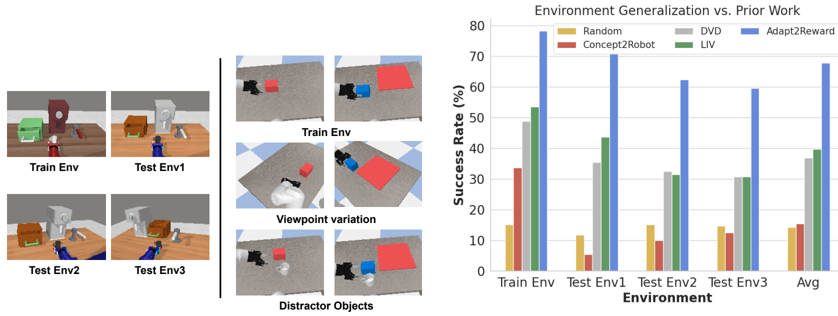
2) Is Adapt2Reward able to generalize to new tasks more effectively?

3) Can Adapt2Reward effectively improve the success rate of robots completing language-instructed tasks?

4) Can Adapt2Reward correctly assign rewards across different viewpoints or distracting objects?

4.1 Experiment Setting

In our experiment, we employ the simulated environment from DVD [5], adapted from Meta-World [60], built upon MuJoCo [49] physical engine, featuring a Sawyer robot arm interacting with various objects including a drawer, faucet, cup, and a coffee machine. We utilize the video-language model Singularity [20], pretrained on 17M human vision-language data and fine-tuned on the Something-Something-V2 (SS-V2) dataset [13]. For human data in our training set, we utilize the Something-Something-V2 dataset, which includes 220,837 videos across



(a) Environment Visualizations **(b)** Environment Generalization Results

Fig. 3: (a) MetaWorld environments (left) consist of the original training environment and three test environments with color, viewpoint, and object arrangement modifications. Concept2Robot environments (right) include the training environment and the testing environments that change the viewpoint or include distractor objects. (b) Comparison of Random Policy, VMPC with Concept2Robot, DVD, LIV, and Adapt2Reward in MetaWorld environments. The depicted bars represent the mean success rate across 4 target tasks computed over 3 seeds of 100 trials.

174 classes of basic human actions with diverse objects and scenes. We select videos from different human tasks. Our robot demonstration data, minimal in volume (560 videos for each task), encompasses successful and failed trajectories within the training environment. Following DVD [5], we train the model with different tasks in different experiments, which will be detailed in the corresponding sections. For the hyper-parameters of Adapt2Reward, we adopt $B_h = B_r = B_f = 8$ and $K = 3$ as defaults. For convenience, we set the trade-off equal to 1 across all losses, including $\mathcal{L}_{CDC}$, $\mathcal{L}_{VLC}$, $\mathcal{L}_{fVLC}$.

Task Execution. Once the reward function $\mathcal{R}\theta$ is trained, we use it to select actions with visual model predictive control (VMPC) [8, 14, 51], which uses a learned visual dynamics model to plan a sequence of actions. We condition $\mathcal{R}\theta$ on a language instruction l for the target task $\mathcal{T}$, using predicted similarity as the reward. We train an action-conditioned video prediction model $p_\phi(s_{t+1:t+H}|s_t, a_{t:t+H})$ with the FitVid framework [2]. Given an input image s_t, we sample G action trajectories of length H and use p_ϕ to predict their future trajectories $\{s_{t+1:t+H}\}^g$. We then score the similarity between the task instruction l and each predicted trajectory using $\mathcal{R}_\theta$. The action trajectory with the highest score is executed.

4.2 Environment Generalization

Ideally, a reward function should be robust to naturalistic visual variations, including changes in camera angles, object colors, or arrangements. Given the impracticality of re-annotating and retraining for each new condition, it's crucial to develop a visually resilient reward function. To explore environmental generalization, we use four progressively challenging variants of a given environment, as

	Human Only	Robot Only	Robot + 3Human Tasks	Robot + 6Human Tasks	Robot + 9Human Tasks	Avg
Random	15.25(0.31)	-	-	-	-	15.25(0.31)
Concept2Robot	33.75(0.20)	-	-	-	-	33.75(0.20)
DVD	-	19.00(1.07)	28.25(0.20)	31.92(2.01)	40.08(0.31)	29.81(0.21)
LIV	-	27.58(1.65)	36.33(0.42)	38.83(1.03)	39.00(1.24)	35.44(0.51)
Adapt2Reward	-	**29.92(0.83)**	**44.67(0.77)**	**69.33(2.42)**	**67.00(1.14)**	**52.73(0.64)**

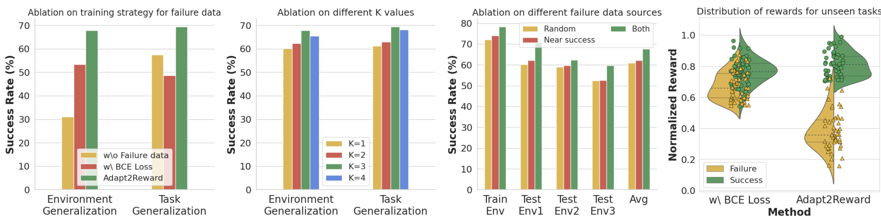
Table 1: Task Generalization with Different Reward Models. Compared with Random Policy, Concept2Robot, DVD, and LIV, Adapt2Reward achieves significantly enhanced performance in novel tasks. We report the average success rate on 4 target tasks, computed over 3 seeds of 100 trials, with the standard deviation in parentheses.

depicted in Fig. 3 (b). These consist of the original training environment (Train Env) with task demonstrations, and three test environments with modifications: Test Env 1 (changed colors), Test Env 2 (changed colors and viewpoint), and Test Env 3 (changed colors, viewpoint, and object arrangement). We evaluate our method on four simulated tasks: (1) *closing drawer*, (2) *moving cup away from the camera*, (3) *moving the handle of the faucet*, and (4) *pushing cup from left to right*. To evaluate our method, we train it on robot videos from the training environment, covering the four tasks with both successful and failed executions. We also incorporated human data from the SS-V2 dataset, encompassing these four tasks and three additional tasks for training.

We compare our approach with previous works. Concept2Robot [42], which uses a pre-trained 174-way action classifier from the SS-V2 dataset, taking the classification score of the predicted robot video as the reward. Unlike Concept2Robot conditioning on the category of the task, DVD [5] calculates rewards by training a domain-agnostic video discriminator to measure the similarity between robot videos and human demonstrations. DVD also involved varying amounts of human data from the SS-V2 dataset, but not including failed robot executions. LIV [26] generates cross-modal embeddings with temporal coherence and semantic alignment, calculating rewards based on current state images and language objectives. We fine-tune LIV and train DVD on the same successful human and robot task videos as Adapt2Reward. We also include a comparison to a random policy. For all reward functions, we use the aforementioned VMPC for action selection to ensure a fair comparison. In Fig. 3, we compare Adapt2Reward with 7 human tasks to these prior methods. Across all environments, Adapt2Reward significantly outperforms the three comparison methods on target tasks, exceeding the best-performing method by over 28% on average.

4.3 Task Generalization

In this experiment, we investigate the impact of incorporating human data on the reward function's capacity to generalize across new tasks. Following DVD [5], we train the Adapt2Reward model on robot videos from three distinct tasks within the training environment without utilizing any robot data from the target tasks.



(a) Failure data **(b)** Cluster Number K **(c)** Failure Data Source **(d)** Distribution

Fig. 4: Ablation study. (a) Different training methods for failure data. (b) Varying K. (c) Training with different sources of failure data. (d) The distribution differences in rewards obtained by different reward methods for an unseen task, with scattered points representing normalized reward values for different trajectories.

Specifically, these three tasks are used for training: (1) *opening drawer*, (2) *pushing cup from right to left*, (3) *poking cup so lightly that it doesn't or almost doesn't move*; the four target tasks are listed in the Section 4.2. Additionally, we incorporate a variable amount of human data into our training regimen. As shown in Table 1, we compare our method with the methods described in our prior experiment. Given our focus on task generalization, all evaluations are conducted within the training environment. To further evaluate the influence of human data on enhancing generalization capabilities, we conduct comparative analyses between our method and prior approaches across various human tasks in Table 1. Following DVD, 'Robot Only' means the reward model is trained solely with robotic data from three training tasks, without any data from four target tasks or any human data. 'Robot + 3 Human Tasks' refers to training with both robot and human videos from the three training tasks. 'Robot + 6 Human Tasks' further incorporates human data from three additional tasks (different from the training and target tasks). In Table 1, the success rates of Adapt2Reward on novel tasks surpass those of previous methods by nearly 30%. Our analysis also shows that training with human videos enhances task generalization by approximately 15-40% over Robot Only, likely due to mitigating overfitting and maintaining human data-derived knowledge. Incorporating 6 human tasks significantly boosts model performance, affirming the efficacy of a hybrid training approach in bridging human and robotic domain knowledge.

4.4 Ablation Study

In this study, we aim to assess the influence of failure data on Adapt2Reward, contrasting two training methodologies: one devoid of failure data and the other utilizing Binary Cross-Entropy (BCE) loss for distinguishing successful from unsuccessful videos. In Fig. 4(a), the average success rates in environmental and task generalization experiments are presented. The results indicate a substantial enhancement in the model's generalization ability with the inclusion of failure data, likely due to learning from a broader range of characteristics and patterns. Moreover, leveraging failure prompts for understanding the causes and

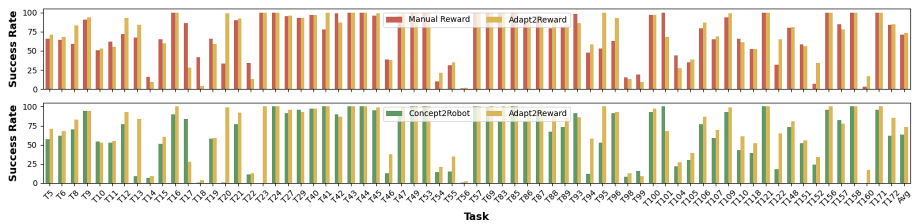


Fig. 5: Task Generalization in C2R-Envs. We report the success rates of reinforcement learning on 68 tasks with manually crafted rewards, Concept2Robot, and Adapt2Reward. The majority of policies obtained by Adapt2Reward match or exceed "manually crafted reward". In comparison to Concept2Robot, Adapt2Reward demonstrated superior performance across most tasks.

contexts of failures proved more effective for generalization than merely using BCE loss, which may compromise the model's learned knowledge from human data. Furthermore, in Fig. 4(b), our ablation study on the cluster number K, indicating the number of failure categories per task, shows our approach is not very sensitive to the hyper-parameter and $K = 3$ as the optimal value.

We also assess how different sources of failure data affect generalization. We use two methods to collect failure data for training: (1) random exploration and (2) near-success scenarios (achieved by adding noise to successful trajectories or collecting from pre-trained RL models). In Fig. 4(c), we present success rates of Adapt2Reward with failure data from different source across several environments. Results indicate that combining failure data from varied sources outperforms using data from a single source, possibly due to a wider range of failure patterns captured from diverse origins.

To demonstrate the superior generalization of Adapt2Reward over models trained with BCE loss for unseen tasks, Fig. 4(d) showcases the reward distribution for various trajectories. Models using BCE loss incorrectly assigne high rewards to failed trajectories. In contrast, Adapt2Reward accurately distribute rewards, clearly differentiating between successful and failed trajectories.

4.5 Concept2Robot Environments Efficacy

To further evaluate our approach, we use Pybullet to simulate each environment associated with 68 tasks. Similar to Concept2Robot [42] Environments (C2R-Envs), the robot is a simulated 7-DoF Franka Panda robot arm with a two-fingered Robotiq 2F-85 gripper. We consider a similar setup as described in Section 4.1, where Adapt2Reward is now trained on robot videos of the 12 robot tasks from the simulation environment including success and failed execution (500 videos for each task), as well as human data of all tasks in SS-V2, and measure performance across seen and unseen robot tasks. In this experiment, unlike the VMPC approach for selecting actions in the previous experiments, we follow the method used in the Concept2Robot [42] and leverage the open-

Task	Concept2Robot	Adapt2Reward
pushing box from left to right	86(-5)	**91(+5)**
pushing box onto book	54(-37)	**81(-12)**
moving book up	89(-6)	**97(-2)**
moving book down	95(-5)	**100(0)**

(a) Viewpoint variation.

Task	Concept2Robot	Adapt2Reward
pushing box from left to right	37(-54)	**66(-20)**
pushing box onto book	39(-52)	**61(-32)**
moving box away from box	53(-44)	**99(+2)**
moving box closer to box	71(-19)	**91(+4)**

(b) Distractor Objects.

Table 2: Visual Robustness. We report the success rates of Concept2Robot and Adapt2Reward trained with different viewpoints or distractor objects in C2R-Envs. Parentheses show the percentage change from an unchanged viewpoint.

loop trajectory generator, which combines deep Deterministic Policy Gradients (DDPG) [23] with the Cross-Entropy Method (CEM) [38] to solve the problem.

In Fig. 5, we display the success rate of 68 tasks under different reward functions. We find that Adapt2Reward-based strategies often equal or surpass "manually crafted reward" strategies. This advantage likely stems from Adapt2Reward's provision of dense reward scores, which favorably assess near-success attempts in contrast to binary ground truth rewards. In further comparisons, Adapt2Reward consistently outperform Concept2Robot, which utilizes the classifier as a reward function, across several tasks, notably Task12 (*dropping a box onto a book*) and Task95 (*pushing a box off a book*). This superiority may result from Concept2Robot's generic templates representing objects as "[Something]" during training, which compromises task-specific object interaction recognition, hindering accurate success state and object relation identification.

Also, We examine reward function robustness by varying viewpoints and objects. Although Adapt2Reward was trained on frames from a fixed camera perspective, we evaluated its performance under different camera angles. As shown in Table 2 (a), changing camera angles significantly impacted Concept2Robot (success rates dropped by 5%-37%), whereas Adapt2Reward showed greater stability (success rates decreased by less than 12%). As shown in Table 2 (b), Adapt2Reward outperformed Concept2Robot by 20-46% in success rates under this new visual arrangement, demonstrating its robustness to visual variations.

5 Conclusion

In our work, we underscore the importance of integrating failed robotic videos into reward learning alongside human videos, addressing the disparity between human video-language datasets and robotic datasets in this context. We introduce learnable failure prompts as an effective method to capture patterns of robotic failures, significantly augmenting the model's adaptability and applicability. Our method shows superior generalization in robotic environments, i.e., Meta-World and Concept2Robot environments, where our reward models are effectively adapted to unseen environments and instructions.

Acknowledgment

This work was supported in part by The National Nature Science Foundation of China (Grant No: 62273303, 62303406), in part by the Key R&D Program of Zhejiang Province, China (2023C01135), in part by Ningbo Key R&D Program (No.2023Z231, 2023Z229), in part by Yongjiang Talent Introduction Programme (Grant No: 2022A-240-G, 2023A-194-G).

References

1. Abbeel, P., Ng, A.: Apprenticeship learning via inverse reinforcement learning. Proceedings of the twenty-first international conference on Machine learning (2004)
2. Babaeizadeh, M., Saffar, M.T., Nair, S., Levine, S., Finn, C., Erhan, D.: Fitvid: Overfitting in pixel-level video prediction. ArXiv **abs/2106.13195** (2021)
3. Brohan, A., Brown, N., Carbajal, J., Chebotar, Y., Choromanski, K., Ding, T., Driess, D., Finn, C., Florence, P.R., Fu, C., Arenas, M.G., Gopalakrishnan, K., Han, K., Hausman, K., Herzog, A., Hsu, J., Ichter, B., Irpan, A., Joshi, N.J., Julian, R.C., Kalashnikov, D., Kuang, Y., Leal, I., Levine, S., Michalewski, H., Mordatch, I., Pertsch, K., Rao, K., Reymann, K., Ryoo, M.S., Salazar, G., Sanketi, P.R., Sermanet, P., Singh, J., Singh, A., Soricut, R., Tran, H., Vanhoucke, V., Vuong, Q.H., Wahid, A., Welker, S., Wohlhart, P., Xiao, T., Yu, T., Zitkovich, B.: Rt-2: Vision-language-action models transfer web knowledge to robotic control. ArXiv **abs/2307.15818** (2023)
4. Brohan, A., Brown, N., Carbajal, J., Chebotar, Y., Dabis, J., Finn, C., Gopalakrishnan, K., Hausman, K., Herzog, A., Hsu, J., Ibarz, J., Ichter, B., Irpan, A., Jackson, T., Jesmonth, S., Joshi, N.J., Julian, R.C., Kalashnikov, D., Kuang, Y., Leal, I., Lee, K.H., Levine, S., Lu, Y., Malla, U., Manjunath, D., Mordatch, I., Nachum, O., Parada, C., Peralta, J., Perez, E., Pertsch, K., Quiambao, J., Rao, K., Ryoo, M.S., Salazar, G., Sanketi, P.R., Sayed, K., Singh, J., Sontakke, S.A., Stone, A., Tan, C., Tran, H., Vanhoucke, V., Vega, S., Vuong, Q.H., Xia, F., Xiao, T., Xu, P., Xu, S., Yu, T., Zitkovich, B.: Rt-1: Robotics transformer for real-world control at scale. ArXiv **abs/2212.06817** (2022)
5. Chen, A.S., Nair, S., Finn, C.: Learning generalizable robotic reward functions from "in-the-wild" human videos. ArXiv **abs/2103.16817** (2021)
6. Das, N., Bechtle, S., Davchev, T., Jayaraman, D., Rai, A., Meier, F.: Model-based inverse reinforcement learning from visual demonstrations. In: Conference on Robot Learning. pp. 1930–1942. PMLR (2021)
7. Du, Y., Konyushkova, K., Denil, M., Raju, A.S., Landon, J., Hill, F., de Freitas, N., Cabi, S.: Vision-language models as success detectors. ArXiv **abs/2303.07280** (2023)
8. Ebert, F., Finn, C., Dasari, S., Xie, A., Lee, A.X., Levine, S.: Visual foresight: Model-based deep reinforcement learning for vision-based robotic control. ArXiv **abs/1812.00568** (2018)
9. Fan, L.J., Wang, G., Jiang, Y., Mandlekar, A., Yang, Y., Zhu, H., Tang, A., Huang, D.A., Zhu, Y., Anandkumar, A.: Minedojo: Building open-ended embodied agents with internet-scale knowledge. ArXiv **abs/2206.08853** (2022)
10. Finn, C., Levine, S., Abbeel, P.: Guided cost learning: Deep inverse optimal control via policy optimization. In: International Conference on Machine Learning (2016)

11. Fu, J., Luo, K., Levine, S.: Learning robust rewards with adverserial inverse reinforcement learning. In: International Conference on Learning Representations (2018)
12. Fu, J., Singh, A., Ghosh, D., Yang, L., Levine, S.: Variational inverse control with events: A general framework for data-driven reward definition. In: Neural Information Processing Systems (2018)
13. Goyal, R., Kahou, S.E., Michalski, V., Materzynska, J., Westphal, S., Kim, H., Haenel, V., Fründ, I., Yianilos, P.N., Mueller-Freitag, M., Hoppe, F., Thurau, C., Bax, I., Memisevic, R.: The "something something" video database for learning and evaluating visual common sense. 2017 IEEE International Conference on Computer Vision (ICCV) pp. 5843–5851 (2017)
14. Hafner, D., Lillicrap, T., Fischer, I., Villegas, R., Ha, D., Lee, H., Davidson, J.: Learning latent dynamics for planning from pixels. In: International conference on machine learning. pp. 2555–2565. PMLR (2019)
15. Jain, A., Hu, M., Ratliff, N.D., Bagnell, D., Zinkevich, M.A.: Maximum margin planning. Proceedings of the 23rd international conference on Machine learning (2006)
16. Jang, E., Irpan, A., Khansari, M., Kappler, D., Ebert, F., Lynch, C., Levine, S., Finn, C.: Bc-z: Zero-shot task generalization with robotic imitation learning. ArXiv **abs/2202.02005** (2022)
17. Kwon, M., Xie, S.M., Bullard, K., Sadigh, D.: Reward design with language models. ArXiv **abs/2303.00001** (2023)
18. Lee, J., Ryoo, M.S.: Learning robot activities from first-person human videos using convolutional future regression. 2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS) pp. 1497–1504 (2017)
19. Lee, K., Su, Y., Kim, T.K., Demiris, Y.: A syntactic approach to robot imitation learning using probabilistic activity grammars. Robotics Auton. Syst. **61**, 1323–1334 (2013)
20. Lei, J., Berg, T.L., Bansal, M.: Revealing single frame bias for video-and-language learning. ArXiv **abs/2206.03428** (2022)
21. Li, J., Li, D., Xiong, C., Hoi, S.C.H.: Blip: Bootstrapping language-image pretraining for unified vision-language understanding and generation. In: International Conference on Machine Learning (2022)
22. Li, Y., Zhao, X., Chen, C., Pang, S., Zhou, Z., Yin, J.: Scenario-driven cyber-physical-social system: Intelligent workflow generation based on capability. Companion Proceedings of the ACM on Web Conference 2024 (2024)
23. Lillicrap, T.P., Hunt, J.J., Pritzel, A., Heess, N.M.O., Erez, T., Tassa, Y., Silver, D., Wierstra, D.: Continuous control with deep reinforcement learning. CoRR **abs/1509.02971** (2015)
24. Liu, Y., Gupta, A., Abbeel, P., Levine, S.: Imitation from observation: Learning to imitate behaviors from raw video via context translation. 2018 IEEE International Conference on Robotics and Automation (ICRA) pp. 1118–1125 (2017)
25. Luo, H., Ji, L., Zhong, M., Chen, Y., Lei, W., Duan, N., Li, T.: Clip4clip: An empirical study of clip for end to end video clip retrieval. Neurocomputing **508**, 293–304 (2021)
26. Ma, Y.J., Liang, W., Som, V., Kumar, V., Zhang, A., Bastani, O., Jayaraman, D.: Liv: Language-image representations and rewards for robotic control. In: International Conference on Machine Learning (2023)
27. Ma, Y.J., Liang, W., Wang, G., Huang, D.A., Bastani, O., Jayaraman, D., Zhu, Y., Fan, L., Anandkumar, A.: Eureka: Human-level reward design via coding large language models. ArXiv **abs/2310.12931** (2023)

28. Ma, Y.J., Sodhani, S., Jayaraman, D., Bastani, O., Kumar, V., Zhang, A.: Vip: Towards universal visual reward and representation via value-implicit pre-training. ArXiv **abs/2210.00030** (2022)
29. Nair, S., Mitchell, E., Chen, K., Ichter, B., Savarese, S., Finn, C.: Learning language-conditioned robot behavior from offline data and crowd-sourced annotation. In: CoRL (2021)
30. Nair, S., Rajeswaran, A., Kumar, V., Finn, C., Gupta, A.: R3m: A universal visual representation for robot manipulation. In: CoRL (2022)
31. Nguyen, A., Kanoulas, D., Muratore, L., Caldwell, D.G., Tsagarakis, N.G.: Translating videos to commands for robotic manipulation with deep recurrent neural networks. 2018 IEEE International Conference on Robotics and Automation (ICRA) pp. 1–9 (2017)
32. OpenAI: Gpt-4 technical report. ArXiv **abs/2303.08774** (2023)
33. Parisi, S., Rajeswaran, A., Purushwalkam, S., Gupta, A.K.: The unsurprising effectiveness of pre-trained vision models for control. In: International Conference on Machine Learning (2022)
34. Radford, A., Kim, J.W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., Krueger, G., Sutskever, I.: Learning transferable visual models from natural language supervision. In: ICML (2021)
35. Radosavovic, I., Xiao, T., James, S., Abbeel, P., Malik, J., Darrell, T.: Real-world robot learning with masked visual pre-training. In: CoRL (2022)
36. Ramesh, A., Dhariwal, P., Nichol, A., Chu, C., Chen, M.: Hierarchical text-conditional image generation with clip latents. ArXiv **abs/2204.06125** (2022)
37. Rothfuss, J., Ferreira, F., Aksoy, E.E., Zhou, Y., Asfour, T.: Deep episodic memory: Encoding, recalling, and predicting episodic experiences for robot action execution. IEEE Robotics and Automation Letters **3**, 4007–4014 (2018)
38. Rubinstein, R.Y., Kroese, D.P.: The cross-entropy method: a unified approach to combinatorial optimization, Monte-Carlo simulation, and machine learning, vol. 133. Springer (2004)
39. Schuhmann, C., Beaumont, R., Vencu, R., Gordon, C., Wightman, R., Cherti, M., Coombes, T., Katta, A., Mullis, C., Wortsman, M., Schramowski, P., Kundurthy, S., Crowson, K., Schmidt, L., Kaczmarczyk, R., Jitsev, J.: Laion-5b: An open large-scale dataset for training next generation image-text models. ArXiv **abs/2210.08402** (2022)
40. Schuhmann, C., Vencu, R., Beaumont, R., Kaczmarczyk, R., Mullis, C., Katta, A., Coombes, T., Jitsev, J., Komatsuzaki, A.: Laion-400m: Open dataset of clip-filtered 400 million image-text pairs. ArXiv **abs/2111.02114** (2021)
41. Sermanet, P., Lynch, C., Chebotar, Y., Hsu, J., Jang, E., Schaal, S., Levine, S.: Time-contrastive networks: Self-supervised learning from video. 2018 IEEE International Conference on Robotics and Automation (ICRA) pp. 1134–1141 (2017)
42. Shao, L., Migimatsu, T., Zhang, Q., Yang, K., Bohg, J.: Concept2robot: Learning manipulation concepts from instructions and human demonstrations. The International Journal of Robotics Research **40**, 1419 – 1434 (2020)
43. Sharma, P., Pathak, D., Gupta, A.K.: Third-person visual imitation learning via decoupled hierarchical controller. In: Neural Information Processing Systems (2019)
44. Shaw, K., Bahl, S., Pathak, D.: Videodex: Learning dexterity from internet videos. In: Conference on Robot Learning (2022)
45. Shridhar, M., Manuelli, L., Fox, D.: Cliport: What and where pathways for robotic manipulation. ArXiv **abs/2109.12098** (2021)

46. Singh, A., Yang, L., Hartikainen, K., Finn, C., Levine, S.: End-to-end robotic reinforcement learning without reward engineering. ArXiv **abs/1904.07854** (2019)
47. Smith, L.M., Dhawan, N., Zhang, M., Abbeel, P., Levine, S.: Avid: Learning multi-stage tasks via pixel-level translation of human videos. ArXiv **abs/1912.04443** (2019)
48. Stone, A., Xiao, T., Lu, Y., Gopalakrishnan, K., Lee, K.H., Vuong, Q.H., Wohlhart, P., Zitkovich, B., Xia, F., Finn, C., Hausman, K.: Open-world object manipulation using pre-trained vision-language models. ArXiv **abs/2303.00905** (2023)
49. Todorov, E., Erez, T., Tassa, Y.: Mujoco: A physics engine for model-based control. 2012 IEEE/RSJ International Conference on Intelligent Robots and Systems pp. 5026–5033 (2012)
50. Wang, C., Fan, L.J., Sun, J., Zhang, R., Fei-Fei, L., Xu, D., Zhu, Y., Anandkumar, A.: Mimicplay: Long-horizon imitation learning by watching human play. ArXiv **abs/2302.12422** (2023)
51. Watter, M., Springenberg, J., Boedecker, J., Riedmiller, M.: Embed to control: A locally linear latent dynamics model for control from raw images. Advances in neural information processing systems **28** (2015)
52. Wu, J., Fan, W., Chen, J., Liu, S., Li, Q., Tang, K.: Disentangled contrastive learning for social recommendation. Proceedings of the 31st ACM International Conference on Information & Knowledge Management (2022),
53. Wu, J., Liu, Q., Hu, H., Fan, W., Liu, S., Li, Q., Wu, X.M., Tang, K.: Leveraging large language models (llms) to empower training-free dataset condensation for content-based recommendation. ArXiv **abs/2310.09874** (2023)
54. Wulfmeier, M., Ondruska, P., Posner, I.: Maximum entropy deep inverse reinforcement learning. arXiv: Learning (2015)
55. Xiao, T., Radosavovic, I., Darrell, T., Malik, J.: Masked visual pre-training for motor control. ArXiv **abs/2203.06173** (2022)
56. Xie, T., Zhao, S., Wu, C.H., Liu, Y., Luo, Q., Zhong, V., Yang, Y., Yu, T.: Text2reward: Automated dense reward function generation for reinforcement learning. ArXiv **abs/2309.11489** (2023)
57. Xu, Y., Jiang, Y., Zhao, X., Li, Y., Li, R.: Personalized repository recommendation service for developers with multi-modal features learning. 2023 IEEE International Conference on Web Services (ICWS) pp. 455–464 (2023)
58. Xu, Y., Qiu, Z., Gao, H., Zhao, X., Wang, L., Li, R.: Heterogeneous data-driven failure diagnosis for microservice-based industrial clouds toward consumer digital ecosystems. IEEE Transactions on Consumer Electronics **70**, 2027–2037 (2024)
59. Yang, Y., Li, Y., Fermüller, C., Aloimonos, Y.: Robot learning manipulation action plans by "watching" unconstrained videos from the world wide web. In: AAAI Conference on Artificial Intelligence (2015)
60. Yu, T., Quillen, D., He, Z., Julian, R.C., Hausman, K., Finn, C., Levine, S.: Meta-world: A benchmark and evaluation for multi-task and meta reinforcement learning. ArXiv **abs/1910.10897** (2019)
61. Yu, W., Gileadi, N., Fu, C., Kirmani, S., Lee, K.H., Arenas, M.G., Chiang, H.T.L., Erez, T., Hasenclever, L., Humplik, J., Ichter, B., Xiao, T., Xu, P., Zeng, A., Zhang, T., Heess, N.M.O., Sadigh, D., Tan, J., Tassa, Y., Xia, F.: Language to rewards for robotic skill synthesis. ArXiv **abs/2306.08647** (2023)
62. Zakka, K., Zeng, A., Florence, P.R., Tompson, J., Bohg, J., Dwibedi, D.: Xirl: Cross-embodiment inverse reinforcement learning. In: Conference on Robot Learning (2021)
63. Ziebart, B.D., Maas, A.L., Bagnell, J.A., Dey, A.K.: Maximum entropy inverse reinforcement learning. In: AAAI Conference on Artificial Intelligence (2008)