VIDEO-BASED OPTIMAL TRANSPORT FOR FEEDBACK-EFFICIENT OFFLINE PREFERENCE-BASED REINFORCEMENT LEARNING

Anonymous authors
Paper under double-blind review

ABSTRACT

Conveying complex objectives to reinforcement learning (RL) agents often requires meticulous reward engineering. Preference-based RL offers a promising alternative by learning reward functions from human feedback, but its scalability is hindered by the large amount of feedback required. Inspired by recent advances in Video Foundation Models (ViFMs), we present Video-based Optimal Transport Preference (VOTP), a semi-supervised preference learning framework that can learn effective reward functions from only a handful of preference labels. By leveraging optimal transport in the representation space of ViFMs for pseudo-labeling, VOTP can utilize large amounts of unlabeled data for reward learning, substantially reducing the need for human supervision. Extensive experiments across locomotion and manipulation tasks show that VOTP outperforms existing PbRL methods under limited feedback. We further validate VOTP on real robotic tasks, demonstrating its ability to learn useful rewards with minimal human input.

1 INTRODUCTION

Reinforcement learning (RL) has been successful in solving various decision-making tasks when a suitable reward function is available (Mnih et al., 2015; Silver et al., 2017; Haarnoja et al., 2018; Chen et al., 2022b). Yet in many real-world scenarios, reward design remains challenging. Constructing dense and informative rewards often requires extensive instrumentation, such as motion capture systems (Gupta et al., 2016), proprioceptive sensors (Zhu et al., 2019), or tactile sensors (Koenig et al., 2022). Even with such resources, reward misspecification can still occur, in which RL agents discover and exploit unintended shortcuts in the reward function (Skalse et al., 2022). In these cases, the reward signal may be maximized, but the resulting behaviors are often undesired or even harmful (Clark & Amodei, 2016; Popov et al., 2017).

Instead of hand-engineering reward functions, many works learn them directly from human data, such as expert demonstrations (Abbeel & Ng, 2004), natural language (Fu et al., 2019), and human feedback (Yuan et al., 2024). Recently, preference-based RL (PbRL) has gained considerable interest, as comparative feedback is easy for humans to provide yet informative enough to guide agents (Kaufmann et al., 2024; Casper et al., 2023). By querying human preferences over pairs of behavior clips, robot agents trained with PbRL have demonstrated the ability to perform novel behaviors (Christiano et al., 2017) and avoid reward exploitation (Lee et al., 2021a). With these promising results, PbRL has gained popularity in both online (Lee et al., 2021b; Cheng et al., 2024) and offline (Shin et al., 2023; Choi et al., 2024) settings. The PbRL framework often consists of two stages: reward learning from preferences, followed by policy optimization with the learned reward.

While PbRL methods can align agents with human intent, effective reward functions requires adequate coverage of both state and action spaces to achieve strong downstream performance (Ibarz et al., 2018; Hejna & Sadigh, 2023). Consequently, reward learning in PbRL is costly, often requiring thousands of human queries (Christiano et al., 2017; Shin et al., 2023; Yuan et al., 2024). To mitigate this challenge, prior work has explored several approaches, including semi-supervised learning (Park et al., 2022; Marta et al., 2024), meta-learning (Hejna III & Sadigh, 2023), active learning (Wang et al., 2022a), and preference ranking (Hwang et al., 2023; Choi et al., 2024). Yet a fundamental aspect remains underexplored—human preferences are shaped by the visual perception of agent behaviors, and leveraging these perceptual distinctions offers a promising direction for improving feedback efficiency. Our key insight is that the expressive and structured representation space of Video Foundation Models (ViFMs)—pre-trained on large-scale video corpora—can be harnessed to infer preferences for new behaviors by comparing them with known preferred examples.

To that end, we introduce Video-based Optimal Transport Preference labeling (VOTP), an algorithm that uses optimal transport over the ViFM representation space to automatically assign preference labels to unlabeled segment pairs, given only a small number of labeled preference queries (*e.g.*, 10 comparisons). Notably, unlabeled segment pairs can be obtained at no additional cost in PbRL settings, *e.g.*, from offline datasets. These pseudo-labeled segment pairs, together with the labeled ones, are then used to train the reward function. Specifically, VOTP uses optimal transport to find optimal alignments between labeled and unlabeled pairs in the ViFM latent space. The pseudo-label for an unlabeled pair is then inferred by aggregating preferences from all labeled pairs, weighted by their relative alignments computed from the optimal alignments. We conduct extensive experiments across three simulated domains—D4RL Gym locomotion (Fu et al., 2020), MetaWorld (Yu et al., 2020), and Robomimic (Mandlekar et al., 2021)—as well as two real-world robotic tasks. The results demonstrate that VOTP can learn effective policies from limited preference labels, substantially increasing feedback efficiency in PbRL. We also perform extensive analyses and ablations to better understand the sources of VOTP's performance gains.

2 Related Work

Preference-based RL (PbRL). PbRL enables agents to align with human intent through pairwise comparisons of behaviors, removing the need for manual reward engineering (Christiano et al., 2017). However, its scalability is constrained by the large amount of costly and labor-intensive human feedback it requires. To improve feedback efficiency, prior work has explored several directions, such as informative query selection (Bıyık et al., 2020; Wang et al., 2022a; Mu et al., 2025), pre-training of RL agents (Ibarz et al., 2018; Lee et al., 2021a), exploration guided by reward uncertainty (Liang et al., 2022), and preference rankings (Hwang et al., 2023; Choi et al., 2024). Other methods leverage pre-collected (sub-optimal) data to pre-train reward functions (Hejna III & Sadigh, 2023; Muslimani & Taylor, 2025). In contrast, we utilize unlabeled segment pairs from offline datasets for reward learning. Unlike (Park et al., 2022), which depends on learned reward models to perform pseudo-labeling, we employ optimal transport within the semantically meaningful latent space of Video Foundation Models (ViFMs) to infer pseudo-labels. This enables VOTP to learn effective reward functions from only a handful of preference feedbacks.

Vision Foundation Models in Reward Learning. With the rapid progress of foundation models, recent studies have explored their potential in constructing reward functions. One line of work leverages pre-trained vision-language models (VLMs) to directly reward RL agents by measuring alignments between trajectories and task descriptions (Cui et al., 2022; Rocamonde et al., 2024; Sontakke et al., 2024). However, these reward signals are often noisy and inconsistent (Wang et al., 2024). Another line of research utilizes the reasoning ability of VLMs to provide feedback (Wang et al., 2024; Luu et al., 2025a; Venkataraman et al., 2025; Luu et al., 2025b). Yet such approaches rely on carefully crafted prompt templates to be effective. In this work, we instead leverage ViFMs to generate pseudo-preference labels, aiming to enhance the feedback efficiency of PbRL.

Optimal Transport in Reinforcement Learning. Optimal Transport (OT) (Cuturi, 2013; Peyré et al., 2019) has been widely studied in domain adaptation (Courty et al., 2016), graph matching (Titouan et al., 2019; Ratnayaka et al., 2025), and semi-supervised learning (Tai et al., 2021; Tan et al., 2024). In the context of RL, prior works have applied OT to imitation learning (Fickinger et al., 2022; Luo et al., 2023; Fu et al., 2024; Huey et al., 2025) by minimizing the Wasserstein distance between the learner's trajectories and expert demonstrations. PEARL (Liu et al., 2024) extended this idea to transfer preferences across domains, but its applicability is restricted to tasks with identical state and action spaces, and cross-domain transfer often introduces high uncertainty for the target task. In contrast, VOTP performs pseudo-labeling directly within the same domain and scales naturally to high-dimensional visual inputs, enabling more stable and reliable reward learning in scenarios where PEARL is not applicable.

3 Preliminaries

Reinforcement Learning. In reinforcement learning (RL), an agent interacts with an environment modeled as a Markov decision process (MDP). MDP is defined by the tuple $\langle \mathcal{S}, \mathcal{A}, \mathcal{T}, r, \gamma \rangle$. At each time step t, the agent receives a state $\mathbf{s}_t \in \mathcal{S}$ and selects an action $\mathbf{a}_t \in \mathcal{A}$ based on its policy π. The environment responds by emitting a reward r_t and transitioning to the next state $\mathbf{s}_{t+1}$ according to the transition probability $\mathcal{T}(\mathbf{s}'|\mathbf{s}, \mathbf{a})$. In our setting, we also consider the observation $\mathbf{o}_t \in \mathcal{O}$, which is an image rendered from the underlying state $\mathbf{s}_t$. The return, $G_t = \sum_{k=0}^{\infty} \gamma^k r(\mathbf{s}_{t+k}, \mathbf{a}_{t+k})$, is defined as the discounted cumulative sum of rewards, with discount factor $\gamma \in [0, 1)$. The objective of RL algorithms is to learn a policy that maximizes the expected return.

Preference-based RL. In offline preference learning, we assume that the true reward function is unknown and instead learn a reward function $\widehat{r}_\psi$ from human preferences (Christiano et al., 2017; Ibarz et al., 2018). A trajectory segment of length H is represented as a sequence of states and actions $\{(\mathbf{s}_1, \mathbf{a}_1), \ldots, (\mathbf{s}_H, \mathbf{a}_H)\}$. Given a pair of segments (σ^0, σ^1), a teacher provides a preference label $\tilde{y} \in \{0, 1, 0.5\}$, where $\tilde{y} = 0$ indicates $\sigma^0 \succ \sigma^1$, $\tilde{y} = 1$ indicates $\sigma^1 \succ \sigma^0$, and $\tilde{y} = 0.5$ indicates equal preference. Here, $\sigma^i \succ \sigma^j$ denotes that segment i is preferred over segment j. Each feedback is stored in a preference dataset $\mathcal{D}$ as a triple $(\sigma^0, \sigma^1, \tilde{y})$. The preference predictor is modeled using the reward function $\widehat{r}_\psi$ following the Bradley-Terry model (Bradley & Terry, 1952):

$$P[\sigma^0 \succ \sigma^1; \psi] = \frac{\exp\left(\sum_t \widehat{r}_\psi(\mathbf{s}_t^0, \mathbf{a}_t^0)\right)}{\exp\left(\sum_t \widehat{r}_\psi(\mathbf{s}_t^0, \mathbf{a}_t^0)\right) + \exp\left(\sum_t \widehat{r}_\psi(\mathbf{s}_t^1, \mathbf{a}_t^1)\right)}. \tag{1}$$

Given the preference dataset, the estimated reward function $\widehat{r}_\psi$ is updated by minimizing the cross-entropy loss between predicted preferences and annotated labels:

$$\mathcal{L}(\psi) = -\mathop{\mathbb{E}}_{(\sigma^0, \sigma^1, \tilde{y}) \sim \mathcal{D}} \Big[(1 - \tilde{y}) \log P[\sigma^0 \succ \sigma^1; \psi] + \tilde{y} \log P[\sigma^1 \succ \sigma^0; \psi] \Big]. \tag{2}$$

In practice, a preference query is typically presented to teachers as a pair of short video clips rendered from trajectory segments. While intuitive, learning an effective reward model often demands hundreds to thousands of annotated comparisons (Kim et al., 2023; Hejna & Sadigh, 2023; Hejna et al., 2024; Choi et al., 2024), which creates an unsustainable annotation burden. To mitigate this challenge, we adopt the semi-supervised preference learning paradigm (Park et al., 2022), which leverages both labeled and unlabeled segment pairs for reward learning.

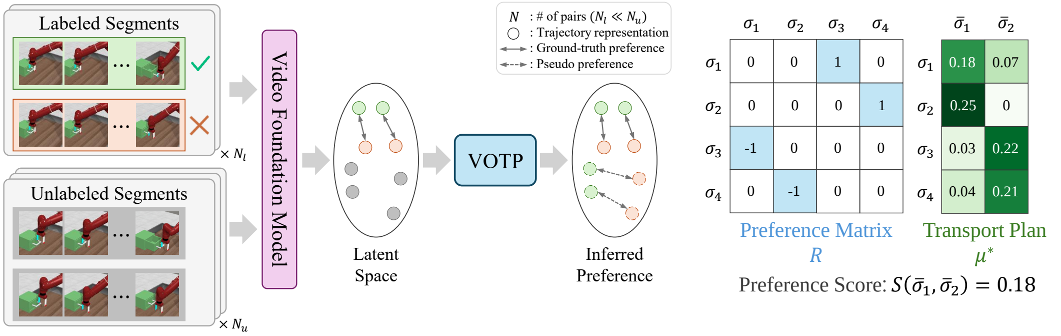
Discrete Optimal Transport. Optimal Transport (OT) (Cuturi, 2013; Peyré et al., 2019) is an optimization problem that finds a coupling between two probability measures with minimal cost. Let $\Delta_n = \{\mathbf{p} \in \mathbb{R}_+^n | \sum_{i=1}^n p_i = 1\}$ denote the probability simplex of dimension n. Consider two probability measures $\mu_x = \sum_{i=1}^n p_i \delta_{x_i}$ and $\mu_y = \sum_{j=1}^m q_j \delta_{y_j}$, supported on $\{x_i\}_{i=1}^n$ and $\{y_j\}_{j=1}^m$, respectively. Here, the weight vector $\mathbf{p} = (p_1, \ldots, p_n)$ and $\mathbf{q} = (q_1, \ldots, q_m)$ belong to Δ_n and Δ_m, respectively, and δ_x denotes the Dirac measure at x. The discrete OT problem between μ_x and μ_y can then be expressed via the Wasserstein distance as:

$$\mathcal{W}_2^2(\mu_x, \mu_y) = \min_{\mu \in \mathcal{M}} \sum_{i=1}^n \sum_{j=1}^m c(x_i, y_j) \mu_{ij}, \tag{3}$$

where $\mathcal{M} = \{\mu \in \mathbb{R}_+^{n \times m} : \mu \mathbf{1}_m = \mu_x, \mu^\top \mathbf{1}_n = \mu_y\}$ is the set of feasible transport plans, $\mathbf{1}_n$ denotes the all-ones vector of dimension n, and $c(x, y)$ is the cost function. The matrix μ specifies a transport plan, where μ_{ij} indicates the mass moved from x_i to y_j. In this work, we leverage OT to compute correspondences between unlabeled and labeled segment pairs, thereby enabling the inference of pseudo-preference labels.

4 Method

Our goal is to improve feedback efficiency in offline preference learning by leveraging unlabeled data. To this end, we introduce Video-based Optimal Transport Preference (VOTP), a semi-supervised framework that infers pseudo-preference labels using optimal transport (OT). The framework consists of two key components: (i) trajectory representation with Video Foundation Models, and (ii) pseudo-preference label generation through the optimal transport plan. An overview is provided in Figure 1.



	σ_1	σ_2	σ_3	σ_4
σ_1	0	0	1	0
σ_2	0	0	0	1
σ_3	-1	0	0	0
σ_4	0	-1	0	0

	$\bar{\sigma}_1$	$\bar{\sigma}_2$
σ_1	0.18	0.07
σ_2	0.25	0
σ_3	0.03	0.22
σ_4	0.04	0.21

(a) Pipeline of Video-based Optimal Transport Preference (VOTP) (b) Computation performed by VOTP

Figure 1: Overview of our framework. (a) VOTP embeds visual segments into a latent space using an off-the-shelf video foundation model and uses the optimal transport plan to propagate preferences with relative alignment strengths. Green dots indicate preferred segments over orange ones. (b) Example computation in VOTP with four labeled segments (σ_i) and two unlabeled segments ($\bar{\sigma}_{i'}$). Preference relations among labeled segments are represented by the preference matrix R. Each entry of the optimal transport plan μ^* specifies the probability that a labeled segment matches an unlabeled segment, and the unnormalized preference score is computed using Eq. (6).

4.1 Trajectory Representation

Representing trajectory segments in a form that enables reliable comparison is central to preference learning (Tian et al., 2024; Mu et al., 2025). We model each segment as a short video clip, $\sigma = \{\mathbf{o}_1, \ldots, \mathbf{o}_H\}$, and embed it into a latent space using a trajectory encoder f_ϕ:

$$\mathbf{z} = f_\phi(\mathbf{o}_{1:H}). \tag{4}$$

An effective encoder must capture both spatial details within frames and temporal dynamics across the segment, as these jointly determine the behavioral differences reflected in human preferences. To meet these requirements, we adopt off-the-shelf video foundation models (ViFMs) (Madan et al., 2024), which are pre-trained on massive collections of human activity videos covering diverse actors, viewpoints, lighting conditions, and backgrounds. This large-scale, heterogeneous pre-training produces actor-agnostic, semantically rich embeddings that are robust to nuisance variation and generalize to unseen robotic environments.

4.2 Pseudo-preference Label Generation

VOTP first identifies correspondences between labeled and unlabeled segment representations, and then assigns preferences via an OT plan. We denote the labeled dataset as $\mathcal{D}_l = \{(\sigma^0, \sigma^1, \tilde{y})^{(i)}\}_{i=1}^{N_l}$ and the unlabeled dataset as $\mathcal{D}_u = \{(\bar{\sigma}^0, \bar{\sigma}^1)^{(i)}\}_{i=1}^{N_u}$. Our objective is to infer pseudo labels for $\mathcal{D}_u$ and use both datasets to learn the reward function $\widehat{r}_\psi$.

We define the labeled set as $L = \{\sigma_i\}_{i=1}^{N}$, where $N = 2N_l$ denotes the total number of segments in $\mathcal{D}_l$. Preference relations among segments are encoded in a preference matrix $R \in \{-1, 0, 1\}^{N \times N}$:

$$R_{ij} = \begin{cases} -1 & \text{if } \sigma_i \succ \sigma_j, \\ 1 & \text{if } \sigma_j \succ \sigma_i, \\ 0 & \text{for } i = j \text{, ties, or no preference is available.} \end{cases}$$

By construction, R is skew-symmetric, *i.e.*, $R^\top = -R$. In parallel, we define the unlabeled set $U = \{\bar{\sigma}_{i'}\}_{i'=1}^{M}$, consisting of M segments sampled from $\mathcal{D}_u$, for which pseudo-preference labels are inferred. Let $\mu_L = \sum_{i=1}^{N} p_i \delta_{\sigma_i}$ and $\mu_U = \sum_{i'=1}^{M} q_{i'} \delta_{\bar{\sigma}_{i'}}$ denote the empirical measures on these sets. For simplicity, we adopt the uniform weights, *i.e.*, $p_i = \frac{1}{N}$ and $q_{i'} = \frac{1}{M}$. The OT plan for aligning labeled and unlabeled segments is then obtained as

$$\mu^* = \arg\min_{\mu \in \mathcal{M}} \sum_{i=1}^{N} \sum_{i'=1}^{M} c(\sigma_i, \bar{\sigma}_{i'}) \mu_{ii'}, \tag{5}$$

where $\mathcal{M} = \{\mu \in \mathbb{R}_+^{N\times M} : \mu\mathbf{1}_M = \frac{1}{N}\mathbf{1}_N, \mu^\top\mathbf{1}_N = \frac{1}{M}\mathbf{1}_M\}$. The cost function is defined as $c(\sigma_i, \bar{\sigma}_{i'}) = d(f_\phi(\sigma_i), f_\phi(\bar{\sigma}_{i'}))$, the distance between encoded visual segments in the latent video space, where d can be chosen as either the Euclidean distance or the cosine distance.

The OT plan μ^* obtained in Eq. (5) provides the correspondences between segments in sets L and U. Concretely, each entry $\mu_{ii'}$ represents the probability that the unlabeled segment $\bar{\sigma}_{i'}$ matches the labeled segment σ_i. Combining these probabilities with the preference matrix R, we can infer preferences between segments in the unlabeled set U. For brevity, we denote the OT plan as μ. We then define the preference score used to determine the preference between the unlabeled pair $(\bar{\sigma}_{i'}, \bar{\sigma}_{j'})$ as follows:

$$S(\bar{\sigma}_{i'}, \bar{\sigma}_{j'}) = \sum_{i=1}^{N}\sum_{j=1}^{N} R_{ij}(\mu_{ii'}\mu_{jj'} - \mu_{ij'}\mu_{ji'}) \tag{6}$$

Interpretation. Consider a labeled pair (i, j) with a non-zero preference (*i.e.*, $R_{ij} \neq 0$). Suppose $R_{ij} = 1$ (*i.e.*, $\sigma_j \succ \sigma_i$). The term $\mu_{ii'}\mu_{jj'}$ measures alignment between (σ_i, σ_j) and $(\bar{\sigma}_{i'}, \bar{\sigma}_{j'})$, while $\mu_{ij'}\mu_{ji'}$ measures the alignment with the reversed pair $(\bar{\sigma}_{j'}, \bar{\sigma}_{i'})$. If the difference $(\mu_{ii'}\mu_{jj'} - \mu_{ij'}\mu_{ji'})$ is positive, then $(\bar{\sigma}_{i'}, \bar{\sigma}_{j'})$ likely shares the preference of (σ_i, σ_j), implying $\bar{\sigma}_{j'} \succ \bar{\sigma}_{i'}$. Conversely, if the difference is negative, the preference is flipped, *i.e.*, $\bar{\sigma}_{i'} \succ \bar{\sigma}_{j'}$. The preference score for $(\bar{\sigma}_{i'}, \bar{\sigma}_{j'})$ is then obtained by aggregating alignment comparisons across all labeled pairs. Since R is skew-symmetric, the inferred preference for $(\bar{\sigma}_{i'}, \bar{\sigma}_{j'})$ is consistent under swapping (i, j). An example of this computation is shown in Figure 1 (b). Overall, VOTP leverages the transport plan to propagate preferences from labeled to unlabeled pairs through relative alignment strengths.

In practice, the entries of the OT plan μ are small because $\sum \mu_{ij} = 1$, which leads to relatively small preference scores. Therefore, we normalize the preference score by

$$S_{\text{max}} = \sum_{i=1}^{N}\sum_{j=1}^{N} \frac{1}{N^2}\mathbb{1}(R_{ij} \neq 0). \tag{7}$$

Here, S_{max} denotes the absolute maximum attainable score under uniform masses (*i.e.*, $p_i = \frac{1}{N}$), assuming the OT plan maximizes the contribution of all non-zero R_{ij} terms. This guarantees that preference scores lie within $[-1, 1]$ across varying numbers of labeled pairs. Finally, to obtain the preference label for the pair $(\bar{\sigma}_{i'}, \bar{\sigma}_{j'})$, we apply a preference threshold τ_P to determine the label[1]:

$$\tilde{y} = \begin{cases} \frac{1}{2}(1 + \text{sign}(S_{\text{norm}}(\bar{\sigma}_{i'}, \bar{\sigma}_{j'}))) & \text{if } |S_{\text{norm}}| \geq \tau_P, \\ 0 & \text{otherwise.} \end{cases} \tag{8}$$

where $\text{sign}(x) = -1$ if $x < 0$, 1 if $x > 0$, and 0 if $x = 0$; and $S_{\text{norm}} = S(\bar{\sigma}_{i'}, \bar{\sigma}_{j'})/S_{\text{max}}$.

4.3 Implementation Details

Obtaining the optimal coupling matrix μ^* in Eq. (5) requires solving a linear program, which is computationally expensive with standard solvers. In practice, we solve the entropy-regularized OT problem using Sinkhorn's algorithm (Cuturi, 2013), which provides both efficiency and numerical stability. Our implementation uses the Sinkhorn solver from the POT toolbox (Flamary et al., 2021). After VOTP annotates the unlabeled dataset with pseudo-preferences, we train the reward function $\widehat{r}_\psi$ using Eq. (2). To mitigate the impact of inaccurate pseudo-labels, we retain only those with scores above the threshold τ_P. During RL training, all state-action pairs in the offline dataset are relabeled using the trained $\widehat{r}_\psi$. The overall procedure is summarized in Algorithm 1 in the Appendix.

5 Experiments

In this section, we conduct experiments across diverse domains to answer the following questions:

1. Can VOTP improve feedback efficiency in limited-data settings?
2. What is the contribution of each component within VOTP?
3. How does VOTP perform under varying numbers of labeled queries?
4. How do key parameters influence the performance of VOTP?
5. Can VOTP be directly applied to real robots?

[1] One could optionally apply an additional threshold to treat pairs with scores near zero as equally preferable.

Table 1: Average scores on D4RL locomotion and success rates on MetaWorld and Robomimic manipulation tasks. We run five seeds and report the final performance at the end of training like Kostrikov et al. (2022). Bold values indicate results within 5% of the best-performing method (excluding IQL). Learning curves and IQM normalized returns are provided in the Appendix.

Dataset	IQL with	Learning with Preference			
	task reward	IPL	P-IQL	SURF	VOTP (Ours)
hopper-medium-replay-v2	87.5 ± 7.4	22.1 ± 4.9	36.5 ± 15.4	9.3 ± 0.6	**91.1 ± 4.7**
hopper-medium-expert-v2	104.5 ± 4.5	62.6 ± 18.4	89.1 ± 18.4	65.5 ± 17.0	**105.7 ± 6.0**
walker2d-medium-replay-v2	72.6 ± 4.9	8.6 ± 5.4	32.4 ± 27.1	**64.9 ± 9.4**	**66.3 ± 5.6**
walker2d-medium-expert-v2	109.9 ± 0.5	92.4 ± 10.2	**103.4 ± 7.0**	**109.7 ± 1.1**	**108.1 ± 2.2**
locomotion average	93.6	46.4	65.3	59.5	92.8
door-open	79.2 ± 5.9	48.8 ± 11.7	36.8 ± 13.2	74.4 ± 10.3	**84.0 ± 8.4**
drawer-open	83.2 ± 4.7	51.2 ± 13.5	36.0 ± 13.6	57.6 ± 15.7	**71.2 ± 11.7**
plate-slide	56.0 ± 11.9	28.0 ± 9.1	15.2 ± 5.9	23.2 ± 5.9	**57.6 ± 5.4**
sweep-into	65.6 ± 5.4	41.6 ± 3.2	36.0 ± 8.0	40.8 ± 4.7	**57.6 ± 7.4**
metaworld average	71.0	42.4	31.0	49.0	67.6
can-mh	65.0 ± 9.5	31.2 ± 8.2	41.0 ± 10.2	28.0 ± 8.1	**70.0 ± 8.4**
can-ph	67.5 ± 7.5	50.0 ± 8.7	43.0 ± 18.3	34.0 ± 13.9	**66.0 ± 5.8**
lift-mh	84.0 ± 4.9	51.2 ± 16.3	40.0 ± 20.2	**68.0 ± 16.3**	**71.0 ± 22.7**
lift-ph	97.0 ± 4.0	**95.0 ± 3.5**	86.0 ± 9.7	84.0 ± 13.6	**97.0 ± 4.0**
robomimic average	78.4	56.9	52.5	53.5	76.0

5.1 Setups

Dataset. In simulated environments, we evaluate VOTP on complex robotic locomotion and manipulation tasks in the offline preference-based RL (PbRL) setting (Shin et al., 2023; Kim et al., 2023; Hejna & Sadigh, 2023; An et al., 2023). Concretely, we consider three domains: D4RL Gym locomotion (Fu et al., 2020), MetaWorld (Yu et al., 2020), and Robomimic (Mandlekar et al., 2021), using offline PbRL datasets from Kim et al. (2023) and Hejna et al. (2024). For the initial labeled dataset, we use only a few labels (5 or 10, depending on the dataset) by randomly selecting queries (pairs of trajectory segments) and utilizing scripted labels[2] derived from ground-truth rewards, a common practice in PbRL evaluation (Lee et al., 2021b; Shin et al., 2023; Choi et al., 2024). For pseudo-labeling, we sample additional pairs uniformly at random from the offline datasets. Specifically, we use a total of 10k queries for D4RL Gym locomotion and Robomimic, and 50k queries for MetaWorld. Since VOTP performs labeling on image-based observations, we render visual observations corresponding to the states in the preference datasets.

Training Details. For computing the optimal coupling, we use the Sinkhorn solver from POT (Flamary et al., 2021), a library for optimal transport that provides efficient computation of the Sinkhorn algorithm with accelerator support. We use Euclidean function as the cost function. As the trajectory encoder, we adopt S3D (Xie et al., 2018; Miech et al., 2020), a ViFM pre-trained on HowTo100M (Miech et al., 2019), which consists of large-scale third-person clips of everyday human activities. For reward learning, we use both labeled and pseudo-labeled pairs, retaining pseudo-labels above the threshold τ_P (Eq. 8). After training the reward model, we replace the original rewards in the offline dataset with the learned rewards and then train the policy using an offline RL algorithm. VOTP can be applied to any offline RL algorithm, but as in prior work, we use IQL (Kostrikov et al., 2022). Across PbRL baselines, both the policy and reward models are trained from states and share the same policy-learning hyperparameters. Thus, the only difference lies in the reward learning process. We also apply temporal data augmentation (Park et al., 2022; Hejna & Sadigh, 2023) across baselines. Further implementation details can be found in the Appendix.

Evaluation. We evaluate performance using normalized scores on D4RL and success rates on MetaWorld and Robomimic. For all experiments, we report the mean and standard deviation across

[2] For hopper-medium-replay-v2, we use human labels from Kim et al. (2023), since scripted labels remain ineffective across baselines even when provided in large quantities.

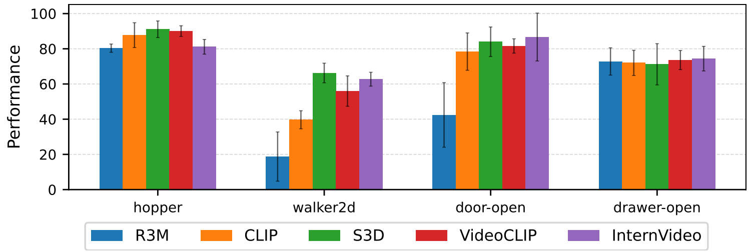


Figure 2: Ablation with various trajectory encoders in D4RL and MetaWorld. For *hopper* and *walker2d*, we use medium-replay datasets. Results are averaged over five runs.

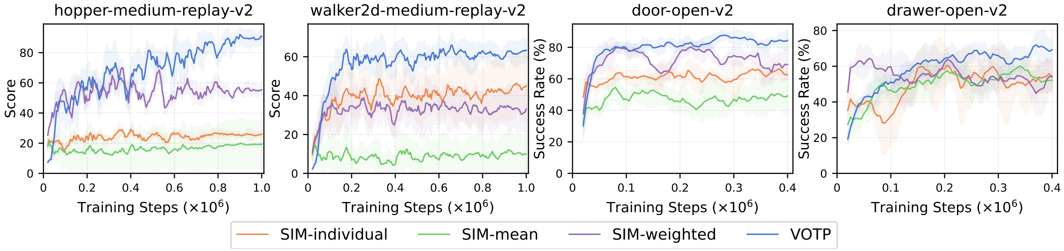


Figure 3: The effectiveness of using optimal transport to infer pseudo-labels. Results are averaged over five runs with standard deviation (shaded area).

five runs, with each run evaluated with 25 episodes per evaluation step. Full learning curves and interquartile mean (IQM) (Agarwal et al., 2021) results are provided in the Appendix.

5.2 Evaluation on the Offline PbRL Benchmark

We compare VOTP with the following baselines. IQL learns policies using task rewards. Preference IQL (P-IQL) learns a reward model from the labeled dataset, then trains a policy with IQL to maximize the learned reward. SURF (Park et al., 2022) is similar to P-IQL, but trains the reward model using both labeled data and pseudo-labels generated from confidence estimates of the preference predictor. Finally, Inverse Preference Learning (IPL) (Hejna & Sadigh, 2023) is a method that learns policies directly from preferences without a reward model and has been shown to be effective in data-limited settings. For a fair comparison, semi-supervised PbRL methods are trained using same labeled and unlabeled datasets across all tasks, while standard PbRL methods are trained using the same labeled datasets.

Table 1 summarizes the performance of all methods across three domains. As shown, VOTP consistently outperforms all preference-based baselines in terms of average performance. Furthermore, it achieves task-reward performance on 8 of 12 datasets, demonstrating its effectiveness for reward learning with limited labeled data. Among standard PbRL methods, IPL generally performs better than P-IQL in MetaWorld and Robomimic, consistent with prior work (Hejna & Sadigh, 2023), yet both remain far below IQL with task rewards. While SURF improves P-IQL on some datasets, its performance is inconsistent and can sometimes degrade, likely due to overconfidence of preference models during pseudo-labeling under limited supervision (Chen et al., 2022a; Tan et al., 2024), resulting in inaccurate pseudo-labels. In contrast, by leveraging the expressive and structured representation space of pre-trained ViFMs, VOTP employs the OT plan to acquire more reliable pairwise comparisons, leading to higher-quality pseudo-labels for reward learning and, consequently, stronger RL agent performance.

5.3 Ablation Studies

Effect of Video Foundation Models. We assess the role of the video encoder in VOTP by comparing image foundation models (IFMs) and video foundation models (ViFMs) in encoding visual segments. For IFMs, we adopt R3M (Nair et al., 2022) and CLIP (Radford et al., 2021), which are widely used for feature extraction and reward computation (Adeniji et al., 2023; Zhang et al.,

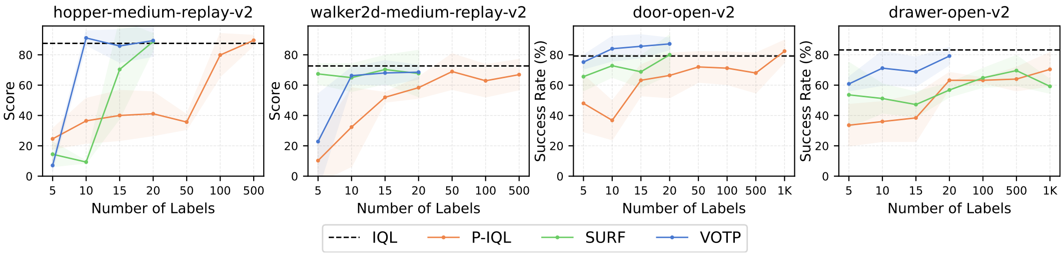


Figure 4: Average performance of each method as the number of preference feedbacks varies.

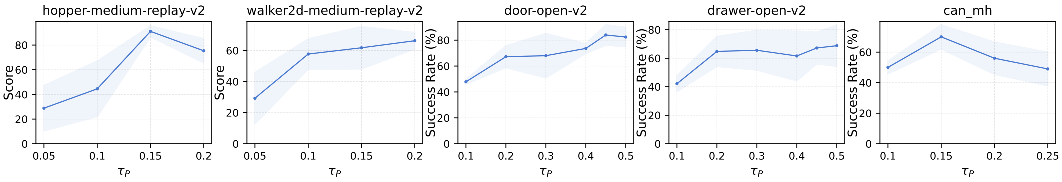


Figure 5: Performance of VOTP under different values of the preference threshold τ_P.

2023; Rocamonde et al., 2024). For ViFMs, we adopt S3D (Xie et al., 2018; Miech et al., 2020), VideoCLIP (Xu et al., 2021), and InternVideo (Wang et al., 2022b).

The results, shown in Figure 2, indicate that ViFMs generally perform better than IFMs, particularly in *walker2d* and *door-open*. This highlights their advantage in providing richer segment representations by capturing temporal dynamics and subtle motion cues, which are crucial for distinguishing behavioral differences when determining preferences. In our framework, we opt for S3D, as it achieves robust performance across tasks while requiring far fewer parameters (31M) than VideoCLIP (208M) and InternVideo (478M). This balance of effectiveness and efficiency makes S3D appealing when operating under limited compute or memory budgets.

Effect of Optimal Transport. To assess the benefits of OT in pseudo-label inference, we compare against baselines that infer preferences solely based on similarity. Specifically, we divide the labeled set into preferred and non-preferred groups. The first baseline, *SIM-individual*, assigns the label of the most similar labeled pair to an unlabeled pair. The second baseline, *SIM-mean*, instead compares with the aggregated representation of each group, obtained by averaging feature vectors. The third baseline, *SIM-weighted*, assigns labels based on a similarity-weighted average of preferences from the labeled pairs. In contrast, VOTP aggregates all preference labels from labeled pairs, weighting their contributions by the relative alignment strengths computed from the OT plan, thereby producing more reliable pseudo-labels. The results in Figure 3 demonstrate a clear advantage of our method. We also observe that *SIM-mean* performs worse than *SIM-individual*, likely because averaging group features discards fine-grained distinctions between pairs, which are crucial for assigning pseudo-preferences. Although *SIM-weighted* improves over *SIM-individual* on some tasks, its overall performance remains noticeably lower and less stable. This underscores the importance of our OT-based formulation for generating robust pseudo-labels.

Varying the number of queries. We evaluate how the number of queries affects PbRL performance in two domains: D4RL and MetaWorld. Concretely, we measure the average performance of each method while varying the labeled dataset size, ranging from 5 to 1000 preference labels depending on the domain. We note that most previous work on D4RL uses up to 500 preferences (Kim et al., 2023; An et al., 2023), while MetaWorld typically uses up to 10k (Hejna & Sadigh, 2023; Hejna et al., 2024). Results are shown in Figure 4. In D4RL, without pseudo-labels, P-IQL requires roughly 50-100 labels to match task-reward performance, whereas in MetaWorld it requires around 1k. Incorporating pseudo-labels improves performance in both domains. Importantly, we find that, except *walker-medium-replay*, VOTP requires fewer labels than baselines to reach task-reward performance. Notably, in *door-open*, VOTP with only 10 labels outperforms the policy trained with ground-truth rewards. Overall, these results demonstrate the high feedback efficiency of VOTP, confirming its effectiveness in limited-data regimes.

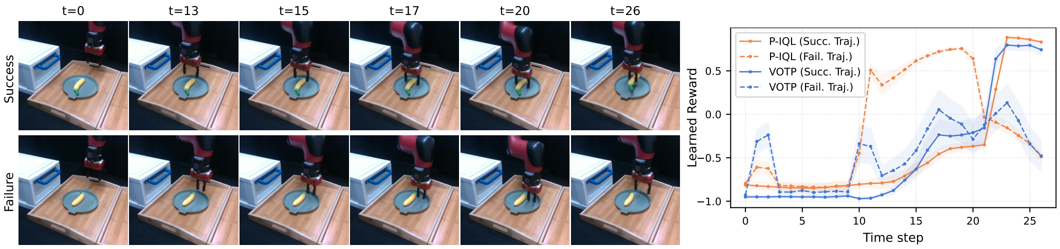


Figure 6: Lift Banana: Examples of successful and failed trajectories at each time step (left) with the corresponding reward outputs over timesteps from VOTP and P-IQL (right).

Impact of the preference threshold. We examine how the preference threshold τ_P affects the performance of VOTP. Concretely, we vary τ_P and measure the corresponding performance of VOTP. Results are shown in Figure 5. We observe that performance generally improves as the threshold increases, but slightly drops with a large value, as seen in *hopper-medium-replay* and *can-mh*. This effect arises because our unlabeled dataset size is fixed due to the rendering cost of visual segments, and only pseudo-labels above τ_P are retained for training. Thus, increasing τ_P enhances label quality but reduces their quantity, which can harm performance. In practice, we tune this parameter to balance the quality-quantity trade-off of pseudo-labels by selecting values within the observed range of normalized preference scores.

5.4 Real Robot Evaluation

Table 2: Success rates over 10 episodes on the 2 real-world manipulation tasks.

Method	Lift Banana	Drawer Open
BC	20.0	40.0
P-IQL	50.0	50.0
VOTP	**80.0**	**70.0**

We further evaluate VOTP in a real-world robotic manipulation setting using a 7-DoF Rethink Sawyer robotic arm. We compare our method against two baselines: Behavior Cloning (BC) and P-IQL. The experiments are conducted with two vision-based manipulation tasks: *Lift Banana* and *Drawer Open*. In our setting, the policy input consists of proprioceptive states and image observations captured from a camera. For each task, we collect 50 demonstrations via keyboard teleoperation with a 50% success rate. To collect preferences, we present pairs of video clips to a human teacher. We use 5 and 10 preference labels for *Lift Banana* and *Drawer Open*, respectively. The number of unlabeled pairs is 2000 and 3000, respectively. The policy is trained using IQL (Kostrikov et al., 2022), with the reward model optimized according to Eq. (2). P-IQL and VOTP are trained in the same way as in the simulated experiments, *i.e.*, P-IQL is trained with a small number of labeled preferences, while VOTP is additionally trained with pseudo-labels. Table 2 reports the comparison with baselines, showing that by leveraging unlabeled data, VOTP enables the agent to achieve higher performance. To highlight the benefit of unlabeled data, Figure 6 shows reward outputs from VOTP and P-IQL on a successful and a failed trajectory. Both methods yield reasonable rewards for the successful trajectory, but P-IQL mistakenly assigns high rewards to failed behavior (timesteps 11-20). In contrast, VOTP produces well-separated rewards between successful and failed trajectories. Additional results are provided in the Appendix.

6 Discussion

In this work, we introduce Video-based Optimal Transport Preference (VOTP), a novel semi-supervised preference learning that employs optimal transport over embedding space of video foundation models (ViFMs) to automatically infer preferences for unlabeled pairs. This enables VOTP to learn effective reward functions from only a handful of preference labels, substantially reducing the need for human supervision. Extensive experiments across locomotion, manipulation, and real-world robotic manipulation tasks validate the effectiveness of our approach, highlighting VOTP as a scalable and practical solution for preference-based reinforcement learning.

Limitations. Since VOTP relies on pre-trained ViFMs to generate pseudo-labels, any inherent biases in these models may be reflected in the learned reward function and, consequently, in the resulting policy. While this does not diminish the effectiveness of our approach, it suggests that careful evaluation of learned policies remains important before deployment in safety-critical applications.

References

Pieter Abbeel and Andrew Y Ng. Apprenticeship learning via inverse reinforcement learning. In *International Conference on Machine learning (ICML)*, 2004.

Ademi Adeniji, Amber Xie, Carmelo Sferrazza, Younggyo Seo, Stephen James, and Pieter Abbeel. Language reward modulation for pretraining reinforcement learning. *arXiv:2308.12270*, 2023.

Rishabh Agarwal, Max Schwarzer, Pablo Samuel Castro, Aaron Courville, and Marc G Bellemare. Deep reinforcement learning at the edge of the statistical precipice. *Conference on Neural Information Processing Systems (NeurIPS)*, 2021.

Gaon An, Junhyeok Lee, Xingdong Zuo, Norio Kosaka, Kyung-Min Kim, and Hyun Oh Song. Direct preference-based policy optimization without reward modeling. *Conference on Neural Information Processing Systems (NeurIPS)*, 2023.

Erdem Bıyık, Nicolas Huynh, Mykel J Kochenderfer, and Dorsa Sadigh. Active preference-based gaussian process regression for reward learning. In *Proceedings ofRobotics: Science and Systems (RSS)*, 2020.

Ralph Allan Bradley and Milton E Terry. Rank analysis of incomplete block designs: I. the method of paired comparisons. *Biometrika*, 1952.

Stephen Casper, Xander Davies, Claudia Shi, Thomas Krendl Gilbert, Jeremy Scheurer, Javier Rando, Rachel Freedman, Tomasz Korbak, David Lindner, Pedro Freire, et al. Open problems and fundamental limitations of reinforcement learning from human feedback. *Transactions on Machine Learning Research (TMLR)*, 2023.

Lili Chen, Kevin Lu, Aravind Rajeswaran, Kimin Lee, Aditya Grover, Misha Laskin, Pieter Abbeel, Aravind Srinivas, and Igor Mordatch. Decision transformer: Reinforcement learning via sequence modeling. *Advances in neural information processing systems*, 2021.

Mingcai Chen, Yuntao Du, Yi Zhang, Shuwei Qian, and Chongjun Wang. Semi-supervised learning with multi-head co-training. In *Proceedings of the AAAI conference on artificial intelligence (AAAI)*, 2022a.

Yuanpei Chen, Tianhao Wu, Shengjie Wang, Xidong Feng, Jiechuan Jiang, Zongqing Lu, Stephen McAleer, Hao Dong, Song-Chun Zhu, and Yaodong Yang. Towards human-level bimanual dexterous manipulation with reinforcement learning. In *Conference on Neural Information Processing Systems (NeurIPS)*, 2022b.

Jie Cheng, Gang Xiong, Xingyuan Dai, Qinghai Miao, Yisheng Lv, and Fei-Yue Wang. Rime: Robust preference-based reinforcement learning with noisy preferences. *ICML*, 2024.

Heewoong Choi, Sangwon Jung, Hongjoon Ahn, and Taesup Moon. Listwise reward estimation for offline preference-based reinforcement learning. *International Conference on Machine Learning (ICML)*, 2024.

Paul F Christiano, Jan Leike, Tom Brown, Miljan Martic, Shane Legg, and Dario Amodei. Deep reinforcement learning from human preferences. In *Conference on Neural Information Processing Systems (NeurIPS)*, 2017.

Jack Clark and Dario Amodei. Faulty reward functions in the wild. https://openai.com/index/faulty-reward-functions/, 2016.

Nicolas Courty, Rémi Flamary, Devis Tuia, and Alain Rakotomamonjy. Optimal transport for domain adaptation. *IEEE transactions on pattern analysis and machine intelligence*, 2016.

Yuchen Cui, Scott Niekum, Abhinav Gupta, Vikash Kumar, and Aravind Rajeswaran. Can foundation models perform zero-shot task specification for robot manipulation? In *Learning for Dynamics and Control Conference (L4DC)*, 2022.

M Cuturi. Lightspeed computation of optimal transportation distances. *Conference on Neural Information Processing Systems (NeurIPS)*, 2013.

Arnaud Fickinger, Samuel Cohen, Stuart Russell, and Brandon Amos. Cross-domain imitation learning via optimal transport. *International Conference on Learning Representations (ICLR)*, 2022.

Rémi Flamary, Nicolas Courty, Alexandre Gramfort, Mokhtar Z. Alaya, Aurélie Boisbunon, Stanislas Chambon, Laetitia Chapel, Adrien Corenflos, Kilian Fatras, Nemo Fournier, Léo Gautheron, Nathalie T.H. Gayraud, Hicham Janati, Alain Rakotomamonjy, Ievgen Redko, Antoine Rolet, Antony Schutz, Vivien Seguy, Danica J. Sutherland, Romain Tavenard, Alexander Tong, and Titouan Vayer. Pot: Python optimal transport. *Journal of Machine Learning Research*, 2021.

Justin Fu, Anoop Korattikara, Sergey Levine, and Sergio Guadarrama. From language to goals: Inverse reinforcement learning for vision-based instruction following. *arXiv:1902.07742*, 2019.

Justin Fu, Aviral Kumar, Ofir Nachum, George Tucker, and Sergey Levine. D4rl: Datasets for deep data-driven reinforcement learning. *arXiv:2004.07219*, 2020.

Yuwei Fu, Haichao Zhang, Di Wu, Wei Xu, and Benoit Boulet. Robot policy learning with temporal optimal transport reward. In *Conference on Neural Information Processing Systems (NeurIPS)*, 2024.

Abhishek Gupta, Clemens Eppner, Sergey Levine, and Pieter Abbeel. Learning dexterous manipulation for a soft robotic hand from human demonstrations. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2016.

Tuomas Haarnoja, Aurick Zhou, Kristian Hartikainen, George Tucker, Sehoon Ha, Jie Tan, Vikash Kumar, Henry Zhu, Abhishek Gupta, Pieter Abbeel, et al. Soft actor-critic algorithms and applications. *arXiv:1812.05905*, 2018.

Joey Hejna and Dorsa Sadigh. Inverse preference learning: Preference-based rl without a reward function. *Conference on Neural Information Processing Systems (NeurIPS)*, 2023.

Joey Hejna, Rafael Rafailov, Harshit Sikchi, Chelsea Finn, Scott Niekum, W Bradley Knox, and Dorsa Sadigh. Contrastive preference learning: learning from human feedback without rl. *International Conference on Learning Representations (ICLR)*, 2024.

Donald Joseph Hejna III and Dorsa Sadigh. Few-shot preference learning for human-in-the-loop rl. In *Conference on Robot Learning*. PMLR, 2023.

William Huey, Huaxiaoyue Wang, Anne Wu, Yoav Artzi, and Sanjiban Choudhury. Imitation learning from a single temporally misaligned video. *ICML*, 2025.

Minyoung Hwang, Gunmin Lee, Hogun Kee, Chan Woo Kim, Kyungjae Lee, and Songhwai Oh. Sequential preference ranking for efficient reinforcement learning from human feedback. *Advances in Neural Information Processing Systems*, 2023.

Borja Ibarz, Jan Leike, Tobias Pohlen, Geoffrey Irving, Shane Legg, and Dario Amodei. Reward learning from human preferences and demonstrations in atari. In *Conference on Neural Information Processing Systems (NeurIPS)*, 2018.

Hyungkyu Kang and Min-hwan Oh. Adversarial policy optimization for offline preference-based reinforcement learning. *ICLR*, 2025.

Timo Kaufmann, Paul Weng, Viktor Bengs, and Eyke Hüllermeier. A survey of reinforcement learning from human feedback. *arXiv:2312.14925*, 2024.

Changyeon Kim, Jongjin Park, Jinwoo Shin, Honglak Lee, Pieter Abbeel, and Kimin Lee. Preference transformer: Modeling human preferences using transformers for rl. *International Conference on Learning Representations (ICLR)*, 2023.

Alexander Koenig, Zixi Liu, Lucas Janson, and Robert Howe. The role of tactile sensing in learning and deploying grasp refinement algorithms. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2022.

Ilya Kostrikov, Ashvin Nair, and Sergey Levine. Offline reinforcement learning with implicit q-learning. *International Conference on Learning Representations (ICLR)*, 2022.

Kimin Lee, Laura Smith, and Pieter Abbeel. Pebble: Feedback-efficient interactive reinforcement learning via relabeling experience and unsupervised pre-training. *ICML*, 2021a.

Kimin Lee, Laura Smith, Anca Dragan, and Pieter Abbeel. B-pref: Benchmarking preference-based reinforcement learning. *Conference on Neural Information Processing Systems (NeurIPS)*, 2021b.

Anqi Li, Dipendra Misra, Andrey Kolobov, and Ching-An Cheng. Survival instinct in offline reinforcement learning. *Conference on Neural Information Processing Systems (NeurIPS)*, 2023.

Xinran Liang, Katherine Shu, Kimin Lee, and Pieter Abbeel. Reward uncertainty for exploration in preference-based reinforcement learning. *ICLR*, 2022.

Runze Liu, Yali Du, Fengshuo Bai, Jiafei Lyu, and Xiu Li. Pearl: Zero-shot cross-task preference alignment and robust reward learning for robotic manipulation, 2024.

Yicheng Luo, Zhengyao Jiang, Samuel Cohen, Edward Grefenstette, and Marc Peter Deisenroth. Optimal transport for offline imitation learning. *ICLR*, 2023.

Tung M Luu, Donghoon Lee, Younghwan Lee, and Chang D Yoo. Policy learning from large vision-language model feedback without reward modeling. *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2025a.

Tung Minh Luu, Younghwan Lee, Donghoon Lee, Sunho Kim, Min Jun Kim, and Chang D Yoo. Enhancing rating-based reinforcement learning to effectively leverage feedback from large vision-language models. *ICML*, 2025b.

Neelu Madan, Andreas Møgelmose, Rajat Modi, Yogesh S Rawat, and Thomas B Moeslund. Foundation models for video understanding: A survey. *arXiv:2405.03770*, 2024.

Ajay Mandlekar, Danfei Xu, Josiah Wong, Soroush Nasiriany, Chen Wang, Rohun Kulkarni, Li Fei-Fei, Silvio Savarese, Yuke Zhu, and Roberto Martín-Martín. What matters in learning from offline human demonstrations for robot manipulation. In *Conference on Robot Learning (CoRL)*, 2021.

Daniel Marta, Simon Holk, Christian Pek, and Iolanda Leite. Sequel: Semi-supervised preference-based rl with query synthesis via latent interpolation. In *IEEE International Conference on Robotics and Automation (ICRA)*, 2024.

Antoine Miech, Dimitri Zhukov, Jean-Baptiste Alayrac, Makarand Tapaswi, Ivan Laptev, and Josef Sivic. Howto100m: Learning a text-video embedding by watching hundred million narrated video clips. In *International Conference on Computer Vision (ICCV)*, 2019.

Antoine Miech, Jean-Baptiste Alayrac, Lucas Smaira, Ivan Laptev, Josef Sivic, and Andrew Zisserman. End-to-end learning of visual representations from uncurated instructional videos. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2020.

Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Andrei A Rusu, Joel Veness, Marc G Bellemare, Alex Graves, Martin Riedmiller, Andreas K Fidjeland, Georg Ostrovski, et al. Human-level control through deep reinforcement learning. *Nature*, 2015.

Ni Mu, Hao Hu, Xiao Hu, Yiqin Yang, Bo Xu, and Qing-Shan Jia. Clarify: Contrastive preference reinforcement learning for untangling ambiguous queries. *International Conference on Machine Learning (ICML)*, 2025.

Calarina Muslimani and Matthew E Taylor. Leveraging sub-optimal data for human-in-the-loop reinforcement learning. *ICLR*, 2025.

Suraj Nair, Aravind Rajeswaran, Vikash Kumar, Chelsea Finn, and Abhinav Gupta. R3m: A universal visual representation for robot manipulation. *Conference on Robot Learning (CoRL)*, 2022.

Jongjin Park, Younggyo Seo, Jinwoo Shin, Honglak Lee, Pieter Abbeel, and Kimin Lee. Surf: Semi-supervised reward learning with data augmentation for feedback-efficient preference-based reinforcement learning. *International Conference on Learning Representations (ICLR)*, 2022.

Gabriel Peyré, Marco Cuturi, et al. Computational optimal transport: With applications to data science. *Foundations and Trends® in Machine Learning*, 2019.

Ivaylo Popov, Nicolas Heess, Timothy Lillicrap, Roland Hafner, Gabriel Barth-Maron, Matej Vecerik, Thomas Lampe, Yuval Tassa, Tom Erez, and Martin Riedmiller. Data-efficient deep reinforcement learning for dexterous manipulation. *arXiv:1704.03073*, 2017.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International Conference on Machine Learning (ICML)*, 2021.

Gathika Ratnayaka, James Nichols, and Qing Wang. Learning partial graph matching via optimal partial transport. *ICLR*, 2025.

Juan Rocamonde, Victoriano Montesinos, Elvis Nava, Ethan Perez, and David Lindner. Vision-language models are zero-shot reward models for reinforcement learning. *International Conference on Learning Representations (ICLR)*, 2024.

Daniel Shin, Anca D Dragan, and Daniel S Brown. Benchmarks and algorithms for offline preference-based reward learning. *Transactions on Machine Learning Research*, 2023.

David Silver, Julian Schrittwieser, Karen Simonyan, Ioannis Antonoglou, Aja Huang, Arthur Guez, Thomas Hubert, Lucas Baker, Matthew Lai, Adrian Bolton, et al. Mastering the game of go without human knowledge. *Nature*, 2017.

Joar Skalse, Nikolaus Howe, Dmitrii Krasheninnikov, and David Krueger. Defining and characterizing reward gaming. In *Conference on Neural Information Processing Systems (NeurIPS)*, 2022.

Sumedh Sontakke, Jesse Zhang, Séb Arnold, Karl Pertsch, Erdem Bıyık, Dorsa Sadigh, Chelsea Finn, and Laurent Itti. Roboclip: One demonstration is enough to learn robot policies. In *Conference on Neural Information Processing Systems (NeurIPS)*, 2024.

Kai Sheng Tai, Peter D Bailis, and Gregory Valiant. Sinkhorn label allocation: Semi-supervised classification via annealed self-training. In *International conference on machine learning*, pp. 10065–10075. PMLR, 2021.

Zhiquan Tan, Kaipeng Zheng, and Weiran Huang. Otmatch: Improving semi-supervised learning with optimal transport. *International Conference on Machine Learning (ICML)*, 2024.

Ran Tian, Chenfeng Xu, Masayoshi Tomizuka, Jitendra Malik, and Andrea Bajcsy. What matters to you? towards visual representation alignment for robot learning. *International Conference on Learning Representations (ICLR)*, 2024.

Vayer Titouan, Nicolas Courty, Romain Tavenard, and Rémi Flamary. Optimal transport for structured data with application on graphs. In *International Conference on Machine Learning*, pp. 6275–6284. PMLR, 2019.

Songjun Tu, Jingbo Sun, Qichao Zhang, Yaocheng Zhang, Jia Liu, Ke Chen, and Dongbin Zhao. In-dataset trajectory return regularization for offline preference-based reinforcement learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 2025.

Sreyas Venkataraman, Yufei Wang, Ziyu Wang, Navin Sriram Ravie, Zackory Erickson, and David Held. Real-world offline reinforcement learning from vision language model feedback. *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2025.

Ruiqi Wang, Weizheng Wang, and Byung-Cheol Min. Feedback-efficient active preference learning for socially aware robot navigation. In *2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2022a.

Yi Wang, Kunchang Li, Yizhuo Li, Yinan He, Bingkun Huang, Zhiyu Zhao, Hongjie Zhang, Jilan Xu, Yi Liu, Zun Wang, et al. Internvideo: General video foundation models via generative and discriminative learning. *arXiv:2212.03191*, 2022b.

Yufei Wang, Zhanyi Sun, Jesse Zhang, Zhou Xian, Erdem Biyik, David Held, and Zackory Erickson. Rl-vlm-f: Reinforcement learning from vision language foundation model feedback. *International Conference on Machine Learning (ICML)*, 2024.

Saining Xie, Chen Sun, Jonathan Huang, Zhuowen Tu, and Kevin Murphy. Rethinking spatiotemporal feature learning: Speed-accuracy trade-offs in video classification. In *European Conference on Computer Vision (ECCV)*, 2018.

Hu Xu, Gargi Ghosh, Po-Yao Huang, Dmytro Okhonko, Armen Aghajanyan, Florian Metze, Luke Zettlemoyer, and Christoph Feichtenhofer. Videoclip: Contrastive pre-training for zero-shot video-text understanding. *EMNLP*, 2021.

Tianhe Yu, Deirdre Quillen, Zhanpeng He, Ryan Julian, Karol Hausman, Chelsea Finn, and Sergey Levine. Meta-world: A benchmark and evaluation for multi-task and meta reinforcement learning. In *Conference on Robot Learning (CoRL)*, 2020.

Yifu Yuan, Jianye Hao, Yi Ma, Zibin Dong, Hebin Liang, Jinyi Liu, Zhixin Feng, Kai Zhao, and Yan Zheng. Uni-rlhf: Universal platform and benchmark suite for reinforcement learning with diverse human feedback. *International Conference on Learning Representations (ICLR)*, 2024.

Zhecheng Yuan, Sizhe Yang, Pu Hua, Can Chang, Kaizhe Hu, and Huazhe Xu. Rl-vigen: A reinforcement learning benchmark for visual generalization. *Advances in Neural Information Processing Systems*, 2023.

Jesse Zhang, Jiahui Zhang, Karl Pertsch, Ziyi Liu, Xiang Ren, Minsuk Chang, Shao-Hua Sun, and Joseph J Lim. Bootstrap your own skills: Learning to solve new tasks with large language model guidance. *Conference on Robot Learning (CoRL)*, 2023.

Zhilong Zhang, Yihao Sun, Junyin Ye, Tian-Shuo Liu, Jiaji Zhang, and Yang Yu. Flow to better: Offline preference-based reinforcement learning via preferred trajectory generation. In *ICLR*, 2024.

Henry Zhu, Abhishek Gupta, Aravind Rajeswaran, Sergey Levine, and Vikash Kumar. Dexterous manipulation with deep reinforcement learning: Efficient, general, and low-cost. In *2019 International Conference on Robotics and Automation (ICRA)*, pp. 3651–3657. IEEE, 2019.

Yuke Zhu, Josiah Wong, Ajay Mandlekar, Roberto Martín-Martín, Abhishek Joshi, Soroush Nasiriany, and Yifeng Zhu. robosuite: A modular simulation framework and benchmark for robot learning. *arXiv:2009.12293*, 2020.

APPENDIX

A DECLARATION OF LARGE LANGUAGE MODEL USAGE

We only used LLMs for minor editing tasks, including grammar correction and word polishing. They were not involved in research conception, experimentation, analysis, or substantive writing.

B DETAILS ON TASKS AND DATASETS

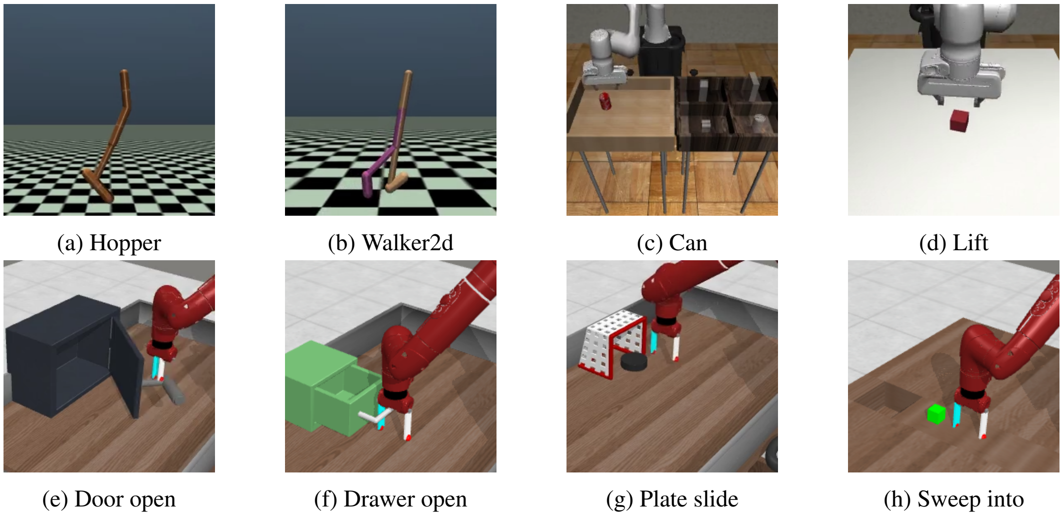


Figure 7: Overview of environments used in our experiments: Gym Locomotion (a–b), Robosuite Manipulation (c–d), and MetaWorld Manipulation (e–h).

B.1 TASK DETAILS

The locomotion tasks from D4RL (Fu et al., 2020) and the manipulation tasks from MetaWorld (Yu et al., 2020) and Robosuite (Zhu et al., 2020) used in our experiments are shown in Figure 7.

D4RL Gym Locomotion. In D4RL Gym locomotion tasks, the goal is to control simulated robots to move forward efficiently while minimizing energy costs for safe behavior. We use two tasks: Hopper and Walker2d, as in previous works (Kim et al., 2023; Hejna & Sadigh, 2023).

MetaWorld Manipulation. In this domain, the agent produces low-level continuous actions to control a simulated 7-DoF Rethinking Sawyer robotic arm, enabling interaction with tabletop objects to perform diverse manipulation tasks. Initial arm position is randomized. We evaluate four tasks:

- *Door Open*: Open the door of a safe.
- *Drawer Open*: Pull open a drawer.
- *Plate Slide*: Slide a black plate into the designated goal region.
- *Sweep Into*: Sweep a green puck into the squared hole.

Robosuite Manipulation. In this domain, similar to MetaWorld, the agent produces low-level continuous actions to control a simulated 7-DoF Franka Emika Panda robot. We evaluate two tasks:

- *Lift*: Lift a cube object.
- *Can*: Pick up a coke can from a table and place it into the target bin.

B.2 DATASET DETAILS

In offline preference-based RL, two types of data are provided: (i) an offline dataset collected from an unknown policy and (ii) a preference dataset consisting of pairs of trajectory segments sampled

Algorithm 1 Pseudo-code for Video-based Optimal Transport Preference (VOTP)

```
Input: Offline dataset $\mathcal{B}$, labeled dataset $\mathcal{D}_l$, number of unlabeled segments $M$, threshold $\tau_P$.
Initialize: pseudo-labeled dataset $\mathcal{D}_u \leftarrow \emptyset$.
for each iteration do
    Sample $\frac{M}{2}$ segment pairs from $\mathcal{B}$
    Compute preference scores for segment pairs using Eq. (6)
    Assign pseudo-labels using Eq. (8) and append to $\mathcal{D}_u$
end for
Construct preference dataset $\mathcal{D} \leftarrow \mathcal{D}_l \cup \mathcal{D}_u$
Train reward model $\widehat{r}_\psi$ using Eq. (2)
Relabel rewards for state-action pairs in $\mathcal{B}$ using trained $\widehat{r}_\psi$
Train policy $\pi_\theta$ using an offline RL algorithm
```

from the offline dataset. For D4RL Gym locomotion, we use *medium-expert-v2*—which mixes equal portions of expert and partially trained demonstrations—and *medium-replay-v2*, which corresponds to the replay buffer of a partially trained policy. For MetaWorld, we use the pre-collected dataset from Hejna et al. (2024). For Robosuite, we use the Robomimic dataset provided by Mandlekar et al. (2021). For the preference dataset, we use pair indices from the publicly available datasets of Kim et al. (2023) and Hejna et al. (2024). For preference labels, we use scripted labels obtained from the ground-truth reward functions, except for *hopper-medium-replay-v2*, where we use human labels. In Robomimic, we regenerate dense rewards by replaying the offline dataset in the simulator. Since the preference dataset from Kim et al. (2023) contains at most 500 preferences, we additionally generate pair indices for unlabeled data using the code from Kim et al. (2023).

B.3 Implementation Details

The hyperparameters used in our main experiments are shown in Table 3, 4, and 5.

Table 3: Hyperparameters of IQL.

Hyperparameter	**Locomotion**	**MetaWorld**	**Robomimic**
Optimizer	Adam	Adam	Adam
Learning rate	3e-4	3e-4	3e-4
Batch size	256	512	256
Hidden layer dim	256	256	256
Hidden layers	2	2	2
Activation	ReLU	ReLU	ReLU
Discount factor	0.99	0.99	0.99
β	3.0	10.0	10.0
τ	0.7	0.9	0.9
Training steps	1e6	4e5	1.5e6 (can-ph), 1e6 (others)

Table 4: Hyperparameters of the reward model.

Hyperparameter	**Locomotion**	**MetaWorld**	**Robomimic**
Optimizer	Adam	Adam	Adam
Learning rate	3e-4	3e-4	3e-4
Batch size	8	32	8
Hidden layer dim	256	128	256
Hidden layers	2	2	2
Activation	ReLU	LeakyReLU	ReLU
Output activation	Identity	Tanh	Identity
Segment length	100	64	50 (ph), 100 (mh)
Subsample length	64	42	32 (ph), 64 (mh)
Training steps	2e4	2e4	2e4

Table 5: Hyperparameters of VOTP

Hyperparameter	Locomotion	MetaWorld	Robomimic
Total #labeled pairs	10	10	5 (ph), 10 (mh)
Total #unlabeled pairs	10k	50k	10k
M (in Alg. 1)	2	2	2
Distance metric in Eq. 6	Euclidean	Euclidean	Euclidean
Preference threshold τ_P	0.15 (hopper-*) 0.2 (walker2d-*)	0.45 (door, sweep) 0.35 (drawer) 0.4 (plate)	0.2 (lift-mh) 0.15 (others)

B.4 Real Robot Experiment Setups

We evaluate our method on two vision-based manipulation tasks using a 7-DoF Rethink Sawyer robotic arm in a tabletop environment. The tasks probe both reaching and object interaction and are defined as follows:

- *Lift banana*: grasp a banana from a plate and lift it.
- *Drawer open*: pull open a drawer beyond a fixed distance.

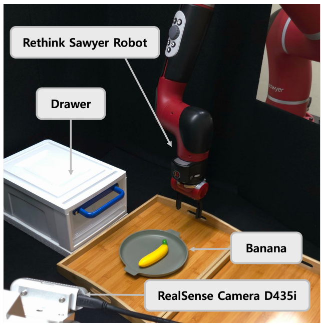


Figure 8: Environment setup in our real robot.

The robot is controlled with end-effector (EE) delta actions that command Cartesian displacements of the gripper. The EE orientation is constrained to yaw only, and control runs at 10Hz. For each task, the initial poses of a banana or a drawer handle are randomized within the workspace and observed by an Intel RealSense D435i RGB camera, which can be found in Figure 8). We collect 40 episodes for *lift banana* and 50 episodes for *drawer open* via keyboard teleoperation. Policies use both low-dimensional states and visual observations. The visual observation is an RGB image at 480 × 480 resolution, resized to 224 × 224. We use ViFM to produce a 512-dimensional visual feature. The low-dimensional state is a 9-dimensional vector comprising the EE Cartesian position (3 dimensions), linear velocity (3 dimensions), yaw orientation (1 dimension), and the gripper status encoded as one-hot (open or closed, 2 dimensions). We concatenate the visual feature and the low-dimensional states to form a 521-dimensional input to the policy. All hyperparameter settings for the real-robot experiments can be found at Table 6. For evaluation, we measure success rate over 10 episodes per task.

C Learning Curves

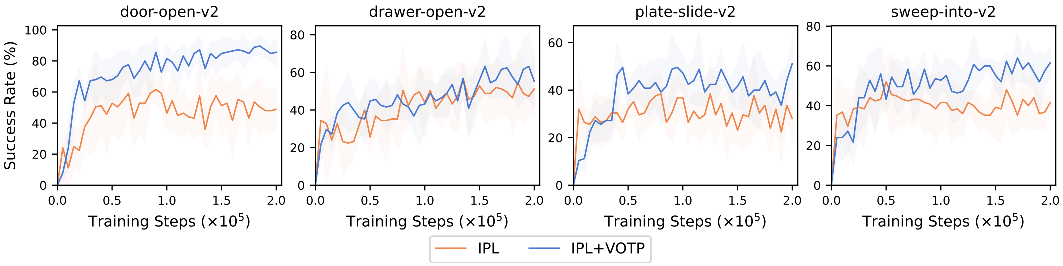


Figure 9: We further evaluate the ability of VOTP to enhance IPL (Hejna & Sadigh, 2023) on Meta-World by training policies directly from preferences (both labeled and pseudo-labeled). Results show mean over 5 runs with standard deviation (shaded). Results show that VOTP substantially improves IPL, demonstrating its potential to generate effective pseudo-preference labels even without explicit reward models. Results averaged over 5 runs.

Table 6: Hyperparameters of real robot experiments.

	Hyperparameter	**Value**
IQL	Optimizer	Adam
	Learning Rate	3e-4
	Batch Size	256
	Hidden layer dim	256
	Hidden layers	2
	Activation	ReLU
	Discount γ	0.99
	β	3.0
	Expectile τ	0.7
	Training Steps	$1e5$
Reward Model	Optimizer	Adam
	Learning rate	3e-4
	Batch size	8
	Hidden layer dim	128
	Hidden layers	2
	Activation	LeakyReLU
	Output activation	Tanh
	Segment length	16
	Training steps	2000
VOTP	Total #labeled pairs	5 (Lift Banana), 10 (Drawer Open)
	Total #unlabeled pairs	2000 (Lift Banana), 3000 (Drawer Open)
	Preference threshold τ_P	0.6

Table 7: Accuracy of generated pseudo-labels: We calculate accuracy by comparing against ground-truth scripted preference labels (excluding equally preferred pairs). Overall, VOTP generates high-quality pseudo-labels with only a handful of labeled preference queries.

Domain	Task	Accuracy (%)
D4RL Gym Locomotion	hopper-medium-expert-v2	90.3
	walker2d-medium-replay-v2	98.8
	walker2d-medium-expert-v2	93.6
MetaWorld Manipulation	door-open	93.1
	drawer-open	97.4
	plate-slide	95.2
	sweep-into	67.0
Robomimic Manipulation	can-mh	72.0
	can-ph	88.6
	lift-mh	87.1
	lift-ph	82.6

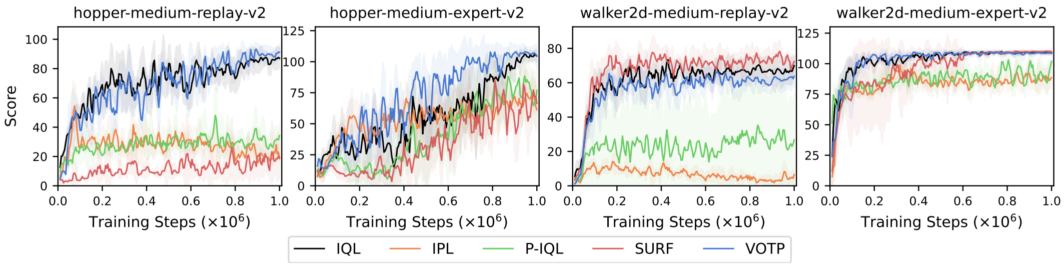


Figure 10: Full learning curves for the D4RL Gym locomotion tasks (Table 1). Results are means of 5 runs with standard deviation (shaded area). We smooth the learning curves using a moving average with a window size of 3.

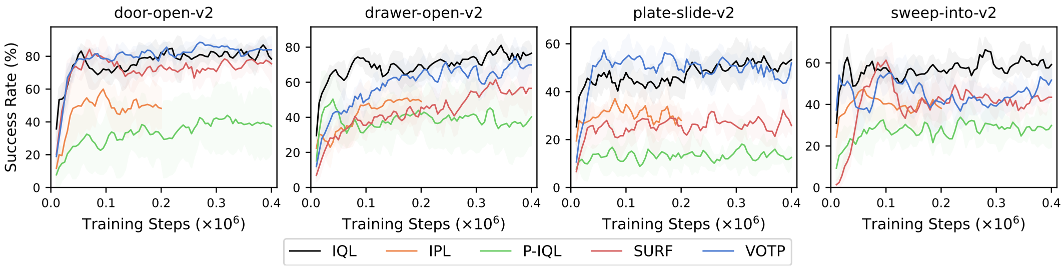


Figure 11: Full learning curves for the MetaWorld manipulation tasks (Table 1). Results are means of 5 runs with standard deviation (shaded area). We smooth the learning curves using a moving average with a window size of 3.

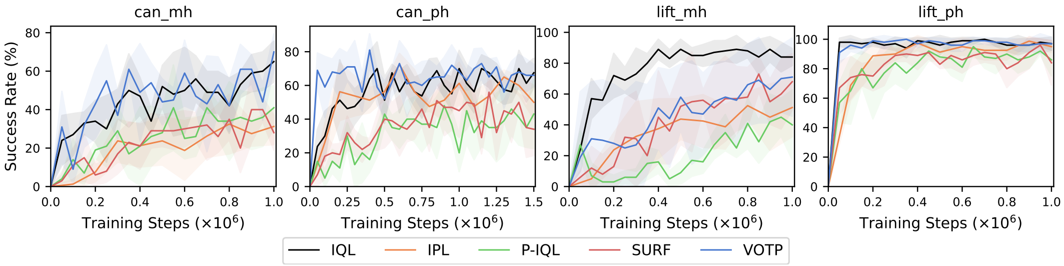
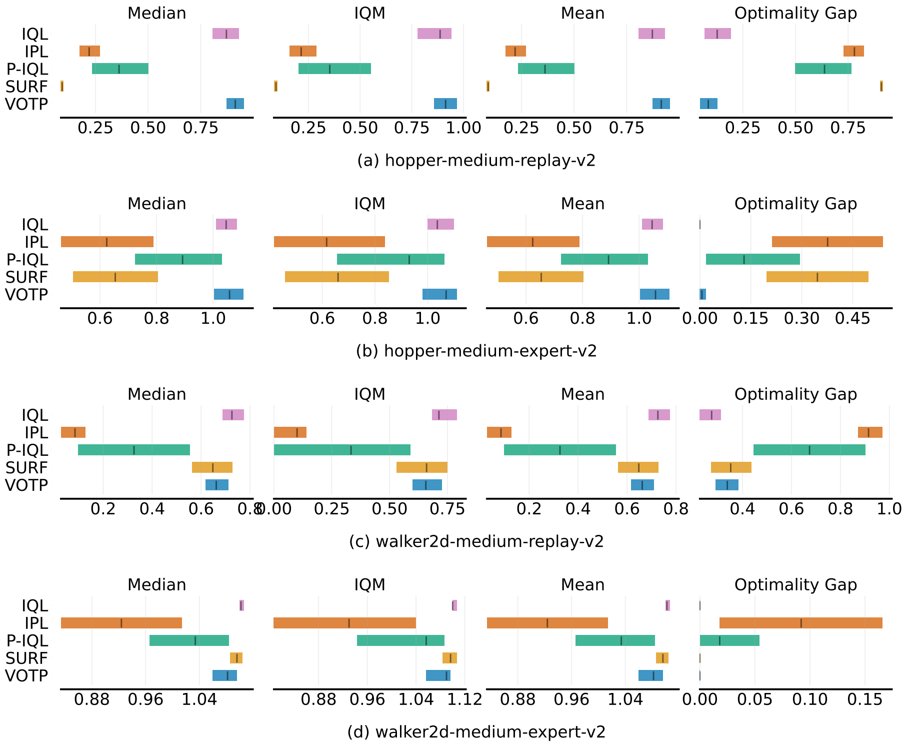


Figure 12: Full learning curves for the Robomimic manipulation tasks (Table 1). Results are means of 5 runs with standard deviation (shaded area).



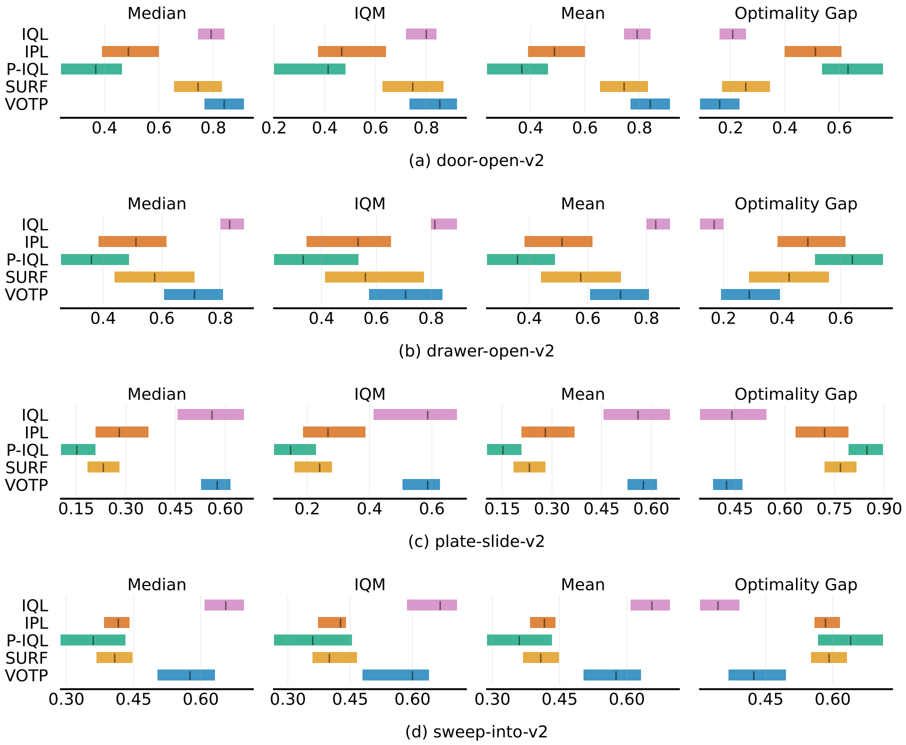
(a) hopper-medium-replay-v2

(b) hopper-medium-expert-v2

(c) walker2d-medium-replay-v2

(d) walker2d-medium-expert-v2

Figure 13: Aggregate metrics on D4RL Gym locomotion tasks with 95% confidence intervals (CIs) across five runs. Higher mean, median and IQM scores and lower optimality gap are better. The CIs are estimated using the percentile bootstrap with stratified sampling.



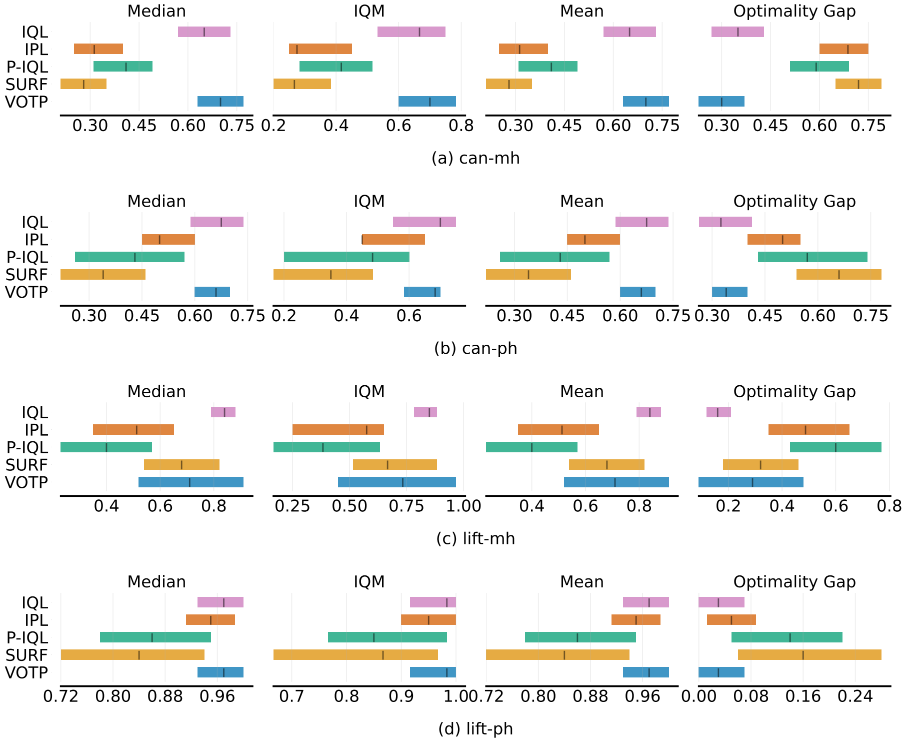
(a) door-open-v2

(b) drawer-open-v2

(c) plate-slide-v2

(d) sweep-into-v2

Figure 14: Aggregate metrics on MetaWorld manipulation tasks with 95% confidence intervals (CIs) across five runs. Higher mean, median and IQM scores and lower optimality gap are better. The CIs are estimated using the percentile bootstrap with stratified sampling.



(a) can-mh

(b) can-ph

(c) lift-mh

(d) lift-ph

Figure 15: Aggregate metrics on Robomimic manipulation tasks with 95% confidence intervals (CIs) across five runs. Higher mean, median and IQM scores and lower optimality gap are better. The CIs are estimated using the percentile bootstrap with stratified sampling.

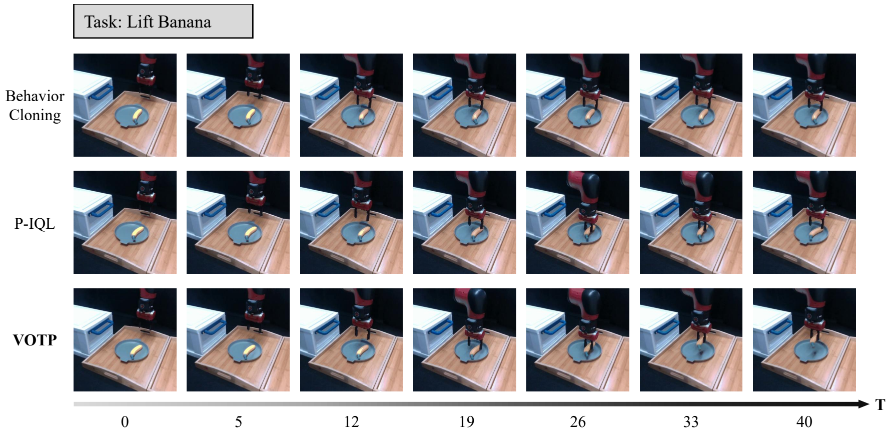


Figure 16: Snapshot of rollouts on *Lift Banana* task from BC, P-IQL and VOTP. Video of rollouts are provided in the Supplementary. The behavior cloning agent fails to descend to the banana and cannot grasp it. The P-IQL agent grasps the banana but does not lift and just release it. VOTP agent successfully reaches the banana, grasps it, and lifts it to a specified height.

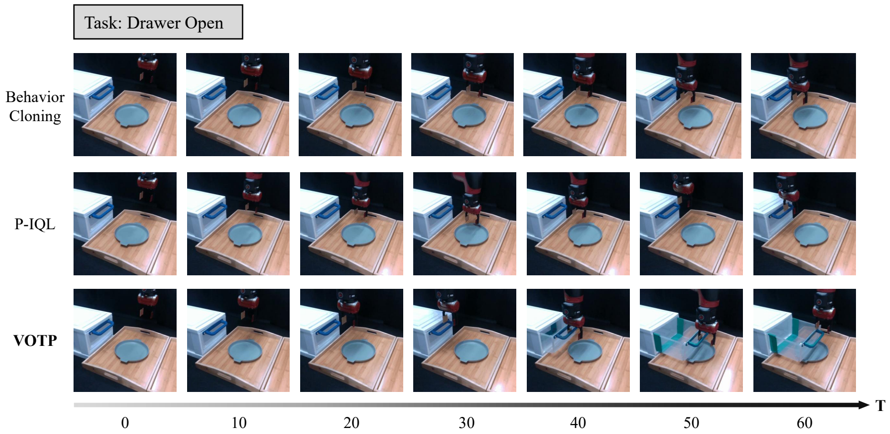


Figure 17: Snapshot of rollouts on *Drawer Open* task from BC, P-IQL and VOTP. Video of rollouts are provided in the Supplementary. In both behavior cloning and P-IQL, the agent barely reaches the handle after wandering and fails to pull the drawer open. VOTP agent, however, reaches the handle directly and pull it open successfully.

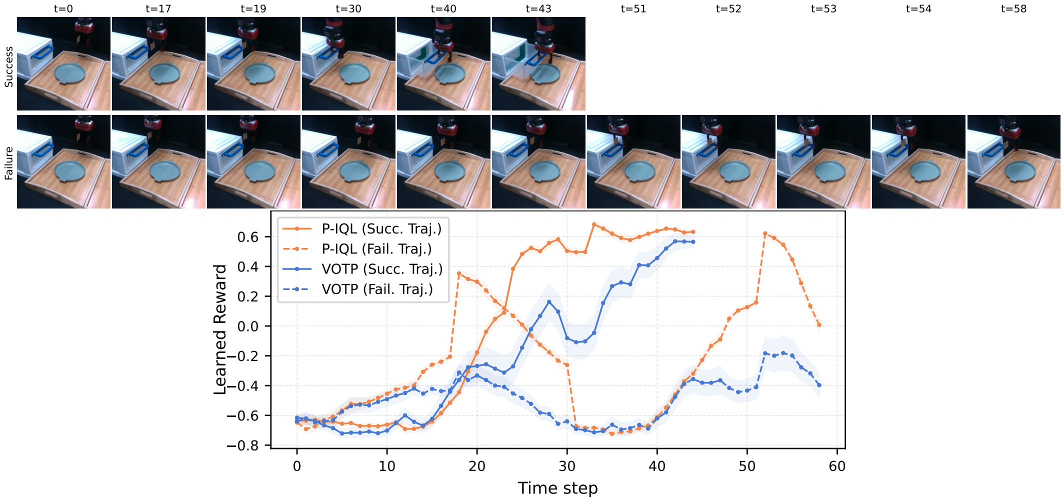


Figure 18: Drawer Open: Examples of successful and failed trajectories at each time step (top) with the corresponding reward outputs over timesteps from VOTP and P-IQL (bottom).

D Additional Results and Analysis

D.1 Ablation Study on Cost Metric

To examine the sensitivity of VOTP to the choice of cost function, we evaluate it using Euclidean and cosine distances when computing the optimal transport plan. Table 8 shows that VOTP performs robustly under both choices.

Table 8: Performance of VOTP with different cost functions.

Dataset	Euclidean	Cosine
hopper-medium-replay-v2	91.1 ± 4.7	87.0 ± 4.2
walker2d-medium-replay-v2	66.3 ± 5.6	64.1 ± 14.3
door-open	84.0 ± 8.4	82.0 ± 8.7
drawer-open	71.2 ± 11.7	75.8 ± 4.2
plate-slide	57.6 ± 5.4	57.0 ± 12.4
sweep-into	57.6 ± 7.4	58.0 ± 11.8
Average	71.3	70.7

D.2 Comparison with Oracle Preference Learning

We additionally conduct an experiment using P-IQL trained on all segment pairs with ground-truth preference labels, *i.e.*, 10k labels for D4RL Gym locomotion and Robomimic, and 50k labels for MetaWorld. This oracle effectively serves as an upper bound for our method. The results are shown in Table 9. As shown, VOTP can closely match the oracle performance on 8 out of 12 datasets.

Table 9: Comparison with Oracle (P-IQL trained on all pairs using ground-truth preferences). We bold all scores within 5% of the maximum per dataset ($\geq 0.95 \cdot \max$)

Dataset	VOTP	Oracle
hopper-medium-replay-v2	**91.1 ± 4.7**	**91.3 ± 3.2**
hopper-medium-expert-v2	**105.7 ± 6.0**	**101.9 ± 4.5**
walker2d-medium-replay-v2	**66.3 ± 5.6**	**66.9 ± 10.3**
walker2d-medium-expert-v2	**108.1 ± 2.2**	**109.6 ± 0.8**
locomotion avg.	92.8	92.4
door-open	84.0 ± 8.4	90.4 ± 3.2
drawer-open	71.2 ± 11.7	80.4 ± 7.4
plate-slide	57.6 ± 5.4	62.4 ± 4.8
sweep-into	**57.6 ± 7.4**	**59.0 ± 14.2**
metaworld avg.	67.6	73.1
can-mh	**70.0 ± 8.4**	**70.1 ± 6.5**
can-ph	**66.0 ± 5.8**	**66.7 ± 11.1**
lift-mh	71.0 ± 22.7	86.3 ± 5.8
lift-ph	**97.0 ± 4.0**	**97.2 ± 3.7**
robomimic avg.	76.0	80.1

D.3 RL Performance between GT Reward and Incorrect Rewards

For each dataset, we verify that RL performance differs when training with ground-truth (GT) rewards versus incorrect rewards. We use the three incorrect reward functions introduced in (Li et al., 2023): (1) Zero: all rewards are set to $r(s, a) = 0$; (2) Random: each transition takes a reward value sampled from a uniform distribution $U(0, 1)$; and (3) Negative: each transition's reward is replaced with the negation of the true reward, $-r(s, a)$. Following (Li et al., 2023), we report the performance of behavior cloning (BC), offline RL with GT rewards, and offline RL with incorrect rewards for each dataset in Table 10.

Table 10: Performance of IQL (Kostrikov et al., 2022) on each dataset under GT and incorrect rewards. Mean and standard deviation are computed over 5 random seeds.

Dataset	BC	GT	Zero	Random	Negative
hopper-medium-replay-v2	33.7 ± 8.5	87.5 ± 7.4	28.0 ± 8.3	48.2 ± 6.7	0.6 ± 0.0
hopper-medium-expert-v2	55.1 ± 2.8	104.5 ± 4.5	53.6 ± 1.4	67.8 ± 9.8	16.8 ± 5.5
walker2d-medium-replay-v2	19.2 ± 7.3	72.6 ± 4.9	19.4 ± 2.7	47.3 ± 13.6	0.1 ± 0.3
walker2d-medium-expert-v2	100.4 ± 13.4	109.9 ± 0.5	100.7 ± 7.7	69.7 ± 7.1	20.9 ± 0.7
door-open	57.5 ± 3.0	79.2 ± 5.9	59.6 ± 2.1	52.0 ± 2.8	39.0 ± 7.7
drawer-open	61.5 ± 3.7	83.2 ± 4.7	61.0 ± 1.7	59.8 ± 3.3	46.8 ± 5.5
plate-slide	39.1 ± 2.5	56 ± 11.9	38.4 ± 2.0	34.0 ± 10.8	24.0 ± 5.7
sweep-into	49.3 ± 2.1	65.6 ± 5.4	46.6 ± 2.6	48.8 ± 4.7	27.0 ± 9.5

D.4 Offline Preference-Based RL Results with more Baselines

Our offline RL experiments in the main paper focus on baselines that leverage semi-supervised learning to improve feedback efficiency. Here, we further compare VOTP with recent methods that also aim to enhance feedback efficiency in offline PbRL. Specifically, we evaluate six baselines: CPL (Hejna et al., 2024), DPPO (An et al., 2023), LiRE (Choi et al., 2024), APPO (Kang & Oh, 2025), FTB (Zhang et al., 2024), and DTR (Tu et al., 2025). For each baseline, we use the official implementations provided in their publicly released repositories, as listed in Table 11. Since prior algorithms are evaluated on either D4RL locomotion or MetaWorld, and some on both, we report performance on these two benchmarks accordingly. We obtain their performance by re-running the author-provided implementations under the same conditions as ours (*e.g.*, same number of labels) and conducting hyperparameter searches within the recommended ranges for a fair comparison. For FTB, we use the default hyperparameters, as training the method takes approximately two days. In addition, DTR uses the Decision Transformer (Chen et al., 2021) as its policy architecture, whereas the other methods use an MLP. We report the average performance over 25 rollouts using the final checkpoint and run all experiments with 5 random seeds.

The results presented in Table 12 show that LiRE achieves performance competitive with ours on average. However, LiRE improves feedback efficiency by exploiting second-order information from ranked lists, which is orthogonal to the pseudo-labeling mechanism of VOTP. Also, our approach is complementary and can be integrated into LiRE to potentially further enhance its efficiency.

Table 11: Source code links and hyperparameter/variant search settings for all baselines.

Algorithm	URL	Hyperparameters Tuning
CPL	`https://github.com/jhejna/cpl`	with/without BC
DPPO	`https://github.com/snu-mllab/DPPO`	$\lambda \in \{0.1, 0.5\}$, smooth $m \in \{5, 10, 15\}$
LiRE	`https://github.com/chwoong/LiRE`	Budget $Q \in \{2, 4, 5, 10\}$
APPO	`https://github.com/oh-lab/APPO`	$\lambda \in \{0.01, 0.3, 0.1\}$
FTB	`https://github.com/Zzl35/flow-to-better`	Default
DTR	`https://github.com/TU2021/DTR`	Coefficient $\eta_{max} \in \{0.1, 1, 3\}$

Table 12: Average performance of all baselines on D4RL locomotion and MetaWorld. For D4RL tasks, "hop" denotes hopper, while "m", "r", and "e" denote medium, replay, and expert, respectively. All methods are re-run under the same conditions. We bold all scores within 5% of the maximum per dataset ($\geq 0.95 \cdot \max$).

Dataset	CPL	DPPO	LiRE	APPO	FTB	DTR	VOTP (Ours)
hop-m-r	6.5 ± 0.5	59.4 ± 12.0	52.1 ± 26.9	1.3 ± 0.6	**90.5 ± 3.9**	30.0 ± 34.2	**91.1 ± 4.7**
hop-m-e	54.2 ± 7.0	70.1 ± 20.2	106.3 ± 7.2	40.7 ± 16.7	**111.9 ± 0.7**	15.9 ± 5.3	105.7 ± 6
walk-m-r	6.6 ± 0.6	35.1 ± 6.9	71.3 ± 16.0	12.8 ± 8.1	62.8 ± 8.8	**84.5 ± 14.9**	66.3 ± 5.6
walk-m-e	39.9 ± 19.9	70.2 ± 12.4	103.2 ± 15.1	31.2 ± 6.2	76.5 ± 2.2	72.4 ± 39.2	**108.1 ± 2.2**
loco avg.	26.8	58.7	83.2	21.5	85.4	50.7	92.8
door-open	44.8 ± 22.3	14.4 ± 12.1	**84.0 ± 4.9**	73.6 ± 7.8	43.2 ± 6.6	41.6 ± 10.0	**84.0 ± 8.4**
drawer-open	52.8 ± 13.9	59.0 ± 12.4	70.0 ± 6.3	64.8 ± 14.8	52.8 ± 7.2	**88.8 ± 5.2**	71.2 ± 11.7
plate-slide	34.4 ± 12.0	37.1 ± 17.1	38.0 ± 7.5	24.0 ± 17.2	41.6 ± 3.6	39.2 ± 6.6	**57.6 ± 5.4**
sweep-into	37.6 ± 6.0	48.0 ± 14.8	**64.0 ± 10.2**	44.0 ± 16.2	56.8 ± 13.7	48.8 ± 5.9	57.6 ± 7.4
mw avg.	42.4	39.6	64.0	51.6	48.6	54.6	67.6

D.5 Computational Cost for Pseudo-Preference Genration

We train VOTP on an RTX 4090 GPU with 24 CPU cores (AMD Ryzen Threadripper 7960X). Feature extraction takes approximately 20 minutes for 50k segments. The time required for generating pseudo-preference labels depends on the number of labels, as shown in Table 13. Policy training is implemented using IPL's source code [3] and takes roughly 1.5 hours for each run.

Table 13: Computational cost of VOTP for generating pseudo-preference labels for 10k unlabeled pairs under different dataset sizes.

N labels	10	25	50	100	200	500
Time (minutes)	0.15	0.2	0.75	3	12	60

D.6 Preference Learning from Human Teachers

We further evaluate VOTP using real human preference feedback across six datasets, with results summarized in Table 14. For D4RL locomotion, we use the human preference labels provided by Kim et al. (2023). For MetaWorld, we collect preferences from four non-robotic participants across four datasets. We observe a slight performance drop in *walker2d* but largely stable performance in the remaining environments.

Table 14: Performance of VOTP with preference feedbacks from human teachers. Mean and standard deviation are computed over 5 random seeds.

Dataset	Scripted Teacher	Human Teacher
hopper-medium-expert-v2	105.7 ± 6.0	109.3 ± 1.7
walker2d-medium-replay-v2	66.3 ± 5.6	59.4 ± 8.8
walker2d-medium-expert-v2	108.1 ± 2.2	90.8 ± 7.0
door-open	84.0 ± 8.4	85.6 ± 8.6
drawer-open	71.2 ± 11.7	70.4 ± 6.5
plate-slide	57.6 ± 5.4	55.9 ± 11.3
sweep-into	57.6 ± 5.4	56.8 ± 11.3
Average	78.6	75.5

D.7 Online PbRL

Figure 19 shows the experimental results for online PbRL, comparing VOTP against PEBBLE (Lee et al., 2021b). In VOTP, we also collect unlabeled pairs during the query phase, and the reward function is trained in the same way as PEBBLE during this stage. After reaching the query budget (*e.g.*, $N = 100$), we generate pseudo-labels and perform an additional round of reward training using both labeled and unlabeled data, after which we train the policy normally with the learned reward function. The results show the potential of using VOTP in online PbRL setting.

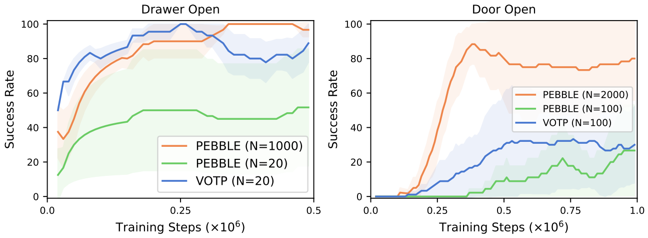
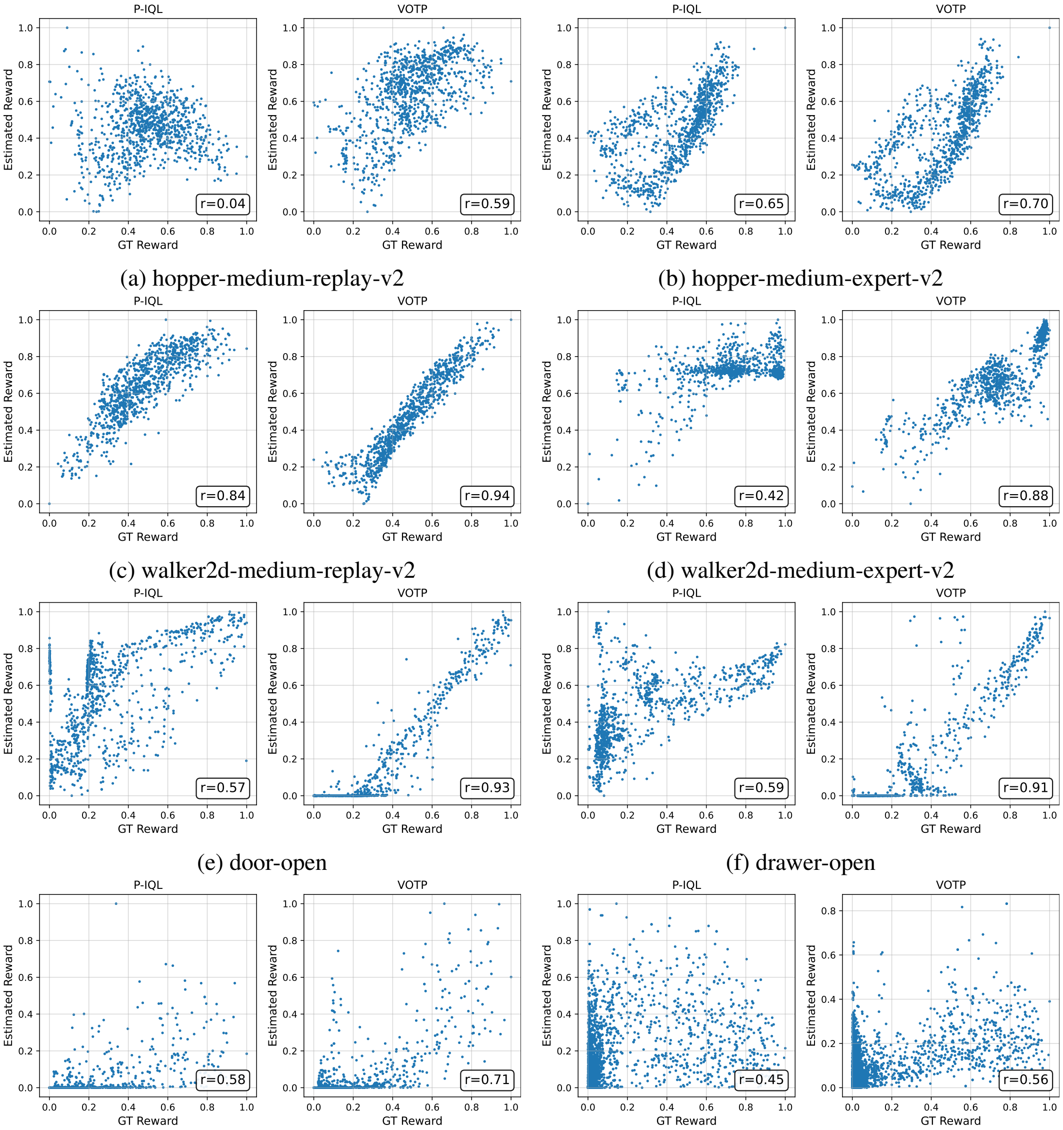


Figure 19: Online experiments on two MetaWorld environments. N denotes the number of queries used for reward training. Results are means of 3 seeds with standard deviation (shaded area).

[3] https://github.com/jhejna/inverse-preference-learning

D.8 Comparison of Learned Rewards with Ground-truth Rewards

We examine the reward values estimated by the learned reward models of P-IQL and VOTP. Figure 20 shows scatter plots of the estimated rewards of segments (y-axis) against the GT rewards (x-axis) for each dataset. As shown, the reward estimates produced by VOTP exhibit a significantly stronger correlation with GT rewards compared to those of P-IQL.



(a) hopper-medium-replay-v2

(b) hopper-medium-expert-v2

(c) walker2d-medium-replay-v2

(d) walker2d-medium-expert-v2

(e) door-open

(f) drawer-open

(g) plate-slide

(h) sweep-into

Figure 20: Correlation between learned rewards and ground-truth rewards for P-IQL and VOTP. Pearson correlation coefficients (r) are shown for each dataset.

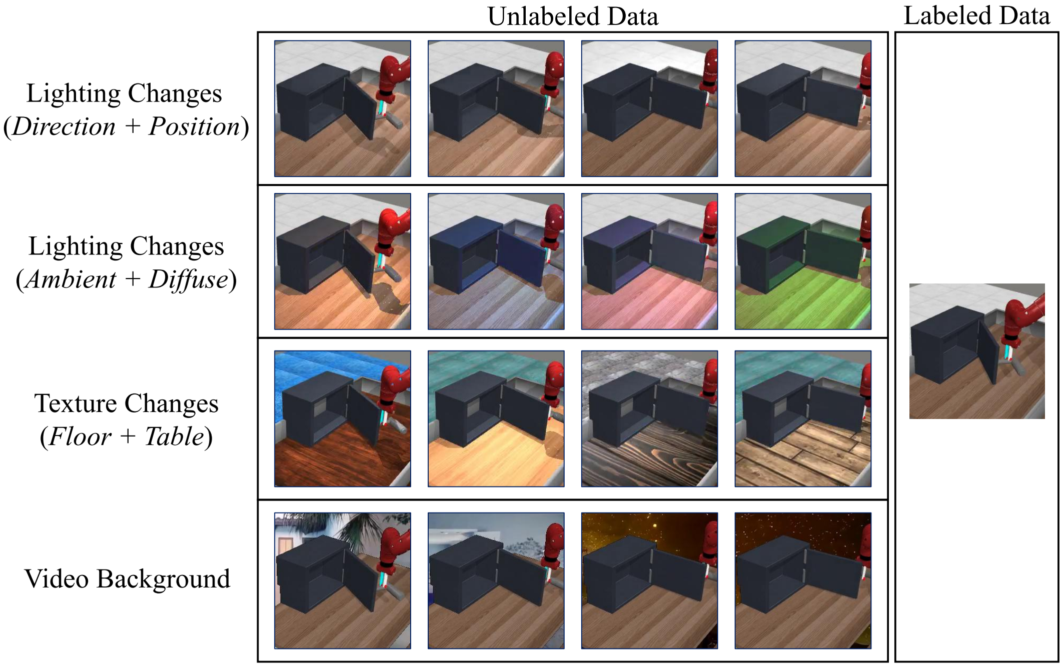

Unlabeled Data
Labeled Data
Lighting Changes (*Direction* + *Position*)
Lighting Changes (*Ambient* + *Diffuse*)
Texture Changes (*Floor* + *Table*)
Video Background

Figure 21: Examples of scenarios with different types of visual distractions. For lighting changes, we consider two distractors: (1) variations in the direction and position of the light source, and (2) variations in ambient and diffuse lighting. For texture changes, we randomize the textures of the table and floor (20 possible combinations). For video background distractions, we use both the easy and hard videos from Yuan et al. (2023). Example segment clips are available at this link.

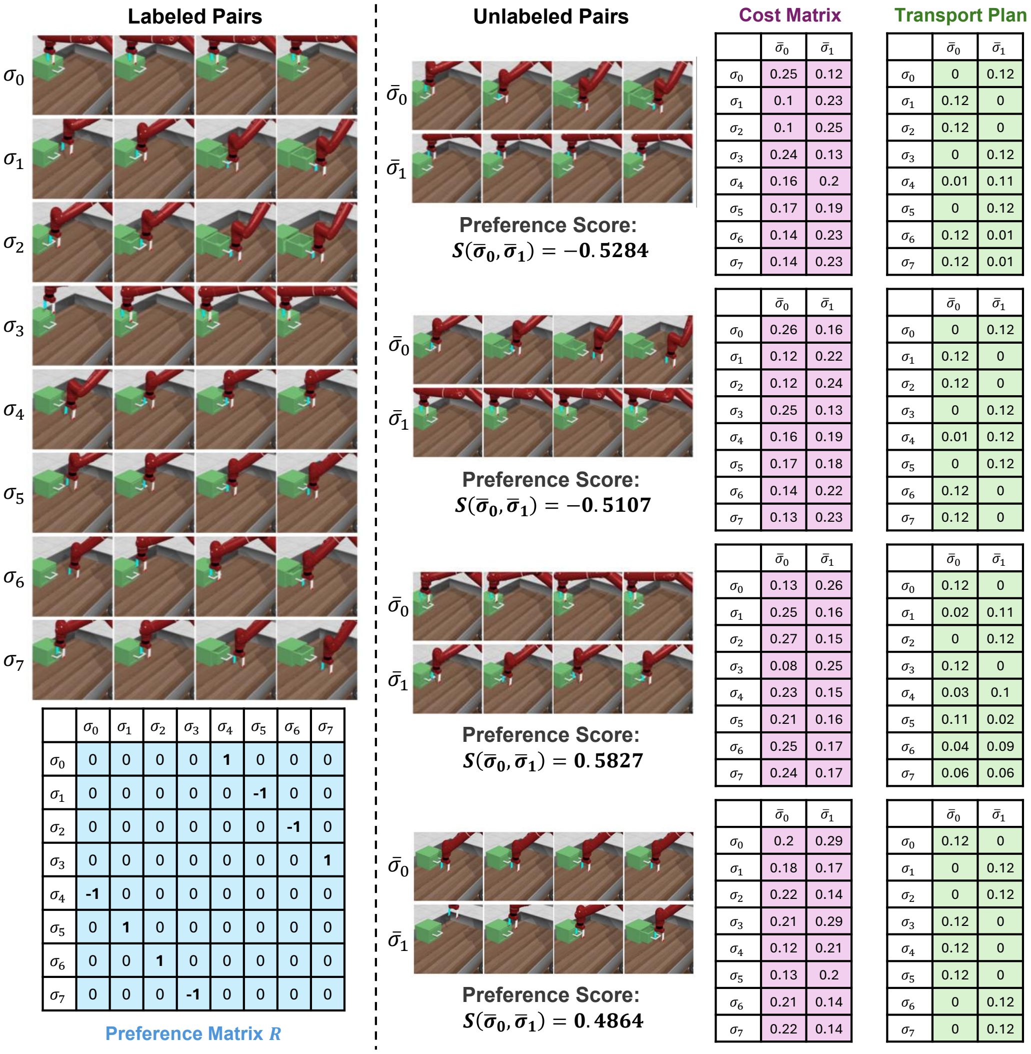
D.9 Robustness to Nuisance Variation

One benefit of using ViFMs to encode visual segments is their strong generalization across varied visual conditions. To examine this property, following Yuan et al. (2023), we modify MetaWorld environments with controlled visual distractions, including changes in lighting, visual appearance, and dynamic video backgrounds. Examples of these scenarios are shown in Figure 21. To evaluate the robustness of VOTP, for each type of distraction we generate unlabeled visual segments with the corresponding perturbation, while keeping the labeled segments fixed across scenarios. Note that both the policy and reward models are trained from state inputs, as described in Section 5.1. This experiment tests the ability of VOTP to generate reliable pseudo-labels under controlled visual distractors. As shown in Table 15, VOTP maintains strong performance despite significant visual variations.

Table 15: Performance of VOTP under various types of visual distractions. Mean and standard deviation are computed over 5 random seeds.

Dataset	Same Domain	Light Changes (pos.+dir.)	Light Changes (amb.+diff.)	Texture Changes	Video (easy)	Video (hard)
door-open	84.0 ± 8.4	88.8 ± 3.0	79.2 ± 3.0	76.8 ± 12.7	79.2 ± 6.9	80.4 ± 4.1
drawer-open	71.2 ± 11.7	74.4 ± 9.2	77.6 ± 7.4	72.0 ± 4.4	68.0 ± 5.7	68.8 ± 8.2
Average	77.6	81.6	78.4	74.4	73.6	74.6

D.10 Visualization for Pseudo-labeling Process



	σ_0	σ_1	σ_2	σ_3	σ_4	σ_5	σ_6	σ_7
σ_0	0	0	0	0	**1**	0	0	0
σ_1	0	0	0	0	0	**-1**	0	0
σ_2	0	0	0	0	0	0	**-1**	0
σ_3	0	0	0	0	0	0	0	**1**
σ_4	**-1**	0	0	0	0	0	0	0
σ_5	0	**1**	0	0	0	0	0	0
σ_6	0	0	**1**	0	0	0	0	0
σ_7	0	0	0	**-1**	0	0	0	0

Preference Matrix *R*

Preference Score:
$\boldsymbol{S(\bar{\sigma}_0, \bar{\sigma}_1) = -0.5284}$

	$\bar{\sigma}_0$	$\bar{\sigma}_1$
σ_0	0.25	0.12
σ_1	0.1	0.23
σ_2	0.1	0.25
σ_3	0.24	0.13
σ_4	0.16	0.2
σ_5	0.17	0.19
σ_6	0.14	0.23
σ_7	0.14	0.23

	$\bar{\sigma}_0$	$\bar{\sigma}_1$
σ_0	0	0.12
σ_1	0.12	0
σ_2	0.12	0
σ_3	0	0.12
σ_4	0.01	0.11
σ_5	0	0.12
σ_6	0.12	0.01
σ_7	0.12	0.01

Preference Score:
$\boldsymbol{S(\bar{\sigma}_0, \bar{\sigma}_1) = -0.5107}$

	$\bar{\sigma}_0$	$\bar{\sigma}_1$
σ_0	0.26	0.16
σ_1	0.12	0.22
σ_2	0.12	0.24
σ_3	0.25	0.13
σ_4	0.16	0.19
σ_5	0.17	0.18
σ_6	0.14	0.22
σ_7	0.13	0.23

	$\bar{\sigma}_0$	$\bar{\sigma}_1$
σ_0	0	0.12
σ_1	0.12	0
σ_2	0.12	0
σ_3	0	0.12
σ_4	0.01	0.12
σ_5	0	0.12
σ_6	0.12	0
σ_7	0.12	0

Preference Score:
$\boldsymbol{S(\bar{\sigma}_0, \bar{\sigma}_1) = 0.5827}$

	$\bar{\sigma}_0$	$\bar{\sigma}_1$
σ_0	0.13	0.26
σ_1	0.25	0.16
σ_2	0.27	0.15
σ_3	0.08	0.25
σ_4	0.23	0.15
σ_5	0.21	0.16
σ_6	0.25	0.17
σ_7	0.24	0.17

	$\bar{\sigma}_0$	$\bar{\sigma}_1$
σ_0	0.12	0
σ_1	0.02	0.11
σ_2	0	0.12
σ_3	0.12	0
σ_4	0.03	0.1
σ_5	0.11	0.02
σ_6	0.04	0.09
σ_7	0.06	0.06

Preference Score:
$\boldsymbol{S(\bar{\sigma}_0, \bar{\sigma}_1) = 0.4864}$

	$\bar{\sigma}_0$	$\bar{\sigma}_1$
σ_0	0.2	0.29
σ_1	0.18	0.17
σ_2	0.22	0.14
σ_3	0.21	0.29
σ_4	0.12	0.21
σ_5	0.13	0.2
σ_6	0.21	0.14
σ_7	0.22	0.14

	$\bar{\sigma}_0$	$\bar{\sigma}_1$
σ_0	0.12	0
σ_1	0	0.12
σ_2	0	0.12
σ_3	0.12	0
σ_4	0.12	0
σ_5	0.12	0
σ_6	0	0.12
σ_7	0	0.12

Figure 22: Additional visualizations of the pseudo-labeling process from VOTP on the drawer-open task. We use 4 labeled pairs. All cost matrix and transport plan entries are rounded to two decimals. For clarity, each segment (originally 64 frames) is uniformly downsampled to 4 frames for visualization.