2026 IEEE International Conference on Robotics and Automation (ICRA 2026)
June 1-5, 2026. Vienna, Austria

VistaBot: View-Robust Robot Manipulation via Spatiotemporal-Aware View Synthesis

Songen Gu[1,2*], Yuhang Zheng[2,4*], Weize Li[2], Yupeng Zheng[2,3], Yating Feng[2],
Xiang Li[2], Yilun Chen[2], Pengfei Li[2], Wenchao Ding[1,2✉]
[1]Fudan University, [2]TARS Robotics, [3]UCAS, [4]NUS
✉ Corresponding Author, * Equal Contribution

***Abstract*— Recently, end-to-end robotic manipulation models have gained significant attention for their generalizability and scalability. However, they often suffer from limited robustness to camera viewpoint changes when training with a fixed camera. In this paper, we propose VistaBot, a novel framework that integrates feed-forward geometric models with video diffusion models to achieve view-robust closed-loop manipulation without requiring camera calibration at test time. Our approach consists of three key components: 4D geometry estimation, view synthesis latent extraction, and latent action learning. VistaBot is integrated into both action-chunking (ACT) and diffusion-based (π_0) policies and evaluated across simulation and real-world tasks. We further introduce the View Generalization Score (VGS) as a new metric for comprehensive evaluation of cross-view generalization. Results show that VistaBot improves VGS by 2.79× and 2.63× over ACT and π_0, respectively, while also achieving high-quality novel view synthesis. Our contributions include a geometry-aware synthesis model, a latent action planner, a new benchmark metric, and extensive validation across diverse environments. The code and models will be made publicly available.**

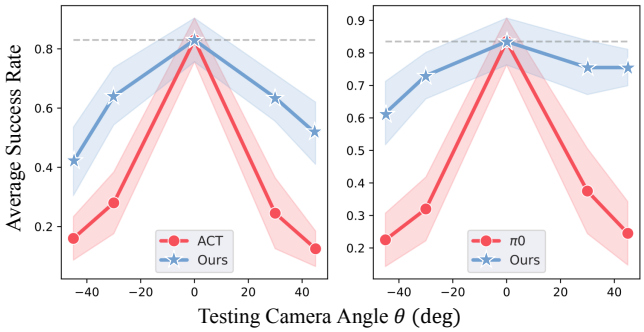

Fig. 1: Our proposed VistaBot demonstrates superior cross-view generalizability compared with SOTA visuomotor policy (π_0 and ACT). As shown in the figure, when the camera observation angle undergoes significant changes, VistaBot consistently maintains a high average success rate even under substantial camera viewpoint changes, whereas the success rates of baseline policies drop to nearly zero as the viewpoint deviates from the training condition.

I. Introduction

End-to-end robotic manipulation have recently gained momentum in robotics, ranging from specialist visuomotor policies trained via imitation learning [1], [2] to generalist vision-language-action (VLA) models trained on large-scale multi-task datasets [3], [4]. These approaches promise scalability: imitation learning efficiently maps demonstrations to actions, while VLAs aspire to unify instructions and control across diverse tasks. However, a fundamental bottleneck undermines both paradigms: poor generalization across camera viewpoints. Unlike appearance shifts such as lighting or texture, viewpoint variations disrupt the spatial grounding between perception and action. As a result, even slight changes in perspective can cause dramatic policy failures (Fig. 1), often forcing practitioners to re-collect demonstrations or retrain models—an outcome directly at odds with the very scalability that end-to-end frameworks claim to deliver.

To address cross-view generalization, prior efforts have primarily fallen into two directions: (1) Reconstruction-based methods [5], [6] attempt to recover the underlying 3D geometry from synchronized multi-view sequences. While conceptually appealing, they are structurally impractical for real-world deployment: precise multi-camera calibration is tedious, and occlusions inevitably lead to unreliable reconstructions. The tedious process of multi-camera calibration further complicates real-world training and deployment. (2) Video generation-based methods [7], [8] leverage multi-view-consistent video generative models and derive actions from predicted future frames. However, these typically lack integrated action learning and suffer from low inference efficiency, making them unsuitable for closed-loop robotic manipulation.

In this paper, we present VistaBot, a new framework that fuses feed-forward geometric models with video diffusion models, combining their complementary priors into a unified policy learning pipeline. Unlike prior reconstruction-only or generation-only approaches, VistaBot leverages geometry for structural consistency and diffusion for spatiotemporal completion, yielding a representation that is both physically grounded and visually coherent. Our key objective is to enable dynamic, closed-loop manipulation that generalizes across camera views—using only a single training viewpoint for demonstrations and arbitrary test viewpoints at inference, without requiring camera calibration or pose information.

Concretely, our framework consists of three key components: (1) **4D Geometry Estimation**. We fine-tune a feed-forward geometric model to predict depth and relative camera pose from arbitrary inference frames, aligning test views with the training viewpoint. This enables us to lift 2D observations into 3D point clouds, providing a structural scaffold for cross-view reasoning. (2) **Synthesis Latent Extraction**.

We employ a conditional video diffusion model (VDM) to consume the reprojected canonical views and encode them into spatiotemporally rich latent features. Furthermore, we introduce a memory mechanism that injects historical latents, yielding a 4D representation that fuses geometry, appearance, and temporal context for closed-loop control. (3) **Closed-Loop Action Learning**. We train the policy to operate directly on diffusion latents rather than decoded images, allowing it to learn future actions from representations already imbued with object-level and geometric understanding. This design reduces inference overhead and strengthens the coupling between perception and control.

To rigorously evaluate cross-view generalization in robotic manipulation, we embed VistaBot into two representative end-to-end frameworks: the imitation-learning-based ACT [1] and the large-scale VLA model π_0 [4]. We further introduce a new evaluation metric—View Generalization Score (VGS)—to directly quantify policy robustness under viewpoint variation, filling a gap not addressed by existing benchmarks. In extensive experiments across diverse tasks and both simulation and real-world environments, VistaBot consistently improves VGS by 2.79x over ACT and 2.63x over π_0, achieving state-of-the-art performance in viewpoint-robust manipulation. In addition, we assess the fidelity of novel view synthesis (NVS), confirming that VistaBot delivers not only robust closed-loop control but also high-quality cross-view reconstructions. Together, these results demonstrate that VistaBot is both effective and broadly applicable, bridging the gap between novel view synthesis and dynamic robotic manipulation.

Our contributions can be summarized as follows:

- We adapt feed-forward geometric models and video diffusion models to robotic closed-loop control, yielding 4D spatiotemporally consistent latent representations for novel views without requiring camera pose inputs. This removes the dependency on calibration and enables scalable deployment.
- We propose a latent planner that operates directly on VDM latent features, allowing the policy to learn actions from representations already infused with object-level and geometric understanding. This design enables efficient closed-loop manipulation with strong cross-view generalization.
- We introduce the View Generalization Score (VGS), a metric that quantitatively measures policy robustness under viewpoint variation, filling a gap in current evaluation protocols.
- We extensively validate VistaBot across both specialist (ACT) and generalist (π_0) end-to-end frameworks, in simulation and real-world settings, showing that it delivers robust, state-of-the-art performance in both novel view synthesis and closed-loop manipulation.

II. Related work

A. End-to-End Model for Robotic Manipulation

With the advancement of visual foundation models [9], [10] and large-scale robotic datasets [11], [12], end-to-end imitation learning has achieved remarkable progress in robotics. End-to-end policies can be broadly divided into open-loop and closed-loop paradigms. Open-loop approaches [13], [14] predict keyframes or short-horizon actions without feedback, which makes them scalable across tasks but prone to compounding errors. Closed-loop visuomotor policies have become the dominant trend. Specialist models such as ACT [1], Diffusion Policy [2], DP3 [15], and 3D-Diffuser Actor [14] achieve strong task-specific performance but require retraining for new skills. In contrast, generalist Vision-Language-Action (VLA) models like OpenVLA [3] , π_0 [4], and recent extensions [16], [17], [18], [19] leverage large-scale pretraining to support multi-task and zero-shot control. This progression from open-loop to closed-loop, and from specialists to generalists, marks significant progress. However, both paradigms remain highly sensitive to viewpoint changes, which we address by building viewpoint-robust closed-loop control.

B. Generative Models for Robot Planning

With the recent advancements in generation models in terms of image quality, temporal consistency, and scene generalization, many works have begun to explore their potential in autonomous driving [20], [21], [22] and robotics [23]. In the realm of robot planning, some works directly employ generative models to produce action videos and then generate the corresponding actions based on the video outputs. These videos can serve as synthesized sub-goals to provide visual guidance for subsequent policy generation [24], [25], or be utilized to extract robot actions by leveraging inverse dynamics models [26], [27], [28] or pose tracking models [29], [8]. VISTA [30], on the other hand, leverages generative models to synthesize images from novel viewpoints for data augmentation, thereby enhancing the policy's robustness to variations in camera pose. Another line of research proposes unified models that simultaneously predict both future frames and robot actions [31], [32], [19], handling both scene understanding and action generation within a single framework. Some approaches leverage generative models as world simulators to support model-based reinforcement learning algorithms [33], [34] or to enable closed-loop verification of policies [35]. Rooted in end-to-end robotic manipulation frameworks, our approach aims to produce latent representations from varying observational viewpoints and learn action policies, thereby achieving generalizable manipulation across viewpoints.

C. View-Robust Perception for Robotics

A long-standing challenge in robotic manipulation is the lack of robustness to camera viewpoint changes. methods [36], [37], [5], [38], [7], [39] , which synthesize unseen views from limited observations, provide robust perception for robot manipulation. Existing NVS-based approaches generally fall into two categories: reconstruction-based and generation-based. Reconstruction-based methods explicitly recover 3D structure from synchronized multi-view data. NeRF- and 3DGS-based methods [5], [38] represent the

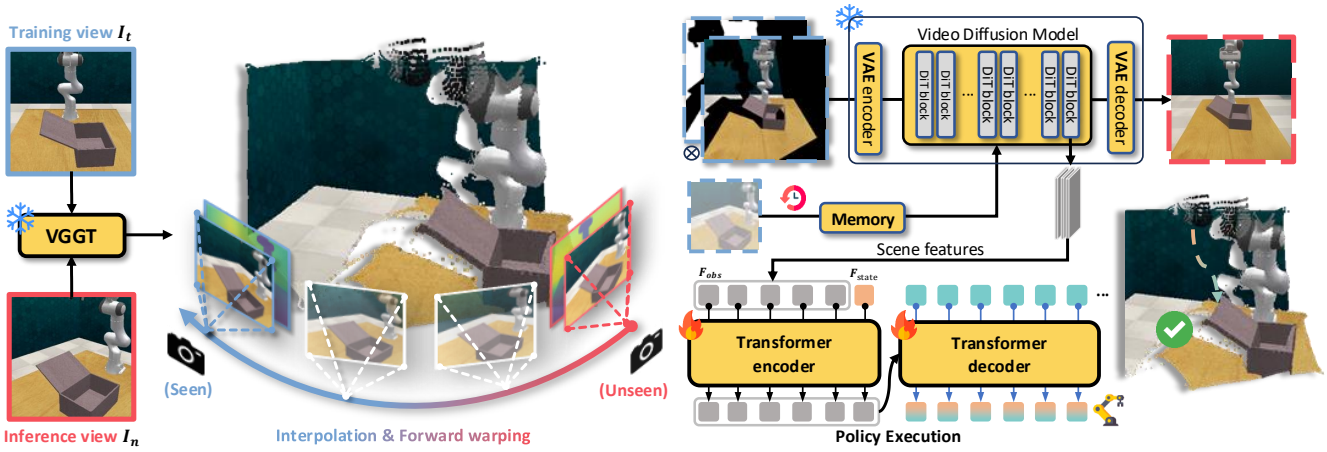

Fig. 2: Architecture of **VistaBot**. (1) 4D Geometry Estimation with VGGT for pose and depth prediction; (2) View Synthesis via a video diffusion model with memory to generate spatiotemporal-consistent latent features; (3) Policy Execution using a Transformer that fuses scene and robot state features for closed-loop manipulation under unseen views.

scene as a volume or a 3D Gaussian field and can synthesize high-fidelity novel views. While effective as simulators for offline robot learning, these methods are impractical for closed-loop robotic manipulation since real-time scene updates of this 3D representation are time-consuming. Generation-based methods [7], [39] leverage image or video generative models to synthesize novel views for downstream policies. Although they can generate plausible observations, they often struggle with precise camera control since the generation process is not physics-aware. Unlike the methods above, our approach leverages feed-forward reconstruction to obtain geometric priors and a video generation model to produce viewpoint-aligned spatiotemporal representations, which directly support efficient closed-loop policy learning.

III. Method

In this section, we introduce VistaBot, a framework that combines geometric and video diffusion models for robust manipulation without test-time calibration. We first describe the problem setting in Section III-A. Then, we present the three key components of our approach: 4D geometry estimation, view-synthesis latent extraction, and latent action learning, detailed in Section III-B, Section III-C, and Section III-D, respectively.

A. Problem Formulation

To evaluate our proposed policy's generalization ability across different viewpoints, we propose the following task setting: the model is trained exclusively on RGB observations $\{I_t\}$ collected from a fixed camera pose T_t. During inference, the model is required to perform closed-loop manipulation tasks based on new observations $\{I_n^i\}$ captured from a set of novel camera viewpoints $T_n^1, T_n^2, \ldots, T_n^m$, which are distinct from the training viewpoint.

B. 4D Geometry Estimation

As illustrated in Fig. 2, given the initial observation of the training view I_t and an observation of the novel view I_n, where $I_t, I_n \in \mathbb{R}^{h\times w\times 3}$, we first utilize a feed-forward geometric model VGGT [40] to estimate the relative pose $\mathbf{T_{n\to t}}$ between I_n and I_t, along with the depth map D_n of I_n. It is noteworthy that since the observation I_n may represent any arbitrary frame during closed-loop operation, directly applying a feed-forward geometric model cannot guarantee temporal consistency. Moreover, existing feed-forward geometric models often underperform on category-specific objects in embodied scenarios, such as the robot arm and gripper. To address these limitations, we collect a modest amount of simulated and real-world data to fine-tune the model, enabling consistent 4D geometric estimation in embodied settings.

Next, we lift I_n to obtain its point cloud $P_n \in \mathbb{R}^{hw\times 6}$ with the estimated depth D_n and the camera intrinsic matrix $\mathbf{K} \in \mathbb{R}^{3\times 3}$. Then the point cloud is transformed to the training view using the relative pose $\mathbf{T}$ estimated by the feed-forward geometric model, yielding P_t:

$$P_t = T_{n\to t} \cdot \text{proj}^{-1}(I_n, D_n, \mathbf{K}), \tag{1}$$

where $\text{proj}^{-1}(\cdot,\cdot,\cdot)$ denotes the inverse perspective projection.

Finally, we render P_t into the training view, yielding an image I_r:

$$I_r = \text{proj}(P_t, \mathbf{K}), \tag{2}$$

where $\text{proj}(\cdot,\cdot)$ represents the perspective projection.

C. Spatial-temporal Latent Feature Extraction

In this section, our goal is to synthesize the source training view from the point cloud rendering image I_r using a video diffusion model. To generate higher-quality source view images and better adapt the video diffusion model for closed-loop robotic control, we design a spatiotemporal conditional strategy, as illustrated in Figure 2. Finally, we extract the corresponding spatiotemporal latent features from the video diffusion model as scene representations for the policy.

Point Rendering Inpainting. Due to scene occlusions and viewpoint differences, the point cloud rendering I_r contains noticeable holes. We use the projection mask M_r during the point cloud projection to mark the missing regions. We then use I_r and M_r as conditions to guide a conditional video diffusion model in generating high-quality images at the target training viewpoint. We employ a video diffusion model to inpaint the masked regions.

Spatial Viewpoint Interpolation. Image-based inpainting models can fill masked regions with an image and a mask. In our case, however, they attempt to fill large holes caused by substantial viewpoint changes. This process often introduces artifacts and blurred details in the inpainted regions.

To help the generation model capture spatial viewpoint changes and better exploit the contextual information of the video diffusion model, we interpolate camera poses and perform frame-by-frame point cloud rendering to obtain multi-frame interpolated images between the novel and original viewpoints, thereby ensuring a smooth transition during novel-view generation:

$$\mathbf{T} = t\,\mathbf{T_n} + (1-t)\,\mathbf{T_t}. \tag{3}$$

where $\mathbf{T_n}$ denotes the camera pose of the novel view, and $\mathbf{T_t}$ denotes the source view of the generation target, t is interpolation parameter.

Notably, we adopt CogVideoX [41] as the backbone video diffusion model. It consists of a 3D VAE for video latent compression and a DiT for diffusion denoising.

Specifically, the interpolated images I_r and corresponding masks M are encoded via the VAE and flattened to produce spatial tokens that preserve viewpoint geometry.

Temporal Memory Reference. To maintain temporal consistency of policy observations during closed-loop inference, we introduce a memory module as the temporal condition, where historical references from previous steps are incorporated into subsequent generations. We cache the observation view frame at the last inference and encode it using the same 3D VAE encoder above.

Inspired by [42], we adopt a DiT block with cross-attention: spatial tokens serve as queries, while temporal tokens act as keys and values. This design integrates temporal information into the spatial context, yielding 4D-consistent conditions suitable for closed-loop operation.

D. Closed-loop Action Learning

Instead of directly using the generated source view as the policy input, we use the synthesis latent as our policy's input. Inspired by Lexicon3D [43], video diffusion models (VDM) have demonstrated remarkable performance in object-level and geometric understanding tasks, which is crucial for learning cross-view consistent representations. Moreover, compared to explicitly using decoded synthesized images, learning action policies directly in the latent space of the VDM is more efficient, since it omits the decoding of the VAE and the vision encoding of policy.

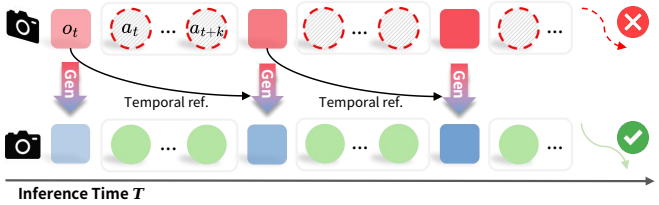

Fig. 3: Closed-loop manipulation during inference. Top: unseen-view observations cause action drift and task failure. Bottom: VistaBot combines unseen-view observations with historical references to generate the training (seen) view, enabling consistent action prediction and successful task execution. "Gen" refers to our view synthesis process. "o" and "a" refer to observation and action, respectively.

Specifically, we extract scene features from the final DiT block of the diffusion model after adaptive layer normalization. This representation captures high-level spatiotemporal semantics while preserving generation-relevant structure, making it well suited for visuomotor policy learning.

Following ACT [1], we replace the scene features originally extracted by ResNet-50 [44] with the scene features from the diffusion model $\mathbf{F}_{\text{obs}}$, while retaining the original robot state features $\mathbf{F}_{\text{state}}$ and action chunks.

IV. Experiments

In this section, we first describe our experimental setup and introduce the generalization score metric, which is used to evaluate the policy's ability to generalize across view variations. We then assess the effectiveness of our method through both simulation and real-world experiments. Finally, we conduct an ablation study to validate the key design, followed by a discussion. Collectively, these experiments demonstrate the effectiveness of our method in enhancing view generalization in robotic manipulation policies.

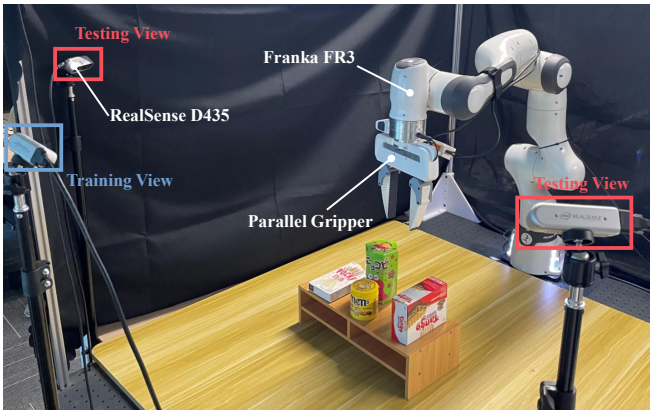

Fig. 4: Real-world setup with a Franka FR3 arm and three RealSense D435 cameras. The front camera provides training observations, while the side cameras serve as unseen testing views to evaluate cross-view generalization.

A. Experimental Setup

To ensure the reproducibility and benchmarking of our experiments, we first conducted the experiments in RLBench [45] simulated tasks. Subsequently, we also set up

TABLE I: **Quantitative evaluation on RLBench.** Success rates across 8 tasks under different inference views. Green cell color indicates better performance, while red indicates the opposite.

Base Policy	Inference		Tasks								Metrics	
	Setting	**Angle**	close box	close laptop	meat on grill	open box	push buttons	stack wine	take lid off	toilet seat down	**Avg. S.R.**	**Avg. VGS**
ACT [1]	–	0°	0.84	0.60	0.84	0.96	0.64	0.80	0.96	1.00	0.83	–
	Default	-45°	0.20	0.20	0.00	0.08	0.08	0.00	0.32	0.40	0.16	0.24
		-30°	0.32	0.20	0.00	0.32	0.32	0.08	0.40	0.68	0.28	
		+30°	0.56	0.20	0.04	0.08	0.08	0.04	0.32	0.64	0.25	
		+45°	0.28	0.16	0.00	0.08	0.08	0.00	0.08	0.32	0.13	
	Ours	-45°	0.40	0.36	0.40	0.32	0.52	0.12	0.32	0.92	0.42	**0.67**
		-30°	0.76	0.44	0.40	0.72	0.64	0.52	0.64	1.00	0.64	
		+30°	0.60	0.44	0.52	0.76	0.68	0.48	0.68	0.92	0.64	
		+45°	0.56	0.40	0.32	0.28	0.64	0.40	0.60	0.92	0.52	
π_0 [4]	–	0°	0.96	0.84	0.84	0.96	0.68	0.56	0.92	0.92	0.84	–
	Default	-45°	0.24	0.24	0.04	0.28	0.16	0.00	0.32	0.52	0.23	0.33
		-30°	0.20	0.32	0.24	0.24	0.20	0.12	0.56	0.68	0.32	
		+30°	0.72	0.60	0.20	0.12	0.32	0.00	0.64	0.40	0.38	
		+45°	0.60	0.40	0.08	0.16	0.04	0.08	0.24	0.36	0.24	
	Ours	-45°	0.44	0.60	0.56	0.36	0.76	0.48	0.80	0.92	0.62	**0.87**
		-30°	0.76	0.60	0.68	0.52	0.72	0.76	0.80	1.00	0.73	
		+30°	0.92	0.56	0.76	0.92	0.80	0.48	0.72	0.96	0.76	
		+45°	0.76	0.72	0.72	0.88	0.76	0.52	0.84	0.84	0.76	

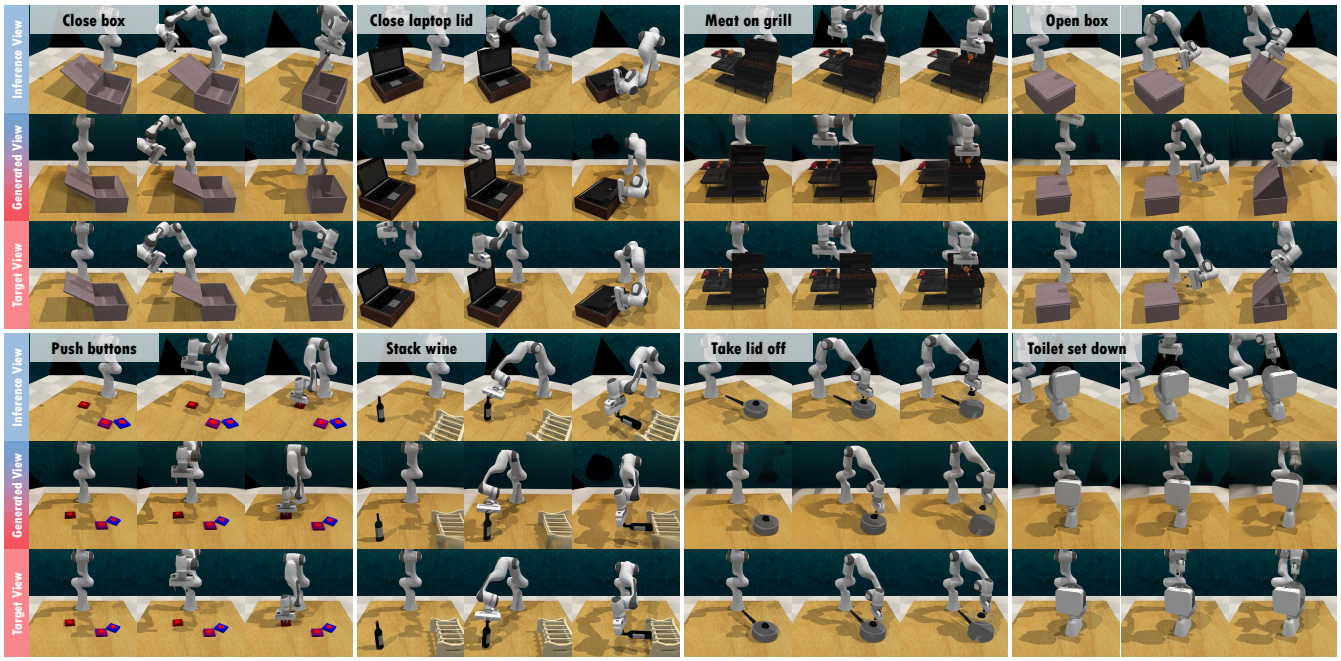

Fig. 5: **Qualitative results on RLBench.** For each task, VistaBot synthesizes the training viewpoint (middle row) from a novel inference perspective (top row), which is then compared against the ground-truth training viewpoint (bottom row).

a series of tabletop robotic manipulation tasks to evaluate the robustness of our method to view variations in real-world environments. In addition, we proposed a viewpoint generalization metric, *"View Generalization Score"*, to more comprehensively evaluate the robustness of manipulation policies to viewpoint shifts during testing.

Simulation. We select 8 tasks that are chosen to cover diverse and representative manipulation scenarios from the RLBench tasksuites [45], as shown in Fig. 5. Unlike previous approaches [46], [47] that regress the discrete keyframe action and manipulate in an open-loop manner, our method employ a close-loop visuomotor policy. During the training stage, we collected 50 demonstrations for each task, where the visual inputs consisted exclusively of RGB images from the front camera. In the testing phase, we used RGB observations from camera viewpoints at $-45°$, $-30°$, $30°$, and $45°$ as inputs, where the rotation angle is defined around the upright axis centered in the robot's manipulation space. For each testing viewpoint, we evaluated our policy 25 times.

Real Robot. As shown in Fig. 4, we constructed a real-world

tabletop robotic platform consisting of a 7-DoF Franka arm equipped with a parallel gripper and three D435 cameras. We selected 4 tasks to evaluate the robustness of our method in real-world scenarios. Similar to the simulation experiments, for each task, we collected 50 trajectories with the observation from the front camera for training. During testing, we evaluated our policy 25 times under each of the left and right camera (novel viewpoints). The whole pipeline predicts robot commands around 3 Hz.

Baselines. To validate the effectiveness of our method in improving cross-view generalization of manipulation policies and generating high-quality novel-view observations, we set up two groups of baseline comparisons: **(1) Base visuomotor policies:** We use ACT [1] and π_0 [4] as representative generalist and specialist manipulation models, respectively, as our base policies. **(2) Novel-view synthesis baselines:** We select the state-of-the-art novel-view synthesis methods, Anysplat [48] and LangScene-X [7], as baselines to evaluate the effectiveness of our approach in generating novel-view.

Evaluation Metrics. To quantitatively evaluate the robustness of our method in cross-view manipulation and the quality of novel-view observation generation, we design three groups of metrics:

(1) Average Success Rate (Avg. S.R): The average task success rate across all tasks under each testing viewpoint.

(2) View Generalization Score (VGS): We introduce a novel metric, View Generalization Score (VGS), to quantify the robustness of policy performance under viewpoint changes. Let $S(\theta)$ denote the success rate under viewpoint θ, and $S(\theta_0)$ the success rate under the baseline (training) viewpoint θ_0. Given N sampled viewpoints $\theta_i i = 1^N$ within the predefined range Θ, VGS is defined as:

$$\text{VGS} = \frac{1}{N}\sum_{i=1}^{N}\frac{S(\theta_i)}{S(\theta_0)}. \tag{4}$$

VGS measures the average relative performance across novel viewpoints compared to the baseline view, with values closer to 1 indicating stronger robustness to viewpoint variations.

(3) Image and video quality metrics: We adopt FVD, FID, SSIM, PSNR, and LPIPS to comprehensively evaluate the quality of generated novel-view observations.

B. Simulation Experiments

We first evaluate the impact of novel testing views on the base policies, as shown in Table I. Compared with the unchanged view (0°), testing on novel views (default rows) results in a substantial drop in closed-loop success rate, which is roughly proportional to the magnitude of the viewpoint shift. This trend is observed for both ACT and π_0. For example, the average success rate of ACT decreases from 0.80 to 0.13 under a 45° view change, representing a reduction of approximately 84%.

The VGS score provides a measure of policy view generalizability. The baseline ACT achieves a VGS score of only 0.24, indicating weak view generalizability, while π_0 performs slightly better with a score of 0.33. By applying our method, both VGS scores and average task success rates improve significantly: the VGS score rises from 0.24 to 0.67 for ACT and from 0.33 to 0.87 for π_0, demonstrating robust view generalizability under camera viewpoint changes.

We further compare the visual quality of our method with other novel-view synthesis approaches. For reconstruction-based methods, we select the feed-forward approach AnySplat, and for generation-based methods, we select LangScene-X. For AnySplat, we use four views ($-45°$, $-30°$, 30°, and 45°) as inputs to reconstruct the 3D Gaussian and render it from the 0° viewpoint. For LangScene-X, we use $-45°$ and 45° as the first and last frames, then perform video generation to produce an interpolated video; the center frame of this video is used for evaluation.

The generated novel views are shown in Fig. 6. Even when provided with more input views (four in total), AnySplat fails to fill background holes. The limited input also leads to floating artifacts and blurred edges on the robot arm. On the other hand, LangScene-X struggles to accurately estimate the relative poses of the first and last frames, resulting in less precise viewpoints in the target view and often producing "melted" frames that blend the two inputs.

Our method, by contrast, combines the strengths of both reconstruction and generation. Geometric priors and target camera poses are obtained from VGGT, while unseen regions are filled using the video diffusion model, resulting in plausible visual quality for novel views.

TABLE II: Quantitative results of view synthesis.

Method	FVD↓	FID ↓	SSIM ↑	PSNR ↑	LPIPS ↓
AnySplat [48]	825.83	102.71	0.27	12.07	0.23
LangScene-X [7]	551.91	118.34	0.44	15.02	0.17
Ours	**471.04**	**69.56**	**0.59**	**18.34**	**0.09**

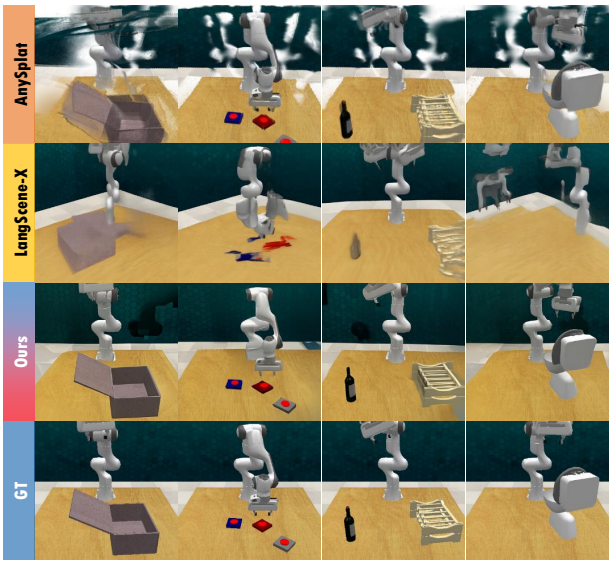

Fig. 6: Unseen-to-seen view synthesis comparison. **VistaBot** (Ours) generates sharper and more consistent results than AnySplat and LangScene-X, closely matching the ground truth (GT).

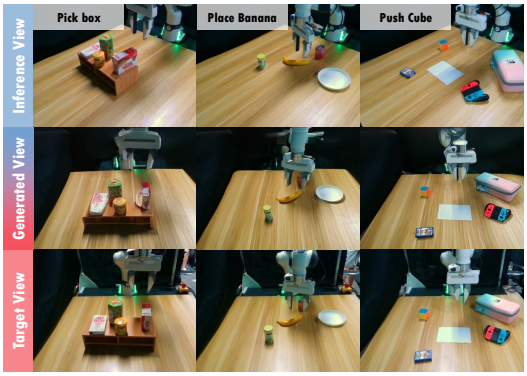

Fig. 7: Qualitative results on Real-robot experiments.

C. Real-World Experiments

Fig. 7 presents the results of our real-world experiments, including three distinct tasks. The top row displays the viewpoint in test time, the bottom row corresponds to the training viewpoint, and the middle row showcases the training viewpoint generated by our proposed method. These qualitative results demonstrate the strong novel view synthesis capability of our method in real-world environments. Additionally, the quantitative results are shown in the Table III. It can be seen that our method exhibits strong cross-view generalization ability in real robot manipulation experiments.

TABLE III: **Quantitative evaluation on real robot.** Average Success rates across 4 tasks under different inference view settings.

Base Policy	Inference		Tasks				Metrics	
	Setting	Angle	box on shelf	place banana	push cube	unplug charger	Avg. S.R.	Avg. VGS
ACT [1]	–	0°	0.80	0.88	0.56	0.92	0.79	–
	Default	-45°	0.12	0.20	0.04	0.24	0.15	0.21
		+45°	0.24	0.16	0.08	0.24	0.18	
	Ours	-45°	0.32	0.60	0.48	0.72	0.53	**0.72**
		+45°	0.64	0.64	0.48	0.68	0.61	
π_0 [4]	–	0°	0.92	0.92	0.64	1.00	0.87	–
	Default	-45°	0.16	0.28	0.12	0.32	0.22	0.27
		+45°	0.32	0.28	0.16	0.36	0.28	
	Ours	-45°	0.40	0.80	0.52	0.88	0.65	**0.79**
		+45°	0.68	0.76	0.56	0.88	0.72	

D. More Discussion

Ablation Study. Table IV presents the results of our ablation study. Compared with Row 3, our approach using estimated 4D geometry closely approximates the performance achieved with ground-truth (GT) geometry. In contrast to Row 4, the results underscore the contribution of the memory module to effective closed-loop control.

TABLE IV: **Ablation study.**

Method	VGS ↑
ACT	0.24
VistaBot (ours)	0.67
w/ GT Depth and extr.	**0.79**
w/o Memory	0.48

Estimation of View Generalizability. We now examine in detail what happens when testing under view disturbances, why this degrades policy performance, and how our method alleviates this issue. Since policy learning typically relies on a visual encoder (e.g., ResNet in ACT or ViT in π_0) to capture environment observations, we encode the source training view, novel testing view, and the generated view from our method using a pretrained InceptionV3 across the dataset. We then apply PCA to reduce dimensionality, and the resulting features are shown in Fig. 8.

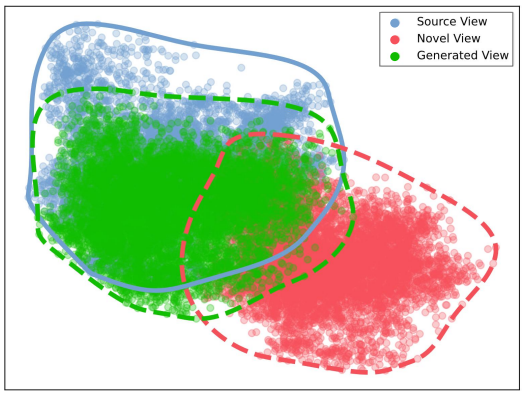

Fig. 8: Visualization of feature distributions from different viewpoints in the principal component space. Novel views (Red) diverge from the source view (Blue), while **VistaBot**-generated views (Green) align closely with the source, highlighting improved cross-view generalization.

It is evident that when a novel view is used during testing, the feature distribution shifts significantly from the source training view. This distributional shift induces an out-of-distribution (OOD) problem for the action policy, causing the estimated actions to deviate from the correct task trajectory. Such deviations further exacerbate the OOD issue, ultimately leading to failure in closed-loop inference. In contrast, when using our generation model, the distribution of the warped observation features remains much closer to the source training view. As a result, the action head produces outputs that are more consistent with the trained policy, thereby mitigating task failure and preserving task completion capability.

In summary, the key to view generalizability in image-based imitation learning lies in maintaining stable visual representations under view changes. Our proposed method achieves this by producing feature distributions that align more closely with the source training view, thereby enhancing view generalizability.

V. Conclusion

In this paper, we present VistaBot, a novel framework for view-robust robot manipulation that integrates feed-forward geometry and video diffusion models. Without requiring camera poses, our method achieves spatiotemporally consistent 4D representation learning and closed-loop action inference in novel views. We introduced a view generalization score (VGS) for systematic evaluation and showed significant improvements over strong baselines like ACT and π_0 in both simulated and real-world environments.

Limitation. Although VistaBot enables the synthesis of novel views for robotic manipulation, it has certain limitations. The model struggles to generate high-quality synthesized views under severe occlusions.

References

[1] T. Z. Zhao, V. Kumar, S. Levine, and C. Finn, "Learning fine-grained bimanual manipulation with low-cost hardware," *arXiv:2304.13705*, 2023.

[2] C. Chi, Z. Xu, S. Feng, E. Cousineau, Y. Du, B. Burchfiel, R. Tedrake, and S. Song, "Diffusion policy: Visuomotor policy learning via action diffusion," *The International Journal of Robotics Research*, 2023.

[3] M. J. Kim, K. Pertsch, S. Karamcheti, T. Xiao, A. Balakrishna, S. Nair, R. Rafailov, E. Foster, G. Lam, P. Sanketi, *et al.*, "Openvla: An open-source vision-language-action model," *arXiv:2406.09246*, 2024.

[4] K. Black, N. Brown, D. Driess, A. Esmail, M. Equi, C. Finn, N. Fusai, L. Groom, K. Hausman, B. Ichter, *et al.*, "pi_0: A vision-language-action flow model for general robot control," *arXiv:2410.24164*, 2024.

[5] Y. Zheng, X. Chen, Y. Zheng, S. Gu, R. Yang, B. Jin, P. Li, C. Zhong, Z. Wang, L. Liu, *et al.*, "Gaussiangrasper: 3d language gaussian splatting for open-vocabulary robotic grasping," *IEEE Robotics and Automation Letters*, 2024.

[6] O. Shorinwa, J. Tucker, A. Smith, A. Swann, T. Chen, R. Firoozi, M. Kennedy III, and M. Schwager, "Splat-mover: Multi-stage, open-vocabulary robotic manipulation via editable gaussian splatting," *arXiv:2405.04378*, 2024.

[7] F. Liu, H. Li, J. Chi, H. Wang, M. Yang, F. Wang, and Y. Duan, "Langscene-x: Reconstruct generalizable 3d language-embedded scenes with trimap video diffusion," *arXiv:2507.02813*, 2025.

[8] Z. Liu, S. Li, E. Cousineau, S. Feng, B. Burchfiel, and S. Song, "Geometry-aware 4d video generation for robot manipulation," *arXiv:2507.01099*, 2025.

[9] H. Touvron, T. Lavril, G. Izacard, X. Martinet, M.-A. Lachaux, T. Lacroix, B. Rozière, N. Goyal, E. Hambro, F. Azhar, *et al.*, "Llama: Open and efficient foundation language models," *arXiv:2302.13971*, 2023.

[10] M. Oquab, T. Darcet, T. Moutakanni, H. Vo, M. Szafraniec, V. Khalidov, P. Fernandez, D. Haziza, F. Massa, A. El-Nouby, *et al.*, "Dinov2: Learning robust visual features without supervision," *arXiv:2304.07193*, 2023.

[11] A. O'Neill, A. Rehman, A. Maddukuri, A. Gupta, A. Padalkar, A. Lee, A. Pooley, A. Gupta, A. Mandlekar, A. Jain, *et al.*, "Open x-embodiment: Robotic learning datasets and rt-x models: Open x-embodiment collaboration 0," in *ICRA*, 2024.

[12] H.-S. Fang, H. Fang, Z. Tang, J. Liu, C. Wang, J. Wang, H. Zhu, and C. Lu, "Rh20t: A comprehensive robotic dataset for learning diverse skills in one-shot," *arXiv:2307.00595*, 2023.

[13] A. Jaegle, F. Gimeno, A. Brock, O. Vinyals, A. Zisserman, and J. Carreira, "Perceiver: General perception with iterative attention," in *ICML*, 2021.

[14] T.-W. Ke, N. Gkanatsios, and K. Fragkiadaki, "3d diffuser actor: Policy diffusion with 3d scene representations," *arXiv:2402.10885*, 2024.

[15] Y. Ze, G. Zhang, K. Zhang, C. Hu, M. Wang, and H. Xu, "3d diffusion policy: Generalizable visuomotor policy learning via simple 3d representations," *arXiv:2403.03954*, 2024.

[16] J. Wen, Y. Zhu, J. Li, Z. Tang, C. Shen, and F. Feng, "Dexvla: Vision-language model with plug-in diffusion expert for general robot control," *arXiv:2502.05855*, 2025.

[17] J. Liu, H. Chen, P. An, Z. Liu, R. Zhang, C. Gu, X. Li, Z. Guo, S. Chen, M. Liu, *et al.*, "Hybridvla: Collaborative diffusion and autoregression in a unified vision-language-action model," *arXiv:2503.10631*, 2025.

[18] D. Qu, H. Song, Q. Chen, Y. Yao, X. Ye, Y. Ding, Z. Wang, J. Gu, B. Zhao, D. Wang, *et al.*, "Spatialvla: Exploring spatial representations for visual-language-action model," *arXiv:2501.15830*, 2025.

[19] J. Cen, C. Yu, H. Yuan, Y. Jiang, S. Huang, J. Guo, X. Li, Y. Song, H. Luo, F. Wang, *et al.*, "Worldvla: Towards autoregressive action world model," *arXiv:2506.21539*, 2025.

[20] L. Russell, A. Hu, L. Bertoni, G. Fedoseev, J. Shotton, E. Arani, and G. Corrado, "Gaia-2: A controllable multi-view generative world model for autonomous driving," *arXiv:2503.20523*, 2025.

[21] R. Gao, K. Chen, E. Xie, L. Hong, Z. Li, D.-Y. Yeung, and Q. Xu, "Magicdrive: Street view generation with diverse 3d geometry control," *arXiv:2310.02601*, 2023.

[22] Y. Wang, J. He, L. Fan, H. Li, Y. Chen, and Z. Zhang, "Driving into the future: Multiview visual forecasting and planning with world model for autonomous driving," in *CVPR*, 2024.

[23] N. Agarwal, A. Ali, M. Bala, Y. Balaji, E. Barker, T. Cai, P. Chattopadhyay, Y. Chen, Y. Cui, Y. Ding, *et al.*, "Cosmos world foundation model platform for physical ai," *arXiv:2501.03575*, 2025.

[24] Q. Bu, J. Zeng, L. Chen, Y. Yang, G. Zhou, J. Yan, P. Luo, H. Cui, Y. Ma, and H. Li, "Closed-loop visuomotor control with generative expectation for robotic manipulation," *NeurIPS*, 2024.

[25] J. Cen, C. Wu, X. Liu, S. Yin, Y. Pei, J. Yang, Q. Chen, N. Duan, and J. Zhang, "Using left and right brains together: Towards vision and language planning," *arXiv:2402.10534*, 2024.

[26] Y. Du, S. Yang, B. Dai, H. Dai, O. Nachum, J. Tenenbaum, D. Schuurmans, and P. Abbeel, "Learning universal policies via text-guided video generation," *NeurIPS*, 2023.

[27] A. Ajay, S. Han, Y. Du, S. Li, A. Gupta, T. Jaakkola, J. Tenenbaum, L. Kaelbling, A. Srivastava, and P. Agrawal, "Compositional foundation models for hierarchical planning," *NeurIPS*, 2023.

[28] Y. Tian, S. Yang, J. Zeng, P. Wang, D. Lin, H. Dong, and J. Pang, "Predictive inverse dynamics models are scalable learners for robotic manipulation," *arXiv:2412.15109*, 2024.

[29] J. Liang, R. Liu, E. Ozguroglu, S. Sudhakar, A. Dave, P. Tokmakov, S. Song, and C. Vondrick, "Dreamitate: Real-world visuomotor policy learning via video generation," *arXiv:2406.16862*, 2024.

[30] 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," *arXiv:2409.03685*, 2024.

[31] C. Zhu, R. Yu, S. Feng, B. Burchfiel, P. Shah, and A. Gupta, "Unified world models: Coupling video and action diffusion for pretraining on large robotic datasets," *arXiv:2504.02792*, 2025.

[32] S. Li, Y. Gao, D. Sadigh, and S. Song, "Unified video action model," *arXiv:2503.00200*, 2025.

[33] J. Wu, S. Yin, N. Feng, X. He, D. Li, J. Hao, and M. Long, "ivideogpt: Interactive videogpts are scalable world models," *NeurIPS*, 2024.

[34] A. L. Chandra, I. Nematollahi, C. Huang, T. Welschehold, W. Burgard, and A. Valada, "Diwa: Diffusion policy adaptation with world models," *arXiv:2508.03645*, 2025.

[35] Y. Liao, P. Zhou, S. Huang, D. Yang, S. Chen, Y. Jiang, Y. Hu, J. Cai, S. Liu, J. Luo, *et al.*, "Genie envisioner: A unified world foundation platform for robotic manipulation," *arXiv:2508.05635*, 2025.

[36] B. Kerbl, G. Kopanas, T. Leimkühler, and G. Drettakis, "3d gaussian splatting for real-time radiance field rendering." *ACM Trans. Graph.*, vol. 42, no. 4, pp. 139–1, 2023.

[37] B. Mildenhall, P. P. Srinivasan, M. Tancik, J. T. Barron, R. Ramamoorthi, and R. Ng, "Nerf: Representing scenes as neural radiance fields for view synthesis," *Communications of the ACM*, vol. 65, no. 1, pp. 99–106, 2021.

[38] G. Lu, S. Zhang, Z. Wang, C. Liu, J. Lu, and Y. Tang, "Manigaussian: Dynamic gaussian splatting for multi-task robotic manipulation," in *European Conference on Computer Vision*. Springer, 2024, pp. 349–366.

[39] J. Seo, K. Fukuda, T. Shibuya, T. Narihira, N. Murata, S. Hu, C.-H. Lai, S. Kim, and Y. Mitsufuji, "Genwarp: Single image to novel views with semantic-preserving generative warping," *Advances in Neural Information Processing Systems*, vol. 37, pp. 80 220–80 243, 2024.

[40] J. Wang, M. Chen, N. Karaev, A. Vedaldi, C. Rupprecht, and D. Novotny, "Vggt: Visual geometry grounded transformer," in *CVPR*, 2025.

[41] Z. Yang, J. Teng, W. Zheng, M. Ding, S. Huang, J. Xu, Y. Yang, W. Hong, X. Zhang, G. Feng, *et al.*, "Cogvideox: Text-to-video diffusion models with an expert transformer," *arXiv preprint arXiv:2408.06072*, 2024.

[42] M. YU, W. Hu, J. Xing, and Y. Shan, "Trajectorycrafter: Redirecting camera trajectory for monocular videos via diffusion models," *arXiv preprint arXiv:2503.05638*, 2025.

[43] Y. Man, S. Zheng, Z. Bao, M. Hebert, L.-Y. Gui, and Y.-X. Wang, "Lexicon3d: Probing visual foundation models for complex 3d scene understanding," in *NeurIPS*, 2024.

[44] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," *CVPR*, 2015.

[45] S. James, Z. Ma, D. R. Arrojo, and A. J. Davison, "Rlbench: The robot learning benchmark & learning environment," *IEEE Robotics and Automation Letters*, 2020.

[46] M. Shridhar, L. Manuelli, and D. Fox, "Perceiver-actor: A multi-task transformer for robotic manipulation," in *Conference on Robot Learning*. PMLR, 2023, pp. 785–799.

[47] Y. Ze, G. Yan, Y.-H. Wu, A. Macaluso, Y. Ge, J. Ye, N. Hansen, L. E. Li, and X. Wang, "Gnfactor: Multi-task real robot learning with generalizable neural feature fields," in *Conference on robot learning*. PMLR, 2023, pp. 284–301.

[48] L. Jiang, Y. Mao, L. Xu, T. Lu, K. Ren, Y. Jin, X. Xu, M. Yu, J. Pang, F. Zhao, *et al.*, "Anysplat: Feed-forward 3d gaussian splatting from unconstrained views," *arXiv preprint arXiv:2505.23716*, 2025.