

Video2Act: A Dual-System Video Diffusion Policy with Robotic Spatio-Motional Modeling

Yueru Jia^{1,2*}, Jiaming Liu^{1,2†}, Shengbang Liu^{3*}, Rui Zhou⁴, Wanhe Yu¹, Yuyang Yan¹, Xiaowei Chi⁵, Yandong Guo², Boxin Shi¹, Shanghang Zhang^{1✉}

¹State Key Laboratory of Multimedia Information Processing, School of Computer Science, Peking University; ²AI²Robotics; ³Sun Yat-sen University;

⁴Wuhan University; ⁵Hong Kong University of Science and Technology

*Equal Contribution, †Project lead, ✉Corresponding Author

Project Page: <https://video2act.github.io/>

Abstract

Robust perception and dynamics modeling are fundamental to real-world robotic policy learning. Recent methods employ video diffusion models (VDMs) to enhance robotic policies, improving their understanding and modeling of the physical world. However, existing approaches overlook the coherent and physically consistent motion representations inherently encoded across frames in VDMs. To this end, we propose Video2Act, a framework that efficiently guides robotic action learning by explicitly integrating spatial and motion-aware representations. Building on the inherent representations of VDMs, we extract foreground boundaries and inter-frame motion variations while filtering out background noise and task-irrelevant biases. These refined representations are then used as additional conditioning inputs to a diffusion transformer (DiT) action head, enabling it to reason about what to manipulate and how to move. To mitigate inference inefficiency, we propose an asynchronous dual-system design, where the VDM functions as the slow System 2 and the DiT head as the fast System 1, working collaboratively to generate adaptive actions. By providing motion-aware conditions to System 1, Video2Act maintains stable manipulation even with low-frequency updates from the VDM. For evaluation, Video2Act surpasses previous state-of-the-art VLA methods by 7.7% in simulation and 21.7% in real-world tasks in terms of average success rate, further exhibiting strong generalization capabilities.

1. Introduction

Robot learning aims to acquire action policies through interaction with the environment, leveraging dynamic visual

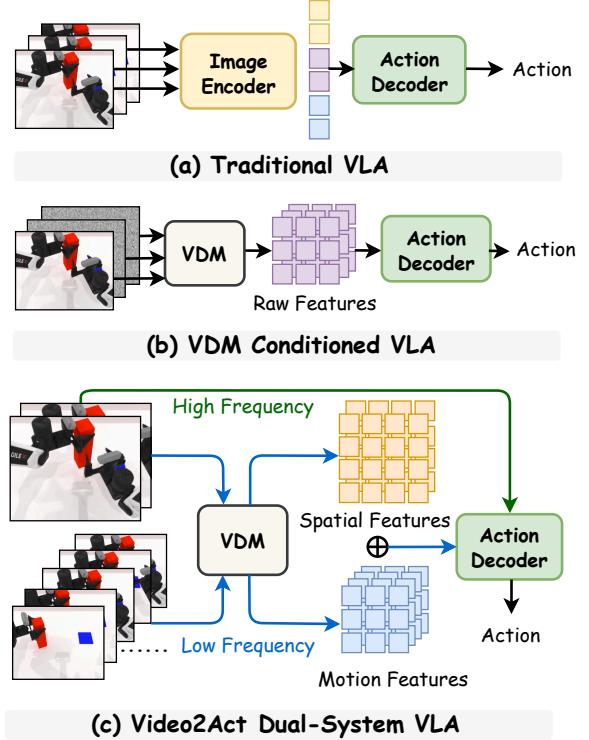


Figure 1. Overview of Video2Act. Unlike the static image–token concatenation used in traditional VLA models (a) or the direct VDM feature conditioning approach (b), our asynchronous dual-system model (c) employs a slow-system perceptual VDM to explicitly extract spatial and motion representations, while a fast-system action decoder performs high-frequency and stable robot control.

observations to guide decision-making [25, 31, 39]. From a cognitive perspective, humans acquire manipulation skills

not only by understanding the spatial structure of manipulated objects but also by abstracting transferable motion patterns across tasks [30]. For example, when closing a laptop lid, humans rely on understanding the spatial relationship between the screen and the hinge, while abstracting a motion sequence that involves first grasping the screen and then closing it. This insight suggests that effective robotic policy learning requires representations expressly designed for robust perception and dynamics modeling.

A straightforward approach is to concatenate visual tokens extracted from a static visual encoder, but this merely allows the model to observe multiple frames without enabling it to understand the temporal and causal dependencies among them. The recent success of video diffusion models (VDMs) [20] in capturing realistic dynamics demonstrates their strong ability to model physical environments. Inspired by this, recent advances have incorporated representations derived from VDMs to enhance robotic policy learning, enabling a richer understanding of scene context and future evolution. However, most existing policies [12, 22] do not explicitly explore the spatial structures and motion dynamics inherently encoded in the raw representations of VDMs, which could serve as more efficient and informative conditions for action learning.

To investigate the spatio-motion information encoded within them, we conduct a systematic analysis of the latent features from VDM [43]. Following the inversion technique applied to real videos [45, 50], we extract clean feature signals from the early inversion stages to suppress noise interference. We design two robotic validation scenarios: (1) capturing manipulation trajectories with a static third-person camera, and (2) observing manipulated objects using a dynamic wrist camera. The inferred features from both settings are visualized with Grad-CAM [48]. As shown in Figure 2, compared with static vision encoders (e.g., SigLIP [54] and DINOv2 [42]), VDM exhibits a stronger ability to capture the structures and motion consistency of foreground objects, mitigating the effects of robot and viewpoint motion.

Building on these findings, we introduce **Video2Act**, a vision-language-action (VLA) model that explicitly integrates refined **spatial- and motion-aware representations** from a VDM into policy learning. For spatial cues, we innovatively apply Spatial Filtering Operators to high-resolution images to extract salient foreground boundaries while filtering out background noise and task-irrelevant biases. For motion cues, we leverage the Fast Fourier Transform (FFT) on long-frame sequences to highlight the motion dynamics of both the robotic arm and the manipulated objects. Both representations are injected into the DiT action head via cross-attention conditioning, enabling the model to understand *what* to manipulate and *how* to move. Given the computational cost of the VDM, we design an asynchronous dual-system strategy. In this framework, the VDM functions

as a slow perceptual module (System 2) that provides spatio-motion conditions, while the DiT action head serves as a fast execution module (System 1), receiving real-time visual inputs and low-frequency features from System 2. Notably, we find that by explicitly providing motion-aware conditions to System 1, it can adaptively generate stable actions when receiving varying frequency inputs.

For evaluation, we assess the multi-task performance of Video2Act on the RoboTwin simulation benchmark [10, 40] and six real-world manipulation tasks using the ALOHA dual-arm robot [56]. Video2Act achieves state-of-the-art performance, outperforming previous methods by 7.7% in simulation and 21.7% in real-world experiments in terms of average success rate. Moreover, it operates at a real-time action generation frequency, demonstrating both high efficiency and robust control capability.

In summary, our contributions are as follows:

- We systematically analyze VDM representations in robotic settings, revealing that they capture stable structural and motion-consistent features that are robust to robot arm and wrist-camera dynamics.
- We propose Video2Act, which explicitly integrates spatio-motion representations through Spatial Filtering Operators and FFT into VLA model learning, enabling the model to understand what to manipulate and how to move.
- We develop an asynchronous dual-system strategy, where the VDM acts as a slow perceptual system and the DiT head as a fast execution system, enabling adaptive and stable action generation under high-frequency inputs.

2. Related Work

2.1. Video Diffusion Models for Robotics

Recent progress in video diffusion models (VDMs) has significantly advanced the ability to generate temporally consistent video sequences [20]. VDMs learn to model long-horizon dynamics through iterative denoising in the latent space, enabling strong representations of object motion, scene transitions, and causal dependencies across time. These capabilities have led to growing interest in using VDMs for downstream robotic perception and control [13, 16, 21, 24, 34, 53, 57]. Early works use VDMs to predict future videos and employ inverse dynamics models to infer actions [2, 6, 28, 38], but these approaches often suffer from accumulated errors and high inference latency. More recent work explores leveraging VDMs in an implicit manner; for example, VPP [22] learns to extract intermediate feature maps from VDMs and inject them into policy networks to end-to-end learn action policies. Despite these advances, existing approaches typically directly use raw VDM features without fully understanding or disentangling the spatial structures and motion dynamics encoded within them [12, 32, 34]. Such structural and temporal information

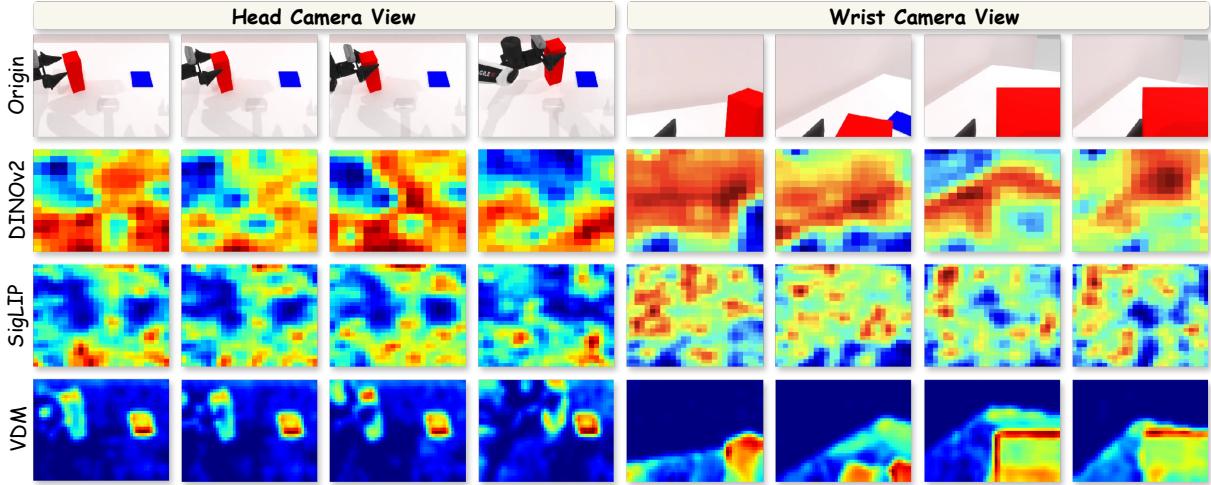


Figure 2. Qualitative analysis of latent representations. We visualize Grad-CAM activations for DINOv2, SigLIP and the Video Diffusion Model (VDM) during the block handover task, observed from two common robotic settings: a static third-person (Head Camera View) and a dynamic ego-centric (Wrist Camera View). The heatmaps for standard image encoders (DINOv2, SigLIP) are diffuse, unstable, and shift focus irregularly. In contrast, the VDM features consistently attend to the foreground objects being manipulated, demonstrating strong spatial structure awareness even under severe ego-motion.

remains underutilized, even though it provides strong priors for action learning. Our work addresses this gap by explicitly extracting and refining spatial- and motion-aware representations, and integrating them as conditions for diffusion-based action policy learning.

2.2. Vision-Language-Action (VLA) Models

Recent progress in developing robust and generalizable manipulation policies has led to the emergence of Vision–Language–Action (VLA) models, which integrate visual observations, natural language instructions, and policy learning into a unified framework [4, 5, 26, 33, 44, 51, 52]. Most VLAs employ image encoders to extract visual representations that are fused with language embeddings to guide an action decoder [23, 36, 41], which limits their ability to capture long-horizon temporal dynamics and detailed spatial structure in complex environments. To reduce the computational overhead of large vision–language models, several recent approaches introduce a dual-system decomposition, separating high-level reasoning from low-level action generation [3, 7, 9, 15, 19, 55]. In these frameworks, System 2 typically performs semantic or task-level processing to produce a latent representation that conditions a diffusion-based System 1 action policy. In our work, the VDM is also computationally expensive, and its strong temporal modeling capability naturally fits the role of System 2 within such a dual-system design.

3. Method

To explore the inherent spatio-motional information encoded within video diffusion models (VDM), we first conduct a qualitative analysis of their latent feature representations under two common robotic observation settings: a static third-person camera view and a dynamic wrist-mounted camera view (Section 3.1). The results show that VDM activations consistently emphasize foreground objects and exhibit high temporal stability, revealing strong spatial structure and motion-aware characteristics. Motivated by this observation, we introduce the Video2Act model in Section 3.2, a vision-language-action (VLA) framework that explicitly incorporates spatial- and motion-aware representations extracted from the VDM. To support real-time control, we design an asynchronous dual-system scheme, presented in Section 3.3, where the VDM operates as System 2, providing perceptual updates, while a DiT-based action head serves as System 1, producing high-frequency action generation.

3.1. Motivation

Existing VLA methods [4, 26, 35–37] rely on the static image encoders of pretrained VLMs to extract visual features. However, such image encoders lack the temporal perception and dynamics modeling capabilities required for robotic manipulation. Recently, motivated by the strong ability of video diffusion models (VDMs) to capture realistic dynamics in the physical world, several VLA approaches [12, 22] have begun to directly incorporate raw representations derived from VDMs to improve robotic policy learning. However,

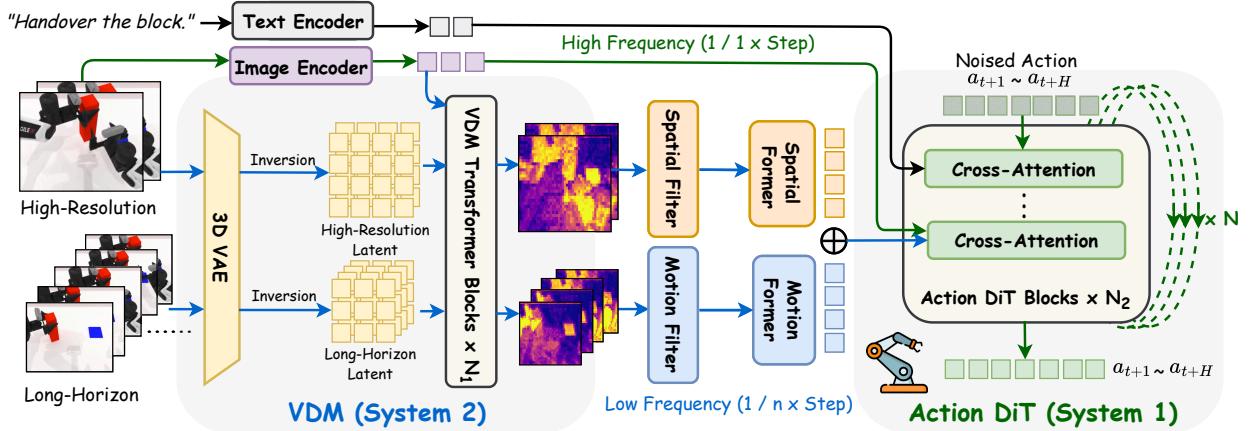


Figure 3. Video2Act Framework. Video2Act employs an asynchronous dual-system framework consisting of a slow perceptual VDM (System 2) and a fast action head (System 1). System 2 extracts refined spatial and motion representations from two image inputs: high-resolution images for spatial filtering via Sobel operators and long-horizon sequences for motion extraction via FFT. These low-frequency spatio-motional features serve as conditioning inputs to System 1, which simultaneously receives high-frequency image tokens. Through cross-attention conditioning, these asynchronously updated signals are effectively fused, enabling robust and real-time action generation.

why and how the latent features of VDMs benefit robotic manipulation has not been explicitly revealed. Therefore, we conduct an intuitive qualitative analysis of VDM latent representations under two common robotic observation settings: (1) a static scenario with a third-person camera and (2) a dynamic scenario with a wrist camera.

As shown in Figure 2, we visualize the Grad-CAM activations [48] of two widely used image encoders in VLA (SigLIP [54] and DINOv2 [42]), and compare them with those from the VDM [29]. We use the *block handover* task from RoboTwin [40] as the input sequence for all models to generate class-discriminative localization maps by weighting feature activations with their output gradients. The highlighted regions indicate spatial locations that contribute most to the model’s feature response, providing an attention-like visualization of where the model focuses. Under the static third-person view, image encoders exhibit unstable and inconsistent attention across consecutive frames, with their focus regions shifting irregularly as the manipulation progresses. In contrast, VDM features consistently attend to the foreground object over time. Even as the block is lifted by the gripper, the highlighted regions remain stably aligned with the manipulated object, demonstrating strong spatial structure representation even under dynamic scenarios. This consistency also persists under the dynamic wrist camera setting, where ego-motion is severe, yet the attention of VDM remains firmly anchored to the manipulated object.

3.2. Video2Act Framework

Building on these findings, we present Video2Act, a vision-language-action (VLA) framework that explicitly integrates spatial- and motion-aware representations from a video dif-

fusion transformer into policy learning.

3.2.1. Overview

Problem Definition. Following recent advances in VLA model imitation learning [26, 37], our objective is to learn a conditional policy that generates temporally coherent action trajectories conditioned on multimodal observations. Given the observations $o_t = (I_{t-T:t}, s_t)$ and the language instruction l , where I denotes the image sequence and s represents the robot state. The policy π_θ is trained to maximize the log-likelihood of demonstrated future actions:

$$\max_{\theta} \mathbb{E}_{(o_t, l, a_{t+1:t+H}) \sim \mathcal{D}} [\log \pi_\theta(a_{t+1:t+H} | o_t, l)].$$

The predicted action chunk $a_{t+1:t+H}$ contains the joint positions used for robot control.

Model Architecture. We show the Video2Act architecture in Figure 3. Specifically, we employ SigLIP-ViT-L/14 [54] as the image encoder to extract visual tokens, and a text encoder [46] to produce instruction embeddings. For System 2, we adopt the transformer-based Video Diffusion Model Hunyuan [29], which includes a pretrained 3D VAE to encode images into latent space and consists of 60 transformer blocks, from which we use the first 25 blocks to extract features. The spatial and motion features are extracted using our proposed spatial and motion filters, and are then compressed by two lightweight Q-formers [27] to reduce token redundancy while preserving global consistency. For System 1, we employ a 1B-parameter diffusion transformer. The image tokens, text tokens, and VDM tokens are integrated into action DiT through cross-attention.

3.2.2. Spatio-Motional Representation Extraction

To obtain clean feature signals while maintaining efficiency, we adopt an inversion-based feature extraction strategy. Following prior diffusion inversion works [45, 50], given an RGB observation sequence $\{I_t\}_{t=1}^T$, we encode it into the latent space of the video diffusion transformer \mathbf{V}_θ and extract features from the *beginning of the inversion trajectory* (i.e., the denoising step at $t_{\text{diff}}=0$), where the features preserve the original representation and minimize denoising artifacts:

$$F_{t-T:t} = \mathbf{V}_\theta(I_{t-T:t}; t_{\text{diff}}=0). \quad (1)$$

To capture both fine-grained spatial details and long-horizon motion dynamics, we decompose the input into two complementary video streams processed independently by the VDM \mathbf{V}_θ :

$$F_{t-T_s:t}^H = \mathbf{V}_\theta(I_{t-T_s:t}^H), \quad F_{t-T_l:t}^L = \mathbf{V}_\theta(I_{t-T_l:t}^L), \quad (2)$$

where H denotes the high-resolution input $I_{t-T_s:t}^H \in \mathbb{R}^{512 \times 768 \times 3}$ with the short-horizon window ($T_s = 2$), and L denotes the normal-resolution input $I_{t-T_l:t}^L \in \mathbb{R}^{256 \times 256 \times 3}$ with the long-horizon window ($T_l = 16$), and both feature branches are extracted from the inversion step.

Building on the observations in Section 3.1, we adopt a simple yet effective strategy to make these implicit cues explicit by applying non-learnable, channel-wise operators in the high-dimensional feature space to separately extract structure-aware and motion-aware representations. As shown in Figure 3, we apply a Sobel-based Spatial Filtering Operator to extract structural boundaries for each frame, and a frequency-domain Fast Fourier Transform (FFT) to capture coherent motion dynamics across adjacent frames. Formally, given the two types of features $F_{t-T_s:t}^H \in \mathbb{R}^{32 \times 48 \times 3072}$ and $F_{t-T_l:t}^L \in \mathbb{R}^{16 \times 16 \times 3072}$, we define:

$$S_t = \text{Sobel}(F_{t-T_s:t}^H), \quad M_t = \mathcal{F}_{\text{FFT}}(F_{t-T_l:t}^L). \quad (3)$$

Spatial Structure Representation (Sobel). For each channel c , we compute spatial gradients over a short temporal window $F_{t-T_s:t}^{H,(c)}$ using the standard 3×3 Sobel kernels [8, 49]:

$$S_x = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix}, \quad S_y = \begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix}.$$

The horizontal and vertical gradients are obtained via channel-wise convolution for each frame:

$$G_x^{(c)} = S_x * F_{t-T_s:t}^{H,(c)}, \quad G_y^{(c)} = S_y * F_{t-T_s:t}^{H,(c)}.$$

and the gradient magnitude is computed as

$$S_t^{(c)} = \sqrt{(G_x^{(c)})^2 + (G_y^{(c)})^2}.$$

The resulting representation $S_t = \{S_t^{(1)}, \dots, S_t^{(C)}\}$ is obtained by applying the filter independently across channels in the high-dimensional feature space, enhancing structure-aware consistency across adjacent frames.

Motion Representation (FFT). For each spatial location (i, j) and channel c , we apply a one-dimensional Discrete Fourier Transform [14] along the temporal axis:

$$\widehat{F}_k^{(c)}(i, j) = \sum_{\tau=0}^{T_l-1} F_{t-\tau}^{L,(c)}(i, j) e^{-i 2\pi k \tau / T_l},$$

where $k = 0, \dots, T_l-1$. We then apply a frequency mask \mathcal{B} , corresponding to the high-pass filter in implementation, and reconstruct the filtered sequence via an inverse DFT:

$$M_{t-\tau}^{(c)}(i, j) = \text{Re} \left(\frac{1}{T_l} \sum_{k \in \mathcal{B}} \widehat{F}_k^{(c)}(i, j) e^{i 2\pi k \tau / T_l} \right),$$

where $\tau = 0, \dots, T_l-1$. This operation suppresses low-frequency background components and highlights coherent motion patterns over time. The resulting tensors S_t and M_t explicitly encode *spatial* and *motion* cues for the action generation head.

3.3. Asynchronous Dual-System Scheme

Robots require efficient control to perform closed-loop and complex manipulation tasks, yet incorporating a VDM introduces substantial computational cost. Since our approach explicitly extracts spatio-motional features from the VDM, we investigate whether these features can be temporally reused, enabling us to reduce the frequency of VDM inference without sacrificing action generation stability. Therefore, we propose an asynchronous dual-system paradigm within the Video2Act VLA model, where the VDM serves as System 2, a slow perceptual module that provides spatially and temporally rich representations, while an additional diffusion-transformer (DiT) head serves as System 1, a fast execution module that generates precise actions.

Asynchronous Frequency. Since System 2 (VDM) is a large-scale pretrained model, it operates at a low frequency to conduct high-level contextual reasoning and extract spatio-motional features. As shown in Figure 3, the feature outputs of System 2 serve as latent conditions that temporally guide System 1's action generation over the subsequent H time steps. In contrast, System 1 focuses on real-time action execution. At each time step, it uses the most recent observation to produce an action, while being conditioned on the periodically updated feature outputs from System 2. This design resembles intuitive, reactive control, positioning System 1 as a high-frequency action generation module. Empirically (Section 4.2.2), we find that with a reasonable operating frequency ratio of $1 : n$, control accuracy remains stable while inference speed improves substantially. This asynchronous

execution paradigm provides a practical balance between computational efficiency and real-time responsiveness.

Training Objective. System 1 (the DiT head) is trained under a conditional diffusion objective. At each diffusion step, Gaussian noise ϵ is added to the action sequence [11], and System 1 \mathbf{D}_θ predicts noise conditioned on high-frequency updated image features (F_I), low-frequency updated VDM features ($F_{VDM} = \text{Cat}(S_t, M_t)$), and textual features(F_l), where τ_n denotes the diffusion timestep embedding:

$$\epsilon_\theta = \mathbf{D}_\theta(\tilde{a}_{t:t+H}, \tau_n | F_I, F_{VDM}, F_l).$$

These compressed tokens are injected into the DiT blocks of \mathbf{D}_θ via cross-attention layers. The denoising objective is then defined as:

$$\mathcal{L}_{\text{diff}} = \mathbb{E}_{a_{t:t+H}, \epsilon \sim \mathcal{N}(0,1)} [\|\epsilon_\theta - \epsilon\|_2^2]. \quad (4)$$

4. Experiments

To evaluate the capability of Video2Act, we first compare the multi-task performance of our method with existing approaches on the RoboTwin bimanual manipulation benchmark [40] in Section 4.1. We then conduct ablation studies in Section 4.2 to illustrate the effectiveness of the spatial and motion features extracted from the VDM and to analyze the relationship between system frequency ratios and manipulation performance. In Section 4.3, we present six real-world experiments to evaluate the robustness of our approach. In Section 4.4, we analyze how our method facilitates action learning through the visualization of action distributions and generalization scenarios in real-robot experiments.

4.1. Simulation Experiment

Simulation Benchmark.

To systematically evaluate our method, we conduct experiments in the **RoboTwin** simulation environment [40], which is built upon the Sapien simulator. We adopt six manipulation tasks from RoboTwin: *block handover*, *container place*, *dual bottles pick easy*, *dual bottles pick hard*, *empty cup place*, and *pick apple messy*. All tasks are performed using the ALOHA-AgileX dual-arm robot system. For each task, we generate 100 expert demonstrations.

Baselines. We compare **Video2Act** against six representative methods in robot imitation learning: Diffusion Policy [11]; ACT [56], which uses a transformer-based CVAE for bimanual manipulation; RDT-1B [37], which leverages a large-scale diffusion transformer (DiT) conditioned on a pre-trained SigLIP encoder to model bimanual action distributions; π_0 [4] and which augment a pre-trained VLM with a diffusion expert trained via conditional flow matching; and the Video Prediction Policy (VPP) [22], which uses

one-step video predictive features from a fine-tuned UNet-based Stable Video Diffusion model to condition the action head. Notably, the first four rely on static image encoders (e.g., CLIP and SigLIP) that lack explicit temporal reasoning, whereas VPP uses raw VDM features.

Training and Evaluation Details. To ensure a fair comparison, all methods use a single model for multiple tasks and are trained and evaluated under the same configuration. For VPP, we fine-tune Stable Video Diffusion on the six-task demonstration videos and replace its action head with the same pretrained diffusion head used in our method to ensure a fair comparison. During evaluation, we conduct 50 rollouts for each task with different seeds, repeating the evaluation three times per task and reporting the variance to ensure a robust comparison.

Quantitative Results. As shown in Table 1, Video2Act achieves an average success rate of 54.6% across six diverse tasks, outperforming the previous state-of-the-art methods RDT and π_0 by margins of 9.7% and 7.7%, respectively. Notably, Video2Act achieves superior performance on four out of six tasks, demonstrating the robustness of its action generation. By explicitly modeling spatio-motional representations, our approach gains a more accurate understanding of the manipulated object’s structural and dynamic states, enabling more precise action generation. In contrast, methods relying on static visual features lack temporal perception, limiting their effectiveness in dynamic modeling during the manipulation process. While VPP also leverages VDM features, it relies on raw representations that are not explicitly refined, and thus tend to contain noisy and redundant information. Therefore, Video2Act achieves an 8.3% improvement over VPP across six diverse tasks. The results demonstrate that Video2Act can explicitly extract spatial structures and motion trajectories via Spatial Filtering and FFT, filtering out task-irrelevant information while preserving clean structural boundaries and coherent motion patterns.

4.2. Ablation

To validate the effectiveness of each contribution, we conduct detailed ablation studies on the six simulated tasks.

4.2.1. Spatio-Motional Feature Extraction in VDM

To evaluate the contribution of different visual representations extracted by the VDM, we compare five configurations: (a) w/o VDM feature; (b) raw VDM feature, using the unprocessed VDM features; (c) +Sobel, using spatial-only features; (d) +FFT, using motion-only features; (e) +Sobel+FFT, combining both spatial and motion features. All variants use a fixed dual-system operating ratio of 1:1 to ensure a consistent comparison. As shown in Figure 5 (a), introducing Sobel-based spatial filtering improves the task success rate by 7.6%, while adding FFT-based motion representation further enhances temporal smoothness, yielding an additional 8.6% improvement. When both are applied,

Table 1. Simulation experiment results across six RoboTwin manipulation tasks. We compare Video2Act with five baselines, including Diffusion Policy (DP), ACT, RDT-1B, π_0 , and Video Prediction Policy (VPP). All methods are trained under a multi-task setting using 100 expert demonstrations per task and are evaluated over 50 rollouts with different seeds, repeated three times.

Methods	Block Handover	Container Place	Cup Place	Bottles Easy	Bottles Hard	Pick Apple	Mean S.R.
DP	3.7 ± 0.6	10.0 ± 0.0	11.7 ± 1.5	76.3 ± 0.6	40.0 ± 1.0	13.0 ± 1.0	25.8 ± 0.2
ACT	13.7 ± 5.9	15.0 ± 5.6	23.0 ± 4.4	19.3 ± 7.6	19.7 ± 4.5	14.3 ± 4.0	17.5 ± 2.2
RDT	95.0 ± 2.0	11.7 ± 0.6	42.3 ± 2.5	78.0 ± 1.0	32.0 ± 2.0	10.7 ± 1.5	44.9 ± 0.5
π_0	52.7 ± 3.2	25.3 ± 0.6	37.7 ± 0.6	93.3 ± 0.6	36.3 ± 2.1	36.0 ± 1.0	46.9 ± 0.8
VPP	82.7 ± 6.1	22.7 ± 1.2	34.7 ± 1.2	66.7 ± 4.2	45.3 ± 1.2	26.0 ± 3.5	46.3 ± 0.9
Video2Act	96.7 ± 1.2	18.0 ± 3.5	43.3 ± 3.1	96.0 ± 3.5	46.7 ± 1.2	26.7 ± 3.1	54.6 ± 1.1

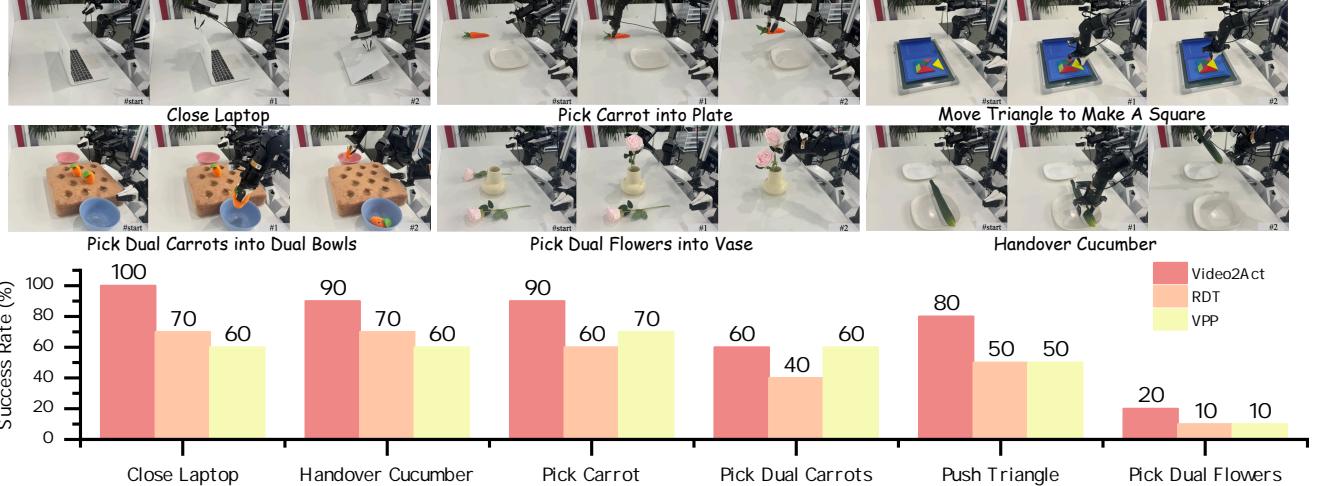


Figure 4. Real-world experiment results across six manipulation tasks on the Agilex Cobot Magic platform. All methods are trained on 100 demonstrations per task and compared against two closely related baselines, RDT and VPP. We report success rates over 10 rollouts per task under diverse tabletop configurations.

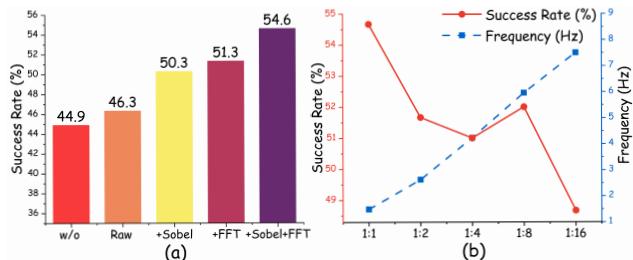


Figure 5. Ablation Study. We investigate (a) the effectiveness of spatio-motion feature extraction and (b) how the operating ratio influences success rate and action generation frequency.

the model achieves the highest overall performance (54.6%). These results indicate that our method effectively extracts spatial and temporal cues that provide complementary benefits: Sobel emphasizes structural information, while FFT captures motion consistency, and together they enhance the stability of action generation.

4.2.2. Dual-System Operating Frequency Ratio

We further investigate how different operating frequency ratios affect performance. The ratio between the slow VDM component (System 2) and the fast action component (System 1) is varied from 1:1 to 1:16. The ratio corresponds to different update intervals between the VDM latent features and the action module's visual features. As shown in Figure 5 (b), the model achieves the best trade-off between accuracy and inference efficiency when the ratio is set to 1:8, resulting in a model inference frequency of 5.96 Hz. Since we set the action chunk size to $H = 64$, the overall system achieves real-time, high-frequency control (approximately 380 Hz). These results indicate that the VDM-extracted features inherently preserve temporal consistency across time steps, supporting asynchronous operation that improves inference efficiency without sacrificing accuracy.

4.3. Real-World Experiment

Dataset Collection. In our real-world robot experiments setup, we employ the Agilex Cobot Magic platform, which is equipped with a front-view and two wrist-view cameras.

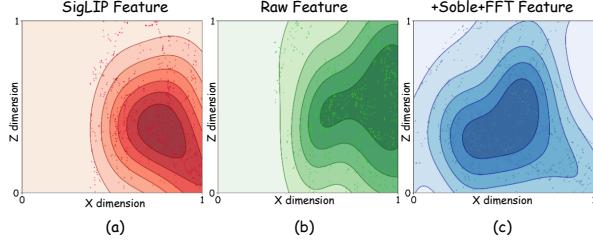


Figure 6. Action Distribution Visualization. We project the right-hand end-effector poses (X-Z dimensions) from 40 successful rollouts in the *dual bottles pick hard* task. The comparison shows the learned distributions for: (a) SigLIP Feature; (b) Raw VDM Feature, and (c) +Sobel+FFT VDM Feature.

We perform six distinct manipulation tasks: 1) *close laptop*, 2) *pick carrot*, 3) *pick dual carrots*, 4) *pick dual flowers*, 5) *handover cucumber* and 6) *push triangle*, which involve various objects and action types. For each task, 100 demonstrations are collected via master-puppet teleoperation, with objects placed in varying positions on the table to ensure diversity. Additional real-world dataset details can be found in Appendix A.

Training and Evaluation Details. All methods are trained to learn task-specific policies under the same configuration in simulation. We evaluate Video2Act against RDT [37], π_0 [4] and VPP [22]. The final checkpoint is used to perform 10 rollouts across varied tabletop positions.

Quantitative and Qualitative Results. As shown in the second row of Figure 4, Video2Act achieves superior real-world performance with an average success rate of **73.3%** on the Agilex Robot, outperforming all baseline methods across six real-world tasks. Notably, Video2Act attains substantially higher success rates on bimanual manipulation tasks that require precise spatial reasoning and dynamic coordination. These results validate the effectiveness of our proposed Video2Act and its explicit spatio-motion representations in improving real-world robustness for complex manipulation tasks. As shown in the first row of Figure 4, Video2Act can accurately execute pick-and-place, articulated object manipulation, bimanual handover, and other precise manipulation tasks, highlighting the generality of our approach across diverse tasks. Additional visualizations, failure case analyses, and execution videos are provided in Appendix C, Appendix D, and the supplementary material, respectively.

4.4. Analysis

4.4.1. Action Distribution Analysis

To further analyze how our extracted spatio-motion features contribute to policy robustness and generalization, we visualize the action distribution of Video2Act. The experiment is conducted in the *dual bottles pick hard* task, where the initial positions of the bottles are fixed to ensure con-

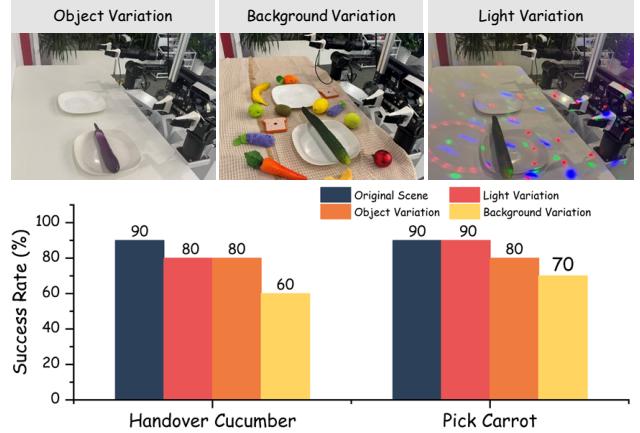


Figure 7. Generalization results under unseen real-world scenarios. We evaluate Video2Act on the *pick dual flowers* and *handover cucumber* tasks across three shifts, including object variation, background variation, and lighting variation.

trolled experimental conditions. For each model configuration, we collect 40 successful trajectories and record the end-effector poses of the right hand.

We then compare the distributions along the X and Z action dimensions across three different model configurations: (a) SigLIP feature; (b) raw VDM feature; (c) +Sobel+FFT feature, combining both spatial and motion features. As shown in Figure 6, the base policy (a), which relies only on static image features, exhibits a **narrowly concentrated** action distribution. Interestingly, directly adding the unprocessed VDM features (b) also demonstrates a relatively concentrated distribution, failing to fully learn the diverse trajectories. In contrast, Video2Act (c), conditioned on the refined +Sobel+FFT features, learns a significantly **broad**er distribution. The results demonstrate that our method successfully captures object structural information and inter-object motion relationships, enabling it to learn a more diverse range of manipulation trajectories from the demonstrations rather than overfitting to a narrow subset of the distribution.

4.4.2. Generalization Experiments

To further evaluate the zero-shot generalization capability of our method, we conduct experiments under several unseen configurations in the real-world tasks *pick dual flowers* and *handover cucumber*. The unseen configurations include: 1) Object Variation, 2) Background Variation, and 3) Lighting Variation. As shown in Figure 7, Video2Act maintains consistent success rates across all generalization scenarios. The explicit spatial-aware representations enable our model to preserve robust object-structure understanding under varying object appearances and background contexts, while the motion-aware representations provide stable dynamic priors that exhibit strong consistency to visual disturbances. The performance demonstrates that our approach has the

potential to capture essential manipulation concepts that generalize effectively to novel environments.

5. Conclusion

In this paper, we presented Video2Act, an asynchronous dual-system VLA framework that leverages the spatio-temporal representations of video diffusion models to enhance robotic policy learning. Our systematic analysis revealed that VDM features naturally encode spatial- and motion-aware representations that remain robust to robot motion and viewpoint changes. Building on this insight, we proposed explicit spatial and motion representation extraction methods using Spatial Filtering Operators and FFT, which effectively capture what to manipulate and how to move. Meanwhile, an asynchronous dual-system strategy is introduced, in which the VDM is designated as the slow perceptual System 2, and the action head is assigned as the fast execution System 1, thereby enabling high-frequency and stable action generation. Extensive experiments in both simulation and real-world environments demonstrate that Video2Act achieves SOTA performance across various manipulation tasks.

References

- [1] Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*, 2023.
- [2] Anurag Ajay, Seungwook Han, Yilun Du, Shuang Li, Abhi Gupta, Tommi Jaakkola, Josh Tenenbaum, Leslie Kaelbling, Akash Srivastava, and Pulkit Agrawal. Compositional foundation models for hierarchical planning. *Advances in Neural Information Processing Systems*, 36:22304–22325, 2023.
- [3] Johan Bjorck, Fernando Castañeda, Nikita Cherniadev, Xingye Da, Runyu Ding, Linxi Fan, Yu Fang, Dieter Fox, Fengyuan Hu, Spencer Huang, et al. Gr0ot n1: An open foundation model for generalist humanoid robots. *arXiv preprint arXiv:2503.14734*, 2025.
- [4] Kevin Black, Noah Brown, Danny Driess, Adnan Esmail, Michael Equi, Chelsea Finn, Niccolo Fusai, Lachy Groom, Karol Hausman, Brian Ichter, Szymon Jakubczak, Tim Jones, Liyiming Ke, Sergey Levine, Adrian Li-Bell, Mohith Mothukuri, Suraj Nair, Karl Pertsch, Lucy Xiaoyang Shi, James Tanner, Quan Vuong, Anna Walling, Haohuan Wang, and Ury Zhilinsky. π_0 : A vision-language-action flow model for general robot control. *arXiv preprint arXiv:2410.24164*, 2024.
- [5] Anthony Brohan, Noah Brown, Justice Carbajal, Yevgen Chebotar, Xi Chen, Krzysztof Choromanski, Tianli Ding, Danny Driess, Avinava Dubey, Chelsea Finn, Pete Florence, Chuyuan Fu, Montse Gonzalez Arenas, Keerthana Gopalakrishnan, Kehang Han, Karol Hausman, Alex Herzog, Jasmine Hsu, Brian Ichter, Alex Irpan, Nikhil Joshi, Ryan Julian, Dmitry Kalashnikov, Yuheng Kuang, Isabel Leal, Lisa Lee, Tsang-Wei Edward Lee, Sergey Levine, Yao Lu, Henryk Michalewski, Igor Mordatch, Karl Pertsch, Kanishka Rao, Krista Reymann, Michael Ryoo, Grecia Salazar, Pan-nag Sanketi, Pierre Sermanet, Jaspiar Singh, Anikait Singh, Radu Soricut, Huong Tran, Vincent Vanhoucke, Quan Vuong, Ayzaan Wahid, Stefan Welker, Paul Wohlhart, Jialin Wu, Fei Xia, Ted Xiao, Peng Xu, Sichun Xu, Tianhe Yu, and Brianna Zitkovich. Rt-2: Vision-language-action models transfer web knowledge to robotic control. In *arXiv preprint arXiv:2307.15818*, 2023.
- [6] Jake Bruce, Michael D Dennis, Ashley Edwards, Jack Parker-Holder, Yuge Shi, Edward Hughes, Matthew Lai, Aditi Mavalankar, Richie Steigerwald, Chris Apps, et al. Genie: Generative interactive environments. In *Forty-first International Conference on Machine Learning*, 2024.
- [7] Qingwen Bu, Hongyang Li, Li Chen, Jisong Cai, Jia Zeng, Heming Cui, Maoqing Yao, and Yu Qiao. Towards synergistic, generalized, and efficient dual-system for robotic manipulation, 2025.
- [8] Qiong Chang, Xiang Li, Yun Li, and Jun Miyazaki. Multi-directional sobel operator kernel on gpus. *Journal of parallel and distributed computing*, 177:160–170, 2023.
- [9] Hao Chen, Jiaming Liu, Chenyang Gu, Zhuoyang Liu, Renrui Zhang, Xiaoqi Li, Xiao He, Yandong Guo, Chi-Wing Fu, Shanghang Zhang, et al. Fast-in-slow: A dual-system foundation model unifying fast manipulation within slow reasoning. *arXiv preprint arXiv:2506.01953*, 2025.
- [10] 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.
- [11] Cheng Chi, Zhenjia Xu, Siyuan Feng, Eric Cousineau, Yilun Du, Benjamin Burchfiel, Russ Tedrake, and Shuran Song. Diffusion policy: Visuomotor policy learning via action diffusion. *The International Journal of Robotics Research*, 2024.
- [12] Xiaowei Chi, Kuangzhi Ge, Jiaming Liu, Siyuan Zhou, Peidong Jia, Zichen He, Yuzhen Liu, Tingguang Li, Lei Han, Sirui Han, et al. Mind: Unified visual imagination and control via hierarchical world models. *arXiv preprint arXiv:2506.18897*, 2025.
- [13] Xiaowei Chi, Peidong Jia, Chun-Kai Fan, Xiaozhu Ju, Weishi Mi, Kevin Zhang, Zhiyuan Qin, Wanxin Tian, Kuangzhi Ge, Hao Li, et al. Wow: Towards a world omniscient world model through embodied interaction. *arXiv preprint arXiv:2509.22642*, 2025.
- [14] James W Cooley and John W Tukey. An algorithm for the machine calculation of complex fourier series. *Mathematics of computation*, 19(90):297–301, 1965.
- [15] Figure AI Team. Helix: A vision-language-action model for generalist humanoid control. <https://www.figure.ai/news/helix>, 2025. Accessed: 2025-05-07.
- [16] Chelsea Finn, Ian Goodfellow, and Sergey Levine. Unsupervised learning for physical interaction through video prediction. *Advances in neural information processing systems*, 29, 2016.
- [17] Zipeng Fu, Tony Z Zhao, and Chelsea Finn. Mobile aloha: Learning bimanual mobile manipulation with low-cost whole-body teleoperation. *arXiv preprint arXiv:2401.02117*, 2024.
- [18] Rafael C Gonzalez and Richard E Woods. *Digital image processing*. Prentice Hall, 2008.

- [19] ByungOk Han, Jaehong Kim, and Jinhyeok Jang. A dual process vla: Efficient robotic manipulation leveraging vlm, 2024.
- [20] Jonathan Ho, Tim Salimans, Alexey Gritsenko, William Chan, Mohammad Norouzi, and David J Fleet. Video diffusion models. *Advances in Neural Information Processing Systems*, 35:8633–8646, 2022.
- [21] Yingdong Hu, Fanqi Lin, Tong Zhang, Li Yi, and Yang Gao. Look before you leap: Unveiling the power of gpt-4v in robotic vision-language planning. *arXiv preprint arXiv:2311.17842*, 2023.
- [22] 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.
- [23] Physical Intelligence, Kevin Black, Noah Brown, James Darpinian, Karan Dhabalia, Danny Driess, Adnan Esmail, Michael Equi, Chelsea Finn, Niccolo Fusai, Manuel Y. Gallicher, 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.
- [24] Yueru Jia, Aosong Cheng, Yuhui Yuan, Chuke Wang, Ji Li, Huihu Jia, and Shanghang Zhang. Designedit: Unify spatial-aware image editing via training-free inpainting with a multi-layered latent diffusion framework. In *AAAI Conference on Artificial Intelligence*, 2025.
- [25] Dmitry Kalashnikov, Alex Irpan, Peter Pastor, Julian Ibarz, Alexander Herzog, Eric Jang, Deirdre Quillen, Ethan Holly, Mrinal Kalakrishnan, Vincent Vanhoucke, et al. Scalable deep reinforcement learning for vision-based robotic manipulation. In *Conference on robot learning*, pages 651–673. PMLR, 2018.
- [26] 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.
- [27] Sungkyung Kim, Adam Lee, Junyoung Park, Andrew Chung, Jusang Oh, and Jay-Yoon Lee. Towards efficient visual-language alignment of the q-former for visual reasoning tasks. *arXiv preprint arXiv:2410.09489*, 2024.
- [28] Po-Chen Ko, Jiayuan Mao, Yilun Du, Shao-Hua Sun, and Joshua B Tenenbaum. Learning to act from actionless videos through dense correspondences. *arXiv preprint arXiv:2310.08576*, 2023.
- [29] Weijie Kong, Qi Tian, Zijian Zhang, Rox Min, Zuozhuo Dai, Jin Zhou, Jiangfeng Xiong, Xin Li, Bo Wu, Jianwei Zhang, et al. Hunyuanyvideo: A systematic framework for large video generative models. *arXiv preprint arXiv:2412.03603*, 2024.
- [30] Brenden M Lake, Tomer D Ullman, Joshua B Tenenbaum, and Samuel J Gershman. Building machines that learn and think like people. *Behavioral and Brain Sciences*, 40:e253, 2017.
- [31] Sergey Levine, Chelsea Finn, Trevor Darrell, and Pieter Abbeel. End-to-end training of deep visuomotor policies. *Journal of Machine Learning Research*, 17(39):1–40, 2016.
- [32] Shuang Li, Yihuai Gao, Dorsa Sadigh, and Shuran Song. Unified video action model. *arXiv preprint arXiv:2503.00200*, 2025.
- [33] Xiaoqi Li, Mingxu Zhang, Yiran Geng, Haoran Geng, Yuxing Long, Yan Shen, Renrui Zhang, Jiaming Liu, and Hao Dong. Manipllm: Embodied multimodal large language model for object-centric robotic manipulation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18061–18070, 2024.
- [34] Yue Liao, Pengfei Zhou, Siyuan Huang, Donglin Yang, Shengcong Chen, Yuxin Jiang, Yue Hu, Jingbin Cai, Si Liu, Jianlan Luo, et al. Genie envisioner: A unified world foundation platform for robotic manipulation. *arXiv preprint arXiv:2508.05635*, 2025.
- [35] Fanqi Lin, Ruiqian Nai, Yingdong Hu, Jiacheng You, Junming Zhao, and Yang Gao. Onetwovla: A unified vision-language-action model with adaptive reasoning. *arXiv preprint arXiv:2505.11917*, 2025.
- [36] Jiaming Liu, Hao Chen, Pengju An, Zhuoyang Liu, Renrui Zhang, Chenyang Gu, Xiaoqi Li, Ziyu Guo, Sixiang Chen, Mengzhen Liu, et al. Hybridvla: Collaborative diffusion and autoregression in a unified vision-language-action model. *arXiv preprint arXiv:2503.10631*, 2025.
- [37] Songming Liu, Lingxuan Wu, Bangguo Li, Hengkai Tan, Huayu Chen, Zhengyi Wang, Ke Xu, Hang Su, and Jun Zhu. Rdt-1b: a diffusion foundation model for bimanual manipulation. *arXiv preprint arXiv:2410.07864*, 2024.
- [38] Yunhao Luo and Yilun Du. Grounding video models to actions through goal conditioned exploration. *arXiv preprint arXiv:2411.07223*, 2024.
- [39] Corey Lynch, Mohi Khansari, Ted Xiao, Vikash Kumar, Jonathan Tompson, Sergey Levine, and Pierre Sermanet. Learning latent plans from play. In *Conference on robot learning*, pages 1113–1132. Pmlr, 2020.
- [40] Yao Mu, Tianxing Chen, Zanxin Chen, Shijia Peng, Zhiqian Lan, Zeyu Gao, Zhixuan Liang, Qiaojun Yu, Yude Zou, Mingkun Xu, Lunkai Lin, Zhiqiang Xie, Mingyu Ding, and Ping Luo. Robotwin: Dual-arm robot benchmark with generative digital twins. In *Proceedings of the Computer Vision and Pattern Recognition Conference (CVPR)*, pages 27649–27660, 2025.
- [41] Octo Model Team, Dibya Ghosh, Homer Walke, Karl Pertsch, Kevin Black, Oier Mees, Sudeep Dasari, Joey Hejna, Charles Xu, Jianlan Luo, Tobias Kreiman, You Liang Tan, Dorsa Sadigh, Chelsea Finn, and Sergey Levine. Octo: An open-source generalist robot policy. <https://octo-models.github.io>, 2023.
- [42] Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, et al. Dinov2: Learning robust visual features without supervision. *arXiv preprint arXiv:2304.07193*, 2023.

- [43] William Peebles and Saining Xie. Scalable diffusion models with transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 4195–4205, 2023.
- [44] Karl Pertsch, Kyle Stachowicz, Brian Ichter, Danny Driess, Suraj Nair, Quan Vuong, Oier Mees, Chelsea Finn, and Sergey Levine. Fast: Efficient action tokenization for vision-language-action models. *arXiv preprint arXiv:2501.09747*, 2025.
- [45] Alexander Pondaven, Aliaksandr Siarohin, Sergey Tulyakov, Philip Torr, and Fabio Pizzati. Video motion transfer with diffusion transformers. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 22911–22921, 2025.
- [46] Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of machine learning research*, 21(140):1–67, 2020.
- [47] Hanno Scharr. Optimal operators in digital image processing. In *Dissertation, University of Heidelberg*, 2000.
- [48] Ramprasaath R Selvaraju, Michael Cogswell, Abhishek Das, Ramakrishna Vedantam, Devi Parikh, and Dhruv Batra. Grad-cam: Visual explanations from deep networks via gradient-based localization. In *Proceedings of the IEEE international conference on computer vision*, pages 618–626, 2017.
- [49] Irwin Sobel and Gary Feldman. An isotropic 3×3 gradient operator. Technical Report SAIL TR 1968, Stanford Artificial Intelligence Laboratory, 1968.
- [50] Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. *arXiv preprint arXiv:2010.02502*, 2020.
- [51] Junjie Wen, Minjie Zhu, Yichen Zhu, Zhibin Tang, Jinming Li, Zhongyi Zhou, Chengmeng Li, Xiaoyu Liu, Yixin Peng, Chaomin Shen, et al. Diffusion-vla: Scaling robot foundation models via unified diffusion and autoregression. *arXiv preprint arXiv:2412.03293*, 2024.
- [52] Junjie Wen, Yichen Zhu, Jinming Li, Minjie Zhu, Kun Wu, Zhiyuan Xu, Ning Liu, Ran Cheng, Chaomin Shen, Yixin Peng, et al. Tinyvla: Towards fast, data-efficient vision-language-action models for robotic manipulation. *arXiv preprint arXiv:2409.12514*, 2024.
- [53] Mengjiao Yang, Yilun Du, Kamyar Ghasemipour, Jonathan Tompson, Dale Schuurmans, and Pieter Abbeel. Learning interactive real-world simulators. *arXiv preprint arXiv:2310.06114*, 1(2):6, 2023.
- [54] Xiaohua Zhai, Basil Mustafa, Alexander Kolesnikov, and Lucas Beyer. Sigmoid loss for language image pre-training. In *International Conference on Computer Vision (ICCV)*, 2023.
- [55] Jianke Zhang, Yanjiang Guo, Xiaoyu Chen, Yen-Jen Wang, Yucheng Hu, Chengming Shi, and Jianyu Chen. Hirt: Enhancing robotic control with hierarchical robot transformers. *arXiv preprint arXiv:2410.05273*, 2024.
- [56] Tony Z. Zhao, Vikash Kumar, Sergey Levine, and Chelsea Finn. Learning Fine-Grained Bimanual Manipulation with Low-Cost Hardware. In *Proceedings of Robotics: Science and Systems*, Daegu, Republic of Korea, 2023.
- [57] Siyuan Zhou, Yilun Du, Jiaben Chen, Yandong Li, Dit-Yan Yeung, and Chuang Gan. Robodreamer: Learning compositional world models for robot imagination. *arXiv preprint arXiv:2404.12377*, 2024.

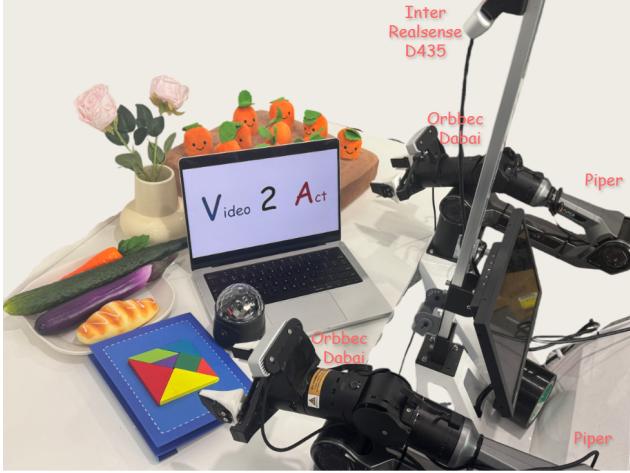


Figure 8. Real-world robot setup and experimental assets. We utilize the Agilex Cobot Magic platform with four 6-DoF Agilex Piper robotic arms, equipped with an Intel RealSense D435 head camera and two Orbbec Dabai wrist cameras, along with the object assets used across all real-world tasks.

We provide additional details, as well as quantitative and qualitative results of our Video2Act in this supplementary material. The outline is shown below.

- **A. Additional Experimental Setup and Data Details (Appendix A)**
 - Simulation Setup and Data Collection
 - Real-World Robot Hardware Setup
 - Real-World Data Collection
- **B. Additional Ablation Study (Appendix B)**
 - Spatial Filter Choice Ablation
 - Layer-Depth Configuration Ablation
 - Further Action Distribution Analysis
- **C. Additional Visualizations (Appendix C)**
 - Simulation Qualitative Results
 - Real-World Qualitative Results
- **D. Failure Analysis (Appendix D)**
 - Failure Case Analysis and Visualization

A. Additional Experimental Setup and Data Details

In this section, we provide additional details on the simulation configuration, real-world robot hardware, and data collection procedures that complement the descriptions in the main text.

A.1. Simulation Setup and Data Collection

In RoboTwin [40], demonstrations are automatically generated via a generative digital twin pipeline. First, diverse 3D assets are created from single 2D images using generative foundation models and annotated with functional axes and contact points. Subsequently, GPT-4 [1] decomposes the

manipulation tasks and infers spatial constraints based on these annotations. Finally, these constraints are translated into executable code that drives a motion planner to solve for collision-free, kinematically feasible trajectories. For each task, we generate 100 expert demonstrations using this automated framework. Detailed descriptions of six simulation tasks are as follows:

1. **block handover.** A long block is initialized on the left side of the table. The left arm grasps the upper side of the block and performs a handoff to the right arm, which then places the block onto a blue mat on the right side. This task requires high-level inter-arm coordination to ensure a stable transition of the object between grippers without dropping it.
2. **container place.** Random containers, such as cups or bowls, are placed arbitrarily on the table. The robot must identify the container’s location to select the appropriate arm (left or right) and move the container into a fixed plate. This task tests the model’s ability to handle object diversity and make dynamic arm-selection decisions based on spatial distribution.
3. **dual bottles pick easy.** A red bottle and a green bottle are placed upright on the left and right sides of the table, respectively. The robot utilizes both arms simultaneously to grasp and lift the two bottles to a designated location. This task demands synchronized control of both end-effectors to execute parallel manipulation actions efficiently.
4. **dual bottles pick hard.** Similar to the easy setting, but the bottles are initialized with random postures (e.g., lying down) rather than standing upright. The model must perceive the complex 6D poses of the objects and perform precise orientation adjustments to align the grippers for a successful dual-arm grasp.
5. **empty cup place.** An empty cup and a coaster are randomly placed on either the left or right side of the table. The robot must grasp the cup and accurately place it onto the coaster. This task requires fine-grained spatial reasoning to align the cup with the coaster’s surface while avoiding collisions.
6. **pick apple messy.** An apple is placed on the table surrounded by four random distractor items. The robot is required to identify, grasp, and lift the apple amidst the clutter. This task challenges the model’s visual robustness and ability to plan collision-free trajectories in a cluttered environment.

A.2. Real-World Robot Hardware Setup

In our real-world robot experiments, we employ the Agilex Cobot Magic platform equipped with four 6-DoF Agilex

Table 2. Agilex Piper Arm Joint Specifications

Joint Name	Range	Maximum Speed
J1	$-154^\circ \sim 154^\circ$	$180^\circ/\text{s}$
J2	$0^\circ \sim 195^\circ$	$195^\circ/\text{s}$
J3	$-175^\circ \sim 0^\circ$	$180^\circ/\text{s}$
J4	$-100^\circ \sim 112^\circ$	$225^\circ/\text{s}$
J5	$-75^\circ \sim 75^\circ$	$225^\circ/\text{s}$
J6	$-170^\circ \sim 170^\circ$	$225^\circ/\text{s}$

Table 3. Configurations for cameras

Parameter	Head Camera	Wrist Camera
Resolution (H×W)	640×480	640×480
FOV (H×W)	$56^\circ \times 43^\circ$	$67^\circ \times 52^\circ$
Frequency	30 fps	30 fps

Piper robotic arms. Both the inference and execution puppet arms are equipped with parallel grippers featuring an 85 mm stroke. The robotic arms operate under joint position control, with the motion range of each joint detailed in Table 2. The robot is integrated with three cameras: a head-view Intel RealSense D435 camera mounted on the head, and two wrist-view Orbbec Dabai cameras—one attached to the left wrist and the other to the right wrist. The specific parameters of the two cameras are shown in Table 3. The head camera provides a global perspective for environmental perception, while the two wrist cameras offer localized visual feedback for fine-grained manipulation tasks performed by the respective arms. The overall hardware configuration and real-world experimental assets are shown in Figure 8. All algorithms utilize RGB information from all three cameras.

A.3. Details of Real-World Data Collection

Building upon our robot hardware setup, we collect six challenging real-world tasks, comprising three single-arm tasks and three bimanual tasks. For each task, 100 demonstrations were collected via master–puppet teleoperation [17]. Each demonstration comprises time-synchronized recordings of the puppet arm’s joint positions and RGB video streams from three fixed perspectives. To ensure data diversity, objects were placed in varying positions on the table. Detailed descriptions of six robotic tasks are as follows:

1. close laptop. The robot uses its arm to close the laptop’s opened folding screen. This task requires precise spatial perception to locate the screen and the hinge, as well as controlled force exertion to avoid damaging the screen during contact. The motion trajectory must be smooth and consistent to ensure stable manipulation of the articulated object.

2. pick carrot. The robot moves its arm to the carrot’s position, grasps the carrot, transfers it above the plate, and releases it into the plate. The task relies heavily on visual localization of the carrot and the plate, while the grasping and releasing actions demand accurate pose prediction and gripper control. The motion must be stable to prevent the carrot from rolling or falling during transfer.

3. pick dual carrots. The robot places the right carrot into the right bowl, then the left carrot into the left bowl (with carrot position variations). This bimanual task requires coordinated motion planning and spatial reasoning to handle positional variations. The model must perceive the geometric relationship between each carrot and its target bowl, and execute sequential actions without interference between the two arms.

4. pick dual flowers. The robot first grasps the right flower and inserts it into the vase, then grasps the left flower and inserts it into the vase. The insertion process demands fine-grained spatial awareness to align the flower stem with the vase opening. The model must also avoid colliding with the first flower when inserting the second, highlighting the need for dynamic trajectory adjustment.

5. handover cucumber. The robot grasps the cucumber from the plate with its left hand, transfers it to the right hand, and places it into the right plate. This task involves inter-arm coordination and precise timing during the handover phase. The model must ensure a stable grip transition and avoid dropping the cucumber, relying on both spatial and motion-aware representations to synchronize the dual-arm actions.

6. push triangle. The robot pushes a triangle to complete a tangram pattern into a square. The task requires understanding of geometric relationships and spatial composition. The pushing motion must be carefully planned to align the triangle with the existing pattern, involving both structural perception and trajectory optimization to achieve the target configuration.

B. Additional Ablation Study

B.1. Spatial Filter Choice Ablation

Spatial filtering in Video2Act aims to extract stable structure-aware cues from the VDM latent features. We compare Sobel [49] with two representative alternatives: a second-order Laplacian [18] operator and a high-quality first-order Scharr [47] operator. We keep the motion branch (FFT [14]) fixed and only vary the spatial filter.

As shown in Figure 9, we can see that +Scharr+FFT reaches an average success rate of 53.3 %, while Laplacian

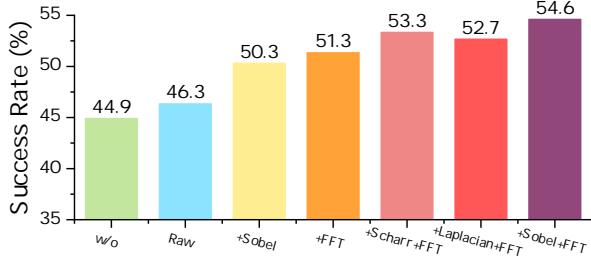


Figure 9. Spatial filter ablation. We add +Scharr+FFT and +Laplacian+FFT to our spatial filtering module and evaluate their six-task average success rates on RoboTwin under the same settings as in the main text.

reaches 52.7 %, indicating that spatial filtering indeed benefits manipulation and that Scharr performs very similarly to Sobel. The slightly lower score of Laplacian may be due to its isotropic second-order formulation, which tends to amplify high-frequency noise and weaken directional edge responses. In contrast, first-order operators such as Sobel and Scharr emphasize gradient orientation and object contours more effectively, producing spatial cues that are more aligned with manipulation-relevant boundaries. This suggests that preserving directional edge information is more beneficial than enforcing rotational isotropy when extracting structure from VDM latents.

B.2. Layer-Depth Configuration Ablation

The video diffusion model used in Video2Act follows the dual-stream/single-stream hierarchy of the Hunyuan VDM architecture [29]. The first 20 *dual-stream transformer blocks* perform multimodal self-attention among the SigLIP image embeddings, text embeddings, and latent features, enabling early fusion of visual and instruction information. The subsequent 40 *single-stream transformer blocks* apply cross-attention between the latent features and the fused multimodal tokens, progressively enriching the representation with deeper semantic structure.

While this hierarchical design is well suited for video generation, it is not clear how much depth is actually needed for extracting the spatio-motional cues required by our policy, or whether using more cross-attention blocks offers tangible benefit beyond increasing inference latency. To investigate this behavior, we perform a structured ablation aligned with the natural stage boundaries of the dual-stream and single-stream blocks.

We vary the depth across these two stages as follows:

- 5 dual-stream transformer blocks
- 20 dual-stream transformer blocks
- 20 dual-stream blocks + 5 single-stream blocks (ours)
- 20 dual-stream blocks + 20 single-stream blocks
- 20 dual-stream blocks + 40 single-stream blocks

We evaluate how the average success rate and the System-

Table 4. Layer-depth configuration ablation using different numbers of dual-stream (DS) and single-stream (SS) transformer blocks.

Configuration	Success (%)	Latency (ms)
5 DS	51.3	253.6
20 DS	52.7	488.4
20 DS + 5 SS (ours)	54.6	587.9
20 DS + 20 SS	52.3	886.5
20 DS + 40 SS	54.0	1284.6

2 perceptual latency—measured as the combined time of the VDM forward pass and feature-processing pipeline—change across these configurations. The results show that adding a small number of single-stream blocks on top of the 20 dual-stream blocks provides the best trade-off between depth and performance. The 20 DS + 5 SS configuration (ours) achieves the highest success rate of 54.6%, indicating that a lightweight amount of cross-attention is sufficient for extracting the spatio-motional cues needed for policy learning. Increasing the number of single-stream blocks to 20 or 40 does not consistently improve performance and instead leads to diminishing or fluctuating gains, suggesting that deeper cross-attention introduces additional latency without contributing meaningful new information.

B.3. Further Action Distribution Analysis

We provide further visualization of the action distribution to verify the robustness of our method.

Unlike the previous experiment where the green bottle was fixed, we standardize the initial position of the **red bottle** within the *dual bottles pick hard* task. To provide a more comprehensive analysis of the manipulation strategy, we record the end-effector poses of **the left arm** across 40 successful trajectories for each model configuration.

We maintain the same comparison settings: (a) SigLIP feature; (b) raw VDM feature; and (c) +Sobel+FFT feature. As illustrated in Figure 11, the base policy (a) exhibits a **narrowly concentrated** distribution, indicating a tendency to overfit to a specific subset of expert behaviors. Similarly, the inclusion of unprocessed VDM features (b) fails to significantly expand the diversity of the learned trajectories, resulting in a relatively restricted workspace usage.

In contrast, Video2Act (c), utilizing the refined +Sobel+FFT features, generates a significantly **broader and more diverse** action distribution. These results further confirm that our method effectively leverages spatio-motional cues to learn robust, multi-modal manipulation strategies that generalize well across different object configurations, rather than collapsing into a single behavioral mode.

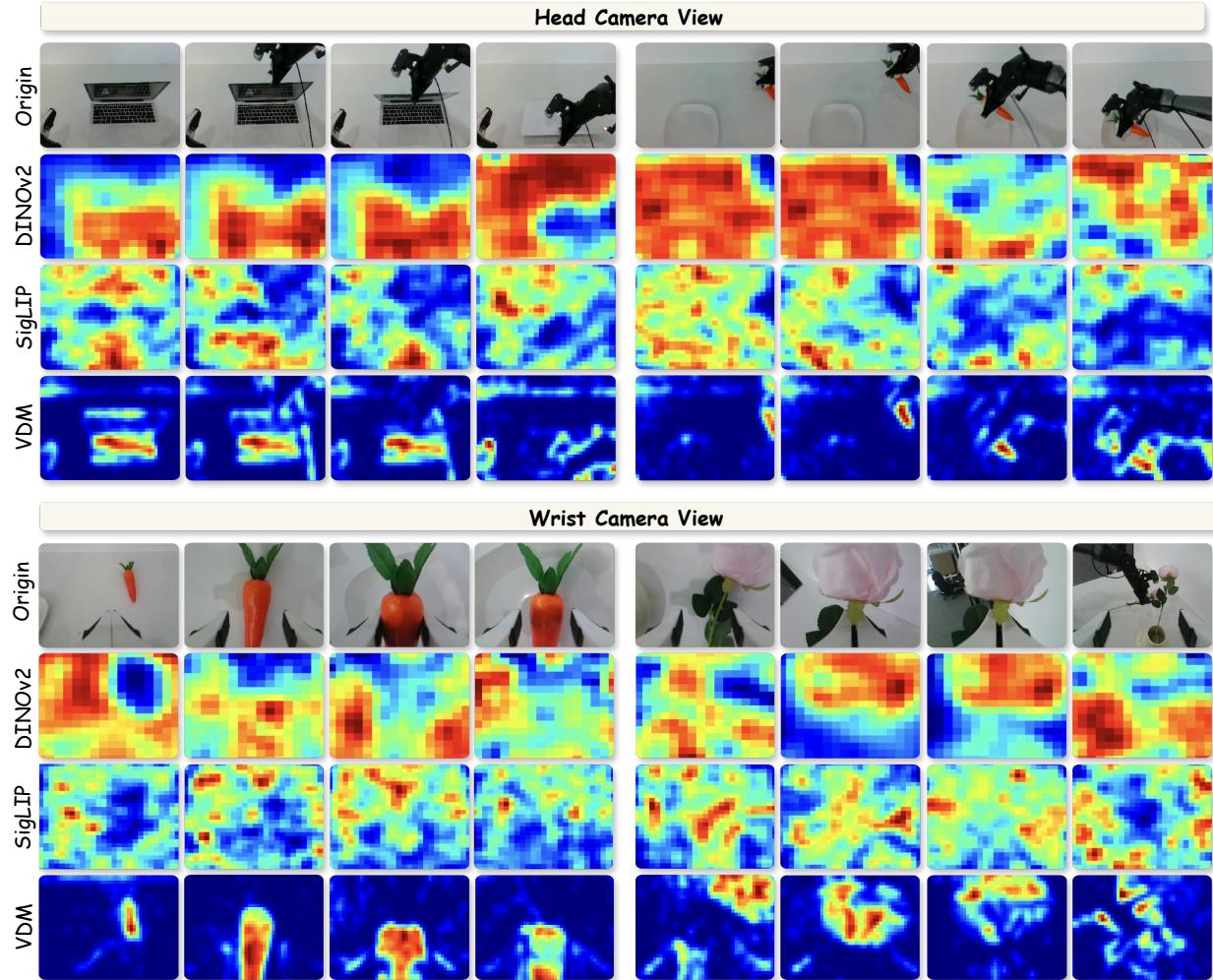


Figure 10. **Grad-CAM comparison on real-world scenarios.** We compare DINOv2, SigLIP, and our VDM-based representation on *close laptop* and *pick and place carrot* from the head camera view, and on *pick and place carrot* and *pick dual flowers* from the wrist camera view. DINOv2 and SigLIP exhibit scattered and unstable attention that often drifts across frames, whereas the VDM representation consistently maintains an object-centric focus.

C. Additional Visualizations

C.1. Real-World Grad-CAM Visualizations

We further visualize feature-level Grad-CAM [48] on real-world scenarios and compare DINOv2 [42], SigLIP [54], and our VDM-based representation. As shown in Figure 10, across both the head camera and wrist camera views, our method exhibits noticeably more stable and coherent activations over time, whereas DINOv2 and SigLIP often produce inconsistent or scattered focus regions. In a few individual frames—such as in the *close laptop* task from the head view or the *pick dual flowers* task from the wrist view—DINOv2 can occasionally localize the target object well. However, in

most frames, its attention drifts to irrelevant regions, leading to unstable and unreliable activation patterns. In contrast, VDM representation consistently attends to the manipulated objects and their immediate surroundings, aligning more closely with task-relevant areas. A remaining limitation is that VDM localization becomes less precise when the object and background share nearly identical colors, such as in the fourth frame of the *close laptop* task where both appear white.

C.2. Simulation Qualitative Results

Figure 12 presents keyframe visualizations of Video2Act executing six distinct tasks within the RoboTwin simulation

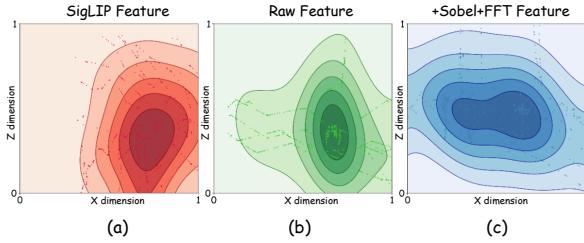


Figure 11. Action Distribution Visualization. We project the left-hand end-effector poses (X-Z dimensions) from 40 successful rollouts in the *dual bottles pick hard* task. The comparison shows the learned distributions for: (a) SigLIP Feature; (b) Raw VDM Feature, and (c) +Sobel+FFT VDM Feature.

environment. These sequences demonstrate how our dual-system framework effectively translates high-level reasoning into precise, temporally coherent actions.

Bimanual Coordination. In the *block handover* task (Row 1), Video2Act exhibits precise inter-arm coordination. The left arm stably grasps the object and synchronizes with the right arm for a smooth transfer, validating the effectiveness of our motion-aware features (FFT) in modeling temporal dependencies between two end-effectors. Similarly, in *dual bottles pick easy* (Row 3), the model demonstrates the ability to control both arms simultaneously to grasp upright objects. Crucially, in the *dual bottles pick hard* task (Row 4), where bottles are initialized in random fallen poses, the robot accurately perceives the complex 6D poses and adjusts the gripper orientation for a successful grasp. This highlights that our spatial filtering (Sobel) effectively captures fine-grained structural cues even under significant pose variations.

Robustness in Clutter and Variation. In the *pick apple messy* task (Row 6), the robot successfully identifies and grasps the target apple amidst distinct distractor objects (e.g., banana, brush), demonstrating that the VDM-based System 2 effectively filters out task-irrelevant visual biases. In *container place* (Row 2) and *empty cup place* (Row 5), the model shows robust spatial reasoning, accurately aligning the grasped object with the target placement zone (plate or coaster) despite variations in object appearance and initial positions.

These qualitative results confirm that Video2Act does not merely memorize trajectories but learns a generalized representation of what to manipulate and how to move, enabling stable execution across diverse dynamic scenarios.

C.3. Real-World Qualitative Results

Figure 13 visualizes the execution of six diverse real-world manipulation tasks by Video2Act. These qualitative results highlight three key capabilities of our system:

Dynamic Bimanual Coordination. In the *handover cucumber* task (Row 5), the system demonstrates **precise temporal**

synchronization. The left arm (giver) and right arm (receiver) align their velocities perfectly during the transfer phase, preventing the object from falling. This verifies that our FFT-based motion extraction effectively captures the inter-arm temporal dependencies required for dynamic handover. Similarly, in the *pick dual carrots* task (Row 3), Video2Act exhibits robust sequential planning. It coordinates the two arms to transport the carrots one after another, maintaining continuous spatial awareness to execute the second grasp accurately without causing interference or collisions with the first arm’s trajectory.

Fine-Grained Geometric Reasoning. The *push triangle* task (Row 6) requires precise spatial reasoning to align a geometric shape with a target slot. Video2Act successfully perceives the orientation of the triangle and plans a pushing trajectory that completes the square pattern, validating the benefit of our Spatial Filtering Operators in capturing object boundaries. Furthermore, in the *pick dual flowers* task (Row 4), the model accurately locates and grasps thin flower stems, which poses a significant challenge for traditional encoders. This demonstrates the superior fine-grained perception of our VDM-based representations during the sequential insertion process.

Articulated Object and Trajectory Modeling. In the *close laptop* task (Row 2), the robot exhibits an understanding of the articulated object’s constraints, generating a smooth, circular trajectory that follows the natural mechanics of the laptop hinge without applying excessive force. Finally, the *pick and place carrot* task (Row 1) confirms the baseline stability of our method in standard pick-and-place scenarios, showing robust grasping and precise release placement into the plate.

D. Failure Analysis

Through real-world experiments on the Agilex platform, we categorize the observed failure modes into four distinct types, as visualized in Figure 14:

1. The first case illustrates a failure driven by **common errors** during the *pick dual flowers* task. The extremely thin geometry of the flower stems leads to a minor positional offset during the initial grasp. This slight deviation accumulates throughout the trajectory, causing the robot to drift from the optimal path and ultimately resulting in a misalignment failure when attempting to insert the flower into the vase.
2. The second case, observed in the *pick dual carrots* task, involves **manipulation height errors** regarding the manipulation height (z -axis). The gripper initiates the closing action before reaching the optimal grasping depth for the carrot, resulting in a “grasping air” failure.
3. The third case reveals a **grasp slip** during the *handover cucumber* task. The left gripper (giver) releases the object prematurely before the right gripper (receiver) has

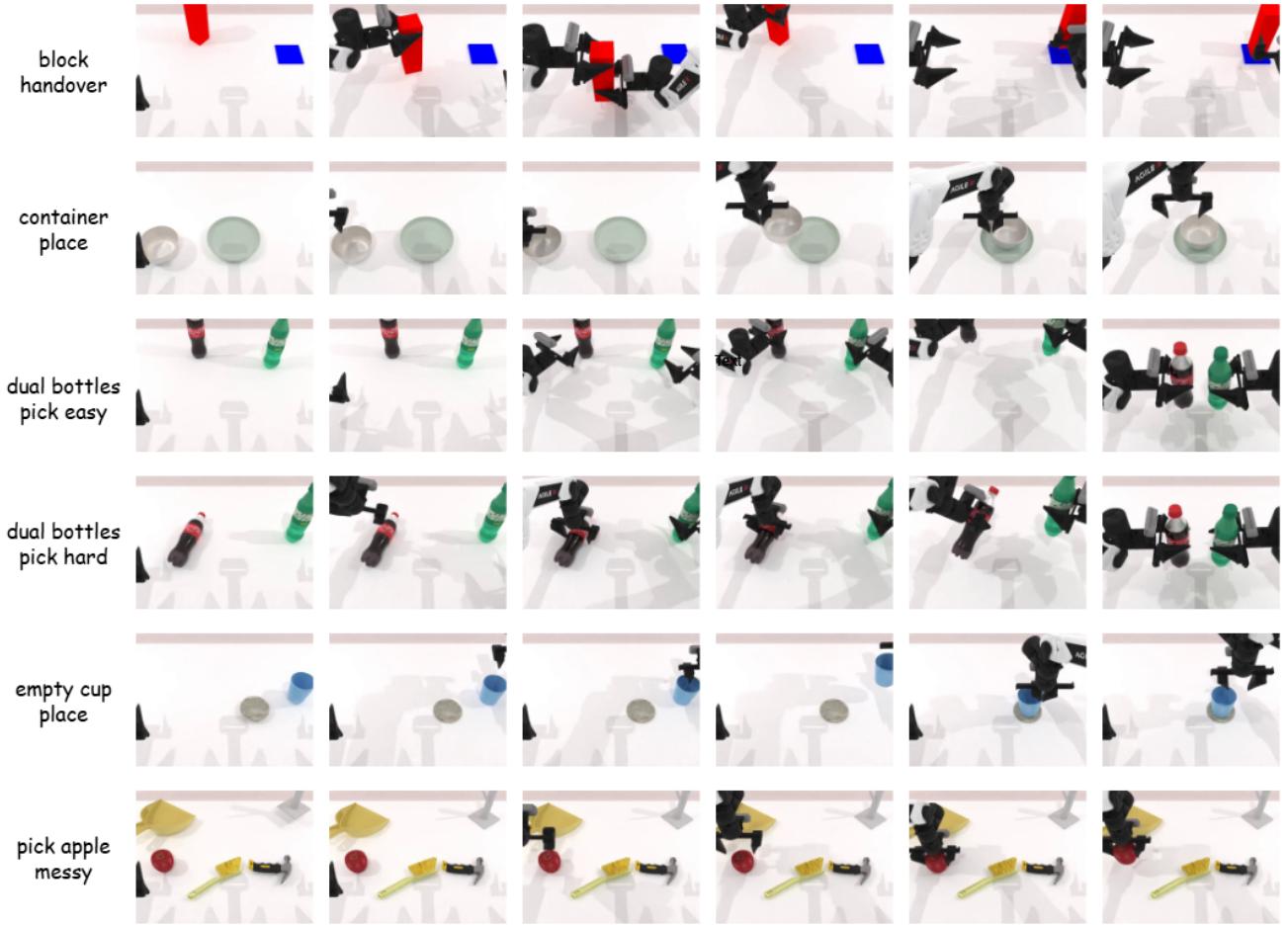


Figure 12. **Robot execution progress in simulation tasks.** We visualize key frames of the robot’s execution process from a static exterior view in simulation tasks.

established a stable hold, causing the cucumber to fall. This reflects a dissonance in the temporal logic between the two arms.

4. The fourth case presents a limitation in **minor hardware inaccuracies** during the *push triangle* task. The gripper tends to move further than needed (overshooting), breaking the original shape and failing to build a square in the end. This indicates a deficiency in the model’s fine-grained control regarding motion termination.

To alleviate these limitations, we plan to scale up the collection of high-quality demonstrations and introduce rigorous constraints during training, thereby enhancing robustness in physical environments. Moreover, enabling our System 2 to autonomously detect and rectify erroneous actions will be a key direction for future work.

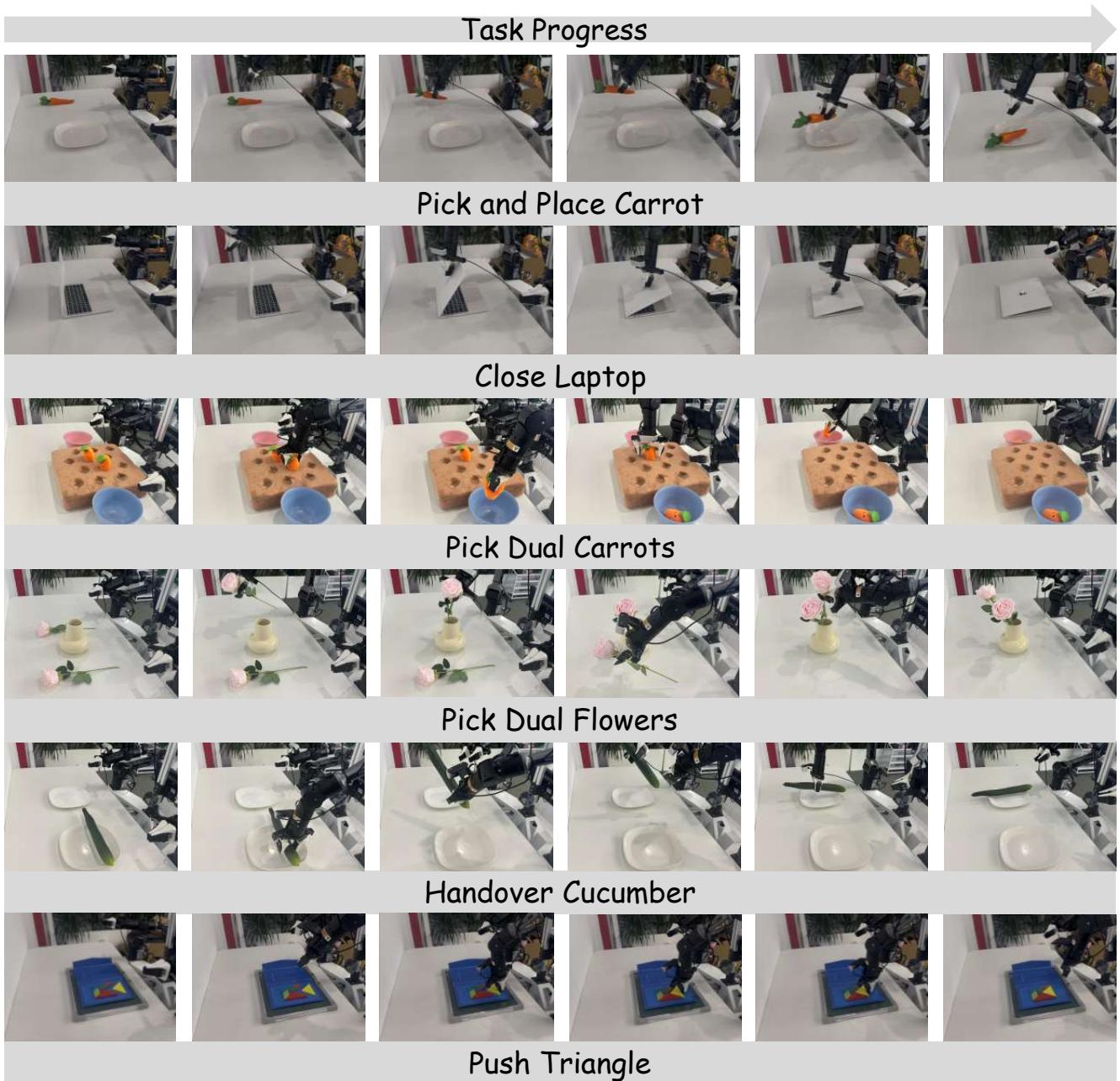


Figure 13. **Robot execution progress in real-world tasks.** We visualize key frames of the robot’s execution process from a static exterior view in real-world tasks.

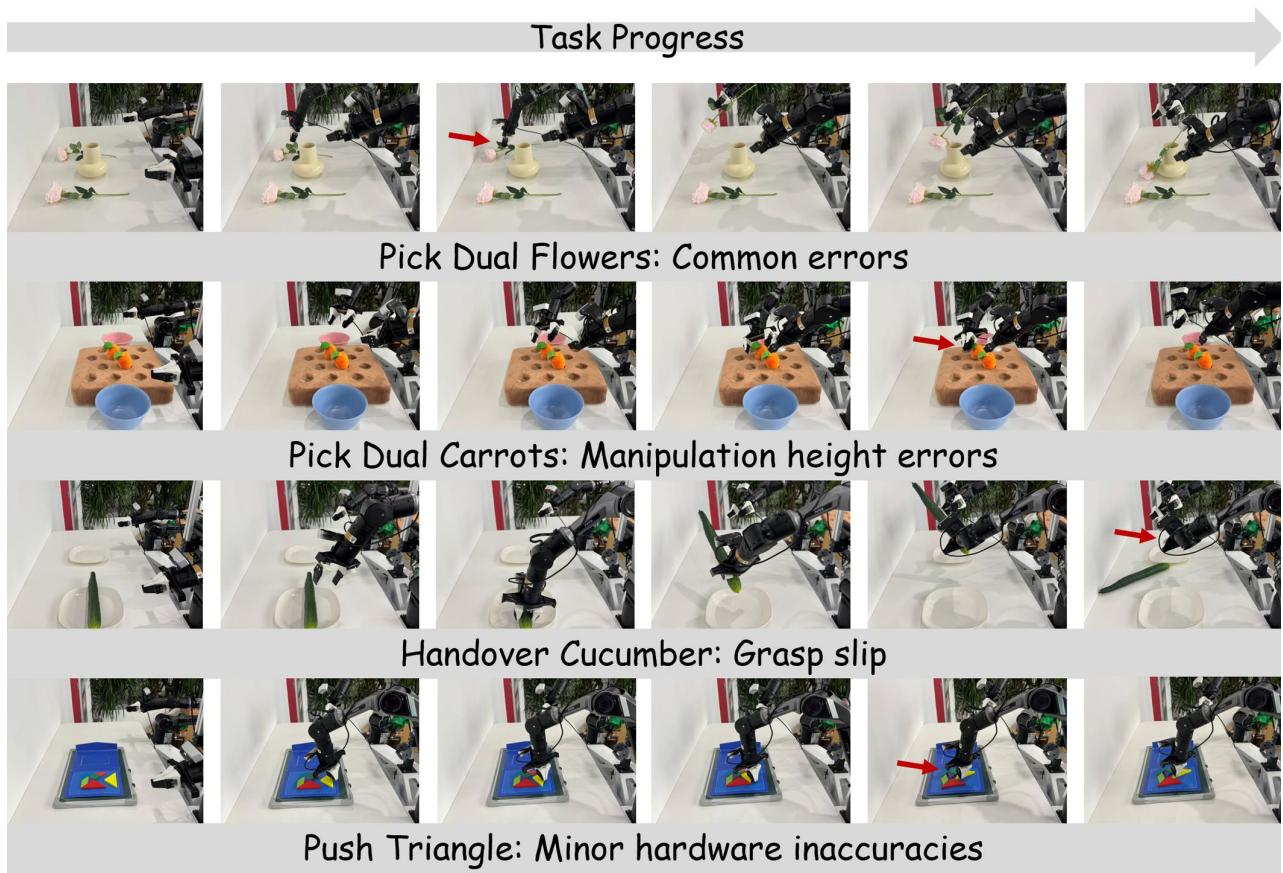


Figure 14. **Failure cases in real-world tasks.** We visualize the failure cases observed in four real-world experiments, with key error frames during execution highlighted using red arrows.