OSVI-WM: One-Shot Visual Imitation for Unseen Tasks using World-Model-Guided Trajectory Generation

Raktim Gautam Goswami[1] **Prashanth Krishnamurthy**[1] **Yann LeCun**[2,3] **Farshad Khorrami**[1]

[1]New York University Tandon School of Engineering
[2]New York University Courant Institute of Mathematical Sciences [3]Meta-FAIR

Abstract

Visual imitation learning enables robotic agents to acquire skills by observing expert demonstration videos. In the one-shot setting, the agent generates a policy after observing a single expert demonstration without additional fine-tuning. Existing approaches typically train and evaluate on the same set of tasks, varying only object configurations, and struggle to generalize to unseen tasks with different semantic or structural requirements. While some recent methods attempt to address this, they exhibit low success rates on *hard* test tasks that, despite being visually similar to some training tasks, differ in context and require distinct responses. Additionally, most existing methods lack an explicit model of environment dynamics, limiting their ability to reason about future states. To address these limitations, we propose a novel framework for one-shot visual imitation learning via world-model-guided trajectory generation. Given an expert demonstration video and the agent's initial observation, our method leverages a learned world model to predict a sequence of latent states and actions. This latent trajectory is then decoded into physical waypoints that guide the agent's execution. Our method is evaluated on two simulated benchmarks and three real-world robotic platforms, where it consistently outperforms prior approaches, with over 30% improvement in some cases. The code is available at https://github.com/raktimgg/osvi-wm.

1 Introduction

> *Imitation is the sincerest of flattery.*
> — Charles Caleb Colton[1]

Intelligent beings like humans are capable of learning a wide range of skills by observing and imitating others [50]. Even toddlers, with a basic understanding of the world's dynamics, solve complex tasks by watching *expert demonstrations*. Inspired by this natural learning process, principles have been applied in robotics [13, 26, 35, 62] where expert demonstrations, typically collected through human teleoperation, are used to teach agents to perform tasks through imitation. Imitation learning finds uses across diverse domains, including medical robotics, collaborative robotics, and industrial automation.

In a successful imitation, the agent discovers the skill demonstrated by the expert, adapts it to its own embodiment, and executes it within its environment [58]. In the one-shot visual imitation (OSVI) setting, the agent must derive a policy from a single expert demonstration video, without any additional training [12]. Notably, the expert and agent may differ in embodiment [10]; in fact, both prior work [51, 43, 37] and our approach have utilized human demonstration videos as input. Existing methods, however, often rely on the strong assumption that the training and testing tasks are nearly identical, typically differing only in object locations. As a result, generalization

---

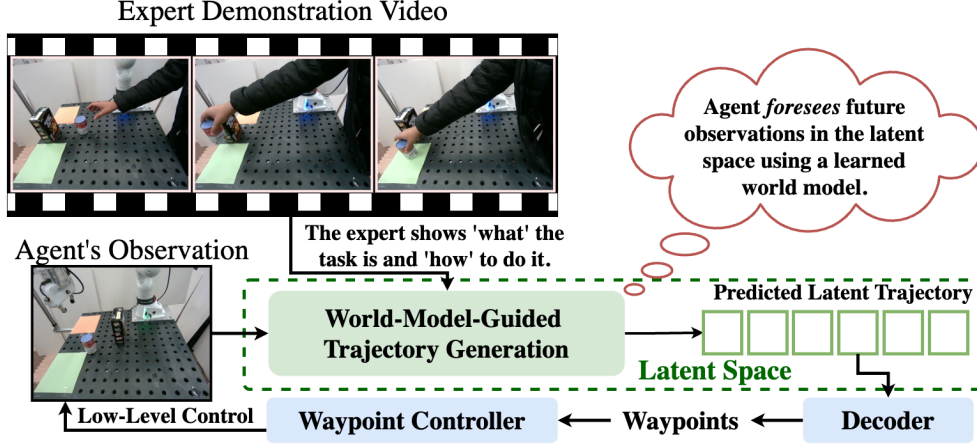[1]*Lacon: Or Many Things in Few Words*, 1820.

Figure 1: OSVI-WM infers the task from the expert demonstration and, along with the agent's observation "foresees" future latent states using a world-model-guided trajectory generation module. The predicted trajectory is decoded into physical waypoints for control.

to semantically or structurally different tasks remains limited. MOSAIC [34] and AWDA [5] were among the first to investigate generalization to unseen tasks. In particular, AWDA [5] demonstrated that the performance of standard methods like [10, 60] degrades severely when evaluated on tasks that differ even marginally from those seen during training. Although AWDA outperforms earlier methods [10, 60], it exhibits low success rates on test tasks that are visually similar but contextually different, such as opening vs. closing a sliding window. Moreover, none of these approaches incorporates an explicit model of the environment's dynamics, limiting their ability to reason about long-term consequences or plan into the future.

Mathematically modeling the environment using raw camera images is challenging due to the presence of diverse objects and their complex interactions. To address this, neural networks have been used to learn world models that predict how the environment will evolve in response to the agent's actions. These models support applications in reinforcement learning [48, 21, 28, 39, 6, 23, 56] and behavior cloning [8, 37]. Inspired by such advances, we propose *OSVI-WM* (Fig. 1), a framework for one-shot visual imitation for unseen tasks using world-model-guided trajectory generation. Given an expert demonstration video and the agent's initial observation, OSVI-WM encodes each image into a shared latent space representing the environment state for the respective image. These embeddings are recursively processed through learnable action and world models to predict a trajectory of future latent states, allowing the agent to plan into the future. Predicting actions in the latent space also helps mitigate the multi-modality of real-world action distributions. The predicted latent trajectory is then decoded into physical waypoints [5] using pooling and Multi-Layer Perceptron (MLP) layers.

The predicted waypoints guide the agent's execution and provide supervision for end-to-end training. To enhance the world model's ability to capture environment dynamics, an auxiliary loss is applied on the predicted latent trajectory during training. Further, OSVI-WM supports re-planning: if the agent doesn't accurately complete the task by following the predicted waypoints, it takes a new observation and plans a fresh set of waypoints based on its current state. Notably, unlike methods that rely on large-scale pretraining or extensive demonstration data [37, 3, 55], OSVI-WM is trained directly within the task domain. In summary, our main contributions are:

- An efficient end-to-end imitation learning architecture trained solely on in-domain data, without requiring large-scale pretraining.

- A novel world-model-guided trajectory generation module tailored for OSVI on unseen tasks.

- Robustness enhancement at test time by using a waypoint controller with re-planning.

- Extensive experiments in both simulated and real-world settings, demonstrating that OSVI-WM outperforms existing methods on unseen tasks.

2 Related Works

One-Shot Imitation Learning: Imitation learning, or learning from demonstrations, is broadly categorized into inverse reinforcement learning [1, 40] and behavior cloning [46, 31, 7, 22, 42]. While the field has been extensively studied for years, a comprehensive review is beyond the scope of this paper; we, therefore, refer readers to existing surveys [13, 35, 62]. Our method falls under behavior cloning, specifically in the one-shot visual imitation (OSVI) setting. In standard behavior cloning, expert demonstrations and corresponding expert actions are available to train the agent. In contrast, OSVI [12] provides only a video of the expert performing the task, without access to the underlying actions. One of the earliest works on OSVI [12] introduced the problem setting and proposed a soft-attention-based learning framework for block stacking. Since then, a number of methods [15, 10, 4, 63, 27] have improved performance in this setting. While ego-centric setups (e.g., DINOBot [11]) have been explored, we focus on methods using external camera views. One such work, T-OSVI [10], uses transformers [49] to model long-range dependencies in demonstrations, and like many other works [29, 51, 60], it supports expert-agent embodiment mismatch.

While these approaches have shown promising results, they are typically evaluated on the same tasks that were used for training with minor variations, such as changes in object positions or quantities. MOSAIC [34] proposed a more challenging setup with completely unseen test tasks, building on which, AWDA [5] demonstrated that existing methods like T-OSVI [10] and DAML [60] perform poorly in such settings. AWDA improved generalization through attributed waypoints, demonstration augmentation, and image mixup, but achieved low success rates on *hard* tasks requiring different responses despite visual similarity. While IMOP [63] used point clouds to aid generalization, we focus solely on monocular RGB images, which are more accessible and require no specialized hardware. Notably, none of these approaches models environment dynamics, limiting their ability to reason about future states. OSVI-WM addresses this gap through a world-model-guided trajectory generation module that enables future-aware planning and improved generalization to unseen tasks.

World Model in Robotics: Intelligent beings like animals are believed to possess internal models of the world that support control and planning for task execution [52, 53, 16, 38]. Inspired by this, researchers have increasingly explored the use of world models in robotics [45]. In reinforcement learning (RL), several works [17, 28, 19–21, 18, 39, 56, 6, 44, 14] have focused on learning world models from agent rollouts. These models are then used as environments for RL policy training, often also predicting rewards based on the agent's actions. Most prior works train world models directly in the pixel space, i.e., conditioned on the current image and action, the model predicts the next-frame image, often also integrated with diffusion-based frameworks. These world models, however, requires modeling low-level pixel details that are often irrelevant for downstream tasks [2]. To address this, recent methods [24, 23, 36, 30] learn world models in a latent embedding space, allowing for more efficient and abstract representations of the environment.

Beyond RL, world models have been explored for broader robotic learning tasks. For instance, SWIM [37] pre-trains a world model from human videos and fine-tunes it on robot data in an unsupervised manner. GR1 [54] and [59] propose pre-training strategies to improve downstream robotic motor control. Most of these approaches, however, rely on large-scale internet data or massive pretraining datasets, leading to high computational demands. DynaMo [8], in contrast, proposed in-domain latent dynamics pretraining for the encoder, which was used to train downstream policies. Similarly, DINO-WM [64] used the pre-trained DINOv2 [41] model (with frozen weights) to build in-context world models and applied model predictive control based optimization for task execution. Inspired by these advances, our proposed framework, OSVI-WM, learns a world model entirely from in-domain data. It predicts latent trajectories based on an expert demonstration and the agent's current observation. Unlike prior work, OSVI-WM, including its world model, is trained end-to-end, without any pretraining, simplifying the pipeline and reducing computational complexity.

3 Our Method

Problem Formulation: We formulate the problem following prior work in OSVI [10, 34, 5]. Let $\mathcal{T} = \{\mathcal{T}_1, \ldots, \mathcal{T}_K\}$ denote a set of tasks, partitioned into disjoint training ($\mathcal{T}_{train}$) and test ($\mathcal{T}_{test}$) sets. In contrast to [10] and consistent with [5], our setup uses different training and test tasks, rather than minor variations of the same task. Each training task consists of multiple sequences of expert and agent trajectories. The expert trajectory comprises demonstration video frames $\{E_1, \ldots, E_N\}$, while
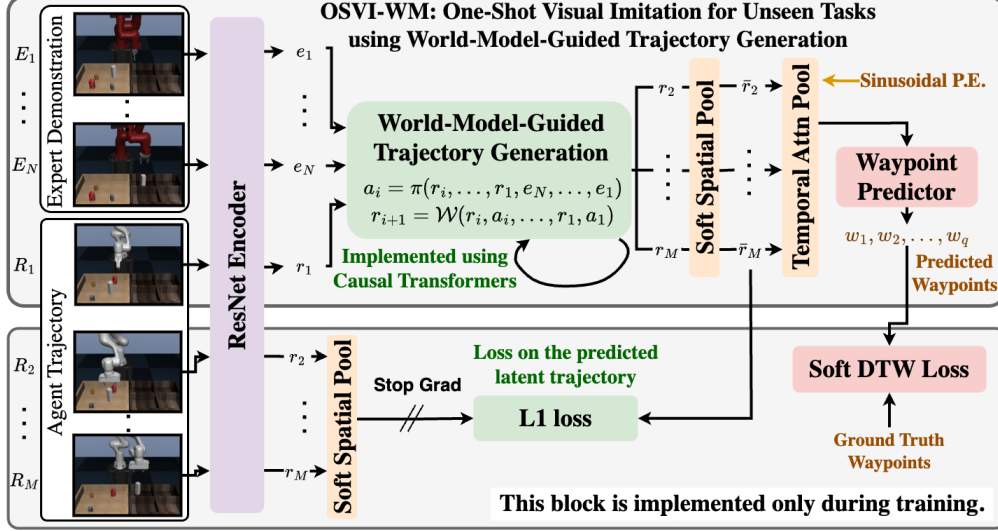
Figure 2: **OSVI-WM**: The expert demonstration frames $E_1, \ldots, E_N$ and the agent's current observation R_1 are encoded into a latent space using a ResNet encoder. A world-model-guided trajectory generation module predicts future latent states $r_2, \ldots, r_M$, which are decoded into physical waypoints for control. During training, supervision is applied on the predicted waypoints and latent states.

the agent trajectories are represented by observation-state pairs $(\mathcal{R}, \mathcal{S})$ where $\mathcal{R} = \{R_1, \ldots, R_M\}$ denotes the observation frames and $\mathcal{S}$ denotes the corresponding agent states. Within each task, all expert and agent trajectories correspond to variations (e.g., different object configurations) of the same high-level task. The model, trained on $\mathcal{T}_{train}$, is evaluated on $\mathcal{T}_{test}$, which contains only expert demonstration videos (no agent rollouts). The performance is measured by recording the success rate of the agent in completing these test tasks. The key challenges in this setup are: (a) test tasks $\mathcal{T}_{test}$ differ from the training tasks $\mathcal{T}_{train}$; (b) the expert and the agent may have different embodiments; (c) there is no direct alignment between expert and agent states or actions; and (d) to avoid reliance on costly large-scale data, the method should be trained in-domain.

Method Overview: Given expert demonstration video frames $E_1, \ldots, E_N$ and the agent's initial observation R_1, OSVI-WM (Fig. 2) encodes them using a shared ResNet-18 [25] encoder to obtain latent states for the expert and agent. These are passed to the world-model-guided trajectory generation module (Sec. 3.1) to predict a trajectory of future latent states, which is pooled, first spatially, then temporally, into a compact representation. Using this as input, an MLP-based waypoint predictor generates physical waypoints (Sec. 3.2). Loss functions are applied to the predicted waypoints and the latent states predicted by the world model (Sec. 3.3).

3.1 World-Model-Guided Trajectory Generation

The expert demonstration and the agent's initial observation frames are each processed through a ResNet-18 [25] encoder, producing feature maps of shape (F, H, W) before the encoder's global pooling layer, where $F = 512$ and (H, W) depend on the input resolution. These features are flattened to produce latent states for the expert frames $(e_1, \ldots, e_N)$ and the agent's observation (r_1), each frame with shape $(F \times H \times W)$. A latent state is a compact, abstract representation of the environment that encodes key features such as the configuration of objects and dynamic elements. A latent action similarly represents an abstract transformation that drives state transitions. Operating in this latent space further helps mitigate the multi-modality of real-world action distributions. The
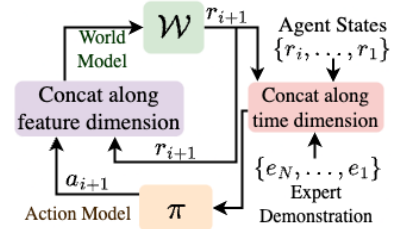

Figure 3: **WM-Guided Trajectory Generation**: Starting with the agent's initial observation and the expert demonstration, future states are recursively predicted using action and world models.

world-model-guided trajectory generation module consists of an action model π and world model $\mathcal{W}$. At timestep i, π predicts latent action a_i using the agent's current and past latent states along with the

expert demonstration's latent states. This action is used by $\mathcal{W}$ to predict the agent's next latent state:

$$a_i = \pi(r_i, \ldots, r_1, e_N, \ldots, e_1), \quad r_{i+1} = \mathcal{W}(r_i, a_i, \ldots, r_1, a_1); \quad i = \{1, \ldots, M-1\}. \quad (1)$$

This multi-step recursive process (Fig. 3) generates all future latent states of the agent $(r_2, \ldots, r_M)$, forming the generated trajectory. The world model (WM) essentially learns the environment's dynamics to predict the next latent state based on the current latent state and latent action.

Within the action model π, at each step i, the expert states $\{e_N, \ldots, e_1\}$ and agent states $\{r_i, \ldots, r_1\}$ are concatenated along the temporal dimension, resulting in shape $(N + i, F \times H \times W)$. This is processed through causal transformer blocks, and the output corresponding to the final temporal index is used as the latent action a_i. For the world model $\mathcal{W}$, the latent actions $\{a_i, \ldots, a_1\}$ are concatenated with the corresponding agent states $\{r_i, \ldots, r_1\}$ along the feature dimension to obtain shape $(i, 2F \times H \times W)$. This sequence is processed through a causal transformer, and the output at the last time step is projected to shape $(F \times H \times W)$ to produce the next state r_{i+1}.

3.2 Predicting Waypoints

Spatial and Temporal Pooling: The predicted trajectory of future states $(r_2, \ldots, r_M)$ is reshaped back to shape (F, H, W) at each timestep to retain spatial structure. A spatial pooling module [10] then applies a 2D softmax across each feature map and computes an expected 2D coordinate over a $[-1, 1]$ grid for each feature dimension. These coordinates are concatenated, producing a spatially pooled output of shape $2F$ for each of the $M - 1$ timesteps $(\bar{r}_2, \ldots, \bar{r}_M)$. Sinusoidal positional embeddings are then added along the temporal dimension, and an attention pooling module with a learned query summarizes the sequence into a single $2F$-dimensional vector.

Attributed Waypoint Prediction: The pooled vector is passed through a two-layer MLP to predict 5 attributed waypoints, each of dimension 4. Following [5], each waypoint encodes the end-effector's 3D position in the camera's coordinate frame and a binary gripper state (open/closed). These positions are transformed into the robot's coordinate frame using a known camera-to-robot transformation. A waypoint controller, using inverse kinematics of the robot, computes low-level control commands for execution. To ensure accurate grasping, especially in the final centimeters before contact, we follow [5] and use an end-effector-mounted depth camera to correct residual pose errors.

Waypoint Re-Planning: To improve robustness during execution, OSVI-WM supports waypoint re-planning, which activates if the agent fails to complete the task using the predicted waypoints. In simulation, success or failure is automatically determined from environment configurations. Although we do not use re-planning on real-world benchmarks in this work, success (or completion) signals could be obtained via learned reward/value models [24] or vision–language-based completion detectors [32, 57]. In OSVI-WM, once all waypoints are executed, the agent captures a new observation, which, combined with the original expert demonstration, generates updated waypoints. Notably OSVI-WM outperforms existing baselines even without re-planning (Fig. 8).

3.3 OSVI-WM Training

Loss Functions: We jointly optimize the world model and the waypoint predictions. To supervise the world model, ground-truth agent observations $R_2, \ldots, R_M$ from the training set are encoded into latent states using our encoder, followed by the spatial pooling layer to produce $\hat{r}_2, \ldots, \hat{r}_M$. An L_1 loss is applied between these and the predicted latent states after spatial pooling $(\bar{r}_2, \ldots, \bar{r}_M)$: $\mathcal{L}_{wm} = \frac{1}{M-1} \sum_{i=2}^{M} \|\bar{r}_i - \hat{r}_i\|_1$. To prevent model collapse and ensure stable training, we apply a stop-gradient to $\hat{r}_2, \ldots, \hat{r}_M$, preventing backpropagation through the ground-truth latent states. For waypoint supervision, we interpolate the ground-truth agent trajectory into a denser sequence and use the differentiable soft dynamic time-warping loss [9] ($\mathcal{L}_{sdtw}$). Although only 5 waypoints are used at inference, we follow [5] and predict multiple sets (1 to 5 waypoints) during training, applying separate losses for each set to improve supervision across different trajectory granularities. The overall training objective is $\mathcal{L} = \mathcal{L}_{sdtw} + \alpha(\tau)\mathcal{L}_{wm}$ where $\alpha(\tau)$ is a scheduling factor that balances the two loss terms based on the training iteration τ. For the Meta-World [61] benchmark experiments in Sec. 4, $\alpha(\tau)$ is initialized to 1 and decays exponentially to 0.05 by the end of training. For Human-Franka-PP, $\alpha(\tau) = 10$; for all other datasets, we keep $\alpha(\tau) = 1$.

Training Settings: Each expert demonstration is downsampled to $N = 10$ frames, and each agent trajectory to $M = 6$ frames. Following prior work [10, 5], we pair every expert demonstration
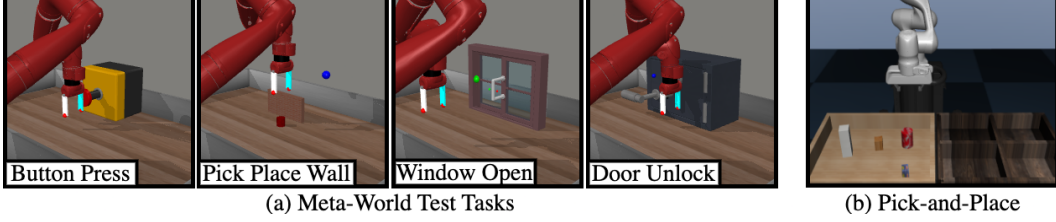
(a) Meta-World Test Tasks   (b) Pick-and-Place

Figure 4: **Simulation Environments**: Test tasks are different from the ones used for training. Additionally, Pick-and-Place uses different embodiments for expert and agent.

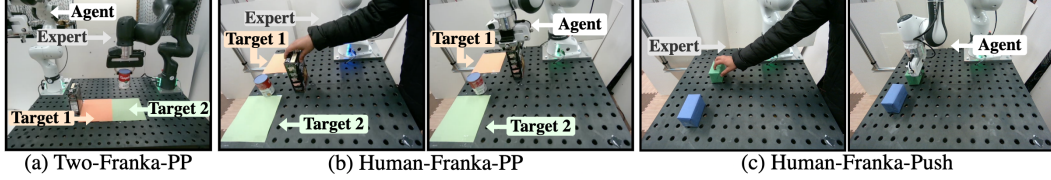(a) Two-Franka-PP   (b) Human-Franka-PP   (c) Human-Franka-Push

Figure 5: **Real-World Environments**: (a) Pick-and-place setup with expert (gray) and agent (white) Franka arms mounted at different locations. (b) Similar setup with a human expert and Franka agent. (c) Box push setup with human expert and Franka agent.

with every agent trajectory within the same task, resulting in a combinatorial dataset expansion. We train for 10 epochs in simulation and 100 in real-world settings using AdamW [33] optimizer with a one-cycle learning rate scheduler [47] (start: 0.0002, end: 2×10^{-7}), and a batch size of 128. We apply asymmetric demonstration image mixup [5] regularization to help the model focus on task-relevant cues rather than video semantics. Although training runs for 10 epochs, we use early stopping for Meta-World to avoid overfitting and report results using the overall best-performing checkpoint. For simulation runs, we used an RTX A4000 GPU with 128 GB RAM and Intel i9 CPU, and for real-world, we used an RTX 2080 Ti, 64 GB RAM, and Intel i7 CPU.

4 Experiments

Our experiments are designed to address the following key questions: (a) How well does OSVI-WM generalize to novel, unseen tasks? (b) Can OSVI-WM be used for real-world robotic tasks? (c) What is the impact of the world-model-guided trajectory generation module on overall performance? (d) Which components of OSVI-WM are most critical to its success?

4.1 Evaluation Settings

Environments: We evaluate OSVI-WM on simulation (Fig. 4) and real-world benchmarks (Fig. 5):

(a) **Meta-World** [61]: A simulation benchmark of 50 manipulation tasks, split into 46 for training and 4 for testing following [5]. Test tasks are different from train tasks and are divided into 'easy' tasks (Button-Press-V2, Pick-Place-Wall-V2), which differ from training tasks due to the presence/absence of distractor, and 'hard' tasks Window-Open-V2, Door-Unlock-V2, which, despite being visually similar to Window-Close-V2 and Door-Lock-V2 from the training set, respectively, differ in context and require distinct responses.

(b) **Pick-and-Place** [10]: A simulated benchmark involving 4 objects and 4 targets, yielding 16 tasks. Following [5], we use 14 tasks for training and 2 for testing. This benchmark features different embodiments: the expert uses a Sawyer arm, and the agent uses a Franka arm.

(c) **Two-Franka-PP** (Fig. 5a): A real-world setup with two Franka arms mounted at different tabletop locations, with the gray arm as the expert and the white arm as the agent. The tasks are similar to Pick-and-Place above, with two objects and two targets (4 tasks total). We use 3 tasks for training and 1 for testing. The difference in mounting location between the expert and agent introduces additional imitation complexity.

(d) **Human-Franka-PP** (Fig. 5b): Similar to Two-Franka-Env, but the expert is a human arm and the agent is a Franka robot. This greater embodiment mismatch increases task difficulty. As before, we use 3 tasks for training and 1 for testing.

Table 1: Success rates (in %) comparison on the Meta-World [61] and Pick-and-Place [10] simulation benchmarks. Best results are highlighted. We also report if a method uses additional training data.

Method	Add. Data	Pick Place	Meta-World		
			Easy	Hard	All
DAML [60]	No	1	4	8	6
T-OSVI [10]	No	10	50	7	28.5
AWDA [5]	No	**100**	74	11	42.5
AWDA [5]	Yes	**100**	73	30	51.5
OSVI-WM (Ours)	No	**100**	**96**	**71.5**	**83.8**

Table 2: Real-World experiments: Success rates and execution breakdowns (in %) are reported. T-OSVI* denotes T-OSVI [10] aided with end-effector depth sensing for improved grasping.

Method	Two-Franka-PP			Human-Franka-PP			Human-Franka-Push	
	Overall Success	Breakdown		Overall Success	Breakdown		Overall Success	Breakdown
		Reach Obj.	Grasp Obj.		Reach Obj.	Grasp Obj.		Reach Obj.
T-OSVI [10]	0	52	0	0	0	0	16	96
T-OSVI* [10]	4	52	36	0	0	0	-	-
AWDA [5]	88	88	88	**92**	92	92	76	100
OSVI-WM (Ours)	**96**	96	96	**92**	92	92	**100**	100

(e) **Human-Franka-Push** (Fig. 5c): This features two colored boxes on a tabletop. The task involves selecting a box and pushing it either forward or backward (four tasks total). We train on three and test on the held-out one. The expert is a human and agent is a Franka arm.

In the pick-and-place tasks above, sequences from test object-target pairs are excluded from training. While this may appear to be an extension of training tasks, most prior methods require training on all pairs to succeed. Generalizing to an unseen pair requires the model to truly follow the expert, rather than rely solely on scene semantics. Additional environmental details are in the appendix.

Baselines: As noted in Sec. 1 and Sec. 2, most one-shot visual imitation methods are evaluated on tasks that are minor variations of training tasks. AWDA [5] is the only method that uses the same sensors and environment settings as ours and evaluates on unseen test tasks, making it our primary baseline. We also compare against two widely used methods: DAML [60] and T-OSVI [10]. In real-world experiments, T-OSVI fails to achieve successful grasps. To mitigate this, we augment it with the same depth-based gripper correction used in OSVI-WM and AWDA, where an end-effector-mounted depth camera corrects residual pose errors. We refer to this variant as T-OSVI* and report its results in Table 2. For OSVI-WM, re-planning is used only on the MetaWorld 'hard' tasks, as we observe strong performance even without it on all other benchmarks.

4.2 Results

How well does OSVI-WM generalize to novel, unseen tasks? We compare OSVI-WM against baseline methods in Table 1 on the Pick-and-Place [10] and Meta-World [61] benchmarks. OSVI-WM is evaluated over 100 rollouts per task with varied object configurations, and success rates are reported. Baseline results are reported from [5]. OSVI-WM outperforms prior methods, achieving success rates up to 30% higher than AWDA even when AWDA uses additional training data. OSVI-WM's ability to predict future latent states enables effective planning, resulting in strong performance even on the Meta-World 'hard' tasks. Failure cases in this setting often occurred in edge cases where objects, such as windows or doors, were positioned far from the agent, making manipulation challenging due to depth ambiguity in external RGB observations. Nonetheless, OSVI-WM achieves a 71.5% success rate on these tasks, outperforming all baselines.

Can OSVI-WM be used for real-world robotic tasks? We evaluate OSVI-WM and baselines on the real-world setups, each tested over 25 rollouts using the same 25 object configurations across methods for fair comparison (Table 2). These setups introduce added complexity: in Two-Franka-PP, the expert and agent robots are mounted at different locations, while in Human-Franka-PP and Human-Franka-Push, the expert is a human. Despite these challenges, OSVI-WM achieves consistently high success rates, showing strong real-world applicability. The table also breaks performance into sub-tasks like
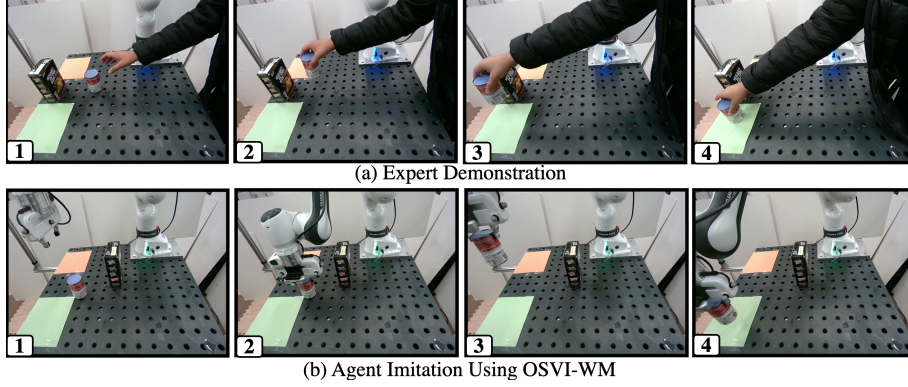
(a) Expert Demonstration

(b) Agent Imitation Using OSVI-WM

Figure 6: **Real-World Qualitative Example**: The human expert demonstrates the test task of picking and placing a can in the green target, which the agent successfully imitates using OSVI-WM.
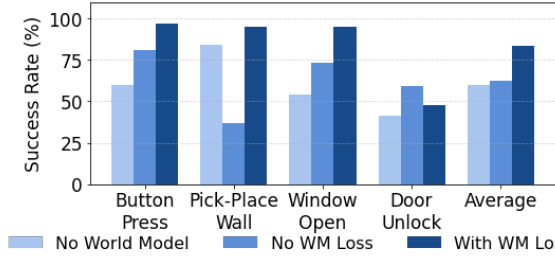

Figure 7: Ablation study on the world model (WM) and WM loss. Using both achieves the best performance.
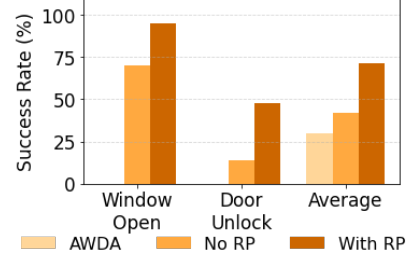

Figure 8: Ablation Study on OSVI-WM's Re-Planning (RP).

reaching and grasping. In Two-Franka-PP, T-OSVI [10] reaches the object in 52% of trials but fails to grasp it. T-OSVI* (i.e., T-OSVI with grasp correction) improves grasping to 36% but completes the full task only once (4%). AWDA succeeds in 88% of the trials, failing due to incorrect object selection. OSVI-WM performs best, succeeding in 96% of the trials. Human-Franka-PP is more challenging due to the embodiment gap. In this setting, T-OSVI and T-OSVI* fail to reach the correct object, while AWDA and OSVI-WM both succeed in 92 % of the trials. In Human-Franka-Push, all methods consistently reach the correct object. However, due to the low margin for error in this setting, AWDA and T-OSVI often collided with the object during fine manipulation, leading to reduced success rates of 76% and 16%, respectively. OSVI-WM executed these movements more accurately, achieving 100% success. Since there is no grasping involved in this task, we do not run T-OSVI*.

We present a qualitative example of OSVI-WM in the Human-Franka-PP real-world setup. In Fig. 6a, a human expert demonstrates the task of placing a can on a green target. Using only this demonstration, the agent (white Franka arm) successfully completes the task (Fig. 6b), despite changes in object locations and embodiment differences. This task was excluded from training, illustrating OSVI-WM's ability to generalize to unseen tasks by following the expert's intent rather than relying solely on visual semantics. This example showcases OSVI-WM's effectiveness under all key conditions outlined in our problem setup: unseen tasks, embodiment mismatch, unaligned trajectories, and in-domain training. More visualizations on different tasks are in the appendix.

What is the impact of the world-model-guided trajectory generation module? We assess the impact of this module through two ablations on the Meta-World dataset. The first removes the world model loss $\mathcal{L}_{wm}$; the second removes the world-model-guided trajectory generation module entirely, replacing it with a single action model that predicts an action vector from the expert demonstration and the agent's initial observation. This vector is passed through the pooling and waypoint prediction modules. As shown in Fig. 7, removing the world model yields the lowest performance, while adding it without $\mathcal{L}_{wm}$ provides limited gains. The loss $\mathcal{L}_{wm}$ is applied to future latent states predicted by the world model and is independent of ground-truth waypoint supervision. It acts as an auxiliary dynamics-consistency regularizer and encourages robust, task-agnostic representations, thus improving generalization to unseen tasks. Hence, combining the world model with $\mathcal{L}_{wm}$ achieves the best performance, underscoring the importance of both the module and its training signal.

(a) Example Ground Truth Trajectory

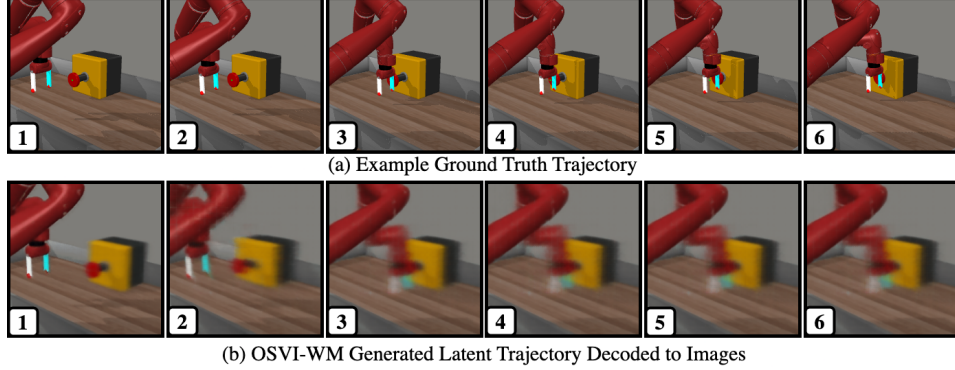(b) OSVI-WM Generated Latent Trajectory Decoded to Images

Figure 9: (a) An example ground truth rollout of the Button-Press-V2 task from Meta-World [61]. (b) Latent trajectory generated using OSVI-WM's world model decoded into images for visualization.

In Fig.9, we visualize the predicted trajectory generated by OSVI-WM's world model on the Button-Press-V2 task [61]. To convert the latent states into images, we train a Transpose Convolution-based image predictor. Images from the task are encoded using OSVI-WM's trained encoder and spatial pooling layer, and the image predictor is trained to predict back the pixels. During evaluation, we generate the future latent states $(r_2, \ldots, r_M)$ using the expert demonstration and initial agent frame, and decode them into images (Fig.9b) using the image predictor. A ground truth rollout is shown for comparison in Fig. 9a. Two key observations emerge: (a) the predicted trajectory captures meaningful, task-completing states, effectively summarizing the task; (b) the latent states are informative enough to reconstruct interpretable images. As expected, later states appear more pixelated due to accumulated errors in recursive latent state predictions. Despite this, the decoded trajectory clearly reflects the full task execution and produces accurate downstream performance.

Which components of OSVI-WM are most critical to its success? As shown above, the world-model-guided trajectory generation module, along with supervision from the WM loss $\mathcal{L}_{wm}$, is critical to OSVI-WM's success. Re-planning further boosts performance, particularly on the challenging Meta-World 'hard' tasks. Fig. 8 isolates the impact of re-planning, showing an approximate 30-point gain. For comparison, we also report the average success rate of AWDA [5] (with additional training data). Notably, even without re-planning, OSVI-WM outperforms AWDA. Table 3 presents further ablation studies evaluating key architectural choices. Removing stop-gradient in the WM loss significantly degrades performance, dropping Meta-World accuracy by 7% and Pick-and-Place by 80%, clearly indicating model collapse. Applying spatial pooling before the world-model-guided trajectory generation block introduces an information bottleneck, resulting in a 85% drop in Pick-and-Place and an 4% drop in Meta-World. Switching from multi-step recursive to single-step trajectory prediction has no effect on Pick-and-Place but lowers Meta-World performance by over 15% due to Meta-World's greater complexity. Together, these results

Table 3: Ablation results (avg. success rates) on the Pick-Place [10] and Meta-World [61], evaluating the impact of different architectural choices by individually disabling them.

Stop Grad in WM Loss	Spatial Pool after WM	Multi-Step Prediction	Benchmarks	
			Pick-Place	Meta-World
×	✓	✓	20	76.5
✓	×	✓	15	79.5
✓	✓	×	**100**	67.3
✓	✓	✓	**100**	**83.8**

show that OSVI-WM's effectiveness stems from key components: (a) the world-model-guided trajectory generation with WM loss, (b) re-planning, (c) stop-gradient in WM supervision, (d) spatial pooling after the world model, and (e) multi-step trajectory prediction. Regularization strategies like asymmetric demonstration image mixup [5] further improve generalization.

Division of train and test tasks: The choice of train–test task division affects performance, especially in the Meta-World simulation environment. During training, the model learns two key components: inferring tasks from the expert and acquiring the skills needed to complete these tasks. Hence, careful consideration is required when choosing training tasks to ensure meaningful generalization to unseen tasks. To further investigate this, we conducted two ablation studies on the Meta-World benchmark. In the first, we excluded all button-related tasks (Button-Press-Topdown, Button-Press-Topdown-Wall, Button-Press-Wall) from training and evaluated on Button-Press. The model still achieved a 97%

success rate, nearly identical to when the button-related tasks were included. This is because button pressing is relatively kinematically less complicated and the required skills were already acquired from other training tasks. In the second ablation, we removed Window-Close from the training set and evaluated on Window-Open. Here, performance decreased from 95% to 80%. The sliding window tasks here involve more complex physical dynamics. So the absence of related training data led to a noticeable drop in generalization. If the training set included a very large variety of tasks with diverse motions and skills, the model would potentially perform well on any task, even if no similar tasks were present in the training set. However, this approach would require large out-of-context datasets, significant training resources, and extended training times. In OSVI-WM, we follow the train–test split proposed in AWDA [5] for consistency with prior work.

5 Conclusion

We introduced OSVI-WM, a novel framework for one-shot visual imitation on unseen tasks. Given a single expert demonstration and the agent's initial observation, OSVI-WM encodes them into latent states and uses a world-model-guided approach to predict a trajectory of future latent states. This trajectory is decoded into physical waypoints that control the robot. Extensive experiments in both simulation and real-world settings demonstrate OSVI-WM's strong performance, and ablation studies confirm the importance of key design choices in the framework. By generalizing effectively from a single demonstration to unseen tasks, OSVI-WM takes a step toward enabling autonomous robotic systems capable of operating in complex, real-world domains such as industrial automation and medical assistance, without constant human supervision.

Limitations: While OSVI-WM demonstrates promising results, there are limitations that open avenues for future work. Like prior work [10, 5, 34], it only predicts 3D positions and a gripper state, without modeling the end-effector's orientation. This may limit its applicability to tasks requiring rotational actions, such as screwing or side-grasping. The current formulation focuses on single-task execution; extending the framework to sequential or multi-task settings is an interesting direction for future work. The division of train and test task can potentially influence the performance of OSVI-WM as detailed in Section 4. Further, although the test tasks are unseen during training, they belong to the same family of manipulators as the training data. While these limitations do not hinder current performance, they highlight promising directions for enhancing the framework in more complex task settings.

6 Acknowledgements

References

[1] Pieter Abbeel and Andrew Y Ng. Apprenticeship learning via inverse reinforcement learning. In *Proceedings of the International Conference on Machine Learning*, page 1, 2004.

[2] 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 *Proceedings of the Conference on Computer Vision and Pattern Recognition*, pages 15619–15629, 2023.

[3] Homanga Bharadhwaj, Abhinav Gupta, Shubham Tulsiani, and Vikash Kumar. Zero-shot robot manipulation from passive human videos. *arXiv preprint arXiv:2302.02011*, 2023.

[4] Vishal Bhutani, Anima Majumder, Madhu Vankadari, Samrat Dutta, Aaditya Asati, and Swagat Kumar. Attentive one-shot meta-imitation learning from visual demonstration. In *Proceedings of the International Conference on Robotics and Automation*, pages 8584–8590. IEEE, 2022.

[5] Matthew Chang and Saurabh Gupta. One-shot visual imitation via attributed waypoints and demonstration augmentation. In *Proceedings of the International Conference on Robotics and Automation*, pages 5055–5062. IEEE, 2023.

[6] Chang Chen, Yi-Fu Wu, Jaesik Yoon, and Sungjin Ahn. Transdreamer: Reinforcement learning with transformer world models. *arXiv preprint arXiv:2202.09481*, 2022.

[7] Zichen Jeff Cui, Yibin Wang, Nur Muhammad Mahi Shafiullah, and Lerrel Pinto. From play to policy: Conditional behavior generation from uncurated robot data. In *Proceedings of the International Conference on Learning Representations*, 2023.

[8] Zichen Jeff Cui, Hengkai Pan, Aadhithya Iyer, Siddhant Haldar, and Lerrel Pinto. Dynamo: In-domain dynamics pretraining for visuo-motor control. In A. Globerson, L. Mackey, D. Belgrave, A. Fan, U. Paquet, J. Tomczak, and C. Zhang, editors, *Proceedings of the Advances in Neural Information Processing Systems*, volume 37, pages 33933–33961. Curran Associates, Inc., 2024.

[9] Marco Cuturi and Mathieu Blondel. Soft-dtw: a differentiable loss function for time-series. In *Proceedings of the International Conference on Machine Learning*, pages 894–903. PMLR, 2017.

[10] Sudeep Dasari and Abhinav Gupta. Transformers for one-shot visual imitation. In Jens Kober, Fabio Ramos, and Claire Tomlin, editors, *Proceedings of the Conference on Robot Learning*, volume 155 of *Proceedings of Machine Learning Research*, pages 2071–2084. PMLR, 16–18 Nov 2021.

[11] Norman Di Palo and Edward Johns. Dinobot: Robot manipulation via retrieval and alignment with vision foundation models. In *2024 IEEE International Conference on Robotics and Automation (ICRA)*, pages 2798–2805. IEEE, 2024.

[12] Yan Duan, Marcin Andrychowicz, Bradly Stadie, OpenAI Jonathan Ho, Jonas Schneider, Ilya Sutskever, Pieter Abbeel, and Wojciech Zaremba. One-shot imitation learning. In I. Guyon, U. Von Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Proceedings of the Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017.

[13] Chrisantus Eze and Christopher Crick. Learning by watching: A review of video-based learning approaches for robot manipulation. *arXiv preprint arXiv:2402.07127*, 2024.

[14] Chelsea Finn and Sergey Levine. Deep visual foresight for planning robot motion. In *Proceedings of the International Conference on Robotics and Automation*, pages 2786–2793. IEEE, 2017.

[15] Chelsea Finn, Tianhe Yu, Tianhao Zhang, Pieter Abbeel, and Sergey Levine. One-shot visual imitation learning via meta-learning. In *Proceedings of the Conference on robot learning*, pages 357–368. PMLR, 2017.

[16] J Randall Flanagan and Alan M Wing. The role of internal models in motion planning and control: evidence from grip force adjustments during movements of hand-held loads. *Journal of Neuroscience*, 17(4):1519–1528, 1997.

[17] David Ha and Jürgen Schmidhuber. Recurrent world models facilitate policy evolution. In S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett, editors, *Proceedings of the Advances in Neural Information Processing Systems*, volume 31. Curran Associates, Inc., 2018.

[18] Danijar Hafner, Timothy Lillicrap, Ian Fischer, Ruben Villegas, David Ha, Honglak Lee, and James Davidson. Learning latent dynamics for planning from pixels. In *Proceedings of the International Conference on Machine Learning*, pages 2555–2565. PMLR, 2019.

[19] Danijar Hafner, Timothy Lillicrap, Jimmy Ba, and Mohammad Norouzi. Dream to control: Learning behaviors by latent imagination. In *Proceedings of the International Conference on Learning Representations*, 2020.

[20] Danijar Hafner, Timothy P Lillicrap, Mohammad Norouzi, and Jimmy Ba. Mastering atari with discrete world models. In *Proceedings of the International Conference on Learning Representations*, 2021.

[21] Danijar Hafner, Jurgis Pasukonis, Jimmy Ba, and Timothy Lillicrap. Mastering diverse domains through world models. *arXiv preprint arXiv:2301.04104*, 2023.

[22] Siddhant Haldar, Zhuoran Peng, and Lerrel Pinto. Baku: An efficient transformer for multi-task policy learning. In *Proceedings of the Annual Conference on Neural Information Processing Systems*, 2024.

[23] Nicklas Hansen, Hao Su, and Xiaolong Wang. Td-mpc2: Scalable, robust world models for continuous control. In *Proceedings of the International Conference on Learning Representations*, 2024.

[24] Nicklas A Hansen, Hao Su, and Xiaolong Wang. Temporal difference learning for model predictive control. In *Proceedings of the International Conference on Machine Learning*, pages 8387–8406. PMLR, 2022.

[25] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the Conference on Computer Vision and Pattern Recognition*, pages 770–778, 2016.

[26] Ahmed Hussein, Mohamed Medhat Gaber, Eyad Elyan, and Chrisina Jayne. Imitation learning: A survey of learning methods. *ACM Comput. Surv.*, 50(2), April 2017. ISSN 0360-0300.

[27] 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 *Proceedings of the Conference on Robot Learning*, pages 991–1002. PMLR, 2022.

[28] Łukasz Kaiser, Mohammad Babaeizadeh, Piotr Miłos, Błażej Osiński, Roy H Campbell, Konrad Czechowski, Dumitru Erhan, Chelsea Finn, Piotr Kozakowski, Sergey Levine, et al. Model based reinforcement learning for atari. In *Proceedings of the International Conference on Learning Representations*, 2020.

[29] Kushal Kedia, Prithwish Dan, Angela Chao, Maximus Adrian Pace, and Sanjiban Choudhury. One-shot imitation under mismatched execution. *arXiv preprint arXiv:2409.06615*, 2024.

[30] Alex X Lee, Anusha Nagabandi, Pieter Abbeel, and Sergey Levine. Stochastic latent actor-critic: Deep reinforcement learning with a latent variable model. *Advances in Neural Information Processing Systems*, 33:741–752, 2020.

[31] Seungjae Lee, Yibin Wang, Haritheja Etukuru, H Jin Kim, Nur Muhammad Mahi Shafiullah, and Lerrel Pinto. Behavior generation with latent actions. In *Proceedings of the International Conference on Machine Learning*, pages 26991–27008. PMLR, 2024.

[32] Haichao Liu, Sikai Guo, Pengfei Mai, Jiahang Cao, Haoang Li, and Jun Ma. Robodexvlm: Visual language model-enabled task planning and motion control for dexterous robot manipulation. *arXiv preprint arXiv:2503.01616*, 2025.

[33] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017.

[34] Zhao Mandi, Fangchen Liu, Kimin Lee, and Pieter Abbeel. Towards more generalizable one-shot visual imitation learning. In *Proceedings of the International Conference on Robotics and Automation*, pages 2434–2444. IEEE, 2022.

[35] Robert McCarthy, Daniel CH Tan, Dominik Schmidt, Fernando Acero, Nathan Herr, Yilun Du, Thomas G Thuruthel, and Zhibin Li. Towards generalist robot learning from internet video: A survey. *arXiv preprint arXiv:2404.19664*, 2024.

[36] Russell Mendonca, Oleh Rybkin, Kostas Daniilidis, Danijar Hafner, and Deepak Pathak. Discovering and achieving goals via world models. In M. Ranzato, A. Beygelzimer, Y. Dauphin, P.S. Liang, and J. Wortman Vaughan, editors, *Proceedings of the Advances in Neural Information Processing Systems*, volume 34, pages 24379–24391. Curran Associates, Inc., 2021.

[37] Russell Mendonca, Shikhar Bahl, and Deepak Pathak. Structured World Models from Human Videos. In *Proceedings of Robotics: Science and Systems*, Daegu, Republic of Korea, July 2023. doi: 10.15607/RSS.2023.XIX.012.

[38] R Chris Miall and Daniel M Wolpert. Forward models for physiological motor control. *Neural networks*, 9(8):1265–1279, 1996.

[39] Vincent Micheli, Eloi Alonso, and François Fleuret. Transformers are sample-efficient world models. In *Proceedings of the International Conference on Learning Representations*, 2023.

[40] Andrew Y Ng, Stuart Russell, et al. Algorithms for inverse reinforcement learning. In *Proceedings of the International Conference on Machine Learning*, volume 1, page 2, 2000.

[41] Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, et al. Dinov2: Learning robust visual features without supervision. *Transactions on Machine Learning Research Journal*, pages 1–31, 2024.

[42] Deepak Pathak, Parsa Mahmoudieh, Guanghao Luo, Pulkit Agrawal, Dian Chen, Yide Shentu, Evan Shelhamer, Jitendra Malik, Alexei A Efros, and Trevor Darrell. Zero-shot visual imitation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition workshops*, pages 2050–2053, 2018.

[43] Ilija Radosavovic, Tete Xiao, Stephen James, Pieter Abbeel, Jitendra Malik, and Trevor Darrell. Real-world robot learning with masked visual pre-training. In Karen Liu, Dana Kulic, and Jeff Ichnowski, editors, *Proceedings of the Conference on Robot Learning*, volume 205 of *Proceedings of Machine Learning Research*, pages 416–426. PMLR, 14–18 Dec 2023.

[44] Jan Robine, Marc Höftmann, Tobias Uelwer, and Stefan Harmeling. Transformer-based world models are happy with 100k interactions. In *Proceedings of the International Conference on Learning Representations*, 2023.

[45] Ryo Sakagami, Florian S Lay, Andreas Dömel, Martin J Schuster, Alin Albu-Schäffer, and Freek Stulp. Robotic world models—conceptualization, review, and engineering best practices. *Frontiers in Robotics and AI*, 10:1253049, 2023.

[46] Nur Muhammad Shafiullah, Zichen Cui, Ariuntuya (Arty) Altanzaya, and Lerrel Pinto. Behavior transformers: Cloning k modes with one stone. In S. Koyejo, S. Mohamed, A. Agarwal, D. Belgrave, K. Cho, and A. Oh, editors, *Proceedings of the Advances in Neural Information Processing Systems*, volume 35, pages 22955–22968. Curran Associates, Inc., 2022.

[47] Leslie N Smith and Nicholay Topin. Super-convergence: Very fast training of neural networks using large learning rates. In *Artificial intelligence and machine learning for multi-domain operations applications*, volume 11006, pages 369–386. SPIE, 2019.

[48] Richard S. Sutton and Andrew G. Barto. *Reinforcement Learning: An Introduction*. MIT Press, 2 edition, 2018.

[49] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

[50] Stefan Vogt and Roland Thomaschke. From visuo-motor interactions to imitation learning: behavioural and brain imaging studies. *Journal of Sports Sciences*, 25(5):497–517, 2007.

[51] Chen Wang, Linxi Fan, Jiankai Sun, Ruohan Zhang, Li Fei-Fei, Danfei Xu, Yuke Zhu, and Anima Anandkumar. Mimicplay: Long-horizon imitation learning by watching human play. In Jie Tan, Marc Toussaint, and Kourosh Darvish, editors, *Proceedings of the Conference on Robot Learning*, volume 229 of *Proceedings of Machine Learning Research*, pages 201–221. PMLR, 06–09 Nov 2023.

[52] Daniel M Wolpert, Zoubin Ghahramani, and Michael I Jordan. An internal model for sensorimotor integration. *Science*, 269(5232):1880–1882, 1995.

[53] Daniel M Wolpert, R Chris Miall, and Mitsuo Kawato. Internal models in the cerebellum. *Trends in cognitive sciences*, 2(9):338–347, 1998.

[54] Hongtao Wu, Ya Jing, Chilam Cheang, Guangzeng Chen, Jiafeng Xu, Xinghang Li, Minghuan Liu, Hang Li, and Tao Kong. Unleashing large-scale video generative pre-training for visual robot manipulation. In *Proceedings of the International Conference on Learning Representations*, 2024.

[55] Jialong Wu, Haoyu Ma, Chaoyi Deng, and Mingsheng Long. Pre-training contextualized world models with in-the-wild videos for reinforcement learning. In A. Oh, T. Naumann, A. Globerson, K. Saenko, M. Hardt, and S. Levine, editors, *Proceedings of the Advances in Neural Information Processing Systems*, volume 36, pages 39719–39743. Curran Associates, Inc., 2023.

[56] Philipp Wu, Alejandro Escontrela, Danijar Hafner, Pieter Abbeel, and Ken Goldberg. Daydreamer: World models for physical robot learning. In *Proceedings of the Conference on Robot Learning*, pages 2226–2240. PMLR, 2023.

[57] Chuyan Xiong, Chengyu Shen, Xiaoqi Li, Kaichen Zhou, Jiaming Liu, Ruiping Wang, and Hao Dong. Autonomous interactive correction mllm for robust robotic manipulation. In *8th Annual Conference on Robot Learning*, 2024.

[58] Mengda Xu, Zhenjia Xu, Cheng Chi, Manuela Veloso, and Shuran Song. Xskill: Cross embodiment skill discovery. In *Proceedings of the Conference on Robot Learning*, pages 3536–3555. PMLR, 2023.

[59] 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.

[60] Tianhe Yu, Chelsea Finn, Annie Xie, Sudeep Dasari, Tianhao Zhang, Pieter Abbeel, and Sergey Levine. One-shot imitation from observing humans via domain-adaptive meta-learning. *arXiv preprint arXiv:1802.01557*, 2018.

[61] 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 *Proceedings of the Conference on Robot Learning*, pages 1094–1100. PMLR, 2020.

[62] Maryam Zare, Parham M Kebria, Abbas Khosravi, and Saeid Nahavandi. A survey of imitation learning: Algorithms, recent developments, and challenges. *IEEE Transactions on Cybernetics*, 2024.

[63] Xinyu Zhang and Abdeslam Boularias. One-shot imitation learning with invariance matching for robotic manipulation. *arXiv preprint arXiv:2405.13178*, 2024.

[64] Gaoyue Zhou, Hengkai Pan, Yann LeCun, and Lerrel Pinto. Dino-wm: World models on pre-trained visual features enable zero-shot planning. *arXiv preprint arXiv:2411.04983*, 2024.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The abstract and introduction clearly state the paper's core contribution: a framework for one-shot visual imitation for unseen tasks via world-model-based trajectory synthesis. This has been described in detail in Sections 1, 2, and 3 and supported with experiments on benchmark environments demonstrating generalization to unseen tasks (Section 4), aligning well with the stated contributions.

   Guidelines:

   - The answer NA means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: The paper discusses the limitations in the second paragraph of Section 5.

   Guidelines:

   - The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

Answer: [NA]

Justification: The paper does not include theoretical results.

Guidelines:

- The answer NA means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

Answer: [Yes]

Justification: The paper provides sufficient detail in Sections 3 and 4 to reproduce the main experiments, with additional implementation specifics in the appendix. Code will be released upon acceptance, further supporting reproducibility.

Guidelines:

- The answer NA means that the paper does not include experiments.
- If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
- If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
- Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
  (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
  (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
  (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
  (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

    Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

    Answer: [Yes]

    Justification: The paper uses publicly available simulation benchmarks. The code and real-world experimental data will be made publicly available through the Github repository linked in the abstract.

    Guidelines:

    - The answer NA means that paper does not include experiments requiring code.
    - Please see the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
    - While we encourage the release of code and data, we understand that this might not be possible, so "No" is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
    - The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
    - The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
    - The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
    - At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
    - Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

    Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

    Answer: [Yes]

    Justification: The paper provides comprehensive training and evaluation details, including data splits, hyperparameters, and optimization settings in Sections 3.3 and 4, with additional specifics in the appendix.

    Guidelines:

    - The answer NA means that the paper does not include experiments.
    - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
    - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

    Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

    Answer: [Yes]

    Justification: We have provided extensive experiments showing the benefits of the proposed approach. Since we have empirically found the model's output waypoints to be highly repeatable for the same environment configuration, error bars are not explicitly provided. In Section 4, we report mean success rates over multiple rollouts with different environment configurations and initialization showing the performance improvements over prior baselines.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
- If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: The computer resources used are provided in Section 3.3 of the paper.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: The paper follows the NeurIPS Code of Ethics.

   Guidelines:

   - The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

Justification: The paper discusses, in the introduction and conclusion, how the proposed methodology for one-shot imitation learning can positively impact society by enabling autonomous robotic systems in areas such as medical assistance and industrial automation. Since the paper addresses foundational research, there are no known negative societal impacts.

Guidelines:

- The answer NA means that there is no societal impact of the work performed.
- If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

Answer: [NA]

Justification: The paper poses no such risks.

Guidelines:

- The answer NA means that the paper poses no such risks.
- Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
- Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
- We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

Answer: [Yes]

Justification: All assets used in the paper are properly cited, the license and terms of use explicitly mentioned and properly respected.

Guidelines:

- The answer NA means that the paper does not use existing assets.
- The authors should cite the original paper that produced the code package or dataset.
- The authors should state which version of the asset is used and, if possible, include a URL.
- The name of the license (e.g., CC-BY 4.0) should be included for each asset.
- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [NA]

    Justification: The paper does not release new assets.

    Guidelines:

    - The answer NA means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [NA]

    Justification: The paper does not involve crowdsourcing nor research with human subjects.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [NA]

    Justification: The paper does not involve crowdsourcing nor research with human subjects.

Guidelines:

- The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does not impact the core methodology, scientific rigorousness, or originality of the research, declaration is not required.

    Answer: [NA]

    Justification: The core method development in this research does not involve LLMs as any important, original, or non-standard components.

    Guidelines:

    - The answer NA means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
    - Please refer to our LLM policy (`https://neurips.cc/Conferences/2025/LLM`) for what should or should not be described.