# Guide Your Agent with Adaptive Multimodal Rewards

Anonymous Authors[1]

## Abstract

Recent work have shown that incorporating pre-trained multimodal representations can enhance the ability of an instruction-following agent to generalize to unseen situations. Yet training such agents often requires a dataset consisting of diverse demonstrations, which may not be available for target domains and incur a huge cost to collect. In this paper, we instead propose to utilize the knowledge captured within large vision-language models for improving the generalization capability of control agents. To this end, we present Multimodal Reward Decision Transformer (MRDT), a simple yet effective method that uses the visual-text alignment score as a reward. This reward, which adapts based on the progress towards achieving the text-specified goals, is used to train a return-conditioned policy that guides the agent towards the desired goals. We also introduce a fine-tuning scheme that adapts pre-trained multimodal models using in-domain data to improve the quality of rewards. Our experiments demonstrate that MRDT significantly improves generalization performance in test environments with unseen goals. Moreover, we introduce new metrics for evaluating the quality of multimodal rewards and show that generalization performance increases as the quality of rewards improves.

## 1. Introduction

Deep reinforcement learning (RL) and imitation learning (IL) have achieved remarkable success in training visual control agents to solve tasks based on visual observations (Akkaya et al., 2019; Brohan et al., 2022; Mnih et al., 2015; Schrittwieser et al., 2020; Vinyals et al., 2019). However, these approaches frequently struggle to adapt to new test environments, as trained agents tend to overfit specific elements of the training data. This results in a lack of meaningful behavior when faced with different environments (Cobbe et al., 2019; de Haan et al., 2019; Kirk et al., 2021; Song et al., 2020; Zhang et al., 2018), where the agents struggle to exhibit any meaningful behavior due to overfitting to various aspects of training data.

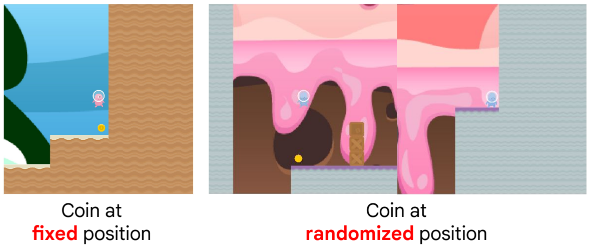


Figure 1: A motivating example of goal misgeneralization from Di Langosco et al. (2022).

In particular, recent work have investigated the *goal misgeneralization* problem, where agents fail to achieve desired goals at test time even if they retain their competencies. This occurs when the agent pursues an undesired goal that was desired during training (Di Langosco et al., 2022; Ngo, 2022; Shah et al., 2022). For instance, an agent trained to collect a coin at a fixed position may learn a behavior that heads towards the fixed position in the environment, instead of collecting the coin, thus failing to learn the intended goal (see Figure 1). Goal misgeneralization poses a significant problem in safety-critical scenarios, as systems that follow misaligned goals can have unexpected negative effects, even if they are proven to be capable in training environments (Amodei et al., 2016; Hendrycks et al., 2021).

One potential approach to address goal misgeneralization is to directly specify the desired goals using natural language instructions and train instruction-following agents to achieve text-specified goals (Ahn et al., 2022; Brohan et al., 2022; Hill et al., 2020; Lynch et al., 2022; Nair et al., 2021; Shridhar et al., 2023; Winograd, 1972). Recent studies indeed have shown that training text-conditioned policies upon large pre-trained multimodal models (Geng et al., 2022; Radford et al., 2021) enables agents to achieve unseen goals specified with text descriptions containing unseen semantic concepts such as objects and colors (Liu et al., 2022a; Shridhar et al., 2022). Yet training such agents require

[1] Anonymous Institution, Anonymous City, Anonymous Region, Anonymous Country. Correspondence to: Anonymous Author <anon.email@domain.com>.

Preliminary work. Under review by the Workshop on New Frontiers in Learning, Control, and Dynamical Systems at the International Conference on Machine Learning (ICML). Do not distribute.

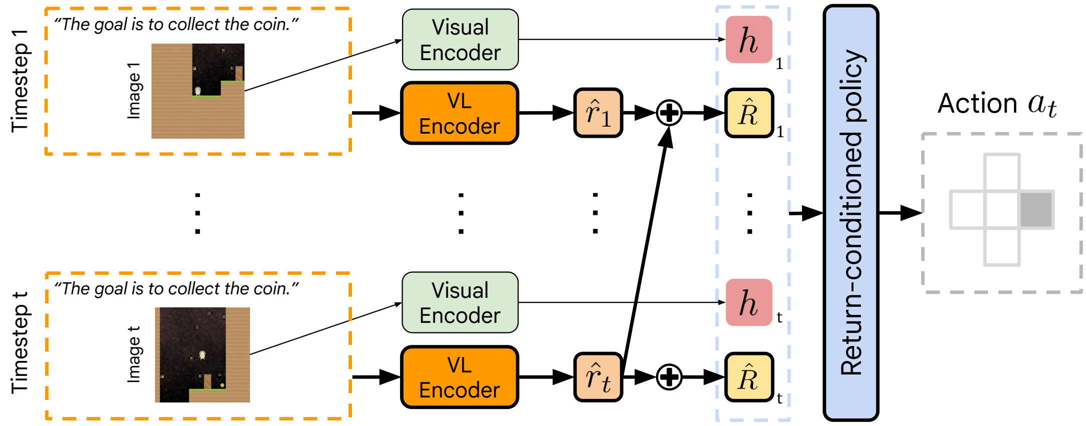


Figure 2: Illustration of our framework. First, multimodal reward $\hat{r}$ for each timestep is computed using visual-text similarity obtained from the pre-trained multimodal representations (denoted as VL Encoder). Then, we train transformer-based policy conditioned on the target return of multimodal rewards. MRDT then can be adapted to unseen situations by using the multimodal reward signal at deployment time. The key components of our contributions are highlighted in bold.

a dataset consisting of demonstrations from diverse tasks (*e.g.,* tasks with different coin positions) or with detailed text descriptions (*e.g.,* description of how much to jump or which obstacles to dodge.). This is problematic because such dataset may not be available for target domains and collecting diverse demonstrations could incur a huge cost. Motivated by the recent success of large vision-language models in various applications (Radford et al., 2021; Alayrac et al., 2022; Chen et al., 2022; Ramesh et al., 2021; 2022; OpenAI, 2023), we instead aim to develop a method that leverages the knowledge captured within such large models for improving the generalization to diverse goals.

In this paper, we present Multimodal Reward Decision Transformer (MRDT), a novel method for behavior learning that utilizes the image-text multimodal alignment score (Radford et al., 2021) as a reward to train a return-conditioned policy (see Figure 2). Our key idea is that the agent can learn to achieve desired goals by following the adaptive reward signal that adjusts based on the progress made towards achieving text-specified goals. Specifically, we propose training a return-conditioned policy using a transformer architecture (Chen et al., 2021; Vaswani et al., 2017) which predicts actions conditioned on a sequence of multimodal representations, previous actions, and multimodal returns. Unlike prior work that rely on static multimodal representations for behavior learning (Liu et al., 2022a), MRDT makes decisions based on the adaptive multimodal reward signal, leading to improved generalization to different goals in test environments. Furthermore, we introduce a fine-tuning scheme that adapts pre-trained multimodal encoders using in-domain data, enhancing the quality of the reward signal. Remarkably, our findings demonstrate that when using the reward from fine-tuned encoders of improved quality, the agent exhibits better generalization to test environments with previously unseen desired goals.

In summary, our key contributions are as follows:

- We propose a novel imitation learning framework that trains a return-conditioned policy which makes decisions based on the adaptive signal from image-text multimodal rewards at deployment time, which we call Multimodal Reward Decision Transformer (MRDT).
- We introduce a fine-tuning scheme that adapts pre-trained multimodal models using in-domain expert demonstrations, to improve the quality of multimodal rewards.
- We demonstrate that MRDT successfully guides the agent to achieve unseen goals specified with natural language instructions in various environments from OpenAI Procgen benchmark (Cobbe et al., 2020).
- We provide analysis on how MRDT helps addressing goal misgeneralization based on new metrics we introduced for evaluating the quality of multimodal rewards.

Overall, our framework shows promising results in terms of robustness and goal alignment, offering an effective approach and insights to address the challenges of goal misgeneralization for imitation learning in complex environments.

## 2. Related Work

**Generalization in behavior learning** Addressing the challenge of generalization in behavior learning is crucial for deploying trained agents in real-world scenarios. Various approaches have been proposed to enhance agent robustness to different types of variations. These include regularization techniques (Cobbe et al., 2019; Farebrother et al., 2018; Wang et al., 2020), data augmentation (Laskin et al., 2020a;b; Tobin et al., 2017; Kostrikov et al., 2020; Yarats et al., 2021; Lee et al., 2020; Hansen & Wang, 2021; Raileanu et al., 2021), contrastive learning (Agarwal et al., 2021; Mazoure et al., 2022), and domain randomization (Tobin et al., 2017; Peng et al., 2018; James et al., 2019). Recently, there has been growing interest in leveraging pre-trained representations for robot learning algorithms that benefit from large-scale data (Nair et al., 2022; Xiao et al., 2022; Seo et al., 2022; Ma et al., 2023). In particular, instruction-following agents have seen significant advancements by leveraging pre-trained vision-language models (Liu et al., 2022a; Shridhar et al., 2022; Jiang et al., 2022; Zeng et al., 2022), drawing inspiration from the effectiveness of multimodal representation learning techniques like CLIP (Radford et al., 2021). For example, *Instruct*RL (Liu et al., 2022a) utilizes a pre-trained multimodal encoder (Geng et al., 2022) to encode the alignment between multiple camera observations and text instructions, and leverages this representation for training a transformer-based policy.

**Language-conditioned behavior learning** Humans excel at understanding and utilizing language instructions to adapt to unfamiliar situations. Consequently, there has been significant interest in training policies that incorporate natural language in both RL (Goyal et al., 2021; Misra et al., 2017; Jiang et al., 2019) and IL (Lynch & Sermanet, 2020; Stepputtis et al., 2020; Jang et al., 2022). A related approach to ours is presented in Goyal et al. (2021), which maps language instructions to pixel observations using a reward scalar. Recent studies have also leveraged large language models (LLMs) for robotic manipulation tasks (Ahn et al., 2022; Liang et al., 2022; Huang et al., 2022a;b; Hill et al., 2020; Nair et al., 2021; Driess et al., 2023). For instance, SayCan (Ahn et al., 2022) leverages PaLM (Chowdhery et al., 2022) to generate plans for intermediate steps based on language instructions, then executes them by connecting each plan to the appropriate candidate among equipped skills. Our method can be thought as one of language-conditioned behavior cloning which leverages natural language instructions as the form of reward signal by utilizing the alignment between pre-trained multimodal representations.

**Goal misgeneralization** Goal misgeneralization has been the focus of several research work. Di Langosco et al. (2022) formalize goal misgeneralzation as distinct from capability misgeneralization in RL and provide empirical examples in various games from the Procgen benchmarks (Cobbe et al., 2020). Shah et al. (2022) broaden the definition of goal misgeneralization to arbitrary learning settings and demonstrate extensive examples under diverse conditions. In this paper, we alleviate the goal misgeneralization problem by harnessing the alignment among pre-trained multimodal visual and text representations.

**Intrinsic reward** Intrinsic reward in RL (Şimşek & Barto, 2006; Schmidhuber, 2010) was initially proposed to facilitate better exploration in early interactions with the environment. It has been formulated using various types of frameworks, such as using the errors from predictive models (Schmidhuber, 1991; Oudeyer et al., 2007; Stadie et al., 2015; Pathak et al., 2017; 2019; Sekar et al., 2020), visitation counts (Thrun, 1992; Bellemare et al., 2016; Tang et al., 2017; Ostrovski et al., 2017; Burda et al., 2019), and coverage of visited states (Lee et al., 2019; Hazan et al., 2019; Mutti et al., 2022a;b; Liu & Abbeel, 2021; Tao et al., 2020; Seo et al., 2021). Instead of designing an intrinsic bonus for exploration, we use the intrinsic reward from multimodal models as an adaptive signal that guides the agent to achieve the desired goals. Similar to our work, there have been approaches that use the text-image alignment score to solve sparse reward tasks with RL (Fan et al., 2022; Cui et al., 2022). In this work, we focus on the adaptability of the multimodal reward, which can guide the agent to achieve desired goals in unseen environments at test time.

## 3. Method

In this section, we present Multimodal Reward Decision Transformer (MRDT), a framework that uses the visual-text alignment score as a reward for guiding the agent to achieve desired goals. We first describe the problem setup (see Section 3.1), then introduce how we define our multimodal reward and use it for training a return-conditioned policy (see Section 3.2). We then introduce our fine-tuning scheme that adapts the pre-trained multimodal encoder with in-domain data to improve the quality of rewards (see Section 3.3). We provide the overview of MRDT in Figure 2.

### 3.1. Problem Setup

We formulate our control task as a Markov Decision Process (MDP) (Sutton & Barto, 2018) without an explicit reward function, which is defined as a tuple $(\mathcal{O}, \mathcal{A}, p, \mathcal{G}, \mathcal{X})$. $\mathcal{O}$ is the observation space, $\mathcal{A}$ is the action space, $p(o_t|o_{<t}, a_{<t})$ is the transition dynamics, $\mathcal{G} \subset \mathcal{O}$ is the goal space which is a set of assignments to the state, and $\mathcal{X}$ is the space of natural language. In this paper, we focus on the generalization setup where the goal used for collecting demonstrations and testing is sampled from different distributions $p_{\texttt{train}}(g)$ and $p_{\texttt{test}}(g)$, respectively. We assume that we have an access to $\mathcal{D} = \{\tau_i\}_{i=1}^N$ consisting of $N$ expert state-action trajectories $\tau = (o_0, a_0^*, ..., o_T, a_T^*)$ where $T$ denotes the maximum timestep. Policy $\pi(a_t|o_{\leq t}, \mathbf{x})$ outputs $a \in \mathcal{A}$,

conditioned on the history of observations $o_{\leq t}$ and a text instruction $\mathbf{x} \in \mathcal{X}$. The aim of our paper is to train a policy $\pi(a_t|o_{\leq t}, \mathbf{x})$ that can achieve expert performance in reaching the goal state $g$ sampled from both $p_{\texttt{train}}$ and $p_{\texttt{test}}$ using $\mathcal{D}$.

### 3.2. Multimodal Reward Decision Transformer

**Multimodal reward** To address the goal misgeneralization problem, we propose to utilize the visual-text alignment score computed using the large multimodal models as a reward. Our motivation is that the adaptive reward, which adjusts based on the progress made towards achieving text-specified goals, can guide the agent to achieve desired goals. Specifically, we use the distance between pre-trained multimodal representations as our multimodal reward:

$$\hat{r}(o_t, \mathbf{x}) = D(\phi(o_t), \psi(\mathbf{x})) \tag{1}$$

where $D$ is a distance metric in the representation space of multimodal models consisting of a visual encoder $\phi$ and a language encoder $\psi$. While our method is compatible with any multimodal model and metric, we adopt the image-text similarity in the representation space of CLIP (Radford et al., 2021) for $D$ and its visual and text encoders for $\phi$ and $\psi$, respectively. This is inspired by recent work that leverages the CLIP score for various applications (Hessel et al., 2021; Crowson et al., 2022; Kwon & Ye, 2022; Cho et al., 2022; Fan et al., 2022; Jeong et al., 2023).

**Return-conditioned policy** To train the agent that makes decisions based on the adaptive multimodal reward signal, we train a return-conditioned policy $\theta$ based on transformer architecture (Chen et al., 2021; Lee et al., 2022). Specifically, we train a decoder-only transformer to autoregressively model the following sequence:

$$\langle h_1, \hat{R}_1, a_1, h_2, \hat{R}_2, a_2, ..., h_T, \hat{R}_T, a_T \rangle$$

where $h_t = \phi(o_t)$ is a visual representation and $\hat{R}_t = \sum_{i=t}^{T} \hat{r}(o_i, \mathbf{x})$ is the target return computed at timestep $t$ using the multimodal reward. Because we compute the multimodal reward $\hat{r}_t$ at every timestep, the return-conditioned policy that models the trajectory is trained to output actions based on the adaptive multimodal reward signal, enabling adaptation at deployment time.

**Objective** Given the expert trajectory $\tau$, we first compute the target returns $\{\hat{R}_i^*\}_{i=1}^T$ of expert demonstrations by computing the multimodal reward in Equation 1. Following Lee et al. (2022), we train the model to predict not only the next action but also the next multimodal return by minimizing the objective below:

$$\mathcal{L}_{\text{MRDT}} = \mathbb{E}_{\tau \sim \mathcal{D}} \left[ \sum_{t \leq T} \text{CE}(a_t, a_t^*) + \lambda \cdot \text{MSE}(\hat{R}_t, \hat{R}_t^*) \right] \tag{2}$$

where CE is the cross entropy loss, MSE is the mean squared error, and $\lambda$ is a hyperparameter that adjusts the scale of return prediction. We find that predicting the multimodal target returns improves goal generalization by encouraging the agent to be more aware of the adaptive multimodal reward signals (see Table 1 for supporting experiments).

### 3.3. Fine-tuning the Pre-trained Multimodal Encoder

Despite the effectiveness of our method with pre-trained CLIP multimodal representations, there may be a domain gap between the images used for pre-training and the visual observations available from the environment. This domain gap can sometimes lead to the generation of unreliable, misleading reward signals. To address this issue, we propose fine-tuning schemes for the pre-trained multimodal encoders $(\phi, \psi)$ using the demonstration data $\mathcal{D}$ in order to improve the quality of multimodal rewards. Specifically, we propose fine-tuning objectives based on the following two desiderata: reward should (i) be temporally smooth and (ii) be robust to visual distractions that cannot affect the agent.

**Temporal smoothness** To encourage the temporal smoothness of the multimodal reward, we adopt the objective of value implicit pre-training (VIP) (Ma et al., 2023) that aims to learn smooth reward functions from action-free videos. The main idea of VIP is to (i) capture long-range dependency by attracting the representations of the first and goal frames and (ii) inject local smoothness by encouraging the distance between intermediate frames to represent a progress toward the goal. We extend this idea to our multimodal setup by replacing the goal frame with the natural language description $\mathbf{x}$ that describes the goal and using our multimodal reward $\hat{r}$ based on CLIP distance $D$ as below:

$$\begin{aligned} \mathcal{L}_{\text{VIP}} = \; & \underbrace{(1-\gamma) \cdot \mathbb{E}_{o_1 \sim \mathcal{O}_1} [\hat{r}(o_1, \mathbf{x})]}_{\text{long-range dependency loss}} \\ & + \underbrace{\log \mathbb{E}_{(o_t, o_{t+1}) \sim \mathcal{D}} [\hat{r}(o_t, \mathbf{x}) + 1 - \gamma \cdot \hat{r}(o_{t+1}, \mathbf{x})]}_{\text{local smoothness loss}} \end{aligned} \tag{3}$$

where $\mathcal{O}_1$ denotes a set of initial visual observations in $\mathcal{D}$. One can see that the local smoothness loss is the one-step temporal difference loss which recursively trains the $\hat{r}(o_t, \mathbf{x})$ to regress $-1 + \gamma \cdot \hat{r}(o_{t+1}, \mathbf{x})$. This then induces the reward to represent the remaining steps to achieve the text-specified goal $\mathbf{x}$ (Huang et al., 2019), making rewards from consecutive observations smooth.

**Robustness to visual distractions** To further encourage our multimodal reward to be robust to visual distractions that should not affect the agent (*e.g.,* changing textures or backgrounds), we introduce the inverse dynamics model (IDM) objective (Pathak et al., 2017; Christiano et al., 2016; Islam et al., 2022; Lamb et al., 2023; Tomar et al., 2023):

$$\mathcal{L}_{\text{IDM}} = \mathbb{E}_{(o_t, o_{t+1}, a_t)\sim\mathcal{D}}[\text{MSE}(g(\phi(o_t), \phi(o_{t+1}), \psi(\mathbf{x})), a_t)] \tag{4}$$

where $g(\cdot)$ denotes the prediction layer which outputs $\hat{a}_t$, predicted estimate of $a_t$. By learning to predict actions taken by the agent using the observations from consecutive timesteps, fine-tuned encoder learns to ignore aspects within the observations that should not affect the agent.

**Fine-tuning objective** We combine both the VIP loss and the IDM loss as the training objective to fine-tune the pre-trained multimodal encoder in our model:

$$\mathcal{L}_{\text{FT}} = \mathcal{L}_{\text{VIP}} + \beta \cdot \mathcal{L}_{\text{IDM}} \tag{5}$$

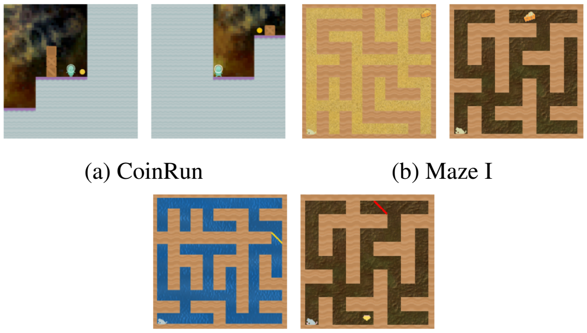
where $\beta$ is a scale hyperparameter. We find that both objectives synergistically contribute to improving the performance (see Table 1 for supporting experiments).

**Architecture** To effectively fine-tune the pre-trained CLIP embeddings without overfitting, we adopt the CLIP-Adapter (Gao et al., 2021; Zhang et al., 2022). Specifically, we attach extra linear layers to both the visual and language encoders and perform residual-style feature blending with the original pre-trained features. Throughout training, we only apply gradients to the weight of these adapter layers and freeze both the visual and textual encoders of CLIP. Furthermore, we utilize multi-scale features obtained by concatenating intermediate layer representations with the final output representation as the input for the adapter layers, drawing inspiration from Liu et al. (2022a) and Walmer et al. (2022). Finally, the multimodal reward is computed using the cosine similarity between the multi-scale features from the image and text encoders. See Appendix B for qualitative results of our multimodal rewards.

## 4. Experiments

In this section, we verify the effectiveness of our framework in generalization to different goals in test environments. We first present evaluation results in various environments addressing goal misgeneralization (see Section 4.1). We then analyze the effectiveness of the proposed multimodal rewards in test time with new metrics (see Section 4.2). We also conduct ablation studies to validate the effectiveness of our proposed components (see Section 4.3).

**Environments** We evaluate MRDT on three different environments proposed in Di Langosco et al. (2022). These environments are built upon the OpenAI Procgen benchmarks (Cobbe et al., 2020), which are widely used to assess the generalization capabilities of models in the face of visual changes. For training, we collect a number of expert demonstrations from 500 different levels that exhibit ample visual variations, and use them for training the agent. We then evaluate the zero-shot performance of the agents in test environments from different levels with unseen goals that differ from the goals used for training. Specifically, we consider following three environments for our experiments:



(a) CoinRun

(b) Maze I

(c) Maze II

Figure 3: Image observation of OpenAI Procgen benchmarks (Cobbe et al., 2020) used in our experiments. We train our agents using expert demonstrations collected in environments with multiple visual variations (left). We then perform evaluations on environments from unseen levels with unseen goals (right). See Section 4 for more details.

- CoinRun: The training dataset consists of expert demonstrations where the agent collects a coin that is consistently positioned on the far right of the map. In the held-out evaluation environment, the location of the coin is randomized (see Figure 3a). We use “The goal is to collect the coin.” as a natural language instruction $\mathbf{x}$.
- Maze I: The training dataset consists of expert demonstrations where the agent reaches a yellow cheese that is always located at the top right corner. In the held-out evaluation environment, the cheese is placed at a random position (see Figure 3b). We use “Navigate a maze to collect the yellow cheese.” as a natural language instruction $\mathbf{x}$.
- Maze II: The training dataset consists of expert demonstrations where the agent approaches a yellow diagonal line located at a random position. For evaluation, we consider a modified environment with two objects: a yellow gem and a red diagonal line, where the goal of the agent is to reach the diagonal line regardless of its color as in training environments (see Figure 3c). We use “Navigate a maze to collect the line.” as a natural language instruction $\mathbf{x}$.

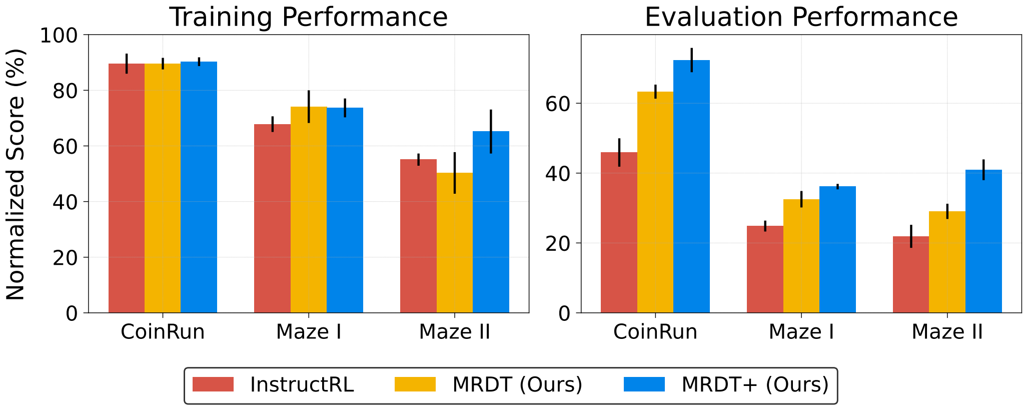


Figure 4: Expert-normalized scores on training/evaluation environments. The result shows the mean and standard variation averaged over three runs. MRDT denotes our method that uses frozen CLIP representations for computing the multimodal reward, and MRDT+ denotes the model that incorporates fine-tuning scheme in Section 3.3.

**Implementation** For all experiments, we utilize the open-sourced pre-trained CLIP model[1] with ViT-B/16 architecture to generate multimodal rewards and we fine-tune CLIP based on that model. We resize images to 224 × 224 × 3 before computing multimodal rewards. To fine-tune CLIP, we attach a 2-layer MLP (Rosenblatt, 1958) at the end of both CLIP image and text encoders and we also add an additional 2-layer MLP as an action prediction layer for the IDM objective. Our return-conditioned policy is implemented based on the official implementation of *Instruct*RL (Liu et al., 2022a) and implementation details are same unless otherwise specified. To collect expert demonstrations used for training data, we first train PPG (Cobbe et al., 2021) agents on 500 training levels for 200M timesteps per task. We then gather 500 rollouts for CoinRun and 1000 rollouts for Maze in training environments. All models are trained for 50 epochs on two GPUs with a batch size of 64 and a context length of 4. Further training details, including hyperparameter settings, can be found in Appendix A.

**Evaluation** To evaluate the performance of trained agents, we report the expert-normalized scores on both training and held-out evaluation environments. For reporting training performance, we measure the average success rate of trained agents over 100 rollouts in training environments, and divide it with the average success rate from the expert PPG agent used for collecting demonstrations. For evaluation performance, we train a separate expert PPG agent in held-out evaluation environments, and use the score from this agent for computing the expert-normalized scores.

**Baseline and our method** As a baseline, we consider *Instruct*RL (Liu et al., 2022a), which trains the instruction-following agent using static multimodal representations from M3AE (Geng et al., 2022). For our method, we use the same M3AE model to encode visual observations. We refer to the model that uses frozen CLIP representations for computing the multimodal reward as MRDT, and the model that incorporates fine-tuning scheme in Section 3.3 as MRDT+ in all our experiments.

### 4.1. Results on Procgen Environments

Figure 4 shows that our method significantly outperforms the baseline in all three tasks. In particular, MRDT outperforms *Instruct*RL on held-out evaluation environments even though the training performance is similar. This shows that our method can indeed guide the agent to achieve unseen goals with the adaptive multimodal reward signal. Moreover, we observe that MRDT+, which uses the multimodal reward from the fine-tuned model, achieves superior performance to MRDT. Considering that the only difference between MDRT and MRDT+ is using different multimodal rewards, this result shows that improving the quality of reward can lead to better generalization performance.

### 4.2. Evaluating Multimodal Rewards

To provide insights on how MRDT helps improving the generalization performance, this section introduces new metrics that evaluate the quality of multimodal rewards. Our metrics are designed to reflect the desiderata which multimodal rewards should satisfy for effectively guiding the agent: (i) they should consistently assign similar reward values to similar behaviors, even in the presence of different visual variations (*e.g.,* different backgrounds or colors) and (ii) they should effectively differentiate between goal-reaching behaviors and misleading behaviors. To illustrate this, consider the CoinRun environment as a simple example. When the agent successfully reaches a coin, it should receive a similar reward regardless of the specific map configuration.

[1] https://github.com/openai/CLIP

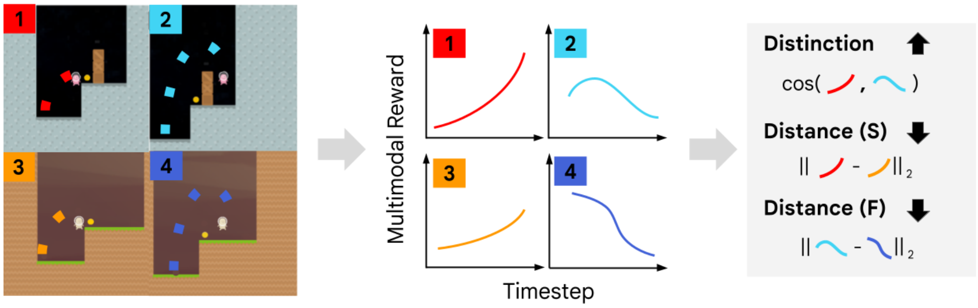


Figure 5: In this example, we have two successful trajectories (red, orange) and two failure trajectories (cyan, blue). We extract a $K$-length sequence of multimodal rewards from each trajectory, and compute three metrics: (1) **Distinction**, which measures the ability of multimodal rewards to differentiate between successful and failure behaviors; (2) **Distance (S)**, which measures reward similarity between successful behaviors; and (3) **Distance (F)**, which measures reward similarity between failure behaviors.

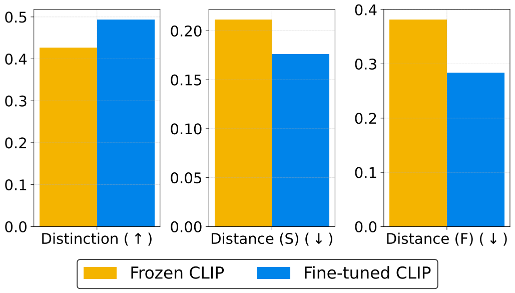


Figure 6: Quality measures of multimodal rewards computed with different representations. (↑) and (↓) imply that higher/lower values are better, respectively.

In contrast, multi-modal rewards should provide different signals to the agent according to whether it collects the coin or not (*i.e.,* achieving a goal or not).[2]

To this end, we first fix the configuration of the map and apply only visual changes according to a context vector $c$ in the environment. We collect a set of success/failure trajectories where each success/failure trajectory has the same sequence of actions and differs only in observations, respectively. We then extract a $K$-length subsequence of multimodal rewards from each trajectory as follows:

$$\tau^c_{\texttt{succ}} := (\hat{r}^c_{T_{\texttt{succ}}-K+1}, \hat{r}^c_{T_{\texttt{succ}}-K+2}, ..., \hat{r}^c_{T_{\texttt{succ}}})$$
$$\tau^c_{\texttt{fail}} := (\hat{r}^c_{T_{\texttt{fail}}-K+1}, \hat{r}^c_{T_{\texttt{fail}}-K+2}, ..., \hat{r}^c_{T_{\texttt{fail}}})$$

where $T_{\texttt{succ}}$ denotes the timestep when the agent succeeds at the task (by reaching a coin) in the success trajectory and $T_{\texttt{fail}}$ denotes the timestep when the agent fails the task (by skipping the coin) in the failed trajectory. Now, to quantify the quality of the multimodal rewards, we define three metrics:

1. **Distinction** (↑): We compute the expectation of the cosine distance between $\tau^c_{\texttt{succ}}$ and $\tau^c_{\texttt{fail}}$ over the context distribution: $1-\mathbb{E}_{c\sim C}[\tau^c_{\texttt{succ}} \cdot \tau^c_{\texttt{fail}} / (\|\tau^c_{\texttt{succ}}\| \cdot \|\tau^c_{\texttt{fail}}\|)]$. This metric measures the ability of multimodal rewards to differentiate between successful and failure behaviors.

2. **Distance between success trajectories (Distance (S))** (↓): We calculate the expected Euclidean distance between pairs of successful trajectories $\{\tau^c_{\texttt{succ}}\}_c$ under different contexts: $\mathbb{E}_{(c,c')\sim C}[\|\tau^c_{\texttt{succ}} - \tau^{c'}_{\texttt{succ}}\|_2]$. This metric evaluates whether the reward consistently assigns similar reward values to successful behaviors, regardless of different visual contexts.

3. **Distance between failed trajectories (Distance (F))** (↓): In a similar manner, we compute the expected Euclidean distance between pairs of failed trajectories $\{\tau^c_{\texttt{fail}}\}_c$ under different contexts: $\mathbb{E}_{(c,c')\sim C}[\|\tau^c_{\texttt{fail}} - \tau^{c'}_{\texttt{fail}}\|_2]$.

Note that (↑) / (↓) implies that higher/lower value is better, respectively. For evaluation, we first select 5 different levels in CoinRun evaluation environment where the position of the coin is randomized. Each level consists of a map with the same configuration, but the colors of the background and objects are different. We then collect success and failure trajectories from each level where the order of action is the same. We use $K = 10$ for generating subsequences.

**Comparison between frozen/fine-tuned CLIP** In Figure 6, we evaluate the quality of multimodal rewards from the CLIP model with and without fine-tuning. We find that the multimodal reward from the fine-tuned CLIP outperforms the baseline without fine-tuning in terms of all metrics,

[2] We do not prioritize the sign or relative magnitude of the multimodal reward because we consider the IL setup. We instead focus on whether the multimodal reward exhibits distinct patterns for different behaviors.

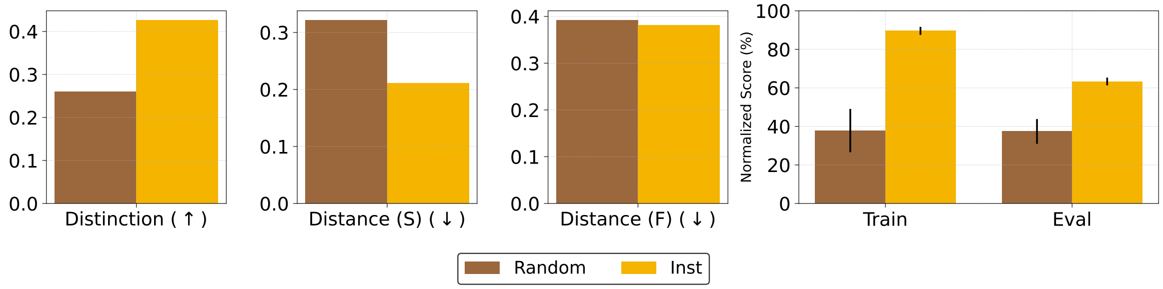


Figure 7: Quality measures of multimodal rewards generated with (i) instructive text (*i.e.,* Inst) and (ii) random text (*i.e.,* Random). (↑) and (↓) imply that higher/lower values are better, respectively.

Table 1: Ablation study of the return prediction loss $\text{MSE}(\hat{R}_t, \hat{R}_t^*)$ in CoinRun environments.

| $\text{MSE}(\hat{R}_t, \hat{R}_t^*)$ | $\mathcal{L}_{\text{FT}}$ | Train (%) | Eval (%) |
|---|---|---|---|
| ✗ | ✗ | 90.28% ± 4.21% | 56.32% ± 3.55% |
| | ✓ | 92.01% ± 3.18% | 56.28% ± 2.01% |
| ✓ | ✗ | 89.58% ± 2.08% | 63.32% ± 2.01% |
| | ✓ | 90.28% ± 1.59% | 72.36% ± 3.48% |

Table 2: Ablation study of the fine-tuning objectives: VIP Loss $\mathcal{L}_{\text{VIP}}$ and IDM Loss $\mathcal{L}_{\text{IDM}}$ in CoinRun environments.

| $\mathcal{L}_{\text{VIP}}$ | $\mathcal{L}_{\text{IDM}}$ | Train (%) | Eval (%) |
|---|---|---|---|
| ✗ | ✗ | 89.58% ± 2.08% | 63.32 % ± 2.01% |
| ✗ | ✓ | 89.24% ± 6.01% | 67.34 % ± 2.66% |
| ✓ | ✗ | 90.28% ± 2.17% | 70.35 % ± 1.01% |
| ✓ | ✓ | 90.28% ± 1.59% | 72.36 % ± 3.48% |

which shows that the proposed fine-tuning scheme effectively improves the quality of multimodal rewards. We also emphasize that this result is consistent with the results in Figure 4 where MRDT+ with the fine-tuned reward outperforms MRDT. This shows that our metrics are well-aligned with the generalization performance of return-conditioned policies, supporting the usefulness of our metrics for evaluating the quality of multimodal rewards.

**Comparison between different types of text instructions** To investigate whether MRDT makes decisions based on the adaptive signal from the multimodal reward, we evaluate the quality of rewards generated with (i) instructive text (*i.e.,* Inst) and (ii) random text (*i.e.,* Random). Specifically, we use “The goal is to collect the coin.” for Inst and “NeurIPS 2023 will be held again at the New Orleans Ernest N. Morial Convention Center.” for Random. As shown in Figure 7, we observe that using the instructive text leads to multimodal rewards of better quality compared to using the random text. Moreover, we find that using the random text instruction significantly degrades the performance in both training and held-out evaluation environments (the rightmost one in Figure 7). These results highlight the importance of using the instructive text and demonstrate that MRDT indeed depends on the adaptive signal from the multimodal reward for acheiving goals at deployment time.

### 4.3. Ablation Studies

**Effect of return prediction** We investigate the effect of including the return prediction loss $\text{MSE}(\hat{R}_t, \hat{R}_t^*)$ in Equation 2, which encourages the policy to be more aware of conditioned returns. In Table 1, we observe that the performance of MRDT becomes much more sensitive to the quality of multimodal rewards when trained with the return prediction loss. For instance, without the return prediction loss, the evaluation performance becomes almost the same with or without the fine-tuning scheme, which suggests that model is insensitive to the quality of rewards. On the other hand, with the prediction loss, the performance increases as the quality of reward improves. This implies that the model gets to become aware of the returns and is thus able to follow the adaptive signal from the multimodal reward.

**Effect of fine-tuning objectives** In Table 2, we examine the effect of fine-tuning objectives by reporting the performance of our method with or without the VIP loss $\mathcal{L}_{\text{VIP}}$ (Equation 3) and the IDM loss $\mathcal{L}_{\text{IDM}}$ (Equation 4). We find that the performance improves with either $\mathcal{L}_{\text{VIP}}$ or $\mathcal{L}_{\text{IDM}}$, which shows the effectiveness of the proposed losses that encourages temporal smoothness and robustness to visual distractions. We also note that the performance with both objectives is the best, which implies that both losses synergistically contribute to improving the quality of the rewards.

## 5. Conclusion

In this paper, we have presented Multimodal Reward Decision Transformer, an imitation learning framework that guides the agent to achieve desired goals by using the adaptive signal from vision-language multimodal reward. Our experiments demonstrate that leveraging multimodal rewards, which represent how the current observation is close to achieving the text-specified goals, enables the agent to achieve even unseen goals at deployment time.

## References

Agarwal, R., Machado, M. C., Castro, P. S., and Bellemare, M. G. Contrastive behavioral similarity embeddings for generalization in reinforcement learning. *arXiv preprint arXiv:2101.05265*, 2021.

Ahn, M., Brohan, A., Brown, N., Chebotar, Y., Cortes, O., David, B., Finn, C., Gopalakrishnan, K., Hausman, K., Herzog, A., et al. Do as i can, not as i say: Grounding language in robotic affordances. *arXiv preprint arXiv:2204.01691*, 2022.

Akkaya, I., Andrychowicz, M., Chociej, M., Litwin, M., McGrew, B., Petron, A., Paino, A., Plappert, M., Powell, G., Ribas, R., et al. Solving rubik's cube with a robot hand. *arXiv preprint arXiv:1910.07113*, 2019.

Alayrac, J.-B., Donahue, J., Luc, P., Miech, A., Barr, I., Hasson, Y., Lenc, K., Mensch, A., Millican, K., Reynolds, M., et al. Flamingo: a visual language model for few-shot learning. *Advances in Neural Information Processing Systems*, 35:23716–23736, 2022.

Amodei, D., Olah, C., Steinhardt, J., Christiano, P., Schulman, J., and Mané, D. Concrete problems in ai safety. *arXiv preprint arXiv:1606.06565*, 2016.

Bacchus, F., Boutilier, C., and Grove, A. Rewarding behaviors. In *Proceedings of the National Conference on Artificial Intelligence*, pp. 1160–1167, 1996.

Bacchus, F., Boutilier, C., and Grove, A. Structured solution methods for non-markovian decision processes. In *AAAI/IAAI*, pp. 112–117. Citeseer, 1997.

Bellemare, M., Srinivasan, S., Ostrovski, G., Schaul, T., Saxton, D., and Munos, R. Unifying count-based exploration and intrinsic motivation. In *Advances in Neural Information Processing Systems*, 2016.

Brohan, A., Brown, N., Carbajal, J., Chebotar, Y., Dabis, J., Finn, C., Gopalakrishnan, K., Hausman, K., Herzog, A., Hsu, J., et al. Rt-1: Robotics transformer for real-world control at scale. *arXiv preprint arXiv:2212.06817*, 2022.

Burda, Y., Edwards, H., Storkey, A., and Klimov, O. Exploration by random network distillation. In *International Conference on Learning Representations*, 2019.

Chen, L., Lu, K., Rajeswaran, A., Lee, K., Grover, A., Laskin, M., Abbeel, P., Srinivas, A., and Mordatch, I. Decision transformer: Reinforcement learning via sequence modeling. *Advances in neural information processing systems*, 34:15084–15097, 2021.

Chen, X., Wang, X., Changpinyo, S., Piergiovanni, A., Padlewski, P., Salz, D., Goodman, S., Grycner, A., Mustafa, B., Beyer, L., et al. Pali: A jointly-scaled multilingual language-image model. *arXiv preprint arXiv:2209.06794*, 2022.

Cho, J., Yoon, S., Kale, A., Dernoncourt, F., Bui, T., and Bansal, M. Fine-grained image captioning with CLIP reward. In *Findings of the Association for Computational Linguistics: NAACL 2022*, pp. 517–527, Seattle, United States, July 2022. Association for Computational Linguistics. doi: 10.18653/v1/2022.findings-naacl.39. URL https://aclanthology.org/2022.findings-naacl.39.

Chowdhery, A., Narang, S., Devlin, J., Bosma, M., Mishra, G., Roberts, A., Barham, P., Chung, H. W., Sutton, C., Gehrmann, S., et al. Palm: Scaling language modeling with pathways. *arXiv preprint arXiv:2204.02311*, 2022.

Christiano, P., Shah, Z., Mordatch, I., Schneider, J., Blackwell, T., Tobin, J., Abbeel, P., and Zaremba, W. Transfer from simulation to real world through learning deep inverse dynamics model. *arXiv preprint arXiv:1610.03518*, 2016.

Cobbe, K., Klimov, O., Hesse, C., Kim, T., and Schulman, J. Quantifying generalization in reinforcement learning. In *International Conference on Machine Learning*, pp. 1282–1289. PMLR, 2019.

Cobbe, K., Hesse, C., Hilton, J., and Schulman, J. Leveraging procedural generation to benchmark reinforcement learning. In *International conference on machine learning*, pp. 2048–2056. PMLR, 2020.

Cobbe, K. W., Hilton, J., Klimov, O., and Schulman, J. Phasic policy gradient. In *International Conference on Machine Learning*, pp. 2020–2027. PMLR, 2021.

Crowson, K., Biderman, S., Kornis, D., Stander, D., Hallahan, E., Castricato, L., and Raff, E. Vqgan-clip: Open domain image generation and editing with natural language guidance. In *Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part XXXVII*, pp. 88–105. Springer, 2022.

Cui, Y., Niekum, S., Gupta, A., Kumar, V., and Rajeswaran, A. Can foundation models perform zero-shot task specification for robot manipulation? In *Learning for Dynamics and Control Conference*, 2022.

de Haan, P., Jayaraman, D., and Levine, S. Causal confusion in imitation learning. In *Advances in Neural Information Processing Systems*, 2019.

Di Langosco, L. L., Koch, J., Sharkey, L. D., Pfau, J., and Krueger, D. Goal misgeneralization in deep reinforcement learning. In *International Conference on Machine Learning*, pp. 12004–12019. PMLR, 2022.

Driess, D., Xia, F., Sajjadi, M. S., Lynch, C., Chowdhery, A., Ichter, B., Wahid, A., Tompson, J., Vuong, Q., Yu, T., et al. Palm-e: An embodied multimodal language model. *arXiv preprint arXiv:2303.03378*, 2023.

Fan, L., Wang, G., Jiang, Y., Mandlekar, A., Yang, Y., Zhu, H., Tang, A., Huang, D.-A., Zhu, Y., and Anandkumar, A. Minedojo: Building open-ended embodied agents with internet-scale knowledge. *arXiv preprint arXiv:2206.08853*, 2022.

Farebrother, J., Machado, M. C., and Bowling, M. Generalization and regularization in dqn. *arXiv preprint arXiv:1810.00123*, 2018.

Gao, P., Geng, S., Zhang, R., Ma, T., Fang, R., Zhang, Y., Li, H., and Qiao, Y. Clip-adapter: Better vision-language models with feature adapters. *arXiv preprint arXiv:2110.04544*, 2021.

Geng, X., Liu, H., Lee, L., Schuurams, D., Levine, S., and Abbeel, P. Multimodal masked autoencoders learn transferable representations. *arXiv preprint arXiv:2205.14204*, 2022.

Goyal, P., Niekum, S., and Mooney, R. Pixl2r: Guiding reinforcement learning using natural language by mapping pixels to rewards. In *Conference on Robot Learning*, pp. 485–497. PMLR, 2021.

Hansen, N. and Wang, X. Generalization in reinforcement learning by soft data augmentation. In *2021 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 13611–13617. IEEE, 2021.

Hazan, E., Kakade, S., Singh, K., and Van Soest, A. Provably efficient maximum entropy exploration. In *International Conference on Machine Learning*, 2019.

Hendrycks, D., Carlini, N., Schulman, J., and Steinhardt, J. Unsolved problems in ml safety. *arXiv preprint arXiv:2109.13916*, 2021.

Hessel, J., Holtzman, A., Forbes, M., Bras, R. L., and Choi, Y. Clipscore: A reference-free evaluation metric for image captioning. *arXiv preprint arXiv:2104.08718*, 2021.

Hill, F., Mokra, S., Wong, N., and Harley, T. Human instruction-following with deep reinforcement learning via transfer-learning from text. *arXiv preprint arXiv:2005.09382*, 2020.

Huang, W., Abbeel, P., Pathak, D., and Mordatch, I. Language models as zero-shot planners: Extracting actionable knowledge for embodied agents. *arXiv preprint arXiv:2201.07207*, 2022a.

Huang, W., Xia, F., Xiao, T., Chan, H., Liang, J., Florence, P., Zeng, A., Tompson, J., Mordatch, I., Chebotar, Y., Sermanet, P., Jackson, T., Brown, N., Luu, L., Levine, S., Hausman, K., and brian ichter. Inner monologue: Embodied reasoning through planning with language models. In *6th Annual Conference on Robot Learning*, 2022b. URL https://openreview.net/forum?id=3R3Pz5i0tye.

Huang, Z., Liu, F., and Su, H. Mapping state space using landmarks for universal goal reaching. *Advances in Neural Information Processing Systems*, 2019.

Islam, R., Tomar, M., Lamb, A., Efroni, Y., Zang, H., Didolkar, A., Misra, D., Li, X., van Seijen, H., Combes, R. T. d., et al. Agent-controller representations: Principled offline rl with rich exogenous information. *arXiv preprint arXiv:2211.00164*, 2022.

James, S., Wohlhart, P., Kalakrishnan, M., Kalashnikov, D., Irpan, A., Ibarz, J., Levine, S., Hadsell, R., and Bousmalis, K. Sim-to-real via sim-to-sim: Data-efficient robotic grasping via randomized-to-canonical adaptation networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 12627–12637, 2019.

Jang, E., Irpan, A., Khansari, M., Kappler, D., Ebert, F., Lynch, C., Levine, S., and Finn, C. Bc-z: Zero-shot task generalization with robotic imitation learning. In *Conference on Robot Learning*, pp. 991–1002. PMLR, 2022.

Jeong, J., Zou, Y., Kim, T., Zhang, D., Ravichandran, A., and Dabeer, O. WinCLIP: Zero-/few-shot anomaly classification and segmentation. *arXiv preprint arXiv:2303.14814*, 2023.

Jiang, Y., Gu, S. S., Murphy, K. P., and Finn, C. Language as an abstraction for hierarchical deep reinforcement learning. *Advances in Neural Information Processing Systems*, 32, 2019.

Jiang, Y., Gupta, A., Zhang, Z., Wang, G., Dou, Y., Chen, Y., Fei-Fei, L., Anandkumar, A., Zhu, Y., and Fan, L. Vima: General robot manipulation with multimodal prompts. *arXiv preprint arXiv:2210.03094*, 2022.

Kim, C., Park, J., Shin, J., Lee, H., Abbeel, P., and Lee, K. Preference transformer: Modeling human preferences using transformers for RL. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=Peot1SFDX0.

Kirk, R., Zhang, A., Grefenstette, E., and Rocktäschel, T. A survey of generalisation in deep reinforcement learning. *arXiv preprint arXiv:2111.09794*, 2021.

Kostrikov, I., Yarats, D., and Fergus, R. Image augmentation is all you need: Regularizing deep reinforcement learning from pixels. *arXiv preprint arXiv:2004.13649*, 2020.

Kwon, G. and Ye, J. C. Clipstyler: Image style transfer with a single text condition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 18062–18071, 2022.

Lamb, A., Islam, R., Efroni, Y., Didolkar, A. R., Misra, D., Foster, D. J., Molu, L. P., Chari, R., Krishnamurthy, A., and Langford, J. Guaranteed discovery of control-endogenous latent states with multi-step inverse models. *Transactions on Machine Learning Research*, 2023. ISSN 2835-8856. URL https://openreview.net/forum?id=TNocbXm5MZ.

Laskin, M., Lee, K., Stooke, A., Pinto, L., Abbeel, P., and Srinivas, A. Reinforcement learning with augmented data. *Advances in neural information processing systems*, 33: 19884–19895, 2020a.

Laskin, M., Srinivas, A., and Abbeel, P. Curl: Contrastive unsupervised representations for reinforcement learning. In *International Conference on Machine Learning*, pp. 5639–5650. PMLR, 2020b.

Lee, K., Lee, K., Shin, J., and Lee, H. Network randomization: A simple technique for generalization in deep reinforcement learning. In *International Conference on Learning Representations*, 2020. URL https://openreview.net/forum?id=HJgcvJBFvB.

Lee, K.-H., Nachum, O., Yang, M. S., Lee, L., Freeman, D., Guadarrama, S., Fischer, I., Xu, W., Jang, E., Michalewski, H., et al. Multi-game decision transformers. *Advances in Neural Information Processing Systems*, 35: 27921–27936, 2022.

Lee, L., Eysenbach, B., Parisotto, E., Xing, E., Levine, S., and Salakhutdinov, R. Efficient exploration via state marginal matching. *arXiv preprint arXiv:1906.05274*, 2019.

Liang, J., Huang, W., Xia, F., Xu, P., Hausman, K., brian ichter, Florence, P., and Zeng, A. Code as policies: Language model programs for embodied control. In *Workshop on Language and Robotics at CoRL 2022*, 2022. URL https://openreview.net/forum?id=fmtvpopfLC6.

Liu, H. and Abbeel, P. Behavior from the void: Unsupervised active pre-training. In *Advances in Neural Information Processing Systems*, 2021.

Liu, H., Lee, L., Lee, K., and Abbeel, P. Instruction-following agents with jointly pre-trained vision-language models. *arXiv preprint arXiv:2210.13431*, 2022a.

Liu, Y., Xiong, P., Xu, L., Cao, S., and Jin, Q. Ts2-net: Token shift and selection transformer for text-video retrieval. In *Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part XIV*, pp. 319–335. Springer, 2022b.

Loshchilov, I. and Hutter, F. Decoupled weight decay regularization. In *International Conference on Learning Representations*, 2019. URL https://openreview.net/forum?id=Bkg6RiCqY7.

Luo, H., Ji, L., Zhong, M., Chen, Y., Lei, W., Duan, N., and Li, T. Clip4clip: An empirical study of clip for end to end video clip retrieval and captioning. *Neurocomputing*, 508:293–304, 2022.

Lynch, C. and Sermanet, P. Grounding language in play. *arXiv preprint arXiv:2005.07648*, 40:105, 2020.

Lynch, C., Wahid, A., Tompson, J., Ding, T., Betker, J., Baruch, R., Armstrong, T., and Florence, P. Interactive language: Talking to robots in real time. *arXiv preprint arXiv:2210.06407*, 2022.

Ma, Y. J., Sodhani, S., Jayaraman, D., Bastani, O., Kumar, V., and Zhang, A. VIP: Towards universal visual reward and representation via value-implicit pre-training. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=YJ7o2wetJ2.

Mazoure, B., Kostrikov, I., Nachum, O., and Tompson, J. Improving zero-shot generalization in offline reinforcement learning using generalized similarity functions. In Oh, A. H., Agarwal, A., Belgrave, D., and Cho, K. (eds.), *Advances in Neural Information Processing Systems*, 2022. URL https://openreview.net/forum?id=Ls0yzIkEk1.

Misra, D., Langford, J., and Artzi, Y. Mapping instructions and visual observations to actions with reinforcement learning. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pp. 1004–1015, Copenhagen, Denmark, September 2017. Association for Computational Linguistics. doi: 10.18653/v1/D17-1106. URL https://aclanthology.org/D17-1106.

Mnih, V., Kavukcuoglu, K., Silver, D., Rusu, A. A., Veness, J., Bellemare, M. G., Graves, A., Riedmiller, M., Fidjeland, A. K., Ostrovski, G., et al. Human-level control through deep reinforcement learning. *nature*, 2015.

Mutti, M., De Santi, R., and Restelli, M. The importance of non-markovianity in maximum state entropy exploration. In *International Conference on Machine Learning*, 2022a.

Mutti, M., Mancassola, M., and Restelli, M. Unsupervised reinforcement learning in multiple environments. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 2022b.

Nair, S., Mitchell, E., Chen, K., brian ichter, Savarese, S., and Finn, C. Learning language-conditioned robot behavior from offline data and crowd-sourced annotation. In *5th Annual Conference on Robot Learning*, 2021. URL https://openreview.net/forum?id=tfLu5W6SW5J.

Nair, S., Rajeswaran, A., Kumar, V., Finn, C., and Gupta, A. R3m: A universal visual representation for robot manipulation. *arXiv preprint arXiv:2203.12601*, 2022.

Ngo, R. The alignment problem from a deep learning perspective. *arXiv preprint arXiv:2209.00626*, 2022.

OpenAI. Gpt-4 technical report, 2023.

Ostrovski, G., Bellemare, M. G., Oord, A. v. d., and Munos, R. Count-based exploration with neural density models. In *International Conference on Machine Learning*, 2017.

Oudeyer, P.-Y., Kaplan, F., and Hafner, V. V. Intrinsic motivation systems for autonomous mental development. *IEEE transactions on evolutionary computation*, 2007.

Pathak, D., Agrawal, P., Efros, A. A., and Darrell, T. Curiosity-driven exploration by self-supervised prediction. In *International Conference on Machine Learning*, 2017.

Pathak, D., Gandhi, D., and Gupta, A. Self-supervised exploration via disagreement. In *International Conference on Machine Learning*, 2019.

Peng, X. B., Andrychowicz, M., Zaremba, W., and Abbeel, P. Sim-to-real transfer of robotic control with dynamics randomization. In *2018 IEEE international conference on robotics and automation (ICRA)*, pp. 3803–3810. IEEE, 2018.

Radford, A., Kim, J. W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., et al. Learning transferable visual models from natural language supervision. In *International Conference on Machine Learning*, pp. 8748–8763. PMLR, 2021.

Raileanu, R., Goldstein, M., Yarats, D., Kostrikov, I., and Fergus, R. Automatic data augmentation for generalization in reinforcement learning. *Advances in Neural Information Processing Systems*, 34:5402–5415, 2021.

Ramesh, A., Pavlov, M., Goh, G., Gray, S., Voss, C., Radford, A., Chen, M., and Sutskever, I. Zero-shot text-to-image generation. In *International Conference on Machine Learning*, pp. 8821–8831. PMLR, 2021.

Ramesh, A., Dhariwal, P., Nichol, A., Chu, C., and Chen, M. Hierarchical text-conditional image generation with clip latents. *arXiv preprint arXiv:2204.06125*, 2022.

Rasheed, H., khattak, M. U., Maaz, M., Khan, S., and Khan, F. S. Finetuned clip models are efficient video learners. In *The IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023.

Rosenblatt, F. The perceptron: a probabilistic model for information storage and organization in the brain. *Psychological review*, 65(6):386, 1958.

Schmidhuber, J. A possibility for implementing curiosity and boredom in model-building neural controllers. In *Proc. of the international conference on simulation of adaptive behavior: From animals to animats*, pp. 222–227, 1991.

Schmidhuber, J. Formal theory of creativity, fun, and intrinsic motivation (1990–2010). *IEEE transactions on autonomous mental development*, 2(3):230–247, 2010.

Schrittwieser, J., Antonoglou, I., Hubert, T., Simonyan, K., Sifre, L., Schmitt, S., Guez, A., Lockhart, E., Hassabis, D., Graepel, T., et al. Mastering atari, go, chess and shogi by planning with a learned model. *Nature*, 2020.

Sekar, R., Rybkin, O., Daniilidis, K., Abbeel, P., Hafner, D., and Pathak, D. Planning to explore via self-supervised world models. In *International Conference on Machine Learning*, pp. 8583–8592. PMLR, 2020.

Seo, Y., Chen, L., Shin, J., Lee, H., Abbeel, P., and Lee, K. State entropy maximization with random encoders for efficient exploration. In *International Conference on Machine Learning*, pp. 9443–9454. PMLR, 2021.

Seo, Y., Lee, K., James, S. L., and Abbeel, P. Reinforcement learning with action-free pre-training from videos. In *International Conference on Machine Learning*, pp. 19561–19579. PMLR, 2022.

Shah, D., Osiński, B., Levine, S., et al. Lm-nav: Robotic navigation with large pre-trained models of language, vision, and action. In *Conference on Robot Learning*, pp. 492–504. PMLR, 2023.

Shah, R., Varma, V., Kumar, R., Phuong, M., Krakovna, V., Uesato, J., and Kenton, Z. Goal misgeneralization: Why correct specifications aren't enough for correct goals. *arXiv preprint arXiv:2210.01790*, 2022.

Shridhar, M., Manuelli, L., and Fox, D. Cliport: What and where pathways for robotic manipulation. In *Conference on Robot Learning*, pp. 894–906. PMLR, 2022.

Shridhar, M., Manuelli, L., and Fox, D. Perceiver-actor: A multi-task transformer for robotic manipulation. In *Conference on Robot Learning*, pp. 785–799. PMLR, 2023.

Şimşek, Ö. and Barto, A. G. An intrinsic reward mechanism for efficient exploration. In *Proceedings of the 23rd international conference on Machine learning*, pp. 833–840, 2006.

Song, X., Jiang, Y., Tu, S., Du, Y., and Neyshabur, B. Observational overfitting in reinforcement learning. In *International Conference on Learning Representations*, 2020.

Stadie, B. C., Levine, S., and Abbeel, P. Incentivizing exploration in reinforcement learning with deep predictive models. *arXiv preprint arXiv:1507.00814*, 2015.

Stepputtis, S., Campbell, J., Phielipp, M., Lee, S., Baral, C., and Ben Amor, H. Language-conditioned imitation learning for robot manipulation tasks. *Advances in Neural Information Processing Systems*, 33:13139–13150, 2020.

Sutton, R. S. and Barto, A. G. *Reinforcement learning: An introduction*. MIT press, 2018.

Tang, H., Houthooft, R., Foote, D., Stooke, A., Chen, O. X., Duan, Y., Schulman, J., DeTurck, F., and Abbeel, P. # exploration: A study of count-based exploration for deep reinforcement learning. In *Advances in Neural Information Processing Systems*, 2017.

Tao, R. Y., François-Lavet, V., and Pineau, J. Novelty search in representational space for sample efficient exploration. In *Advances in Neural Information Processing Systems*, 2020.

Thrun, S. Efficient exploration in reinforcement learning. *Technical Report. Carnegie Mellon University*, 1992.

Tobin, J., Fong, R., Ray, A., Schneider, J., Zaremba, W., and Abbeel, P. Domain randomization for transferring deep neural networks from simulation to the real world. In *2017 IEEE/RSJ international conference on intelligent robots and systems (IROS)*, pp. 23–30. IEEE, 2017.

Tomar, M., Islam, R., Levine, S., and Bachman, P. Ignorance is bliss: Robust control via information gating. *arXiv preprint arXiv:2303.06121*, 2023.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., and Polosukhin, I. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

Vinyals, O., Babuschkin, I., Czarnecki, W. M., Mathieu, M., Dudzik, A., Chung, J., Choi, D. H., Powell, R., Ewalds, T., Georgiev, P., et al. Grandmaster level in starcraft ii using multi-agent reinforcement learning. *Nature*, 2019.

Walmer, M., Suri, S., Gupta, K., and Shrivastava, A. Teaching matters: Investigating the role of supervision in vision transformers. *arXiv preprint arXiv:2212.03862*, 2022.

Wang, K., Kang, B., Shao, J., and Feng, J. Improving generalization in reinforcement learning with mixture regularization. *Advances in Neural Information Processing Systems*, 33:7968–7978, 2020.

Wang, Y., Li, K., Li, Y., He, Y., Huang, B., Zhao, Z., Zhang, H., Xu, J., Liu, Y., Wang, Z., et al. Internvideo: General video foundation models via generative and discriminative learning. *arXiv preprint arXiv:2212.03191*, 2022.

Winograd, T. Understanding natural language. *Cognitive psychology*, 3(1):1–191, 1972.

Xiao, T., Radosavovic, I., Darrell, T., and Malik, J. Masked visual pre-training for motor control. *arXiv preprint arXiv:2203.06173*, 2022.

Yarats, D., Fergus, R., Lazaric, A., and Pinto, L. Mastering visual continuous control: Improved data-augmented reinforcement learning. *arXiv preprint arXiv:2107.09645*, 2021.

Zeng, A., Wong, A., Welker, S., Choromanski, K., Tombari, F., Purohit, A., Ryoo, M., Sindhwani, V., Lee, J., Vanhoucke, V., et al. Socratic models: Composing zero-shot multimodal reasoning with language. *arXiv preprint arXiv:2204.00598*, 2022.

Zhang, C., Vinyals, O., Munos, R., and Bengio, S. A study on overfitting in deep reinforcement learning. *arXiv preprint arXiv:1804.06893*, 2018.

Zhang, R., Guo, Z., Zhang, W., Li, K., Miao, X., Cui, B., Qiao, Y., Gao, P., and Li, H. Pointclip: Point cloud understanding by clip. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 8552–8562, 2022.

# Supplementary Material

## Guide Your Agent with Adaptive Multimodal Rewards

## A. Experiment Details

In this section, we describe the details for implementing Multimodal Reward Decision Transformer.

**Procgen details** We utilize a publicly available implementation[3] to replicate the environments introduced by Di Langosco et al. (2022). We modify the simulator of the environments to render higher-resolution images to leverage pre-trained multimodal representations for both our method and baselines. In this particular setup, the observations obtained from the environment at each timestep $t$ comprise an RGB image with dimensions of $256 \times 256 \times 3$ and a natural language instruction that delineates the desired goal. Throughout our experiments, we adhere to the *hard environment difficulty* as described in (Cobbe et al., 2020). Maximum episode length for all tasks is 500. To gather expert demonstrations used for training data, we train PPG (Cobbe et al., 2021) agents on 500 training levels for 200M timesteps per task using hyperparameters provided in Cobbe et al. (2021). For evaluation purposes, we assess the test performance on 1000 different levels, encompassing previously unseen themes and goals that differ from those employed in training.

**Architecture details** Both *Instruct*RL (Liu et al., 2022) and MRDT employ ViT-B/16 as the transformer-policy and pre-trained multimodal transformer encoder (M3AE; (Geng et al., 2022)) in all experiments, unless stated otherwise. Inspired by Gao et al. (2021), we attach an additional 2-layer MLP to the end of a pre-trained multimodal transformer encoder and perform residual-style feature blending with the pre-trained features. In the training phase, we apply gradients only to the weight of these linear layers. Through empirical evaluation, we observe that this architecture yields superior performance in both our method and the baseline.

**Training details** We use $256 \times 256 \times 3$ RGB observations for training the return-conditioned policy. To stabilize training, we normalize multimodal returns following the method proposed by Chen et al. (2021), dividing them by 1000 in all experiments. We use the AdamW optimizer (Loshchilov et al., 2018) with a learning rate of $5 \times 10^{-4}$ and weight decay $5 \times 10^{-5}$. A cosine decay schedule is utilized to adjust the training learning rate. In CoinRun experiments, data augmentation techniques such as color jitter and random rotation are applied to the RGB images $o_t$ while maintaining alignment in the context. However, no augmentation is applied to RGB images in Maze I/II experiments. For scaling the return prediction loss in training the return-conditioned policy, we set $\lambda = 0.01$ in CoinRun experiments and $\lambda = 0.001$ in Maze I/II experiments. During the fine-tuning of the pre-trained multimodal encoder, a 2-layer MLP is attached to the end of both CLIP image and text encoders. Additionally, an extra 2-layer MLP is added as an action prediction layer for the IDM objective. The model is trained for 20 epochs, and the one with the lowest validation loss is used for generating multimodal rewards. To scale the IDM loss in fine-tuning CLIP, we employ $\beta = 1.5$ in CoinRun experiments and $\beta = 2.0$ in Maze I/II experiments.

**Computation** We use 24 CPU cores (Intel Xeon CPU @ 2.2GHz) and 2 GPUs (NVIDIA A100 40GB GPU) for training return-conditioned policy. The training of MRDT for 50 epochs takes approximately 4 hours for CoinRun experiments with the largest dataset size. For fine-tuning CLIP, we use 24 CPU cores (Intel Xeon CPU @ 2.2GHz) and 1 GPU (NVIDIA A100 40GB GPU), and it takes approximately 1.5 hours for Coinrun experiments.

[3] https://github.com/JacobPfau/procgenAISC

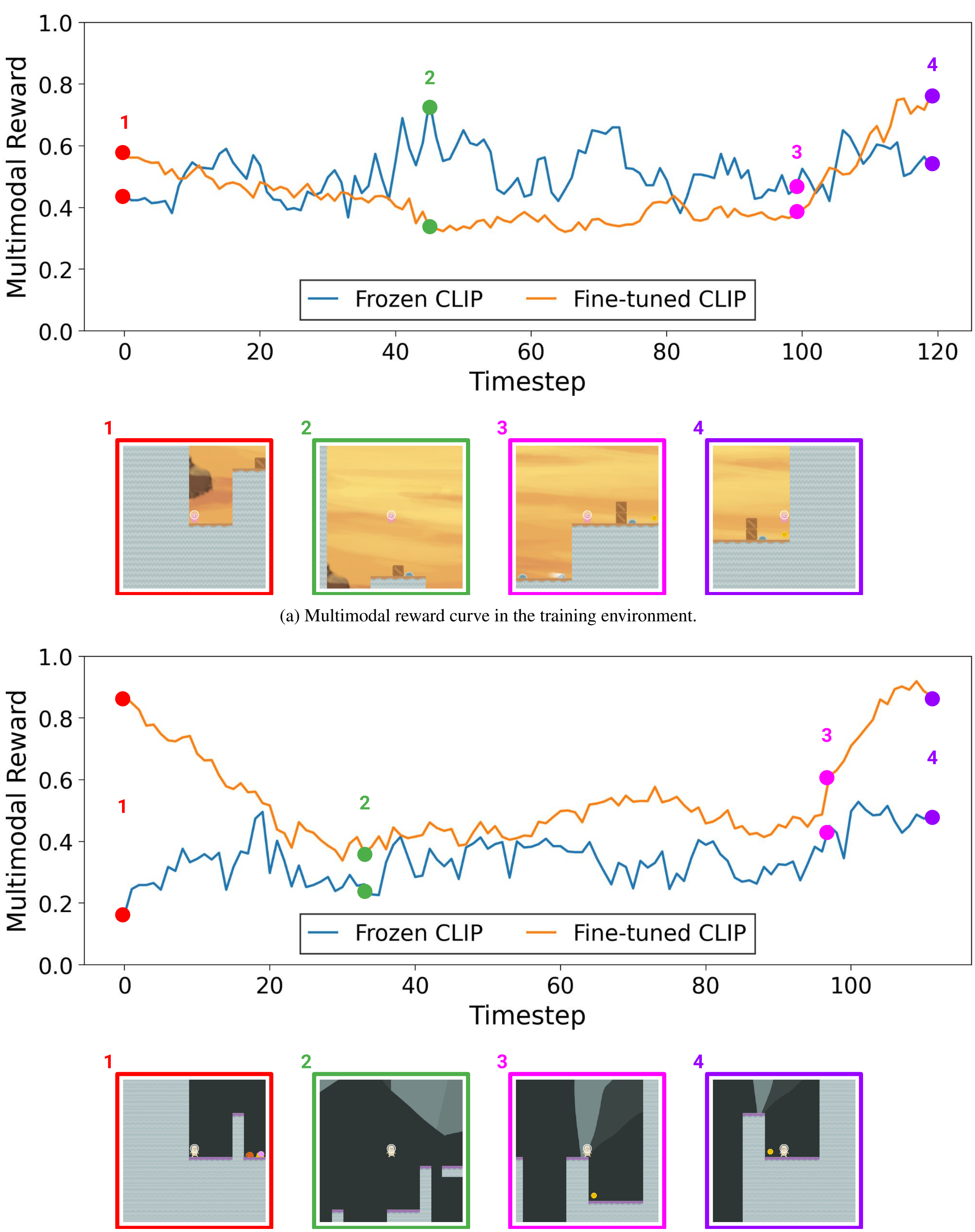
**Hypeparameters** We report the hyperparameters used in our experiments in Table 3.

Table 3: Hyperparameters of Multimodal Reward Decision Transformer (MRDT). Unless specified, we use the same hyperparameters used in *Instruct*RL (Liu et al., 2022a).

| **Hyperparameter** | **Value** |
|---|---|
| Policy batch size | 64 |
| Policy epochs | 50 |
| Policy context length | 4 |
| Policy learning rate | 0.0005 |
| Policy optimizer | AdamW (Loshchilov & Hutter, 2019) |
| Policy optimizer momentum | $\beta_1 = 0.9$, $\beta_2 = 0.999$ |
| Policy weight decay | 0.00005 |
| Policy learning rate decay | Linear warmup and cosine decay (see code for details) |
| Policy context length | 4 |
| Policy transformer size | 2 layers, 4 heads, 768 units |
| Fine-tuned CLIP batch size | 64 |
| Fine-tuned CLIP epochs | 20 |
| Fine-tuned CLIP learning rate | 0.0001 |
| Fine-tuned CLIP weight decay | 0.001 |
| Fine-tuned CLIP adapter layer size | 2 layers, 1024 units |
| Fine-tuned CLIP optimizer | AdamW (Loshchilov & Hutter, 2019) |
| Fine-tuned CLIP optimizer momentum | $\beta_1 = 0.9$, $\beta_2 = 0.999$ |

## B. Qualitative Results of Multimodal Rewards

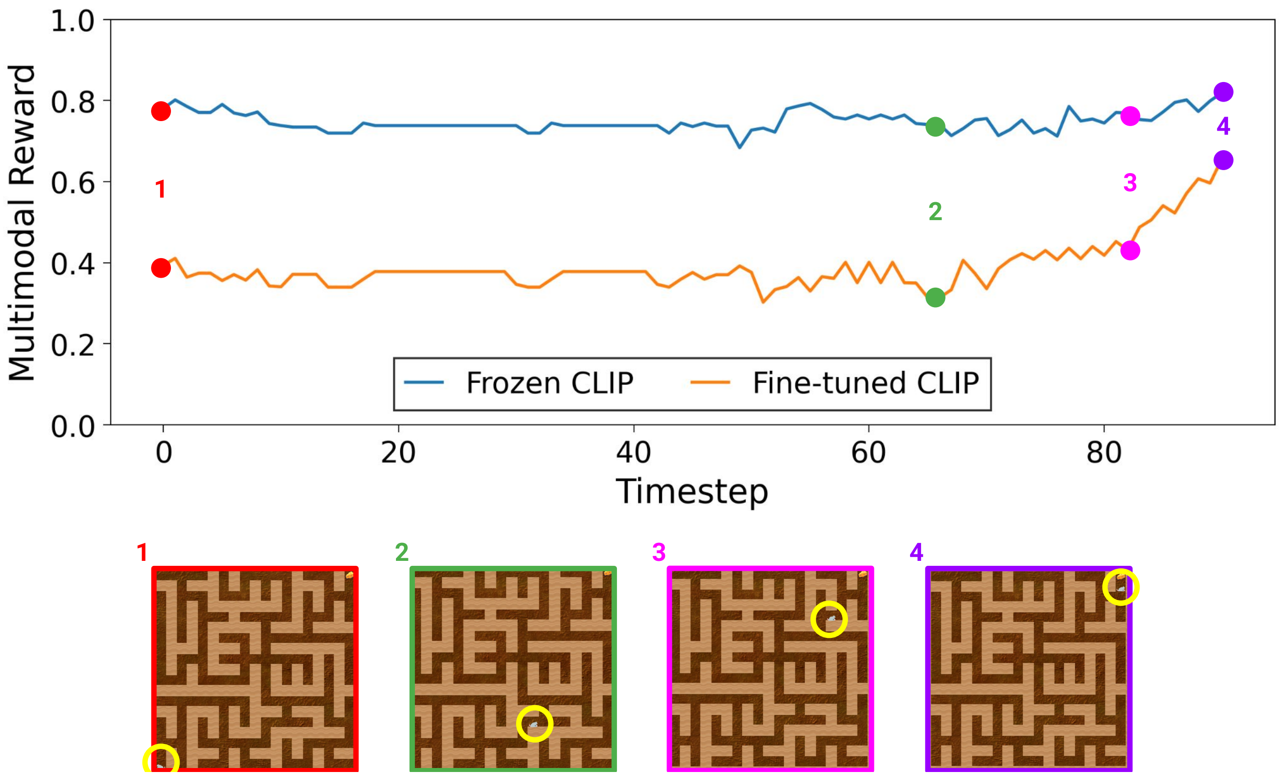
In Figure 8, 9, 10, we present the curves of multimodal rewards for frozen/fine-tuned CLIP in the trajectories from training/held-out evaluation environments. We find that the multimodal reward exhibits an overall increasing trend as the agent approaches the goal in both frozen and fine-tuned CLIP, irrespective of the training and held-out evaluation environments. Furthermore, we observe that fine-tuned CLIP not only induces a reward that is temporally smoother in the intermediate stages compared to frozen CLIP (see Figure 8) but also demonstrates a steeper upward reward curve (see Figure 9, 10). These results support the claim that the quality of multimodal rewards from the fine-tuned CLIP outperforms those from the frozen CLIP (Section 4.2).



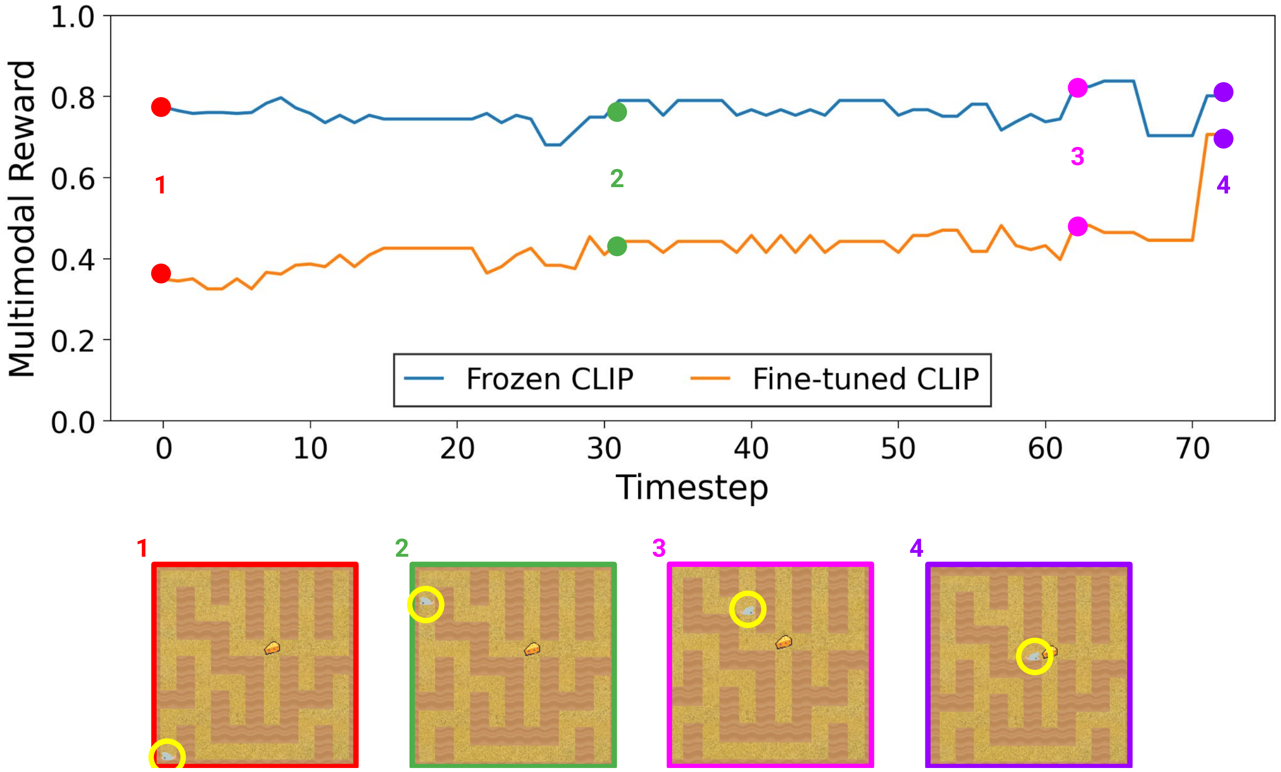
(a) Multimodal reward curve in the training environment.

(b) Multimodal reward curve in the held-out evaluation environment.

Figure 8: Qualitative results of multimodal rewards in CoinRun environments.

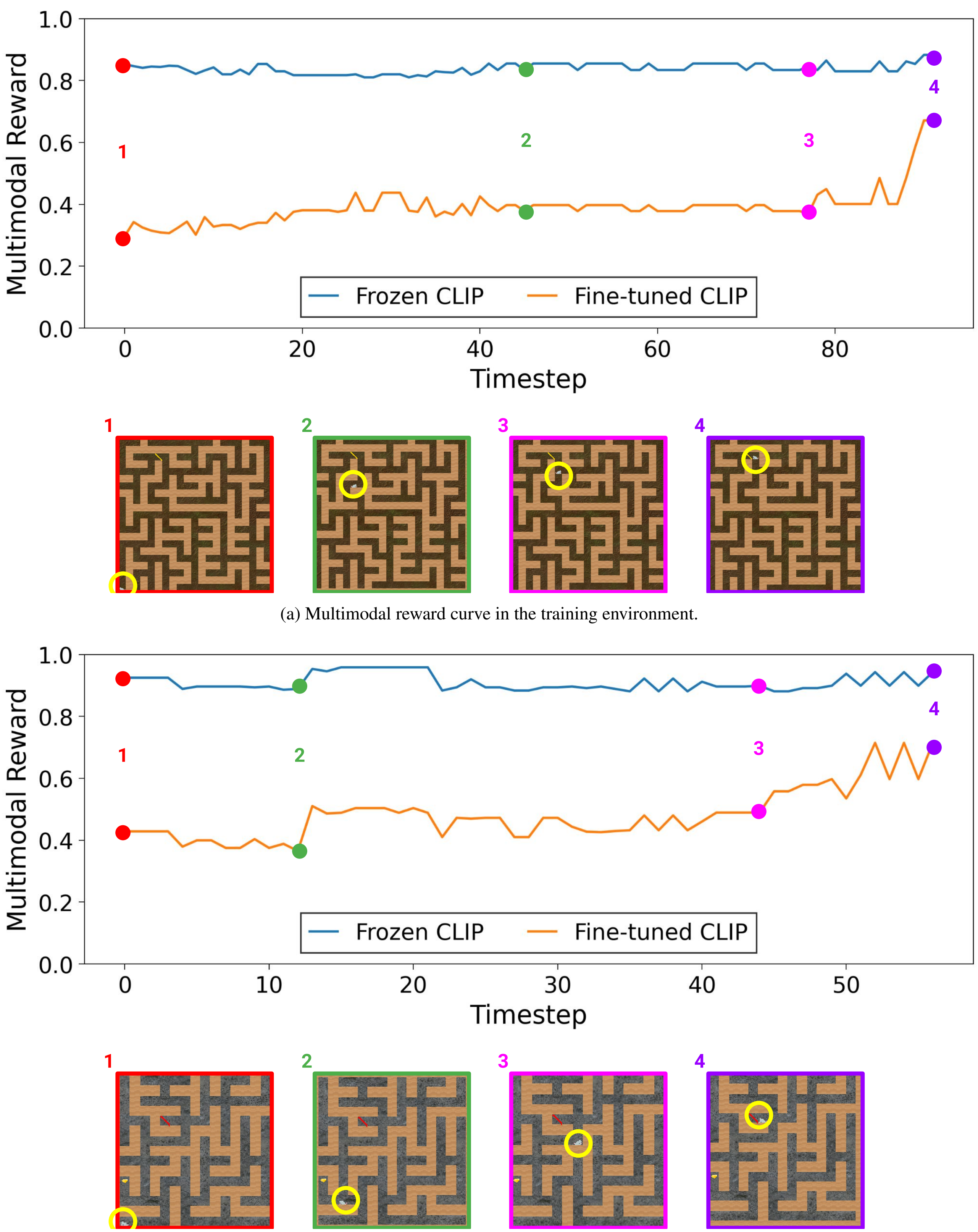


(a) Multimodal reward curve in the training environment.



(b) Multimodal reward curve in the held-out evaluation environment.

Figure 9: Qualitative results of multimodal rewards in Maze I environments.



(a) Multimodal reward curve in the training environment.

(b) Multimodal reward curve in the held-out evaluation environment.

Figure 10: Qualitative results of multimodal rewards in Maze II environments.

## C. Additional Experiments

Table 4: Expert-normalized scores on training/evaluation CoinRun environments investigating the effect of hyperparameter $\lambda$ adjusting the scale of return prediction loss in training return-conditioned policy. The result shows the mean and standard variation averaged over three runs.

| $\lambda$ | $\mathcal{L}_{\text{FT}}$ | Train (%) | Eval (%) |
|---|---|---|---|
| 0.001 | ✗ | 89.93% ± 3.94% | 62.65% ± 10.12% |
| | ✓ | 85.42% ± 1.80% | 71.69% ± 5.71% |
| 0.01 | ✗ | 89.58% ± 2.08% | 63.32% ± 2.01% |
| | ✓ | 90.28% ± 1.59% | 72.36% ± 3.48% |
| 0.1 | ✗ | 87.15% ± 2.62% | 62.65% ± 10.31% |
| | ✓ | 85.76% ± 3.18% | 73.37% ± 3.48% |
| 1.0 | ✗ | 87.15% ± 4.70% | 61.64% ± 6.38% |
| | ✓ | 81.25% ± 1.04% | 73.37% ± 2.66% |

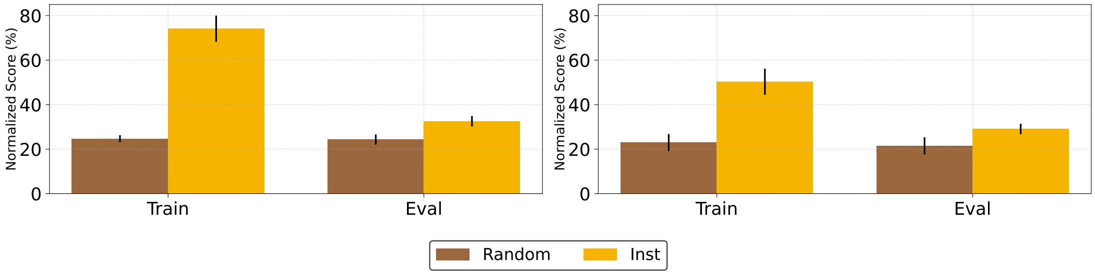


Figure 11: Expert-normalized scores on training/evaluation environments of MRDT trained using multimodal rewards generated with (i) instructive text (*i.e.,* Inst) and (ii) random text (*i.e.,* Random) in Maze I environments (left) and Maze II environments (right). The result shows the mean and standard deviation averaged over three runs.

**Effect of scaling return prediction loss** We investigate how the coefficient $\lambda$, which determines the weight of the return prediction loss in training return-conditioned policy, affects the performance of MRDT. To this end, we test various values of $\lambda$ in CoinRun environments. Table 4 shows the performance of MRDT in training/held-out evaluation environments with different $\lambda$. We find that performance is not significantly different according to the value of $\lambda$ in the held-out evaluation environments. These results indicate that MRDT is robust to the choice of hyperparameter $\lambda$.

**Extra ablation study on text instructions** In Figure 11, we further investigate whether MRDT leverages adaptive signals from multimodal rewards in decision-making. We evaluate the quality of rewards generated with instructive text (*i.e.,* Inst) and random text (*i.e.,* Random) in Maze I/II environments. Specifically, we use a natural language instruction for each environment, as described in Section 4 for Inst, and ”NeurIPS 2023 will be held again at the New Orleans Ernest N. Morial Convention Center” for Random. We find that using random text instructions results in a decline in performance in both training and evaluation environments. These findings align with the trend observed in Figure 7.

## D. Limitation and Future Work

One limitation of our work is that we currently rely on a single image-text pair to compute the multimodal reward at every timestep $t$. Although our approach has shown effectiveness both quantitatively and qualitatively, there are tasks where rewards depend on the history of past observations (*i.e.,* non-Markovian) (Bacchus et al., 1996; 1997; Kim et al., 2023). To address this limitation, it would be valuable to explore the extension of our method to incorporate video-text pairs for calculating multimodal rewards. This extension could involve generating multimodal rewards using pre-trained video-text multimodal representations (Liu et al., 2022b; Luo et al., 2022; Wang et al., 2022; Rasheed et al., 2023), which presents an intriguing avenue for better generalization across various goals in behavior learning. Another aspect to consider is that the tasks we have examined so far are relatively simple, as they involve only a single condition for success. To tackle more complex problems, we are interested in investigating approaches that leverage large language models (Huang et al., 2022a;b; Ahn et al., 2022; Driess et al., 2023) in conjunction with our method. Finally, an interesting direction to explore would be the utilization of multimodal rewards in combination with extrinsic rewards (Seo et al., 2021; Pathak et al., 2019; Burda et al., 2019).

## E. Potential Negative Societal Impacts

We do not anticipate significant negative societal impacts in that our method is now limited to playing simple simulation games. However, if our method is applied in real-world scenarios, privacy concerns may arise considering that behavior cloning agents used in such applications, like autonomous driving (Shah et al., 2023) or real-time control (Brohan et al., 2022; Driess et al., 2023), require large amounts of data, which often contain controversial information. Additionally, a behavior cloning policy presents a challenge as it imitates specified demonstrations, potentially including undesirable actions. If some bad actions are included in expert demonstrations (*e.g.,* behaviors that may be violent or harmful to the pedestrians are contained in the training data for mobile manipulation tasks), the policy could have significant negative impacts on users. To address this concern, future directions should focus on developing agents with safe adaptation in addition to performance enhancement efforts.