Learning Video-Conditioned Policy on Unlabelled Data with Joint Embedding Predictive Transformer

Anonymous authors
Paper under double-blind review

Abstract

The video-conditioned policy takes prompt videos of the desired tasks as a condition and is regarded for its prospective generalizability. Despite its promise, training a video-conditioned policy is non-trivial due to the need for abundant demonstrations. In some tasks, the expert rollouts are merely available as videos, and costly and time-consuming efforts are required to annotate action labels. To address this, we explore training video-conditioned policy on a mixture of demonstrations and unlabeled expert videos to reduce reliance on extensive manual annotation. We introduce the Joint Embedding Predictive Transformer (JEPT) to learn a video-conditioned policy through sequence modeling. JEPT is designed to jointly learn visual transition prediction and inverse dynamics. The visual transition is captured from both demonstrations and expert videos, on the basis of which the inverse dynamics learned from demonstrations is generalizable to the tasks without action labels. Experiments on a series of simulated visual control tasks evaluate that JEPT can effectively leverage the mixture dataset to learn a generalizable policy. JEPT outperforms baselines in the tasks without action-labeled data and unseen tasks. We also experimentally reveal the potential of JEPT as a simple visual priors injection approach to enhance the video-conditioned policy.

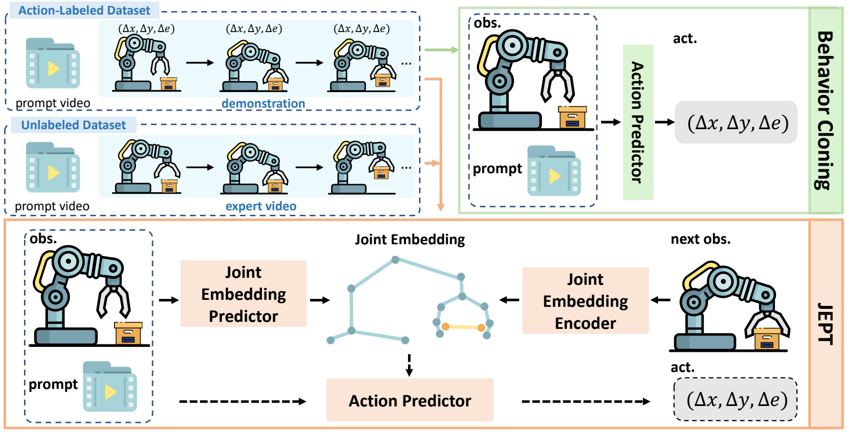

Figure 1: Illustration of the BC and JEPT for visual-conditioned policy learning. Video-conditioned Behavior Cloning (BC) (right top) directly models the policy from the demonstrations. JEPT (bottom) decomposes imitation learning into visual transition prediction (solid arrows) and inverse dynamics learning (dashed arrows). BC is constrained to the action-labeled dataset, whereas JEPT enables visual transition prediction on unlabeled data and leverages the mixture dataset.

1 Introduction

Learning generalizable policies through multi-task imitation learning remains a considerable challenge. A generalizable policy should adapt seamlessly to novel tasks whose demonstrations are absent from the training dataset. In response to this, the video-conditioned policy (Jiang et al., 2023;

Bahl et al., 2023; Shah et al., 2023) has garnered attention for its superior potential to generalize across tasks. A video-conditioned policy takes a prompt video depicting the task as a condition and executes the desired task in the dynamics encountered. Utilizing a prompt video as a policy condition exhibits a better potential for generalization compared to other forms of task specifications, such as instructions or goals given in language (Brohan et al., 2023; Padalkar & et al., 2023; Jiang et al., 2019; Yenamandra et al., 2023) or image (Bousmalis et al., 2024; Lee et al., 2020; Du et al., 2024; Nair et al., 2018) formats. The abundant information conveyed in videos offers sufficient policy guidance for novel tasks, thus circumventing the misleading caused by language ambiguities or the absence of process depiction in static images (Jain et al., 2024).

Although comprehensive information in prompt videos imbues video-conditioned policy with flexibility and generalization, learning such a policy is more than trivial. Imitating the tasks depicted in prompt videos requires models capable of both temporal reasoning to understand the task and fine-grained control to replicate it. To this end, previous methods (Shah et al., 2023; Jain et al., 2024) typically conduct behavior cloning on large datasets of paired prompt videos and expert demonstrations. However, procuring expert demonstrations can be prohibitively costly and time-consuming. In the tasks where expert policies are hard to obtain, we can acquire the videos of the expert rollouts through human intervention, but these videos are devoid of action labels. Conducting behavior cloning in this scenario requires additional efforts to annotate the action labels. This raises the question: Can these expert videos[1] be effectively harnessed without additional annotation, thus alleviating the data collection burden? Accordingly, we explore the possibility of training a generalizable video-conditioned policy on a dataset containing both expert demonstrations and unlabeled expert videos paired with prompt videos.

Unlike methods that directly imitate expert demonstrations, behavior cloning is infeasible on the mixture dataset due to the absence of the action labels from the data samples with expert videos. Intuitively, we decompose the behavior cloning into two synergistic subtasks to fully exploit the mixture dataset of expert demonstrations and expert videos. Specifically, the process of video-conditioned imitation can be broken down into *visual transition prediction* and *inverse dynamics learning*. In the visual transition prediction, the model learns how the prompt videos should manifest in the dynamics of the tasks. In inverse dynamics learning, the model infers the actions required to realize the visual transition. By combining them, the model can learn to predict plausible future observations and then convert them into actions, enabling video-conditioned imitation.

This design potentially suits the setting of mixture dataset. On the one hand, visual transition prediction is more task-specific in visual imitation learning task and can be more sufficiently learned from both expert videos and demonstrations. On the other hand, although the inverse dynamics is merely embedded in the demonstrations, it remains universally applicable across tasks. Consequently, the expert videos could enhance the learning of visual transitions, while the inverse dynamics derived from the demonstrations could generalize to guide the execution of the tasks without action labels.

In light of these insights, we propose the Joint Embedding Predictive Transformer (**JEPT**) to encapsulate both visual transition and inverse dynamics for video-conditioned policy learning. Building on the framework of Decision Transformers (Chen et al., 2021; Lee et al., 2022; Furuta et al., 2022), we employ a video-conditioned Transformer-based architecture to perform sequence modeling, serving as the policy. As shown in Figure 1, we modify the typical behavior cloning sequence into a two-step process. First, JEPT predicts the embeddings of the next observations, thus capturing visual transition. Then, conditioned on these predicted embeddings, JEPT predicts the corresponding actions, thereby learning inverse dynamics. Additionally, we integrate the Joint Embedding Predictive Architecture (JEPA) (LeCun, 2022) into the video-conditioned Transformer. Prior works (Assran et al., 2023; Bardes et al., 2024) have demonstrated that JEPA is an effective visual representation learning approach, compressing visual inputs into predictive features. By incorporating JEPA, we aim to learn predictive visual representations of future observations, thereby improving the transfer of inverse dynamics between action-labeled and unlabeled data.

To evaluate the effectiveness of JEPT, we conduct experiments on Meta-World (Yu et al., 2020a) and Robosuite (Zhu et al., 2020). In our Meta-World experiments, JEPT outperforms all the baselines regarding the average success rates of the tasks within the dataset and the unseen tasks. In the

[1] For simplicity and distinction, we use ‘prompt videos’ referring to videos depicting the desired task and used as the policy condition, while ‘expert videos’ for the visual observation sequence of the expert rollouts.

Robosuite experiments, we further validate the effectiveness of JEPT in handling tasks where there is a larger discrepancy between prompt videos and task dynamics. Additionally, we explore the potential of JEPT in injecting visual priors via adjusting the inputs of the joint embedding encoder. Our experiments on Meta-World reveal that the optical flow priors can effectively improve the performance. These results demonstrate the effectiveness of JEPT in jointly leveraging the mixture dataset of expert demonstrations and expert videos to train a generalizable video-conditioned policy.

In summary, our contributions are as follows:

- We propose a novel paradigm, JEPT, to learn video-conditioned policies using the dataset containing a mixture of expert videos and demonstrations.
- We design JEPT as a joint embedding predictive architecture to learn an abstract visual representation for learning a generalizable video-conditioned policy.
- We explore the JEPT as an approach of visual priors injection and experimentally find that JEPT can effectively leverage visual prior knowledge and achieve better performance.

2 Related Works

Video Prompt Policy Learning. As a densely informative data form, video inherently contains abundant information for task completion when used as a task description. The detailed guidance of the task completion process in the video makes it possible to learn the policy generalizable to novel tasks beyond the training dataset. Some works (Finn et al., 2017; Duan et al., 2017; Yu et al., 2018) employ meta-learning methods to adapt the policy to novel tasks depicted by prompt video, which requires the similarity between the tasks. More recent works focus on learning policy via behavior cloning on datasets of paired prompt videos and expert demonstrations. These works take the prompt videos as a condition input to the policy network and generalize to unseen tasks with corresponding prompt videos. Various auxiliary mechanisms have been explored to achieve generalizable imitation, such as inverse dynamics prediction (Dasari & Gupta, 2021), cross-painiting (Chen et al., 2024), skill decomposition (Shin et al., 2024; 2023), hierarchical policy learning (Jain et al., 2023), text-aligned representation (Jang et al., 2022), contrastive video encoder (Chane-Sane et al., 2023) and observation-attentive representation (Jain et al., 2024). Some works (Sivakumar et al., 2022) design object-centric decomposition for some specific manipulation tasks and achieve generalization within the same task category. The core of these methods remains direct behavior cloning, necessitating access to action-labeled demonstration data. Recent works (Jain et al., 2024; Shah et al., 2023; Jiang et al., 2023) construct large-scale datasets for video-conditioned policy learning, evaluating that large-scale and well-aligned data can effectively improve the video-conditioned policy learning in terms of generalization and success rates. However, these works require extensive demonstration annotation efforts, which is costly and time-consuming. Our work basically follows the video-conditioned policy learning setting. Still, we explore how the video-conditioned policy can learn effectively when the dataset is a mixture of expert videos and demonstrations paired with prompt videos.

Learning from Videos. Given the relative accessibility of video data compared to action-labeled demonstrations, some works explore leveraging unlabeled videos to aid in policy learning. Despite the absence of action labels, videos contain substantial decision-related knowledge due to their rich temporal information. A path for learning from videos involves self-supervised representation learning on videos, including masked autoencoder (Radosavovic et al., 2023; Xiao et al., 2022; Yang et al., 2024), temporal contrastive learning (Li et al., 2024; Nair et al., 2023), and video prediction (Seo et al., 2022). These methods aim to learn a compact and informative representation from videos that aid in policy learning. Other approaches focus on learning reward functions (Escontrela et al., 2023; Yu et al., 2020b; Bruce et al., 2023) or value functions (Bobrin et al., 2024; Chang et al., 2022) to expedite online policy learning. Due to the absence of action labels, these methods require fine-tuning on the downstream tasks for adaptation. Additionally, some research utilizes datasets combining videos and demonstrations, learning an inverse dynamics model (IDM) (Baker et al., 2022; Schmeckpeper et al., 2021; Zheng et al., 2023; Zhang et al., 2022; Kim et al., 2023) or a latent inverse dynamics model (Schmidt & Jiang, 2024; Ye et al., 2022; Edwards et al., 2019) on the action-labeled data. The learned IDM can be used to annotate the video data, facilitating imitation learning to derive policies. In these works, the decoupling of inverse dynamics learning and visual imitation may limit the generalizability of the IDM. In our work, however, the learning of visual transitions and inverse dynamics is conducted jointly, offering better generalization.

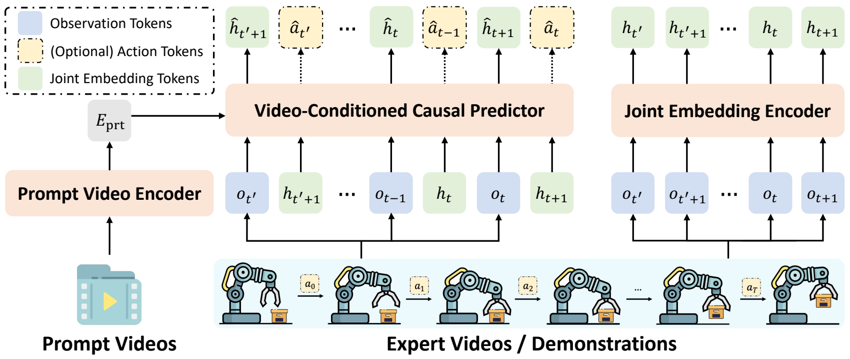
Observation Tokens
(Optional) Action Tokens
Joint Embedding Tokens
$\hat{h}_{t'+1}$ $\hat{a}_{t'}$ ⋯ $\hat{h}_t$ $\hat{a}_{t-1}$ $\hat{h}_{t+1}$ $\hat{a}_t$
$h_{t'}$ $h_{t'+1}$ ⋯ h_t h_{t+1}
E_{prt}
Video-Conditioned Causal Predictor
Joint Embedding Encoder
Prompt Video Encoder
$o_{t'}$ $h_{t'+1}$ ⋯ o_{t-1} h_t o_t h_{t+1}
$o_{t'}$ $o_{t'+1}$ ⋯ o_t o_{t+1}
a_0 a_1 a_2 ... a_T
Prompt Videos
Expert Videos / Demonstrations

Figure 2: The architecture of the Joint Embedding Predictive Transformer (JEPT). The model consists of a joint embedding encoder, a prompt video encoder, and a video-conditioned causal predictor. The demonstrations and the expert videos are processed as a sequence of tokens, and the causal predictor predicts the tokens causally. The visual embedding token of the next observation is predicted to learn a predictive joint embedding. We use $t' := t - k + 1$ to denote the start of the context.

Joint Embedding Predictive Architecture. The Joint Embedding Predictive Architecture (JEPA) (LeCun, 2022) emerges as a promising representation learning framework. JEPA optimizes a predictive loss on the encoded embeddings to learn an embedding space where the embeddings are predictable to each other. Unlike the generative architectures, JEPA optimizes the predictive loss on embeddings rather than raw data, potentially discarding extraneous information to learn more compact and meaningful representations. In order to train a JEPA, it is a common practice to construct data pairs to predict and be predicted. JEPA is a versatile framework. Various works employ different data pair construction methods to enable JEPA to learn embedding spaces tailored to specific information, enhancing tasks such as image classification (Assran et al., 2023), mask classification (Kim et al., 2024), video understanding (Bardes et al., 2024), and motion and content learning (Bardes et al., 2023). Our work integrates JEPA with sequence modeling on trajectories, providing a novel data pair construction method for JEPA. This extends JEPA to architectures akin to the Decision Transformers, aiming to learn visual representations that facilitate inverse dynamics knowledge transfer.

3 Methodology

In this section, we describe how our Joint Embedding Predictive Transformer (JEPT) works on the mixture of expert demonstrations and expert videos paired with prompt videos. We first introduce the problem formulation in Section 3.1. We then elaborate on the components of JEPT, including a prompt video encoder, a joint embedding encoder, and a video-conditioned causal predictor in Section 3.2. We further detail the training procedure in Section 3.3.

3.1 Preliminaries

We aim to learn a video-conditioned policy across a set of tasks. The environment is viewed as a collection of Partially Observable Markov Decision Process variants (POMDPs), with each task represented by a POMDP variant $\mathcal{M}_i := (\mathcal{S}, \mathcal{A}, \mathcal{P}, \mathcal{O}, \Omega, \mathcal{V}_i)$. Here, $\mathcal{S}, \mathcal{A}, \mathcal{P}, \mathcal{O}, \Omega$ denote the state space, action space, transition function, observation space, and observation function, respectively, shared across all tasks. Specifically, $\mathcal{V}_i$ comprises a set of prompt videos depicting the task $\mathcal{M}_i$.

Video-Conditioned Policy. During interaction with the environment, the agent receives a prompt video $V \in \mathcal{V}_i$ illustrating the desired task at the start of each rollout. At each timestep t, the agent obtains a visual observation $o_t \in \mathcal{O}$, derived from the current state $s_t \in \mathcal{S}$ via the observation function $\Omega \colon \mathcal{S} \to \mathcal{O}$. Upon executing an action $a_t \in \mathcal{A}$, the environment transitions to the subsequent state s_{t+1} according to the transition function $\mathcal{P} \colon \mathcal{S} \times \mathcal{A} \times \mathcal{S} \to [0, 1]$. A video-conditioned policy generates actions based on the visual observation context and the prompt video. Formally, a video-conditioned policy is denoted as $\pi\left(a_t | o_{[t-k+1:t]}, V\right)$, where $o_{[t-k+1:t]}$ represents the visual observation context over the last k timesteps.

Mixture Dataset. We consider training the video-conditioned policy on a mixture dataset. Ideally, we collect three types of data for imitation learning: (1) the prompt video $V := (v_1, v_2, \dots, v_N)$ specifying the task with N frames, (2) the expert video $O := (o_0, o_1, \dots, o_T)$, the visual observation sequence recorded alongside a T-step expert rollout of the task, and (3) the corresponding action sequence $A := (a_0, a_1, \dots, a_{T-1})$ executed during the same rollout. In the tasks where expert demonstrations are available, we assume access to the dataset $\mathcal{D}_{\text{demo}}$, and each data sample $d_{\text{demo}} \in \mathcal{D}_{\text{demo}}$ is in the form:

$$d^{(i)}_{\text{demo}} = \left((v_1, v_2, \dots, v_N), (o_0, a_0, o_1, a_1, \dots, o_T) \sim \pi^\star_i\right). \tag{1}$$

Here, $\pi^\star$ represents the expert policies, $\sim$ denotes the rollout process, and i indexes the dataset sample. Conversely, in the tasks where merely videos of expert rollouts are available, we denote this dataset as $\mathcal{D}_{\text{vid}}$, and each data sample $d_{\text{vid}} \in \mathcal{D}_{\text{vid}}$ is in the form:

$$d^{(i)}_{\text{vid}} = \left((v_1, v_2, \dots, v_N), (o_0, o_1, \dots, o_T) \sim \pi^\star_i\right). \tag{2}$$

Each data sample in $\mathcal{D}_{\text{demo}}$ consists of a prompt video V and a paired expert demonstration (O, A), while each data sample in $\mathcal{D}_{\text{vid}}$ consists of a prompt video V and a paired expert video O. In summary, our objective is to train the video-conditioned policy $\pi\left(a_t | o_{[t-k+1:t]}, V\right)$ on a dataset mixed of these two kinds of data, denoted as $\mathcal{D}_{\text{demo}} \cup \mathcal{D}_{\text{vid}}$.

3.2 Model Architecture

Our JEPT employs a Transformer-based architecture to perform sequence modeling on the data, following the supervised learning paradigm of the Generalized Decision Transformers (Furuta et al., 2022). In this context, demonstrations are treated as sequences of observation and action tokens, where the sequence modeling serves as a video-conditioned policy via predicting the corresponding action tokens. To perform the sequence modeling, JEPT comprises two kinds of modules: (1) visual encoders that process the high-dimensional visual input and (2) a causal predictor aggregating the visual representations to causally predict tokens in the sequence. The comprehensive architecture of JEPT is illustrated in Figure 2, and we elucidate each component in detail below.

3.2.1 Visual Encoders

Joint Embedding Encoder. A Joint Embedding Encoder is employed to learn the visual representation for each observation. In our design, the Joint Embedding Encoder captures information from both single-timestep and contextual levels. Structurally, the Joint Embedding Encoder comprises a spatial encoder for single-timestep encoding and a bi-directional temporal encoder for contextual encoding. Formally, the Joint Embedding Encoder Ψ_{obs} takes as input a visual observation context and outputs the visual representations:

$$(h_{t-k+1}, \dots, h_{t-1}, h_t) = \Psi_{\text{obs}}\left(o_{t-k+1}, \dots, o_{t-1}, o_t\right). \tag{3}$$

Each representation h_t consists of N_{obs} tokens. The visual embeddings of the next observation is predicted by the causal predictor and thus a joint embedding will be learned. Via learning joint embeddings predictive of the previous observation tokens, the encoder can compress the high-dimensional visual space into a low-dimensional space and discard extraneous information for effective and generalizable policy learning.

Prompt Video Encoder. A Prompt Video Encoder encodes a prompt video into an embedding that serves as a reference for the policy. To specify the desired task, the embedding of the prompt video is used for an implicit understanding of task execution. Specifically, the Prompt Video Encoder comprises a per-frame encoder and a Perceiver Resampler (Jaegle et al., 2022). The Perceiver Resampler aggregates frame embeddings from the per-frame encoder to produce an overall embedding E_{prt} of the prompt video. Each representation E_{prt} consists of N_{prt} tokens. E_{prt} abstracts task-relevant attributes from videos and serves as the policy condition. Formally, the Prompt Video Encoder outputs as:

$$E_{\text{prt}} = \Psi_{\text{prt}}\left(v_1, v_2, \dots, v_N\right). \tag{4}$$

3.2.2 Causal Predictor

Connecting to the visual encoders, a video-conditioned Causal Predictor integrates the visual representations of prompt videos and observations to conduct sequence modeling on the mixture dataset.

To effectively leverage the mixture dataset, the predictor is designed to conduct two subtasks of behavior cloning, visual transition prediction and inverse dynamics learning. The visual transition delineates how observations should evolve to accomplish tasks depicted by the prompt videos. Formally, visual transition prediction approximates the distribution $P\left(o_{t+1}|o_{\leq t}, V\right)$, which is embedded in both $\mathcal{D}_{\text{demo}}$ and $\mathcal{D}_{\text{vid}}$. Through visual transition prediction, the model learns to align the visual observation sequence with the prompt videos. Inverse dynamics learning approximates the distribution $P\left(a_t|o_t, o_{t+1}\right)$, reflecting the actions required to realize a given visual transition. Although the action labels are only available in $\mathcal{D}_{\text{demo}}$, $P\left(a_t|o_t, o_{t+1}\right)$ is shared across tasks due to the shared transition function $\mathcal{P}$. Given the challenge of predicting in the raw visual observation space, we replace o_{t+1} with the joint embedding of the next observation h_{t+1} in seek of a compressed embedding space conducive to knowledge transfer. Accordingly, the causal predictor is tasked with capturing $P\left(h_{t+1}|o_{\leq t}, V\right)$ and $P\left(a_t|o_t, h_{t+1}\right)$.

The input and output sequences of the causal predictor are designed to concurrently capture the visual transition $P\left(h_{t+1}|o_{\leq t}, V\right)$ and the inverse dynamics $P\left(a_t|o_t, h_{t+1}\right)$. As shown in Figure 2, with the video representation E_{prt} as the prompt, the causal predictor takes the trajectory sequence $(o_{t-k+1}, h_{t-k+2}, \ldots, o_t, h_{t+1})$ as input to causally predict $(\hat{h}_{t-k+2}, \hat{a}_{t-k+1}, \ldots, \hat{h}_{t+1}, \hat{a}_t)$ respectively. Structurally, the predictor comprises a causal Transformer encoder Ψ_{pred} and two prediction heads, Γ_{obs} and Γ_{act}. Respectively, Γ_{obs} predicts the joint embedding of the next observation h_{t+1} from the hidden states of observation tokens o_t, and Γ_{act} predicts the action tokens a_t from the hidden states of h_{t+1}. Formally, the predictor encodes the input sequence:

$$\boldsymbol{Z} = \Psi_{\text{pred}}\left(E_{\text{prt}}, o_{t-k+1}, h_{t-k+2}, \ldots, o_t, h_{t+1}\right) \tag{5}$$

and predicts the tokens:

$$\hat{h}_{t+1} = \Gamma_{\text{obs}}\left(Z_t^{(o)}\right), \quad \hat{a}_t = \Gamma_{\text{act}}\left(Z_{t+1}^{(h)}\right). \tag{6}$$

Here, $Z_t^{[\cdot]}$ denotes the hidden states of the corresponding tokens in $\boldsymbol{Z}$, while $\hat{h}_{t+1}$ and $\hat{a}_t$ represent the predicted tokens.

In this predictive form, the causal predictor alternatively models $P\left(h_{t+1}|E_{\text{prt}}, h_{\leq t}, o_{\leq t}\right)$ to capture the visual transition and $P\left(a_t|E_{\text{prt}}, o_{\leq t}, h_{\leq t+1}\right)$ to capture the inverse dynamics. During inference, the predictor iteratively feeds the predicted joint embedding tokens back into the input sequence to predict tokens. By combining the two types of token prediction together, the causal predictor indeed works as a planning-based policy, which predicts the desired next observations first and then converts the plan into actions.

3.3 Training Procedure

With the designed sequence modeling form, JEPT can simultaneously capture visual transition and inverse dynamics. In this subsection, we describe how JEPT is optimized on the mixture dataset $\mathcal{D}_{\text{demo}} \cup \mathcal{D}_{\text{vid}}$. Specifically, two predictive losses, the visual transition loss and the inverse dynamics loss, are optimized to predict the joint embedding tokens and the action tokens, respectively.

Visual Transition Loss. For the joint embedding prediction, we compute the average L_2 distance between the predicted joint embedding and the encoded joint embedding as the visual transition loss $\mathcal{L}_{\text{obs}}$ to approximate the visual transition $P\left(h_{t+1}|E_{\text{prt}}, h_{\leq t}, o_{\leq t}\right)$. For both $\mathcal{D}_{\text{demo}}$ and $\mathcal{D}_{\text{vid}}$, the joint embedding of the next observations is available. The visual transition loss $\mathcal{L}_{\text{obs}}$ is defined as:

$$\mathcal{L}_{\text{obs}} = \mathbb{E}_{(V,O)\sim\mathcal{D}_{\text{demo}}\cup\mathcal{D}_{\text{vid}}}\left[\frac{1}{k}\sum_{i=t-k+1}^{t}\left\|h_{i+1} - \hat{h}_{i+1}\right\|_2\right]. \tag{7}$$

Inverse Dynamics Loss. For the action prediction, we compute the average Cross-Entropy between the predicted actions and the discrete ground-truth actions as the inverse dynamics loss $\mathcal{L}_{\text{act}}$ to approximate the inverse dynamics $P\left(a_t|E_{\text{prt}}, o_{\leq t}, h_{\leq t+1}\right)$. The ground-truth action label is merely available in $\mathcal{D}_{\text{demo}}$. The inverse dynamics loss $\mathcal{L}_{\text{act}}$ is defined as:

$$\mathcal{L}_{\text{act}} = \mathbb{E}_{(V,O,A)\sim\mathcal{D}_{\text{demo}}}\left[\frac{1}{k}\sum_{i=t-k+1}^{t} a_i \log \hat{a}_i\right]. \tag{8}$$

Via optimizing $\mathcal{L}_{\text{obs}}$, JEPT learns to generate the visual transitions aligned with the prompt videos. By optimizing $\mathcal{L}_{\text{act}}$, JEPT learns to convert the visual transitions into actions. Formally, all the components of JEPT are trained with the loss $\mathcal{L}_{\text{total}}$:

$$\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{obs}} + \mathbb{1}_{d \in \mathcal{D}_{\text{demo}}} \cdot c \cdot \mathcal{L}_{\text{act}}, \tag{9}$$

where c is a hyperparameter balancing the two losses, and $\mathbb{1}_{d \in \mathcal{D}_{\text{demo}}}$ is an indicator function that equals 1 when the data sample d is from $\mathcal{D}_{\text{demo}}$ and 0 otherwise. Considering that JEPT predicts the encoded joint embeddings instead of the raw visual observations, we adopt an alternative training procedure to separately optimize the Joint Embedding Encoder and other components in case of potential model collapse. Additionally, an exponential moving average (EMA) of the joint embedding is used when the Joint Embedding Encoder is a fixed target network to stabilize the training, which is widely adapted in the previous JEPAs (Assran et al., 2023; Bardes et al., 2024). The overall process is shown in Algorithm 1 in Appendix A.1.

4 Experiemntal Results

In this section, we evaluate the effectiveness of our proposed JEPT on the mixture dataset, where the model learns from action-labeled expert demonstrations and unlabeled expert videos. We evaluate JEPT and baselines on two simulated benchmarks, Meta-World (Yu et al., 2020a) and RoboSuite (Zhu et al., 2020). Both benchmarks provide a variety of robotic manipulation tasks and are widely used for evaluating visual control tasks. Via the experiments, we aim to figure out: (1) whether JEPT effectively leverages the additional $\mathcal{D}_{\text{vid}}$ to improve the performance on unlabeled tasks, (2) whether the learned policy can generalize to unseen tasks, and (3) whether the joint embedding predictive mechanism in JEPT is essential for knowledge transfer.

4.1 Experiment Setup

We first introduce the experimental setup, including the environments, dataset, metrics, and baselines used in our experiments. More details are available in Appendix B.

Environments. We replace the language task descriptions in the environments with prompt videos recorded in the environments as task specifications. In our Meta-World experiments, a Sawyer robot arm interacts with various objects. The prompt videos are recorded with the same robot arm performing identical manipulations, which means there is no visual gap between the prompt videos and the expert videos. In our Robosuite experiments, a Panda robot arm interacts with objects. The prompt videos are recorded by performing the same manipulation with various robot arms, including Panda, Sawyer, IIWA, and UR5e, introducing a visual gap between the prompt videos and the expert videos.

Dataset. We select 18 tasks for the Meta-World task set and 15 tasks for the Robosuite task set. In order to construct the mixture dataset $\mathcal{D}_{\text{demo}} \cup \mathcal{D}_{\text{vid}}$ and evaluate on the unseen tasks, we split each task set into three subsets: (1) $\mathcal{T}_{\text{demo}}$: the tasks with the prompt videos and the paired expert demonstrations, (2) $\mathcal{T}_{\text{vid}}$: the tasks with the prompt videos and the paired expert videos, and (3) $\mathcal{T}_{\text{unseen}}$: the tasks with merely the prompt videos. For each task, we construct an expert policy in the vector state space and collect demonstrations or videos by running this expert policy in the rendered environments. $\mathcal{D}_{\text{demo}}$ is collected from the tasks in $\mathcal{T}_{\text{demo}}$, while $\mathcal{D}_{\text{vid}}$ is collected from the tasks in $\mathcal{T}_{\text{vid}}$. We also collect prompt videos for the tasks from $\mathcal{T}_{\text{unseen}}$, which are not included in the mixture dataset.

Metrics. We evaluate the model performance in terms of success rate, which is the percentage of successful episodes within 50 trials with different random seeds. The success rates of the tasks from $\mathcal{T}_{\text{demo}}$, $\mathcal{T}_{\text{vid}}$ and $\mathcal{T}_{\text{unseen}}$ reflect the model's ability of **behavior cloning**, **learning from videos**, and **one-shot imitation learning**, respectively. We also calculate the average success rates of seen and unseen tasks to evaluate the overall performance.

Baselines. The baselines chosen for comparison with JEPT in our setting are as follows:

- **Vid2Robot** (Jain et al., 2024): conducts behavior cloning and auxiliary representation learning on a large dataset where all the data is action-labeled. We train two variants of Vid2Robot: **Vid2Robot-D** refers to Vid2Robot trained merely on $\mathcal{D}_{\text{demo}}$, while **Vid2Robot-M** refers to Vid2Robot trained on $\mathcal{D}_{\text{demo}} \cup \mathcal{D}_{\text{vid}}$ with behavior cloning loss masked from $\mathcal{D}_{\text{vid}}$.

Table 1: Success Rates (%) of JEPT and the baselines calculated from 50 trials for each task in Meta-World. The average success rate of the tasks in $\mathcal{T}_{\text{demo}}$ and $\mathcal{T}_{\text{vid}}$ are listed in the first two rows, while the individual success rate of the 4 tasks in $\mathcal{T}_{\text{unseen}}$ are listed in the following rows. The average success rates of the seen and unseen tasks are listed in the last two row.

Task	**Vid2Robot-D**	**Vid2Robot-M**	**BC+IDM**	**JEPT+MWM**	**DT***	**JEPT**
$\mathcal{T}_{\text{demo}}$	51.8	46.3	49.5	44.5	**61.3**	51.3
$\mathcal{T}_{\text{vid}}$	4.7	28.7	8.3	21.3	13.0	**31.7**
Handle Press	10.0	22.0	8.0	18.0	6.0	**28.0**
Lever Pull	0.0	4.0	0.0	4.0	0.0	**10.0**
Plate Slide Back	4.0	8.0	0.0	2.0	0.0	**14.0**
Faucet Open	4.0	14.0	4.0	14.0	0.0	**22.0**
Seen Average	28.2	37.5	28.9	32.9	37.1	**41.5**
Unseen Average	4.5	12.0	3.0	9.5	1.5	**18.5**

Table 2: Success Rates (%) of the ablations calculated from 50 trials for each task in Meta-World. We train JEPT on a dataset with various task numbers for $\mathcal{D}_{\text{vid}}$ collection. The average success rate of the tasks in $\mathcal{T}_{\text{demo}}$ is listed in the first row, while the individual success rate of the 4 tasks in $\mathcal{T}_{\text{unseen}}$ are listed in the following rows. The average success rate of the unseen tasks is calculated in the last row.

Task	**None**	**2 Tasks**	**4 Tasks**	**6 Tasks (JEPT)**
$\mathcal{T}_{\text{demo}}$	**59.5**	49.8	47.0	51.3
Handle Press	6.0	4.0	16.0	**28.0**
Lever Pull	0.0	0.0	4.0	**10.0**
Plate Slide Back	0.0	0.0	10.0	**14.0**
Faucet Open	0.0	2.0	20.0	**22.0**
Unseen Average	1.5	1.5	12.5	**18.5**

- **BC+IDM**: Zheng et al. (2023); Kim et al. (2023) proposed a paradigm leveraging the mixture dataset for visual imitation learning. These methods separately learn an inverse dynamics model for action annotation and conduct behavior cloning on the mixture dataset. We accommodate these methods to video-conditioned imitation learning.
- **DT***: We remove the joint embedding predictive mechanism from JEPT, which indeed conducts behavior cloning with a DT variant on $\mathcal{D}_{\text{demo}}$.
- **JEPT+MWM** (Seo et al., 2023): We replace the joint embedding in JEPT with the representation learned with MWM as a baseline. MWM provides a practical representation learning method for visual control tasks. Unlike the joint embedding in JEPT, MWM learns a generative Masked Auto-Encoder (MAE).

4.2 Meta-World Experiments

In the Meta-World experiments, we divide the 18 tasks into three subsets: $\mathcal{T}_{\text{demo}}$, $\mathcal{T}_{\text{vid}}$ and $\mathcal{T}_{\text{unseen}}$, respectively containing 8, 6 and 4 tasks.

Comparisons with Baselines. The task success rates of JEPT and the baselines are presented in Table 1, where JEPT surpasses all baselines in terms of average success rates. Regarding the success rates of tasks from $\mathcal{T}_{\text{demo}}$, JEPT performs slightly worse than DT*, similar to the gap between Vid2Robot-M and Vid2Robot-D. This may be a side-effect of the more tasks integrated into the policy, leading to a decrease in the performance of individual tasks. Notably, JEPT exhibits a significant advantage in tasks from both $\mathcal{T}_{\text{vid}}$ and $\mathcal{T}_{\text{unseen}}$, indicating that JEPT effectively leverages the additional $\mathcal{D}_{\text{vid}}$ to learn a more generalizable policy. Compared to Vid2Robot-M, the sequence modeling design in JEPT, extracting visual transition and inverse dynamics, might suit the mixture dataset more. Although BC+IDM also captures the inverse dynamics, JEPT integrates transition-aware representation learning with inverse dynamics learning on the mixture dataset instead of an IDM learned merely on $\mathcal{D}_{\text{demo}}$, which may capture more generalizable inverse dynamics knowledge and accounts for the better performance of JEPT. Additionally, the outperformance of JEPT over DT* and JEPT+MWM indicates that the predictive joint embedding in JEPT is more suitable for learning on mixture datasets than non-specialized or alternative embedding mechanisms.

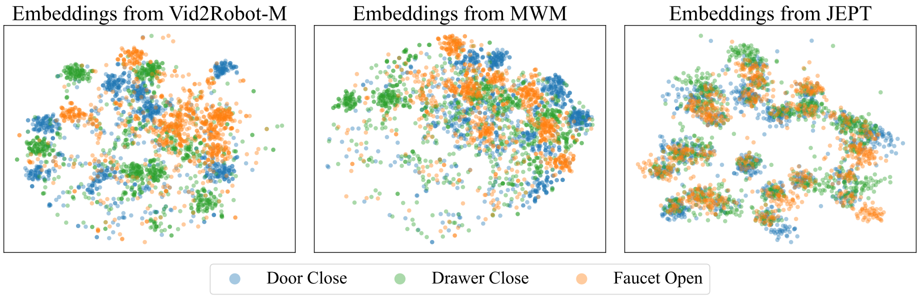

Figure 3: Visualization of the observation embeddings learned in JEPT (right), MWM (mid), and Vid2Robot-M (left). The observation embeddings are calculated in a rollout video from 'Door Close' in $\mathcal{T}_{\text{demo}}$, 'Drawer Close' in $\mathcal{T}_{\text{vid}}$, and 'Facet Open' in $\mathcal{T}_{\text{unseen}}$. We apply t-SNE on one whole embedding set and split the projected vectors according to the algorithms into three sub-figures.

Table 3: Success Rates (%) of JEPT and the baselines calculated from 50 trials for each task in Robosuite. The average success rate of the tasks in $\mathcal{T}_{\text{demo}}$ and $\mathcal{T}_{\text{vid}}$ are listed in the first two rows, while the individual success rate of the 4 tasks in $\mathcal{T}_{\text{unseen}}$ are listed in the following rows. **'X T'** refers to the task where Panda robot arm performs the task **T** with the prompt video recorded with **X** robot arm. The average success rates of the seen and unseen tasks are listed in the last two rows.

Task	Vid2Robot-D	Vid2Robot-M	BC+IDM	JEPT+MWM	DT*	JEPT
$\mathcal{T}_{\text{demo}}$	26.8	20.7	26.7	24.3	**36.3**	27.2
$\mathcal{T}_{\text{vid}}$	4.0	11.6	8.4	10.8	2.0	**12.8**
Panda Lift	4.0	16.0	6.0	22.0	8.0	**38.0**
Sawyer Lift	0.0	2.0	0.0	6.0	0.0	**16.0**
IIWA Lift	0.0	4.0	0.0	2.0	0.0	**12.0**
UR5e Lift	0.0	0.0	0.0	0.0	0.0	**8.0**
Seen Average	15.5	16.1	17.5	17.6	19.2	**20.1**
Unseen Average	1.0	5.5	1.5	7.5	2.0	**18.5**

Ablation Study. We conduct an ablation study to investigate the impact of the size of $\mathcal{D}_{\text{vid}}$. As is listed in Table 2, we train JEPT on datasets where $0, 2, 4$ and 6 tasks are covered in $\mathcal{D}_{\text{vid}}$. The success rates of tasks increases as the size of $\mathcal{T}_{\text{vid}}$ grows. The results indicates that the additional video data is benificial and JEPT can effectively leverage the data in $\mathcal{D}_{\text{vid}}$ to learn a more generalizable video-conditioned policy. Additionally, when the $\mathcal{D}_{\text{vid}}$ grows from 0 tasks to 6 tasks, the performance degradation on the seen tasks might result from a higher integration of the policy, which stands with our observation in Table 1.

Visualizations. We also visualize the visual embeddings learned in JEPT, MWM, and Vid2Robot-M. Specifically, we select one task from each of $\mathcal{T}_{\text{demo}}$, $\mathcal{T}_{\text{vid}}$, and $\mathcal{T}_{\text{unseen}}$ and calculate the observation embeddings of a rollout video from each task. We apply t-SNE to the whole set containing the observation embeddings of the three videos from JEPT, MWM, and Vid2Robot to project them into a shared 2D space. As shown in Figure 3, the observation embeddings from Vid2Robot-M and MWM demonstrate severe out-of-distribution phenomena among the action-labeled, unlabeled, and unseen tasks. The embeddings from JEPT exhibit greater similarity in the distribution, potentially contributing to its superior generalization.

4.3 Robosuite Experiments

We also conduct experiments on the Robosuite tasks, which offers a variety of robotic arms for manipulation tasks. Thus we evaluate the performance of JEPT in the case where there is a visual gap between prompt videos and expert videos. We split the 15 tasks into three subsets: $\mathcal{T}_{\text{demo}}$, $\mathcal{T}_{\text{vid}}$, and $\mathcal{T}_{\text{unseen}}$, containing 6, 5, and 4 tasks, respectively. The results of these experiments are shown in Table 3. Although the success rates in these tasks are somewhat lower than those in the Meta-World experiments due to the increased difficulty, JEPT still outperforms the baselines across all tasks. Similarly to the Meta-world results, the success rate of JEPT in $\mathcal{T}_{\text{demo}}$ is lower than that of DT*, possibly due to the integration of more tasks into a single policy. However, JEPT still exceeds all

Table 4: Success Rates (%) of JEPT injected with different visual priors calculated from 50 trials for each task in Meta-World tasks. The average success rate of the tasks in $\mathcal{T}_{\text{demo}}$ and $\mathcal{T}_{\text{vid}}$ are listed in the first two rows, while the individual success rate of the 4 tasks in $\mathcal{T}_{\text{unseen}}$ are listed in the following rows. The average success rate of the seen and unseen tasks are calculated in the last two row.

Task	JEPT+FlowFormer	JEPT+VideoMAE-v2	JEPT+Dino-v2	JEPT+SAM
$\mathcal{T}_{\text{demo}}$	**59.8**	33.3	38.5	41.3
$\mathcal{T}_{\text{vid}}$	**35.3**	11.7	9.3	19.0
Handle Press	**36.0**	8.0	8.0	16.0
Lever Pull	**10.0**	2.0	2.0	4.0
Plate Slide Back	**6.0**	0.0	0.0	2.0
Faucet Open	**26.0**	4.0	4.0	10.0
Seen Average	**47.5**	22.5	23.9	30.1
Unseen Average	**19.5**	3.5	3.5	8.0

the baselines in $\mathcal{T}_{\text{vid}}$ and $\mathcal{T}_{\text{unseen}}$, indicating that JEPT can effectively leverage $\mathcal{D}_{\text{vid}}$ to learn more generalizable policies despite visual gaps.

4.4 Visual Prior Injection

Considering that visual priors learned from external datasets may enhance the generalization of the model due to their universality, we additionally explore the possibility of injecting such priors from the pre-trained models into JEPT. A directly injection approach is adopted in our practice. We simply replace the output of the single-timestep spatial encoder in the Joint Embedding Encoder with the output from the visual encoders of some pre-trained visual models. Specifically, we experiment on Meta-World with FlowFormer (Huang et al., 2022), VideoMAE-v2 (Wang et al., 2023), Dino-v2 (Oquab et al., 2024), and SAM (Kirillov et al., 2023), representing visual priors related to optical flow, video reconstruction, depth estimation, and visual segmentation, respectively. As shown in Table 4, JEPT+FlowFormer outperforms other visual priors and surpasses the original JEPT. This indicates that the optical flow priors are beneficial for visual control tasks and can generalize across tasks. Meanwhile, JEPT can effectively incorporate the optical flow priors to enhance model performance. The poor performance of other visual priors may result from the inherent unsuitability of these priors for visual control tasks or the inadequacy of the injection approach, requiring further exploration in future work.

5 Conclusion

In this paper, we propose the Joint Embedding Predictive Transformer (JEPT), a novel approach for video-conditioned policy learning. JEPT is designed to learn from a mixture of expert demonstrations and expert videos paired with prompt videos, aiming to reduce the burden of action label annotation. To suit the mixture dataset, we decompose the video-conditioned policy learning into two subtasks: visual transition prediction and inverse dynamics learning. By jointly learning the two subtasks in the sequence modeling, JEPT works as a planning-based policy. We implement JEPT as an extension of the joint embedding predictive architecture to learn an abstract representation of visual observation, which aids in generalizing video-conditioned policy. Experimentally, we evaluate the effectiveness of JEPT on a series of visual control tasks. Additionally, we explore the JEPT as a simple visual priors injection approach and find it valid in injecting optical flow knowledge.

Limitations. A series of experimental results of JEPT indicate that joint learning visual transition and inverse dynamics allow it to effectively leverage the mixture dataset and derive a generalizable policy. Despite this, we have also identified some limitations. Since our work focuses on solving problems within datasets containing expert videos, our method does not include additional designs for the potential visual gap between the prompt videos and task dynamics. We have validated the effectiveness of JEPT in addressing a certain level of visual gap in Robosuite environments, but more design and validation are needed to apply JEPT to one-shot visual imitation with more discrepancy prompt videos such as human videos. In this regard, our attempt at injecting visual priors could be one beneficial approach.

REFERENCES

Mahmoud Assran, Quentin Duval, Ishan Misra, Piotr Bojanowski, Pascal Vincent, Michael Rabbat, Yann LeCun, and Nicolas Ballas. Self-supervised learning from images with a joint-embedding predictive architecture. In *Proc. of Computer Vision and Pattern Recognition*, 2023.

Shikhar Bahl, Russell Mendonca, Lili Chen, Unnat Jain, and Deepak Pathak. Affordances from human videos as a versatile representation for robotics. In *Proc. of Computer Vision and Pattern Recognition*, 2023.

Bowen Baker, Ilge Akkaya, Peter Zhokov, Joost Huizinga, Jie Tang, Adrien Ecoffet, Brandon Houghton, Raul Sampedro, and Jeff Clune. Video pretraining (vpt): Learning to act by watching unlabeled online videos. In *Proc. of Neural Information Processing Systems*, 2022.

Adrien Bardes, Jean Ponce, and Yann LeCun. Mc-jepa: A joint-embedding predictive architecture for self-supervised learning of motion and content features. *arXiv preprint arXiv:2307.12698*, 2023.

Adrien Bardes, Quentin Garrido, Jean Ponce, Xinlei Chen, Michael Rabbat, Yann LeCun, Mido Assran, and Nicolas Ballas. V-JEPA: Latent video prediction for visual representation learning, 2024. URL https://openreview.net/forum?id=WFYbBOEOtv.

Maksim Bobrin, Nazar Buzun, Dmitrii Krylov, and Dmitry V Dylov. Align your intents: Offline imitation learning via optimal transport. *arXiv preprint arXiv:2402.13037*, 2024.

Konstantinos Bousmalis, Giulia Vezzani, Dushyant Rao, Coline Manon Devin, Alex X. Lee, Maria Bauza Villalonga, Todor Davchev, Yuxiang Zhou, Agrim Gupta, Akhil Raju, Antoine Laurens, Claudio Fantacci, Valentin Dalibard, Martina Zambelli, Murilo Fernandes Martins, Rugile Pevceviciute, Michiel Blokzijl, Misha Denil, Nathan Batchelor, Thomas Lampe, Emilio Parisotto, Konrad Zolna, Scott Reed, Sergio Gómez Colmenarejo, Jonathan Scholz, Abbas Abdolmaleki, Oliver Groth, Jean-Baptiste Regli, Oleg Sushkov, Thomas Rothörl, Jose Enrique Chen, Yusuf Aytar, David Barker, Joy Ortiz, Martin Riedmiller, Jost Tobias Springenberg, Raia Hadsell, Francesco Nori, and Nicolas Heess. Robocat: A self-improving generalist agent for robotic manipulation. *Transactions on Machine Learning Research*, 2024.

Anthony Brohan, Noah Brown, Justice Carbajal, Yevgen Chebotar, Xi Chen, Krzysztof Choromanski, Tianli Ding, Danny Driess, Avinava Dubey, Chelsea Finn, et al. Rt-2: Vision-language-action models transfer web knowledge to robotic control. *arXiv preprint arXiv:2307.15818*, 2023.

Jake Bruce, Ankit Anand, Bogdan Mazoure, and Rob Fergus. Learning about progress from experts. In *Proc. of International Conference on Learning Representations*, 2023.

Elliot Chane-Sane, Cordelia Schmid, and Ivan Laptev. Learning video-conditioned policies for unseen manipulation tasks. In *Proc. of International Conference on Robotics and Automation*, 2023.

Matthew Chang, Arjun Gupta, and Saurabh Gupta. Learning value functions from undirected state-only experience. In *Proc. of International Conference on Learning Representations*, 2022.

Lawrence Yunliang Chen, Kush Hari, Karthik Dharmarajan, Chenfeng Xu, Quan Vuong, and Ken Goldberg. Mirage: Cross-embodiment zero-shot policy transfer with cross-painting. *arXiv preprint arXiv:2402.19249*, 2024.

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. In *Proc. of Neural Information Processing Systems*, 2021.

Sudeep Dasari and Abhinav Gupta. Transformers for one-shot visual imitation. In *Proc. of Conference on Robot Learning*, 2021.

Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. In *Proc. of International Conference on Learning Representations*, 2021.

Yilun Du, Sherry Yang, Bo Dai, Hanjun Dai, Ofir Nachum, Josh Tenenbaum, Dale Schuurmans, and Pieter Abbeel. Learning universal policies via text-guided video generation. In *Proc. of Neural Information Processing Systems*, 2024.

Yan Duan, Marcin Andrychowicz, Bradly Stadie, OpenAI Jonathan Ho, Jonas Schneider, Ilya Sutskever, Pieter Abbeel, and Wojciech Zaremba. One-shot imitation learning. In *Proc. of Neural Information Processing Systems*, 2017.

Ashley Edwards, Himanshu Sahni, Yannick Schroecker, and Charles Isbell. Imitating latent policies from observation. In *Proc. of International Conference on Machine Learning*, 2019.

Alejandro Escontrela, Ademi Adeniji, Wilson Yan, Ajay Jain, Xue Bin Peng, Ken Goldberg, Youngwoon Lee, Danijar Hafner, and Pieter Abbeel. Video prediction models as rewards for reinforcement learning. In *Proc. of Neural Information Processing Systems*, 2023.

Chelsea Finn, Tianhe Yu, Tianhao Zhang, Pieter Abbeel, and Sergey Levine. One-shot visual imitation learning via meta-learning. In *Proc. of Conference on Robot Learning*, 2017.

Hiroki Furuta, Yutaka Matsuo, and Shixiang Shane Gu. Generalized decision transformer for offline hindsight information matching. In *Proc. of International Conference on Learning Representations*, 2022.

Zhaoyang Huang, Xiaoyu Shi, Chao Zhang, Qiang Wang, Ka Chun Cheung, Hongwei Qin, Jifeng Dai, and Hongsheng Li. Flowformer: A transformer architecture for optical flow. In *Proc. of European Conference on Computer Vision*, 2022.

Andrew Jaegle, Sebastian Borgeaud, Jean-Baptiste Alayrac, Carl Doersch, Catalin Ionescu, David Ding, Skanda Koppula, Daniel Zoran, Andrew Brock, Evan Shelhamer, et al. Perceiver io: A general architecture for structured inputs & outputs. In *Proc. of International Conference on Learning Representations*, 2022.

Vidhi Jain, Yixin Lin, Eric Undersander, Yonatan Bisk, and Akshara Rai. Transformers are adaptable task planners. In *Proc. of Conference on Robot Learning*, 2023.

Vidhi Jain, Maria Attarian, Nikhil J Joshi, Ayzaan Wahid, Danny Driess, Quan Vuong, Pannag R Sanketi, Pierre Sermanet, Stefan Welker, Christine Chan, et al. Vid2robot: End-to-end video-conditioned policy learning with cross-attention transformers. *arXiv preprint arXiv:2403.12943*, 2024.

Eric Jang, Alex Irpan, Mohi Khansari, Daniel Kappler, Frederik Ebert, Corey Lynch, Sergey Levine, and Chelsea Finn. Bc-z: Zero-shot task generalization with robotic imitation learning. In *Proc. of Conference on Robot Learning*, 2022.

Yiding Jiang, Shixiang Shane Gu, Kevin P Murphy, and Chelsea Finn. Language as an abstraction for hierarchical deep reinforcement learning. In *Proc. of Neural Information Processing Systems*, 2019.

Yunfan Jiang, Agrim Gupta, Zichen Zhang, Guanzhi Wang, Yongqiang Dou, Yanjun Chen, Li Fei-Fei, Anima Anandkumar, Yuke Zhu, and Linxi Fan. Vima: General robot manipulation with multimodal prompts. In *Proc. of International Conference on Machine Learning*, 2023.

Dong-Hee Kim, Sungduk Cho, Hyeonwoo Cho, Chanmin Park, Jinyoung Kim, and Won Hwa Kim. Joint-embedding predictive architecture for self-supervised learning of mask classification architecture. *arXiv preprint arXiv:2407.10733*, 2024.

Moo Jin Kim, Jiajun Wu, and Chelsea Finn. Giving robots a hand: Learning generalizable manipulation with eye-in-hand human video demonstrations. *arXiv preprint arXiv:2307.05959*, 2023.

Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C Berg, Wan-Yen Lo, et al. Segment anything. In *Proc. of International Conference on Computer Vision*, 2023.

Yann LeCun. A path towards autonomous machine intelligence version 0.9. 2, 2022-06-27. 2022.

Kuang-Huei Lee, Ofir Nachum, Mengjiao Sherry Yang, Lisa Lee, Daniel Freeman, Sergio Guadarrama, Ian Fischer, Winnie Xu, Eric Jang, Henryk Michalewski, et al. Multi-game decision transformers. In *Proc. of Neural Information Processing Systems*, 2022.

Lisa Lee, Ben Eysenbach, Russ R Salakhutdinov, Shixiang Shane Gu, and Chelsea Finn. Weakly-supervised reinforcement learning for controllable behavior. In *Proc. of Neural Information Processing Systems*, 2020.

Siyuan Li, Shijie Han, Yingnan Zhao, By Liang, and Peng Liu. Auxiliary reward generation with transition distance representation learning. *arXiv preprint arXiv:2402.07412*, 2024.

Ajay Mandlekar, Soroush Nasiriany, Bowen Wen, Iretiayo Akinola, Yashraj Narang, Linxi Fan, Yuke Zhu, and Dieter Fox. Mimicgen: A data generation system for scalable robot learning using human demonstrations. In *Proc. of Conference on Robot Learning*, 2023.

Ashvin V Nair, Vitchyr Pong, Murtaza Dalal, Shikhar Bahl, Steven Lin, and Sergey Levine. Visual reinforcement learning with imagined goals. In *Proc. of Neural Information Processing Systems*, 2018.

Suraj Nair, Aravind Rajeswaran, Vikash Kumar, Chelsea Finn, and Abhinav Gupta. R3m: A universal visual representation for robot manipulation. In *Proc. of Conference on Robot Learning*, 2023.

Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy V. Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel HAZIZA, Francisco Massa, Alaaeldin El-Nouby, Mido Assran, Nicolas Ballas, Wojciech Galuba, Russell Howes, Po-Yao Huang, Shang-Wen Li, Ishan Misra, Michael Rabbat, Vasu Sharma, Gabriel Synnaeve, Hu Xu, Herve Jegou, Julien Mairal, Patrick Labatut, Armand Joulin, and Piotr Bojanowski. DINOv2: Learning robust visual features without supervision. *Transactions on Machine Learning Research*, 2024.

Abhishek Padalkar and et al. Open x-embodiment: Robotic learning datasets and rt-x models. *arXiv preprint arXiv:2310.08864*, 2023.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. Pytorch: An imperative style, high-performance deep learning library. In *Proc. of Neural Information Processing Systems*, 2019.

Ilija Radosavovic, Tete Xiao, Stephen James, Pieter Abbeel, Jitendra Malik, and Trevor Darrell. Real-world robot learning with masked visual pre-training. In *Proc. of Conference on Robot Learning*, 2023.

Karl Schmeckpeper, Oleh Rybkin, Kostas Daniilidis, Sergey Levine, and Chelsea Finn. Reinforcement learning with videos: Combining offline observations with interaction. In *Proc. of Conference on Robot Learning*, 2021.

Dominik Schmidt and Minqi Jiang. Learning to act without actions. In *Proc. of International Conference on Learning Representations*, 2024.

Younggyo Seo, Kimin Lee, Stephen L James, and Pieter Abbeel. Reinforcement learning with action-free pre-training from videos. In *Proc. of International Conference on Machine Learning*, 2022.

Younggyo Seo, Danijar Hafner, Hao Liu, Fangchen Liu, Stephen James, Kimin Lee, and Pieter Abbeel. Masked world models for visual control. In *Proc. of Conference on Robot Learning*, 2023.

Rutav Shah, Roberto Martín-Martín, and Yuke Zhu. MUTEX: Learning unified policies from multimodal task specifications. In *Proc. of Conference on Robot Learning*, 2023.

Sangwoo Shin, Daehee Lee, Minjong Yoo, Woo Kyung Kim, and Honguk Woo. One-shot imitation in a non-stationary environment via multi-modal skill. In *Proc. of International Conference on Machine Learning*, 2023.

Sangwoo Shin, Minjong Yoo, Jeongwoo Lee, and Honguk Woo. Semtra: A semantic skill translator for cross-domain zero-shot policy adaptation. In *Proc. of Association for the Advancement of Artificial Intelligence*, 2024.

Aravind Sivakumar, Kenneth Shaw, and Deepak Pathak. Robotic telekinesis: Learning a robotic hand imitator by watching humans on youtube. In *Proc. of Robotics: Science and Systems*, 2022.

Limin Wang, Bingkun Huang, Zhiyu Zhao, Zhan Tong, Yinan He, Yi Wang, Yali Wang, and Yu Qiao. Videomae v2: Scaling video masked autoencoders with dual masking. In *Proc. of Computer Vision and Pattern Recognition*, 2023.

Tete Xiao, Ilija Radosavovic, Trevor Darrell, and Jitendra Malik. Masked visual pre-training for motor control. *arXiv preprint arXiv:2203.06173*, 2022.

Jiange Yang, Bei Liu, Jianlong Fu, Bocheng Pan, Gangshan Wu, and Limin Wang. Spatiotemporal predictive pre-training for robotic motor control. *arXiv preprint arXiv:2403.05304*, 2024.

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

Weirui Ye, Yunsheng Zhang, Pieter Abbeel, and Yang Gao. Become a proficient player with limited data through watching pure videos. In *Proc. of International Conference on Learning Representations*, 2022.

Sriram Yenamandra, Arun Ramachandran, Karmesh Yadav, Austin S. Wang, Mukul Khanna, Theophile Gervet, Tsung-Yen Yang, Vidhi Jain, Alexander Clegg, John M. Turner, Zsolt Kira, Manolis Savva, Angel X. Chang, Devendra Singh Chaplot, Dhruv Batra, Roozbeh Mottaghi, Yonatan Bisk, and Chris Paxton. Homerobot: Open-vocabulary mobile manipulation. In *Proc. of Conference on Robot Learning*, 2023.

Tianhe Yu, Pieter Abbeel, Sergey Levine, and Chelsea Finn. One-shot hierarchical imitation learning of compound visuomotor tasks. *arXiv preprint arXiv:1810.11043*, 2018.

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 *Proc. of Conference on Robot Learning*, 2020a.

Xingrui Yu, Yueming Lyu, and Ivor Tsang. Intrinsic reward driven imitation learning via generative model. In *Proc. of International Conference on Machine Learning*, 2020b.

Qihang Zhang, Zhenghao Peng, and Bolei Zhou. Learning to drive by watching youtube videos: Action-conditioned contrastive policy pretraining. In *Proc. of European Conference on Computer Vision*, 2022.

Qinqing Zheng, Mikael Henaff, Brandon Amos, and Aditya Grover. Semi-supervised offline reinforcement learning with action-free trajectories. In *Proc. of International Conference on Machine Learning*, 2023.

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 preprint arXiv:2009.12293*, 2020.