CRAFT: Video Diffusion for Bimanual Robot Data Generation

Jason Chen, I-Chun Arthur Liu, Gaurav S. Sukhatme, Daniel Seita
University of Southern California

Fig. 1: **CRAFT generates visually diverse bimanual manipulation demonstrations.** A mosaic of synthetic training videos produced by our video diffusion framework, spanning seven axes of visual variation: lighting conditions, object pose, object color, background environment, camera viewpoint, wrist and third-person perspectives, and cross-embodiment transfer. Starting from a small set of real-world teleoperation demonstrations, CRAFT synthesizes large-scale, photorealistic and temporally coherent demonstration datasets paired with action labels for coordinated dual-arm manipulation.

***Abstract*—Bimanual robot learning from demonstrations is fundamentally limited by the cost and narrow visual diversity of real-world data, which constrains policy robustness across viewpoints, object configurations, and embodiments. We present Canny-guided Robot Data Generation using Video Diffusion Transformers (CRAFT), a video diffusion-based framework for scalable bimanual demonstration generation that synthesizes temporally coherent manipulation videos while producing action labels. By conditioning video diffusion on edge-based structural cues extracted from simulator-generated trajectories, CRAFT produces physically plausible trajectory variations and supports a unified augmentation pipeline spanning object pose changes, camera viewpoints, lighting and background variations, cross-embodiment transfer, and multi-view synthesis. We leverage a pre-trained video diffusion model to convert simulated videos, along with action labels from the simulation trajectories, into action-consistent demonstrations. Starting from only a few real-world demonstrations, CRAFT generates a large, visually diverse set of photorealistic training data, bypassing the need to replay demonstrations on the real robot (Sim2Real). Across simulated and real-world bimanual tasks, CRAFT improves success rates over existing augmentation strategies and straightforward data scaling, demonstrating that diffusion-based video generation can substantially expand demonstration diversity and improve generalization for dual-arm manipulation tasks. Our project website is available at: https://craftaug.github.io/.**

I. Introduction

Imitation learning coupled with large teleoperated datasets has enabled increasingly capable bimanual manipulation systems [1]–[3]. However, scaling these models to diverse embodiments, viewpoints, and task variations remains data-intensive, motivating scalable methods to expand demonstration diversity without additional real-world data [4]. While data augmentation has emerged as a promising strategy [5]–[7], existing works focus on a subset of augmentations, such as augmenting only third-person [5] or wrist-camera views [8, 9], or solely transferring demonstrations across robot embodiments [10]–[12], without combining these into a unified pipeline.

To address these gaps, we present a novel and *unified* data augmentation framework for visual imitation learning, called **C**anny-guided **R**obot D**a**ta Generation using Video Di**f**fusion **T**ransformers (CRAFT). CRAFT constructs a digital twin of a real-world setup to generate simulation trajectories, extracts Canny-edge control videos, and conditions a video diffusion model [13] on these edges alongside a real-world reference image and language instruction. Our key insight is that Canny-edge conditioning guides the diffusion model by preserving the most relevant structural contours while abstracting away low-level simulation details. This enables a unified augmentation pipeline spanning object pose, object color, background, lighting, camera viewpoints, wrist and third-person multi-view generation, and bimanual cross-embodiment transfer.

We evaluate CRAFT across three simulated and three real-world bimanual manipulation tasks using the RoboTwin [14] benchmark and a physical setup with two bimanual xArm7 and Franka arms. See Figure 1 for a visualization of the diverse synthetic training videos produced by CRAFT, spanning seven axes of visual variation. Using ACT [15] as our downstream policy, CRAFT consistently improves robustness and generalization over existing augmentation baselines across all seven augmentation techniques.

All authors are with the Thomas Lord Department of Computer Science at the University of Southern California, USA. Correspondence: jchen567@usc.edu.

The contributions of this paper include:

- CRAFT, a novel and unified method to utilize Canny edge images [16] as a control input to condition video generative models to generate high-quality and diverse robot videos.
- A new pipeline for bimanual cross-embodiment manipulation that can perform additional image augmentations compared to prior approaches.
- Simulation and real-world experiments demonstrating that policies trained on CRAFT-generated data significantly outperform baselines, with ablations quantifying the benefit of each augmentation technique.

II. Related Work

A. Video Generation For Robotics

Recent years have seen the emergence of powerful diffusion-based [17] video generative models that can generate high-fidelity video frames given conditioning inputs (e.g., text or images) [18, 19]. In this work, we use Wan 2.1 [13] but other video generative models are applicable if they can condition on modalities such as Canny edge [16] images. Some works use video prediction models to generate robot trajectories [20] or action-conditioned predictions with wrist and third-person views [21], while others reconstruct physically grounded world models for policy training [22]. In contrast, we use video diffusion to synthesize additional action-labeled demonstrations for imitation learning using a simulator, without learning a predictive world model. More relevant to our work, AnchorDream [23] conditions generation on rendered robot motion traces without explicit simulator rollouts, but the lack of a simulator limits augmentation diversity. CRAFT instead leverages a simulator and digital twin pipeline to produce physically plausible trajectory variations across diverse scene configurations, yielding more diverse, higher-fidelity demonstrations across bimanual and multi-view settings, though at the cost of requiring access to a simulator and object meshes.

B. Cross-Embodiment Learning

A growing body of work tackles data scarcity in robotics by editing or synthesizing images to reduce distribution shift when transferring to a target robot. Mirage [11] performs "cross-painting" by inpainting the target robot over a source robot at evaluation time, enabling zero-shot policy transfer. RoVi-Aug [12] extends this idea by leveraging diffusion models to generate images of a target robot to expand training data. Shadow [10], instead, aligns train and test distributions through deterministic and composite robot mask overlays, and avoids pixel-level inpainting. While effective for single-arm cross-embodiment data transfer, these methods have not been evaluated for bimanual manipulation. Moreover, they do not generate new action labels, in contrast to our proposed framework. Another direction in cross-embodiment learning trains a policy on mixed data from multiple robot embodiments [24]–[26], but this requires extensive data from each robot embodiment. Finally, other works in cross-embodiment learning study applications in orthogonal domains such as inverse reinforcement learning [27] and human-robot cross-embodiment [28, 29], which are beyond the scope of our work.

C. Data Augmentation for Imitation Learning

Data augmentation has emerged as a practical tool for scaling imitation learning without additional human demonstrations. Prior work uses generative models to alter visual context such as backgrounds or objects while keeping actions fixed [6, 7, 30]–[32], in contrast to state-based augmentation approaches [33]–[35]. CRAFT similarly adjusts visual context but additionally expands the action distribution and does not require high-fidelity scene reconstruction as in [36]. Recent work has also studied viewpoint augmentation, from third-person perspectives for single-arm [5] and bimanual [37] contexts, as well as from wrist-cameras [8, 9, 38]–[40], but none unify both views in a single framework which CRAFT supports. Finally, another direction explores both state and action data augmentation through Real2Sim2Real simulation rollouts [41, 42]. While our method also uses a digital twin, it does not require the last Sim2Real step, since we employ a video diffusion model to synthesize photorealistic images from simulator-generated trajectories. This allows us to expand trajectory diversity in a way that preserves coordination constraints and contact dynamics to facilitate data-efficient bimanual manipulation.

III. Problem Statement

Our focus is on scalable data generation for vision-based imitation learning in bimanual manipulation, where a policy π_θ with parameters θ is trained from expert demonstrations using third-person RGB, wrist-camera, or combined RGB image observations. We denote a camera image at time t as I_t, simulation-generated images as I_t^{s}, video-diffusion-synthesized images as I_t^{d}, and ground truth deployment images as I_t^{g}. At deployment, the policy receives I_t^{g} and produces actions $a_t = \pi_\theta(I_t^{\mathrm{g}})$, where $a_t = (a_t^{\mathrm{l}}, a_t^{\mathrm{r}})$ specifies target joint positions and gripper actuation for the left and right arms, respectively.

We assume access to a small set of M real-world teleoperation demonstrations $\mathscr{D}^{\mathrm{real}} = \{\tau_1^{\mathrm{real}}, \ldots, \tau_M^{\mathrm{real}}\}$ and a simulation environment generating source videos $\mathbf{V}^{\mathrm{s}}$ via a digital twin pipeline. Each demonstration is a sequence of ground truth image observations and corresponding actions:

$$\tau_i^{\mathrm{real}} = (I_1^{\mathrm{g}}, a_1^{\mathrm{l}}, a_1^{\mathrm{r}}, \ldots, I_T^{\mathrm{g}}, a_T^{\mathrm{l}}, a_T^{\mathrm{r}}), \tag{1}$$

for a demonstration of T timesteps. Our goal is to synthesize a large, visually diverse set of generated demonstrations $\mathscr{D}^{\mathrm{gen}}$, with $|\mathscr{D}^{\mathrm{gen}}| \gg |\mathscr{D}^{\mathrm{real}}|$, where each synthesized demonstration contains diffusion-synthesized observations I_t^{d} resembling real-world images, to train a policy on $\mathscr{D}^{\mathrm{real}} \cup \mathscr{D}^{\mathrm{gen}}$.

IV. Method: CRAFT

CRAFT leverages a video diffusion model to synthesize photorealistic and visually diverse training videos for bimanual manipulation. Given a simulation-generated source

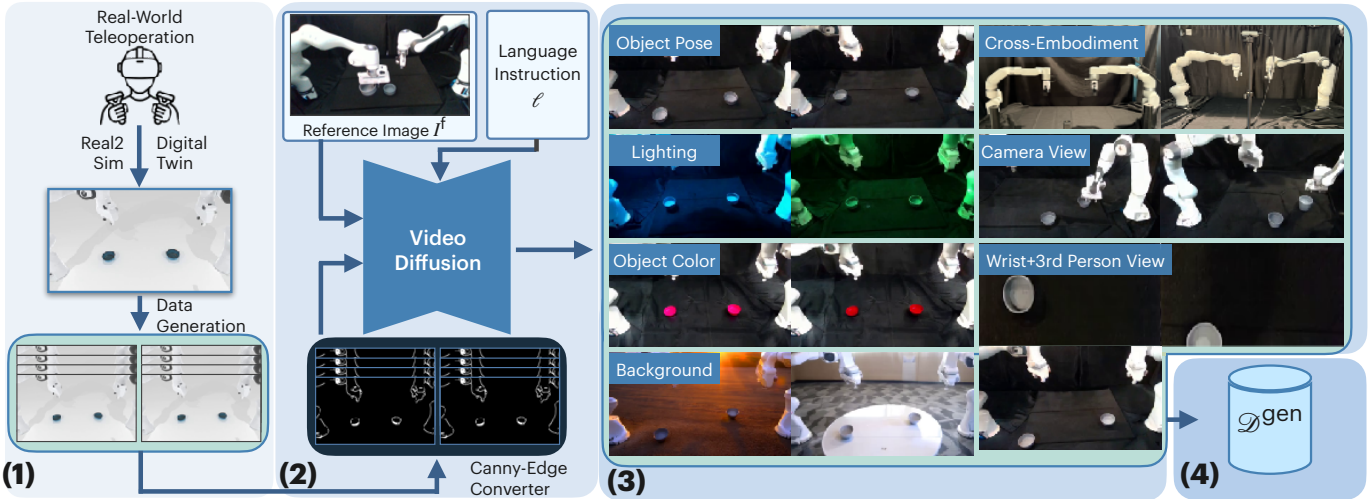


Fig. 2: **Method Overview. (1) Trajectory Expansion:** Real-world teleoperation data is first collected, and a digital twin pipeline transfers the objects and robot into simulation (Real2Sim). This simulation environment is then used for large-scale data generation. **(2) Video Generation:** The simulation trajectories are rendered into source videos and passed through a Canny-Edge Converter to extract structural edge representations, which are then combined with a real-world reference image and language instructions to condition a video diffusion model that synthesizes photorealistic video outputs. **(3) Augmented Dataset Construction:** The resulting generated videos support a wide range of visual variations, including object pose, lighting conditions, object color, background, cross-embodiment transfer, camera viewpoint, and combined wrist and third-person camera perspectives. **(4) Generated Dataset:** The synthesized videos are paired with action labels from the simulation trajectories, producing action-consistent demonstrations $\mathscr{D}^{\text{gen}}$ for downstream policy training.

video $\mathbf{V}^{\text{s}}$ produced from a digital twin pipeline, a real-world reference image I^{f}, and a language instruction ℓ, the model outputs a photorealistic target video $\mathbf{V}^{\text{d}} = \{I_1^{\text{d}}, \ldots, I_T^{\text{d}}\}$ that preserves robot motion structure while matching diverse real-world visual appearance. This is achieved through three stages: trajectory expansion (Section IV-A), video generation (Section IV-B), and augmented dataset construction for policy training (Section IV-C). CRAFT repeatedly applies this procedure to obtain $\mathscr{D}^{\text{gen}}$. Figure 2 provides an overview.

A. Trajectory Expansion

We construct a simulation counterpart $\mathscr{D}^{\text{sim}}$ from $\mathscr{D}^{\text{real}}$ using a digital twin pipeline, leveraging AprilTags [43] for object localization and known object meshes from RoboTwin [14], though any pipeline reconstructing object meshes and robot models in simulation is applicable [44]. Each real trajectory τ_i^{real} is replayed in simulation to generate a source video $\mathbf{V}^{\text{s}}$ and corresponding simulation trajectory τ_i^{sim}, resulting in a new simulation data of equal size as the original: $|\mathscr{D}^{\text{sim}}| = |\mathscr{D}^{\text{real}}|$.

To scale up data collection, we expand $\mathscr{D}^{\text{sim}}$ inspired by DexMimicGen [42]: each trajectory τ_i^{real} is decomposed into object-centric subtasks by annotating per-arm timestep boundaries, and a transformation operator $\mathscr{T}$ is applied to produce a new candidate trajectory $\mathscr{T}(\tau_i^{\text{real}})$ consistent with a novel sampled scene configuration. Each candidate is executed in simulation and validated for task success, retaining only successful trajectories to expand $\mathscr{D}^{\text{sim}}$. The validated trajectories are rendered into source videos $\mathbf{V}^{\text{s}}$, from which Canny-edge control videos $\mathbf{V}^{\text{c}}$ are extracted by filtering for salient structural edges (Figure 3, Tile 2), and fed into the video generation stage (Section IV-B) to synthesize $\mathscr{D}^{\text{gen}}$.

B. Video Generation

Diffusion-based video generative models learn to approximate a distribution over video sequences through iterative denoising. The model learns the conditional distribution:

$$p_\phi(\mathbf{V}^{\text{d}} \mid I^{\text{f}}, \mathbf{V}^{\text{c}}, \ell), \tag{2}$$

where $\mathbf{V}^{\text{c}}$ is the Canny-edge control video from Section IV-A, I^{ref} is the real-world reference image, ℓ is the language instruction, and ϕ denotes the model parameters. We use Wan2.1-Fun-Control [13] as our backbone, which supports Canny-edge, depth, and skeleton pose conditioning. We choose Canny-edge conditioning over depth or skeleton pose because skeleton pose captures only robot arm structure without encoding object information, while depth conditioning retains too much scene detail, reducing the model's flexibility to synthesize diverse visual appearances. Canny edges balance both by preserving salient structural features of robot arms and objects while discarding fine-grained details, giving the model freedom to vary visual appearance. By selectively varying I^{f} and ℓ while keeping $\mathbf{V}^{\text{c}}$ fixed, the model synthesizes diverse target videos $\mathbf{V}^{d}$ without altering robot motion structure. Depending on the desired variation, we modify I^{f}, ℓ, or both, and leverage an LLM to automatically generate semantic variants of ℓ (see Section IV-C).

C. Augmented Dataset Construction

We leverage Canny edges to guide demonstration augmentation across seven dimensions: object pose, lighting, object color, background, cross-embodiment, camera viewpoint, and wrist with third-person view generation. *CRAFT is flexible and modular where users can apply any subset of augmentation techniques and control the number of generated demonstrations.* Several augmentation techniques leverage LLMs to

Fig. 3: **Reference Image and Canny-edge Visualization.** Examples of reference images used to condition the video diffusion model for different augmentation techniques: **(1)** A standard reference image of the scene capturing gripper-object contact, used to condition the video diffusion model. **(2)** An example **Canny-edge frame** extracted from the simulation source video $\mathbf{V}^{\mathrm{c}}$, used as structural control input. **(3)** A lighting-modified reference image generated using Veo3 [45] under green ambient illumination. **(4)** An empty table reference image with no objects, used for object color generation. **(5)** A tiled reference image combining a third-person view (top left), left wrist (top right), and right wrist (bottom left), with the fourth tile left blank, supporting up to four simultaneous camera viewpoints. Reference images include top and bottom padding (not shown).

automatically generate diverse prompts and complete prompt lists are provided in the supplementary material.

1) Object Pose: To augment object poses, we introduce variations during trajectory expansion (Section IV-A). For each source trajectory τ_i^{real}, the simulator applies random translations and rotations to the target object's pose, sampled from a uniform distribution with ranges set based on the physically feasible workspace. We also find that using a reference image capturing gripper-object contact yields higher fidelity contact synthesis in generated videos (Figure 3, Tile 1).

2) Lighting: To generate diverse lighting conditions, we augment the reference image I^{f} by prompting an image generation model, Veo3 [45], to synthesize variants under different ambient illumination, such as blue or green lighting. Unlike simple color jitter or RGB channel manipulation, this approach preserves scene properties such as shadows and surface reflections (Figure 3, Tile 3). The augmented reference images are then used to condition the video diffusion model, producing target videos $\mathbf{V}^{\mathrm{d}}$ with photorealistic lighting variations while preserving the underlying robot motion structure.

3) Object Color: To generate diverse object colors, we use a reference image I^{f} of the empty table scene without any objects as illustrated in Figure 3 (Tile 4). Conditioning on a reference image that contains objects would anchor the generated scene to the object color present in the reference, limiting color diversity. Since the reference image contains no objects, the Canny-edge control video $\mathbf{V}^{\mathrm{c}}$ provides the object contours to inform the diffusion model of their location, while the language instruction ℓ specifies the desired color to guide the appearance of the synthesized objects. By modifying the language instruction to specify the desired object color, the video diffusion model synthesizes target videos $\mathbf{V}^{\mathrm{d}}$ with the specified object appearance while preserving the scene layout and robot motion structure. To avoid manual prompt editing, we prompt an LLM to generate a list of object colors, from which we sample randomly during dataset construction.

4) Background: To generate diverse backgrounds, we omit the reference image I^{f} from the video diffusion model, as conditioning on it anchors the generated scene to the original environment. Instead, we modify the instruction ℓ to describe the desired background. To scale background diversity without manual prompting, we leverage an LLM to automatically generate a large set of varied background descriptions, which are then used to condition the video diffusion model to produce target videos $\mathbf{V}^{\mathrm{d}}$ with diverse scene appearances.

5) Cross-Embodiment: To enable cross-embodiment transfer, we map demonstrations from a source robot to a target robot using forward and inverse kinematics, and replace images of the source robot with photorealistic images of the target robot. This allows us to directly use the transferred demonstrations as training data for the target robot, without requiring any additional real-world data collection. At each timestep t, we apply forward kinematics $\mathrm{FK}(\cdot)$ to extract the end-effector poses for the left and right arms:

$$p_t^{\mathrm{l}} = \mathrm{FK}(a_t^{\mathrm{l}}), \quad p_t^{\mathrm{r}} = \mathrm{FK}(a_t^{\mathrm{r}}), \tag{3}$$

where p_t^{l} and p_t^{r} denote the Cartesian end-effector positions for the left and right arms, respectively. We then apply inverse kinematics $\mathrm{IK}(\cdot)$ to compute the corresponding joint configurations for the target robot:

$$\hat{a}_t^{\mathrm{l}} = \mathrm{IK}(p_t^{\mathrm{l}}), \quad \hat{a}_t^{\mathrm{r}} = \mathrm{IK}(p_t^{\mathrm{r}}), \tag{4}$$

while preserving the gripper actuation from the source trajectory τ_i^{real}. The retargeted demonstrations are replayed in simulation to generate the Canny-edge control video $\mathbf{V}^{\mathrm{c}}$, which conditions the video diffusion model to synthesize target videos $\mathbf{V}^{\mathrm{d}} = \{I_1^{\mathrm{d}}, \ldots, I_T^{\mathrm{d}}\}$ of the new robot embodiment.

6) Camera Viewpoint: To generate diverse camera viewpoints, we place additional cameras inside the simulator and tile up to four simultaneous views into a single image. Formally, given $1 \leq N \leq 4$ camera views $\{I_t^{\mathrm{s},1}, \ldots, I_t^{\mathrm{s},N}\}$, we construct a tiled source image $I_t^{\mathrm{s,tile}} = \{I_t^{\mathrm{s},1}, \ldots, I_t^{\mathrm{s},N}\}$ from which the Canny-edge control video $\mathbf{V}^{\mathrm{c}}$ is extracted. The reference image I^{f} is similarly tiled to match, and both are fed into the video diffusion model to synthesize target videos $\mathbf{V}^{\mathrm{d}}$

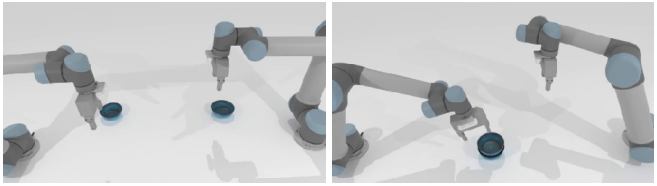


Fig. 4: **Simulation Environment.** The SB (see Section V) task adapted from RoboTwin [14] on the bimanual UR5 with WSG grippers, showing the initial state (left) and success state (right).

across multiple camera perspectives simultaneously. The video diffusion model automatically preserves the tiled structure in the generated output and each viewpoint remains spatially contained within its corresponding tile without going into adjacent tiles. Figure 3 (Tile 5) shows an example of the tiled reference image using wrist cameras; for camera view augmentation, the wrist camera tiles would instead be replaced with the desired third-person camera viewpoints.

7) Wrist and Third-Person View: Here, we follow the same tiling approach as camera viewpoint generation. Instead of tiling multiple third-person views, we tile the left wrist camera $I_t^{\text{s,l}}$, right wrist camera $I_t^{\text{s,r}}$, and a third-person (external) camera $I_t^{\text{s,ext}}$ into a single image $I_t^{\text{s,tile}} = \{I_t^{\text{s,l}}, I_t^{\text{s,r}}, I_t^{\text{s,ext}}, \emptyset\}$, leaving the fourth tile empty, from which the Canny-edge control video $\mathbf{V}^{\text{c}}$ is extracted. Tiling ensures spatial consistency across all viewpoints, and the reference image I^{f} is tiled accordingly as shown in Figure 3 (Tile 5) before being fed into the video diffusion model to synthesize target videos $\mathbf{V}^{\text{d}}$.

V. Simulation Experiments

Simulation experiments are conducted to evaluate the cross-embodiment capabilities of CRAFT, using the open-source RoboTwin [14] benchmark for bimanual manipulation. We modify the tasks to better align with the Action Chunking with Transformers (ACT) [15] policy and evaluate cross-embodiment transfer from a bimanual UR5 with WSG grippers (see Figure 6) to a bimanual Franka Panda across three tasks spanning different coordination strategies:

- **`Lift Pot (LP)`:** A coordinated bimanual task where both arms must simultaneously grasp and lift a pot.
- **`Place Cans in Plasticbox (PC)`:** A parallel task where both arms independently pick up cans and place them into a container.
- **`Stack Bowls (SB)`:** A sequential task where two bowls must be stacked on top of each other in order.

We evaluate the cross-embodiment variant of CRAFT, where the goal is to transfer demonstrations collected on a bimanual UR5 with WSG grippers (source robot) to train a policy for a bimanual Franka Panda (target robot) in simulation, without collecting any demonstrations on the target robot.

- **Collected Target:** Demonstrations collected directly on the target robot, serving as an upper-bound reference for cross-embodiment performance.
- **Shadow [10]:** A data editing approach that replaces robot observations with a composite segmentation mask of the source and target robots, which we extend to bimanual settings by masking both arms simultaneously.
- **CRAFT (Target):** Our method applied using Wan2.1-Fun-Control, without any additional generated demonstrations beyond the collected robot data.
- **CRAFT (Ours):** Our full pipeline, which expands the source robot dataset with 1000 generated demonstrations using object pose, lighting, and background augmentation before transferring to the target robot embodiment.

	Simulation (%)			Real-World (/ 20)		
Method	LP	PC	SB	LR	PC	SB
Collected Target	55.0%	69.0%	59.0%	5 / 20	2 / 20	3 / 20
Shadow [10]	2.0%	2.3%	6.0%	2 / 20	1 / 20	1 / 20
CRAFT (Target)	11.3%	6.0%	21.6%	4 / 20	1 / 20	2 / 20
CRAFT (Ours)	**82.6%**	**89.3%**	**86.0%**	**17 / 20**	**15 / 20**	**16 / 20**

TABLE I: **Cross-Embodiment Results.** Simulation success rates (%) and real-world successes out of 20 trials for cross-embodiment transfer. Simulation evaluates transfer from a bimanual UR5 to a bimanual Franka Panda on LP, PC, and SB (see Section V-A). Real-world evaluates transfer from a bimanual xArm7 to a bimanual Franka Panda on LR, PC, and SB (see Section VI-B). CRAFT (Target) denotes source-to-target transfer demos, with the same number of demos as Target. CRAFT (Ours) uses 1000 generated demos without collecting any target robot demos, in contrast to Collected Target which requires 100 demos on the target robot. All methods are trained and evaluated using ACT.

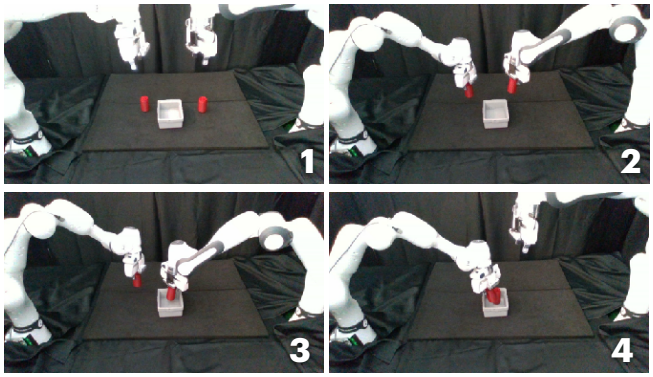
Method	Stack Bowls (SB)
Collected Demos (Upper Bound)	59.0%
CRAFT w/o Canny	10.3%
CRAFT w/ Canny	21.6%

TABLE II: **Ablation Study.** Success rates out of 150 demonstrations on `Stack Bowls` comparing video generation with and without Canny-edge control input. Collected Demos (Upper Bound) serves as the upper bound. No augmentation is performed in this ablation, as the objective is to isolate and evaluate the effect of synthesized image quality on ACT performance.

To ensure fair comparisons, all methods use identical training, validation, and test splits with consistent environment seeds, evaluated over 3 random seeds. Video generation is performed **zero-shot** using the Wan2.1-Fun-Control 1.3B model. ACT policy training and inference are conducted on a single NVIDIA RTX 4090 GPU. For small-scale video generation we use a single NVIDIA RTX 5090, while large-scale generation is distributed across 3 NVIDIA RTX 6000 GPUs. At inference time, input images are center-cropped, padded, and resized using OpenCV to 512×512 pixels before being passed to the video generation model.

A. Simulation Results

As shown in Table I, our Target variant already outperforms Shadow, which struggles on precision-demanding tasks as its segmentation mask occludes critical gripper-object contact points, degrading ACT performance. While the Target variant better preserves visibility, it alone does not yield substantial improvement. Scaling to 1000 generated demonstrations leads to substantially higher success rates: $11.3\% \rightarrow 82.6\%$, $6.0\% \rightarrow 89.3\%$, and $21.6\% \rightarrow 86.0\%$. Notably, CRAFT (Ours) surpasses the target demonstration baseline (target robot demos) despite never collecting target robot data, showing that diverse



Place Cans In Plasticbox

Fig. 5: **Real-world rollout** trained on ACT of the `PC` (see Section VI-B), with the bottom-right corner of each image showing the progression of the task execution. Images are labeled in order of task progression.

data generation spanning varied object poses is a scalable alternative to target robot data collection.

B. Ablation Studies

We examine whether converting simulation videos to Canny-edge representations improves generation quality over raw simulation images. We replace $\mathbf{V}^{\mathrm{c}}$ with $\mathbf{V}^{\mathrm{s}}$ as input, referring to this variant as **CRAFT w/o Canny**. Note that no additional data augmentation is applied in this ablation since we solely isolate the effect of Canny-edge conditioning on the quality of synthesized images, comparing generated demonstrations against the collected demonstration reference. As shown in Table II, CRAFT w/ Canny achieves nearly twice the success rate of CRAFT w/o Canny on `Stack Bowls`. This is because raw simulation images retain too much low-level detail, causing the diffusion model to struggle with salient structural features such as gripper-object contact, leading to degraded synthesis. Canny edges discard irrelevant details while preserving robot arm and object structure, giving the diffusion model clear guidance on what to generate.

VI. Real-World Experiments

A. Real-World Experiment Setup

We use a bimanual Franka Research 3 with GELLO [47], one or three Intel RealSense D435i cameras depending on whether wrist-camera observations are needed, an NVIDIA RTX 5090 for ACT training and inference, and **zero-shot** video generation via Wan2.1-Fun-Control 1.3B. AprilTags from the third-person camera estimate object positions and camera extrinsics, with object meshes sourced from RoboTwin, though other digital twin pipelines [44] can be substituted. We evaluate each policy's success rate over 20 trials per task across three tasks spanning a range of bimanual coordination strategies.

- **`Lift Roller (LR)`:** A coordinated task where both arms simultaneously grasp and lift a dough roller.
- **`Place Cans in Plasticbox (PC)`:** A parallel task where both arms independently pick up cans and place them into a container.
- **`Stack Bowls (SB)`:** A sequential task where two bowls must be stacked on top of each other in order.

B. Real-World Results

We evaluate CRAFT across seven augmentation techniques (see Section IV-C): object pose, lighting, object color, background, cross-embodiment, camera viewpoint, and wrist with third-person view generation. For each augmentation type, we compare policies trained on: (1) real-world demonstrations only (i.e., $\mathscr{D}^{\mathrm{real}}$), (2) demonstrations with object pose augmentation only (CRAFT Pose-Only), (3) real-world demonstrations augmented with an augmentation-specific baseline method, and (4) demonstrations generated via our full CRAFT pipeline. CRAFT Pose-Only is included to isolate the contribution of object pose variation from the remaining augmentation techniques.

For each augmentation type, we compare the same baselines and evaluate under test conditions that vary only along that specific dimension. For example, unseen lighting conditions for lighting and unseen backgrounds for background, while all other visual factors remain fixed. We provide further details on the test conditions for each augmentation techniques in the supplementary material. Due to task simplicity, `Lift Roller` uses fewer demonstrations than `Place Cans in Plasticbox` and `Stack Bowls` across all methods:

- **ACT w/o Aug:** 50 (`LR`) / 100 (`PC`, `SB`) real-world demonstrations collected under standard conditions trained on ACT [15].
- **CRAFT Pose-Only:** 100 (`LR`) / 200 (`PC`, `SB`) demonstrations with object pose augmentation only. Inspired by RoboSplat [36], we include this baseline to assess the standalone impact of varying object poses.
- **ACT with Baseline Aug (augmentation-specific):** 50 (`LR`) / 100 (`PC`, `SB`) demonstrations with an augmentation-specific method. The specific baseline used for each augmentation type is noted in the corresponding subsection.
- **CRAFT (Ours):** 1000 (`LR`, `PC`, `SB`) generated demonstrations and the original real-world demonstrations using our full augmentation pipeline.

1) Lighting: To evaluate lighting generalization, we test policy deployment under four distinct lighting conditions: blue, green, red, and yellow ambient illumination. We use **Color Jitter** as the augmentation-specific baseline, implemented by applying random per-channel scale offsets and Gaussian noise to each frame using OpenCV; we adopt this baseline as it was used in RoboSplat [36] for the same purpose of evaluating lighting robustness. As shown in Table III, policies trained only on collected demonstrations struggle significantly under unseen lighting conditions. While Color Jitter provides modest improvement, it fails to capture photorealistic lighting variations at evaluation time. CRAFT (Ours) achieves the highest success rates across all three tasks, demonstrating that photorealistic lighting augmentation via video diffusion is more effective than standard 2D color augmentation.

Method	Lighting			Background			Camera View			Object Color			Wrist + 3rd Person		
	LR	PC	SB	LR	PC	SB	LR	PC	SB	LR	PC	SB	LR	PC	SB
ACT w/o Aug	3 / 20	1 / 20	0 / 20	4 / 20	0 / 20	0 / 20	6 / 20	3 / 20	2 / 20	2 / 20	0 / 20	1 / 20	15 / 20	11 / 20	13 / 20
CRAFT Pose-Only	5 / 20	3 / 20	2 / 20	7 / 20	2 / 20	3 / 20	13 / 20	5 / 20	7 / 20	5 / 20	2 / 20	3 / 20	13 / 20	8 / 20	10 / 20
ACT w/ Baseline Aug	13 / 20	9 / 20	8 / 20	4 / 20	5 / 20	6 / 20	14 / 20	8 / 20	6 / 20	15 / 20	9 / 20	11 / 20	N/A†	N/A†	N/A†
CRAFT (Ours)	**17 / 20**	**14 / 20**	**12 / 20**	**18 / 20**	**15 / 20**	**10 / 20**	**19 / 20**	**18 / 20**	**18 / 20**	**18 / 20**	**18 / 20**	**17 / 20**	**20 / 20**	**19 / 20**	**20 / 20**

† No suitable baseline augmentation method exists for this augmentation type.

TABLE III: **Real-World Augmentation Results.** Success rates out of 20 for `LR`, `PC`, and `SB` (see Section VI-B) across five augmentation techniques. For each augmentation technique, all methods are evaluated under test conditions that vary only along that specific dimension (e.g., unseen lighting conditions for Lighting, unseen backgrounds for Background), while all other visual factors remain fixed. All CRAFT (Ours) columns use 1000 generated demonstrations combined with the real-world collected demonstrations (100 for `LR`, 200 for `PC`, and 150 for `SB`), where the number of collected demonstrations per task was determined based on the minimum required to achieve reasonable ACT policy performance. Baseline methods use the same number of collected demonstrations augmented with their respective augmentation technique. The "ACT w/ Baseline Aug" row refers to a different baseline method for each augmentation type, chosen as the most relevant prior work for a specific augmentation: Lighting (Color Jitter), Background (RoboEngine [7]), Camera View (VISTA [5]), and Object Color (SAM3 [46]). All methods are trained and evaluated using an ACT policy on the bimanual Franka.

2) Background: To evaluate background generalization, we test policy deployment across three distinct background scenarios. We use **RoboEngine** [7] as the augmentation-specific baseline, which segments objects and robot arms and applies 2D image inpainting to replace the background. As shown in Table III, CRAFT (Ours) achieves the highest success rates across all tasks. RoboEngine struggles in segmentation for certain objects, leading to frame-level inconsistencies and degraded gripper-object contact regions. Furthermore, its frame-by-frame processing cannot ensure temporal consistency or support multi-view generation, both of which CRAFT naturally handles through video diffusion.

3) Camera View: To evaluate camera view generalization, we test policy deployment across four camera perspectives. We use **Fine-Tuned VISTA** [5] as the augmentation-specific baseline, a diffusion-based novel view synthesis method that leverages ZeroNVS [48] to augment third-person viewpoints from a single third-person view, which we extend to the bimanual manipulation setting. As shown in Table III, CRAFT (Ours) substantially outperforms VISTA across all three tasks, with particularly large margins on `Lift Roller` and `Place Cans in Plasticbox`, demonstrating the advantage of video diffusion-based multi-view generation over single-view synthesis approaches.

4) Object Color: To evaluate object color generalization, all baselines are trained on demonstrations with red objects and evaluated on gray objects at deployment time. We use **SAM3** [46] as the augmentation-specific baseline, which segments objects of interest and applies color jitter to the segmented regions. A relevant prior work is ROSIE [32], an object editing method, but its code is not publicly available, making SAM3 the most practical alternative. Similar to RoboEngine, SAM3 struggles to consistently segment small or distant objects, leading to temporal inconsistencies across frames and degraded quality at gripper-object contact regions. As shown in Table III, CRAFT (Ours) achieves substantially higher success rates, demonstrating the advantage of photorealistic video diffusion-based object color augmentation over segmentation-based color editing.

5) Wrist + 3rd Person View: To evaluate wrist and third-person view generation, we test policy deployment using a left wrist camera, right wrist camera, and a third-person camera simultaneously. Unlike the other augmentation dimensions, there are no suitable public baselines for this setting at the time of writing. Instead, we treat **ACT w/o Aug** trained on 100 real wrist-camera demonstrations collected directly with the target camera configuration as a performance reference, representing what is achievable with real collected wrist-camera data. As shown in Table III, CRAFT Pose-Only with 100 generated demonstrations already approaches the performance of ACT w/o Aug without collecting any real wrist-camera data. Scaling to 1000 generated demonstrations with CRAFT (Ours) further improves performance, achieving perfect success rates of 20/20 on `Lift Roller` and `Stack Bowls`, demonstrating the effectiveness and scalability of CRAFT for multi-view camera generation.

6) Cross-Embodiment: To evaluate real-world cross-embodiment transfer, we collect demonstrations on a bimanual xArm7 and apply forward and inverse kinematics in MuJoCo simulation [49] to retarget the trajectories to the bimanual Franka Panda. The retargeted demonstrations are then replayed in the RoboTwin simulator to generate Canny-edge control videos for the video diffusion model. We use a bimanual xArm7 as the source robot rather than the bimanual UR5 used in simulation, as we do not have access to WSG grippers in our real-world setup. These experiments complement our simulation results (Section V-A) by validating cross-embodiment transfer on physical hardware with a different source robot. We compare against **Shadow** [10] as our baseline, following the same bimanual adaptation described in Section V-A for simulation experiments. Unlike Shadow, CRAFT preserves photorealistic gripper and object contact regions, which is critical for precise manipulation tasks. As shown on the right side of Table I, CRAFT (Ours) not only outperforms Shadow but also surpasses the collected demonstration baseline, demonstrating that our generated demonstrations can serve as a scalable and effective substitute for real target robot data collection.

VII. Conclusion

We present CRAFT, a scalable data generation pipeline for bimanual imitation learning that synthesizes photorealistic demonstrations across seven augmentation techniques via video diffusion conditioned on Canny-edge control videos, reference images, and language instructions. CRAFT consistently outperforms augmentation-specific baselines in simulation and the real world, demonstrating that scalable data generation

can substitute for costly real-world data collection. We hope CRAFT inspires further work in video generation for robot learning.

VIII. Acknowledgments

We thank Josh Shangguan, Kelly Wu, and Kyle Hatch for helpful discussions and writing feedback, the Physical Superintelligence Lab for lending equipment, and Weiduo Yuan and Tien Toan Nguyen for hardware support.

References

[1] K. Black, N. Brown, D. Driess, A. Esmail, M. Equi, C. Finn, N. Fusai, L. Groom, and et al., "π_0: A Vision-Language-Action Flow Model for General Robot Control," in Robotics: Science and Systems (RSS), 2024.

[2] P. Intelligence, K. Black, N. Brown, J. Darpinian, K. Dhabalia, D. Driess, A. Esmail, M. Equi, C. Finn, and et al., "$\pi_{0.5}$: a Vision-Language-Action Model with Open-World Generalization," in Conference on Robot Learning (CoRL), 2025.

[3] S. Liu, L. Wu, B. Li, H. Tan, H. Chen, Z. Wang, K. Xu, H. Su, and J. Zhu, "RDT-1B: a Diffusion Foundation Model for Bimanual Manipulation," in International Conference on Learning Representations (ICLR), 2025.

[4] K. Goldberg, "Good old-fashioned engineering can close the 100,000-year "data gap" in robotics," Science Robotics, vol. 10, no. 105, p. eaea7390, 2025. [Online]. Available: https://www.science.org/doi/abs/10.1126/scirobotics.aea7390

[5] S. Tian, B. Wulfe, K. Sargent, K. Liu, S. Zakharov, V. Guizilini, and J. Wu, "View-Invariant Policy Learning via Zero-Shot Novel View Synthesis," in Conference on Robot Learning (CoRL), 2024.

[6] X. Wang, K. Wu, Z. Zhao, H. Cao, Y. Zhao, Z. Xu, M. Li, S. Fan, D. Wu, Y. Zhang, N. Liu, Z. Che, and J. Tang, "RoboAug: One Annotation to Hundreds of Scenes via Region-Contrastive Data Augmentation for Robotic Manipulation," arXiv preprint arXiv:2602.14032, 2026.

[7] C. Yuan, S. Joshi, S. Zhu, H. Su, H. Zhao, and Y. Gao, "RoboEngine: Plug-and-Play Robot Data Augmentation with Semantic Robot Segmentation and Background Generation," in IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), 2025.

[8] I.-C. A. Liu, J. Chen, G. Sukhatme, and D. Seita, "D-CODA: Diffusion for Coordinated Dual-Arm Data Augmentation," in Conference on Robot Learning (CoRL), 2025.

[9] X. Zhang, M. Chang, P. Kumar, and S. Gupta, "Diffusion Meets DAgger: Supercharging Eye-in-hand Imitation Learning," in Robotics: Science and Systems (RSS), 2024.

[10] M. Lepert, R. Doshi, and J. Bohg, "Shadow: Leveraging Segmentation Masks for Cross-Embodiment Policy Transfer," in Conference on Robot Learning (CoRL), 2024.

[11] L. Y. Chen, K. Hari, K. Dharmarajan, C. Xu, Q. Vuong, and K. Goldberg, "Mirage: Cross-Embodiment Zero-Shot Policy Transfer with Cross-Painting," in Robotics: Science and Systems (RSS), 2024.

[12] L. Y. Chen, C. Xu, K. Dharmarajan, M. Z. Irshad, R. Cheng, K. Keutzer, M. Tomizuka, Q. Vuong, and K. Goldberg, "RoVi-Aug: Robot and Viewpoint Augmentation for Cross-Embodiment Robot Learning," in Conference on Robot Learning (CoRL), 2024.

[13] T. Wan, A. Wang, B. Ai, B. Wen, C. Mao, C. Xie, D. Chen, and et al., "Wan: Open and Advanced Large-Scale Video Generative Models," arXiv preprint arXiv:2503.20314, 2025.

[14] T. Chen, Z. Chen, B. Chen, Z. Cai, Y. Liu, Z. Li, Q. Liang, X. Lin, Y. Ge, Z. Gu et al., "Robotwin 2.0: A scalable data generator and benchmark with strong domain randomization for robust bimanual robotic manipulation," arXiv preprint arXiv:2506.18088, 2025.

[15] T. Z. Zhao, V. Kumar, S. Levine, and C. Finn, "Learning Fine-Grained Bimanual Manipulation with Low-Cost Hardware," in Robotics: Science and Systems (RSS), 2023.

[16] J. Canny, "A Computational Approach to Edge Detection," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. PAMI-8, no. 6, pp. 679–698, 1986.

[17] J. Ho, A. Jain, and P. Abbeel, "Denoising Diffusion Probabilistic Models," in Neural Information Processing Systems (NeurIPS), 2020.

[18] NVIDIA, "Cosmos World Foundation Model Platform for Physical AI," arXiv preprint arXiv:2501.03575, 2025.

[19] Z. Yang, J. Teng, W. Zheng, B. Xu, X. Gu, Y. Dong, J. Tang, and et al., "CogVideoX: Text-to-Video Diffusion Models with An Expert Transformer," in International Conference on Learning Representations (ICLR), 2026.

[20] J. Jang, S. Ye, Z. Lin, J. Xiang, D. Fox, J. Kautz, S. Reed, Y. Zhu, L. Fan, , and et al., "DreamGen: Unlocking Generalization in Robot Learning through Video World Models," arXiv preprint arXiv:2505.12705, 2025.

[21] Y. Guo, L. X. Shi, J. Chen, and C. Finn, "Ctrl-World: A Controllable Generative World Model for Robot Manipulation," in International Conference on Learning Representations (ICLR), 2026.

[22] J. Mao, S. He, H.-N. Wu, Y. You, S. Sun, Z. Wang, Y. Bao, H. Chen, L. Guibas, V. Guizilini, H. Zhou, and Y. Wang, "Robot Learning from a Physical World Model," in IEEE International Conference on Robotics and Automation (ICRA), 2026.

[23] J. Ye, R. Xue, B. V. Hoorick, P. Tokmakov, M. Z. Irshad, Y. Wang, and V. Guizilini, "AnchorDream: Repurposing Video Diffusion for Embodiment-Aware Robot Data Synthesis," in IEEE International Conference on Robotics and Automation (ICRA), 2026.

[24] J. Yang, C. Glossop, A. Bhorkar, D. Shah, Q. Vuong, C. Finn, D. Sadigh, and S. Levine, "Pushing the Limits of Cross-Embodiment Learning for Manipulation and Navigation," in Robotics: Science and Systems (RSS), 2024.

[25] R. Doshi, H. Walke, O. Mees, S. Dasari, and S. Levine, "Scaling Cross-Embodied Learning: One Policy for Manipulation, Navigation, Locomotion and Aviation," in Conference on Robot Learning (CoRL), 2024.

[26] E. Bauer, E. Nava, and R. K. Katzschmann, "Latent Action Diffusion for Cross-Embodiment Manipulation," arXiv preprint arXiv:2506.14608, 2025.

[27] K. Zakka, A. Zeng, P. Florence, J. Tompson, J. Bohg, and D. Dwibedi, "XIRL: Cross-embodiment Inverse Reinforcement Learning," in Conference on Robot Learning (CoRL), 2021.

[28] T. G. W. Lum, O. Y. Lee, C. K. Liu, and J. Bohg, "Crossing the Human-Robot Embodiment Gap with Sim-to-Real RL using One Human Demonstration," in Conference on Robot Learning (CoRL), 2025.

[29] P. Dan, K. Kedia, A. Chao, E. W. Duan, M. A. Pace, W.-C. Ma, and S. Choudhury, "X-Sim: Cross-Embodiment Learning via Real-to-Sim-to-Real," in Conference on Robot Learning (CoRL), 2025.

[30] Z. Chen, Z. Mandi, H. Bharadhwaj, M. Sharma, S. Song, A. Gupta, and V. Kumar, "Semantically Controllable Augmentations for Generalizable Robot Learning," in International Journal of Robotics Research (IJRR), 2024.

[31] H. Bharadhwaj, J. Vakil, M. Sharma, A. Gupta, S. Tulsiani, and V. Kumar, "RoboAgent: Generalization and Efficiency in Robot Manipulation via Semantic Augmentations and Action Chunking," in IEEE International Conference on Robotics and Automation (ICRA), 2024.

[32] T. Yu, T. Xiao, A. Stone, J. Tompson, A. Brohan, S. Wang, J. Singh, C. Tan, D. M, J. Peralta, B. Ichter, K. Hausman, and F. Xia, "Scaling Robot Learning with Semantically Imagined Experience," in Robotics: Science and Systems (RSS), 2023.

[33] P. Mitrano and D. Berenson, "Data Augmentation for Manipulation," in Robotics: Science and Systems (RSS), 2022.

[34] L. Ke, Y. Zhang, A. Deshpande, S. Srinivasa, and A. Gupta, "CCIL: Continuity-based Data Augmentation for Corrective Imitation Learning," in International Conference on Learning Representations (ICLR), 2024.

[35] M. Laskey, J. Lee, R. Fox, A. D. Dragan, and K. Goldberg, "DART: Noise Injection for Robust Imitation Learning," in Conference on Robot Learning (CoRL), 2017.

[36] S. Yang, W. Yu, J. Zeng, J. Lv, K. Ren, C. Lu, D. Lin, and J. Pang, "Novel demonstration generation with gaussian splatting enables robust one-shot manipulation," in Robotics: Science and Systems (RSS), 2025.

[37] J. Chen, I.-C. A. Liu, G. Sukhatme, and D. Seita, "ROPA: Synthetic Robot Pose Generation for RGB-D Bimanual Data Augmentation," in IEEE International Conference on Robotics and Automation (ICRA), 2026.

[38] A. Zhou, M. J. Kim, L. Wang, P. Florence, and C. Finn, "NeRF in the Palm of Your Hand: Corrective Augmentation for Robotics via Novel-View Synthesis," in IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2023.

[39] H. Ding, A. Duan, Z. Sun, D. Song, and Y. Nakamura, "Imagination at Inference: Synthesizing In-Hand Views for Robust Visuomotor Policy Inference," arXiv preprint arXiv:2509.15717, 2025.

[40] Y. Xie, Y. Wang, S. Zhao, C.-E. Wu, M. Tomizuka, J. Xie, and H.-S. Fang, "Multi-camera view scaling for data-efficient robot imitation learning," arXiv preprint arXiv:2604.00557, 2026.

[41] 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 Conference on Robot Learning (CoRL), 2023.

[42] Z. Jiang, Y. Xie, K. Lin, Z. Xu, W. Wan, A. Mandlekar, L. Fan, and Y. Zhu, "DexMimicGen: Automated Data Generation for Bimanual Dexterous Manipulation via Imitation Learning," in IEEE International Conference on Robotics and Automation (ICRA), 2025.

[43] J. Wang and E. Olson, "Apriltag 2: Efficient and robust fiducial detection," in IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), 2016.

[44] Z. Chen, A. Walsman, M. Memmel, K. Mo, A. Fang, K. Vemuri, A. Wu, D. Fox, and A. Gupta, "Urdformer: A pipeline for constructing articulated simulation environments from real-world images," in Robotics: Science and Systems (RSS), 2024.

[45] Google, "Veo 3," https://deepmind.google/models/veo/, 2025.

[46] N. Carion, L. Gustafson, Y.-T. Hu, N. Ravi, K. Saenko, P. Zhang, C. Feichtenhofer, and et al., "Sam 3: Segment anything with concepts," arXiv:2511.16719, 2025.

[47] P. Wu, Y. Shentu, Z. Yi, X. Lin, and P. Abbeel, "GELLO: A General, Low-Cost, and Intuitive Teleoperation Framework for Robot Manipulators," in IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), 2024.

[48] K. Sargent, Z. Li, T. Shah, C. Herrmann, H.-X. Yu, Y. Zhang, E. R. Chan, D. Lagun, L. Fei-Fei, D. Sun, and J. Wu, "ZeroNVS: Zero-Shot 360-Degree View Synthesis from a Single Image," in IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2024.

[49] E. Todorov, T. Erez, and Y. Tassa, "MuJoCo: A Physics Engine for Model-Based Control," in IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), 2012.
[50] M. Lepert, J. Fang, and J. Bohg, "Phantom: Training robots without robots using only human videos," in Conference on Robot Learning (CoRL), 2025.
[51] M. Moghani, M. Azizian, A. Garg, Y. Zhu, S. Huver, and A. Mandlekar, "Softmimicgen: A data generation system for scalable robot learning in deformable object manipulation," arXiv preprint arXiv:2603.25725, 2026.

APPENDIX

A. Simulation Task Details

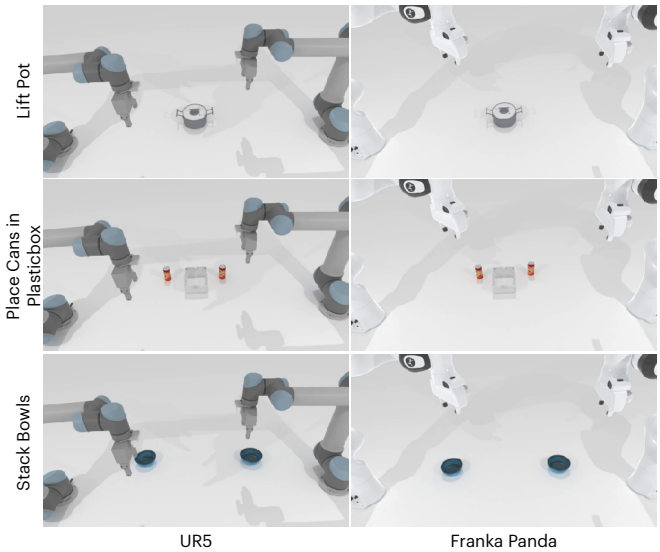


Fig. 6: **Simulation Environments.** Bimanual manipulation tasks adapted from RoboTwin [14], shown for the bimanual UR5 with WSG grippers (left) and the bimanual Franka Panda (right). Tasks from top to bottom: `Lift Pot (LP)`, `Place Cans in Plasticbox (PC)`, and `Stack Bowls (SB)`.

For simulation experiments, we use RoboTwin as our simulator. We modify the original task setup, which uses both wrist and third-person cameras for policy training, to use only the third-person camera for experiments that do not require wrist-camera observations, improving ACT policy performance.

For cross-embodiment transfer, we use MuJoCo [49] to perform FK and IK conversions from the xArm7 to the RoboTwin Franka Panda, as RoboTwin does not natively support the xArm7 at the time of writing. Specifically, we use `mash`, a MuJoCo-based retargeting tool developed internally in our lab, which is not publicly available at the time of writing.

As DexMimicGen [42] did not release their trajectory expansion code at the time of writing, we implemented the trajectory expansion procedure as closely as possible to the method described in their paper.

B. Task Selection Details

We select tasks that span a diverse range of bimanual coordination strategies, demonstrating that CRAFT generalizes across different task configurations. `Lift Roller` and `Lift Pot` are coordinated tasks requiring both arms to simultaneously grasp and lift an object with precise synchronization. `Place Cans in Plasticbox` is a parallel task where each arm operates independently to pick and place cans. `Stack Bowls` is a sequential task where subtasks must be completed in a specific order, requiring the policy to reason about task progression.

C. Real-World Task Details

The `Lift Roller` task uses a dough roller as the manipulation object. To initialize object poses in simulation, we use AprilTags [43] to estimate the position of each object relative to the robot arms from the third-person camera, and set the corresponding object poses in the RoboTwin simulator accordingly.

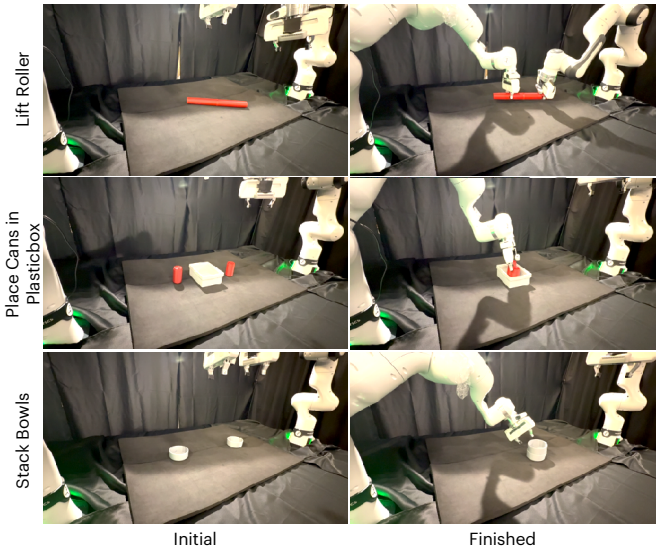


Fig. 7: **Real-World Environments.** Bimanual manipulation tasks adapted from RoboTwin [14], shown for the bimanual Franka Panda. Tasks from top to bottom: `Lift Roller (LR)`, `Place Cans in Plasticbox (PC)`, and `Stack Bowls (SB)`. The left images show the initial state and the right images show the final states.

We exclude the simulation `Lift Pot` task from real-world evaluation due to safety concerns, as the task requires the grippers to operate in close proximity to the workspace table.

D. Real-World Setup

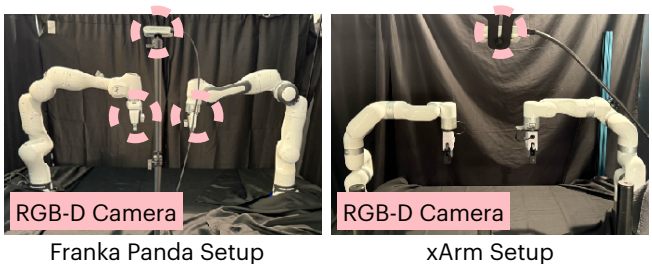


Fig. 8: **Real-World Setup.** Bimanual cross-embodiment setup showing the Franka Panda (left) and xArm7 (right). Pink dotted circles indicate the Intel RealSense D435i camera placements.

We leverage the GELLO [47] setup to collect demonstration data for both the Franka Panda and xArm7 setups.

E. Reference Image Comparison

We compare video generation quality when conditioning on a reference image that captures gripper-object contact versus one that does not. As shown in Figure 9, including gripper-object contact in the reference image leads to noticeably higher fidelity contact synthesis and fewer visual artifacts on the robot arms. This is because Wan2.1-Fun-Control uses the reference image as a strong appearance prior, by providing an image that explicitly shows the gripper in contact with the object,

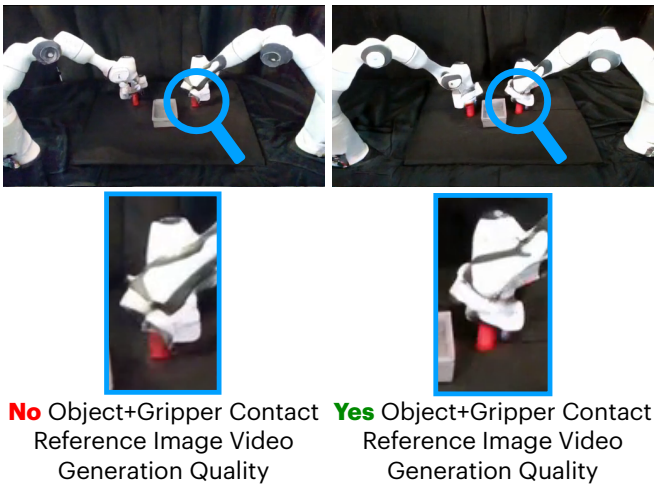


Fig. 9: **Video Generation Quality Comparison.** Generated video frames without (left) and with (right) a reference image capturing gripper-object contact. As seen in the blue bordered images, including gripper-object contact in the reference image leads to higher fidelity contact synthesis in the generated video.

the diffusion model can better infer the spatial relationship between the gripper and object, resulting in more realistic and geometrically consistent contact regions in the generated video.

F. Additional Baseline Implementation Details

Shadow [10] is not publicly available so we implemented the baseline using the open-source code from their follow-up work Phantom [50] adapting it as closely as possible to the original Shadow method.

G. Additional Real-World Experiment Details

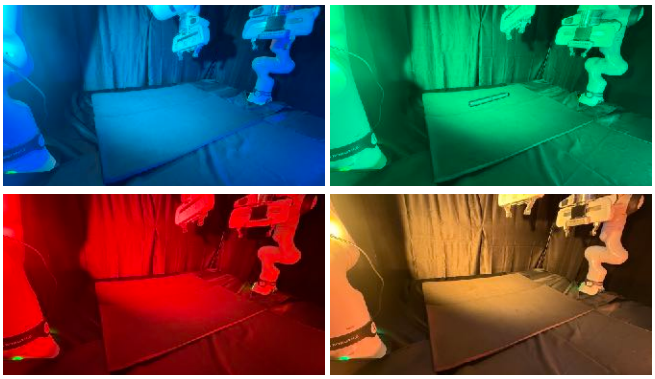


Fig. 10: **Lighting Environments.** Test lighting conditions used for evaluation: blue (top left), green (top right), red (bottom left), and yellow (bottom right) ambient illumination.

1) Lighting: For lighting, we leverage one NanLite Forza 60C and one NanLite PavoTube II 15C to display different colored lights in the scene. We show the different lighting conditions in Figure 10.

2) Background: For background generalization, we evaluate across three distinct background conditions: a flower fabric, a rose fabric, and an open lab environment with all curtains removed to simulate a cluttered real-world setting. We show the different background scenarios in Figure 11

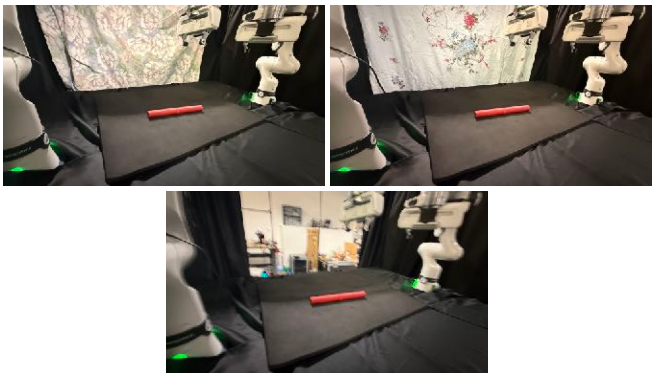


Fig. 11: **Background Environments.** Test background conditions used for evaluation: flower fabric (top left), rose fabric (top right), and no curtain (bottom left).

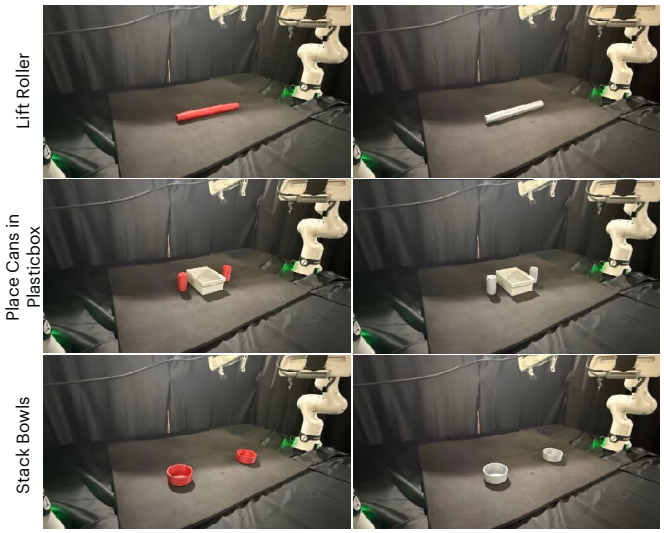


Fig. 12: **Object Color Conditions.** Object color variations used for evaluation across each task. From top to bottom: `Lift Roller (LR)`, `Place Cans in Plasticbox (PC)`, and `Stack Bowls (SB)`.

3) Object Color: For object color generalization, we evaluate across two object color variations. The objects were fabricated using a 3D printer, allowing us to vary color by swapping filament; we evaluate two colors to avoid unnecessary material waste. However, our method is not limited to only gray and red object colors. The two object color examples are shown in Figure 12.

H. Canny-Edge Filtering

Rather than using raw Canny-edge outputs, we apply two post-processing steps to produce cleaner and more informative control signals: (1) edge thickening to strengthen the structural control signal, and (2) edge connectivity to bridge nearby disconnected edges and reduce fragmented edge artifacts. These steps ensure that the Canny-edge control video provides clear and coherent structural guidance to the video diffusion model. We show an example of the non-filtered and filtered canny-edge comparison in Figure 13.

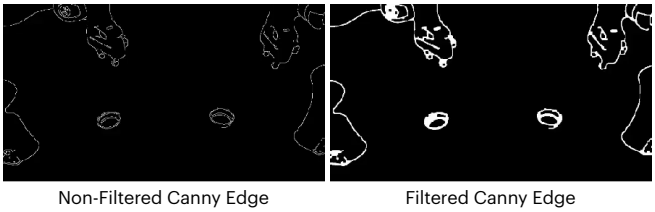


Fig. 13: **Canny-Edge Filtering Comparison.** Unfiltered (left) vs. filtered (right) Canny-edge representations, showing how edge thickening and connectivity post-processing produce cleaner structural control signals for the video diffusion model.

I. Limitations and Future Work Opportunities

1) Like all video generative model-based approaches, synthesized videos may contain visual artifacts or temporal inconsistencies that hinder downstream policy learning.
2) The third-person camera must be positioned close to the robot and objects of interest, as distant views produce noisy Canny-edge representations that degrade generation quality, particularly at gripper-object contact regions.
3) Although video generation is performed zero-shot, achieving high-quality results requires careful prompt engineering and informative reference images.
4) CRAFT's trajectory expansion procedure requires access to a simulator and object meshes to construct a digital cousin, similar to DexMimicGen [42]. While this is a shared assumption, it may limit applicability to tasks or objects that are difficult to simulate accurately.
5) CRAFT assumes tasks can be decomposed into object-centric subtasks for trajectory expansion.
6) CRAFT has not been tested on deformable objects however future work could leverage recent approaches such as SoftMimicGen [51] to extend it in this direction.