Learning 4D Embodied World Models

Supplementary Material

1. Implementation Details

1.1. Video Diffusion Model Details

We trained an RGB-DN video diffusion model using the CogVideoX [13] architecture. On the input side, our depth normal projector and RGB projector shared the same architecture. On the output side, our `Conv3DNet` consisted of three layers, while the MLP had two layers, both with a dimension of 1024. The model outputs videos with 49 frames, utilizing gradient checkpointing to optimize memory usage. We set a global batch size of 16 and used `bf16` precision to accelerate. For sampling, we employed the DDPM scheduler across 50 steps and set a classifier-free guidance scale of 7.5. The training spanned 40,000 iterations with an initial learning rate of 1×10^{-4}, a gradient clipping of 1.0, and a 1,000-step warmup. The optimizer used was Adam with ϵ set to 1×10^{-15}, and an exponential moving average (EMA [8]) decay of 0.99 was applied to stabilize training.

1.2. 4D Scene Generation

The parameters for the loss term in Eq.12 are set differently for the RT-1 [3], Bridge [12] and RLBench [6] datasets, as shown in the table below. It is worth noting that the selection of λ varies for different scenarios. In practice, achieving the best performance requires tuning these parameters.

Dataset	λ_d	λ_b	λ_{g1}	λ_{g2}
RT-1, Bridge	20	200	20	20
RLBench	20	200	2	2

Table 1. Loss Term Parameters for RT-1 and RLBench Datasets

In Figure 1, we present a visualization of the 3D robotic scene reconstruction optimized using our proposed method in the BridgeV2 [12] dataset. After estimating the depth and normal with the estimator, we refine the outputs to reconstruct the scene accurately. The figure includes untextured rendering and texture-rendered views, where the wall textures are significantly enhanced due to normal optimization. The side perspective view shows the improved shape and geometry reconstruction. Notably, the wall and table surfaces are well-aligned, appearing perpendicular to each other, further validating the effectiveness of our optimization process in capturing accurate spatial relationships.

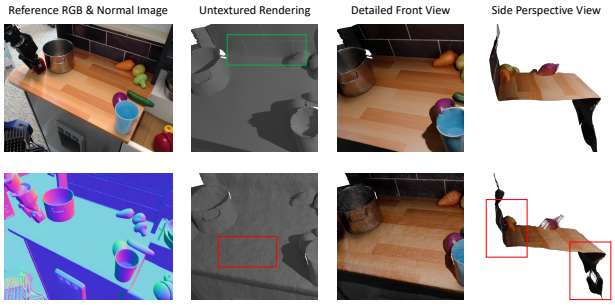


Figure 1. Visualization of the optimized 3D robotic scene reconstruction. The untextured renderings show enhanced detail (green) and improved surface smoothness (red). The side perspective view highlights accurate shape and geometry optimization, including the perpendicular alignment of the wall and table (red).

1.3. Implementation Details for Robotics Planning

For the RLBench training, we adopted the same architecture and methods as our video diffusion model, with the primary difference being that we used 13 frames and fine-tuned the model. For the action prediction stage, we first filter out the background and floor from the data, focusing only on the points of the table and the objects manipulated by the robotic arm, and then sample 8192 points from the filtered point cloud. In our inverse dynamic model, the PointNet extracts features from this point cloud, concatenated with the instruction's language embedding, and passed into a 4-layer MLP, finally outputting the 7DoF actions. To augment the data and better adapt to the output of video diffusion models, we add significant Gaussian noise (with a relative magnitude of 20%) to both the image and point cloud coordinates.

2. More Generation Results

2.1. Data Annotation

In this section, we first compare our data generation method with 3D-VLA [14]. They use ZoeDepth [1] for depth map estimation and directly map them into 3D space. The comparison results, shown in Figure 2, evaluate the quality of point cloud generation for both methods, with cubes replacing vertices for rendering. Our generated data demonstrates higher realism, while 3D-VLA exhibits noticeable shape distortion. Figure 12 showcases some of the RGB, depth, and normal images from the datasets we used, along with the corresponding natural language instructions.

2.2. 4D Video Generation

Our world model demonstrates strong generalization capabilities in complex, unseen scenes and with novel objects. Qual-

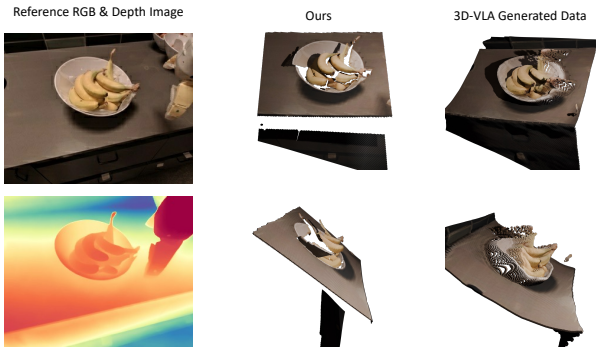


Figure 2. Comparison of point cloud generation quality between our method and 3D-VLA

itative results are illustrated in Figure 7 and Figure 8. The input images were sourced from various domains, including the π_0 [2] website's table bussing section, the RT-2 [4] teaser, photography collections from Unsplash, and films such as Sherlock. We designed challenging prompts that require an understanding of articulated objects and compositional reasoning, highlighting the full potential of our model. We further present in-domain video generation results on the RT1, Bridge, and RLBench datasets, as shown in Figure 9, Figure 10, and Figure 11, respectively. Additional videos are provided in the TesserAct website gallery for extended analysis and qualitative evaluation.

To assess our model's zero-shot video generation performance, we conducted a comparative evaluation against the Video Prediction Policy [5] (VPP) using both GPT-4o evaluation and SSIM metrics. The evaluation dataset includes in-the-wild images and previously unseen, new collected real-world data. The GPT prompt is provided in Figure 3. Quantitative results are summarized in Table 2, where our model outperforms VPP across all metrics.

Datasets	In-the-Wild	New Collected	
Models	GPT Eval. ↑	GPT Eval. ↑	SSIM ↑
VPP	5.10	6.51	0.740
TesserAct	**7.70**	**7.75**	**0.746**

Table 2. Zero-shot video generation comparison between VPP and our model (TesserAct), evaluated using GPT-4o and SSIM. The two evaluation settings include in-the-wild samples and newly collected real-world data.

3. More Robot Manipulation Results

3.1. Real World Experiments

To validate the real-world applicability of our 4D embodied world model, we conducted experiments on three manipulation tasks using a robotic platform. A total of 100 demonstration samples were collected, covering 2 tasks **1)** cloth (deformable objects) moving and folding and **2)** picking up a cup of a specified color. For each task, the policy model was retrained on the collected data and evaluated over 20 independent trials. Notably, the cloth folding task represents a zero-shot video generation challenge for TesserAct, as the Bridge dataset [12] from Open-X collection [11] does not contain raw cloth-folding samples. Figure 4 shows an example prediction generated by our model and robot execution sequence for the cloth manipulation task. The results of the real-world experiments are summarized in Table 3.

LLM Evaluation Prompt

You are a video quality assessment expert. I will provide you with a set of frames generated by a model (and a set of original video frames), with the text description used during generation. Please determine whether the generated results accurately reproduce the described content and whether there are any distortions, misalignments, or inconsistencies. Please evaluate the results based on the following criteria:

1. **Content Accuracy:** Do the generated frames accurately represent the content described in the prompt?
2. **Temporal Consistency:** Are the video frames temporally coherent?
3. **Perceptual Quality:** Are there noticeable blurs, artifacts, or structural abnormalities?
4. **Consistency with GT:** If the ground-truth video is provided, are the generated frames similar to the GT frames in terms of content and motion?

Please provide a brief evaluation comment and a score (0–10) based on the above criteria.

Figure 3. Evaluation prompt used for GPT-4o-based assessment of generated videos. The prompt instructs the model to rate video quality based on content accuracy, temporal consistency, perceptual quality, and-when available-consistency with ground truth. For in-the-wild samples where GT is not available, only the prompt and generated frames are provided.

	`pick cup`	`move cloth`	`fold cloth`
Success Rate	16 / 20	10 / 20	4 / 20

Table 3. Success rates of the policies in real-world experiments.

Our experimental environment consists of a workstation, a robotic arm, and one external camera. The specific configuration is as follows: **FR3 Robotic Arm:** The FR3 (Franka Research 3) robotic arm is a high-precision 7-Degree-of-Freedom collaborative robot arm equipped with flexible movement capabilities and high repeatability. This robotic

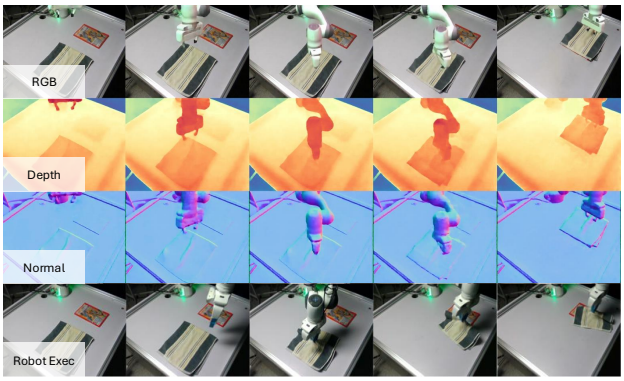


Figure 4. Example from our model for the real-world cloth folding task.

arm is responsible for executing the predicted actions from our inverse dynamics models. **Workstation:** The workstation is a desk that serves as the fixed base for securely mounting the robotic arm and provides a stable platform for object placement and experimental tasks. **Camera System:** A single Intel Realsense D435i camera is mounted on a rigid pole attached to the workstation, positioned to cover the entire workspace. It captures real-time RGB and depth images of the robotic arm's movements and the operational activities on the desk.

3.2. Explicit Action Planning

One potential application of our generated mesh is to extract action trajectories directly. As illustrated in Figure 5, we track the robotic arm in the video to capture its motion path. This trajectory is subsequently lifted into 3D space, enabling the reconstruction of the robot arm's action trajectory. The red line in the visualization represents the captured action trajectory.

Figure 5. Tracking of robotic arm trajectories on the Bridge dataset

4. Inference Time and Memory Usage

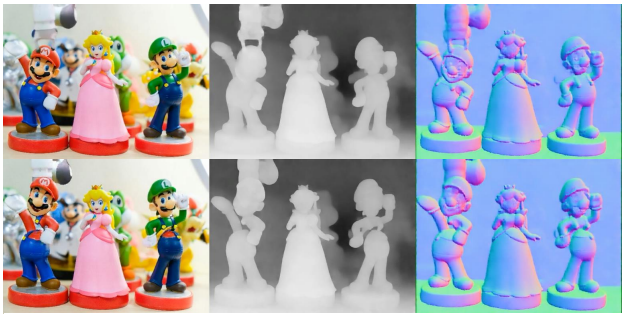
There exists an inherent trade-off between achieving strong generalization capabilities and minimizing time and memory consumption. Our method, while more computationally intensive than Video Prediction Policy [5] (VPP), provides significantly better video generalization performance. Compared to OpenVLA [7], our system is faster and more memory-efficient. We show time and memory usage in the Table 4 below. Looking ahead, we aim to further accelerate inference through Latent Consistency Models [10], as well as caching strategies like TeaCache [9] and EasyCache [15].

	Diffusion	3D Recon.	PointNet	VPP	OpenVLA
Time / Memory	12.4s / 20G	3.2s / 0.4G	0.5s / 2G	8s / 5.4G	18s / 25G

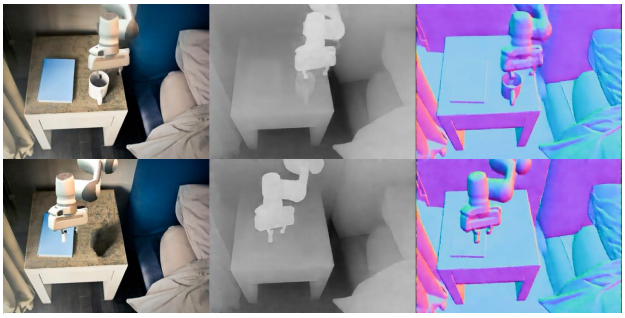
Table 4. Inference time and peak memory usage for each stage of our TesserAct pipeline compared to VPP and OpenVLA.

5. Limitations

While our RGB-DN representation of a 4D world model is cheap and easy to predict, it only captures a single view of the world. To construct a more complete 4D world model, it may be interesting in the future to have a generative model that generates multiple RGB-DN views of the world, which can then be integrated to form a more complete 4D world model. Despite the strengths of our approach, the generated videos still exhibit limitations, as shown in the examples in Figure 6. These include visual hallucinations such as object disappearance, incomplete or incorrect functional understanding, and constrained generalization to novel or unseen objects and environments.



Limited generalization to unseen objects



Object disappearance

Figure 6. Failure cases of our method

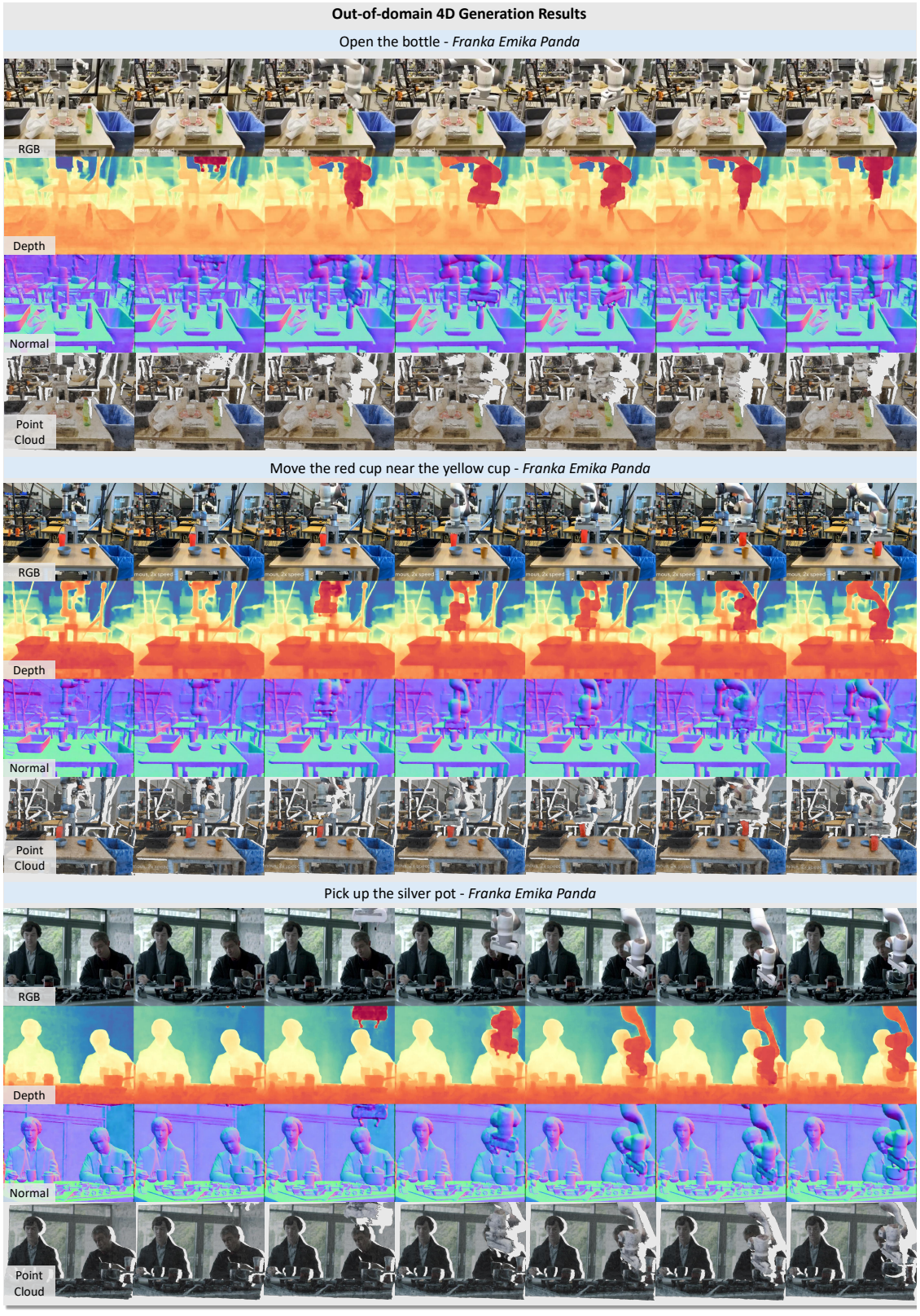


Figure 7. Out-of-domain 4D **generation results**

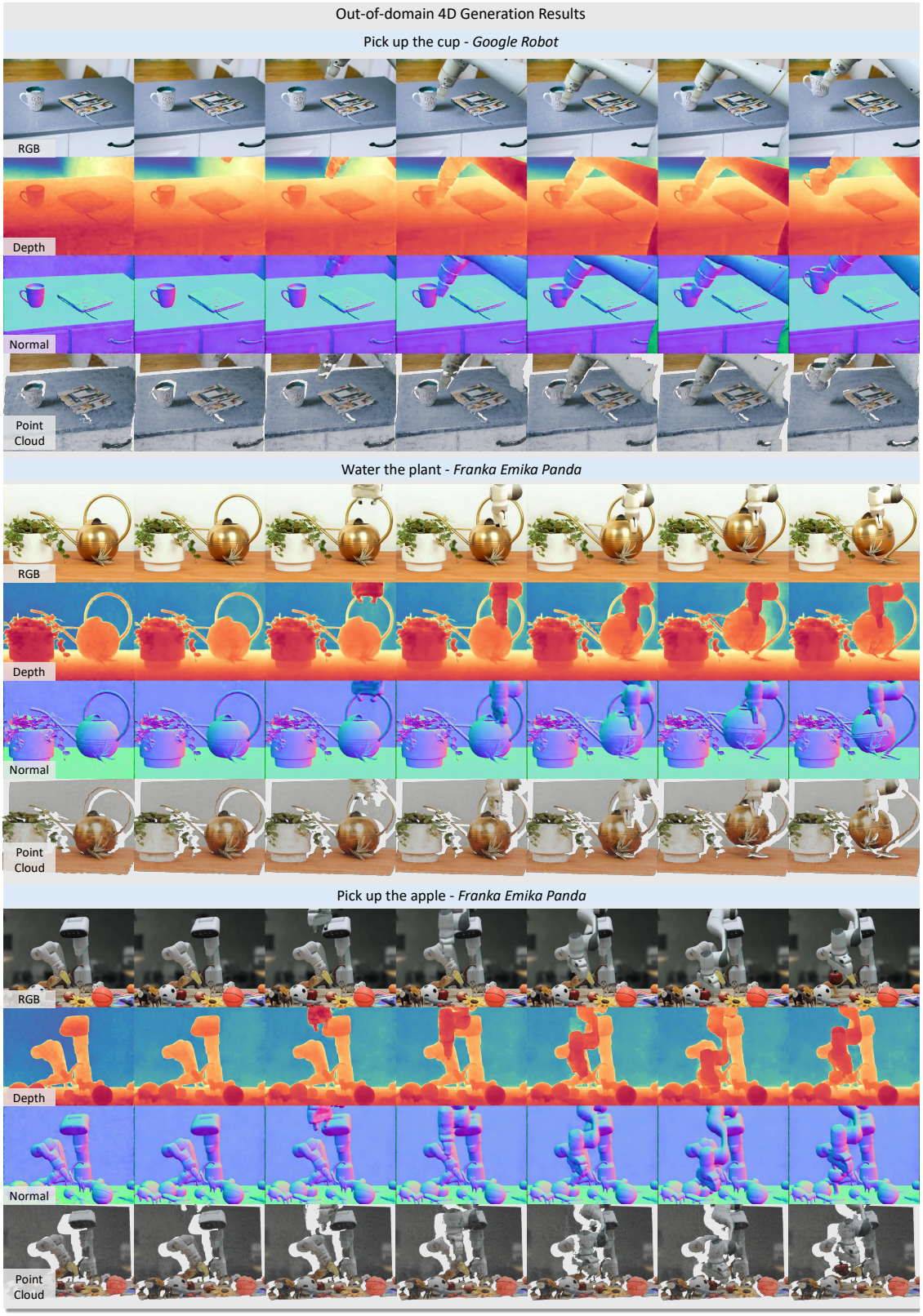


Figure 8. Out-of-domain 4D **generation results**

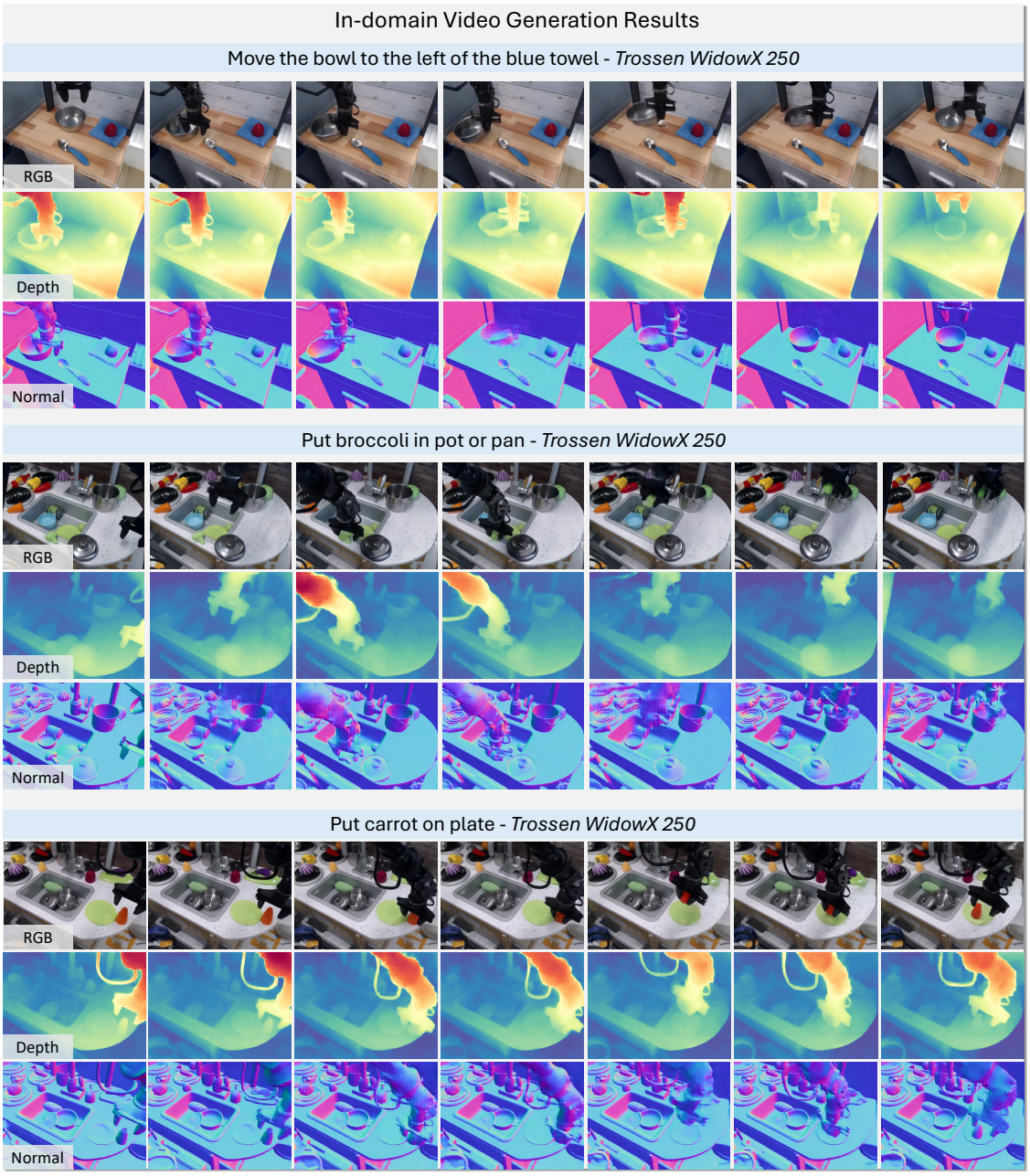


Figure 9. In-domain RGB-DN video **generation results** on Bridge dataset

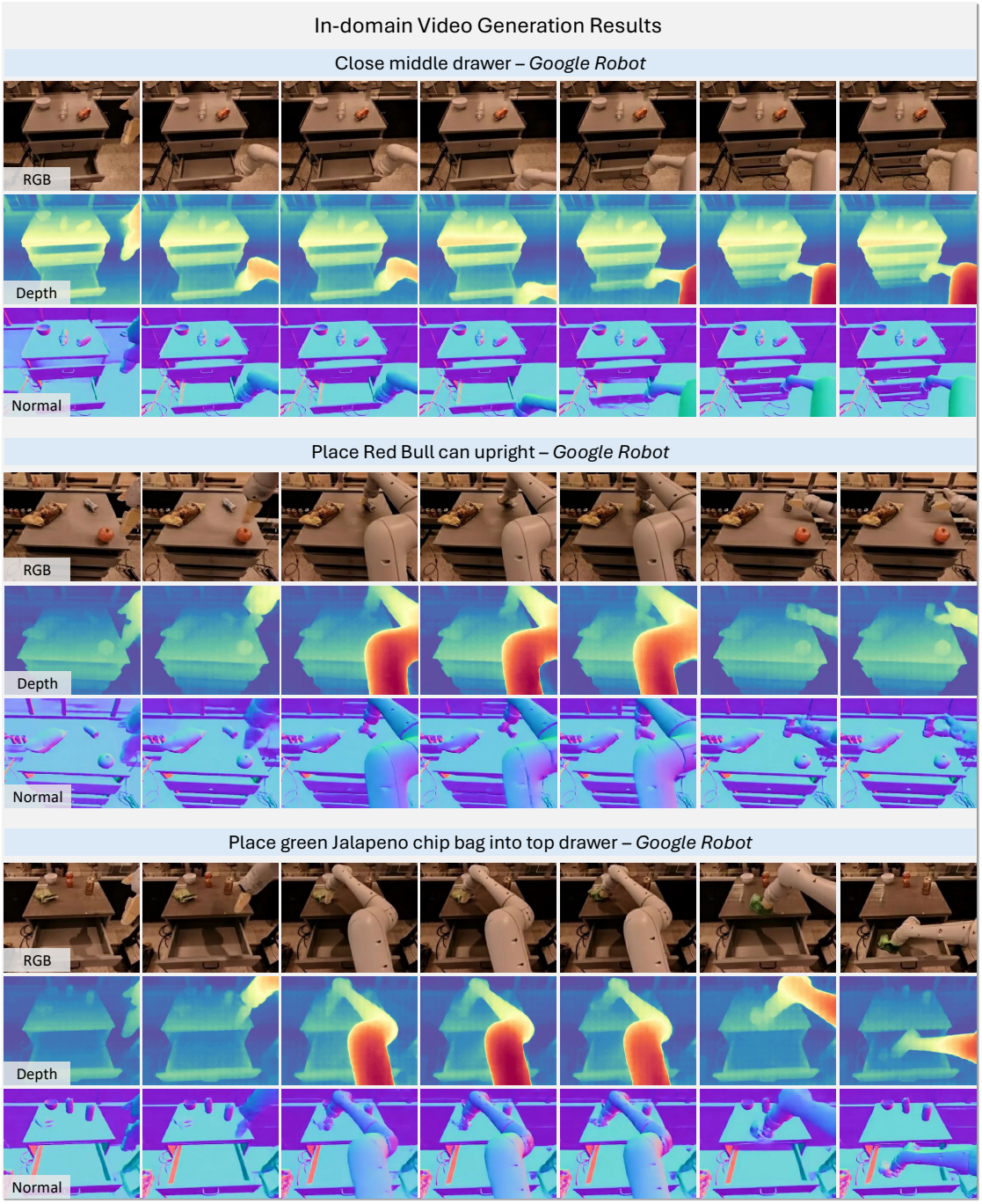


Figure 10. In-domain RGB-DN video **generation results** on RT1 dataset

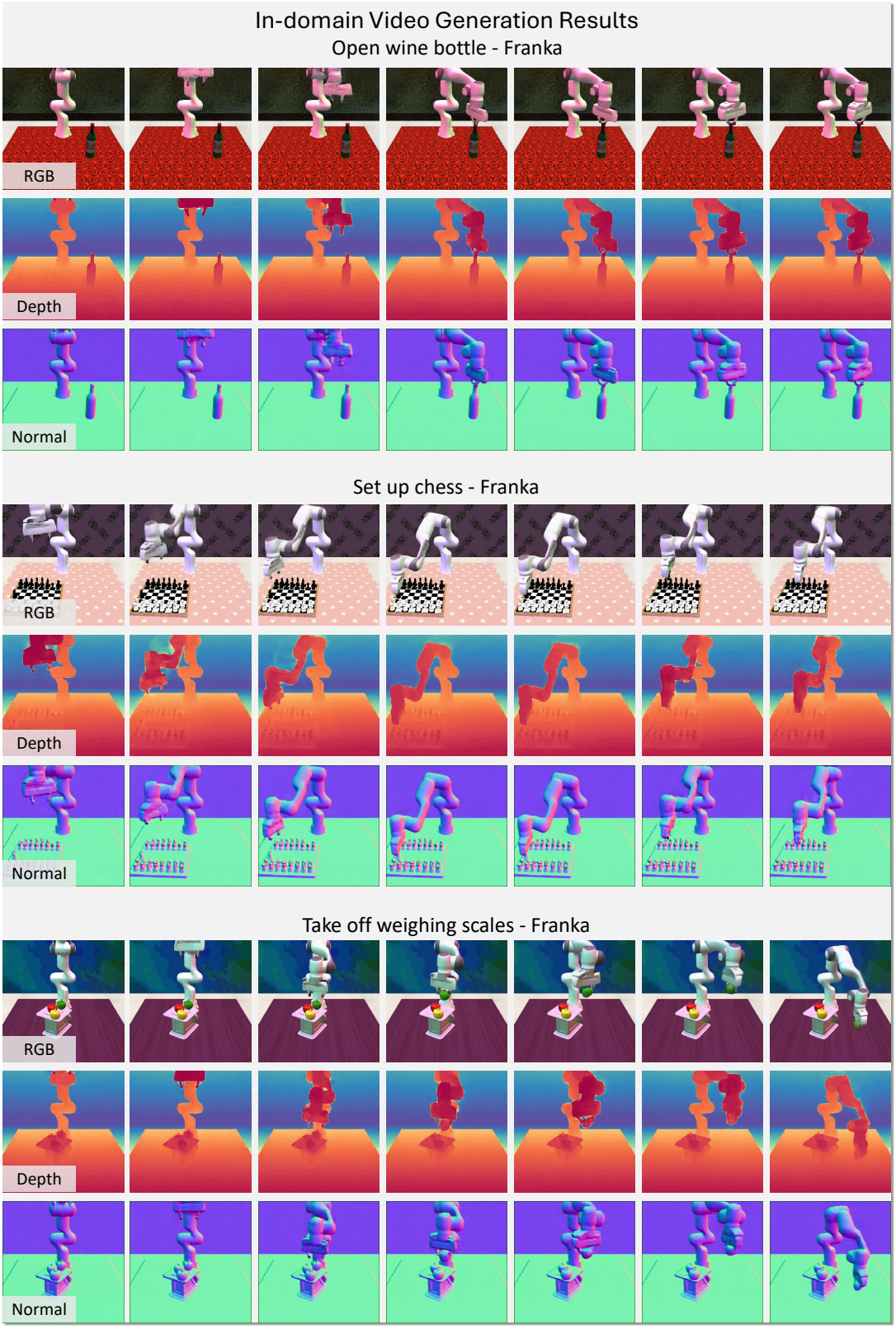


Figure 11. In-domain RGB-DN video **generation results** on RLBench dataset

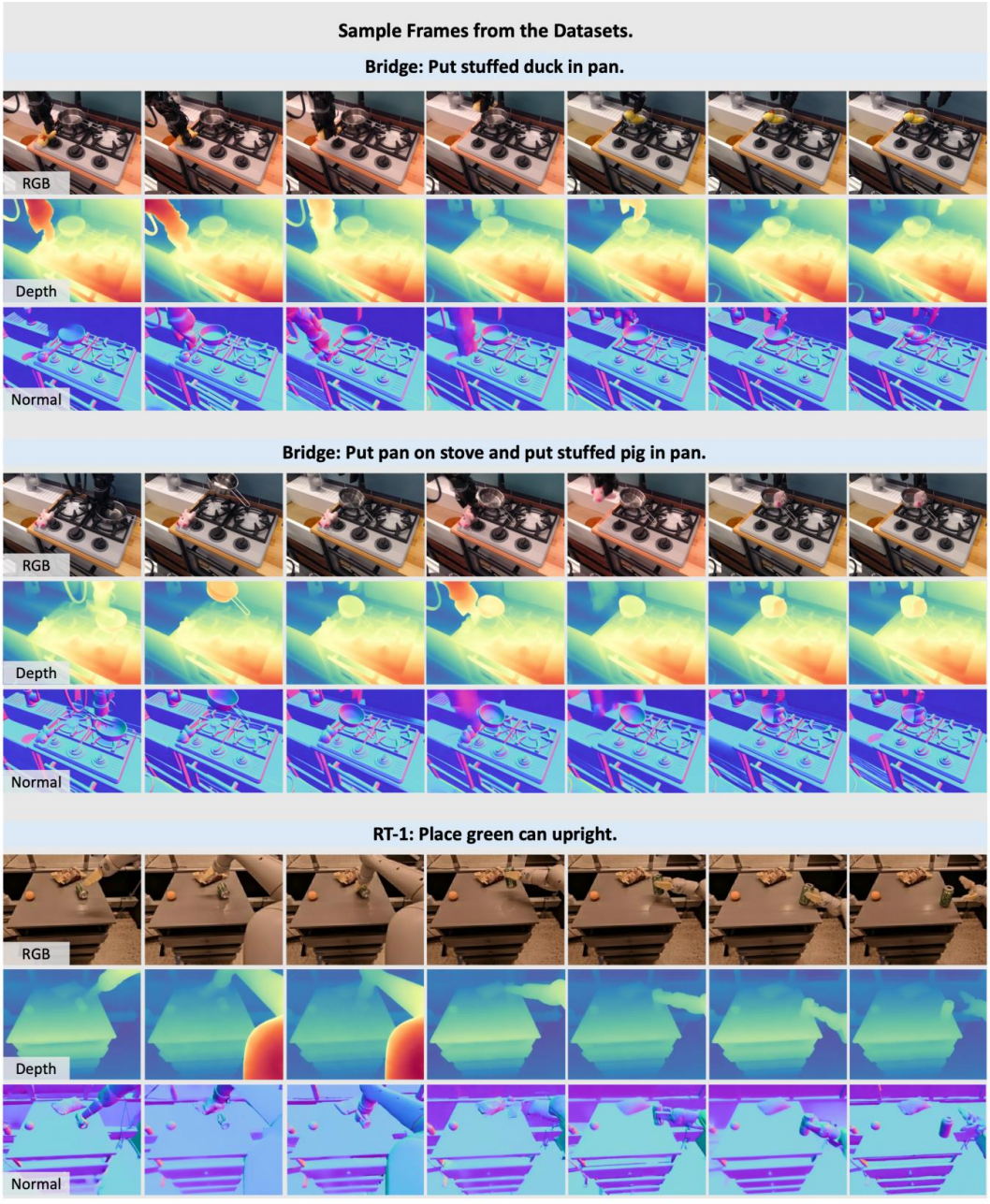


Figure 12. Some sample frames extracted from the **datasets** Bridge [12] and RT-1 [3].

References

[1] Shariq Farooq Bhat, Reiner Birkl, Diana Wofk, Peter Wonka, and Matthias Müller. Zoedepth: Zero-shot transfer by combining relative and metric depth. *arXiv preprint arXiv:2302.12288*, 2023. 1

[2] Kevin Black, Noah Brown, Danny Driess, Adnan Esmail, Michael Equi, Chelsea Finn, Niccolo Fusai, Lachy Groom, Karol Hausman, Brian Ichter, et al. $\pi 0$: A vision-language-action flow model for general robot control. corr, abs/2410.24164, 2024. doi: 10.48550. *arXiv preprint ARXIV.2410.24164*, 2024. 2

[3] Anthony Brohan, Noah Brown, Justice Carbajal, Yevgen Chebotar, Joseph Dabis, Chelsea Finn, Keerthana Gopalakrishnan, Karol Hausman, Alex Herzog, Jasmine Hsu, et al. Rt-1: Robotics transformer for real-world control at scale. *arXiv preprint arXiv:2212.06817*, 2022. 1, 9

[4] Anthony Brohan, Noah Brown, Justice Carbajal, Yevgen Chebotar, Xi Chen, Krzysztof Choromanski, Tianli Ding, Danny Driess, Avinava Dubey, Chelsea Finn, et al. Rt-2: Vision-language-action models transfer web knowledge to robotic control. *arXiv preprint arXiv:2307.15818*, 2023. 2

[5] Yucheng Hu, Yanjiang Guo, Pengchao Wang, Xiaoyu Chen, Yen-Jen Wang, Jianke Zhang, Koushil Sreenath, Chaochao Lu, and Jianyu Chen. Video prediction policy: A generalist robot policy with predictive visual representations. *arXiv preprint arXiv:2412.14803*, 2024. 2, 3

[6] Stephen James, Zicong Ma, David Rovick Arrojo, and Andrew J. Davison. Rlbench: The robot learning benchmark & learning environment. *IEEE Robotics and Automation Letters*, 2020. 1

[7] Moo Jin Kim, Karl Pertsch, Siddharth Karamcheti, Ted Xiao, Ashwin Balakrishna, Suraj Nair, Rafael Rafailov, Ethan Foster, Grace Lam, Pannag Sanketi, et al. Openvla: An open-source vision-language-action model. *arXiv preprint arXiv:2406.09246*, 2024. 3

[8] Frank Klinker. Exponential moving average versus moving exponential average. *Mathematische Semesterberichte*, 58: 97–107, 2011. 1

[9] Feng Liu, Shiwei Zhang, Xiaofeng Wang, Yujie Wei, Haonan Qiu, Yuzhong Zhao, Yingya Zhang, Qixiang Ye, and Fang Wan. Timestep embedding tells: It's time to cache for video diffusion model. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 7353–7363, 2025. 3

[10] Simian Luo, Yiqin Tan, Longbo Huang, Jian Li, and Hang Zhao. Latent consistency models: Synthesizing high-resolution images with few-step inference. *arXiv preprint arXiv:2310.04378*, 2023. 3

[11] Open X-Embodiment Team. Open X-Embodiment: Robotic learning datasets and RT-X models. https://arxiv.org/abs/2310.08864, 2023. 2

[12] Homer Walke, Kevin Black, Abraham Lee, Moo Jin Kim, Max Du, Chongyi Zheng, Tony Zhao, Philippe Hansen-Estruch, Quan Vuong, Andre He, Vivek Myers, Kuan Fang, Chelsea Finn, and Sergey Levine. Bridgedata v2: A dataset for robot learning at scale. In *Conference on Robot Learning (CoRL)*, 2023. 1, 2, 9

[13] Zhuoyi Yang, Jiayan Teng, Wendi Zheng, Ming Ding, Shiyu Huang, Jiazheng Xu, Yuanming Yang, Wenyi Hong, Xiaohan Zhang, Guanyu Feng, et al. Cogvideox: Text-to-video diffusion models with an expert transformer. *arXiv preprint arXiv:2408.06072*, 2024. 1

[14] Haoyu Zhen, Xiaowen Qiu, Peihao Chen, Jincheng Yang, Xin Yan, Yilun Du, Yining Hong, and Chuang Gan. 3d-vla: A 3d vision-language-action generative world model. *arXiv preprint arXiv:2403.09631*, 2024. 1

[15] Xin Zhou, Dingkang Liang, Kaijin Chen, Tianrui Feng, Xiwu Chen, Hongkai Lin, Yikang Ding, Feiyang Tan, Hengshuang Zhao, and Xiang Bai. Less is enough: Training-free video diffusion acceleration via runtime-adaptive caching. *arXiv preprint arXiv:2507.02860*, 2025. 3