Distilling World Understanding from Video Generators for Robot Control

Anonymous authors
Paper under double-blind review

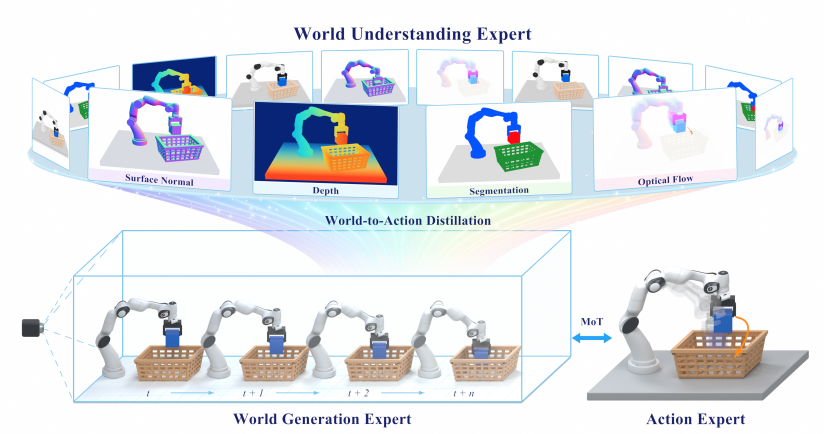

Figure 1: **Distilling multimodal world understanding into action.** KRUG develops a World Understanding Expert through multimodal instruction tuning, making implicit world knowledge explicit. World-to-Action Distillation transfers this knowledge into a student policy that couples an RGB video generator (World Generation Expert) with an Action Expert through joint attention for action prediction.

Abstract

Video generation models have shown promising results in robot control through future scene generation supported by their implicit understanding of the physical world. However, generating realistic textures does not guarantee an accurate representation of physical properties such as depth and motion, which are important for robot control. To leverage **K**nowledge for **R**obot control from the implicit **U**nderstanding in video **G**enerators, we propose **KRUG**, a two-stage framework based on generation as understanding. To activate this implicit world understanding, we apply multimodal instruction tuning to a pretrained video generator using a small amount of multimodal data, yielding a single World Understanding Expert that generates depth, surface normals, segmentation masks, or optical flow conditioned on modality-specific instructions. The challenge is to transfer this understanding to a student policy that uses RGB as its only visual input. We introduce World-to-Action Distillation (W2AD), which uses the expert as a frozen teacher and aligns features retrieved from the teacher and student using the same action queries to guide action prediction. For model training, we construct a multimodal embodied AI dataset by annotating RGB videos from nine sources, with 600,000 training clips and over 1,200 hours counted across available modalities. **KRUG** achieves strong performance on RoboTwin 2.0 and advances the **state of the art** on DexJoCo with an average success rate of 60.4%.

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

1 Introduction

Robotic manipulation can be formulated as learning a policy that maps visual observations and task instructions to executable actions. Success depends on how effectively the policy interprets the instruction and understands the physical state of the scene. Understanding therefore provides a foundation for linking perception and control. Two main design philosophies guide policy learning, one based on understanding current observations and the other on generating future scenes.

Vision-Language-Action (VLA) models such as RT-2 (Zitkovich et al., 2023), OpenVLA (Kim et al., 2024), and π_0 (Black et al., 2024) build on the visual and language understanding capabilities of pretrained vision-language models. Although these models benefit from broad semantic knowledge, adapting them to robot control still requires massive amounts of paired observation–instruction–action data, which are costly to collect. World Action Models (WAMs) such as VideoVLA (Shen et al., 2025), LingBot-VA (Li et al., 2026a), and DreamZero (Ye et al., 2026) use video generation to predict future scenes and guide action learning. However, video generative pretraining primarily focuses on visual fidelity rather than world understanding. A generated scene may look realistic yet lack depth awareness or precise object boundaries. Small pixel shifts in generated videos may have little effect on visual quality but can reduce the accuracy of action prediction. Recent findings from Vision Banana (Gabeur et al., 2026) and GenCeption (Wang et al., 2026c) suggest a way forward by showing that instruction-driven adaptation can elicit world knowledge already encoded in image and video generators.

We propose **KRUG** (**K**nowledge for **R**obot control from **U**nderstanding in video **G**enerators) to turn the implicit world understanding of video generators into direct supervision for robot control. Building on WAMs, we redefine the video model's role by developing it into an explicit generative world understanding model. This raises two challenges. The first is how to extract the physical understanding encoded in a video generator. The second is to transfer this understanding to an action model to improve action prediction.

As illustrated in Fig. 1, we first develop a **World Understanding Expert** through multimodal instruction tuning of a pretrained video generator to activate its implicit world understanding, then transfer its world knowledge to an RGB-based action model through World-to-Action Distillation (W2AD). For the first challenge, we start from the information required for robot manipulation. Depth and surface normals support 3D spatial understanding, while semantic segmentation identifies individual objects. Optical flow provides motion information. Inspired by Vision Banana (Gabeur et al., 2026), we convert these modalities into RGB-style videos with three channels to match the input format of the pretrained video VAE. We then fine-tune the generator with a small amount of multimodal data, allowing it to express its implicit world understanding through multimodal prediction.

For the second challenge, a straightforward approach is to jointly train separate video generators for different modalities with an Action Expert, but this adds parameters. Yet these modalities reflect related physical information. For example, depth and optical flow both reflect a robot arm moving away from a fixed camera. To help the Action Expert capture these relationships, we distill world knowledge from the World Understanding Expert into a student policy through W2AD.

The student consists of a **World Generation Expert** for future RGB video generation and an **Action Expert** for action prediction. The teacher receives depth, surface normals, semantic segmentation, and optical flow, while the student takes RGB observations during both training and inference. Distillation guides the student's representations toward the physical scene structure relevant to action prediction. This allows the student to learn from multimodal world knowledge while retaining its standard RGB-to-action interface.

To implement this design, we fine-tune the pretrained Wan2.2-5B model (Wan et al., 2025) in Stage 1 using 600 hours of multimodal data from nine datasets. In Stage 2, the student follows a Mixture-of-Transformers (MoT) architecture, with shared joint attention between the World Generation Expert and the Action Expert. During distillation, we use the same action tokens to query teacher and student features, then align the resulting representations to guide knowledge transfer toward action prediction.

To support training and expand multimodal data resources for embodied AI, we collect data from AgiBot (Bu et al., 2025) , CALVIN (Mees et al., 2022) , Dex-YCB (Chao et al., 2021) , DexJoCo (Wang et al., 2026a) , DROID (Khazatsky et al., 2024), Galaxea (Jiang et al., 2025) , HOI4D (Liu et al., 2022), LIBERO (Liu et al., 2023) , and RoboTwin (Chen et al., 2025), and generate depth, surface normal, segmentation, and optical flow annotations. The collection covers simulated and real-world robot manipulation and human hand–object interaction. Approximately 140,000 RGB clips yield around 600,000 training clips and over 1,200 hours when counted across available modalities.

Experiments on RoboTwin (Chen et al., 2025) and DexJoCo (Wang et al., 2026a) demonstrate the effectiveness of multimodal supervision for robotic manipulation, with KRUG setting a new **state of the art** on DexJoCo. The results further show that our distillation strategy extracts shared features across modalities that support action prediction.

Overall, our contributions can be summarized as follows:

- We propose KRUG, a framework built on generation as understanding, which develops a pretrained video generator into a World Understanding Expert through instruction tuning with a small amount of multimodal data.
- We propose World-to-Action Distillation (W2AD), which uses shared action-token queries to align teacher and student representations, transferring action-relevant multimodal knowledge while preserving RGB-only inference.
- We collect a multimodal embodied interaction dataset by collecting data from nine sources and generating multimodal annotations, yielding approximately 600,000 modality-specific training clips totaling over 1,200 hours across modalities.

2 Related Work

Vision–Language–Action Models. Vision–Language–Action (VLA) models adapt pretrained vision–language representations for robot control. RT-2 (Zitkovich et al., 2023) represents actions as text tokens to transfer web-scale knowledge, while OpenVLA (Kim et al., 2024) provides an open model trained on diverse robot demonstrations. $\pi_{0.5}$ (Intelligence et al., 2025) uses flow matching for continuous action chunks and co-training on heterogeneous data for open-world generalization. These models connect semantic understanding with action prediction without explicitly predicting future scene geometry and motion. Our work complements this approach by making such world knowledge explicit and transferring it into action learning.

World Action Models. World action models jointly learn future visual dynamics and robot actions. WorldVLA (Cen et al., 2025) unifies image and action generation within an autoregressive framework. LingBot-VA (Li et al., 2026a) couples future-frame prediction and action generation through an autoregressive diffusion model with a Mixture-of-Transformers architecture. Motus (Bi et al., 2026) integrates understanding, video generation, and action experts within a unified framework. Future RGB prediction provides temporal supervision, but does not explicitly express physical quantities or semantic labels. Building on these models, we train a shared video generator to express world understanding through multimodal predictions and distill this knowledge into an RGB-based action policy.

Knowledge Distillation. Cross-modal distillation (Gupta et al., 2016) transfers supervision between paired modalities by aligning intermediate representations. In embodied control, Learning by Cheating (Chen et al., 2020) distills a privileged teacher into a camera-based driving policy, while RialTo (Torne et al., 2024) distills a privileged state-based policy into a deployable point-cloud policy. Our W2AD transfers multimodal world understanding into an RGB-based action policy. Shared action queries retrieve task-relevant features from a frozen multimodal teacher and an RGB student, and aligning these features provides supervision tailored to action prediction.

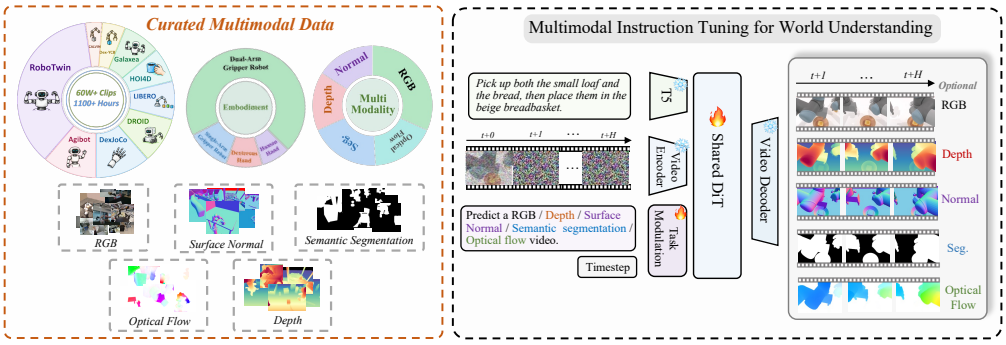

Figure 2: **Multimodal Instruction Tuning for World Understanding.** We curate multimodal manipulation data and train a shared video generator to express its implicit world knowledge through multimodal predictions.

3 Method

3.1 Problem Formulation and Framework Overview

At time t, the action policy receives RGB observations $\mathcal{O}_t = \{\mathbf{o}_t^v\}_{v=1}^V$ from V camera views, the robot state $\mathbf{s}_t$, and a language instruction c. It predicts an action chunk of length H_a as

$$\widehat{\mathbf{a}}_{t+1:t+H_a} = \pi_\Theta\left(\mathcal{O}_t, \mathbf{s}_t, c\right), \tag{1}$$

where Θ denotes the policy parameters.

As shown in Fig. 2 and Fig. 3, KRUG learns this policy in two stages. Stage 1 adapts a pretrained video generator into a **World Understanding Expert** $\mathcal{U}_\Phi$ through multimodal instruction tuning. Conditioned on the current RGB frame, the instruction c, and target modality ID, it predicts future RGB, depth, surface normal, segmentation, or optical flow clips. Stage 2 introduces World-to-Action Distillation (W2AD). We first tune an Action Expert $\mathcal{A}_\Omega$ to attend to relevant features in the World Understanding Expert, then distill the retrieved information into an RGB-based student policy.

3.2 Stage 1. Multimodal Instruction Tuning for World Understanding

Stage 1 aims to activate the implicit world understanding of a pretrained video generator through multimodal instruction tuning with a small amount of multimodal data. World understanding includes estimating physical distances, identifying objects, and predicting their future motion. Video generators implicitly capture this knowledge to produce coherent videos, but their RGB outputs do not explicitly express physical quantities. We make this understanding explicit through multimodal predictions, such as depth for physical distances, segmentation for object regions, and optical flow for motion. Our design converts these targets into RGB-style representations and learns their generation within a single model through unified task modulation.

To support multimodal instruction tuning, we construct a multimodal dataset by collecting existing human and robot manipulation videos and generating multimodal annotations. The collection comprises approximately 1,200 hours and 600,000 video clips. For Stage 1 instruction tuning, we use a subset of this collection totaling approximately 600 hours across modalities. This subset consists primarily of RGB videos, supplemented with a small proportion of clips from auxiliary modalities. Examples are shown in Fig. 2, with dataset and construction details in Appendix C.

Multimodal RGB-style Conversion. We use Wan2.2-TI2V-5B (Wan et al., 2025) as our pretrained video backbone. Its video VAE processes RGB videos, whereas depth, surface normals, segmentation, and optical flow represent quantities with different numerical ranges and meanings. We convert each target modality into a three-channel RGB-style video. Let $\mathcal{M} = \{\text{rgb}, \text{depth}, \text{normal}, \text{seg}, \text{flow}\}$ denote the set of target modalities. For each $m \in \mathcal{M}$, a reversible mapping R_m converts the target clip $\mathbf{y}^m$ into an RGB-style clip $\mathbf{x}^m = R_m(\mathbf{y}^m)$.

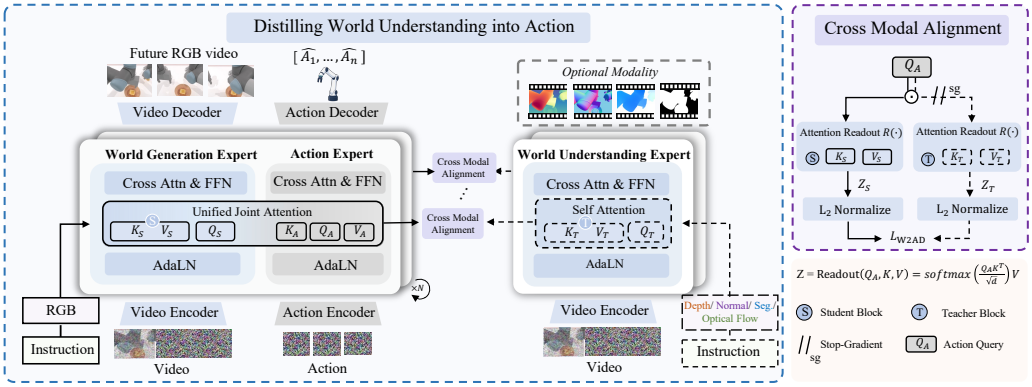

Figure 3: **Distilling World Understanding into Action.** Action queries guide feature alignment between the frozen multimodal teacher and RGB student, transferring world understanding into the action policy.

The inverse mapping recovers the corresponding physical quantities or labels. Details of the mapping are provided in Appendix D.

Model Architecture and Training Strategy. As illustrated in Fig. 2, we train a single Wan model to predict RGB-style clips across all modalities. Training separate generators may lead to inconsistent representations of these shared properties, whereas a shared model encourages consistent scene representations across modalities.

We process each camera view independently and sample a target modality m. The video VAE encodes the current RGB frame and the converted future clip of the target modality m. The current-frame latent remains clean, while the future latents are perturbed with Gaussian noise at timestep τ. A T5 text encoder maps instruction c to e_c, and Unified Task Modulation combines m and τ into $e_{\tau,m}$. These conditions guide the shared DiT, which processes the concatenated current and future latents. The shared video decoder produces RGB-style predictions without modality-specific heads. We apply the inverse mappings R_m^{-1} to these predictions to recover the corresponding physical quantities or semantic labels, which are visualized in Fig. 2. We optimize the DiT and task modulation using a flow matching objective on future latents, keeping the video VAE and T5 encoder frozen.

Unified Task Modulation. Prepending or appending a modality prompt (e.g., "Predict a depth video.") to the language instruction c mixes modality information into the instruction representation, which may make subsequent action learning more difficult. We associate each target modality m with a fixed text prompt and encode it separately using the same frozen T5 encoder. A learnable projector P_η maps the resulting text features to a modality embedding e_m, which is added to the timestep embedding e_τ, giving $e_{\tau,m} = e_\tau + e_m$.

3.3 Stage 2. Distilling World Understanding into Action

Stage 1 produces a World Understanding Expert that captures geometry, semantics, and motion through multimodal predictions. In Stage 2, we aim to transfer this knowledge into an action policy. We approach this transfer through two questions: *where the Action Expert should attend to access relevant world knowledge* and, further, *what information it should extract for action prediction*. In this section, we propose a World-to-Action Distillation (W2AD) strategy that addresses these two questions in sequence, enabling effective knowledge transfer for precise action prediction.

Distillation Framework. The action policy uses RGB as its only visual input during deployment. We therefore distill multimodal world knowledge into the policy, turning the questions above into *where to distill from* and *what to distill*. As shown in Fig. 3, we adopt a Mixture-of-Transformers (MoT) architecture, in which video and action experts interact through joint attention across stacked transformer blocks. We first tune the Action Expert with the frozen World Understanding Expert, then use this expert as the teacher to distill the retrieved knowledge into an RGB student initialized from the same video expert.

Where to distill from. We first perform *action-query tuning*, coupling the frozen World Understanding Expert $\mathcal{U}_\Phi$ with a trainable Action Expert $\mathcal{A}_\Omega$ through joint attention. Only Ω is optimized using video and action losses across all five modalities. Different modalities provide complementary knowledge about the same scene. The video loss updates the Action Expert through joint attention, encouraging it to attend to information needed for multi-modal prediction. Training across modalities helps the Action Expert access complementary cues about the same scene. The learned action queries then guide subsequent distillation.

What to distill. After action-query tuning, we retain $\mathcal{U}_\Phi$ as the frozen teacher and copy its weights to initialize the trainable **World Generation Expert** $\mathcal{G}_\Psi$ as an RGB student for distillation. Together with the tuned Action Expert, $\mathcal{G}_\Psi$ forms the policy π_Θ through the same MoT architecture. The teacher processes future-video latents from auxiliary modalities $\mathcal{M}_{\text{aux}} = \mathcal{M} \setminus \{\text{rgb}\}$, while the student processes paired RGB latents. Both branches share the current RGB observations, instruction, timestep, and noise.

Let $\mathcal{I}$ denote the set of selected transformer layers. At each $\ell \in \mathcal{I}$, we use the Action Expert's queries Q_A^ℓ to retrieve video features from the student and teacher:

$$Z_S^\ell = \text{Attn}(Q_A^\ell, K_S^\ell, V_S^\ell), \qquad Z_T^\ell = \text{Attn}(\text{sg}(Q_A^\ell), K_T^\ell, V_T^\ell), \tag{2}$$

where (K_S^ℓ, V_S^ℓ) and (K_T^ℓ, V_T^ℓ) are the student and teacher video keys and values, respectively. The stop-gradient operator sg blocks gradients through the teacher readout, preventing query updates that reduce alignment error by adjusting the teacher target. The queries receive gradients from both the action loss and the student-side distillation.

As illustrated in the right panel of Fig. 3, we transfer auxiliary-modality information by aligning the normalized teacher and student features:

$$\mathcal{L}_{\text{W2AD}} = \mathbb{E}\left[\frac{1}{|\mathcal{I}|}\sum_{\ell\in\mathcal{I}} \left\|\mathcal{N}(Z_S^\ell) - \text{sg}\left(\mathcal{N}(Z_T^\ell)\right)\right\|_2^2\right], \tag{3}$$

where $\mathcal{N}$ denotes ℓ_2 normalization, and the squared difference is averaged over action tokens and attention heads. This objective supervises the student's visual representations through information selected for action prediction.

Joint Optimization. During distillation, we jointly optimize the World Generation Expert and Action Expert while keeping the World Understanding Expert frozen:

$$\mathcal{L}_{\text{S2}} = \mathcal{L}_{\text{act}} + \lambda_{\text{rgb}}\mathcal{L}_{\text{RGB}} + \lambda_{\text{dist}}\mathcal{L}_{\text{W2AD}}, \tag{4}$$

where $\mathcal{L}_{\text{act}}$ and $\mathcal{L}_{\text{RGB}}$ are the action and RGB video flow matching losses, respectively, and λ_{rgb} and λ_{dist} weight the video and distillation losses.

4 Experiment

4.1 Experimental Setup

Implementation Details. Our model is built on Wan2.2-TI2V-5B (Wan et al., 2025) and trained on 120 × H200 GPUs. We use a learning rate of 5×10^{-5} for both Stage 1 training and the Stage 2 Action Expert, and 1×10^{-5} for the RGB student video expert during distillation. We use the AdamW (Loshchilov & Hutter, 2019) optimizer with $\beta_1 = 0.9$, $\beta_2 = 0.95$, and a weight decay of 0.01. At inference time, we deploy only the RGB student and Action Expert, with RGB observations as the only visual input.

Datasets and Platforms. We curate approximately 600,000 multimodal video clips totaling 1,200 hours, covering simulated and real-world manipulation and human–object interaction. We generated missing annotations using Qwen2-VL (Wang et al., 2024) for task descriptions, DPFlow (Morimitsu et al., 2025) for optical flow, MoGe-2 (Wang et al., 2026d) for depth and normals, and LangSAM (Liu et al., 2024; Ravi et al., 2025) for segmentation. Construction details are provided in Appendix C. We train and evaluate policies on three platforms, as described below.

RoboTwin (Chen et al., 2025). We replay official training trajectories to obtain synchronized multimodal data. We jointly train on 27,500 trajectories across 50 tasks from the Clean and Random configurations and evaluate under both settings.



Figure 4: **Qualitative results of KRUG.** We present rollouts on the real-world Galaxea R1 Lite robot, DexJoCo, and RoboTwin.

Table 1: Evaluation on RoboTwin 2.0 simulation tasks under clean and randomized settings.

Simulation Task	$\pi_{0.5}$		X-VLA		Motus		LingBot-VA		Ours	
	Clean	Rand.	Clean	Rand.	Clean	Rand.	Clean	Rand.	Clean	Rand.
adjust_bottle	**100**	**99**	**100**	**99**	89	93	90	94	98	**99**
... (50 tasks)										
place fan	87	85	80	75	91	87	**99**	93	**99**	**97**
stamp seal	79	55	76	82	93	92	96	97	**100**	**98**
Average (%)	82.74	76.76	72.80	72.84	88.66	87.02	**92.90**	91.50	92.76	**91.82**

DexJoCo (Wang et al., 2026a). We similarly replay 1,100 official demonstrations spanning six single-arm and five bimanual tasks, following the official multi-task training setting.

Real-World Robot. We collect demonstrations for three manipulation tasks using a Galaxea R1 Lite robot (Jiang et al., 2025) and evaluate the trained policies on the physical platform.

4.2 Evaluation in Simulation

In this section, we evaluate KRUG through quantitative comparisons and qualitative rollouts on RoboTwin and DexJoCo. As shown in Table 1, KRUG achieves success rates of 92.76% and 91.82% on RoboTwin 2.0 under the Clean and Random settings, respectively, performing comparably to LingBot-VA (Li et al., 2026a) on Clean and achieving the best average performance on Random. KRUG also substantially outperforms X-VLA (Zheng et al., 2026) under both settings. Complete quantitative results on RoboTwin 2.0 are provided in Appendix E.4. As shown in Table 2, KRUG achieves an average success rate of 60.4% on DexJoCo, outperforming DECO (Li et al., 2026b) (56.4%) and $\pi_{0.5}$ (Intelligence et al., 2025) (45.5%), and ranks first on 7 of 11 tasks. Fig. 4 presents qualitative rollouts on both simulation benchmarks, illustrating successful task execution.

4.3 Real-World Post-Training

Given the difficulty of collecting high-quality multimodal data on real robots, we support post-training without auxiliary multimodal data. Using RGB observations, robot states, instructions, and actions, we jointly train the pretrained World Understanding Expert from Stage 1 and an Action Expert with $\mathcal{L}_{\text{post}}^{\text{RGB}} = \mathcal{L}_{\text{RGB}} + \mathcal{L}_{\text{act}}$. Through joint attention, the Action Expert uses pretrained world knowledge for action prediction. We collect demonstrations for three tasks using a Galaxea R1 Lite robot (Jiang et al., 2025), then train and evaluate a separate policy for each. Benefiting from the pretrained world understanding in Stage 1, our method supports real-world post-training without auxiliary annotations. Successful rollouts are shown in Fig. 4, with additional examples in Appendix E.3.

4.4 World Understanding Evaluation

We evaluate the Stage 1 World Understanding Expert's depth, surface-normal, and segmentation predictions on RoboTwin and DexJoCo using simulator-provided ground truth. We exclude optical flow because its supervision uses DPFlow (Morimitsu et al., 2025) estimates. For each dataset, we sample 100 observations at time t and predict the corresponding modalities at $t+1$, while baselines estimate them directly from the ground-truth RGB frame at $t+1$. We report mean angular error (mAE) for surface normals, absolute relative error (AbsRel) for metric depth, and mean intersection over union (mIoU) averaged over the foreground and background segmentations.

Table 2: **Multi-task performance on DexJoCo.** Success rates (%) under the *rand-obj* configuration. The suffix /B denotes bimanual tasks. DP-T and $\pi_{0.5}$ results are from DexJoCo; DECO results are from its official release. Bold marks the best result in each row. DP-T's reported average is 33.2% (mean of displayed values: 28.2%).

Task	DP-T	$\pi_{0.5}$	DECO	Ours
Hammer Nail	58.7	**86.7**	68.0	80.0
Click Mouse	38.7	80.7	**90.0**	80.0
Pick Bucket	55.3	83.3	80.0	**90.0**
Pinch Tongs	6.0	45.3	46.0	**76.0**
Fold Glasses	11.3	42.0	50.0	**72.0**
Water Plant	60.0	84.0	**88.0**	76.0
Unlock iPad/B	0.0	0.7	40.0	**42.0**
Hanoi/B	8.0	6.0	18.0	**22.0**
Assembly/B	1.3	3.3	0.0	**8.0**
Microwave/B	42.7	39.3	64.0	**78.0**
Photograph/B	28.0	29.3	**76.0**	40.0
Avg.	33.2	45.5	56.4	**60.4**

Table 3: Quantitative results of world understanding on RoboTwin and DexJoCo. ↑ indicates that higher is better, and ↓ indicates that lower is better. "–" indicates that the method does not support prediction of the corresponding modality.

Method	**Normal** (mAE ↓)		**Depth** (AbsRel ↓)		**Seg.** (mIoU ↑)	
	RoboTwin	DexJoCo	RoboTwin	DexJoCo	RoboTwin	DexJoCo
MoGe-2	**10.29**	**12.94**	2.3178	0.970	–	–
VGGT-Ω	–	–	2.0270	1.4994	–	–
LangSAM	–	–	–	–	0.7861	**0.8284**
Ours	13.15	16.02	**0.3026**	**0.5947**	**0.9335**	0.8104

As shown in Table 3, our model achieves substantially lower depth error than VGGT-Ω (Wang et al., 2026b). Compared with LangSAM (Liu et al., 2024; Ravi et al., 2025), our model achieves better segmentation performance on RoboTwin and comparable performance on DexJoCo. MoGe-2 (Wang et al., 2026d) obtains lower surface-normal error with an additional camera-intrinsics estimation step.

4.5 Common Attention Regions.

Visualization of Common Attention Regions. Different modalities of the same scene encode shared physical information. We therefore examine whether the Action Expert attends to similar locations across modalities. After action-query tuning, we identify common attention regions based on action-token attention to future-video positions across all five modalities. The top row of Fig. 5 shows similar high-attention regions across all five modalities. Attention maps are overlaid on the model's direct RGB-style predictions for each modality. We refer to regions that receive high attention across modalities as *common attention regions*. The matrix at the bottom left further quantifies this cross-modal consistency, supporting the observed consistency in where the Action Expert attends across modalities. Experimental details and additional examples are provided in Appendix E.2.

Importance of Common Attention Regions. After distillation, we retain or mask the top 10% of positions by shared attention score in the RGB student's action-to-video attention. Normalized action retention measures agreement with the full model's outputs. Retaining these positions yields 81.2% retention, compared with 54.9% and 11.8% for equally sized randomly selected and lowest-scoring sets, respectively. Masking them yields 84.4%, compared with 98.5% when masking an equal number of random positions. These results support the relevance of common attention regions to the final RGB action policy.

4.6 Ablation Study

We conduct ablations on multimodal instruction tuning, action-query tuning, and distillation on RoboTwin 2.0. As shown in Table 4, the full model (ID 5) achieves the highest

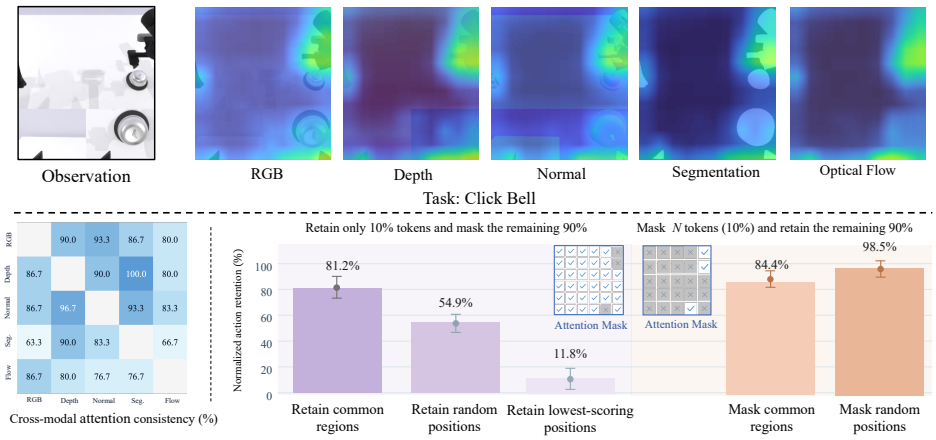

Figure 5: Cross-modal attention maps (top), attention consistency (bottom left), and normalized action retention when retaining or masking selected positions (bottom right).

Table 4: Ablation study of different training components on RoboTwin 2.0. Success rates are reported under the Clean and Random settings.

ID	Stage 1	Action-Query Tuning	Distillation	Success Rate (%) ↑	
				Clean	Random
1	✗	✓	✓	76.35	74.31
2	✗	✗	✗	83.74	82.90
3	✓	✗	✓	85.62	84.58
4	✓	✓	✗	87.23	86.48
5	✓	✓	✓	**92.76**	**91.82**

success rates of 92.76% and 91.82% under Clean and Random, respectively. ID 2 jointly trains the original video generator and an Action Expert using RGB video and action losses.

Multimodal Instruction Tuning. ID 1 skips Stage 1 and tunes the Action Expert with the frozen original video generator on multimodal targets before distillation. Its success rates drop to 76.35% and 74.31%, below those of ID 2. Without prior multimodal adaptation, the frozen generator has not learned to express world knowledge through auxiliary modalities. Updating only the Action Expert may therefore be insufficient to identify useful world information and may even encourage attention to regions irrelevant to action prediction.

Action-Query Tuning. Skipping action-query tuning (ID 3) reduces the success rates to 85.62% and 84.58%. This step trains the Action Expert to attend to useful regions across RGB and auxiliary modalities. The learned queries then guide feature retrieval from the RGB student and multimodal teacher during distillation. Without this step, alignment may focus on less relevant information. The larger drop compared with ID 4 highlights the importance of learning where to attend in this setting.

Distillation. Removing teacher supervision (ID 4) leaves the student trained only with RGB video and action objectives, yielding success rates of 87.23% and 86.48%. The student video expert already contains world knowledge acquired during Stage 1, and the Action Expert has learned where to attend through action-query tuning. The improvement of ID 4 over ID 2 also supports the feasibility of RGB post-training without auxiliary supervision. Explicit multimodal alignment provides further gains by transferring the teacher's world understanding into the visual features used for action prediction.

5 Conclusion

Video generators encode implicit world knowledge, yet high-fidelity video predictions alone do not ensure precise robot control. KRUG makes this knowledge explicit by adapting a pretrained video generator into a World Understanding Expert through multimodal instruction tuning. World-to-Action Distillation then transfers action-relevant knowledge into an RGB-based policy by learning *where to distill from* and *what to distill*. Experiments on RoboTwin, DexJoCo, and real-world manipulation tasks demonstrate the effectiveness of KRUG, highlighting the value of explicit world understanding for action prediction.

6 Reproducibility Statement

To ensure the reproducibility of the results presented in this paper, we have made comprehensive efforts to provide necessary information and resources across the main content and appendix. The appendix includes implementation details, analyses of multimodal representations and attention, complete per-task results on RoboTwin 2.0, and additional qualitative results. We encourage readers to explore these resources for a deeper understanding of our method. Moreover, we provide our code and rollout videos from both simulation and real-world experiments in the supplementary material.

7 Ethics Statement

The data used in this study include publicly available datasets, multimodal annotations generated from these datasets, and robot demonstrations collected by the authors. This research does not involve any human subjects, human experiments, or collection of private or sensitive information from individuals. No potential ethical risks (such as bias, discrimination, harm to individuals or groups, or misuse of research outcomes) are associated with our study. Throughout the entire research process, we have strictly adhered to the principles outlined in the ICLR Code of Ethics, ensuring full compliance with academic ethics and social responsibility.

References

Hongzhe Bi, Hengkai Tan, Shenghao Xie, Zeyuan Wang, Shuhe Huang, Haitian Liu, Ruowen Zhao, Yao Feng, Chendong Xiang, Yinze Rong, et al. Motus: A Unified Latent Action World Model. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 35101–35113, 2026.

Kevin Black, Noah Brown, Danny Driess, Adnan Esmail, Michael Equi, Chelsea Finn, Niccolo Fusai, Lachy Groom, Karol Hausman, Brian Ichter, et al. π_0: A Vision-Language-Action Flow Model for General Robot Control, 2024.

Qingwen Bu, Jisong Cai, Li Chen, Xiuqi Cui, Yan Ding, Siyuan Feng, Shenyuan Gao, Xindong He, Xuan Hu, Xu Huang, et al. AgiBot World Colosseo: A Large-Scale Manipulation Platform for Scalable and Intelligent Embodied Systems, 2025.

Jun Cen, Chaohui Yu, Hangjie Yuan, Yuming Jiang, Siteng Huang, Jiayan Guo, Xin Li, Yibing Song, Hao Luo, Fan Wang, et al. WorldVLA: Towards Autoregressive Action World Model, 2025.

Yu-Wei Chao, Wei Yang, Yu Xiang, Pavlo Molchanov, Ankur Handa, Jonathan Tremblay, Yashraj S. Narang, Karl Van Wyk, Umar Iqbal, Stan Birchfield, Jan Kautz, and Dieter Fox. DexYCB: A Benchmark for Capturing Hand Grasping of Objects. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2021*, 2021.

Dian Chen, Brady Zhou, Vladlen Koltun, and Philipp Krähenbühl. Learning by Cheating. In *Conference on robot learning*, pp. 66–75. PMLR, 2020.

Tianxing Chen, Zanxin Chen, Baijun Chen, Zijian Cai, Yibin Liu, Zixuan Li, Qiwei Liang, Xianliang Lin, Yiheng Ge, Zhenyu 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.

Valentin Gabeur, Shangbang Long, Songyou Peng, Paul Voigtlaender, Shuyang Sun, Yanan Bao, Karen Truong, Zhicheng Wang, Wenlei Zhou, Jonathan T Barron, et al. Image Generators Are Generalist Vision Learners, 2026.

Saurabh Gupta, Judy Hoffman, and Jitendra Malik. Cross Modal Distillation for Supervision Transfer. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2827–2836, 2016.

Physical Intelligence, Kevin Black, Noah Brown, James Darpinian, Karan Dhabalia, Danny Driess, Adnan Esmail, Michael Equi, Chelsea Finn, Niccolo Fusai, Manuel Y. Galliker, Dibya Ghosh, Lachy Groom, Karol Hausman, Brian Ichter, Szymon Jakubczak, Tim Jones, Liyiming Ke, Devin LeBlanc, Sergey Levine, Adrian Li-Bell, Mohith Mothukuri, Suraj Nair, Karl Pertsch, Allen Z. Ren, Lucy Xiaoyang Shi, Laura Smith, Jost Tobias Springenberg, Kyle Stachowicz, James Tanner, Quan Vuong, Homer Walke, Anna Walling, Haohuan Wang, Lili Yu, and Ury Zhilinsky. $\pi_{0.5}$: a Vision-Language-Action Model with Open-World Generalization, 2025.

Tao Jiang, Tianyuan Yuan, Yicheng Liu, Chenhao Lu, Jianning Cui, Xiao Liu, Shuiqi Cheng, Jiyang Gao, Huazhe Xu, and Hang Zhao. Galaxea Open-World Dataset and G0 Dual-System VLA Model, 2025.

Alexander Khazatsky, Karl Pertsch, Suraj Nair, Ashwin Balakrishna, Sudeep Dasari, Siddharth Karamcheti, Soroush Nasiriany, Mohan Kumar Srirama, Lawrence Yunliang Chen, Kirsty Ellis, Peter David Fagan, Joey Hejna, Masha Itkina, Marion Lepert, Yecheng Jason Ma, Patrick Tree Miller, Jimmy Wu, Suneel Belkhale, Shivin Dass, Huy Ha, Arhan Jain, Abraham Lee, Youngwoon Lee, Marius Memmel, Sungjae Park, Ilija Radosavovic, Kaiyuan Wang, Albert Zhan, Kevin Black, Cheng Chi, Kyle Beltran Hatch, Shan Lin, Jingpei Lu, Jean Mercat, Abdul Rehman, Pannag R. Sanketi, Archit Sharma, Cody Simpson, Quan Vuong, Homer Rich Walke, Blake Wulfe, Ted Xiao, Jonathan Heewon Yang, Arefeh Yavary, Tony Z. Zhao, Christopher Agia, Rohan Baijal, Mateo Guaman Castro, Daphne Chen, Qiuyu Chen, Trinity Chung, Jaimyn Drake, Ethan Paul Foster, Jensen Gao, David Antonio Herrera, Minho Heo, Kyle Hsu, Jiaheng Hu, Donovon Jackson, Charlotte Le, Yunshuang Li, Roy Lin, Zehan Ma, Abhiram Maddukuri, Suvir Mirchandani, Daniel Morton, Tony Nguyen, Abigail O'Neill, Rosario Scalise, Derick Seale, Victor Son, Stephen Tian, Emi Tran, Andrew E. Wang, Yilin Wu, Annie Xie, Jingyun Yang, Patrick Yin, Yunchu Zhang, Osbert Bastani, Glen Berseth, Jeannette Bohg, Ken Goldberg, Abhinav Gupta, Abhishek Gupta, Dinesh Jayaraman, Joseph J. Lim, Jitendra Malik, Roberto Martín-Martín, Subramanian Ramamoorthy, Dorsa Sadigh, Shuran Song, Jiajun Wu, Michael C. Yip, Yuke Zhu, Thomas Kollar, Sergey Levine, and Chelsea Finn. DROID: A Large-Scale In-The-Wild Robot Manipulation Dataset. In Dana Kulic, Gentiane Venture, Kostas E. Bekris, and Enrique Coronado (eds.), *Robotics: Science and Systems XX, Delft*, 2024.

Moo Jin Kim, Karl Pertsch, Siddharth Karamcheti, Ted Xiao, Ashwin Balakrishna, Suraj Nair, Rafael Rafailov, Ethan Paul Foster, Pannag R. Sanketi, Quan Vuong, Thomas Kollar, Benjamin Burchfiel, Russ Tedrake, Dorsa Sadigh, Sergey Levine, Percy Liang, and Chelsea Finn. OpenVLA: An Open-Source Vision-Language-Action Model. In Pulkit Agrawal, Oliver Kroemer, and Wolfram Burgard (eds.), *Conference on Robot Learning, 6-9 November 2024, Munich, Germany*, Proceedings of Machine Learning Research, 2024.

Lin Li, Qihang Zhang, Yiming Luo, Shuai Yang, Ruilin Wang, Fei Han, Mingrui Yu, Zelin Gao, Nan Xue, Xing Zhu, et al. Causal World Modeling for Robot Control, 2026a.

Xukun Li, Yu Sun, Lei Zhang, Bosheng Huang, Yibo Peng, Yuan Meng, Haojun Jiang, Shaoxuan Xie, Guocai Yao, Alois Knoll, et al. DECO: Decoupled Multimodal Diffusion Transformer for Bimanual Dexterous Manipulation with a Plugin Tactile Adapter. *arXiv preprint arXiv:2602.05513*, 2026b.

Bo Liu, Yifeng Zhu, Chongkai Gao, Yihao Feng, Qiang Liu, Yuke Zhu, and Peter Stone. LIBERO: Benchmarking Knowledge Transfer for Lifelong Robot Learning. *Advances in Neural Information Processing Systems*, 36:44776–44791, 2023.

Shilong Liu, Zhaoyang Zeng, Tianhe Ren, Feng Li, Hao Zhang, Jie Yang, Qing Jiang, Chunyuan Li, Jianwei Yang, Hang Su, et al. Grounding DINO: Marrying DINO with Grounded Pre-Training for Open-Set Object Detection. In *European conference on computer vision*, pp. 38–55. Springer, 2024.

Yunze Liu, Yun Liu, Che Jiang, Kangbo Lyu, Weikang Wan, Hao Shen, Boqiang Liang, Zhoujie Fu, He Wang, and Li Yi. HOI4D: A 4D Egocentric Dataset for Category-Level

Human-Object Interaction. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 20981–20990. IEEE, 2022.

Ilya Loshchilov and Frank Hutter. Decoupled Weight Decay Regularization. In *7th International Conference on Learning Representations, ICLR*, 2019.

Oier Mees, Lukas Hermann, Erick Rosete-Beas, and Wolfram Burgard. CALVIN: A Benchmark for Language-Conditioned Policy Learning for Long-Horizon Robot Manipulation Tasks. *IEEE Robotics and Automation Letters*, (3), 2022.

Henrique Morimitsu, Xiaobin Zhu, Roberto M Cesar, Xiangyang Ji, and Xu-Cheng Yin. DPFlow: Adaptive Optical Flow Estimation with a Dual-Pyramid Framework. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 17810–17820. IEEE, 2025.

Nikhila Ravi, Valentin Gabeur, Yuan-Ting Hu, Ronghang Hu, Chaitanya Ryali, Tengyu Ma, Haitham Khedr, Roman Rädle, Chloe Rolland, Laura Gustafson, et al. SAM 2: Segment Anything in Images and Videos. In *International Conference on Learning Representations*, volume 2025, pp. 28085–28128, 2025.

Yichao Shen, Fangyun Wei, Zhiying Du, Yaobo Liang, Yan Lu, Jiaolong Yang, Nanning Zheng, and Baining Guo. VideoVLA: Video Generators Can Be Generalizable Robot Manipulators. In Danielle Belgrave, Cheng Zhang, Laura N. Montoya, Hsuan-Tien Lin, Razvan Pascanu, Piotr Koniusz, Marzyeh Ghassemi, Nancy Chen, Iván Vladimir Meza Ruíz, and Arturo Loaiza-Bonilla (eds.), *Annual Conference on Neural Information Processing Systems*, 2025.

Marcel Torne, Anthony Simeonov, Zechu Li, April Chan, Tao Chen, Abhishek Gupta, and Pulkit Agrawal. Reconciling Reality through Simulation: A Real-to-Sim-to-Real Approach for Robust Manipulation. *arXiv preprint arXiv:2403.03949*, 2024.

Team Wan, Ang Wang, Baole Ai, Bin Wen, Chaojie Mao, Chen-Wei Xie, Di Chen, Feiwu Yu, Haiming Zhao, Jianxiao Yang, et al. Wan: Open and Advanced Large-Scale Video Generative Models. *arXiv preprint arXiv:2503.20314*, 2025.

Hanwen Wang, Weizhi Zhao, Xiangyu Wang, Siyuan Huang, He Lin, Boyuan Zheng, Rongtao Xu, Gang Wang, Yao Mu, He Wang, et al. DexJoCo: A Benchmark and Toolkit for Task-Oriented Dexterous Manipulation on MuJoCo, 2026a.

Jianyuan Wang, Minghao Chen, Shangzhan Zhang, Nikita Karaev, Johannes Schönberger, Patrick Labatut, Piotr Bojanowski, David Novotny, Andrea Vedaldi, and Christian Rupprecht. VGGT-Ω. *arXiv preprint arXiv:2605.15195*, 2026b. URL `https://arxiv.org/abs/2605.15195`.

Letian Wang, Chuhan Zhang, Rishabh Kabra, Jasper Uijlings, Steven Waslander, Andrew Zisserman, Joao Carreira, Kaiming He, Misha Andriluka, Eduard Gabriel Bazavan, et al. Video Generation Models Are General-Purpose Vision Learners. In *European Conference on Computer Vision*, pp. 608–627. Springer, 2026c.

Peng Wang, Shuai Bai, Sinan Tan, Shijie Wang, Zhihao Fan, Jinze Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, et al. Qwen2-VL: Enhancing vision-language model's perception of the world at any resolution. *arXiv preprint arXiv:2409.12191*, 2024.

Ruicheng Wang, Sicheng Xu, Yue Dong, Yu Deng, Jianfeng Xiang, Zelong Lv, Guangzhong Sun, Xin Tong, and Jiaolong Yang. MoGe-2: Accurate Monocular Geometry with Metric Scale and Sharp Details. *Advances in Neural Information Processing Systems*, 38:35928–35959, 2026d.

Seonghyeon Ye, Yunhao Ge, Kaiyuan Zheng, Shenyuan Gao, Sihyun Yu, George Kurian, Suneel Indupuru, You Liang Tan, Chuning Zhu, Jiannan Xiang, et al. World Action Models Are Zero-Shot Policies, 2026.

Jinliang Zheng, Jianxiong Li, Zhihao Wang, Dongxiu Liu, Xirui Kang, Yuchun Feng, Yinan Zheng, Jiayin Zou, Yilun Chen, Jia Zeng, et al. X-VLA: Soft-Prompted Transformer as Scalable Cross-Embodiment Vision-Language-Action Model. In *International Conference on Learning Representations*, volume 2026, pp. 60580–60606, 2026.

Brianna Zitkovich, Tianhe Yu, Sichun Xu, Peng Xu, Ted Xiao, Fei Xia, Jialin Wu, Paul Wohlhart, Stefan Welker, Ayzaan Wahid, Quan Vuong, Vincent Vanhoucke, Huong T. Tran, Radu Soricut, Anikait Singh, Jaspiar Singh, Pierre Sermanet, Pannag R. Sanketi, Grecia Salazar, Michael S. Ryoo, Krista Reymann, Kanishka Rao, Karl Pertsch, Igor Mordatch, Henryk Michalewski, Yao Lu, Sergey Levine, Lisa Lee, Tsang-Wei Edward Lee, Isabel Leal, Yuheng Kuang, Dmitry Kalashnikov, Ryan Julian, Nikhil J. Joshi, Alex Irpan, Brian Ichter, Jasmine Hsu, Alexander Herzog, Karol Hausman, Keerthana Gopalakrishnan, Chuyuan Fu, Pete Florence, Chelsea Finn, Kumar Avinava Dubey, Danny Driess, Tianli Ding, Krzysztof Marcin Choromanski, Xi Chen, Yevgen Chebotar, Justice Carbajal, Noah Brown, Anthony Brohan, Montserrat Gonzalez Arenas, and Kehang Han. RT-2: Vision-Language-Action Models Transfer Web Knowledge to Robotic Control. In Jie Tan, Marc Toussaint, and Kourosh Darvish (eds.), *Conference on Robot Learning, CoRL*, Proceedings of Machine Learning Research, 2023.

A Appendix

This appendix provides additional implementation details, analyses, and experimental results. Appendix B discloses the use of large language models. Appendix C describes the data sources and multimodal dataset construction pipeline. Appendix D details the RGB-style modality mappings. Appendix E presents intermediate-representation visualizations, analyses of common attention regions across modalities, complete per-task results on RoboTwin 2.0, and qualitative rollouts on Galaxea R1 Lite, RoboTwin, and DexJoCo.

B THE USAGE OF LARGE LANGUAGE MODELS

We acknowledge the use of a large language model for limited assistance in the writing of this paper. The tool was used exclusively for proofreading and improving the linguistic fluency of the text. All scientific content, including the research ideas, methodology, experiment, and analysis, was conducted solely by the authors.

C Multimodal Dataset Construction

C.1 Data Sources

We curate data from eight widely used datasets for robot learning and an additional real-world dataset collected by us using a Galaxea robot: AgiBot, CALVIN, DexYCB, DexJoCo, DROID, Galaxea, HOI4D, LIBERO, and RoboTwin. Together, they cover simulated and real-world tabletop manipulation, long-horizon task execution, dexterous manipulation, bimanual coordination, human grasping, and human–object interaction. The collection spans human hands, dexterous robotic hands, and single-arm, dual-arm, and mobile manipulators. After temporal segmentation, approximately 140,000 RGB source clips are retained. Expanding each clip according to its available modalities yields approximately 600,000 modality-specific training clips and more than 1,200 hours of data across modalities.

- **AgiBot.** Large-scale real-world data covering diverse manipulation tasks performed by dual-arm mobile robots.
- **CALVIN.** A simulated benchmark for language-conditioned, long-horizon tabletop manipulation.
- **DexYCB.** Multi-view RGB-D recordings of human hand–object interactions with detailed hand and object pose annotations.
- **DexJoCo.** A simulated benchmark for task-oriented dexterous manipulation. We replay 1,100 demonstrations spanning six single-arm and five bimanual tasks to collect synchronized visual, state, and force signals.
- **DROID.** Large-scale in-the-wild demonstrations covering diverse real-world single-arm manipulation tasks and environments.
- **Galaxea.** We collect real-world mobile-manipulation trajectories using a Galaxea robot and organize the resulting data in the LeRobot format.
- **HOI4D.** Egocentric RGB-D videos capturing diverse human–object interactions.
- **LIBERO.** A simulated benchmark for multitask and lifelong robot learning across diverse tabletop manipulation tasks.
- **RoboTwin.** A simulated benchmark covering diverse manipulation skills and embodiments. We replay 27,500 Clean and Random trajectories across 50 tasks to collect synchronized multimodal observations, robot states, and force signals.

Figure 6 visualizes samples from three data sources used in our experiments, covering diverse manipulation tasks and robotic embodiments. For RoboTwin and DexJoCo, simulator replay provides temporally aligned RGB, depth, surface normals, semantic segmentation. Optical flow is estimated by DPFlow. Since post-training on our self-collected Galaxea trajectories uses RGB as the only visual modality, we show only their RGB observations.

C.2 Annotation and Processing Pipeline

Multimodal annotation. We process each complete episode to obtain temporally aligned multimodal annotations. Qwen2-VL generates language descriptions when reliable textual annotations are unavailable. For episodes missing one or more visual modalities, we preserve all available native annotations and generate only the absent modalities: DPFlow estimates optical flow, MoGe-2 produces depth and surface normals, and LangSAM generates instruction-relevant semantic masks. All modalities retain the temporal indices of the original episode to ensure frame-level alignment.

Simulation replay and real-world acquisition. For RoboTwin and DexJoCo, we develop an agentic data engine that restores the recorded initial state, replays actions at the original control frequency, and synchronously records RGB, depth, surface normals, segmentation, robot states, camera poses, and actions. Depth, normals, segmentation, and force signals are obtained directly from the simulator to provide accurate geometric and contact supervision. Since optical flow is not directly available from the simulation environments, we compute it from the rendered RGB sequences using DPFlow. This process produces 27,500 RoboTwin trajectories across 50 tasks and 1,100 DexJoCo demonstrations across six single-arm and five bimanual tasks. For Galaxea demonstrations, we record synchronized RGB observations, robot states, and actions.

Quality filtering and packaging. We remove clips with incomplete sequences, invalid timestamps, corrupted frames, severe blur or occlusion, duplicated content. Auxiliary annotations are further checked for valid depth coverage, normalized surface normals, temporally stable segmentation masks, and plausible optical flow. Simulation replays that diverge from the recorded trajectory or fail the task-success condition are also excluded. Each available camera-view and modality pair is finally packaged as an independent Stage 1 sample, allowing trajectories with partially available modalities to remain useful.

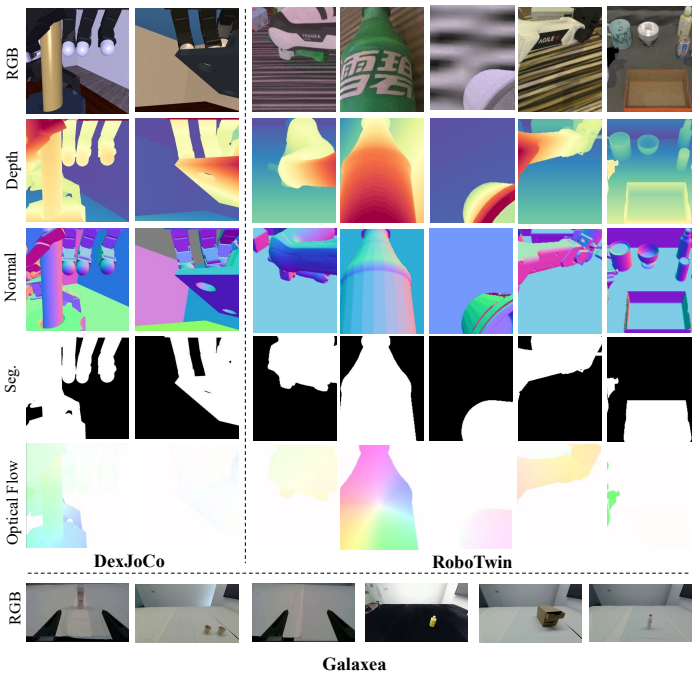

Figure 6: Representative multimodal samples from DexJoCo and RoboTwin and RGB samples from our self-collected Galaxea dataset.

D Modality Mapping

To train all prediction tasks through the native video generation, we convert each target modality into a three-channel RGB-like representation. Let $\mathbf{y}^m \in \mathcal{Y}_m$ denote the original target of modality m. We define a deterministic encoder

$$R_m : \mathcal{Y}_m \to [0,1]^{H\times W\times 3}, \qquad m \in \{\text{rgb}, \text{depth}, \text{normal}, \text{seg}, \text{flow}\}. \tag{5}$$

The encoded video $\mathbf{x}^m = R_m(\mathbf{y}^m)$ can therefore be processed by the Wan video VAE without introducing modality-specific encoders or prediction heads. The target modality is specified through Unified Task Modulation: a fixed modality prompt is encoded separately from the manipulation instruction, projected into a modality embedding, and added to the timestep embedding. The backbone and training objective remain shared across modalities.

Depth mapping. Following Vision Banana, we transform each positive depth value d into a bounded scalar:

$$z(d) = 1 - \left(1 - \frac{d}{\lambda c}\right)^{\lambda+1}, \tag{6}$$

where λ controls the shape of the nonlinear transformation and c determines its depth scale. This transformation allocates greater representational precision to nearby geometry while compressing distant regions. The normalized value is subsequently mapped along a piecewise-linear path over the edges of the RGB color cube, producing a continuous and invertible false-color representation. Non-finite or non-positive depth values are treated as invalid and encoded as black.

Surface-normal mapping. Each surface normal is represented as a camera-space unit vector (x, y, z). After converting the source normal to a consistent right-handed camera coordinate system, we encode the three directional components as

$$R_{\text{normal}}(x, y, z) = \left(\frac{1-x}{2}, \frac{1+y}{2}, \frac{1+z}{2}\right). \tag{7}$$

This channel-wise mapping converts the three signed normal components from $[-1, 1]$ to the RGB range $[0, 1]$, allowing the pretrained video VAE to process surface normals without architectural modifications.

Segmentation mapping. We represent semantic segmentation using instruction-conditioned binary masks. All objects referred to in the text instruction are rendered in white, while all remaining regions are rendered in black. If multiple objects are mentioned, their masks are merged into a white foreground mask.

Optical-flow mapping. Optical flow assigns each pixel a horizontal displacement u and a vertical displacement v. To preserve small motions while accommodating large displacements, we apply signed-log normalization:

$$\phi(x) = \text{sign}(x)\frac{\log\left(1 + \frac{|x|}{a}\right)}{\log\left(1 + \frac{s}{a}\right)}, \tag{8}$$

where s specifies the maximum displacement scale and a controls the strength of logarithmic compression. The normalized horizontal and vertical displacements, together with the normalized motion magnitude $\sqrt{u^2 + v^2}$, form a unified three-channel representation that preserves both motion direction and strength.

E ADDITIONAL EXPERIMENT RESULTS

E.1 Emergence of Multimodal World Representations

To examine how modality-specific structures evolve during generation, we decode intermediate representations at multiple Transformer layers and denoising steps for five generation

targets: RGB, depth, surface normals, segmentation, and optical flow. As shown in Figures 7 and 8, recognizable modality-specific spatial structures are visible in intermediate layers and become clearer as denoising progresses. These visualizations provide qualitative evidence that modality-related information is accessible before the final output. The modality-specific outputs are displayed in the three-channel representations defined in Appendix D.

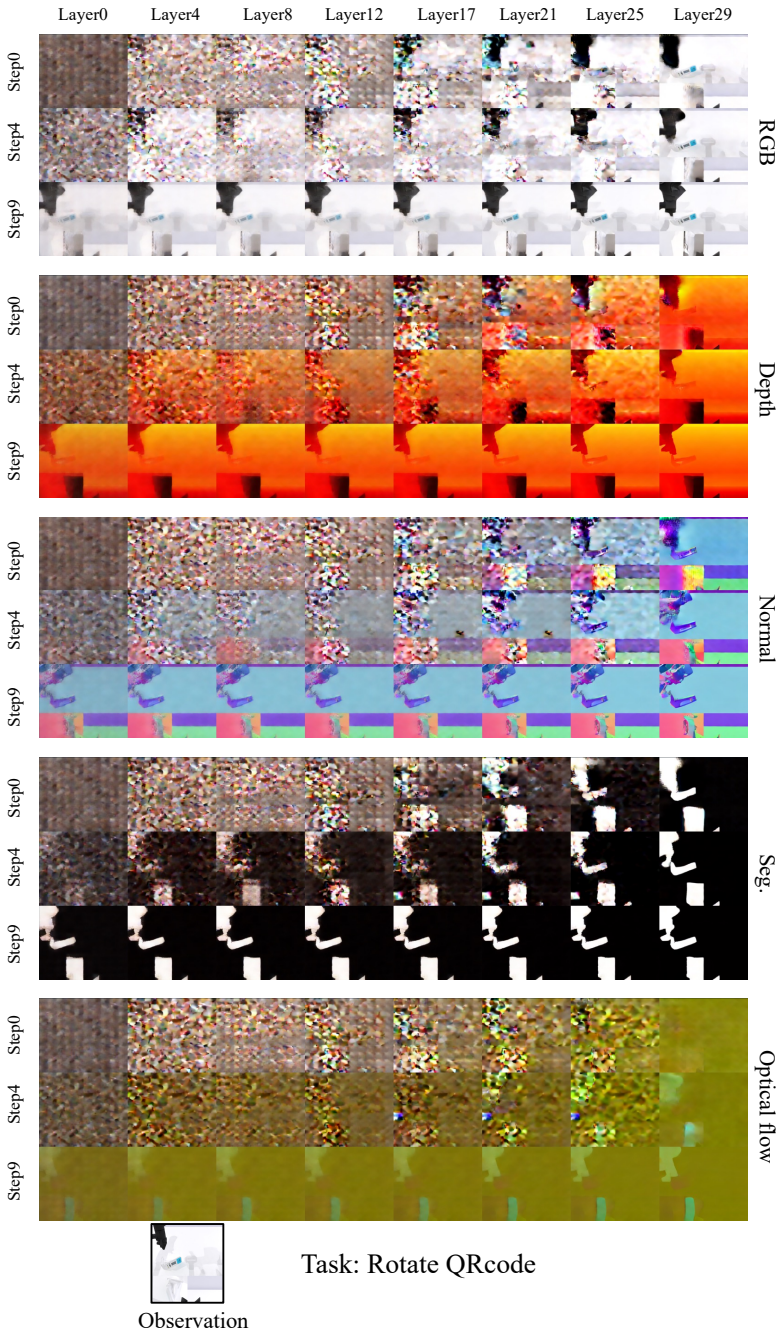

Figure 7: This picture shows that modality-specific spatial structures emerge progressively throughout the generation process for the Rotate QR Code task.

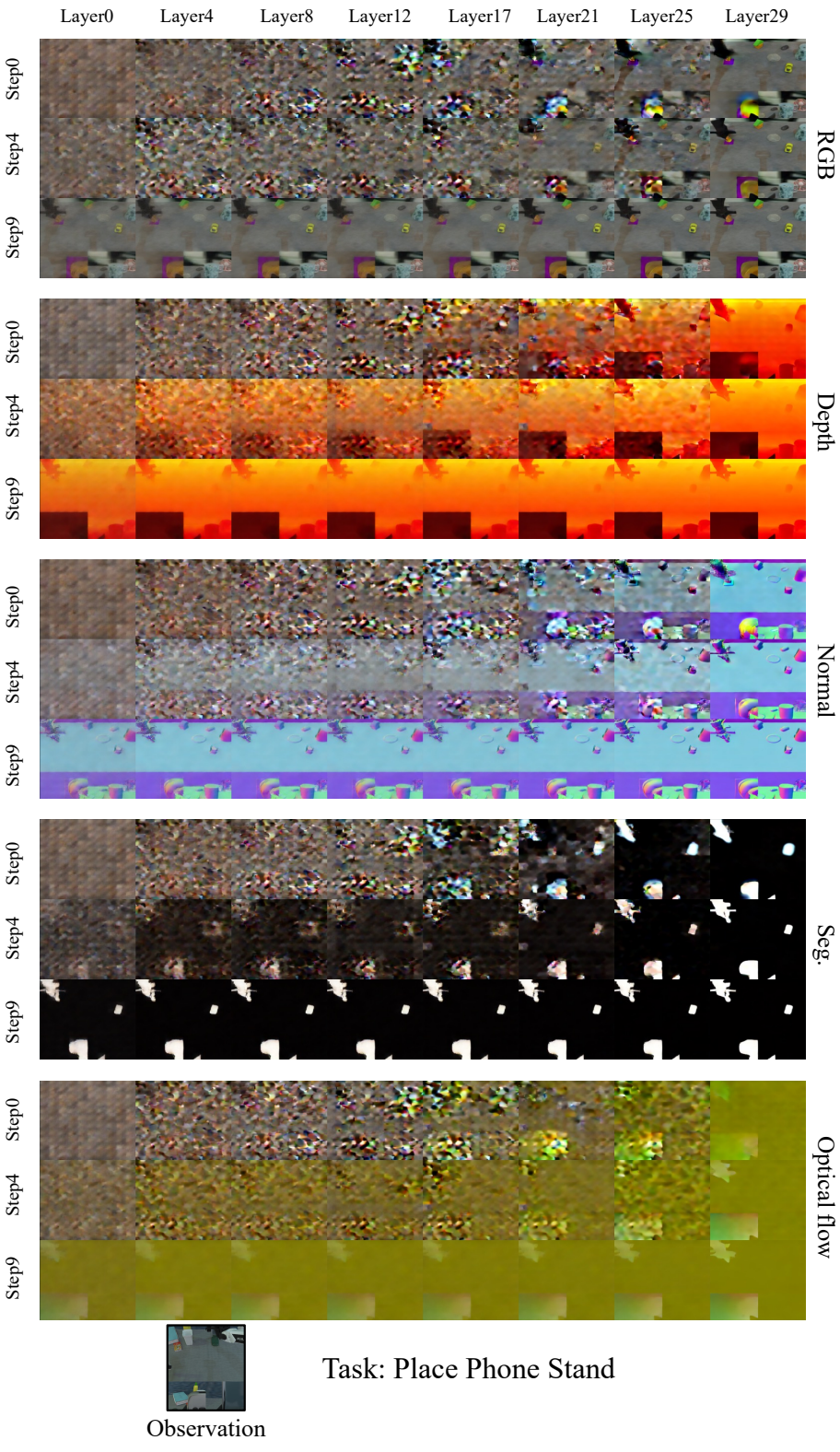

Figure 8: This picture shows that modality-specific spatial structures emerge progressively throughout the generation process for the Place Phone Stand task.

E.2 Multimodal Common Attention Regions.

As shown in Figure 5 in the main paper, the attention maps visualize which spatial regions the action tokens focus on across the five modalities. Here, we provide additional examples across different tasks and Transformer layers. For a controlled comparison, the five modality-specific inference runs for each scenario use the same RGB observation, language instruction, robot state, condition latent, and initial video and action noise; only the generation-task condition is changed to specify RGB, depth, surface normal, segmentation, or optical flow. We perform the standard 10-step joint flow-matching denoising procedure, record the attention at the final denoising step, and visualize Transformer layers 0, 4, 9, 14, 19, 24, and 29. At each selected layer, we extract the post-softmax attention weights from the action queries to the future-video keys. We average these weights over 24 attention heads, the action tokens corresponding to 16 future action steps, and the temporal dimension of the eight-frame future prediction, while preserving the spatial dimensions. The resulting heatmap therefore summarizes the overall spatial focus of the predicted action chunk. As shown in Figure 9, the five modalities exhibit largely consistent high-attention regions at corresponding blocks despite their different encodings. These results suggest that jointly incorporating multimodal world understanding provides complementary information for action prediction.

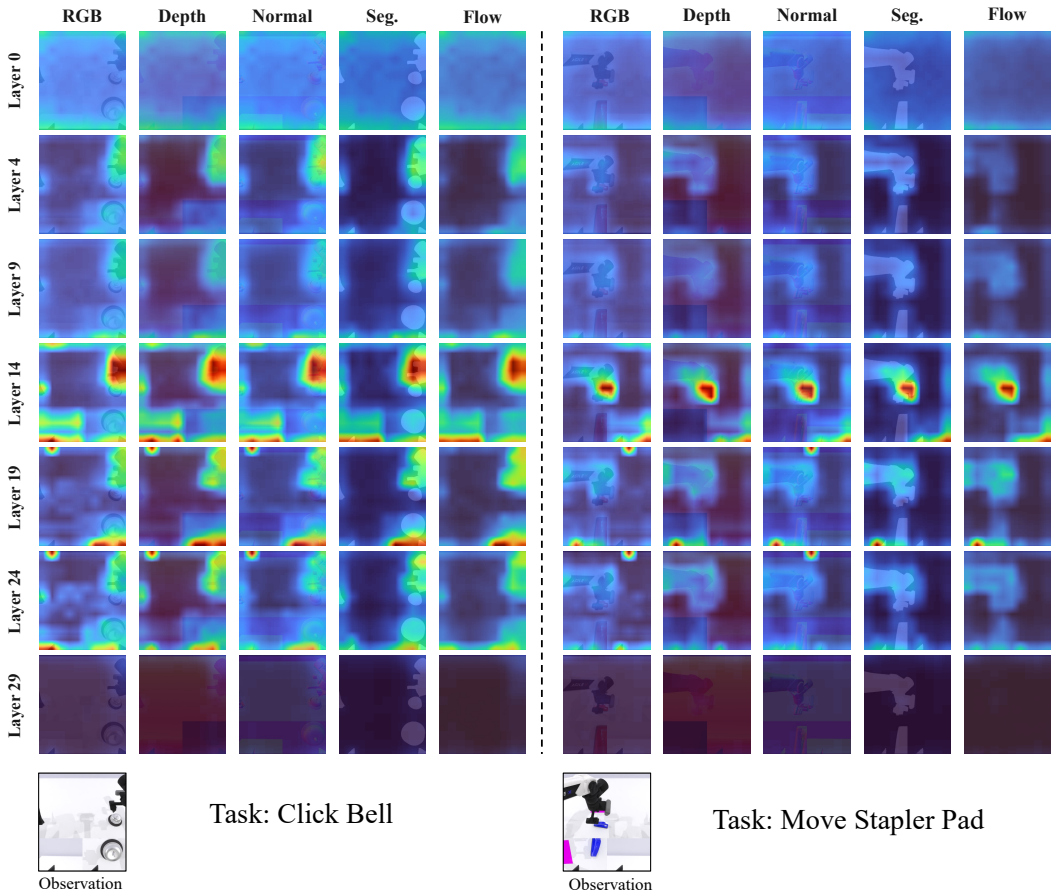

Figure 9: Additional attention visualizations across different tasks and Transformer layers.

E.3 Qualitative Results on the Galaxea

We collect real-world demonstrations with a Galaxea R1 Lite robot to study post-training under the visual and physical conditions of an actual tabletop workspace. Unlike the simulation benchmarks, these trajectories are recorded directly from the robot during object manipulation. Each demonstration contains RGB observations from the head and wrist cameras, robot states, actions, and a language instruction. We use the collected real-world robot data for task-specific post-training.

The demonstrations cover three manipulation skills with different object geometries and contact requirements. *Bottle Laydown* requires the robot to grasp an upright bottle, transport it over the basket, and release it in a horizontal configuration inside the basket. *Bowl Stack* requires the robot to approach the left bowl, lift it without collision, and place it concentrically on the right bowl. *Cola Side-by-Side* requires the robot to push one cola bottle across the tabletop until the two bottles form the intended side-by-side arrangement.

We define success by the final relationship between the manipulated objects and their supporting surfaces. A Bottle Laydown trial is successful only when the bottle is both inside the basket and approximately horizontal. A Bowl Stack trial is successful when the lifted bowl is stably supported by the other bowl after release. A Cola Side-by-Side trial is successful when both bottles remain on the tabletop in adjacent, non-overlapping positions after the pushing motion. These criteria focus on the achieved task state rather than on the intermediate arm trajectory alone.

As shown in Figure 10, each rollout is visualized from synchronized head and wrist views. The eight frames in each row cover the main stages of the interaction, including the initial scene, object approach, contact or grasp, transport or pushing, placement, and the final configuration. The two views provide complementary evidence for evaluating both the global object arrangement and the local end-effector interaction.

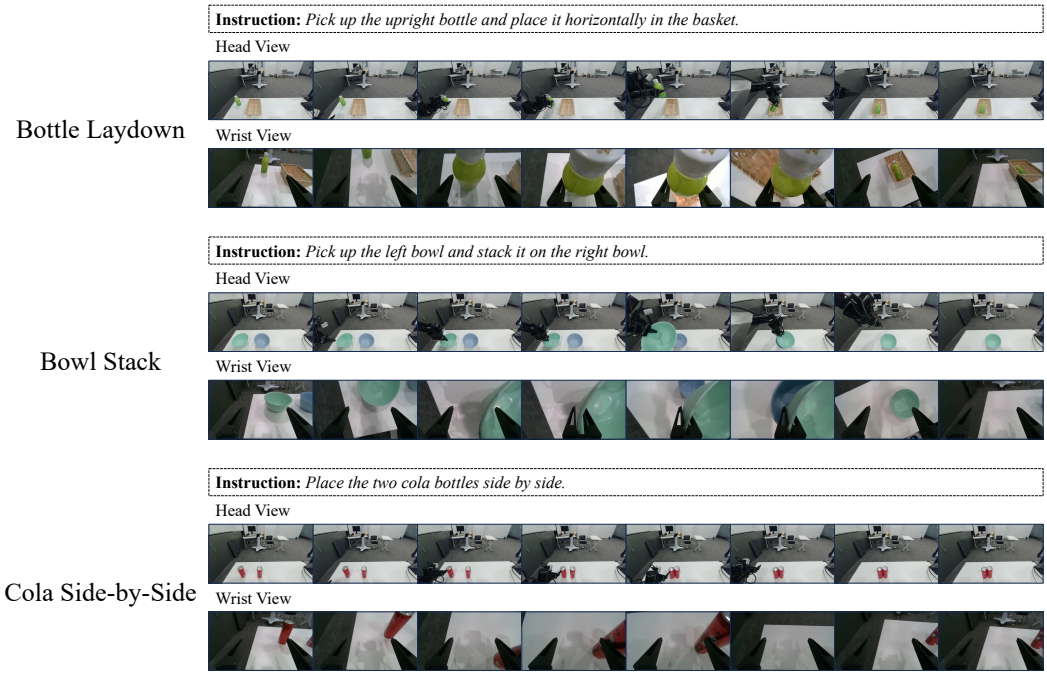

Figure 10: Qualitative real-robot rollouts on three manipulation tasks using the Galaxea R1 Lite robot. Each task is shown from synchronized head and wrist RGB views, with eight frames illustrating the progression from the initial scene to the final task configuration.

E.4 Quantitative Results on the RoboTwin Benchmark

Following the protocol of RoboTwin 2.0, we adopt a multi-task training setup where all models are trained with 2,500 demonstrations collected from clean scenes (50 per task) together with 25,000 demonstrations from heavily randomized scenes (500 per task). In this section, we report the per-task success rates on RoboTwin under clean and randomized evaluation settings.

Table 5: Per-task success rates on RoboTwin under the clean and randomized evaluation settings. We retain the source-reported average of 72.80% for X-VLA under the Clean setting. The arithmetic mean of the 50 displayed per-task success rates is 72.88%. LingBot-VA averages are recomputed from the reported per-task results.

Task	$\pi_{0.5}$		X-VLA		Motus		LingBot-VA		KRUG	
	Clean	Rand.	Clean	Rand.	Clean	Rand.	Clean	Rand.	Clean	Rand.
Adjust Bottle	**100**	**99**	**100**	**99**	89	93	90	94	98	**99**
Beat Block Hammer	**96**	93	92	88	95	88	**96**	**98**	90	94
Blocks Ranking RGB	92	85	83	83	**99**	97	**99**	**98**	**99**	**98**
Blocks Ranking Size	49	26	67	74	75	63	**94**	**96**	67	74
Click Alarmclock	98	89	99	99	**100**	**100**	99	**100**	**100**	**100**
Click Bell	99	66	**100**	**100**	**100**	**100**	**100**	**100**	**100**	**100**
Dump Bin Bigbin	92	**97**	79	77	95	91	89	96	**99**	94
Grab Roller	**100**	**100**	**100**	**100**	**100**	**100**	**100**	**100**	**100**	**100**
Handover Block	66	57	73	37	86	73	**99**	78	91	**85**
Handover Mic	**98**	**97**	0	0	78	63	94	96	89	87
Hanging Mug	18	17	23	27	38	38	40	28	**56**	**61**
Lift Pot	96	85	99	**100**	96	99	**100**	99	97	**100**
Move Can Pot	51	55	89	86	34	74	**94**	**97**	84	86
Move Pillbottle Pad	84	61	73	71	93	96	99	99	**100**	**100**
Move Playingcard Away	96	84	93	98	**100**	96	**100**	99	**100**	**100**
Move Stapler Pad	56	42	78	73	83	**85**	**91**	79	74	72
Open Laptop	90	96	93	**100**	95	91	92	94	**98**	97
Open Microwave	34	77	79	71	**95**	**91**	82	86	91	86
Pick Diverse Bottles	81	71	58	36	90	**91**	89	82	**94**	**91**
Pick Dual Bottles	93	63	47	36	96	90	**100**	**99**	98	97
Place A2B Left	87	82	48	49	88	79	97	93	**100**	**96**
Place A2B Right	87	84	36	36	91	87	97	**95**	**98**	**95**
Place Bread Basket	77	64	81	71	91	94	**97**	**95**	92	93
Place Bread Skillet	85	66	77	67	86	83	**95**	90	88	**93**
Place Burger Fries	94	87	94	94	98	98	97	95	**100**	**100**
Place Can Basket	62	62	49	52	**81**	76	**81**	**84**	75	73
Place Cans Plasticbox	94	84	97	98	98	94	**100**	99	**100**	**100**
Place Container Plate	99	95	97	95	98	**99**	99	97	**100**	98
Place Dual Shoes	75	75	79	88	93	87	94	89	**99**	**99**
Place Empty Cup	**100**	99	**100**	98	99	98	**100**	**100**	**100**	**100**
Place Fan	87	85	80	75	91	87	**99**	93	**99**	**97**
Place Mouse Pad	60	39	70	70	66	68	93	**96**	**96**	85
Place Object Basket	80	76	44	39	81	87	**91**	88	**91**	**93**
Place Object Scale	86	80	52	74	88	85	**96**	95	**96**	**96**
Place Object Stand	91	85	86	88	98	97	**99**	96	**99**	**100**
Place Phone Stand	81	81	88	87	87	86	**97**	**97**	90	81
Place Shoe	92	93	96	95	99	97	98	98	**100**	**99**
Press Stapler	87	83	92	98	93	98	85	82	**100**	**100**
Put Bottles Dustbin	84	79	74	77	81	79	**87**	**91**	77	76
Put Object Cabinet	80	79	46	48	**88**	71	85	**87**	87	81
Rotate QRcode	89	87	34	33	89	73	**96**	91	93	**93**
Scan Object	72	65	14	36	67	66	**96**	**91**	71	71
Shake Bottle	99	97	99	**100**	**100**	97	**100**	97	**100**	**100**
Shake Bottle Horizontally	99	99	**100**	**100**	**100**	98	**100**	99	**100**	**100**
Stack Blocks Three	91	76	6	10	91	95	99	98	**100**	**100**
Stack Blocks Two	97	**100**	92	87	**100**	98	**100**	98	**100**	**100**
Stack Bowls Three	77	71	76	86	79	87	86	83	**87**	**88**
Stack Bowls Two	95	96	96	93	98	98	94	98	**100**	**100**
Stamp Seal	79	55	76	82	93	92	96	97	**100**	**98**
Turn Switch	62	54	40	61	**84**	**78**	44	45	75	65
Average	82.74	76.76	72.80	72.84	88.66	87.02	**92.90**	91.50	92.76	**91.82**

E.5 Qualitative Results on the RoboTwin Benchmark

In this section, we visualize KRUG rollouts across a range of manipulation tasks in the RoboTwin benchmark. As shown in Figure 11, KRUG reliably and effectively completes these tasks across diverse scenarios.

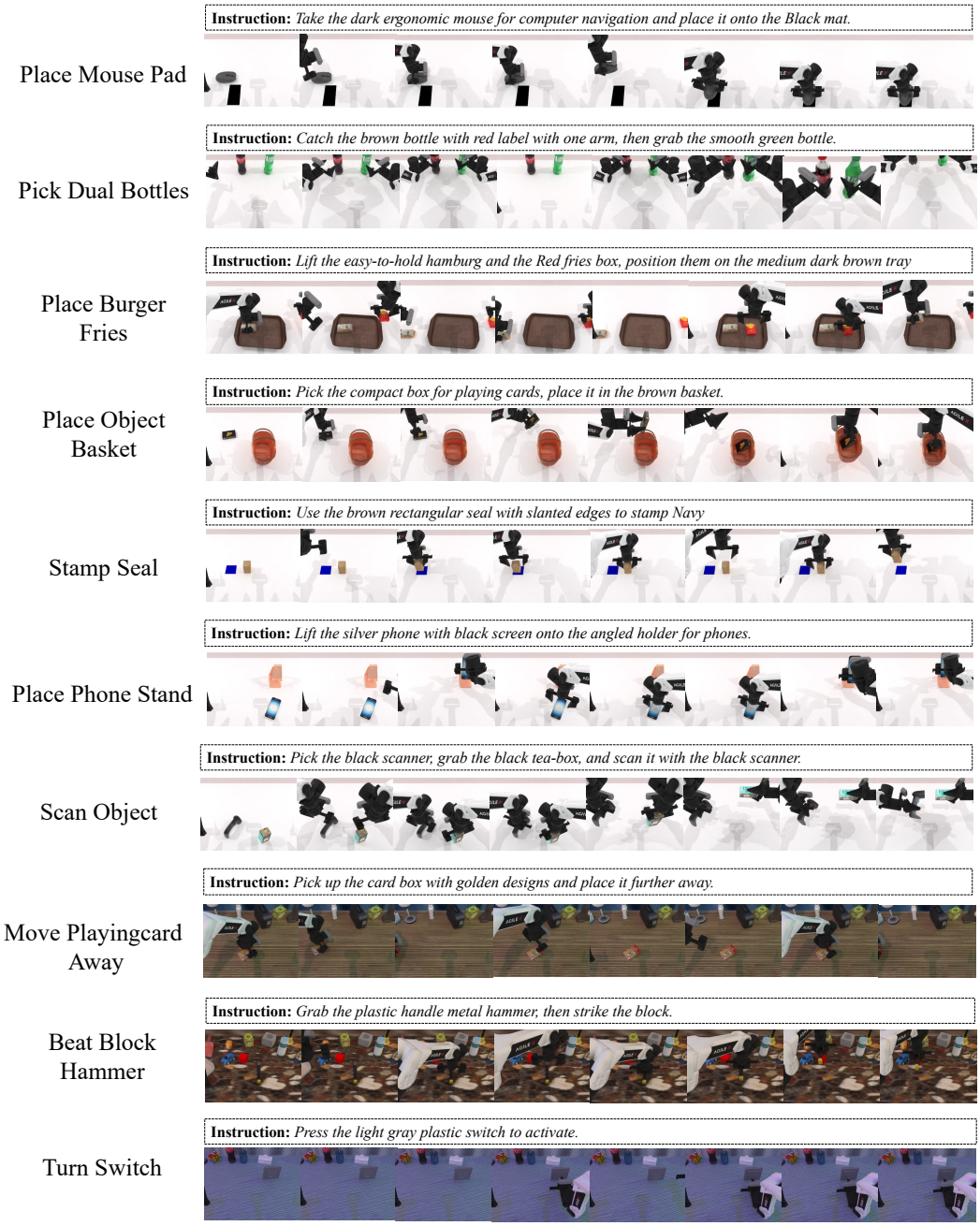

Figure 11: Visualization of KRUG rollouts across multiple RoboTwin manipulation tasks under both randomized and clean conditions.

E.6 Qualitative Results on the DexJoCo Benchmark

We further evaluate KRUG on DexJoCo, a benchmark featuring task-oriented manipulation with both single-arm and bimanual setups. The representative rollouts in Figure 12 show that KRUG can handle diverse object interactions and coordinated bimanual behaviors.

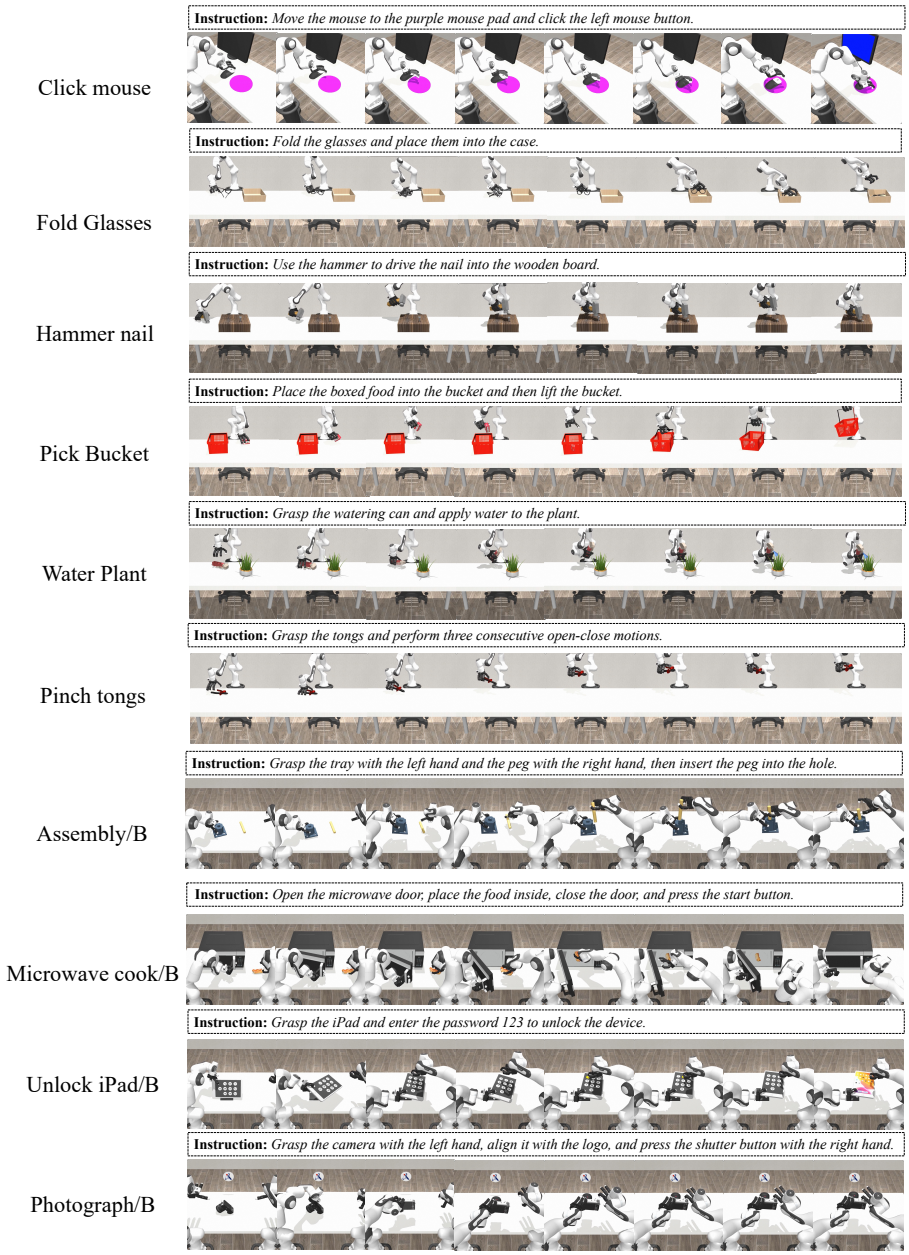

Figure 12: Visualization of KRUG rollouts across multiple single-arm and bimanual manipulation tasks in the DexJoCo benchmark.