World Models for Learning Dexterous Hand-Object Interactions from Human Videos

Raktim Gautam Goswami[1,2], Amir Bar[3,⋆], David Fan[1], Tsung-Yen Yang[1], Gaoyue Zhou[1,2], Prashanth Krishnamurthy[2], Michael Rabbat[1], Farshad Khorrami[2], and Yann LeCun[1,2]

[1] FAIR at Meta
[2] New York University
[3] Imperial College London

Abstract. Modeling dexterous hand–object interactions is challenging as it requires understanding how subtle finger motions influence the environment through contact with objects. While recent world models address interaction modeling, they typically rely on coarse action spaces that fail to capture fine-grained dexterity. We, therefore, introduce DexWM, a Dexterous Interaction World Model that predicts future latent states of the environment conditioned on past states and dexterous actions. To overcome the scarcity of finely annotated dexterous datasets, DexWM represents actions using finger keypoints extracted from egocentric videos, enabling training on over 900 hours of human and non-dexterous robot data. Further, to accurately model dexterity, we find that predicting visual features alone is insufficient; therefore, we incorporate an auxiliary hand consistency loss that enforces accurate hand configurations. DexWM outperforms prior world models conditioned on text, navigation, or full-body actions in future-state prediction and demonstrates strong zero-shot transfer to unseen skills on a Franka Panda arm with an Allegro gripper, surpassing Diffusion Policy by over 50% on average across grasping, placing, and reaching tasks. Codes and dataset are available on the project page: https://raktimgg.github.io/dexwm/.

Keywords: World Model · Dexterous Interactions · Embodied Vision

1 Introduction

As embodied agents become increasingly integrated into daily lives, dexterous manipulation emerges as a critical capability for achieving human-like interaction with the physical world. Everyday tasks like cooking, as well as high-stakes applications like surgery, demand a level of dexterity that is infeasible with commonly used parallel-jaw grippers. Modeled after the human hand, dexterous grippers enable complex tool use, fine-grained movements, and in-hand manipulation [36, 45, 58, 64]. Recent advances in deep learning have produced vision-based manipulation policies [22, 23, 35, 52, 54], yet these methods often

⋆ Work done at Meta.

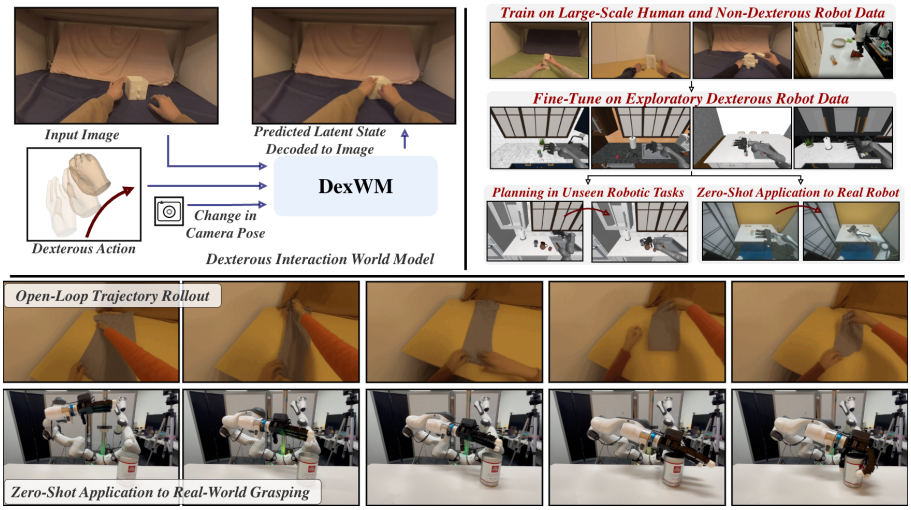

Fig. 1: We introduce **DexWM**, a Dexterous Interaction World Model which predicts future latent states of the environment from past states and dexterous actions. Trained on large-scale human and non-dexterous robot videos, DexWM learns to simulate complex trajectories. Further, with minimal fine-tuning on a small exploratory robot simulation dataset, DexWM enables planning for novel reaching, grasping, and placing tasks in simulation, and transfers zero-shot to real-world robot grasping.

struggle to generalize to unseen tasks and to plan robustly in physical environments [15, 30, 92]. Successful execution requires models that reason about how actions affect the environment, such as recognizing that opening the hands when holding an object will cause it to drop.

World models can learn environmental dynamics from observation and action [53], and thus offer a promising solution. Early work on learned world models [45, 69, 93] has primarily focused on small-scale tasks with constrained environments and limited action spaces. More recent efforts handle complex actions, such as text [2], navigation [10], and whole-body motion [8]. However, the action representations in these methods are typically too coarse to capture the fine-grained structure required for modeling dexterous dynamics. Similar to prior work in world models [6, 8, 10, 93], we define "dynamics" as the modeling of how the agent moves in response to actions, how the environment reacts, and how both appear from the camera's viewpoint. An important application of such dexterous world models is robotic manipulation. However, learning them directly from robot data is challenging due to the lack of large-scale datasets with dexterous grippers.

To address these challenges, we propose DexWM (Fig. 1), the first world model that learns dexterous interaction dynamics from human videos. DexWM predict future latent states based on past states and dexterous hand actions. Inspired by recent work [28, 75] that leverages human training data, we pre-train DexWM on EgoDex [44], a large-scale egocentric human interaction dataset, and further incorporate DROID [49] sequences, consisting of non-dexterous robot

manipulation, to learn cross-embodiment dynamics. DexWM's actions are represented as differences in 3D hand keypoints and camera poses, capturing detailed hand configurations and enabling the model to learn how hand posture changes affect the environment.

We find that accurately simulating hand locations using the next latent state prediction objective alone is difficult. Therefore, we train DexWM to jointly optimize the future environment state and the hand configuration, providing a richer learning signal for dexterity. DexWM outperforms prior world models [2, 8, 10] in open-loop trajectory simulation, better captures fine-grained hand dynamics, and demonstrates reliable controllability under arbitrary action sequences.

Furthermore, DexWM enables strong zero-shot transfer to robot manipulation tasks of grasping, placing, and reaching by optimizing actions at test time within a Model Predictive Control (MPC) framework. Notably, when deployed on a real-world Franka Panda robot with the Allegro grippers, DexWM achieves an 83% success rate in object grasping. Consistent with the robot learning literature [5, 6, 15, 30], we consider zero-shot as performing new skills like "Grasp", "Reach", or "Place" despite not seeing them performed on the same embodiment during training. To bridge the gap between training data and robot embodiment, we fine-tune the model on about four hours of exploratory, non-task-specific dexterous data from the RoboCasa simulation suite [59]. Unlike existing behavior cloning methods [22, 28, 75] that predict actions or waypoints from observation, DexWM is used as a state-transition model within an MPC optimization framework for planning waypoint trajectories, which are executed via low-level controllers, thus offering greater robustness.

In summary, we introduce DexWM, a latent-space world model for learning dexterous hand–object interaction dynamics directly from human videos. To support fine-grained dexterity, we incorporate an auxiliary hand-consistency loss. Furthermore, DexWM demonstrates that world models trained on human data can scale effectively and transfer to zero-shot robotic manipulation tasks such as reaching, grasping, and placing.

2 Related Work

Recent environment modeling approaches are largely built on Diffusion and Flow Matching objectives, which learn to generate videos by denoising or integrating learned velocity fields over time [24, 41, 56, 63, 76, 77]. Combined with large-scale text–video training, these models have improved to the point where they can simulate highly realistic videos from textual prompts [12, 13, 42, 86, 91]. For interactive and streaming applications [18, 20, 67, 79, 90], generating video frames autoregressively is particularly important, but introduces error accumulation over time [27, 29, 55, 81, 83]. Diffusion Forcing [16] addresses this issue by combining next-token prediction with full-sequence diffusion and injecting noise into previously generated context frames, while other autoregressive video diffusion approaches mitigate drift through multistep prediction and refinement [40,51,82].

In this work, we focus primarily on building such action-conditioned predictive models for dexterous hand-object interactions.

Action-conditioned predictive models, often referred to as "World Models", have recently been used to simulate computer game engines [14, 46, 88]. For example, DIAMOND [4] aims to simulate Counter-Strike, while Dreamer [34] has been applied to MineCraft. Other approaches focus on more visually realistic settings with continuous action spaces, including navigation [10, 45, 57] and full-body control [8]. In contrast, modeling dexterous dynamics requires precise, fine-grained control over hand–object interactions, and remains more challenging.

World models have also become increasingly influential in robotics. While commonly used to train reinforcement learning agents [17, 32, 33, 38], they have also proven effective for model-based policy optimization [6, 93]. For dexterous dynamics modeling, prior works in world models have explored cross-embodiment dynamics via particle-based representations [39, 43], goal-conditioned manipulation through articulated object modeling (DexSim2Real^2 [47]), and scene-action-conditioned video diffusion for simulating dexterous behavior (DWM [50]).

In addition to modeling dexterous dynamics, we show that DexWM can be used in a Model Predictive Control (MPC) framework for zero-shot dexterous manipulation tasks such as grasping. Dexterous manipulation, in general, remains a longstanding challenge in robotics. Early approaches relied on hand-crafted gaiting methods for controlling grippers [26, 36, 60], but the high degrees of freedom and complex contact dynamics of such grippers has led to a shift toward learning-based methods [5, 87]. Sim2real techniques, in particular, have enabled robots to perform complex tasks such as solving a Rubik's cube [3], in-hand manipulation [19, 37, 65], and functional grasping [1]. Much of this progress is driven by improved access to training data. Although datasets with dexterous manipulation remain scarce, the availability of large-scale egocentric human video datasets [9, 31, 44, 72] and efficient hand annotation tools [62, 70, 89] has enabled recent research [7, 11, 21, 48, 66, 68, 73, 84, 85] to leverage human demonstrations for learning dexterous manipulation. HOP [75], for example, retargets human videos to robot simulation for sensorimotor learning, while MAPLE [28] pre-trains encoders with dexterous priors. Similarly, we train DexWM on egocentric human videos [44] and sequences of non-dexterous robot data [49] to learn dexterous manipulation dynamics.

3 Proposed Methodology

Our goal is to build a world model that predicts how dexterous hand–object interactions unfold: how the hand moves, how objects respond, and how both appear from an egocentric viewpoint. A schematic illustration of the model is shown in Fig. 2. In what follows, we first describe the problem formulation, then present the State and Action representations (Sec 3.1), the Predictor (Sec 3.2), and the Loss (Sec 3.3). Finally, we explain how to use a trained DexWM for planning manipulation in Sec 3.4.

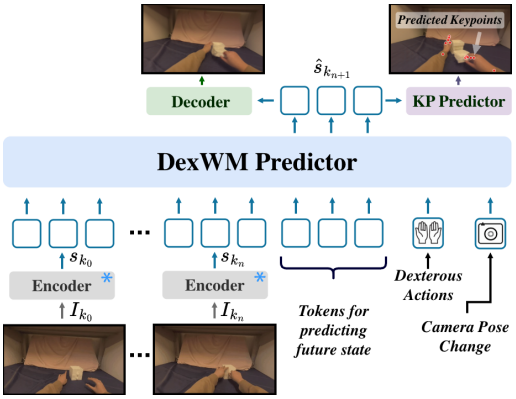

Fig. 2: DexWM Architecture. Images are encoded into latent states using a frozen DINOv2 [61] encoder. The DexWM predictor takes these states, hand actions, and camera motions to predict the next state, which can then be decoded into reconstructed images and hand keypoints.

Problem Formulation. Let $s_{k_i} \in \mathcal{S}$ denote the environment's (latent) state at timestep k_i, for index i, which encodes the configuration of the dexterous hand and the geometry of the surrounding scene. The agent issues a dexterous action $a_{k_1 \to k_2} \in \mathcal{A}$, specifying the hand control executed between timesteps k_1 and k_2.

Since the latent state s_{k_i} is not directly observed, the agent receives an egocentric RGB image $I_{k_i} \in \mathbb{R}^{H \times W \times 3}$, captured by a human or robot equipped with a dexterous hand. We introduce an encoder $E_\phi : \mathbb{R}^{H \times W \times 3} \to \mathcal{S}$, which maps the observation I_{k_i} to a latent state representation $s_{k_i} = E_\phi(I_{k_i})$.

To approximate the environment dynamics, we define the predictor $f_\theta : \mathcal{S} \times \mathcal{A} \to \mathcal{S}$, as a mapping from a current latent state and an action to a predicted future state.[4] Formally, our objective is to learn

$$\hat{s}_{k_2} = f_\theta(s_{k_1}, a_{k_1 \to k_2}) = f_\theta(E_\phi(I_{k_1}), a_{k_1 \to k_2}), \tag{1}$$

where $\hat{s}_{k_2}$ is the predicted environment state after executing $a_{k_1 \to k_2}$.

3.1 State and Action Representation

Next, we describe the states and actions used in DexWM.

Latent States. Pixel-level details are often unnecessary for modeling the underlying system dynamics. For example, in object manipulation, the agent is typically more concerned with the object's shape and structure than its color. To address this, we employ the DINOv2 [61] image encoder (E_ϕ) to transform pixel data into a latent embedding space following [6,30,93]. Specifically, we utilize patch-level features as the latent state such that $s_{k_i} \in \mathbb{R}^{\mathcal{P} \times d}$, where $\mathcal{P}$ is the number of image patches and d is the feature dimension per patch. DINOv2 features are semantically rich and have demonstrated strong generalization across diverse environments and scenarios [61,93].

[4] f_θ can take multiple temporal states as input; Eq. (1) shows a single for simplicity.

Fig. 3: (a) Action Representation. Hand actions are represented as differences in 3D keypoints between frames (e.g., $H_{k_j} - H_{k_i}$), providing a unified representation of dexterous actions in DexWM. This is supplemented with camera motion to capture the agent's movement. **(b)** For the DROID [49] dataset, Parallel-jaw grippers are approximated as dexterous hands using dummy keypoints placed on concentric circles at the end-effector, with radii varying by gripper's open/close state, mimicking finger spread.

Action Representation. A key challenge is how to represent actions in a dexterous dynamics model. Prior work has modeled wrist position [8] or used high-level text [2], but these representations are often too coarse for dexterous skills. Instead, we seek an action representation that precisely captures the change of the agent's hands, as well as how they move while performing a task.

We start by defining the action vector between states s_{k_1} and s_{k_2} as the difference in keypoint positions between timesteps k_1 and k_2. Following the MANO [70] parameterization, each hand is represented by 21 keypoints (Fig. 3a), with coordinates $\mathcal{H}^L_{k_i}, \mathcal{H}^R_{k_i} \in \mathbb{R}^{21\times 3}$ for the left and right hands, respectively, at state s_{k_i}. Specifically, $\mathcal{H}_{k_i} = \{\mathcal{H}^L_{k_i}, \mathcal{H}^R_{k_i}\}$ encodes the hand configurations.

However, simply using hands information to represent actions does not account for how the agent moves. To address this, we apply two modifications. First, instead of representing $\mathcal{H}_{k_2}$ in the camera frame at k_2, we use the known rigid transformation $T^{k_1}_{k_2}$ to express all keypoints in the same coordinate frame k_1. Second, to inform the world model about camera pose changes, we append the change in camera translation $\delta t_{k_1\to k_2} \in \mathbb{R}^3$ and orientation $\delta q_{k_1\to k_2} \in \mathbb{R}^3$ (as Euler angles) to the action vector. The final action vector is defined as

$$a_{k_1\to k_2} = \left[(\mathcal{H}_{k_2} - \mathcal{H}_{k_1})^T,\ \delta t^T_{k_1\to k_2},\ \delta q^T_{k_1\to k_2}\right]^T \in \mathbb{R}^{44\times 3}. \tag{2}$$

Most egocentric human video datasets [9, 44, 80] already include hand keypoint annotations. For the DROID [49] dataset, parallel-jaw grippers are approximated as dexterous hands using dummy keypoints at the end-effector (Fig. 3b). For real robot and RoboCasa [59] simulation data, keypoints are computed via forward kinematics from known joint angles. To map the robot's 4-finger Allegro hand to DexWM's 5-finger action space, the last Allegro finger (equivalent to the human ring finger) keypoints are also used to represent the pinky finger. This mapping incurs a minimal performance trade-off, as detailed in the supplementary material.

3.2 Predictor

With the states and actions of the world model defined, we now describe the predictor model f_θ used in DexWM. The predictor model takes as input a history

of observed environment states $s_{k_0}, \ldots, s_{k_n}$ and current action $a_{k_n \to k_{n+1}}$, and predicts the next state $\hat{s}_{k_{n+1}}$. Formally,

$$\hat{s}_{k_{n+1}} = f_\theta(s_{k_0}, \ldots, s_{k_n}, a_{k_n \to k_{n+1}}) \tag{3}$$

where θ denotes the trainable parameters of the network.

Unlike previous works [8,10], we assume that the environment is deterministic, which leads to faster inference time. Also, while the timesteps $k_0, \ldots, k_n$ can be uniformly spaced, we find that training in a non-fixed frequency by randomly skipping states improves generalization.

Our predictor architecture is based on Conditional Diffusion Transformers (CDiT) [10]. CDiT provides strong action conditioning via AdaLN [63] layers, where the flattened action vector of $44 \times 3 = 132$ dimensions is projected as conditioning input to each transformer block. Unlike diffusion-based CDiT methods [8, 10], we directly regress future latent states using DINOv2 features for faster inference without iterative denoising. To use CDiT with DINOv2, we define tokens for predicting the future state $\hat{s}_{k_{n+1}}$ (Fig. 2), initialized from s_{k_n}.

Multistep Prediction. For inference and planning in robotic tasks, we perform multistep prediction by autoregressively feeding the predicted state $\hat{s}_{k_{n+1}}$ and the subsequent action $a_{k_{n+1} \to k_{n+2}}$ back into f_θ to generate $\hat{s}_{k_{n+2}}$, continuing this process for future timesteps.

3.3 Hand Consistency Training Loss

As the primary goal of DexWM is to predict future latent states of the environment, we use a mean squared error loss on the predicted embeddings:

$$\mathcal{L}_{\text{state}} = \frac{1}{\mathcal{P} \times d} \sum_{p=1}^{\mathcal{P}} \| s_{k_{n+1}}(p) - \hat{s}_{k_{n+1}}(p) \|_2^2 \tag{4}$$

where p indexes the image patches and $s_{k_{n+1}}$ is the DINOv2 embedding of the ground truth image.

However, we observe that relying solely on $\mathcal{L}_{\text{state}}$ is insufficient for capturing the fine-grained details necessary for modeling dexterous dynamics, as the hands occupy only a small region of the image. To address this, we incorporate an additional loss term that encourages the model to capture local information. Specifically, we use a transformer-based network g_θ to predict heatmaps of the fingertip and wrist locations in $\mathcal{H}_{k_{n+1}}$, denoted as $\hat{V}_{k_{n+1}} \in \mathbb{R}^{12 \times H \times W}$. This ensures that the predicted state $\hat{s}_{k_{n+1}}$ is informative enough to recover keypoint positions. This hand consistency (HC) loss is defined as:

$$\mathcal{L}_{\text{HC}} = \frac{1}{12 \times H \times W} \| V_{k_{n+1}} - \hat{V}_{k_{n+1}} \|_2^2 \tag{5}$$

where $V_{k_{n+1}}$ are the ground truth heatmaps. During training, the image encoder is kept frozen, while the rest of the model is optimized using $\mathcal{L} = \mathcal{L}_{\text{state}} + \lambda \mathcal{L}_{\text{HC}}$ with $\lambda = 100$ yielding the best results in our experiments.

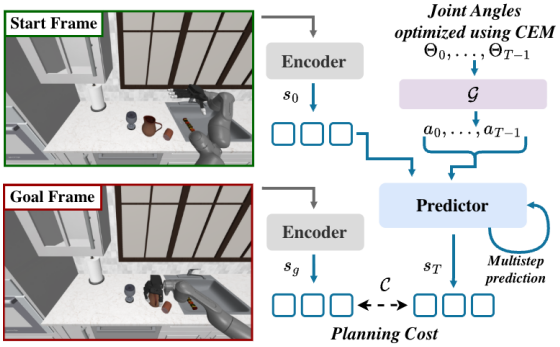

Fig. 4: Goal-Conditioned Planning with DexWM. The joint angles $\Theta_0, \ldots, \Theta_{T-1}$ are optimized using CEM to get actions $a_0, \ldots, a_{T-1}$ to drive the system to the goal.

3.4 Robot Task Planning

In addition to modeling dexterous interaction dynamics, DexWM enables zero-shot transfer to robotic manipulation tasks. Specifically, DexWM is used as a state transition model to plan waypoint trajectories for downstream control.

Planning Optimization. We use a goal-conditioned planning setup with the learned model f_θ. Given initial (s_0) and goal (s_g) states, we solve:

$$\Theta_0^*, \ldots, \Theta_{T-1}^* = \arg\min_{\Theta_0, \ldots, \Theta_{T-1}} \mathcal{C}(s_T, s_g) \tag{6}$$
$$\text{s.t} \quad a_k = \mathcal{G}(\Theta_k), \quad \hat{s}_{k+1} = f_\theta(s_k, \ldots, s_0, a_k), \quad k = 0, \ldots, T-1$$

where Θ_k represent the joint angles of the robot, $\mathcal{G}$ uses the forward kinematics of the robot to calculate a_k, $\mathcal{C}$ is the planning cost, and T is the planning horizon (Fig. 4). In this formulation, we assume uniformly sampled timesteps with states $\{s_0, s_1, \ldots\}$ and actions $\{a_0, a_1, \ldots\}$. We use the Cross-Entropy Method (CEM) [71] to optimize the joint angles. Implementation details of CEM, including the number of samples, planning horizon, iteration count, and computational cost, are provided in the supplementary material.

Planning Cost. We use the planning cost $\mathcal{C} = \mathcal{C}_{\text{state}} + \mu\mathcal{C}_{\text{kp}}$ ($\mu = 0.001$), with $\mathcal{C}_{\text{state}}$ being the L_2 distance between latent states s_T and s_g, and $\mathcal{C}_{\text{kp}}$ the Euclidean distance between keypoint pixels locations corresponding to heatmaps $\hat{V}_T$ and $\hat{V}_g$ predicted from s_T and s_g, respectively. Combining both cost terms improves planning performance over using $\mathcal{C}_{\text{state}}$ alone, indicating latent embeddings alone may be suboptimal for planning. For the grasping task, we also add a cost on end-effector orientation to maintain neutral poses for successful grasps.

4 Experiments and Results

In this section, we first analyze the contribution of DexWM 's individual components. We then evaluate its performance in open-loop dexterous rollouts and

Table 1: DexWM Benefits From Human Video Data. Training DexWM on EgoDex in addition to DROID contributes to downstream open-loop performance in RoboCasa, as measured by lower embedding loss and higher PCK@20.

Dataset	EgoDex				RoboCasa			
	Embedding L2 Error↓		PCK@20↑		Embedding L2 Error↓		PCK@20↑	
	At 4s	Avg	At 4s	Avg	At 4s	Avg	At 4s	Avg
EgoDex	0.67	0.51	**60**	68	1.03	0.79	3	13
DROID	1.39	1.06	1	2	1.3	0.96	2	12
EgoDex+DROID	**0.66**	**0.5**	**60**	**69**	**0.79**	**0.57**	**7**	**17**

show its ability to capture fine-grained hand dynamics and maintain reliable control under arbitrary atomic actions. Finally, we assess DexWM in zero-shot trajectory planning across both simulated and real-world robotic settings.

4.1 Datasets

For training and evaluation, we use EgoDex [44], an egocentric human video dataset containing 829 hours of 1080p footage with rich hand and pose annotations, and DROID [49], a diverse dataset of robot manipulation with parallel-jaw grippers. We also use about 4 hours of exploratory sequences of random arm motions collected in the RoboCasa [59] simulation framework using a Franka Panda arm with Allegro gripper for fine-tuning the model for robotic tasks. For more details, see the supplementary material.

4.2 Ablation Studies

Human Video. We evaluate the impact of training on human videos to downstream performance (see Table 1). We assess zero-shot open-loop rollout performance on *Lift*, a dataset collected in RoboCasa (see supplementary), and on EgoDex. We report embedding L_2 loss and PCK@20 over predicted keypoints $\hat{V}$ (Sec. 3.3) at 4 seconds and on average. As shown in Table 1, adding EgoDex on top of DROID significantly boosts performance on RoboCasa, while preserving performance on EgoDex. This indicates that adding human video contributes to downstream performance across different embodiments.

Model Size. We vary the model's predictor size from DexWM-S to DexWM-XL (30M to 450M parameters) while keeping the training setup, data, and vision encoder consistent. See the supplementary material for model architecture details. Fig. 5 shows that embedding prediction error and keypoint overlap percentage consistently improve with larger model size. This suggests that higher model capacity helps facilitate better dynamics learning. We use DexWM-XL by default, unless otherwise noted. For completeness, we include the NWM* [10] and PEVA* [8] baselines in Fig. 5, and discuss these comparisons in Sec. 4.4.

Backbone. As the pretrained vision encoder defines the latent space of our world model, we evaluate whether DexWM generalizes across different backbones, an

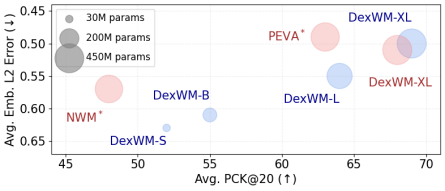

Fig. 5: Scaling With Predictor Size. Larger DexWM models yield better PCK@20 and lower embedding L2 error. Blue: EgoDex+DROID; Red: EgoDex-only training.

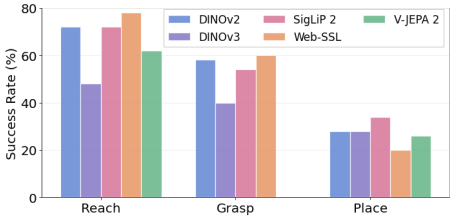

Fig. 6: Encoder Ablation. Downstream robotic task success rates on RoboCasa (simulation).

aspect not fully explored in prior work [8, 10, 93]. We test state-of-the-art self-supervised image encoders (DINOv2 [61], DINOv3 [74], Web-SSL [25]), video models (V-JEPA 2 [6]), and language-supervised models (SigLIP 2 [78]). Keeping everything else fixed, we evaluate success rates in simulation, since latent spaces from different encoders are not directly comparable due to differing scales. In addition, because encoders like V-JEPA 2 and SigLIP 2 use different embedding dimensions than that of DexWM's predictor, we add learnable input/output projections to align with the predictor. DexWM performs well across backbones and is not restricted to DINOv2 (Fig. 6), though performance varies by task, with DINOv2 strongest overall. While this demonstrates DexWM's modularity with respect to the backbone, additional tuning (e.g., embedding normalization) can be important for some backbones to further improve performance.

Hand Consistency Loss. As detailed in Sec. 3.3, we use the hand consistency loss as an auxiliary objective to improve fine-grained dexterity. Table 3 shows that adding hand consistency loss yields up to a 34% increase in PCK@20 at 4 seconds prediction. Beyond open-loop rollout gains, predicted keypoints also benefit robot planning (Sec. 3.4).

Camera-Pose in Action. To account for agent movement, we integrate camera pose changes directly into the action representation (Sec. 3). As shown in Table 2, removing the camera pose (hand-only) or treating it as part of the state degrades performance. This underscores the importance of including camera pose within the action space.

Table 2: Camera-pose in action ablation. Avg. values on EgoDex are reported.

Method	L2↓	PCK@20↑
Hand-Only	0.60	56
Camera in State	0.57	59
DexWM	**0.50**	**69**

4.3 Baselines

We compare DexWM against several strong baselines spanning video prediction, world modeling, and action generation. For full details, see the supplementary material. Cosmos-Predict2 [2] is a diffusion-based "Video-to-World" model that synthesizes future frames from text prompts and an optional starting image.

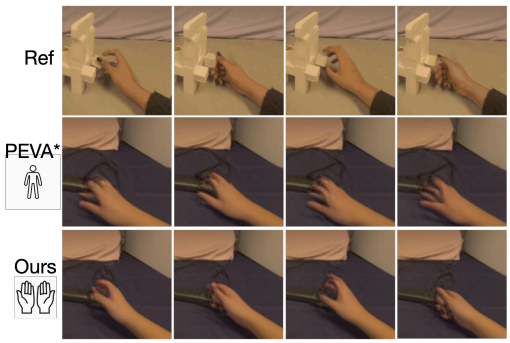

Fig. 7: Action Transfer. Transferring actions from a reference sequence to a new environment using DexWM and PEVA*. DexWM better captures fine-grained hand states that match those in the reference sequence.

Navigation World Model (NWM) [10] predicts future egocentric observations conditioned on navigation actions; for fairness, we adopt a simplified variant, NWM*, that conditions only on camera motion and excludes hand and body dynamics. PEVA [8] generates egocentric videos from 3D human pose trajectories; we use a modified version, PEVA*, that conditions solely on upper-body poses without finger articulation. Finally, Diffusion Policy [22] provides a state-of-the-art generative action policy that predicts multistep actions from the current observation and a goal image.

4.4 Open-Loop Trajectory Evaluation

Experiment. We aim to test how DexWM performs on open-loop rollouts compared to challenging baselines like NWM* and PEVA*. Specifically, given the initial state and an action sequence, each model predicts future latent states for 4 seconds (20 frames at 5 Hz), enabling evaluation of the world models' long-horizon prediction performance.

Training. We train all models on EgoDex for 40 epochs. All models use the DINOv2 encoder, allowing us to measure perceptual similarity via the same L_2 embedding prediction error against ground truth.

Evaluation. The predicted embeddings are passed through a shared keypoint prediction layer, and keypoint overlap within a 20-pixel radius (PCK@20) is computed. Moreover, we measure the L_2 error, which captures overall perceptual similarity, while PCK@20 reflects local hand keypoint accuracy, important for capturing dexterous interactions.

Results. DexWM outperforms other models in open-loop trajectory evaluation (Table 4). NWM* performs poorly due its sole reliance on navigation actions. PEVA* slightly outperforms DexWM in L_2 error, however, we find that lower perceptual similarity does not always imply accurate hand position, which is important for dexterous interactions. DexWM achieves over 5 points higher PCK@20 on average, indicating superior preservation of local hand information.

Table 3: HC Loss improves fine-grained dexterity. Reporting embedding L2 error and Percentage of Correct Keypoints @20 (PCK@20) on EgoDex [44].

	Emb. L2 Error↓		PCK@20↑	
HC Loss	At 4s	Avg	At 4s	Avg
×	0.85	0.61	26	52
✓	**0.66**	**0.50**	**60**	**69**

Table 4: Comparing World Models with Different Action Spaces. PCK typically reflects hand accuracy better than embedding L2 error. All models trained on EgoDex.

Model	Emb. L2 Error↓		PCK@20↑	
	At 4s	Avg	At 4s	Avg
NWM* [10]	0.74	0.57	34	48
PEVA* [8]	**0.62**	**0.49**	56	63
DexWM	0.67	0.51	**60**	**68**

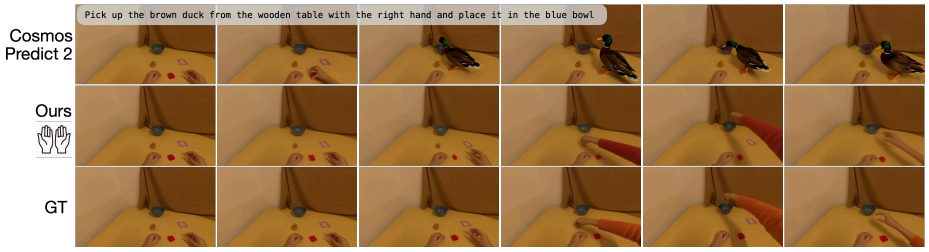

Fig. 8: Conditioning on Hand Motion Enables Precise Control. DexWM utilizes dense dexterous actions, enabling finer-grained physically-grounded prediction compared to text-conditioned World Models like Cosmos-Predict 2 [2].

Action Transfer. To highlight the difference from PEVA, we conduct a qualitative action transfer experiment (see Fig. 7). Actions from a reference trajectory are used to rollout states in a different environment. PEVA*, is conditioned on body poses and produces inaccurate hand configurations, while DexWM predicts fine-grained hand states that closely match those in the reference sequence.

Comparison with Text Conditioned Models. While text-conditioned models like Cosmos Predict 2 generate visually compelling scenes, their predictions are not always physically grounded in hand–object interactions; for example, in Fig. 8, the model introduces a new duck simply because the text input mentions "the brown duck," even though it refers to the dummy duck on the table.

Controllability. To assess the ability to follow structured physical control, we show rollouts with simple atomic actions (e.g., moving the hand up) in Fig. 9. In each sequence, the right hand moves 1 cm per frame in the specified direction. DexWM accurately follows these unseen atomic actions, and when the hand collides with the cup in the third row, the cup moves forward, indicating that DexWM also learns such physical interactions.

4.5 Human Video To Robot Transfer

Experiment. In addition to accurately simulating the world, DexWM transfers to unseen robotic manipulation tasks in simulation and real-world when deployed

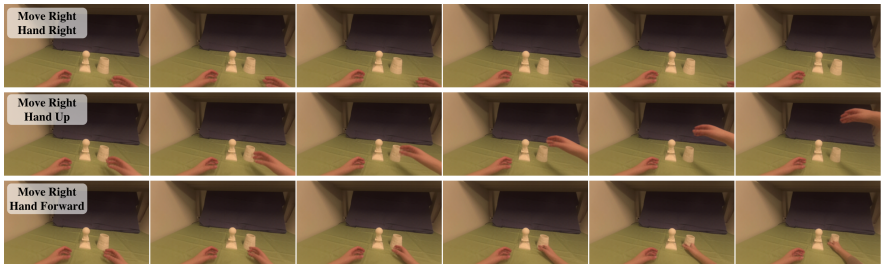

Fig. 9: Simulating Counterfactual Actions. Starting from the same initial state, DexWM predicts future states given different atomic actions. The model reliably follows each action sequence, while capturing environment dynamics (e.g, see third row final frame where the cup moves forward when the hand collides with the cup).

on a Franka arm with Allegro grippers. Specifically, given only small amounts of exploratory robot simulation data, our goal is to evaluate whether DexWM can use the human data pretraining to perform tasks such as *reach*, *place*, and *grasp*, which were previously unseen in the target environment and embodiment.

Exploratory Dataset. We compare two teleoperation-free strategies for collecting exploratory data in RoboCasa [59], both using randomized environments and object placements. *Lift-Initialized Random Exploration* is the first strategy that executes ground-truth trajectories from the *Lift* dataset (see supplementary for details) with added noise. Environment and object randomization, and added noise, prevent successful grasps, promoting broad, unbiased coverage of the environment to bridge the embodiment gap rather than learn task-specific skills. In the second strategy, we remove reliance on *Lift* by sampling random 3D targets in randomized environments and controlling the robot to reach them. This fully programmatic data collection procedure requires no teleoperation or initialization from other datasets. Lift-initialized Random Exploration approach achieves slightly better performance (53% vs. 49% average success over reach, grasp, and place simulation tasks). Therefore, in what follows, we present results corresponding to models trained on this dataset.

Training. We pretrain DexWM on EgoDex [44] and DROID [49], then fine-tune on exploratory trajectories created in RoboCasa [59]. We compare the fine-tuned DexWM model to training a goal-conditioned Diffusion Policy (DP) [22] and a DexWM model on RoboCasa from scratch, without human-data pretraining. For DP, training goals are uniformly sampled from future frames of each trajectory.

Evaluation. We evaluate the resulting model's planning or policy rollout performance in the real world and in simulation, without using any real-world fine-tuning data. In simulation, we conduct 50 trials each for *reach*, *grasp*, and *place* tasks, measuring success by the Euclidean distance between target and actual poses (for *reach* and *place*) and additionally by object-robot contact (for *grasp*). Real-world evaluation consists of 12 grasping trials with varied objects, with success determined by manually observing whether the object is in the hand. See the supplementary for details.

Table 5: Robot Transfer Results. DexWM outperforms Diffusion Policy and DexWM (w/o pre-training) in sim. and real-world, measured by success rate (%).

	RoboCasa Simulation			Real Robot
Model	Reach	Place	Grasp	Grasp
Diffusion Policy [22]	16	8	0	0
DexWM (w/o PT)	18	8	14	0
DexWM (Ours)	**72**	**28**	**58**	**83**

Fig. 10: Real Robot Example. Given goal and start images, DexWM successfully plans and executes the trajectory by finding the optimal actions using the Cross-Entropy Method. Notably, DexWM works zero-shot without any real-robot training.

Simulation Transfer Results. Planning results in Table 5 show that DexWM consistently outperforms all baselines. In the *place* task, where the robot must maintain its grasp and transport the object without additional feedback, such as from force sensors, DexWM achieves 28% success despite the added difficulty. The large gap over the DexWM variant without human video pre-training underscores the importance of human demonstrations. Further, goal-conditioned Diffusion Policy (DP) [22] underperforms due to the exploratory nature of the training dataset, which lacks task annotations and successful completions. To verify that DexWM 's gains over DP in Table 5 are not due to pretraining alone, we conduct an experiment where we pre-train DP on the same human dataset (with a similar action space) and fine-tune it on robot data. DP achieves only 4% average simulation success, far below DexWM's 53%, indicating that the improvement stems from modeling dexterous dynamics rather than pretraining alone. An analysis of DexWM 's failure cases is in the supplementary material.

Zero-Shot Real Robot Grasping Results. We evaluate the zero-shot grasping performance of DexWM when deployed on a real-world Franka arm with Allegro gripper, similar to that in the simulation (see Table 5). Without any fine-tuning on real robot data, DexWM achieves 10 successes out of 12 trials ($\approx 83\%$ success rate), highlighting the planning benefits with a world model (planned trajectory example in Fig. 10). By planning in latent space rather than directly predicting actions, DexWM shows strong generalization. In contrast, Diffusion Policy trained on exploratory simulation data fails on the real task, highlighting the robustness of world models in handling dataset quality and sim-to-real gaps.

5 Limitations

While we demonstrate planning with image-based goals, our framework can be extended to accommodate text-specified goals. To mitigate the embodiment gap, we rely on approximately four hours of exploratory simulation data; while incorporating target-embodiment data could potentially remove this dependency, we leave this exploration to future work. Our current setup assumes static scenes without external agents (including pretraining datasets [44, 49]). Handling dynamic, interactive environments may require modeling stochasticity, e.g. via additional latent variables or diffusion-based prediction. However, such methods are often slower due to additional optimization or iterative denoising.

6 Conclusion

We explored the design of world models for learning dexterous hand–object interaction dynamics directly from human videos. A keypoint-based action representation, patch-level embeddings as state representation, and a hand-consistency loss together enable fine-grained modeling of dexterous behavior. Through several evaluations, we show that the model captures precise interaction dynamics under dexterous actions. Moreover, we demonstrate effective transfer from human-video pretraining to previously unseen robotic tasks such as grasping, reaching, and placing. We hope that our work will inspire future exploration into dexterous modeling and world models for robotics, which will unlock generalizable robots capable of increasingly complex tasks.

Acknowledgements

The authors thank Nicolas Ballas, Naren Devarakonda, Jitendra Malik, Tushar Nagarajan, Michael Psenka, Basile Terver, and Artem Zholus for helpful discussions and feedback. Prof. Khorrami's group was supported in part by NSF CMMI grant 2208189 and by the New York University Abu Dhabi (NYUAD) Center for Artificial Intelligence and Robotics (CAIR), funded by Tamkeen under the NYUAD Research Institute Award CG010.

References

1. Agarwal, A., Uppal, S., Shaw, K., Pathak, D.: Dexterous functional grasping. arXiv preprint arXiv:2312.02975 (2023) 4
2. Agarwal, N., Ali, A., Bala, M., Balaji, Y., Barker, E., Cai, T., Chattopadhyay, P., Chen, Y., Cui, Y., Ding, Y., et al.: Cosmos world foundation model platform for physical ai. arXiv preprint arXiv:2501.03575 (2025) 2, 3, 6, 10, 12
3. Akkaya, I., Andrychowicz, M., Chociej, M., Litwin, M., McGrew, B., Petron, A., Paino, A., Plappert, M., Powell, G., Ribas, R., et al.: Solving rubik's cube with a robot hand. arXiv preprint arXiv:1910.07113 (2019) 4

4. Alonso, E., Jelley, A., Micheli, V., Kanervisto, A., Storkey, A.J., Pearce, T., Fleuret, F.: Diffusion for world modeling: Visual details matter in atari. Advances in Neural Information Processing Systems **37**, 58757–58791 (2024) 4
5. An, S., Meng, Z., Tang, C., Zhou, Y., Liu, T., Ding, F., Zhang, S., Mu, Y., Song, R., Zhang, W., et al.: Dexterous manipulation through imitation learning: A survey. arXiv preprint arXiv:2504.03515 (2025) 3, 4
6. Assran, M., Bardes, A., Fan, D., Garrido, Q., Howes, R., Muckley, M., Rizvi, A., Roberts, C., Sinha, K., Zholus, A., et al.: V-jepa 2: Self-supervised video models enable understanding, prediction and planning. arXiv preprint arXiv:2506.09985 (2025) 2, 3, 4, 5, 10
7. Bahl, S., Mendonca, R., Chen, L., Jain, U., Pathak, D.: Affordances from human videos as a versatile representation for robotics. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 13778–13790 (2023) 4
8. Bai, Y., Tran, D., Bar, A., LeCun, Y., Darrell, T., Malik, J.: Whole-body conditioned egocentric video prediction. arXiv preprint arXiv:2506.21552 (2025) 2, 3, 4, 6, 7, 9, 10, 11, 12
9. Banerjee, P., Shkodrani, S., Moulon, P., Hampali, S., Han, S., Zhang, F., Zhang, L., Fountain, J., Miller, E., Basol, S., et al.: Hot3d: Hand and object tracking in 3d from egocentric multi-view videos. In: Proceedings of the Computer Vision and Pattern Recognition Conference. pp. 7061–7071 (2025) 4, 6
10. Bar, A., Zhou, G., Tran, D., Darrell, T., LeCun, Y.: Navigation world models. In: Proceedings of the Computer Vision and Pattern Recognition Conference. pp. 15791–15801 (2025) 2, 3, 4, 7, 9, 10, 11, 12
11. Bjorck, J., Castañeda, F., Cherniadev, N., Da, X., Ding, R., Fan, L., Fang, Y., Fox, D., Hu, F., Huang, S., et al.: Gr00t n1: An open foundation model for generalist humanoid robots. arXiv preprint arXiv:2503.14734 (2025) 4
12. Blattmann, A., Rombach, R., Ling, H., Dockhorn, T., Kim, S.W., Fidler, S., Kreis, K.: Align your latents: High-resolution video synthesis with latent diffusion models. In: CVPR (2023) 3
13. Brooks, T., Peebles, B., Holmes, C., DePue, W., Guo, Y., Jing, L., Schnurr, D., Taylor, J., Luhman, T., Luhman, E., Ng, C., Wang, R., Ramesh, A.: Video generation models as world simulators. URL https://openai. com/research/video-generation-models-as-world-simulators (2024) 3
14. Bruce, J., Dennis, M.D., Edwards, A., Parker-Holder, J., Shi, Y., Hughes, E., Lai, M., Mavalankar, A., Steigerwald, R., Apps, C., et al.: Genie: Generative interactive environments. In: Forty-first International Conference on Machine Learning (2024) 4
15. Chang, M., Gupta, S.: One-shot visual imitation via attributed waypoints and demonstration augmentation. arXiv preprint arXiv:2302.04856 (2023) 2, 3
16. Chen, B., Monso, D.M., Du, Y., Simchowitz, M., Tedrake, R., Sitzmann, V.: Diffusion forcing: Next-token prediction meets full-sequence diffusion. arXiv preprint arXiv:2407.01392 (2024) 3
17. Chen, C., Wu, Y.F., Yoon, J., Ahn, S.: Transdreamer: Reinforcement learning with transformer world models. arXiv preprint arXiv:2202.09481 (2022) 4
18. Chen, F., Yang, Z., Zhuang, B., Wu, Q.: Streaming video diffusion: Online video editing with diffusion models. arXiv preprint arXiv:2405.19726 (2024) 3
19. Chen, T., Tippur, M., Wu, S., Kumar, V., Adelson, E., Agrawal, P.: Visual dexterity: In-hand reorientation of novel and complex object shapes. Science Robotics **8**(84), eadc9244 (2023) 4

20. Chen, X., Wang, Y., Zhang, L., Zhuang, S., Ma, X., Yu, J., Wang, Y., Lin, D., Qiao, Y., Liu, Z.: Seine: Short-to-long video diffusion model for generative transition and prediction. In: ICLR (2023) 3
21. Chen, Z., Chen, S., Arlaud, E., Laptev, I., Schmid, C.: Vividex: Learning vision-based dexterous manipulation from human videos. In: 2025 IEEE International Conference on Robotics and Automation (ICRA). pp. 3336–3343. IEEE (2025) 4
22. Chi, C., Xu, Z., Feng, S., Cousineau, E., Du, Y., Burchfiel, B., Tedrake, R., Song, S.: Diffusion policy: Visuomotor policy learning via action diffusion. The International Journal of Robotics Research **44**(10-11), 1684–1704 (2025) 1, 3, 11, 13, 14
23. Collaboration, E., O'Neill, A., Rehman, A., Gupta, A., et al.: Open x-embodiment: Robotic learning datasets and rt-x models (2025) 1
24. Esser, P., Kulal, S., Blattmann, A., Entezari, R., Müller, J., Saini, H., Levi, Y., Lorenz, D., Sauer, A., Boesel, F., et al.: Scaling rectified flow transformers for high-resolution image synthesis. In: ICML (2024) 3
25. Fan, D., Tong, S., Zhu, J., Sinha, K., Liu, Z., Chen, X., Rabbat, M., Ballas, N., LeCun, Y., Bar, A., et al.: Scaling language-free visual representation learning. arXiv preprint arXiv:2504.01017 (2025) 10
26. Fearing, R.: Implementing a force strategy for object re-orientation. In: Proceedings. 1986 IEEE International Conference on Robotics and Automation. vol. 3, pp. 96–102 (1986) 4
27. Gao, K., Shi, J., Zhang, H., Wang, C., Xiao, J.: Vid-gpt: Introducing gpt-style autoregressive generation in video diffusion models. arXiv preprint arXiv:2406.10981 (2024) 3
28. Gavryushin, A., Wang, X., Malate, R.J., Yang, C., Jia, X., Goel, S., Liconti, D., Zurbrügg, R., Katzschmann, R.K., Pollefeys, M.: Maple: Encoding dexterous robotic manipulation priors learned from egocentric videos. arXiv preprint arXiv:2504.06084 (2025) 2, 3, 4
29. Ge, S., Hayes, T., Yang, H., Yin, X., Pang, G., Jacobs, D., Huang, J.B., Parikh, D.: Long video generation with time-agnostic vqgan and time-sensitive transformer. In: ECCV (2022) 3
30. Goswami, R.G., Krishnamurthy, P., LeCun, Y., Khorrami, F.: Osvi-wm: One-shot visual imitation for unseen tasks using world-model-guided trajectory generation. arXiv preprint arXiv:2505.20425 (2025) 2, 3, 5
31. Grauman, K., Westbury, A., Byrne, E., Chavis, Z., Furnari, A., Girdhar, R., Hamburger, J., Jiang, H., Liu, M., Liu, X., et al.: Ego4d: Around the world in 3,000 hours of egocentric video. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 18995–19012 (2022) 4
32. Ha, D., Schmidhuber, J.: Recurrent world models facilitate policy evolution. Advances in neural information processing systems **31** (2018) 4
33. Hafner, D., Lillicrap, T., Ba, J., Norouzi, M.: Dream to control: Learning behaviors by latent imagination. arXiv preprint arXiv:1912.01603 (2019) 4
34. Hafner, D., Yan, W., Lillicrap, T.: Training agents inside of scalable world models. arXiv preprint arXiv:2509.24527 (2025) 4
35. Haldar, S., Peng, Z., Pinto, L.: Baku: An efficient transformer for multi-task policy learning (2024) 1
36. Han, L., Trinkle, J.C.: Dextrous manipulation by rolling and finger gaiting. In: Proceedings. 1998 IEEE International Conference on Robotics and Automation (Cat. No. 98CH36146). vol. 1, pp. 730–735. IEEE (1998) 1, 4
37. Handa, A., Allshire, A., Makoviychuk, V., Petrenko, A., Singh, R., Liu, J., Makoviichuk, D., Van Wyk, K., Zhurkevich, A., Sundaralingam, B., Narang, Y.: Dextreme:

Transfer of agile in-hand manipulation from simulation to reality. In: 2023 IEEE International Conference on Robotics and Automation (ICRA). pp. 5977–5984 (2023) 4
38. Hansen, N., Su, H., Wang, X.: Td-mpc2: Scalable, robust world models for continuous control. arXiv preprint arXiv:2310.16828 (2023) 4
39. He, Z., Ai, B., Mu, T., Liu, Y., Wan, W., Fu, J., Du, Y., Christensen, H.I., Su, H.: Scaling cross-embodiment world models for dexterous manipulation. arXiv preprint arXiv:2511.01177 (2025) 4
40. Henschel, R., Khachatryan, L., Hayrapetyan, D., Poghosyan, H., Tadevosyan, V., Wang, Z., Navasardyan, S., Shi, H.: Streamingt2v: Consistent, dynamic, and extendable long video generation from text. arXiv preprint arXiv:2403.14773 (2024) 3
41. Ho, J., Jain, A., Abbeel, P.: Denoising diffusion probabilistic models. In: NeurIPS (2020) 3
42. Hong, W., Ding, M., Zheng, W., Liu, X., Tang, J.: Cogvideo: Large-scale pretraining for text-to-video generation via transformers. In: ICLR (2023) 3
43. Hong, Z., Liu, Y., Hou, H., Ai, B., Wang, J., Mu, T., Qin, Y., Gu, J., Su, H.: Learning particle-based world model from human for robot dexterous manipulation. In: 3rd RSS Workshop on Dexterous Manipulation: Learning and Control with Diverse Data (2025) 4
44. Hoque, R., Huang, P., Yoon, D.J., Sivapurapu, M., Zhang, J.: Egodex: Learning dexterous manipulation from large-scale egocentric video. arXiv preprint arXiv:2505.11709 (2025) 2, 4, 6, 9, 12, 13, 15
45. Hu, A., Russell, L., Yeo, H., Murez, Z., Fedoseev, G., Kendall, A., Shotton, J., Corrado, G.: Gaia-1: A generative world model for autonomous driving (2023) 1, 2, 4
46. HunyuanWorld, T.: Hunyuanworld 1.0: Generating immersive, explorable, and interactive 3d worlds from words or pixels. arXiv preprint (2025) 4
47. Jiang, T., Guan, Y., Ma, L., Xu, J., Meng, J., Chen, W., Zeng, Z., Li, L., Wu, D., Chen, R.: Dexsim2real2: Building explicit world model for precise articulated object dexterous manipulation. IEEE Transactions on Robotics (2025) 4
48. Kareer, S., Patel, D., Punamiya, R., Mathur, P., Cheng, S., Wang, C., Hoffman, J., Xu, D.: Egomimic: Scaling imitation learning via egocentric video. In: 2025 IEEE International Conference on Robotics and Automation (ICRA). pp. 13226–13233. IEEE (2025) 4
49. Khazatsky, A., Pertsch, K., Nair, S., Balakrishna, A., Dasari, S., Karamcheti, S., Nasiriany, S., Srirama, M.K., Chen, L.Y., Ellis, K., et al.: Droid: A large-scale in-the-wild robot manipulation dataset. arXiv preprint arXiv:2403.12945 (2024) 2, 4, 6, 9, 13, 15
50. Kim, B., Kim, T., Lee, J., Joo, H.: Dexterous world models. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (2026) 4
51. Kim, J., Kang, J., Choi, J., Han, B.: Fifo-diffusion: Generating infinite videos from text without training. arXiv preprint arXiv:2405.11473 (2024) 3
52. Kim, M.J., Pertsch, K., Karamcheti, S., Xiao, T., Balakrishna, A., Nair, S., Rafailov, R., Foster, E., Lam, G., Sanketi, P., Vuong, Q., Kollar, T., Burchfiel, B., Tedrake, R., Sadigh, D., Levine, S., Liang, P., Finn, C.: Openvla: An open-source vision-language-action model (2024) 1
53. LeCun, Y.: A path towards autonomous machine intelligence version 0.9. 2, 2022-06-27. Open Review **62**(1), 1–62 (2022) 2

54. Lee, S., Wang, Y., Etukuru, H., Kim, H.J., Shafiullah, N.M.M., Pinto, L.: Behavior generation with latent actions (2024) 1
55. Liang, J., Wu, C., Hu, X., Gan, Z., Wang, J., Wang, L., Liu, Z., Fang, Y., Duan, N.: Nuwa-infinity: Autoregressive over autoregressive generation for infinite visual synthesis. In: NeurIPS (2022) 3
56. Liu, X., Gong, C., Liu, Q.: Flow straight and fast: Learning to generate and transfer data with rectified flow. In: ICLR (2023) 3
57. Lu, T., Shu, T., Xiao, J., Ye, L., Wang, J., Peng, C., Wei, C., Khashabi, D., Chellappa, R., Yuille, A., et al.: Genex: Generating an explorable world. arXiv preprint arXiv:2412.09624 (2024) 4
58. Morgan, A.S., Hang, K., Wen, B., Bekris, K., Dollar, A.M.: Complex in-hand manipulation via compliance-enabled finger gaiting and multi-modal planning. IEEE Robotics and Automation Letters **7**(2), 4821–4828 (2022) 1
59. Nasiriany, S., Maddukuri, A., Zhang, L., Parikh, A., Lo, A., Joshi, A., Mandlekar, A., Zhu, Y.: Robocasa: Large-scale simulation of everyday tasks for generalist robots. In: Robotics: Science and Systems (RSS) (2024) 3, 6, 9, 13
60. Okamura, A., Smaby, N., Cutkosky, M.: An overview of dexterous manipulation. In: Proceedings 2000 ICRA. Millennium Conference. IEEE International Conference on Robotics and Automation. Symposia Proceedings (Cat. No.00CH37065). vol. 1, pp. 255–262 vol.1 (2000) 4
61. Oquab, M., Darcet, T., Moutakanni, T., Vo, H., Szafraniec, M., Khalidov, V., Fernandez, P., Haziza, D., Massa, F., El-Nouby, A., et al.: Dinov2: Learning robust visual features without supervision. arXiv preprint arXiv:2304.07193 (2023) 5, 10
62. Pavlakos, G., Shan, D., Radosavovic, I., Kanazawa, A., Fouhey, D., Malik, J.: Reconstructing hands in 3d with transformers. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 9826–9836 (2024) 4
63. Peebles, W., Xie, S.: Scalable diffusion models with transformers. In: ICCV (2023) 3, 7
64. Qi, H., Kumar, A., Calandra, R., Ma, Y., Malik, J.: In-hand object rotation via rapid motor adaptation. In: Conference on Robot Learning. pp. 1722–1732. PMLR (2023) 1
65. Qi, H., Yi, B., Suresh, S., Lambeta, M., Ma, Y., Calandra, R., Malik, J.: General in-hand object rotation with vision and touch. In: Conference on Robot Learning. pp. 2549–2564. PMLR (2023) 4
66. Qin, Y., Wu, Y.H., Liu, S., Jiang, H., Yang, R., Fu, Y., Wang, X.: Dexmv: Imitation learning for dexterous manipulation from human videos. In: European Conference on Computer Vision. pp. 570–587. Springer (2022) 4
67. Qiu, H., Xia, M., Zhang, Y., He, Y., Wang, X., Shan, Y., Liu, Z.: Freenoise: Tuning-free longer video diffusion via noise rescheduling. In: ICLR (2024) 3
68. Radosavovic, I., Shi, B., Fu, L., Goldberg, K., Darrell, T., Malik, J.: Robot learning with sensorimotor pre-training. In: Conference on Robot Learning. pp. 683–693. PMLR (2023) 4
69. Robine, J., Höftmann, M., Uelwer, T., Harmeling, S.: Transformer-based world models are happy with 100k interactions. arXiv preprint arXiv:2303.07109 (2023) 2
70. Romero, J., Tzionas, D., Black, M.J.: Embodied hands: Modeling and capturing hands and bodies together. ACM Transactions on Graphics, (Proc. SIGGRAPH Asia) **36**(6) (Nov 2017) 4, 6
71. Rubinstein, R.Y.: Optimization of computer simulation models with rare events. European Journal of Operational Research **99**(1), 89–112 (1997) 8

72. Shan, D., Geng, J., Shu, M., Fouhey, D.F.: Understanding human hands in contact at internet scale. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 9869–9878 (2020) 4
73. Shaw, K., Bahl, S., Pathak, D.: Videodex: Learning dexterity from internet videos. In: Conference on Robot Learning. pp. 654–665. PMLR (2023) 4
74. Siméoni, O., Vo, H.V., Seitzer, M., Baldassarre, F., Oquab, M., Jose, C., Khalidov, V., Szafraniec, M., Yi, S., Ramamonjisoa, M., et al.: Dinov3. arXiv preprint arXiv:2508.10104 (2025) 10
75. Singh, H.G., Loquercio, A., Sferrazza, C., Wu, J., Qi, H., Abbeel, P., Malik, J.: Hand-object interaction pretraining from videos. In: 2025 IEEE International Conference on Robotics and Automation (ICRA). pp. 3352–3360. IEEE (2025) 2, 3, 4
76. Sohl-Dickstein, J., Weiss, E., Maheswaranathan, N., Ganguli, S.: Deep unsupervised learning using nonequilibrium thermodynamics. In: ICML (2015) 3
77. Song, Y., Sohl-Dickstein, J., Kingma, D.P., Kumar, A., Ermon, S., Poole, B.: Score-based generative modeling through stochastic differential equations. In: ICLR (2021) 3
78. Tschannen, M., Gritsenko, A., Wang, X., Naeem, M.F., Alabdulmohsin, I., Parthasarathy, N., Evans, T., Beyer, L., Xia, Y., Mustafa, B., et al.: Siglip 2: Multilingual vision-language encoders with improved semantic understanding, localization, and dense features. arXiv preprint arXiv:2502.14786 (2025) 10
79. Wang, F.Y., Chen, W., Song, G., Ye, H.J., Liu, Y., Li, H.: Gen-l-video: Multi-text to long video generation via temporal co-denoising. arXiv preprint arXiv:2305.18264 (2023) 3
80. Wang, X., Kwon, T., Rad, M., Pan, B., Chakraborty, I., Andrist, S., Bohus, D., Feniello, A., Tekin, B., Frujeri, F.V., et al.: Holoassist: an egocentric human interaction dataset for interactive ai assistants in the real world. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 20270–20281 (2023) 6
81. Weng, W., Feng, R., Wang, Y., Dai, Q., Wang, C., Yin, D., Zhao, Z., Qiu, K., Bao, J., Yuan, Y., et al.: Art-v: Auto-regressive text-to-video generation with diffusion models. In: CVPRW (2024) 3
82. Xie, D., Xu, Z., Hong, Y., Tan, H., Liu, D., Liu, F., Kaufman, A., Zhou, Y.: Progressive autoregressive video diffusion models. arXiv preprint arXiv:2410.08151 (2024) 3
83. Yan, W., Zhang, Y., Abbeel, P., Srinivas, A.: Videogpt: Video generation using vq-vae and transformers. arXiv preprint arXiv:2104.10157 (2021) 3
84. Yang, R., Yu, Q., Wu, Y., Yan, R., Li, B., Cheng, A.C., Zou, X., Fang, Y., Cheng, X., Qiu, R.Z., et al.: Egovla: Learning vision-language-action models from egocentric human videos. arXiv preprint arXiv:2507.12440 (2025) 4
85. Yang, Y., Li, Y., Fermuller, C., Aloimonos, Y.: Robot learning manipulation action plans by" watching" unconstrained videos from the world wide web. In: Proceedings of the AAAI conference on artificial intelligence. vol. 29 (2015) 4
86. Yang, Z., Teng, J., Zheng, W., Ding, M., Huang, S., Xu, J., Yang, Y., Hong, W., Zhang, X., Feng, G., et al.: Cogvideox: Text-to-video diffusion models with an expert transformer. arXiv preprint arXiv:2408.06072 (2024) 3
87. Yu, C., Wang, P.: Dexterous manipulation for multi-fingered robotic hands with reinforcement learning: A review. Frontiers in Neurorobotics **16**, 861825 (2022) 4
88. Yu, J., Qin, Y., Wang, X., Wan, P., Zhang, D., Liu, X.: Gamefactory: Creating new games with generative interactive videos (2025) 4

89. Yu, Z., Zafeiriou, S., Birdal, T.: Dyn-hamr: Recovering 4d interacting hand motion from a dynamic camera. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (June 2025) 4
90. Zhang, Z., Wu, B., Wang, X., Luo, Y., Zhang, L., Zhao, Y., Vajda, P., Metaxas, D., Yu, L.: Avid: Any-length video inpainting with diffusion model. In: CVPR (2024) 3
91. Zheng, Z., Peng, X., Yang, T., Shen, C., Li, S., Liu, H., Zhou, Y., Li, T., You, Y.: Open-sora: Democratizing efficient video production for all. arXiv preprint arXiv:2412.20404 (2024) 3
92. Zhou, G., Dean, V., Srirama, M.K., Rajeswaran, A., Pari, J., Hatch, K., Jain, A., Yu, T., Abbeel, P., Pinto, L., Finn, C., Gupta, A.: Train offline, test online: A real robot learning benchmark (2023) 2
93. Zhou, G., Pan, H., LeCun, Y., Pinto, L.: Dino-wm: World models on pre-trained visual features enable zero-shot planning. arXiv preprint arXiv:2411.04983 (2024) 2, 4, 5, 10