



# Repurposing Video Diffusion for Embodiment-Aware Robot Data Synthesis

Junjie Ye<sup>1,2</sup> Rong Xue<sup>2</sup> Basile Van Hoorick<sup>1</sup> Pavel Tokmakov<sup>1</sup>

Muhammad Zubair Irshad<sup>1</sup> Yue Wang<sup>2</sup> Vitor Guizilini<sup>1</sup>

<sup>1</sup>Toyota Research Institute <sup>2</sup>USC Physical Superintelligence (PSI) Lab

<https://jay-ye.github.io/AnchorDream>

**Abstract**—The collection of large-scale and diverse robot demonstrations remains a major bottleneck for imitation learning, as real-world data acquisition is costly and simulators offer limited diversity and fidelity with pronounced sim-to-real gaps. While generative models present an attractive solution, existing methods often alter only visual appearances without creating new behaviors, or suffer from embodiment inconsistencies that yield implausible motions. To address these limitations, we introduce AnchorDream, an embodiment-aware world model that repurposes pretrained video diffusion models for robot data synthesis. AnchorDream conditions the diffusion process on robot motion renderings, anchoring the embodiment to prevent hallucination while synthesizing objects and environments consistent with the robot’s kinematics. Starting from only a handful of human teleoperation demonstrations, our method scales them into large, diverse, high-quality datasets without requiring explicit environment modeling. Experiments show that the generated data leads to consistent improvements in downstream policy learning, with relative gains of 36.4% in simulator benchmarks and nearly double performance in real-world studies. These results suggest that grounding generative world models<sup>1</sup> in robot motion provides a practical path toward scaling imitation learning.

## I. INTRODUCTION

Imitation learning is a core approach for robotic manipulation [1], [2]. By training on large-scale robot demonstrations [3], [4], robots can acquire complex behaviors without hand-designed rewards or task-specific controllers [5]–[7]. However, the effectiveness of imitation learning depends critically on the scale of available data [8]. Collecting large quantities of high-quality robot demonstrations in the real world is expensive. This data bottleneck remains a major obstacle for scaling robot learning.

A growing line of work attempts to scale imitation learning by augmenting existing demonstrations from two angles. The first is to expand the *observation* space [9]–[11], altering the visual appearance of demonstrations while leaving the motions unchanged. These methods typically rely on generative models [12], [13] to diversify scenes and objects. However, the trajectory distribution remains fixed, and no new behaviors are created. The second direction is to expand

<sup>1</sup>We use the term world model in a broader sense than its conventional usage in robotics and RL, where it typically refers to action-conditioned video prediction. Here, world model denotes a model that constructs coherent environments anchored to robot motion.



**Fig. 1: Overview of AnchorDream.** AnchorDream repurposes a pretrained video diffusion model as an embodiment-aware world model. Conditioned on robot motion videos, the model anchors the robot embodiment to prevent hallucination while synthesizing objects and environments consistent with the motion, enabling large-scale, high-quality demonstration generation from only a few real demonstrations.

the *motion* space, generating new trajectories beyond those originally collected. Such approaches, however, often depend on simulators [14], which require labor-intensive setup and suffer from a large real-to-sim gap, or explicit scene modeling [15], which limits their scalability across diverse environments.

To address these challenges, we take a different route by leveraging generative models. Video generative models [16], [17] trained on Internet-scale data, capture broad world priors including object appearances, scene layouts, and temporal consistency in motion. Unlike simulators, they require no handcrafted assets. For robotics, this suggests the potential to synthesize realistic and diverse training data at scale. However, the challenge is *embodiment grounding*. Off-the-shelf generative models are not constrained by embodiment and often hallucinate robot bodies or produce physically inconsistent motions. This highlights the need for a mechanism to ground these priors in real robot behavior.

We introduce AnchorDream, a framework that conditions video generative models on rendered robot trajectories to synthesize demonstrations directly in the visual domain. Our approach begins with a small set of real demonstrations and then heuristically expands trajectories using perturbations of key states and motion segments to generate new trajectories at scale. Instead of reconstructing full environments in a simulator, we render *only* the robot arm motions, without

any scene objects or backgrounds. These trajectory replays serve as the conditioning signal for a video generative model, which synthesizes objects, interactions, and environments consistent with the observed motions. Our key idea is to decouple trajectory and environment rendering, turning actions from an afterthought into a first-class citizen. By taking control of robot trajectories and rendering them deterministically first, plausible environments and objects are generated afterwards to convey the anchoring trajectory. This preserves embodiment consistency while producing photorealistic demonstrations that are immediately suitable for real-world policy training.

Extensive experiments in simulation and on real robots show that AnchorDream can expand small demonstration sets by more than an order of magnitude. The generated data leads to consistent gains in downstream policy learning, with relative improvements of 36.4% in simulator benchmarks, and nearly doubling performance in the real-world. These findings indicate that grounding diffusion priors in robot motion provides a practical path toward scaling imitation learning without the need for massive data collection or explicit environment modeling.

To summarize, our contributions are three-fold:

- We introduce AnchorDream, an embodiment-aware video generation framework that anchors pretrained video diffusion models in robot motion to synthesize trajectory-consistent demonstrations.
- We propose a decoupled trajectory–environment synthesizing paradigm, where robot trajectories are expanded and rendered deterministically, and environments are generated afterwards, avoiding explicit scene modeling while preserving embodiment consistency.
- We validate AnchorDream through extensive simulation and real-robot studies, showing its effectiveness for scaling imitation learning from only a handful of human demonstrations.

## II. RELATED WORK

Our work sits at the intersection of robot imitation learning, data augmentation, and generative models. We therefore discuss prior efforts in scaling up robotic datasets.

### A. Data Generation for Robot Learning

The high cost of collecting real-world robot demonstrations has motivated two primary alternatives: simulation and data augmentation. Simulators [18], [19] offer a cost-effective way to scale robot data. However, they are often limited by the diversity of available assets and suffer from the notorious sim-to-real gap. Domain randomization helps [20], yet requires careful tuning and rarely covers real-world visual and physical variation. Data augmentation provides another pathway by reusing existing real demonstrations instead of collecting new ones. Early methods involved simple transformations like random cropping and color jittering [21], [22]. More recently, generative models [16], [23], [24] have enabled more advanced augmentations. A prominent line of work focuses on augmenting the *observation*

space while keeping the robot’s actions fixed. For instance, ROSIE [9] performs text-guided inpainting to alter objects, backgrounds, and distractors. RoboEngine [10] provides a plug-and-play pipeline combining robot segmentation with task-aware background generation. While these approaches increase visual diversity, they do not expand the underlying distribution of robot trajectories or behaviors. AnchorDream goes beyond visual augmentation by generating new scenes conditioned on novel robot motions, thereby diversifying both observations and behaviors.

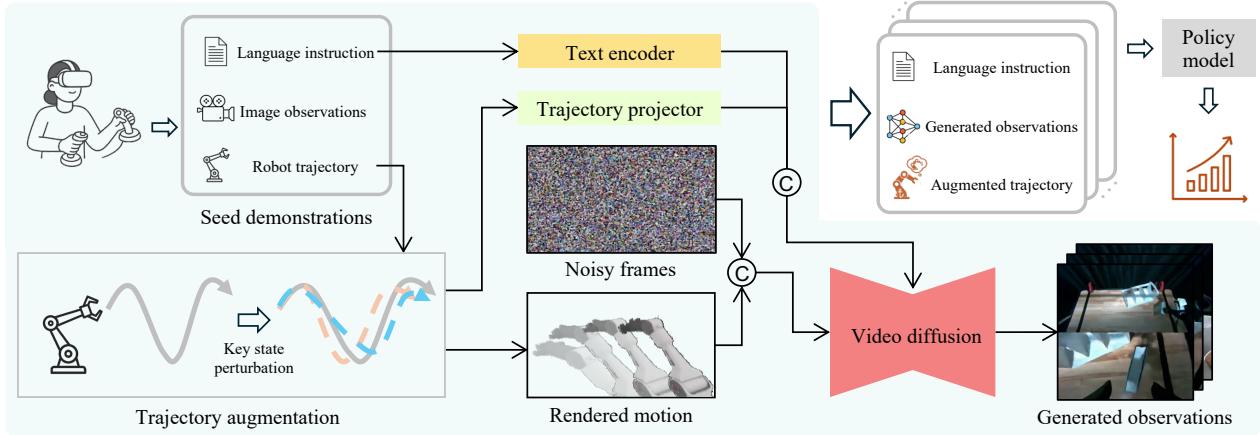
### B. Synthesizing Novel Robot Trajectories

To diversify behaviors, several approaches generate new trajectories, expanding the *action* space. MimicGen [14] generates new motions by composing sub-trajectories from human demonstrations and then uses a planner to execute them in a known, pre-built simulation environment to render new visual observations. This reliance on an explicit simulator, however, reintroduces the challenges of environment modeling and the sim-to-real gap. Real2Render2Real [25] follows a related idea by replaying perturbed real trajectories in simulation to generate novel demonstrations, but similarly depends on accurate environment reconstruction. DemoGen [15] sidesteps photorealistic rendering by operating in point clouds, recombining object-centric sub-trajectories to create obstacle-aware motions. Both expand the action distribution, but either require explicit environment modeling (e.g., simulation assets or reconstruction) or leave the pixel domain where most policies are trained. AnchorDream differs in that it sidesteps the need for explicit environment assets or simulator execution altogether. By decoupling trajectory expansion from scene generation, it leverages robot motion as the sole input for synthesizing diverse and coherent demonstrations, enabling scalability without environment reconstruction.

### C. Generative Models for Embodied Synthesis

Recently, there has been growing interest in using large-scale generative models, particularly video diffusion models, as implicit world models for robotics [10], [26]. These models, trained on vast internet datasets, possess rich priors about object physics, appearance, and temporal dynamics. DreamGen [26] leverages a video model to generate entire scenes, including the robot, from a text prompt and an initial image. It then uses an inverse dynamics model to extract actions from the generated video. However, this approach faces two key challenges: 1) video models often hallucinate the robot’s morphology, leading to kinematically infeasible motions, and 2) the accuracy of the extracted actions is bottlenecked by the quality of the generated video and the performance of the inverse dynamics model.

AnchorDream addresses these limitations with a fundamentally different conditioning scheme. Instead of generating the robot and scene jointly, we *anchor* the generation process on a video of the robot’s motion. By providing the robot’s embodiment as a strong prior, our model is constrained to synthesize only the surrounding environment and objects



**Fig. 2: Outline of our proposed AnchorDream.** Starting from a small set of human teleoperated demonstrations, new trajectories are created by perturbing key states and recombining motion segments to ensure kinematic feasibility. Each augmented trajectory is rendered as a robot-only motion video, which, together with the task description, conditions AnchorDream to synthesize realistic demonstrations where environment objects are consistent with the planned trajectory. This design anchors generation on robot motion, avoiding explicit scene reconstruction and reducing the need for labor-intensive environment modeling. The synthesized demonstrations are then used to train downstream imitation learning policies, enabling limited human data to be expanded into large-scale, high-quality datasets that empower stronger policy learning.

in a manner consistent with the robot’s kinematics. This decoupling avoids robot hallucination and bypasses the need for an inverse dynamics model, allowing us to synthesize high-quality, kinematically-grounded demonstration videos directly suitable for imitation learning.

### III. METHODOLOGY

#### A. Preliminaries

1) *Video Generative Models*: Diffusion-based video generative models learn a distribution over sequences of frames by iterative denoising. Given a video sequence of  $T$  frames  $\mathbf{o}_{1:T} = \{o_1, \dots, o_T\}$ , the training objective is to approximate the conditional distribution

$$p_\theta(\mathbf{o}_{1:T} | \mathbf{c}), \quad (1)$$

where  $\mathbf{c}$  denotes conditioning variables such as text, actions, or other signals. The model is trained to denoise a corrupted version of the sequence through a Markov chain, converging to realistic samples at inference time.

Through large-scale training, these models encode priors on visual appearance, spatial layouts, and temporal consistency. For robotics, such priors can be reused to synthesize diverse and photorealistic demonstrations. However, they are not inherently constrained by robot embodiment, and naive generation often leads to hallucinated robot bodies, inconsistent motions, and a lack of ground truth action labels [23], [27]. A promising direction is to condition the generative process on robot trajectories, which provides embodiment grounding and helps ensure that the synthesized demonstrations remain consistent with the robot’s kinematics.

2) *Procedural Trajectory Synthesis*: Let a robot trajectory be denoted as a sequence of states and actions

$$\tau = \{(s_1, a_1), (s_2, a_2), \dots, (s_T, a_T)\}. \quad (2)$$

Procedural trajectory synthesis aims to expand a dataset  $\mathcal{D} = \{\tau_i\}$  into a larger set  $\mathcal{D}'$  by applying transformations to existing trajectories. MimicGen [14] is a representative

approach: it segments demonstrations into object-centric subtasks, transforms them to match new scene layouts, and executes them in simulation. Formally, given a base trajectory  $\tau$ , a transformation operator  $\mathcal{T}$  produces a new trajectory  $\tau' = \mathcal{T}(\tau)$ . Validation of  $\tau'$  is performed in simulation to ensure task success, and successful samples are retained.

While effective in simulation, such approaches require explicit scene modeling and physics validation, which are costly and not easily generalizable.

#### B. AnchorDream

1) *Overview*: As shown in Fig. 2, AnchorDream generates large-scale demonstrations by anchoring video generative models on robot motion. The method separates robot trajectories from environments. Trajectories are expanded first, then used to condition a video model that produces photorealistic demonstrations. This design expands both the motion space and the observation space without requiring explicit scene modeling or access to simulation assets.

Formally, given a small seed dataset  $\mathcal{D}_0 = \{\tau_i\}$ , AnchorDream aims to construct a dataset  $\mathcal{D}' = \{(\tau', \mathbf{o}_{1:T})\}$  by synthesizing new trajectories  $\tau'$  and generating corresponding observation sequences  $\mathbf{o}_{1:T}$  conditioned on them.

2) *Trajectory Expansion*: We first generate new trajectories  $\tau'$  by perturbing and recombining motion primitives from existing demonstrations, following prior work [14], [15]. Specifically, key states such as contact points are manually grounded and shifted within a range. Object-centric trajectories are then stitched behind to adapt the surrounding motion, ensuring smooth transitions between segments. This produces a large pool of trajectories that remain feasible under the robot’s embodiment.

3) *Robot-Only Rendering*: For each synthesized trajectory  $\tau'$ , we render only the robot arm motion, without objects, textures, or backgrounds:

$$r_{1:T} = \text{Render}(\tau'), \quad (3)$$

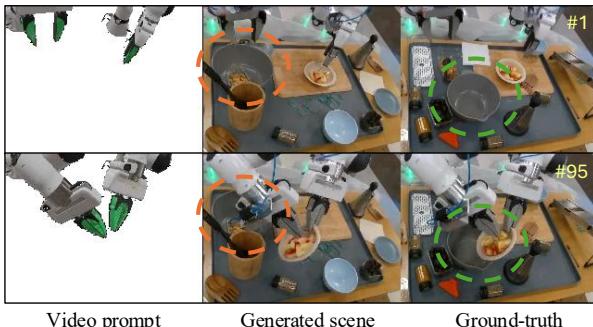
where  $\text{Render}(\cdot)$  denotes projecting the robot's 3D geometry (from its URDF/Mesh model) into a 2D image using the specified camera intrinsics and extrinsics. This produces a sequence of frames  $r_{1:T}$  showing only the robot moving through the trajectory from the chosen camera viewpoints. The result is a clean, embodiment-consistent conditioning signal. By excluding explicit environment modeling, this step avoids the cost of replicating real-world scenes in simulation.

*4) Video Generation:* The video model takes the rendered motion traces  $r_{1:T}$  and language instructions  $l$ , and synthesizes complete demonstrations with plausible environment and object layouts:

$$\mathbf{o}_{1:T} \sim p_\theta(\mathbf{o}_{1:T} | r_{1:T}, l), \quad (4)$$

where  $\mathbf{o}_{1:T}$  denotes photorealistic observations consistent with  $\tau'$ . To support multi-view generation, rendered frames from different viewpoints are concatenated spatially before being passed to the model. To make the motion trace videos compatible with pretrained large-scale video diffusion models, we concatenate them with the initialized noisy input and expand the number of input channels of the first model layer by a factor of two. By anchoring generation on robot motion, the model preserves embodiment while filling in plausible objects and environments, thereby enabling AnchorDream to transform a handful of real demonstrations into large-scale, kinematically grounded datasets for imitation learning.

*5) Global Trajectory Conditioning:* To generate long-horizon episodes, we autoregressively extend sequences by conditioning each new generation window on the last few frames of the previously generated clip. While this strategy allows arbitrarily long rollouts, our preliminary experiments (see Fig. 3) reveal that the synthesized scenes sometimes become incompatible with the robot's future motion, as the model only observes the current window and lacks awareness of upcoming waypoints. To address this issue, we provide the model with the entire trajectory  $\tau'$  as an additional conditioning signal. To help the model localize the current inference window within the global context, we augment each waypoint with a binary indicator  $\varphi$  marking



**Fig. 3: Effect of missing global trajectory conditioning.** Without global conditioning, the generated bowl (highlighted in orange) is placed at a location that is visually plausible but not consistent with the robot's later motion. The ground-truth bowl location (highlighted in green) shows where the apple slices are eventually poured. This illustrates that generations based only on local context may fail to anticipate future motions.

---

#### Algorithm 1: Working pipeline of AnchorDream

---

```

Input: Seed demonstrations  $\mathcal{D}_0 = \{\tau_i\}$ ; heuristic operators  $\mathcal{T}$ ; renderer  $\text{Render}(\cdot)$ ; video model  $p_\theta$ ; augmentation count  $K$  per seed.
Output: Augmented dataset  $\mathcal{D}' = \{(\tau', \mathbf{o}_{1:T})\}$ .
 $\mathcal{D}' \leftarrow \emptyset$ 
foreach  $\tau \in \mathcal{D}_0$  do
    for  $k = 1$  to  $K$  do
        Sample operator  $\mathcal{T}_k \sim \mathcal{T}$  with parameters  $\phi_k$ 
        // Heuristic trajectory expansion
         $\tau' \leftarrow \mathcal{T}_k(\tau; \phi_k)$ 
        // Robot-only rendering (no scene or objects)
         $r_{1:T} \leftarrow \text{Render}(\tau')$ 
        // Trajectory-conditioned video synthesis
         $\mathbf{o}_{1:T} \sim p_\theta(\mathbf{o}_{1:T} | r_{1:T}, l, [\tau', \varphi])$ 
    return  $\mathcal{D}'$ 

```

---

whether it lies inside the current generation window. The global trajectory is projected into an embedding space and concatenated with the language embeddings to form the final conditioning input. The generation process in Eq. 4 is thus reformulated as:

$$\mathbf{o}_{1:T} \sim p_\theta(\mathbf{o}_{1:T} | r_{1:T}, l, [\tau', \varphi]), \quad (5)$$

This global conditioning exposes the model to future motions and ensures that synthesized environments remain coherent over long horizons, with scene layouts aligned to the robot's planned actions. In practice, this reduces layout drift and prevents scene-object mismatches during extended rollouts.

*6) Decoupling Trajectories and Environments:* Our key idea is to decouple the two factors. Trajectories are fixed first, then environments are generated afterwards. This avoids explicit scene modeling and ensures trajectory-environment consistency. The final output is a dataset  $\mathcal{D}' = \{(\tau', \mathbf{o}_{1:T})\}$  containing trajectory-consistent, photorealistic demonstrations. Each sample pairs a kinematically feasible robot trajectory with a synthesized visual sequence aligned to that motion. Because the data is produced directly in the visual domain, it can be used to train policies without additional transfer steps, enabling efficient scaling from a small set of seed demonstrations. A pseudocode summary of the overall data synthesis pipeline is provided in Alg. 1.

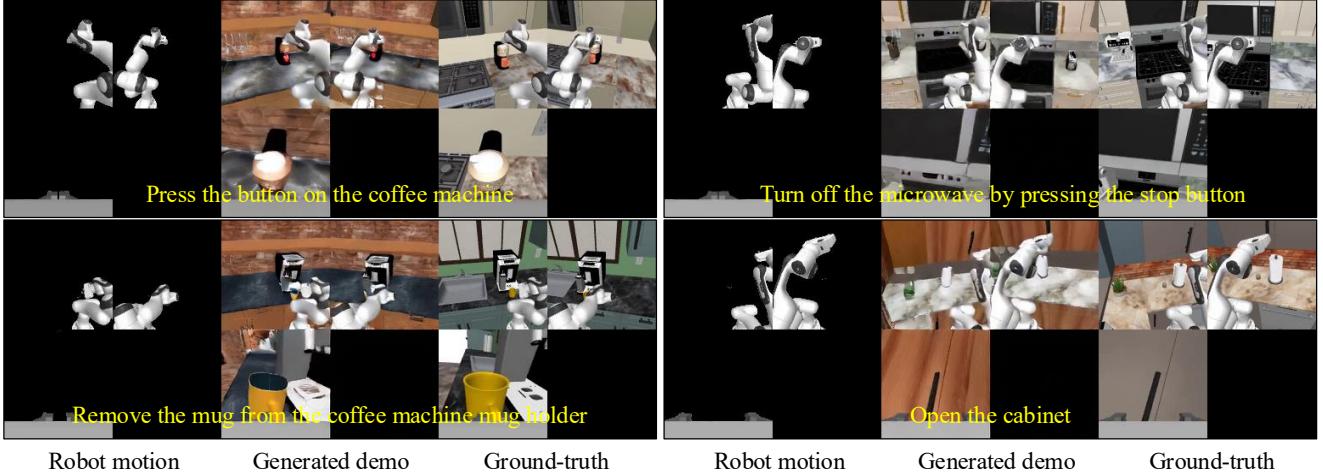
## IV. EXPERIMENTS

In this section, we study the following questions: (1) Can AnchorDream empower better policy from a small seed set, and how close does it get to the simulator-executed upper bound? (Sec. IV-B) (2) Can policies benefit from scaling AnchorDream-generated data? (Sec. IV-C) (3) Which design choices of AnchorDream matter most? (Sec. IV-D) (4) Does AnchorDream transfer to real robots and provide practical gains? (Sec. IV-E)

**TABLE I: Success rate comparison of policies trained with different data regimes.** AnchorDream consistently improves policy performance over Human50 across all skills and approaches the policy trained with MimicGen300, verifying the effectiveness of anchoring video diffusion on robot motion for high-quality demonstration synthesis.

	pick and place	doors	drawers	turning levers	twisting knobs	insertion	pressing buttons	Average (%)
Human50	1.8	31.0	42.0	36.0	10.0	12.0	55.3	22.5
w/ <b>AnchorDream300</b>	4.3	41.5	48.0	54.7	21.0	14.0	68.7	30.7
w/ MimicGen300*	5.8	54.0	57.0	64.7	24.0	14.0	51.3	33.3

\*MimicGen300 serves as an *oracle* upper bound due to its reliance on privileged simulator access.



**Fig. 4: Qualitative results on RoboCasa.** Comparison between rendered robot motion inputs, generated demonstrations, and ground-truth scenes across several tasks. The synthesized demonstrations preserve robot embodiment while producing diverse and visually coherent environments with object placements and interactions that align with the intended motions. These examples illustrate how AnchorDream translates abstract motion traces into realistic demonstrations, enriching the training distribution beyond the limited human demonstrations.

### A. Experimental Setup

1) *Evaluation protocol:* We perform empirical evaluations of AnchorDream in both simulation and real-world settings. For simulation, we use the RoboCasa [28] benchmark, which consists of 24 tabletop tasks, each with 50 human teleoperated demonstrations. RoboCasa further categorizes these 24 tasks into seven foundational manipulation skills, and we report both the average success rate within each skill and the overall average across all tasks unless otherwise noted. For real-world experiments, we design 6 everyday manipulation tasks, including placing a book on a shelf (BookToShelf), opening a drawer (OpenDrawer), closing a drawer (CloseDrawer), placing a toy in a plate (ToyToPlate), grasping and tilting a cup to pour into a bowl (PourToBowl), and sweeping coffee beans with a brush (SweepCoffeeBeans), and collect 50 human demonstrations for each using a single-arm PiPER robot. All tasks are evaluated for 50 rollouts for simulation studies and 20 rollouts for real-world evaluations.

2) *Training setup:* Unless otherwise specified, AnchorDream is fine-tuned from Cosmos-Predict2 2B [23] using the small set of available human teleoperation demonstrations in the simulator and real-world domains, respectively. The training is performed on 8 NVIDIA A100 GPUs over three days with LoRA [29]. At inference, the video diffusion model generates sequences of 189 frames at a resolution of 128x128 for simulation studies and 180x320 for real-world experiments. In RoboCasa, we adopt observations from two static side-

**TABLE II: Comparison of policies trained with only Human50, DreamGen10K, or AnchorDream300.** Training solely on AnchorDream300 slightly surpasses Human50 and remains competitive despite using far fewer demonstrations than DreamGen10K.

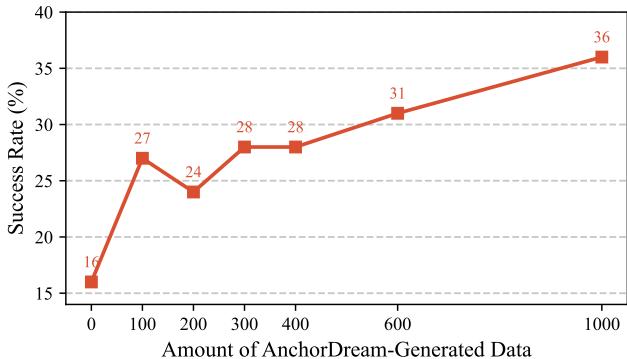
	Human50	DreamGen10K*	AnchorDream300
Average (%)	22.5	20.6	<b>24.8</b>

\*Value taken from the original paper [26].

view cameras together with a wrist-mounted camera, while in real-world settings we use a third-person static camera in combination with a wrist camera. For data synthesis, we render robot-only motion videos from trajectories using RoboCasa [28] in simulation and RoboTwin [30] in the real world. Importantly, AnchorDream leverages these simulators solely for rendering and inverse kinematics (IK) calculations, without accessing privileged environment state, simulating dynamics, or executing rollouts. Without loss of generality, we adopt BC-Transformer [31] for simulator experiments and Diffusion Policy [2] for real-world studies to examine the effect of AnchorDream demonstrations on policy learning.

### B. How much can AnchorDream empower policy learning?

To assess whether AnchorDream improves policy performance from a small seed, we consider three data regimes in a multi-task training setup: 1) Human50: train on the 50 original demonstrations per task; 2) w/ MimicGen300: expands each task with 300 additional trajectories obtained by applying MimicGen’s [14] heuristic trajectory generation strategy and then executing those trajectories in the simulator to collect paired observations; 3) w/ AnchorDream300:



**Fig. 5: Effect of scaling AnchorDream-generated data.** Comparison of policies trained with Human50 alone (0 on the x-axis), or with Human50 plus different amount of AnchorDream-generated demonstrations on a representative subset of RoboCasa tasks. Performance improves steadily as more synthesized data are added, confirming the effectiveness of scaling AnchorDream for stronger policy learning.

**TABLE III: Ablation on design choices in AnchorDream.** Comparison of policies trained with Human50 alone, Human50+AnchorDream300, and two ablated variants without global trajectory conditioning or with a shortened inference window. Both ablations reduce performance relative to the full model, but still surpass Human50, verifying the robustness of AnchorDream for demonstration synthesis.

	Average (%)
Human50	22.5
Human50 + AnchorDream300	<b>30.7</b>
w/o global trajectory (IV-D.1)	26.6
w/ shortened inference window (IV-D.2)	28.1

use the same set of additional trajectories, but instead of executing them in the simulator, render robot-only motion videos and prompt AnchorDream to synthesize the corresponding observations. Since MimicGen demonstrations are realized in the simulator with privileged access to environment state, they provide the best-possible version of these trajectories, and we therefore treat w/ MimicGen300 as an upper bound for evaluating AnchorDream.

1) *Quantitative results:* As shown in Table I, AnchorDream consistently improves policy performance across all skills. Training with 50 human demonstrations alone achieves an average success rate of 22.5%, while adding 300 AnchorDream-generated demonstrations raises this to 30.7%, a **36%** relative improvement. The performance also approaches 33.3% achieved with 300 MimicGen demonstrations, which can be regarded as an *oracle* upper bound since they rely on privileged access to environment assets and simulator execution. These results verify that anchoring video diffusion on robot motion provides high-quality synthesized demonstrations that substantially empower imitation learning, narrowing much of the gap to simulator-executed data expansion without requiring explicit environment modeling.

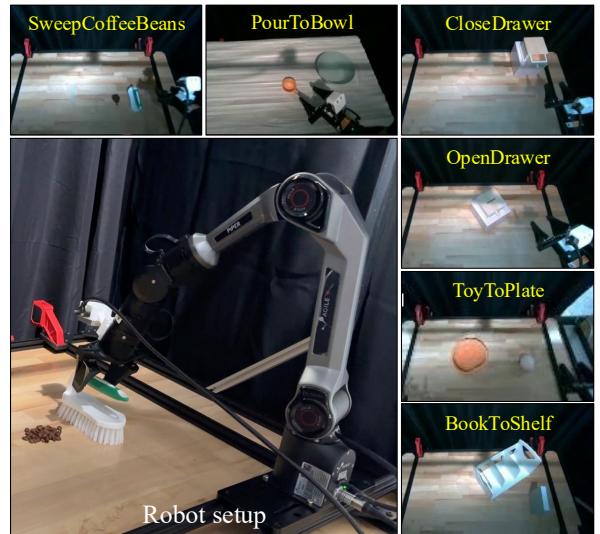
2) *Qualitative results:* To further illustrate the effect of AnchorDream, Figure 4 presents qualitative examples comparing the input robot-only motion videos, the generated demonstrations, and the corresponding ground-truth scenes. The generated demonstrations not only preserve embodiment fidelity but also produce diverse scenes with layouts and

object interactions that closely align with the ground truth. These examples confirm that the model can translate abstract motion traces into visually coherent and varied task executions, enriching the training distribution beyond what is available in the original demonstrations.

3) *Training with generated data alone:* We also explore how far we can go with AnchorDream-generated data alone. Specifically, we train a multi-task policy using only AnchorDream300 demonstrations and compare its performance with a policy trained on Human50. As shown in Table II, policies trained solely with AnchorDream data slightly outperform those trained with the original 50 human demonstrations (24.8% vs. 22.5%). We further compare against DreamGen [26], which generates demonstrations by producing full robot scenes with a video model and then recovering actions via an inverse dynamics model. Despite using 10k generated demonstrations per task and a foundational model [7] pretrained on large-scale robot datasets, DreamGen10K achieves only 20.6% average success rate. In contrast, AnchorDream anchors generation on robot motion, which helps avoid embodiment hallucinations and yields demonstrations that are more consistent with downstream policy learning.

### C. Can scaling AnchorDream data help?

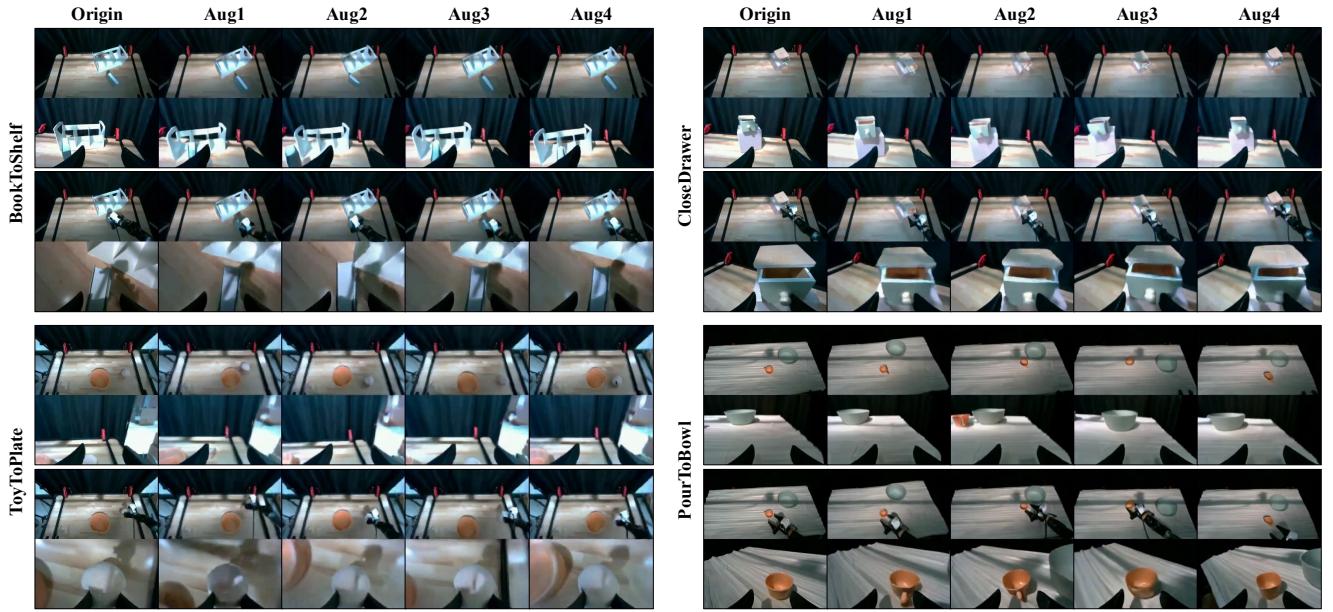
We further study whether increasing the number of synthesized demonstrations contributes to stronger policy learning. On a representative subset of seven RoboCasa tasks that cover foundational skills, we expand each task from 50 human teleoperation demonstrations to 50 plus varying amounts of AnchorDream-generated demonstrations, ranging from 100 up to 1000. As shown in Fig. 5, policy performance improves steadily with more synthesized data, rising from



**Fig. 6: Real-world evaluation setup.** Six everyday manipulation tasks are used in our real-world evaluation: sweeping coffee beans with a brush, grasping and tilting a cup to pour into a bowl, closing a drawer, opening a drawer, placing a toy in a plate, and placing a book on a shelf. The lower panel shows the PIPER robot platform used for data collection and evaluation.

**TABLE IV: Real-robot policy performance.** Comparison of policies trained with 50 human demonstrations per task (Human50) and with Human50 plus  $10\times$  AnchorDream-generated demonstrations across six everyday manipulation tasks. Augmenting with synthesized demonstrations consistently improves success rates on all tasks and raises the overall average, verifying the effectiveness of AnchorDream in real-world settings.

	SweepCoffeeBeans	PourToBowl	OpenDrawer	CloseDrawer	ToyToPlate	BookToShelf	Average (%)
Human50	35	0	0	30	85	20	28
w/ AnchorDream500	95	35	25	75	100	45	63



**Fig. 7: Real-robot qualitative results.** Example visualizations of synthesized demonstrations for several tasks. Each column shows the original trajectory (Origin) and several augmented variants (Aug1–Aug4). The generated demonstrations remain visually realistic, while the augmented trajectories steer scene layouts to diversify object positions and interactions, providing greater variability in the training data compared to the original human demonstrations.

the baseline with Human50 to substantially higher success rates at larger scales. Despite small fluctuations at lower data sizes, the overall trend indicates that scaling AnchorDream data consistently boosts policy performance.

#### D. Design Analyses

We analyze two key design choices that affect long-horizon coherence and embodiment grounding in RoboCasa.

1) *Global trajectory conditioning*: The scene layout often needs to be consistent with future motion beyond the current video generation window (see Fig. 3). Global trajectory conditioning helps the model consider future motions while “imagining” the scene and object layout. As shown in the third row of Table III, removing this conditioning reduces the policy success rate from 30.7% to 26.6%, indicating that global trajectory context is crucial for generating coherent long-horizon demonstrations.

2) *Long inference window*: We shorten the diffusion inference window from 189 frames to 93 frames and generate long sequences autoregressively to analyze the effect of the generation window in data synthesis. As shown in Table III, the success rate decreases from 30.7% to 28.1%, verifying that longer inference windows are important for maintaining temporal consistency across generated sequences.

Both variants nevertheless outperform the Human50 baseline at 22.5%, demonstrating that AnchorDream remains effective even under less favorable design choices, verifying

the robustness of anchoring video diffusion on robot motion for generating useful demonstrations.

#### E. Real-Robot Evaluation

To verify the effectiveness in real-world settings, we evaluate AnchorDream with six everyday manipulation tasks using a PiPER robot platform as shown in Fig. 6, manually collecting 50 human demonstrations per task and fine-tuning AnchorDream on this data. To expand the teleoperation trajectories, we segment each trajectory into object-centric sub-trajectories following [15], then randomly perturb the key states by up to  $\pm 10$  cm in the horizontal plane, render the resulting robot-only motion sequences in [30], and synthesize demonstrations with AnchorDream, expanding each task by  $10\times$ . Figure 7 provides some qualitative visualizations, which show that the synthesized demonstrations are visually realistic and the augmented trajectories successfully steer generated scene layouts, enriching the training distribution.

As shown in Table IV, augmenting the human demonstrations with synthesized data leads to substantial performance gains across all six tasks. Training on the original 50 demonstrations achieves an average success rate of 28.0%. Adding the  $10\times$  AnchorDream-generated demonstrations raises this to 60.0%, doubling the performance. Per-task results indicate consistent benefits. For instance, success on SweepCoffeeBeans improves from 35% to 95% and CloseDrawer from 30% to 75%. These gains confirm that

the synthesized demonstrations are not only visually realistic but also effective for policy learning.

Overall, the results demonstrate that AnchorDream can convert a small seed set of human demonstrations into large-scale, diverse datasets that significantly empower real-robot policies. This validates the practicality of leveraging embodiment-aware video diffusion for scaling imitation learning in real-world manipulation.

## V. CONCLUSION

We present AnchorDream, an embodiment-aware world model that repurposes pretrained video diffusion models for robot data synthesis. By anchoring generation on robot motion, AnchorDream produces kinematically grounded and visually realistic demonstrations, enabling scalable imitation learning without explicit environment modeling or simulator rollouts. These results verify the effectiveness of anchoring video diffusion on robot motion as a practical path to large-scale policy learning and point toward integrating embodiment priors with generative models to expand diversity and usability of synthesized robot data. While our study focuses on tabletop manipulation tasks, extending AnchorDream to broader domains such as mobile or long-horizon manipulation offers an exciting avenue for future work.

## VI. ACKNOWLEDGMENT

We thank our friends and colleagues, including Jiawei Yang, Chen Xu, Masha Itkina, and Mingtong Zhang, for their helpful discussions and insightful suggestions. The USC Physical Superintelligence Lab acknowledges generous support from Toyota Research Institute, Dolby, Google DeepMind, Capital One, Nvidia, and Qualcomm. Yue Wang is also supported by a Powell Research Award.

## REFERENCES

- [1] T. Z. Zhao, V. Kumar, S. Levine, and C. Finn, “Learning Fine-Grained Bimanual Manipulation with Low-Cost Hardware,” in *Proceedings of the Robotics: Science and Systems (RSS)*, 2023.
- [2] C. Chi, S. Feng, Y. Du, Z. Xu, E. Cousineau, B. Burchfiel, and S. Song, “Diffusion Policy: Visuomotor Policy Learning via Action Diffusion,” in *Proceedings of the Robotics: Science and Systems (RSS)*, 2023.
- [3] Open X-Embodiment Collaboration et al., “Open X-Embodiment: Robotic Learning Datasets and RT-X Models,” in *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, 2024.
- [4] A. Khazatsky, K. Pertsch, S. Nair, A. Balakrishna, and S. D. et al., “DROID: A Large-Scale In-The-Wild Robot Manipulation Dataset,” in *Proceedings of Robotics: Science and Systems (RSS)*, 2024.
- [5] TRI LBM Team et al., “A Careful Examination of Large Behavior Models for Multitask Dexterous Manipulation,” *arXiv preprint arXiv:2507.05331*, 2025.
- [6] K. Black, N. Brown, D. Driess, A. Esmail, M. Equi, C. Finn, N. Fusai, L. Groom, K. Hausman, B. Ichter, S. Jakubczak, T. Jones, L. Ke, S. Levine, A. Li-Bell, M. Mothukuri, S. Nair, K. Pertsch, L. X. Shi, J. Tanner, Q. Vuong, A. Walling, H. Wang, and U. Zhilinsky, “πο: A Vision-Language-Action Flow Model for General Robot Control,” in *Proceedings of the Robotics: Science and Systems (RSS)*, 2025.
- [7] NVIDIA, “GR00T N1: An Open Foundation Model for Generalist Humanoid Robots,” *arXiv preprint arXiv:2503.14734*, 2025.
- [8] F. Lin, Y. Hu, P. Sheng, C. Wen, J. You, and Y. Gao, “Data Scaling Laws in Imitation Learning for Robotic Manipulation,” in *Proceedings of the International Conference on Learning Representations (ICLR)*, 2024.
- [9] T. Yu, T. Xiao, A. Stone, J. Tompson, A. Brohan, S. Wang, J. Singh, C. Tan, M. Dee, J. Peralta, B. Ichter, K. Hausman, and F. Xia, “Scaling Robot Learning with Semantically Imagined Experience,” in *Proceedings of the Robotics: Science and Systems (RSS)*, 2023.
- [10] C. Yuan, S. Joshi, S. Zhu, H. Su, H. Zhao, and Y. Gao, “Robo-Engine: Plug-and-Play Robot Data Augmentation with Semantic Robot Segmentation and Background Generation,” in *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2025.
- [11] Y. Fang, Y. Yang, X. Zhu, K. Zheng, G. Bertasius, D. Szafrir, and M. Ding, “ReBot: Scaling Robot Learning with Real-to-Sim-to-Real Robotic Video Synthesis,” in *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2025.
- [12] S. Wang, C. Saharia, C. Montgomery, J. Pont-Tuset, S. Noy, S. Pellegri, Y. Onoe, S. Laszlo, D. J. Fleet, R. Soricut, J. Baldridge, M. Norouzi, P. Anderson, and W. Chan, “Imagen Editor and Edit-Bench: Advancing and Evaluating Text-Guided Image Inpainting,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023.
- [13] J. Ho, T. Salimans, A. Gritsenko, W. Chan, M. Norouzi, and D. J. Fleet, “Video Diffusion Models,” in *Proceedings of the Advances in Neural Information Processing Systems (NeurIPS)*, vol. 35, 2022.
- [14] A. Mandlekar, S. Nasiriany, B. Wen, I. Akinola, Y. Narang, L. Fan, Y. Zhu, and D. Fox, “MimicGen: A Data Generation System for Scalable Robot Learning using Human Demonstrations,” in *Proceedings of the Conference on Robot Learning (CoRL)*, 2023.
- [15] Z. Xue, S. Deng, Z. Chen, Y. Wang, Z. Yuan, and H. Xu, “Demogen: Synthetic demonstration generation for data-efficient visuomotor policy learning,” in *Proceedings of the Robotics: Science and Systems (RSS)*, 2025.
- [16] Team Wan, “Wan: Open and Advanced Large-Scale Video Generative Models,” *arXiv preprint arXiv:2503.20314*, 2025.
- [17] NVIDIA, “Cosmos World Foundation Model Platform for Physical AI,” *arXiv preprint arXiv:2501.03575*, 2025.
- [18] M. Savva, A. Kadian, O. Maksymets, Y. Zhao, E. Wijmans, B. Jain, J. Straub, J. Liu, V. Koltun, J. Malik, D. Parikh, and D. Batra, “Habitat: A Platform for Embodied AI Research,” in *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2019.
- [19] C. Gan, J. Schwartz, S. Alter, D. Mrowca, and M. S. et al., “ThreeD-World: A Platform for Interactive Multi-Modal Physical Simulation,” in *Proceedings of the Annual Conference on Neural Information Processing Systems (NeurIPS) Datasets and Benchmarks Track*, 2021.
- [20] J. Tobin, R. Fong, A. Ray, J. Schneider, W. Zaremba, and P. Abbeel, “Domain Randomization for Transferring Deep Neural Networks from Simulation to the Real World,” in *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2017.
- [21] M. Laskin, K. Lee, A. Stooke, L. Pinto, P. Abbeel, and A. Srinivas, “Reinforcement Learning with Augmented Data,” in *Proceedings of the Advances in Neural Information Processing Systems (NeurIPS)*, vol. 33, 2020.
- [22] D. Yarats, I. Kostrikov, and R. Fergus, “Image Augmentation Is All You Need: Regularizing Deep Reinforcement Learning from Pixels,” in *Proceedings of the International Conference on Learning Representations (ICLR)*, 2021.
- [23] N. Cosmos, “Cosmos-Predict2: General-Purpose World Foundation Models for Physical AI,” <https://github.com/nvidia-cosmos/cosmos-predict2>, 2025, apache License 2.0, accessed December 15, 2025.
- [24] H. Abu Alhaija, J. Alvarez, M. Bala, T. Cai, T. Cao, L. Cha, J. Chen, M. Chen, F. Ferroni, S. Fidler, D. Fox, Y. Ge, J. Gu, A. Hassani, M. Isaev, P. Jannaty, S. Lan, T. Lasser, H. Ling, M.-Y. Liu, X. Liu, Y. Lu, A. Luo, Q. Ma, H. Mao, F. Ramos, X. Ren, T. Shen, S. Tang, T.-C. Wang, J. Wu, J. Xu, S. Xu, K. Xie, Y. Ye, X. Yang, X. Zeng, and Y. Zeng, “Cosmos-Transfer1: Conditional World Generation with Adaptive Multimodal Control,” *arXiv preprint arXiv:2503.14492*, 2025.
- [25] J. Yu, L. Fu, H. Huang, K. El-Refai, R. A. Ambrus, R. Cheng, M. Z. Irshad, and K. Goldberg, “Real2Render2Real: Scaling Robot Data Without Dynamics Simulation or Robot Hardware,” in *Proceedings of the Conference on Robot Learning (CoRL)*, 2025.
- [26] J. Jang, S. Ye, Z. Lin, J. Xiang, J. Bjorck, Y. Fang, F. Hu, S. Huang, K. Kundalia, Y.-C. Lin, L. Magne, A. Mandlekar, A. Narayan, Y. L. Tan, G. Wang, J. Wang, Q. Wang, Y. Xu, X. Zeng, K. Zheng, R. Zheng,

- M.-Y. Liu, L. Zettlemoyer, D. Fox, J. Kautz, S. Reed, Y. Zhu, and L. Fan, “DreamGen: Unlocking Generalization in Robot Learning through Video World Models,” in *Proceedings of the Conference on Robot Learning (CoRL)*, vol. 305, 2025.
- [27] F. Zhu, H. Wu, S. Guo, Y. Liu, C. Cheang, and T. Kong, “IRASim: A Fine-Grained World Model for Robot Manipulation,” in *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2025.
- [28] S. Nasiriany, A. Maddukuri, L. Zhang, A. Parikh, A. Lo, A. Joshi, A. Mandlekar, and Y. Zhu, “RoboCasa: Large-Scale Simulation of Everyday Tasks for Generalist Robots,” in *Proceedings of the Robotics: Science and Systems (RSS)*, 2024.
- [29] E. J. Hu, Y. Shen, P. Wallis, Z. Allen-Zhu, Y. Li, S. Wang, L. Wang, and W. Chen, “LoRA: Low-Rank Adaptation of Large Language Models,” in *Proceedings of the International Conference on Learning Representations (ICLR)*, 2022.
- [30] Y. Mu, T. Chen, Z. Chen, S. Peng, Z. Lan, Z. Gao, Z. Liang, Q. Yu, Y. Zou, M. Xu, L. Lin, Z. Xie, M. Ding, and P. Luo, “RoboTwin: Dual-Arm Robot Benchmark with Generative Digital Twins,” in *Proceedings of the Computer Vision and Pattern Recognition Conference (CVPR)*, 2025.
- [31] A. Mandlekar, D. Xu, J. Wong, S. Nasiriany, C. Wang, R. Kulkarni, L. Fei-Fei, S. Savarese, Y. Zhu, and R. Martín-Martín, “What Matters in Learning from Offline Human Demonstrations for Robot Manipulation,” in *Proceedings of the Conference on Robot Learning (CoRL)*, vol. 164, 2022.