Making Foresight Actionable: Repurposing Representation Alignment in World Action Models

Lu Qiu[1], **Yizhuo Li**[†,1], **Yi Chen**[1], **Yuying Ge**[2], **Yixiao Ge**[2], **Xihui Liu**[†,1]
[1] The University of Hong Kong [2] XPENG Robotics
† Corresponding authors
Project Page: `https://xpeng-robotics.github.io/agra`

Figure 1: Correctly predicted future in World Action Models does not necessarily yield reliable control, because the visual features can remain poorly organized for action decoding and cause attention to drift toward task-irrelevant regions. We propose AGRA, an Action-Grounded Representation Alignment objective that aligns video features with spatially coherent representations from a frozen foundation visual encoder. This alignment makes world model features more action-grounded and focuses action attention on task-relevant regions, thus leading to higher task success rate.

Abstract: World Action Models (WAMs) offer a promising route for robot manipulation by using video generation models to model future scene evolution before producing control actions. However, our empirical observations reveal a phenomenon: generating plausible visual futures does not always guarantee the extraction of accurate actions. To diagnose this failure, we conduct action-head attention analysis and causal interventions. We find that the action decoder fails to focus on task-relevant interaction regions and remains sensitive to perturbations in task-irrelevant areas. This reveals a representation mismatch: hidden states optimized for visual reconstruction are not inherently organized in a form useful for low-level action control. In this paper, we propose AGRA, an Action-Grounded Representation Alignment objective that regularizes the world-action interface by aligning intermediate video diffusion features with spatially coherent semantic representations from a foundation visual encoder. We evaluate AGRA on real-world manipulation tasks. Experiments show that AGRA makes world model representations more action-grounded: by focusing the action decoder on the correct interaction regions, it improves object localization accuracy and affordance understanding, and makes the policy more robust to perturbations in task-irrelevant regions. As a result, AGRA consistently improves both in-distribution performance and out-of-distribution generalization over the baseline world action model.

Keywords: World Action Model, Representation Alignment, Robot Manipulation

---

[*] The work was done during the author's internship at XPENG Robotics.

1 Introduction

Learning generalist robot policies [1, 2, 3] requires models to connect visual perception, instructions, and continuous control for robust physical interaction. A natural approach is to equip a policy with predictive model of the world [4, 5, 6, 7, 8, 9]. World Action Model (WAM) [10] first reasons about how scene may evolve and then uses its prediction to guide action generation. [11, 12, 13] Recent progress in video diffusion models [14, 15, 16] makes this paradigm attractive: large video models can encode rich spatiotemporal structure, predict future, and provide features that contain useful information about motion and task progress in robot manipulation. [10, 11, 12, 17]

However, our empirical observations show that plausible visual foresight does not necessarily translate into reliable action prediction. A WAM can generate a plausible and task-consistent future while the action decoder still produces an incorrect motion. To understand this failure mode, we diagnose the world-action interface from both attentional and causal perspectives. We find that the action decoder in the baseline WAM often fails to concentrate on task-relevant interaction regions, such as the hand-object contact areas. In addition, causal interventions on world model hidden states show that action outputs can be sensitive to perturbations in task-irrelevant background regions.

This evidence reveals an *action-grounding gap* between video prediction and action decoding. Video diffusion models are optimized for pixel reconstruction and therefore encode dense appearance-level information, like texture, color, and background clutter [18, 19, 20]. However, action prediction does not depend on all visual details in the scene; instead, it is determined by a sparse set of spatially localized and functional factors, such as target objects, contact regions, and affordance geometry. We characterize this mismatch through feature visualization. Compared with foundation visual encoders, video diffusion features are more entangled with low-level appearance. As a result, action decoder is prone to attend to spurious or irrelevant regions, leading to erroneous action predictions.

To close this gap, we introduce *Action-Grounded Representation Alignment* (AGRA). AGRA repurposes representation alignment as a mechanism for action grounding in WAMs. The key idea is to align selected hidden states of the video diffusion model with spatially structured semantic features extracted from a frozen foundation visual encoder. These semantic targets provide a stable reference for the representation field seen by the action decoder: regions with similar semantic or functional roles are encouraged to form coherent structures, while appearance variations become less dominant.

We evaluate AGRA on real-world manipulation tasks. Our analysis shows that by applying AGRA, action decoder attends more accurately to hand-object interaction regions, and becomes more robust to perturbations in task-irrelevant areas. This indicates that AGRA makes the visual representations more action-grounded. By focusing action decoder on the critical interaction regions, AGRA improves object localization accuracy and affordance understanding, while also making the policy more robust to shifts in task-irrelevant regions. As a result, AGRA achieves an in-distribution (ID) success rate of 80%, substantially outperforming the baseline WAM which obtains 34%. AGRA also yields stronger robustness under Semantic, Instance-Level, and Attribute Generalization, improving performance by 27%, 32%, and 32%, respectively, compared with the baseline WAM.

Our contributions are threefold. First, we identify and analyze the action-grounding gap in WAMs, showing that plausible visual prediction does not guarantee action-readable representations. Second, we propose AGRA, a representation-alignment objective that regularizes the world-action interface by aligning video model hidden states with spatially coherent pretrained visual features. Third, we validate AGRA in real-world manipulation tasks with the IRON-R01-1.11 humanoid robot, demonstrating improved execution success, stronger out-of-distribution (OOD) generalization, and better cross-embodiment transfer ability.

2 Related Work

Video Generation as World Modeling. Recent progress in video generation suggests that predicting future frames is a direct way to learn world dynamics. Large-scale video generators model spatio-temporal structure with autoregressive or diffusion-based objectives, enabling temporally coherent synthesis [21, 22, 23, 24, 15]. This perspective motivates the view that video generators implicitly model scene evolution, object motion, and physical interaction [14, 25], and that's why they are named "World Models". Some studies make this connection explicit by developing interactive world models [26, 27, 28, 29]. More recently, embodied world models [16, 30, 31, 32, 33] have been trained or adapted for physical AI and robot domains. Trained on large-scale robot demonstrations, these models encode richer priors about physical evolution, making them useful for synthetic data generation [34], evaluation environment [35, 36, 37], and planning in robotic domains.

Video Prediction and Video Representations for Robot Policies. Video prediction has long served as a predictive interface for robot control. A prominent line of work adopts an explicit generation-and-control paradigm: world models forecast pixel-level future frames under actions [28] or instructions, and applying model-predictive control (MPC) [38] or inverse dynamics models (IDMs) to provide policies [39, 40]. Although intuitive, such explicit methods require full video denoising at inference, slowing down policy execution. Another line of work [17, 12, 41, 42, 43, 13, 44] moves to implicit future representations for action prediction. These methods condition policies or IDMs on latent representations of world models that encode plausible future dynamics. More recently, LingbotVA [11] employed an autoregressive paradigm for large-scale training, and DreamZero [10] jointly modeled video and action, demonstrating strong zero-shot generalization capabilities. However, these world model-based policies directly use predictive features as control inputs, and have not systematically studied whether those representations are suitable for downstream action control.

Representation Alignment for Generative Models. Diffusion Transformers (DiT) [45] provide backbone for modern image and video generation. Some studies [46, 18] have shown that the denoising process in generative diffusion models can induce meaningful representations [47, 48, 49], but the quality of these representations still lags behind those learned through self-supervised learning methods [50, 51, 52]. REPA [18] connects these two lines by aligning diffusion model hidden states with representations from foundation visual encoders, improving DiT training efficiency and generation quality. Several variants [53, 54, 55, 56] extend this idea to enable end-to-end training or improving cross-frame consistency in videos. These methods primarily use representation alignment to improve generative modeling, while AGRA repurposes this for action-grounded robot policy learning. Close to our motivation, a recent study [20] explores VAE encoder choices for world models and finds that semantic spaces better preserve policy-facing structure. But such latent world models [57, 58] learn dynamics from scratch in compact latent spaces, while WAMs leverage pretrained video representations that already encode physical dynamics from internet-scale data.

3 AGRA: Action-Grounded Representation Alignment

3.1 Baseline World Action Model Architecture

The baseline WAM follows a dual DiT architecture, where a world model (Video DiT) generates visual future and an action head (Action DiT) transforms the intermediate future representations into continuous actions. Given the current RGB observation o_0, language instruction c, and robot proprioceptive state s_0, the policy predicts a continuous action chunk with K horizon and d_a dimension:

$$\hat{\mathbf{a}}_{1:K} = \pi(o_0, s_0, c), \qquad \mathbf{a}_{1:K} \in \mathbb{R}^{K \times d_a}. \tag{1}$$

Video DiT predicts future frames and exposes its hidden states to action DiT. Let $\mathbf{H}_\ell^{\mathrm{vid}} \in \mathbb{R}^{N_v \times d_v}$ denote the hidden states of video layer ℓ, for the j-th action cross-attention block, a video layer ℓ_j is selected uniformly across the depth of the video DiT to use a multi-layer bridge strategy. Let $\mathbf{X}^{\mathrm{act}}$ denote the hidden states of Action DiT, visual representations are injected into action head by:

$$\mathcal{G}_j = \mathrm{Proj}_j\left(\mathbf{H}_{\ell_j}^{\mathrm{vid}}\right) \in \mathbb{R}^{N_v \times d_{\mathrm{act}}}, \mathbf{X}_j^{\mathrm{act}} = \mathrm{CrossAttn}_j\left(Q = \mathbf{X}_{j-1}^{\mathrm{act}}, K = \mathcal{G}_j, V = \mathcal{G}_j\right). \tag{2}$$

Both branches are trained with flow matching [59]. During action generation, video DiT is evaluated once at a fixed high-noise level $\tau_v^{\mathrm{cond}} = 1$ to extract predictive features, and the action DiT performs iterative denoising in action space, keeping inference efficient. In this paper, we use Cosmos-Predict-2.5 as video DiT. For the training objective $\mathcal{L}_{\mathrm{WAM}}$ and more details, please refer to Appendix.

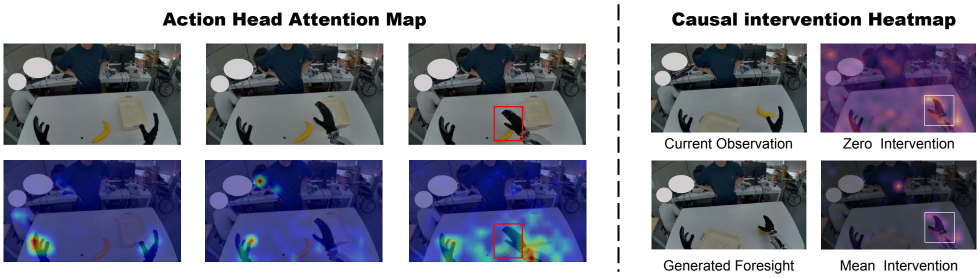


Figure 2: **Diagnosis of the action-grounding gap in the baseline WAM. (Left)** Cross-attention maps show the action head ignoring the critical hand-object interaction region, despite generating a plausible future video. **(Right)** Causal intervention heatmaps (where brighter regions indicate areas that cause the robot's action to drift when disrupted) reveal that the baseline model's decisions are heavily influenced by task-irrelevant background elements rather than the interaction area.

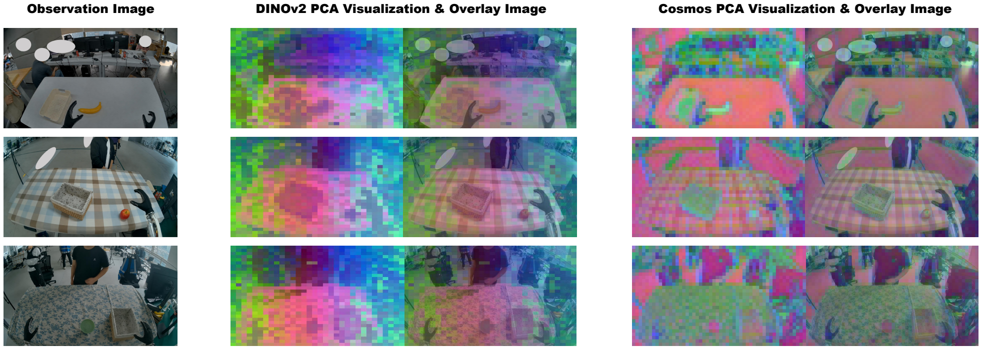


Figure 3: PCA visualization of DINOv2 and Cosmos-Predict-2.5 representations. DINOv2 organizes semantically and functionally similar regions into spatially coherent structures, whereas Cosmos features remain more entangled with local appearance.

3.2 Diagnosing the Action-Grounding Gap in World Action Models

A central assumption behind WAMs is that a predictive world representation should provide useful guidance for action decoding. But in real-world deployment, we observe that generating plausible visual futures does not always guarantee the extraction of accurate control actions.

Action-head attention analysis. To diagnose this failure mode, we inspect the cross-attention maps in the action DiT to understand how the action decoder reads from the world model features. We average the attention weights over action tokens and attention heads, and project them back to the video latent shape. In most cases, the attention maps can localize coarse robot hands areas. However, they often fail to concentrate on the most action-critical interaction site. As illustrated in the left panel of Figure 2, the model's attention is distracted by the stationary left hand and the table background, failing to focus on the crucial interaction region between the right hand and the banana.

Causal intervention for hidden-state actionability. Attention maps reveal where the action decoder reads from, but it does not establish which regions causally affect the predicted action. We therefore perform token-level causal intervention on world model hidden states to estimate the actionability of each spatial location. We apply interventions, such as zero or mean value replacement, to each spatial token. We then use Euclidean distance to measure how much each location affects the action, and generate a heatmap by applying min-max normalization. Higher values on the heatmap represent a greater influence on the action. For visualization, we report the map on the last latent frame, where motion and interaction cues are most pronounced. Ideally, the high-impact regions should align with the task-critical hand-object interaction site. However, as demonstrated by an example in the right panel of Figure 2, when applying the Mean Intervention, the region most influencing the action falls on a person in the background. Under the Zero Intervention, the entire irrelevant background exerts a significant impact on the action. This implies that perturbations in the unrelated areas affect the predicted action significantly, making it difficult for the model to maintain robust performance in OOD scenarios, such as altered backgrounds or added distractors.

Reconstruction-optimized features do not necessarily expose action-readable scene structure. In this paper, we use Cosmos-Predict-2.5 as world model. We visualize the feature space structure of Cosmos and the state-of-the-art self-supervised vision model DINOv2 [51] using Principal Component Analysis (PCA) [60]. For each model, we collect patch features from multiple samples, fit PCA jointly, and map the first three principal components to RGB. As shown in Figure 3, DINOv2 features exhibit a more coherent spatial organization: semantically and functionally similar regions, such as tables and backgrounds, tend to be mapped to consistent colors across visually different scenes. Even cluttered backgrounds are represented smoothly and separably, and table regions are less sensitive to appearance changes. In contrast, Cosmos features are sensitive to visual details. Background clutter, textured tablecloths, and appearance changes of functionally similar surfaces often produce different features. This suggests that its hidden states expose semantics in a less spatially stable and less accessible form for downstream action decoding. This observation is consistent with the motivation of REPA [18]: although diffusion models can develop meaningful internal representations, their representation quality may still lag behind strong self-supervised visual encoders.

3.3 Repurposing Representation Alignment for Action Grounding

Effect of AGRA. The preceding analysis shows that DINOv2 features provide a more spatially organized and semantically grounded feature space than the native world models. We therefore encourage the guidance visual representations used by the action DiT to align with the DINOv2 feature space. Inspired by REPA, which demonstrates that aligning diffusion hidden states with pretrained visual representations improves their semantic quality, we introduce AGRA: an action-grounded representation alignment objective that regularizes the world-action interface. Unlike the original REPA objective, which is designed to improve diffusion model generation, AGRA regularizes the world-action interface used for robot control.

Method. The first step is to construct DINOv2 targets with the same temporal and spatial shape as world model hidden states. Let the video contain T_v latent frames, with each latent frame covering a set of frames $\{r_t^1, \dots, r_t^q\}$. For the t-th latent frame, we use the first corresponding frame $\bar{o}_t = o_{r_t^1}$ as its semantic reference and extract patch-level features from a frozen DINOv2 encoder g_ψ:

$$\mathbf{Y}_t = g_\psi(\bar{o}_t) \in \mathbb{R}^{H_d \times W_d \times d_d}. \tag{3}$$

The world model feature is mapped back to a 2D spatial grid $\mathbf{H}^{\text{vid}} \in \mathbb{R}^{T_v \times H_v \times W_v \times d_v}$. To achieve dimensional alignment for DINOv2, we apply spatial interpolation and temporal concatenation:

$$\tilde{\mathbf{Y}}_t = \text{Interp}(\mathbf{Y}_t; H_v, W_v) \in \mathbb{R}^{H_v \times W_v \times d_d}, \tilde{\mathbf{Y}} = \{\tilde{\mathbf{Y}}_t\}_{t=1}^{T_v} \in \mathbb{R}^{T_v \times H_v \times W_v \times d_d}. \tag{4}$$

The second step is to optimize selected world model hidden states toward these DINOv2 targets. Let $\mathcal{S}_{\text{agra}} = \{\ell_k^{\text{agra}}\}_{k=1}^{K}$ denote the set of world model layers used for alignment. $\mathcal{S}_{\text{agra}}$ can contain either a single layer or multiple layers; in our default setting, unless otherwise specified, we only select one layer for alignment. For each layer, we project the corresponding video hidden state into the DINOv2 feature space via a projector P_k:

$$\mathbf{H}^{\text{vid}}_{\ell_k^{\text{agra}}} = f^{\text{vid}}_{\theta, \ell_k^{\text{agra}}}\left(\mathbf{z}^{\tau_v}_{1:T_v}, o_0, c, \tau_v\right), \mathbf{Z}_k = P_k\left(\mathbf{H}^{\text{vid}}_{\ell_k^{\text{agra}}}\right) \in \mathbb{R}^{T_v \times H_v \times W_v \times d_d}, \tag{5}$$

where f_θ^{vid} is the world model parameter, $\mathbf{z}^{\tau_v}_{1:T_v}$ is the noisy video latent, and τ_v is the corresponding noise level. AGRA minimizes the negative cosine similarity between the projected world model hidden states and the aligned DINOv2 features:

$$\mathcal{L}_{\text{AGRA}} = -\frac{1}{K T_v H_v W_v} \sum_{k=1}^{K} \sum_{t=1}^{T_v} \sum_{u=1}^{H_v} \sum_{v=1}^{W_v} \cos\left(\mathbf{Z}_{k,t,u,v}, \tilde{\mathbf{Y}}_{t,u,v}\right), \cos(\mathbf{x}, \mathbf{y}) = \frac{\mathbf{x}^\top \mathbf{y}}{\|\mathbf{x}\|_2 \|\mathbf{y}\|_2 + \epsilon}. \tag{6}$$

The final training objective is:

$$\mathcal{L} = \mathcal{L}_{\text{WAM}} + \lambda_{\text{agra}} \mathcal{L}_{\text{AGRA}}, \tag{7}$$

where $\mathcal{L}_{\text{WAM}}$ is the training loss of baseline WAM, and λ_{agra} controls the strength of representation alignment. This auxiliary loss does not replace video prediction or action flow matching; it only constrains the intermediate world-model features exposed to the action decoder.

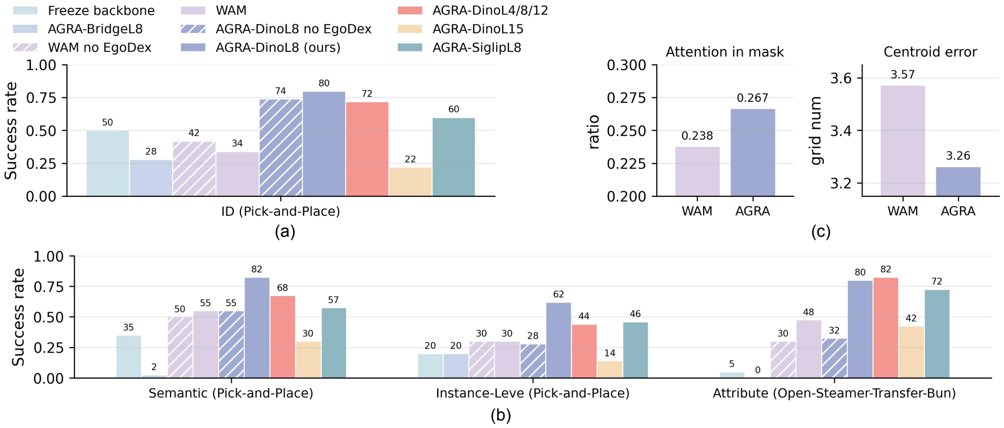


Figure 4: Main results in real-world evaluation. (a) Main results for in-distribution (ID) scenarios. (b) Main results across three out-of-distribution (OOD) generalization scenarios. (c) Quantitative results of the action-head attention analysis.

4 Experiments

4.1 Experimental Setup

We use Cosmos-Predict-2.5-2B with 28 layers as world model, and couple it with an 8-layer action head with 500 M parameters. We evaluate AGRA on the IRON-R01-1.11 humanoid robot, and consider two manipulation tasks, Pick-and-Place and Open-Steamer-Transfer-Bun. The evaluation scenarios includes In-Distribution (ID) setting and three OOD settings (Semantic Generalization, Instance-Level Generalization and Attribute Generalization). More details of the model, dataset, training details, and evaluation setup can be found in Appendix.

Compared variants. We compare following variants. *Freeze backbone* keeps Cosmos frozen and only optimizes action head. *WAM* is the baseline model without representation alignment. ***AGRA-DinoL8* (ours) aligns the 8th layer of Cosmos with DINOv2 features. This is our default model unless otherwise specified.** *AGRA-DinoL15* aligns a deeper Cosmos layer with DINOv2. *AGRA-DinoL4/8/12* aligns multiple Cosmos layers simultaneously. *AGRA-SiglipL8* replaces DINOv2 with SigLIP as the visual alignment target. *AGRA-BridgeL8* feeds the aligned 8th layer feature repeatedly into all action cross-attention layers. Finally, we compare variants trained with and without EgoDex [61] human data to evaluate whether our method improves cross-embodiment transfer.

4.2 Experimental Results

4.2.1 AGRA Improves Action-Grounded Control

We first evaluate whether regularizing the world-action interface improves real-world execution. As shown in Figure 4, the *AGRA* model achieves an ID success rate of 80%, substantially outperforming the baseline *WAM* which obtains 34%. *AGRA* also yields stronger robustness under Semantic, Instance-Level, and Attribute Generalization, boosting performance by 27%, 32%, and 32%.

Action-head attention analysis. We visualize the cross-attention maps of action DiT in Figure 5. We use a Pick-and-Place scene containing an apple and a banana, and evaluate different instructions from the same current observation. Both the baseline *WAM* and *AGRA* can generate plausible future in which the robot arm moves toward the correct object, but they are different in how the action head reads from visual features. For example, in the instruction "put the banana in the box", the task-critical region is the interaction area between the right hand and the banana. The *WAM* often allocates large attention mass to regions that are not causally relevant to action, such as the static left hand. In contrast, *AGRA* concentrates more on this task-critical region. We quantify this effect using manually annotated ground-truth hand-object interaction masks on an evaluation subset. We report two metrics: Attention in Mask Ratio, defined as the fraction of total attention mass inside the

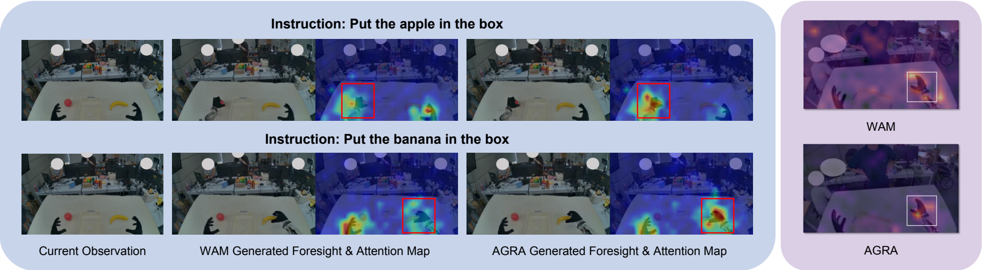


Figure 5: Qualitative analysis of action grounding. Compared with baseline WAM, AGRA directs action-head attention and causal sensitivity toward the task-critical hand-object interaction region.

Figure 6: Real-world execution cases.

Table 1: Matched Sensitivity Ratio under various causal intervention strategies.

Intervention	WAM	AGRA
Mean	8.41	**10.31**
Zero	1.95	**2.82**
Shuffle	0.99	**1.01**
Swap w/ Comp	8.36	**10.19**

interaction region, where higher is better; and Centroid Error, defined as the distance between the attention 2D centroid and the center of the annotated region in grid units, where lower is better. Our empirical results shown in Figure 4(c) indicate that *AGRA* improves both metrics, confirming that *AGRA* makes the action decoder attend more precisely to the spatial regions that determine control.

Causal intervention for hidden-state actionability. We extend the token-level intervention analysis from Section 3.2 to obtain a quantitative metric. For each annotated sample, we define the hand-object interaction token set as $\mathcal{I}$, and sample a background token set $\mathcal{B}$ outside this region. We perturb the hidden states for these two sets to obtain the action deviations $\Delta(\mathcal{I})$ and $\Delta(\mathcal{B})$. To eliminate the bias introduced by magnitude, the Matched Sensitivity Ratio is defined as $R = \Delta(\mathcal{I})/\Delta(\mathcal{B})$. A larger value of R indicates that perturbing the interaction region changes the action more than background region, implying that the hidden states are more action-grounded. We designed different intervention types, including Mean, Zero, Shuffle, and Swap with complement tokens. As shown in Table 1, *AGRA* consistently achieves a higher R than the baseline. Action-sensitivity heatmaps in Figure 5 also show the same trend qualitatively: *AGRA* concentrates high-impact regions around the task-critical hand-object contact area, while *WAM* exhibits more diffuse or background-sensitive actionability. This demonstrates *AGRA*'s robustness against task-irrelevant information, explaining its stronger generalization under OOD scenarios (e.g., changed attributes and backgrounds).

Real-world execution cases. By making visual representations more action-grounded, *AGRA* not only improves generalization by reducing the sensitivity to task-irrelevant noise, but also boosts the basic grasp success rate by focusing the action head on interaction area to enhance object localization and affordance understanding. It translates into more stable physical execution, as shown in Figure 6. The baseline *WAM* often suffers from localization errors, missing the target object due to spatial deviations. *AGRA* produces more stable grasps and better object affordance use. For elongated object such as a banana, *AGRA* accurately targets its center and executes tighter finger closure.

4.2.2 Further Analysis

a. Where Should Representation Alignment Be Applied? We next study where representation alignment should be applied inside the world model. As shown in Figure 4, aligning the 8th layer (*AGRA-DinoL8*) gives better real-world performance than aligning the 15th layer (*AGRA-DinoL15*). This finding suggests that AGRA should be applied at 1/3 of the network depth. By assigning the extraction of semantic representations to shallow layers, the deeper layers are liberated to model motion and fine-grained spatiotemporal dynamics. Enforcing static semantic alignment in deeper layers would disrupt geometric and dynamic details of world model. This conclusion is also wellsupported by REPA's experimental results, and we provide a deeper analysis in Appendix. Furthermore, we compared single-layer with multi-layer aligning. Aligning multiple layers simultaneously, as in *AGRA-DinoL4/8/12*, does not improve performance. We also tested a hierarchical alignment scheme that maps different Cosmos layers to different DINOv2 depths according to their presumed semantic abstraction, but this over-constrains the latent hierarchy and degrades performance.

b. Which Visual Representation Is Better for Action Grounding? We compare DINOv2 and SigLIP as the alignment target. They provide different representational biases: DINOv2 features are more object-centric and spatially coherent, while SigLIP features are optimized for image-text matching and emphasize global language alignment. Therefore, DINOv2 features produce clearer boundaries among objects and background, while SigLIP features are more spatially diffuse and less effective at separating the exact regions involved in manipulation. This can explain why *AGRA-DinoL8* achieves better grasping accuracy and execution stability. Theoretically, SigLIP's language-aligned latent space should bolster the policy's instruction-following capabilities. However, our experiments show that *AGRA-SiglipL8* does not improve semantic generalization. We observe that this is not because *AGRA-SiglipL8* fails to comprehend the language instructions; instead, the performance bottleneck stems from execution-level failures, failing to execute a stable grasp.

c. Semantic Information Alone Is Not Sufficient. The *AGRA-BridgeL8* variant tests whether an aligned semantic representation alone can drive precise control. In this variant, alignment is applied to 8th layer of Cosmos, and this single-layer feature is repeatedly used as the guidance input for all cross-attention layers of action DiT. This removes the multi-level predictive bridge and isolates the contribution of the aligned semantic layer. As shown in Figure 4, *AGRA-BridgeL8* performs poorly and drops to 0% success on the challenging Open-Steamer-Transfer-Bun task. The aligned 8th layer provides strong object identity and scene-layout information, but it lacks the spatial, geometric and dynamic details required for precise manipulation. This confirms that AGRA should be understood as an interface regularizer rather than a replacement for predictive world features.

d. Cross-Embodiment Generalization via Human Data. Finally, we evaluate whether *AGRA* improves the use of cross-embodiment human data from EgoDex. Human data provides broad visual and interactive diversity, but this diversity is only useful if the policy can extract manipulation-relevant structure that transfers across embodiment. Without action-grounded alignment, the hidden states of a generative model may entangle task structure with embodiment-specific appearance. Empirically, adding EgoDex data to the baseline *WAM* yields little improvement, while *AGRA* benefits substantially from human data, especially in OOD scenarios. By anchoring Cosmos hidden states to visual features with stronger semantic coherence, *AGRA* reduces the dependence on embodiment-specific cues and exposes more invariant object-centric and interaction-relevant structure.

5 Conclusion and Limitation

In this work, we studied the action-grounding gap between visual representation and action decoding in WAMs. Through attention analysis and causal interventions, we showed that action decoder may attend to task-irrelevant regions and remain sensitive to background perturbations. To address this, we proposed AGRA that regularizes world-action interface by aligning video diffusion features with spatially coherent semantic representations. Experiments on real-world manipulation demonstrate that AGRA leads to more accurate attention over hand-object interaction regions, stronger causal sensitivity to task-relevant areas, improved object localization, and more robust generalization.

Limitation. First, our real-world experiments are conducted on a limited set of tasks. Evaluation on more diverse and complex tasks would allow a more comprehensive analysis of different visual representations. For example, we did not observe clear gains from SigLIP, partly because the baseline model performs well on current Pick-and-Place tasks; tasks requiring richer semantic reasoning may better reveal the potential advantages. Second, although we identify the action-grounding problem and provide metrics and methods that can partially mitigate this phenomenon, it does not fully solve the video-action mismatch. Quantitatively characterizing this mismatch remains challenging. We leave broader task coverage and more systematic exploration as future work.

References

[1] A. Brohan, N. Brown, J. Carbajal, Y. Chebotar, J. Dabis, C. Finn, K. Gopalakrishnan, K. Hausman, A. Herzog, J. Hsu, et al. Rt-1: Robotics transformer for real-world control at scale. *arXiv preprint arXiv:2212.06817*, 2022.

[2] B. Zitkovich, T. Yu, S. Xu, P. Xu, T. Xiao, F. Xia, J. Wu, P. Wohlhart, S. Welker, A. Wahid, et al. Rt-2: Vision-language-action models transfer web knowledge to robotic control. In *Conference on Robot Learning*, pages 2165–2183. PMLR, 2023.

[3] 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 *2024 IEEE International Conference on Robotics and Automation (ICRA)*, pages 6892–6903. IEEE, 2024.

[4] 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 preprint arXiv:2506.21539*, 2025.

[5] R. Zheng, J. Wang, S. Reed, J. Bjorck, Y. Fang, F. Hu, J. Jang, K. Kundalia, Z. Lin, L. Magne, et al. Flare: Robot learning with implicit world modeling. *arXiv preprint arXiv:2505.15659*, 2025.

[6] H. Bi, H. Tan, S. Xie, Z. Wang, S. Huang, H. Liu, R. Zhao, Y. Feng, C. Xiang, Y. Rong, et al. Motus: A unified latent action world model. *arXiv preprint arXiv:2512.13030*, 2025.

[7] M. Team, C. Xiang, F. Bao, H. Liu, H. Tan, H. Bi, J. Li, J. Liu, J. Pang, K. Jing, et al. Motubrain: An advanced world action model for robot control. *arXiv preprint arXiv:2604.27792*, 2026.

[8] J. Won, K. Lee, H. Jang, D. Kim, and J. Shin. Dual-stream diffusion for world-model augmented vision-language-action model. *arXiv preprint arXiv:2510.27607*, 2025.

[9] H. Wu, Y. Jing, C. Cheang, G. Chen, J. Xu, X. Li, M. Liu, H. Li, and T. Kong. Unleashing large-scale video generative pre-training for visual robot manipulation. In *International Conference on Learning Representations*, volume 2024, pages 10641–10662, 2024.

[10] S. Ye, Y. Ge, K. Zheng, S. Gao, S. Yu, G. Kurian, S. Indupuru, Y. L. Tan, C. Zhu, J. Xiang, et al. World action models are zero-shot policies. *arXiv preprint arXiv:2602.15922*, 2026.

[11] L. Li, Q. Zhang, Y. Luo, S. Yang, R. Wang, F. Han, M. Yu, Z. Gao, N. Xue, X. Zhu, et al. Causal world modeling for robot control. *arXiv preprint arXiv:2601.21998*, 2026.

[12] M. J. Kim, Y. Gao, T.-Y. Lin, Y.-C. Lin, Y. Ge, G. Lam, P. Liang, S. Song, M.-Y. Liu, C. Finn, et al. Cosmos policy: Fine-tuning video models for visuomotor control and planning. *arXiv preprint arXiv:2601.16163*, 2026.

[13] J. Pai, L. Achenbach, V. Montesinos, B. Forrai, O. Mees, and E. Nava. mimic-video: Video-action models for generalizable robot control beyond vlas. *arXiv preprint arXiv:2512.15692*, 2025.

[14] T. Brooks, B. Peebles, C. Holmes, W. DePue, Y. Guo, L. Jing, D. Schnurr, J. Taylor, T. Luhman, E. Luhman, et al. Video generation models as world simulators. *OpenAI Blog*, 1(8):1, 2024.

[15] T. Wan, A. Wang, B. Ai, B. Wen, C. Mao, C.-W. Xie, D. Chen, F. Yu, H. Zhao, J. Yang, et al. Wan: Open and advanced large-scale video generative models. *arXiv preprint arXiv:2503.20314*, 2025.

[16] 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 preprint arXiv:2501.03575*, 2025.

[17] Y. Hu, Y. Guo, P. Wang, X. Chen, Y.-J. Wang, J. Zhang, K. Sreenath, C. Lu, and J. Chen. Video prediction policy: A generalist robot policy with predictive visual representations. *arXiv preprint arXiv:2412.14803*, 2024.

[18] S. Yu, S. Kwak, H. Jang, J. Jeong, J. Huang, J. Shin, and S. Xie. Representation alignment for generation: Training diffusion transformers is easier than you think. *arXiv preprint arXiv:2410.06940*, 2024.

[19] B. Zheng, N. Ma, S. Tong, and S. Xie. Diffusion transformers with representation autoencoders. *arXiv preprint arXiv:2510.11690*, 2025.

[20] S. Jha, A. Zholus, S. Chandar, et al. Reconstruction or semantics? what makes a latent space useful for robotic world models. *arXiv preprint arXiv:2605.06388*, 2026.

[21] W. Yan, Y. Zhang, P. Abbeel, and A. Srinivas. Videogpt: Video generation using vq-vae and transformers. *arXiv preprint arXiv:2104.10157*, 2021.

[22] W. Hong, M. Ding, W. Zheng, X. Liu, and J. Tang. Cogvideo: Large-scale pretraining for text-to-video generation via transformers. *arXiv preprint arXiv:2205.15868*, 2022.

[23] A. Blattmann, R. Rombach, H. Ling, T. Dockhorn, S. W. Kim, S. Fidler, and K. Kreis. Align your latents: High-resolution video synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 22563–22575, 2023.

[24] A. Blattmann, T. Dockhorn, S. Kulal, D. Mendelevitch, M. Kilian, D. Lorenz, Y. Levi, Z. English, V. Voleti, A. Letts, et al. Stable video diffusion: Scaling latent video diffusion models to large datasets. *arXiv preprint arXiv:2311.15127*, 2023.

[25] Z. Zhu, X. Wang, W. Zhao, C. Min, B. Li, N. Deng, M. Dou, Y. Wang, B. Shi, K. Wang, et al. Is sora a world simulator? a comprehensive survey on general world models and beyond. *arXiv preprint arXiv:2405.03520*, 2024.

[26] J. Bruce, M. D. Dennis, A. Edwards, J. Parker-Holder, Y. Shi, E. Hughes, M. Lai, A. Mavalankar, R. Steigerwald, C. Apps, et al. Genie: Generative interactive environments. In *Forty-first International Conference on Machine Learning*, 2024.

[27] R. Feng, H. Zhang, Z. Shu, Z. Yang, L. Tang, Z. Wang, A. Zheng, J. Xiao, Z. Liu, R. Chu, et al. The matrix: Infinite-horizon world generation with real-time moving control. *Advances in Neural Information Processing Systems*, 38:87318–87344, 2026.

[28] J. Yu, J. Bai, Y. Qin, Q. Liu, X. Wang, P. Wan, D. Zhang, and X. Liu. Context as memory: Scene-consistent interactive long video generation with memory retrieval. In *Proceedings of the SIGGRAPH Asia 2025 Conference Papers*, pages 1–11, 2025.

[29] S. Huang, J. Wu, Q. Zhou, S. Miao, and M. Long. Vid2world: Crafting video diffusion models to interactive world models. *arXiv preprint arXiv:2505.14357*, 2025.

[30] X. Chi, P. Jia, C.-K. Fan, X. Ju, W. Mi, K. Zhang, Z. Qin, W. Tian, K. Ge, H. Li, et al. Wow: Towards a world omniscient world model through embodied interaction. *arXiv preprint arXiv:2509.22642*, 2025.

[31] G. Team, A. Ye, B. Wang, C. Ni, G. Huang, G. Zhao, H. Li, J. Zhu, K. Li, M. Xu, et al. Gigaworld-0: World models as data engine to empower embodied ai. *arXiv preprint arXiv:2511.19861*, 2025.

[32] S. Gao, W. Liang, K. Zheng, A. Malik, S. Ye, S. Yu, W.-C. Tseng, Y. Dong, K. Mo, C.-H. Lin, et al. Dreamdojo: A generalist robot world model from large-scale human videos. *arXiv preprint arXiv:2602.06949*, 2026.

[33] Y. Shang, X. Zhang, Y. Tang, L. Jin, C. Gao, W. Wu, and Y. Li. Roboscape: Physics-informed embodied world model. *Advances in Neural Information Processing Systems*, 38: 63674–63698, 2026.

[34] J. Jang, S. Ye, Z. Lin, J. Xiang, J. Bjorck, Y. Fang, F. Hu, S. Huang, K. Kundalia, Y.-C. Lin, et al. Dreamgen: Unlocking generalization in robot learning through video world models. *arXiv preprint arXiv:2505.12705*, 2025.

[35] Y. Li, Y. Zhu, J. Wen, C. Shen, and Y. Xu. Worldeval: World model as real-world robot policies evaluator. *arXiv preprint arXiv:2505.19017*, 2025.

[36] J. Quevedo, A. K. Sharma, Y. Sun, V. Suryavanshi, P. Liang, and S. Yang. Worldgym: World model as an environment for policy evaluation. *arXiv preprint arXiv:2506.00613*, 2025.

[37] Y. Guo, L. X. Shi, J. Chen, and C. Finn. Ctrl-world: A controllable generative world model for robot manipulation. *arXiv preprint arXiv:2510.10125*, 2025.

[38] C. Finn and S. Levine. Deep visual foresight for planning robot motion. In *2017 IEEE international conference on robotics and automation (ICRA)*, pages 2786–2793. IEEE, 2017.

[39] 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. *Advances in neural information processing systems*, 36:9156–9172, 2023.

[40] Y. Feng, H. Tan, X. Mao, C. Xiang, G. Liu, S. Huang, H. Su, and J. Zhu. Vidar: Embodied video diffusion model for generalist manipulation. *arXiv preprint arXiv:2507.12898*, 2025.

[41] J. Liang, P. Tokmakov, R. Liu, S. Sudhakar, P. Shah, R. Ambrus, and C. Vondrick. Video generators are robot policies. *arXiv preprint arXiv:2508.00795*, 2025.

[42] 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 preprint arXiv:2508.05635*, 2025.

[43] T. Ma, J. Zheng, Z. Wang, C. Jiang, A. Cui, J. Liang, and S. Yang. Dit4dit: Jointly modeling video dynamics and actions for generalizable robot control. *arXiv preprint arXiv:2603.10448*, 2026.

[44] T. Yuan, Z. Dong, Y. Liu, and H. Zhao. Fast-wam: Do world action models need test-time future imagination? *arXiv preprint arXiv:2603.16666*, 2026.

[45] W. Peebles and S. Xie. Scalable diffusion models with transformers. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 4195–4205, 2023.

[46] C. Wei, K. Mangalam, P.-Y. Huang, Y. Li, H. Fan, H. Xu, H. Wang, C. Xie, A. Yuille, and C. Feichtenhofer. Diffusion models as masked autoencoders. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 16284–16294, 2023.

[47] X. Chen, Z. Liu, S. Xie, and K. He. Deconstructing denoising diffusion models for self-supervised learning. In *International Conference on Learning Representations*, volume 2025, pages 55458–55472, 2025.

[48] W. Xiang, H. Yang, D. Huang, and Y. Wang. Denoising diffusion autoencoders are unified self-supervised learners. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 15802–15812, 2023.

[49] X. Yang and X. Wang. Diffusion model as representation learner. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 18938–18949, 2023.

[50] M. Caron, H. Touvron, I. Misra, H. Jégou, J. Mairal, P. Bojanowski, and A. Joulin. Emerging properties in self-supervised vision transformers. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 9650–9660, 2021.

[51] 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 preprint arXiv:2304.07193*, 2023.

[52] X. Zhai, B. Mustafa, A. Kolesnikov, and L. Beyer. Sigmoid loss for language image pre-training. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 11975–11986, 2023.

[53] J. Singh, X. Leng, Z. Wu, L. Zheng, R. Zhang, E. Shechtman, and S. Xie. What matters for representation alignment: Global information or spatial structure? *arXiv preprint arXiv:2512.10794*, 2025.

[54] X. Leng, J. Singh, Y. Hou, Z. Xing, S. Xie, and L. Zheng. Repa-e: Unlocking vae for end-to-end tuning of latent diffusion transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 18262–18272, 2025.

[55] S. Hwang, H. Jang, K. Kim, M. Park, and J. Choo. Cross-frame representation alignment for fine-tuning video diffusion models. *arXiv preprint arXiv:2506.09229*, 2025.

[56] X. Zhang, J. Liao, S. Zhang, F. Meng, X. Wan, J. Yan, and Y. Cheng. Videorepa: Learning physics for video generation through relational alignment with foundation models. *Advances in Neural Information Processing Systems*, 38:122647–122676, 2026.

[57] M. Assran, A. Bardes, D. Fan, Q. Garrido, R. Howes, M. Muckley, A. Rizvi, C. Roberts, K. Sinha, A. Zholus, et al. V-jepa 2: Self-supervised video models enable understanding, prediction and planning. *arXiv preprint arXiv:2506.09985*, 2025.

[58] G. Zhou, H. Pan, Y. LeCun, and L. Pinto. Dino-wm: World models on pre-trained visual features enable zero-shot planning. *arXiv preprint arXiv:2411.04983*, 2024.

[59] Y. Lipman, R. T. Chen, H. Ben-Hamu, M. Nickel, and M. Le. Flow matching for generative modeling. *arXiv preprint arXiv:2210.02747*, 2022.

[60] H. Abdi and L. J. Williams. Principal component analysis. *Wiley interdisciplinary reviews: computational statistics*, 2(4):433–459, 2010.

[61] R. Hoque, P. Huang, D. J. Yoon, M. Sivapurapu, and J. Zhang. Egodex: Learning dexterous manipulation from large-scale egocentric video. *arXiv preprint arXiv:2505.11709*, 2025.

[62] Y. Chen, Y. Ge, H. Zhou, M. Ding, Y. Ge, and X. Liu. Dial: Decoupling intent and action via latent world modeling for end-to-end vla. *arXiv preprint arXiv:2603.29844*, 2026.

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

[64] 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*, 44(10-11):1684–1704, 2025.

[65] J. Lyu, K. Liu, X. Zhang, H. Liao, Y. Feng, W. Zhu, T. Shen, J. Chen, J. Zhang, Y. Dong, et al. Lda-1b: Scaling latent dynamics action model via universal embodied data ingestion. *arXiv preprint arXiv:2602.12215*, 2026.

[66] NVIDIA GEAR Team, A. Azzolini, J. Bjorck, V. Blukis, et al. Gr00t n1.6: An improved open foundation model for generalist humanoid robots. `https://research.nvidia.com/labs/gear/gr00t-n1_6/`, December 2025.

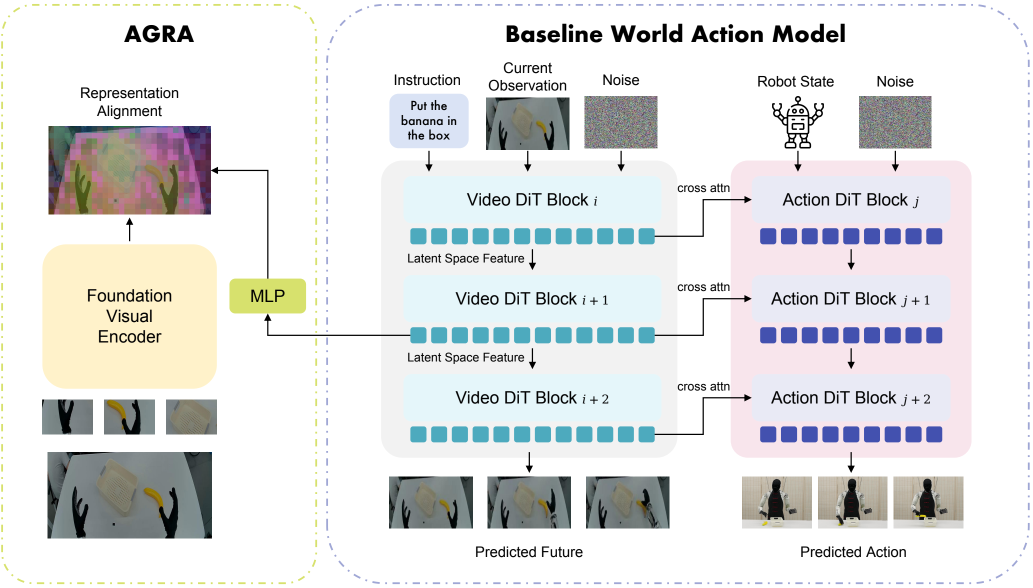


Figure 7: Architecture of baseline World Action Model and the proposed Action-Grounded Representation Alignment.

A Details of Baseline World Action Model Architecture

A.1 Dual-DiT Architecture

Video DiT. The video branch is a flow-matching video DiT initialized from Cosmos-Predict2.5-2B. It predicts future visual latents conditioned on the current observation o_0 and the language instruction c. Let $\mathbf{o}_{1:T}$ denote the future video frames and let $\mathbf{z}_{1:T_v} = E_{\text{vae}}(\mathbf{o}_{1:T})$ be the corresponding latent sequence encoded by the video tokenizer. The video DiT predicts the corresponding velocity field at a noise level $\tau_v \in [0, 1]$:

$$\begin{aligned} \mathbf{z}_{1:T_v}^{\tau_v} &= (1-\tau_v)\mathbf{z}_{1:T_v} + \tau_v \boldsymbol{\epsilon}_v, \qquad \boldsymbol{\epsilon}_v \sim \mathcal{N}(\mathbf{0}, \mathbf{I}), \\ \mathbf{v}_\theta^{\text{vid}} &= f_\theta^{\text{vid}}(\mathbf{z}_{1:T_v}^{\tau_v}, o_0, c, \tau_v), \end{aligned} \tag{8}$$

and is optimized with the flow-matching objective:

$$\mathcal{L}_{\text{vid}} = \mathbb{E}_{\mathbf{z}, \boldsymbol{\epsilon}_v, \tau_v} \left[\left\| f_\theta^{\text{vid}}(\mathbf{z}_{1:T_v}^{\tau_v}, o_0, c, \tau_v) - (\boldsymbol{\epsilon}_v - \mathbf{z}_{1:T_v}) \right\|_2^2 \right]. \tag{9}$$

The timestep is sampled using the default sampling strategy of Cosmos-Predict2.5-2B. Since the video branch is used as a source of predictive visual representations, we expose its intermediate hidden states to the action branch. We denote the hidden states of the ℓ-th video DiT layer as:

$$\mathbf{H}_\ell^{\text{vid}} = \{h_{\ell,i}^{\text{vid}}\}_{i=1}^{N_v} \in \mathbb{R}^{N_v \times d_v}, \tag{10}$$

where N_v is the number of video latent tokens and d_v is the feature dimension.

Action DiT. We employ the flow-matching action DiT from the Gr00T-N1. This action branch generates a future action chunk from pure noise, conditioned on predictive visual representations and robot proprioceptive state s_0. By sampling the action noise level τ_a from a Beta distribution, we construct a noisy action chunk from the ground-truth action sequence $a_{1:K}$:

$$a_{1:K}^{\tau_a} = (1-\tau_a)a_{1:K} + \tau_a \epsilon_a, \qquad \epsilon_a \sim \mathcal{N}(0, I), \tag{11}$$

and the noisy action chunk is concatenated with robot proprioceptive state s_0 to serve as the action DiT input sequence.

The action DiT alternates self-attention over action / state tokens and cross-attention to world-model guidance features. At the j-th cross-attention layer, it updates the action / state tokens as:

$$X_j^{\mathrm{act}} = \mathrm{CrossAttn}_j \left(Q = X_{j-1}^{\mathrm{act}}, K = \mathcal{G}_j, V = \mathcal{G}_j \right). \tag{12}$$

The action branch predicts action velocity field and is optimized with the flow-matching objective:

$$\begin{gathered} v_\phi^{\mathrm{act}} = f_\phi^{\mathrm{act}} \left(a_{1:K}^{\tau_a}, s_0, \tau_a, \mathcal{G} \right), \\ \mathcal{L}_{\mathrm{act}} = \mathbb{E}_{a, \epsilon_a, \tau_a} \left[\left\| f_\phi^{\mathrm{act}} \left(a_{1:K}^{\tau_a}, s_0, \tau_a, \mathcal{G} \right) - \left(\epsilon_a - a_{1:K} \right) \right\|_2^2 \right], \end{gathered} \tag{13}$$

where $\mathcal{G} = \{\mathcal{G}_j\}_{j=1}^N$ denotes the set of guidance features used by the N action cross-attention layers. The concrete construction of $\mathcal{G}$ during training and inference is described in Sections A.2 and A.3.

Bridge Method. This paragraph demonstrates how the hidden states of Video DiT are fed into Action DiT. A common strategy [43] is to select a single video layer $\ell^\star$ and reuse its hidden states for all action cross-attention layers, but it discards the hierarchical structure of the video DiT, which encodes complementary information at different levels. We therefore use a multi-layer bridge. Assume the video DiT contains M layers and the action DiT contains N cross-attention layers. For the j-th action cross-attention layer, we select a video layer uniformly from the depth of the video DiT:

$$\ell_j = \left\lfloor \frac{j(M-1)}{N-1} \right\rceil, \qquad j = 0, \ldots, N-1, \tag{14}$$

The selected video hidden states are then projected to the action feature dimension:

$$\mathcal{G}_j = \mathrm{Proj}_j \left(\mathbf{H}_{\ell_j}^{\mathrm{vid}} \right) \in \mathbb{R}^{N_v \times d_{\mathrm{act}}}, \tag{15}$$

and the resulting guidance features are injected into the action DiT through the cross-attention operation above. This bridge design allows the action head to access multi-level predictive representations from the video world model while keeping the action decoder lightweight.

A.2 Action Sampling

Prior video-action policies [17, 13] have shown that early denoising representations at high noise levels of video diffusion models provide useful predictive visual features for downstream robotic control. In our preliminary experiments, action policies conditioned on single-step video denoising features achieve higher task success than those conditioned on features extracted after four video denoising steps. This suggests that high-noise video representations may preserve global task dynamics and future motion cues that are more useful for action prediction than low-noise representations, which tend to focus more on visual details.

Therefore, during inference, we run the video DiT once on future latent tokens at a fixed high-noise level $\tau_v^{\mathrm{cond}} = 1$, conditioned on the current observation o_0 and language instruction c. We use the layer-wise hidden-state extractor:

$$\mathbf{H}_\ell^{\mathrm{vid}} = f_{\theta,\ell}^{\mathrm{vid}} \left(\mathbf{z}_{1:T_v}^1, o_0, c, \tau_v^{\mathrm{cond}} = 1 \right), \qquad \ell = 1, \ldots, M. \tag{16}$$

The guidance feature for the j-th action cross-attention layer is then computed using the bridge defined in Eq. 15:

$$\mathcal{G}_j = \mathrm{Proj}_j \left(\mathbf{H}_{\ell_j}^{\mathrm{vid}} \right), \qquad j = 1, \ldots, N. \tag{17}$$

The lightweight action DiT then performs multi-step denoising in the action space, conditioned on $\mathcal{G} = \{\mathcal{G}_j\}_{j=1}^N$, to generate the continuous action chunk. This single-step video and multi-step action sampling strategy reduces inference latency while retaining the predictive guidance provided by the video world model.


Pick & Place Task

Instruction: pick the eggplant from the cutting board and place it in the cardboard box

Articulated tasks

Instruction: pick up the can, place it into the drawer and close the drawer

Figure 8: Evaluation results of AGRA on the RoboCasa GR1 benchmark. Videos illustrate representative policy execution trajectories generated by AGRA.

A.3 Optimization Strategy

Training WAM involves two flow-matching objectives: the video prediction loss $\mathcal{L}_{\text{vid}}$ defined in Eq. 9 and the action prediction loss $\mathcal{L}_{\text{act}}$ defined in Eq. 13. The full objective is:

$$\mathcal{L}_{\text{WAM}} = \mathcal{L}_{\text{vid}} + \lambda_{\text{act}} \mathcal{L}_{\text{act}}, \tag{18}$$

where λ_{act} balances the video prediction objective and the action prediction objective.

For the video generation module, we follow the standard diffusion training paradigm. At each training step, the noise level τ_v is randomly sampled from [0, 1]. This exposes the video model to all noise levels, forcing it to learn the full denoising trajectory required to synthesize future frames. The loss is computed following Eq. 8 and Eq. 9.

For the action prediction objective, the action head first requires predictive visual representations from the world model. To keep the visual input to the action module deterministic and consistent with inference, we compute these representations using a separate video DiT forward pass at the fixed high-noise level $\tau_v^{\text{cond}} = 1$, which is the same as the inference process in Section A.2:

$$\mathbf{H}_\ell^{\text{vid}} = f_{\theta,\ell}^{\text{vid}}\left(\mathbf{z}_{1:T_v}^1, o_0, c, \tau_v^{\text{cond}} = 1\right), \qquad \ell = 1, \ldots, M. \tag{19}$$

The multi-layer bridge then constructs the guidance features used by the action DiT:

$$\mathcal{G}_j = \text{Proj}_j\left(\mathbf{H}_{\ell_j}^{\text{vid}}\right), \qquad \mathcal{G} = \{\mathcal{G}_j\}_{j=1}^N. \tag{20}$$

Conditioned on $\mathcal{G}$, the action DiT is optimized with $\mathcal{L}_{\text{act}}$ as defined in Eq. 13. By fixing $\tau_v^{\text{cond}} = 1$ for the visual representations used by the action head, the model trains the action decoder under the same representation distribution used at inference time.

B Simulation Experiment

B.1 RoboCasa GR1 Tabletop Simulation

We conduct simulation experiments on the RoboCasa benchmark utilizing the GR1 robot. The evaluation suite comprises 24 tabletop tasks, each assessed over 50 episodes. This suite includes 18 "**Pick & Place**" rearrangement tasks, where the robot follows language instructions to move objects between containers, and 6 "**Articulated tasks**" that involve more complex interactions such as placing objects inside and subsequently closing cabinets, drawers, or microwaves. We represent the robot's state and actions using 29 joint-space degrees of freedom (DoF), including the dual arms

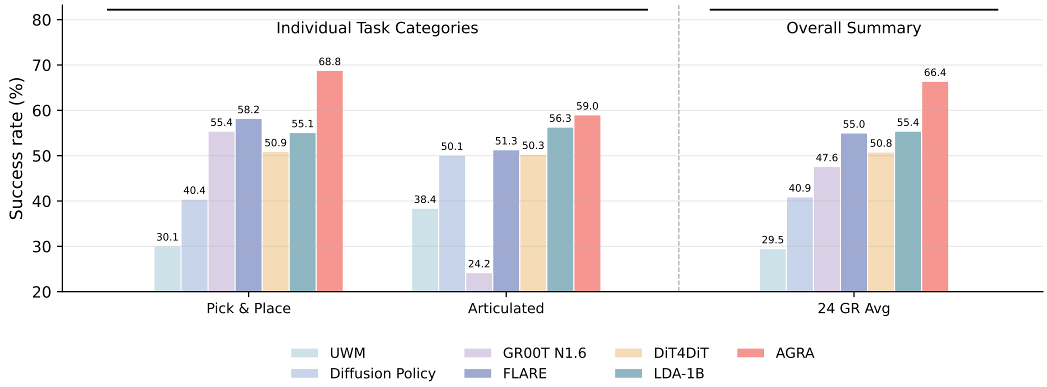


Figure 9: RoboCasa-GR1 tabletop tasks evaluation results with full training data.

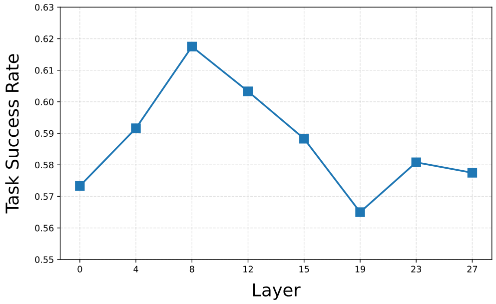


Figure 10: Layer selection for AGRA. The figure visualizes the downstream manipulation performance when applying AGRA to different layers of Cosmos.

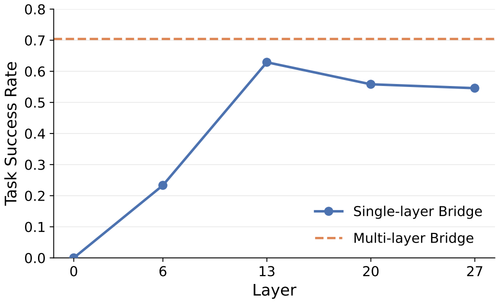


Figure 11: Results of single-layer bridge between dual DiTs. The performance of all single-layer configurations is consistently inferior to that of the multi-layer bridge method.

(14), hands (12), and waist (3). For fair comparison with other SOTA methods, we use the full data regime including 24,000 trajectories (1000 per task) and involve 80,000 training steps. For ablation studies, we use few-shot regime with a 10% subset of 2,400 trajectories (100 per task) with 40,000 training steps, and we also evaluate in three OOD testing scenarios from DIAL [62]: (1) Unseen Appearance (18 tasks), (2) Unseen Object Types (32 tasks), (3) Unseen Combinations (14 tasks).

B.2 Experimental Results

B.2.1 Comparison Against State-Of-The-Art Policies

We benchmark AGRA against state-of-the-art policies on the comprehensive RoboCasa GR1 Tabletop simulation suite, including UWM [63], a transformer unifying video modeling, forward / inverse dynamics and robot policy; Diffusion Policy [64], which represents robot policy as denoising diffusion process; FLARE [5], a flow-matching framework with future latent alignment; DiT4DiT [43], a video action model coupling video diffusion model with an action head; and LDA-1B [65], which learns scalable dynamics in DINO latent space. For VLA, we include GR00T-N1.6 [66] due to its widespread adoption and established efficacy in the robotics community. As illustrated in Figure 9, AGRA achieves an overall success rate of 66.4%, outperforming the strong VLA baseline, GR00T, by a significant absolute margin of 18.8%. When compared with contemporary predictive and generative control methods, including FLARE, DiT4DiT, and LDA-1B—AGRA, consistently demonstrates a performance improvement exceeding 10%.

Table 2: Ablation study on the RoboCasa GR1 tabletop benchmark in the few-shot regime.

Variant	**ID**	**Unseen App.**	**Unseen Obj. Types**	**Unseen Comb.**
Freeze backbone	52.89	49.33	43.87	51.42
WAM	58.41	53.77	43.18	59.57
AGRA-SharedDenoisingPass	58.24	**57.77**	41.87	58.57
AGRA-VideoPass	60.41	55.33	44.56	60.14
AGRA-ActionCondPass	**61.75**	56.55	**45.31**	**66.28**

B.2.2 Ablation study

To isolate the impact of AGRA's architectural choices, we construct several controlled variants.

- *Freeze backbone*: It keeps Cosmos frozen and only optimizes action head.
- *WAM*: It is the baseline model without representation alignment.
- *AGRA-SharedDenoisingPass*: It removes the separate action-conditioning forward pass. During training, the action head directly consumes the Cosmos hidden states produced by the video denoising pass at the sampled diffusion timestep τ_v, i.e., the same forward pass used to compute $\mathcal{L}_{vid}$.
- *AGRA-VideoPass*: It keeps the default action-conditioning branch, but applies AGRA alignment loss to the video denoising pass. The aligned hidden states are taken from the Cosmos forward pass at the sampled video noise level τ_v, the same pass used to compute $\mathcal{L}_{vid}$.
- *AGRA-ActionCondPass*: The AGRA alignment loss is applied to the action-conditioning pass, where Cosmos is evaluated at the fixed high-noise level $\tau_v^{cond} = 1$, and the resulting hidden states are used by the action head to compute $\mathcal{L}_{act}$.

Importance of video-action joint modeling. Freezing the world model backbone after pure-video finetuning yields suboptimal performance, establishing a notably low performance ceiling. We observe that under the *Freeze backbone* setting, the action head tends to overfit to proprioceptive states while ignoring the visual signals. This indicates that purely generative pretraining is insufficient for deriving robust control policies. Jointly optimizing the world model and the action head allows the action gradients to backpropagate into the world model, which is critical for morphing the observation-reconstruction hidden states into an action-conditioned representation space.

Effectiveness of AGRA. Removing AGRA from the jointly trained model (*WAM*) leads to a deterioration across all metrics, with a particularly steep decline in out-of-distribution generalization, dropping from 66.28 to 59.57 on Unseen Combinations. Without explicit semantic anchoring, the generative latents remain heavily entangled with high-frequency rendering details. By aligning the world model's hidden states with DINOv2 features, AGRA provides a structural regularization that grounds the representations, enhancing the action decoder's ability to generalize across novel object attributes and spatial configurations. However, AGRA shows marginal gains in simulation compared to real-world settings. Failure case analysis indicates that in simpler simulated backgrounds, the baseline model often grasps the object successfully but suffers from post-grasp slippage, which cannot be solved purely through visual improvements.

Layer selection for AGRA. As shown in Figure 10, we also evaluated the impact of applying AGRA constraints at different layers in simulation benchmark. As the application depth increases, the downstream task success rate initially rises before declining, peaking at Layer 8. Consistent with the real-world experiments in Section 4.2.2 (a), this reaffirms that applying AGRA at shallow layers, approximately one third of the total depth of the world model, yields optimal performance.

Train-inference consistency in denoising. Our ablation on the denoising timestep strategy reveals that using the shared video denoising pass for action conditioning, i.e., *AGRA-SharedDenoisingPass*, yields inferior results compared to the default configuration. By forcing the world model to perform a separate forward pass at pure noise to extract hidden states for the action head in training, the model explicitly enforces train-inference consistency. This design constraint focuses the action head's learning capacity on the high-noise video representations that it will consume at inference time.

Targeted application of AGRA. We investigate the specific placement of the AGRA objective. *AGRA-VideoPass*, which applies alignment during the random-timestep forward pass used for calculating the diffusion loss, fails to match the performance of applying AGRA to the pure-noise forward pass dedicated to the action head, i.e., *AGRA-ActionCondPass*. Imposing representation alignment directly on the exact hidden states consumed by the action decoder provides a more explicit and effective semantic grounding, leading to superior empirical robustness.

Multi-layer feature aggregation. Using the same setup, we contrasted single- vs. multi-layer bridges between the Video and Action DiTs. Figure 11 illustrates that although mid-to-deep layers perform best within single-layer setups, they are consistently outperformed by the multi-layer approach. This validates that robotic manipulation requires a hierarchical synthesis of information: multi-layer aggregation is indispensable for simultaneously capturing the high-level object-centric topologies and the fine-grained, low-level spatial dynamics necessary for precise continuous control.

C Implementation Details

C.1 Model Framework

We build our policy on top of the Cosmos-Predict-2.5-2B video diffusion backbone. The model takes 17-frame video clips at a resolution of 192 x 336, which are encoded into 5 latent frames using a temporal compression ratio of 4. During training, the first latent frame is used as the visual condition. We extract intermediate visual features from Cosmos transformer layers 0, 4, 8, 12, 15, 19, 23, and 27. These multi-layer features are passed through a project module with layer normalization and a 4-layer self-attention transformer. The action head is a flow-matching diffusion policy head conditioned on the processed Cosmos features. It uses a 8-block DiT with 32 attention heads, hidden size 1024, and cross-attention dimension 2048. Each block contains one self-attention layer and one cross-attention layer. Therefore, the action head has 8 cross-attention layers in total, which are in one-to-one correspondence with the 8 selected Cosmos feature layers. In the default setting, we apply AGRA on Cosmos layer 8. The target representation is extracted from a frozen DINOv2 encoder with input size 448 x 448. At inference time, Cosmos takes the current observation image as the first-frame condition and pure noise for the remaining latent frames. After one denoising step, Cosmos hidden states are extracted and used to condition the action head. The action head then performs 4 flow-matching denoising steps to produce the final action trajectory.

C.2 Real-World Experimental Setup

C.2.1 Dataset Collection

In real-world experiment, we design two tasks:

- **Pick-and-Place**: The robot must follow the instruction to pick up an object and place it into a container.
- **Open-Steamer-Transfer-Bun**: Expanding on the basic Pick-and-Place operation, this task requires the robot to use its left hand to remove and set aside the steamer lid, and its right hand to pick the bun from inside the steamer and transfer it to a plate on the right. This is a more challenging bimanual cooperative task.

For pretraining, we collect a real-world robotic dataset in an industrial factory setting, containing 40k trajectories (92 hours), for the Pick-and-Place task. We additionally use a basic Pick-and-Place subset from EgoDex, which contributes 37k human demonstration trajectories (42 hours). The joint pretraining dataset is used to train the model for 60,000 steps. For laboratory deployment, we collect fine-tuning data: 560 trajectories (1.5 hours) for Pick-and-Place task with a wide variety of objects, and 180 trajectories for Open-Steamer-Transfer-Bun task with 3 types of buns and plates of different colors. The laboratory dataset is used for 2,000 fine-tuning steps.

C.2.2 Evaluation

To comprehensively validate the robustness of the models in physical deployments, we establish an in-distribution (ID) evaluation regime and three out-of-distribution (OOD) generalization regimes. All evaluations are conducted on the real humanoid robot, and each trial is counted as successful only when the robot completes the instructed manipulation goal.

- **In-Distribution (ID):** Evaluated on the Pick-and-Place task, this regime measures the policy's ability to manipulate objects that have been seen in training corpus. We select 10 seen objects and place each object at 5 pre-designed tabletop positions that cover different reachable regions and relative spatial configurations. This results in 50 real-world trials.
- **Semantic Generalization (OOD):** Evaluated on the Pick-and-Place task, the robot is required to accurately grasp a specified object from several distractors or place it into a designated container following the user's instruction. We design 8 scenarios in which the robot must pick a designated object from several distractors, and 2 scenarios in which the robot must place an object into a specified target container. For each scenario, we specify two candidate targets or containers, denoted as a and b. We evaluate both instructions, e.g., picking a and picking b, or placing the object into container a and container b. We then swap the spatial positions of a and b and repeat the two instructions, which tests whether the policy follows the semantic instruction rather than a fixed spatial bias. Therefore, each scenario contains 4 trials, giving 40 trials in total.
- **Instance-Level Generalization (OOD):** Evaluated on the Pick-and-Place task, this regime tests whether the policy can generalize to object instances that are unseen during training. We select 10 novel objects that are not included in the training dataset, and evaluate each object at the same 5 pre-designed tabletop positions. This produces 50 trials in total.
- **Attribute Generalization (OOD):** Conducted on the Open-Steamer-Transfer-Bun task, this regime evaluates robustness to visual attribute shifts in both objects and surrounding scene. We construct 4 plate-variation settings by changing plate type, and 1 background-variation setting by changing the tablecloth color. Under each setting, we evaluate 8 different buns with variations in shape, color, and texture pattern. This yields 40 trials.

C.3 Training Details

We find that embodiment-domain pure-video training of Cosmos is a critical initialization step before video-action joint optimization. In preliminary experiments, directly starting from the original Cosmos checkpoint and jointly training the video branch with the action head causes a sharp drop in downstream success rate. Therefore, we first train Cosmos with only the video denoising objective for approximately 4,000-5,000 steps on the corresponding dataset, and use the resulting checkpoint, rather than the original Cosmos weights, to initialize the video branch for video-action joint training. We use a total batch size of 256 and optimize the model with a cosine learning-rate schedule with a 5% warm-up phase. All training runs are conducted on 32 GPUs, each with 140GB memory. The total training objective consists of the Cosmos denoising loss $\mathcal{L}_{\text{vid}}$, the action denoising loss $\mathcal{L}_{\text{act}}$, and the AGRA loss $\mathcal{L}_{\text{AGRA}}$, with loss weights 1.0, 1.0, and 0.01, respectively.

For real-world experiments, after the initial pure-video adaptation stage, we train the full video-action model on the joint pretraining dataset for 60,000 steps. During this stage, the learning rate of Cosmos is set to 1×10^{-5}, while the learning rate of the action head is set to 1×10^{-4}. We then fine-tune the model on the task-specific fine-tuning datasets for 2,000 steps. During fine-tuning, Cosmos is frozen and only the action head is updated. We use an action horizon of $K = 48$ for a single action chunk. Since Cosmos predicts 4 future latent frames, corresponding to 16 RGB frames, we

Figure 12: Additional real-world execution cases of AGRA. The trajectories show representative AGRA rollouts under different evaluation settings, including in-distribution pick-and-place, semantic generalization, instance-level generalization, and attribute generalization.

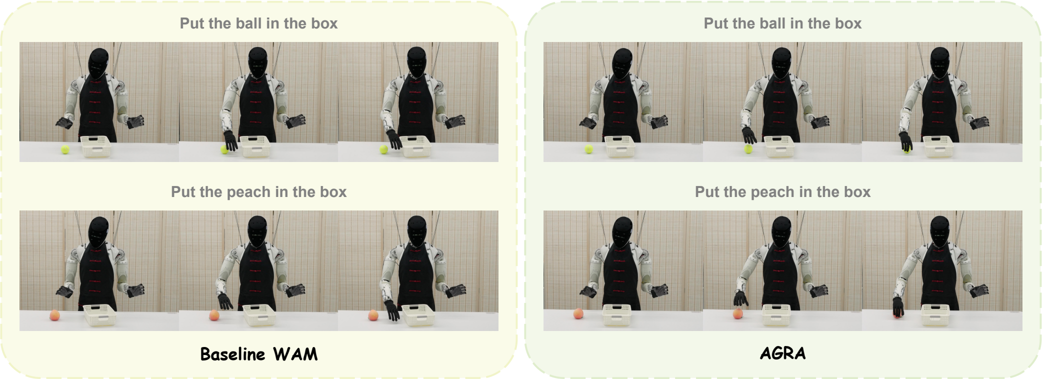


Figure 13: Real-world comparison between WAM and AGRA. Given the instructions “put the ball in the box” and “put the peach in the box”, WAM fails to localize the target accurately, while AGRA reaches and grasps the correct object region.

sample the video frames every 3 frames when pairing video targets with actions. This makes the temporal span of the 16-frame video target consistent with the 48-step action chunk.

For simulation experiments, the model is trained for 80,000 steps in the full-data regime, and trained for 40,000 steps in the few-shot regime. The learning rates of both Cosmos and the action head are set to 1×10^{-4}. We set the action chunk horizon to $K = 16$. The corresponding video target contains 16 RGB frames, so no temporal interval sampling is required.

D Additional Experimental Results

D.1 Real-World Execution Cases

We provide additional real-world execution cases of AGRA in Figure 12. The examples cover both ID and OOD evaluation settings. In the OOD settings, AGRA is evaluated under semantic generalization, instance-level generalization, and attribute generalization. These cases show that AGRA can follow different language instructions, distinguish target containers or objects, and complete the instructed manipulation under variations in object instances, visual attributes, and scene appearance. Figure 13 compares AGRA with the baseline WAM in real-world execution. Under the instructions “put the ball in the box” and “put the peach in the box”, the baseline WAM fails to localize the target object accurately and produces spatially biased grasps. In contrast, AGRA consistently identifies the correct target region and executes more precise reaching and grasping motions.

D.2 Analysis for Representation Alignment Layer

We further analyze why the intermediate layer is a suitable location for applying representation alignment. Figure 14 visualizes the cross-attention between text tokens and video tokens inside Cosmos across different layers. For the instruction “put the banana in the box,” we visualize the attention maps corresponding to the tokens “banana” and “box”. The results show that at the 8th layer, the attention maps are most spatially consistent with the corresponding objects in the generated video: the token “banana” attends to the banana region, and the token “box” attends to the box region. This phenomenon suggests that the 8th layer is where Cosmos exposes relatively clear instruction-conditioned semantic grounding. Applying AGRA at this stage is therefore reasonable: the model has already formed object-level semantic correspondence between language and visual regions, while deeper layers can still remain available for modeling motion, geometry, and fine-grained spatiotemporal dynamics. This supports our previous findings in Section 4.2.2 (a).

We also visualize the cross-attention maps in the action head, as shown in Figure 15. This visualization reveals how the action decoder reads information from the predicted video representation when

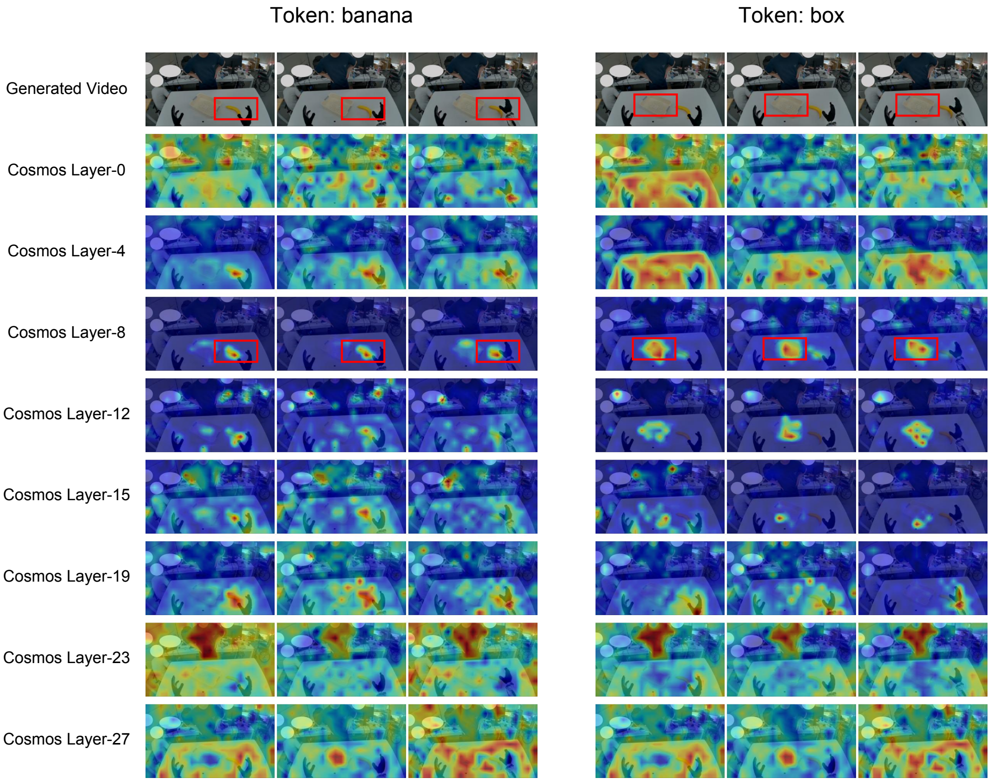


Figure 14: Text-to-video cross-attention visualization across Cosmos layers. We visualize the cross-attention maps for the instruction tokens "banana" and "box" in the generated video. The 8th layer produces the clearest correspondence between each text token and its associated visual region.

generating actions. At early action layers, the attention is relatively global and covers broad scene regions. This indicates that the action head first aggregates coarse contextual information, such as the scene layout, object distribution, and robot configuration. As the action prediction proceeds to middle and later layers, the attention becomes increasingly concentrated around the robot hands, especially the regions relevant to hand-object interaction. This progression suggests a hierarchical decoding process: the action head first establishes global task context and then focuses on local interaction regions needed for precise control.

D.3 Visualization for Different Semantic Feature Types

To complement the analysis in Section 4.2.2 (b), we visualize the feature spaces of different semantic visual encoders in Figure 16. Specifically, we compare DINOv2 and SigLIP features using PCA-based visualization. DINOv2 produces a more spatially structured and object-centric representation: object boundaries, geometric contours, and different functional regions are more clearly separated. In contrast, SigLIP features are more spatially diffuse.

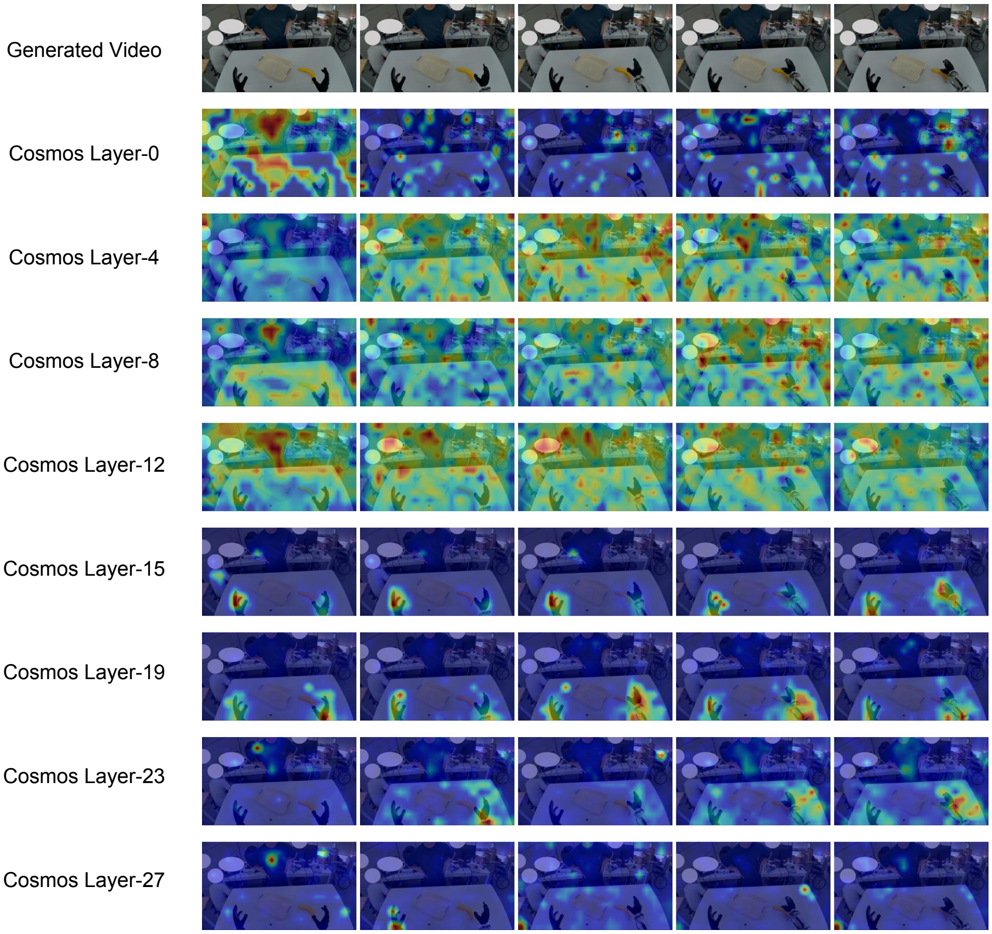
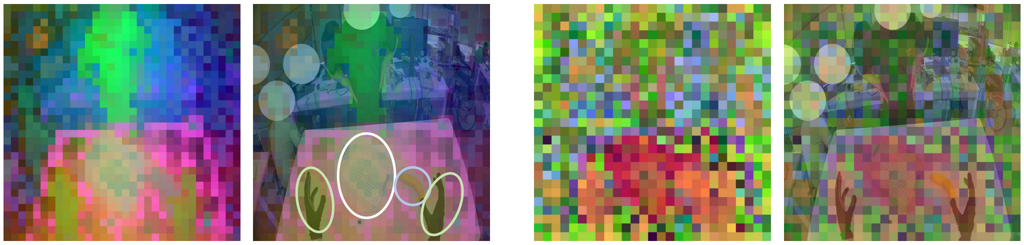


Figure 15: Action-head cross-attention visualization. We visualize where the action head attends when predicting actions from Cosmos video representations. Early layers mainly attend to global scene information, while middle and later layers gradually concentrate on the robot hands and hand-object interaction regions.



DinoV2

Siglip

Figure 16: Visualization of different semantic feature types. We compare PCA visualizations of DINOv2 and SigLIP features. DINOv2 exhibits a more object-centric and spatially coherent feature structure, with clearer object boundaries, geometry, and part-level separation.