applied sciences

MDPI

Article

Robust Vision-Language-Action Models via Object-Centric Learning and Distance-Based Chunk Alignment

Sung-Gil Park [1,2], Yong-Geon Kim [2], Seuk-Woo Ryu [2], Byeong Gil Yoo [2], Sungeun Chung [2], Jeong-Seop Park [1,2], Woo-Jin Ahn [3,*] and Myo-Taeg Lim [1,*]

[1] School of Electrical Engineering, Korea University, 145, Anamro, Seongbuk-gu, Seoul 02841, Republic of Korea; sunggil.park@lge.com (S.-G.P.); jeongseop.park@lge.com (J.-S.P.)
[2] LG Electronics, 30 Magokjungang 10-ro, Gangseo-gu, Seoul 07796, Republic of Korea; yonggeon.kim@lge.com (Y.-G.K.); seukwoo.ryu@lge.com (S.-W.R.); byeonggil.yoo@lge.com (B.G.Y.); sungeun.chung@lge.com (S.C.)
[3] Department of Artificial Intelligence, Inha University, 100, Inha-ro, Michuhol-gu, Incheon 22212, Republic of Korea
* Correspondence: wjahn@inha.ac.kr (W.-J.A.); mlim@korea.ac.kr (M.-T.L.)

Abstract

Vision–language–action (VLA) models have shown strong potential for enabling robots to interpret goals and perform complex manipulation tasks by integrating perception, language, and control. However, existing VLAs rely heavily on large-scale, diverse demonstration datasets, which are difficult and expensive to collect. When trained with limited data, they often overfit to irrelevant visual cues such as background, lighting, or viewpoint, resulting in weak generalization. To overcome this limitation, we propose a simple yet effective object-centric learning framework for VLA. For each sub-task, the framework leverages an instance segmentation foundation model to identify and track task-relevant objects, and trains the policy on both the original RGB scene and two object-focused representations: (i) a masked image emphasizing the target object and (ii) an object-only crop. These multiple visual inputs share the same action supervision, encouraging the policy to attend to the manipulated object rather than the surrounding context. Furthermore, a distance-based chunk alignment mechanism ensures smooth control transitions between consecutive predicted action segments. Experiments conducted in both simulation and real hardware demonstrate that the proposed method achieves robust performance and stable trajectories across various manipulation tasks, validating its practicality and efficiency in training object-aware robotic behaviors.

Keywords: vision–language–action model; object-centric learning; robotics; deep learning

Check for updates

Academic Editors: Rakib Abdur and Mehmet Aydin

Received: 28 February 2026
Revised: 19 March 2026
Accepted: 27 March 2026
Published: 31 March 2026

Copyright: © 2026 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license.

1. Introduction

In recent years, the integration of vision, language, and action has emerged as a promising direction for developing general-purpose robots capable of human-like reasoning and decision-making [1,2]. Vision–language–action (VLA) models [3,4] aim to unify visual perception, linguistic understanding, and motion control within a single framework, enabling robots to perceive their surroundings, interpret natural-language commands, and generate executable action trajectories. VLA have bridged the gap between high-level semantic goals and low-level motor control, allowing robots to perform increasingly complex tasks such as cooking, assembly, or tool manipulation [5].

Despite this advancement, achieving generalization remains extremely challenging. State-of-the-art VLA approaches rely heavily on large-scale datasets [6,7], which require teleoperation [8] or kinesthetic teaching to collect diverse demonstrations. This process is time-consuming, expensive, and fundamentally limited in its coverage of object diversity, texture variability, lighting conditions, and physical interactions. As a result, even hundreds of hours of collected demonstrations do not guarantee robustness in unseen settings or environments distinct from the training distribution.

Simulation-based data augmentation has emerged as a practical alternative to collecting large-scale real-world demonstrations [9,10]. Simulated environments enable automated data generation at scale with precise control over geometry, illumination, and object configurations, allowing robots to execute thousands of trials without physical wear or operational risk. To reduce distribution mismatch, techniques such as domain randomization [11–13] perturb textures, lighting, and material properties, while generative models such as CycleGAN, diffusion-based policies [14], and flow matching [15] attempt to translate or align synthetic observations toward more realistic visual distributions. These approaches improve appearance realism and invariance to low-level visual noise, yet their improvements remain largely appearance-driven. Critically, they do not guarantee that the learned policy grounds its decisions on the correct causal objects or affordances relevant to the manipulation task. Consequently, when deployed beyond the visual statistics of the training simulator, performance still degrades, indicating that current augmentation and domain alignment strategies alone do not sufficiently address the core challenge.

We believe that a core limitation stems from the absence of structured guidance on what portions of the visual input should matter during learning. Although VLA models consume the entire RGB scene, the robot only needs to reason about a small subset of objects that are causally involved in the manipulation. When supervision is not explicitly object-centered, substantial model capacity is wasted on background patterns, textures, and lighting variations that are irrelevant to the control objective. This not only increases sensitivity to dataset bias but also dilutes the learned representation, making it harder for the model to build stable object-affordance associations.

To this end, we introduce an object-centric learning strategy that restricts visual attention to task-relevant entities. The core concept is to constrain the supervision signal toward the object that actually drives the manipulation outcome. Concretely, demonstration videos are decomposed into sub-tasks and the corresponding target objects are isolated. For each sub-task, the policy is trained with three complementary visual views: the original full-scene observation, a masked scene in which only the target object is preserved, and an object-only crop. During training, this triplet is generated after object mask extraction, and all three images are passed through the same visual encoder to ensure feature comparability. All three share the same action labels, implicitly enforcing a consistent object-to-action causal mapping. Furthermore, object appearances are diversified by augmenting the object-only crops with real images obtained through web-scale crawling, which introduces natural variability in coloration, texture, material reflectance, and lighting. This simple yet effective formulation makes the policy repeatedly experience that action supervision originates from the manipulated object rather than from background appearance, helping improve robustness under variations in context, viewpoint, and scene configuration.

In addition, we address the fact that object-centric grounding alone does not guarantee stable execution at inference time. Even if the policy correctly focuses on the manipulated object, VLA models typically generate actions in chunked sequences, which can introduce discontinuities when transitioning between predicted segments. To mitigate this issue, we apply a distance-based alignment step at chunk boundaries. We measure the Cartesian distance between the terminal end-effector pose of chunk t and the initial pose of chunk

$t+1$ and shift the next chunk accordingly, ensuring that predicted actions remain consistent with the same object-centric control reference. This post-alignment step complements our representation-level grounding by reducing drift during rollout, resulting in smoother behavior without modifying the underlying policy architecture, with comprehensive validation in both simulation and real hardware demonstrating stronger generalization and smoother execution.

Our contributions are summarized as follows:

- We introduce a simple object-centric learning strategy for VLA, where each sub-task is trained with a triplet of views (full-scene, masked-scene, and object-only) that share the same action labels, strengthening causal grounding between the manipulated object and the executed action.
- We incorporate web-scale object appearance augmentation into the object-only branch, allowing diverse real-world textures and colors to be injected without additional real robot data collection.
- We apply a distance-based alignment mechanism at action chunk boundaries, which reduces discontinuity during rollout and yields smoother long-horizon behavior without modifying the policy network itself.
- Extensive experiments across both simulation and real hardware validate that our framework improves generalization under visual variation and yields smoother and more consistent trajectories during long-horizon execution.

2. Related Work

2.1. Vision–Language–Action

Advances in vision–language–action (VLA) models have significantly accelerated progress in general-purpose robot learning by unifying visual perception, language understanding, and motion control within a single framework. VLA models enable robots to interpret natural-language instructions, reason about object affordances, and generate goal-directed actions for diverse manipulation tasks. Early systems such as RT-1 [6] and RT-2 [5] pioneered large-scale transformer-based visuomotor policies that learn from paired video and instruction datasets collected through teleoperation. These models demonstrated that large, diverse robot demonstrations can be distilled into end-to-end policies capable of grounding textual goals in perceptual observations. However, their generalization remains limited to domains covered by training data. When deployed in unseen environments or with novel objects, these models often exhibit degraded performance due to overreliance on background cues, lighting, and camera viewpoints.

Subsequent frameworks extended this paradigm by integrating multimodal encoders and action diffusion models to enhance temporal consistency and multimodal reasoning. For instance, ALOHA [3] demonstrated fine-grained bimanual manipulation using low-cost teleoperation, while Pi0 [4] introduced a unified vision–language–action flow that integrates perception, reasoning, and control within a single model. Despite these innovations, VLA models still rely heavily on large-scale real-world demonstrations, which are expensive and difficult to collect at scale.

Existing VLA models demonstrate strong capability in unifying perception, language, and control, but they remain limited by insufficient object-level grounding. Most architectures operate on full-scene observations, causing the policy to allocate capacity to background cues that are not causally linked to task completion. This suggests a need for object-centric representations in which learning is concentrated on the entities that directly determine manipulation outcomes, a perspective that forms the basis of this work.

2.2. Object-Centric Learning

Object-centric learning (OCL) seeks representations that factor a scene into entities with disentangled, slot-like features, which has been shown to improve compositional generalization and data efficiency in vision benchmarks. Conventional approaches [16–18] identify objects and learn per-object latents without supervision. These methods establish that modeling scenes as sets of entities supports out-of-distribution recomposition and relational reasoning.

In robotics, several influential methods adopt object- or region-centric inductive biases for manipulation. Transporter Networks [19] attend to local feature patches and infer spatial displacements for pick-and-place, achieving strong sample efficiency on tabletop tasks but focusing on planar rearrangement with task-specific training. PerAct [20] conditions on language and reasons over 3D voxel grids, enabling multi-task manipulation but at significant computational cost and with dense scene assumptions that complicate deployment. Set-of-Mark Prompting (SOM) [21] introduced the concept of representing visual entities through explicit visual markers. SOM uses a set of symbolic marks to represent objects, spatial relationships, and attention regions within an image. This approach enables precise visual grounding by linking visual tokens directly to their physical counterparts, significantly improving the reasoning capability of multimodal models. While effective, these approaches are not designed to provide generic object-level grounding inside VLA policies and do not explicitly handle long-horizon action continuity.

In addition to general object-centric learning approaches, prior studies in industrial robotics have demonstrated the effectiveness of vision-based perception and segmentation for task execution. For instance, He et al. [22] proposed a Bayesian network-based framework for autonomous decision-making in welding tasks, highlighting the importance of linking visual features to robot action planning. Similarly, He et al. [23] introduced a semantic segmentation-based framework for extracting weld seam contours, enabling precise identification of task-relevant regions in industrial environments.

Unlike prior object-centric or region-centric approaches that focus solely on perception, our method directly injects object-level supervision into VLA training and enforces that action labels be grounded on the manipulated entity.

2.3. Action Chunking

Action chunking has recently emerged as a practical mechanism to improve long-horizon manipulation by predicting short action sequences instead of single-step control. ACT [3] demonstrated that chunk-level prediction significantly reduces compounding error and improves sample efficiency in imitation learning. Diffusion-based policy models [14,15] further showed that multi-step trajectory generation enables expressive and temporally coherent motion. Large VLA models such as Pi0 [4] adopt chunked action decoding as well, generating continuous motor commands over short horizons rather than frame-by-frame control. However, these approaches still execute chunks in an essentially open-loop manner: When moving from one predicted segment to the next, pose drift can accumulate, and discontinuities often appear at chunk boundaries.

In contrast, our method focuses specifically on the boundary consistency problem: We keep the generative structure of chunking, but introduce a lightweight distance-based alignment step so that each subsequent chunk is numerically matched to the preceding end-effector pose. This complements chunk-based VLA policies rather than replacing them and provides explicit geometric continuity that prior chunking-based pipelines do not model.

3. Methodology

We aim to train a VLA policy $\pi(a_{1:T}, |o, l, s)$ that maps an observation (i.e., image) o, instruction l, and robot state s to a sequence of control actions $a_{1:T}$ with stronger generalization and more stable execution. To this end, we adopt a simple two–part formulation. During training, an object-centric supervision scheme constrains learning to task-relevant entities, encouraging the policy to associate actions with the manipulated object rather than background appearance. During execution, a lightweight distance-based alignment step enforces geometric continuity at chunk boundaries, mitigating pose drift between consecutive action segments. These two components operate in complementary stages of the pipeline and result in improved robustness under visual variation and smoother long-horizon trajectories without modifying the policy architecture, as shown in Figure 1.

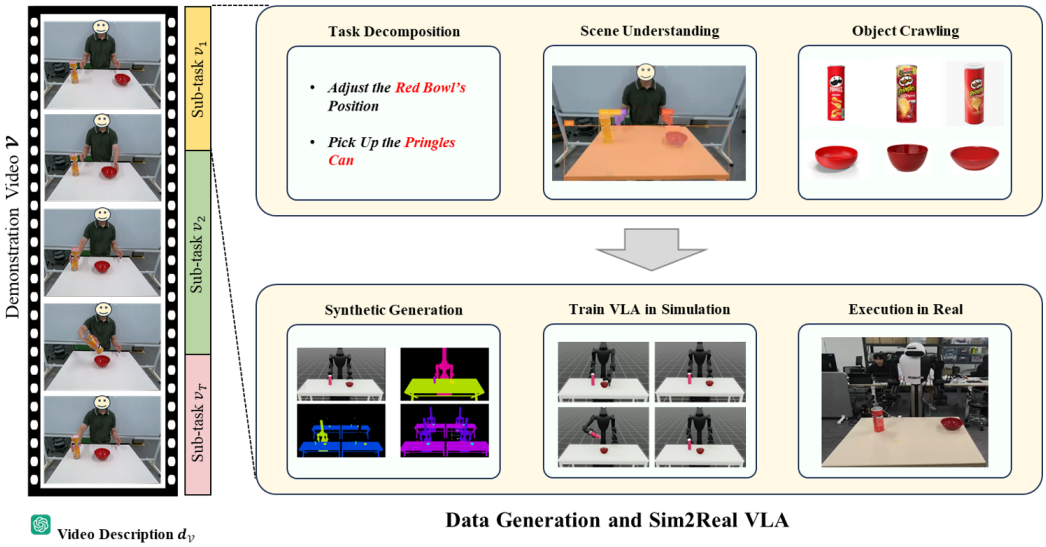


Figure 1. Proposed pipeline of object-centric learning VLA.

3.1. Object-Centric Learning VLA

Generalization in VLA policies is often limited because supervision is applied on full scenes, forcing the model to attend to background variations that are not causally tied to the manipulation. Our key idea is to explicitly constrain visual learning to the object that determines the action.

Given at time step t an observation o_t, a language command l, and robot state s_t, we derive three aligned visual inputs: the original image o_t^{full}, a masked scene retaining only the target object o_t^{mask}, and an object-only crop o_t^{crop} (Figure 2). All three views share the same action label, a_t, and are encoded by the same visual backbone. The policy π therefore receives supervision such that

$$\pi(a_t \mid o_t^{\text{full}}, l, s_t) \equiv \pi(a_t \mid o_t^{\text{mask}}, l, s_t) \equiv \pi(a_t \mid o_t^{\text{crop}}, l, s_t), \tag{1}$$

which enforces that action grounding is driven by the manipulated object rather than incidental scene appearance. The proposed three-view shared supervision enforces consistency across full-scene, masked, and object-only inputs, encouraging the policy to focus on task-relevant object features while suppressing background variations. This consistency strengthens the object–action association and improves generalization under diverse visual conditions.

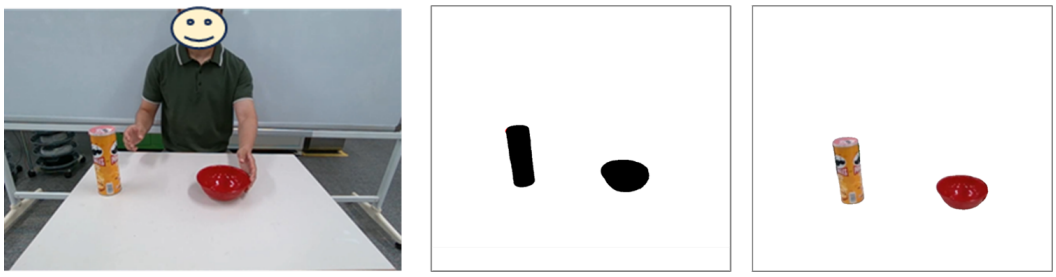


Figure 2. Input images used for training VLA: the original image o_t^{full}, a masked scene retaining only the target object o_t^{mask}, and an object-only cropped image o_t^{crop}.

3.2. Object-Centric Data Construction

A remaining question is how to obtain o_t^{mask} and o_t^{crop} from the original full image o_t^{full}. We construct these views through a two-stage procedure: (i) We decompose each demonstration into object-anchored sub-tasks using Video–Language Models (VLMs) [24], and (ii) we extract the target instance per sub-task using segmentation foundation models [25,26].

Given a demonstration video $V = \{o_1^{\text{full}}, \ldots, o_T^{\text{full}}\}$, we first obtain semantic segments $\{v_1, \ldots, v_K\} = f_{\text{VideoLLM}}(V)$ with the VideoLLM model [27]. For each segment v_k, we obtain a textual description $l_k = f_{\text{VLM}}(v_k)$, which is used as the language input during policy training. We refine this description by extracting the object–action relation using a large language model, $l_k^{\text{obj}} = f_{\text{LLM}}(l_k)$, yielding an object-aware instruction tied to the manipulated entity (Table 1). In parallel, an instance segmentation foundation model [25,28] is applied to each frame to obtain an object mask $m_t = f_{\text{seg}}(o_t^{\text{full}})$ and corresponding object image $x_t^{\text{obj}} = \text{extract}(o_t^{\text{full}}, m_t)$.

Table 1. Example of the proposed pipeline applied to a demonstration in which a person manipulates a Pringles can and a red plastic bowl. The demonstration video is segmented, converted into object-aware language instructions, and paired with object masks and cropped images for object-level representation learning.

Sub-Task	Visual Event	Target Object (s)	Local Goal
S_1	Both objects visible and stationary	Pringles can, red bowl	Recognize and localize objects
S_2	Bowl moved toward center	red bowl	Move bowl to central workspace
S_3	Hand extends toward can	Pringles can	Establish grasp contact
S_4	Can lifted and tilted above bowl	Pringles can, red bowl	Align and simulate pouring
S_5	Can lowered and released	Pringles can	Return object to rest position

These object-level pixel features are projected into a shared vision–language embedding space via a pretrained VLM [26], $e_t^{\text{obj}} = f_{\text{VLM}}(x_t^{\text{obj}}, l_k^{\text{obj}})$, and the embedding e_t^{obj} is attached to all three views $(o_t^{\text{full}}, o_t^{\text{mask}}, o_t^{\text{crop}})$ during supervision to maintain consistent object grounding across full, masked, and cropped representations (Figure 3).

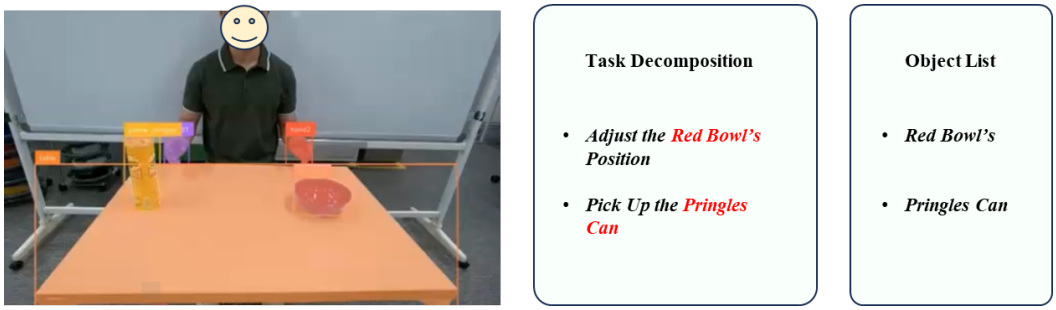


Figure 3. Resulting target object predictions from the scene understanding pipeline. The detected and segmented objects are used for object-aware instruction generation and supervision.

The robot demonstration dataset is internally collected using teleoperation and consists of approximately 150 demonstrations. For augmentation, web-crawled images are retrieved using textual and visual queries and filtered based on similarity in the vision–language embedding space.

3.3. Object-Level Appearance Augmentation

To increase appearance diversity at the object level, we construct additional visual samples by combining web-crawled exemplars with synthetic composition in simulation. Using the object crops $\{x_t^{\text{obj}}\}$ obtained in Section 3.2 as references, we retrieve visually similar images and inject them into the data construction pipeline without requiring additional real robot demonstrations.

Each object crop x_t^{obj} is encoded into a vision-language embedding [26] and used as a retrieval query. We issue both a textual query q^{text} (derived from l_k^{obj}) and a visual query q^{img} (from x_t^{obj}) to web-scale repositories. Retrieved candidates are ranked by cosine similarity in VLM embedding space, and top-M images form an expanded object set $\mathcal{X}^{\text{web}} = \{x_1^{\text{web}}, \ldots, x_M^{\text{web}}\}$ for each object instance.

For each x_i^{web}, we generate synthetic scenes by compositing the object onto randomized backgrounds sampled from simulation. Background type, illumination, material, viewpoint, and geometric transforms are sampled from predefined ranges. Matting and color matching operations are applied to blend the object boundaries, producing synthesized frames $\tilde{o}_t$ that maintain object identity while varying appearance.

The resulting synthetic set $\{\tilde{o}_t\}$ is treated as additional visual observations and paired with the same instruction l_k^{obj} and action labels a_t, thereby increasing object appearance diversity without new robot data collection.

3.4. Trajectory Alignment for Continuous Inference Control

At inference, the policy outputs short-horizon action chunks of fixed length (e.g., 50 steps). When executing multiple chunks sequentially, direct concatenation can introduce pose discontinuities at the boundary between segments. We therefore apply a lightweight alignment step that selects the closest continuation point in the next chunk.

Let p_t denote the current end-effector pose at the last step of the current chunk, and let the next predicted chunk be $C_{\text{next}} = \{p_{t+1}^{(1)}, \ldots, p_{t+1}^{(50)}\}$. We compute the Euclidean distance between p_t and each predicted pose, $d_i = \|p_t - p_{t+1}^{(i)}\|$, and select the minimal index $i^* = \arg\min_i d_i$. Rather than executing C_{next} from its first step, we resume execution from a shifted offset beyond the closest match to avoid temporal overlap:

$$C_{\text{exec}} = \{p_{t+1}^{(i^*+2)}, \ldots, p_{t+1}^{(50)}\}.$$

This alignment step enforces geometric continuity across chunk boundaries and reduces accumulated drift without modifying the policy architecture or re-optimizing trajectories.

4. Experiments

4.1. Experimental Setup

All experiments were conducted in Isaac Lab 5.0 (NVIDIA, Santa Clara, CA, USA) for simulation and on a workstation with a single NVIDIA GeForce RTX 4090 (NVIDIA, Santa Clara, CA, USA) (24 GB) GPU for training and inference. The VLA policy [4] was trained using the hybrid dataset constructed in Section 3.3 and fine-tuned with the object-centric integration described in Section 3.4. For real-world validation, the policy was deployed on the ROBOTIS AI-Worker platform (ROBOTIS, Seoul, Republic of Korea) (Figure 4), a mobile manipulator with dual 6-DoF arms, a 3-DoF torso, and a pan-tilt RGB-D head camera. Control is executed through ROS 2 (Open Robotics, Mountain View, CA, USA), and trajectories predicted in simulation are replayed on the robot via a ROS–Isaac bridge (NVIDIA, Santa Clara, CA, USA) to ensure consistent evaluation. An external top-view RGB camera was used for scene observation during real-robot testing. The robot demonstration dataset is internally collected using teleoperation in a real-world setup and consists of approximately 100 demonstrations across multiple manipulation tasks. For object-level augmentation, web-crawled images are retrieved using both textual and visual queries derived from the target object, and filtered based on similarity in the vision-language embedding space to ensure relevance. The segmentation is performed only during data construction, and three-view inputs are used only in training. Thus, no additional overhead is introduced at inference.



Figure 4. Sim2Real test with ROBOTIS AI-Worker.

For evaluation, we report task success rate. Each task trial is counted as success only if the robot completes the instructed manipulation goal without human intervention. We conduct N rollouts per task (typically $N = 20$) and compute success rate as the ratio of successful executions over total trials. We evaluate three tabletop manipulation tasks: grasping cylindrical objects (e.g., cans), repositioning bowls, and opening-and-closing plastic containers.

4.2. Experimental Results

We evaluate the contribution of (1) object-centric input processing, (2) object-level data augmentation, (3) offset settings in the distance-based chunk alignment mechanism, and (4) inference-time alignment, using Pi0 [4] as the baseline VLA model. Table 2 reports real-

world task success rates under different data scales and alignment settings, while Table 3 summarizes the effect of different offset values in the proposed alignment mechanism. Four key observations can be drawn.

Table 2. Real-world task success rates under varying data scales and inference-time alignment settings.

Single Task	Without Proposed Alignment		With Proposed Alignment	
	50 Data	150 Data	50 Data	150 Data
Baseline [4]	20	30	30	30
+ Segmentation	50	60	60	70
+ Object	30	40	40	50
Full (Ours)	50	60	70	85

Table 3. Effect of offset settings in the distance-based chunk alignment mechanism on real-world task success rate.

Offset Setting	Success Rate (%)
1	80
2 (Ours)	85
5	75
10	30

First, introducing object-centric input processing through segmentation substantially improves performance over the full-scene baseline. Without alignment, segmentation increases success rates from 20% to 50% under 50 demonstrations and from 30% to 60% under 150 demonstrations, indicating that suppressing irrelevant background information and emphasizing the manipulated object significantly enhances robustness. By contrast, using the object-only input alone yields more moderate gains, improving success rates from 20% to 30% under 50 demonstrations and from 30% to 40% under 150 demonstrations without alignment. This suggests that although object-only representations help the policy focus on the manipulated entity, removing broader scene context entirely may limit performance compared with the masked segmentation view.

Second, incorporating object-level augmentation further improves long-horizon stability when combined with alignment. While augmentation alone achieves performance comparable to segmentation without alignment, it leads to larger gains once alignment is enabled, reaching 70% and 85% success under 50 and 150 demonstrations, respectively. This suggests that increasing object appearance diversity strengthens the consistency of object-to-action grounding, particularly during extended rollouts.

Third, the offset setting in the proposed distance-based chunk alignment mechanism affects performance depending on how far the continuation point is shifted. As summarized in Table 3, small offsets generally maintain stable performance, with offset = 2 achieving the best success rate of 85%. Offset = 1 also provides competitive performance (80%), indicating that nearby continuation points do not cause a significant problem. However, when the offset becomes too large, performance drops substantially, decreasing to 75% at offset = 5 and 30% at offset = 10. These results suggest that selecting a continuation point that is close enough to preserve temporal consistency is important, whereas excessively large offsets may skip critical motion transitions and degrade execution stability.

Fourth, the proposed distance-based chunk alignment provides limited improvement for the baseline but yields progressively larger gains for object-centric models. For the baseline, alignment increases performance only marginally, whereas segmentation and augmentation benefit more noticeably. This indicates that geometric alignment is most effective when the underlying policy already exhibits stronger object-level grounding,

helping reduce discontinuities at chunk boundaries and mitigate drift accumulation during long-horizon execution.

We further evaluate the proposed method on a more complex long-horizon manipulation task, namely cabinet opening, as shown in Figure 5. This task requires the robot to localize the cabinet handle, establish a stable grasp, and execute a continuous pulling motion, involving multiple sequential steps. The segmentation results (top row) demonstrate that the proposed method consistently identifies the task-relevant handle, while the execution results (bottom row) show stable and accurate manipulation. In contrast, the baseline Pi0 model often fails to reliably localize and grasp the handle due to background distractions. As summarized in Table 4, the proposed method achieves a significantly higher success rate (80%) compared to the baseline (25%). These results indicate that the proposed object-centric framework generalizes effectively to more complex and long-horizon manipulation scenarios beyond simple tabletop tasks.

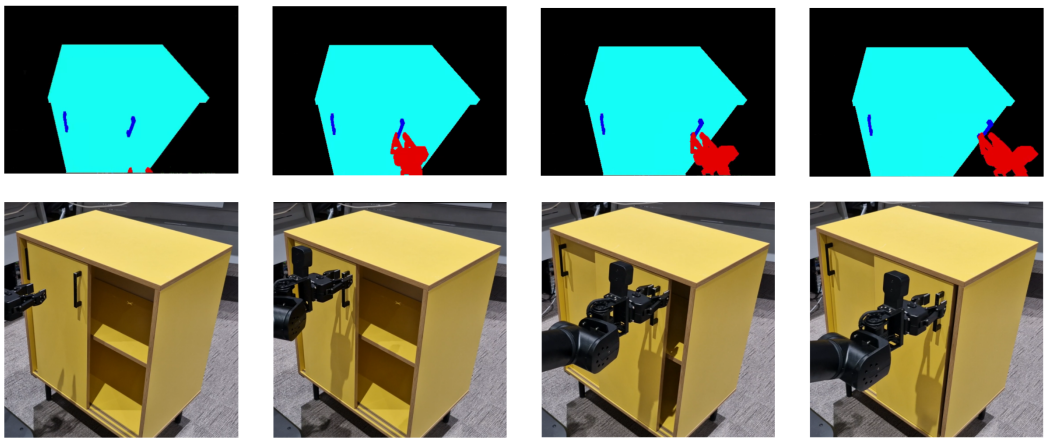


Figure 5. Cabinet opening task. The (**top row**) shows segmentation results, and the (**bottom row**) shows the corresponding robot execution.

Table 4. Task success rate on the cabinet opening task.

Method	Success Rate (%)
Baseline [4]	25
Ours	80

Overall, these results demonstrate that object-centric supervision improves manipulation robustness, and that lightweight inference-time alignment further enhances stability when combined with stronger object-aware training.

5. Conclusions

This work presented a simple object-centric vision–language–action training formulation and a lightweight inference alignment mechanism. On the training side, we introduced triplet supervision using full-scene, masked-scene, and object-only views to explicitly associate actions with the manipulated entity rather than background appearance. On the inference side, we applied a distance-based chunk alignment strategy that enforces geometric continuity across consecutive action segments without modifying the policy architecture. Experimental results in simulation and on a real mobile manipulator showed that both components contributed to higher task success and more stable execution compared to baseline under identical data budgets. These findings suggest that enforcing object-level grounding during supervision and stabilizing temporal transitions during execution are practical and effective inductive biases for VLA-based robotic manipulation.

While the proposed method demonstrates strong performance, several limitations remain. The framework depends on the accuracy of instance segmentation, which may degrade in cluttered or occluded environments. In addition, although we validate the approach on a more complex cabinet opening task, further evaluation in diverse and industrial-scale scenarios such as multi-object interaction, assembly, or welding is needed. The current alignment mechanism is based on a simple heuristic, and more advanced trajectory optimization could further improve performance. These directions will be explored in future work to further enhance the robustness and applicability of the proposed framework.

Author Contributions: Conceptualization, S.-G.P.; methodology, Y.-G.K., S.-W.R., B.G.Y., S.C. and J.-S.P.; software, Y.-G.K., S.-W.R., B.G.Y., S.C. and J.-S.P.; validation, Y.-G.K., S.-W.R., B.G.Y. and S.C.; writing—original draft preparation, S.-G.P.; writing—review and editing, W.-J.A. and M.-T.L.; supervision, W.-J.A. and M.-T.L. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The original robot demonstration data used in this study were internally collected for research purposes and are not publicly available. Additional object appearance images were obtained from publicly accessible web sources (e.g., Google image search) for augmentation purposes. No new publicly archived dataset was generated in this study. Further details regarding the data may be available from the corresponding authors upon reasonable request.

Acknowledgments: This research was carried out with support and assistance from LG Electronics.

Conflicts of Interest: Authors Sung-Gil Park, Yong-Geon Kim, Seuk-Woo Ryu, Byeong Gil Yoo, Sungeun Chung and Jeong-Seop Park were employed by the company LG Electronics. The remaining authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

References

1. Ahn, M.; Brohan, A.; Brown, N.; Chebotar, Y.; Cortes, O.; David, B.; Finn, C.; Fu, C.; Gopalakrishnan, K.; Hausman, K.; et al. Do as i can, not as i say: Grounding language in robotic affordances. *arXiv* **2022**, arXiv:2204.01691. [CrossRef]
2. Wei, J.; Wang, X.; Schuurmans, D.; Bosma, M.; Xia, F.; Chi, E.; Le, Q.V.; Zhou, D. Chain-of-thought prompting elicits reasoning in large language models. *Adv. Neural Inf. Process. Syst.* **2022**, *35*, 24824–24837.
3. Zhao, T.Z.; Kumar, V.; Levine, S.; Finn, C. Learning fine-grained bimanual manipulation with low-cost hardware. *arXiv* **2023**, arXiv:2304.13705.
4. Black, K.; Brown, N.; Driess, D.; Esmail, A.; Equi, M.; Finn, C.; Fusai, N.; Groom, L.; Hausman, K.; Ichter, B.; et al. π0: A vision-language-action flow model for general robot control. *arXiv* **2024**, arXiv:2410.24164.
5. Zitkovich, B.; Yu, T.; Xu, S.; Xu, P.; Xiao, T.; Xia, F.; Wu, J.; Wohlhart, P.; Welker, S.; Wahid, A.; et al. Rt-2: Vision-language-action models transfer web knowledge to robotic control. In Proceedings of the Conference on Robot Learning, Atlanta, GA, USA, 6–9 November 2023; pp. 2165–2183.
6. Brohan, A.; Brown, N.; Carbajal, J.; Chebotar, Y.; Dabis, J.; Finn, C.; Gopalakrishnan, K.; Hausman, K.; Herzog, A.; Hsu, J.; et al. Rt-1: Robotics transformer for real-world control at scale. *arXiv* **2022**, arXiv:2212.06817.
7. Fu, Z.; Zhao, T.Z.; Finn, C. Mobile aloha: Learning bimanual mobile manipulation with low-cost whole-body teleoperation. *arXiv* **2024**, arXiv:2401.02117.
8. He, T.; Luo, Z.; Xiao, W.; Zhang, C.; Kitani, K.; Liu, C.; Shi, G. Learning human-to-humanoid real-time whole-body teleoperation. In Proceedings of the 2024 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Abu Dhabi, United Arab Emirates, 14–18 October 2024; pp. 8944–8951.
9. Savva, M.; Kadian, A.; Maksymets, O.; Zhao, Y.; Wijmans, E.; Jain, B.; Straub, J.; Liu, J.; Koltun, V.; Malik, J.; et al. Habitat: A platform for embodied ai research. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Seoul, Republic of Korea, 27 October–2 November 2019; pp. 9339–9347.

10. Mower, C.; Stouraitis, T.; Moura, J.; Rauch, C.; Yan, L.; Behabadi, N.Z.; Gienger, M.; Vercauteren, T.; Bergeles, C.; Vijayakumar, S. ROS-PyBullet Interface: A framework for reliable contact simulation and human-robot interaction. In Proceedings of the Conference on Robot Learning, Atlanta, GA, USA, 6–9 November 2023; pp. 1411–1423.
11. Zhu, J.Y.; Park, T.; Isola, P.; Efros, A.A. Unpaired image-to-image translation using cycle-consistent adversarial networks. In Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, 22–29 October 2017; pp. 2223–2232.
12. Huber, J.; Hélénon, F.; Watrelot, H.; Amar, F.B.; Doncieux, S. Domain randomization for sim2real transfer of automatically generated grasping datasets. In Proceedings of the 2024 IEEE International Conference on Robotics and Automation (ICRA), Yokohama, Japan, 13–17 May 2024; pp. 4112–4118.
13. Ahn, W.J.; Yang, G.Y.; Choi, H.D.; Lim, M.T. Style blind domain generalized semantic segmentation via covariance alignment and semantic consistence contrastive learning. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 16–22 June 2024; pp. 3616–3626.
14. Chi, C.; Xu, Z.; Feng, S.; Cousineau, E.; Du, Y.; Burchfiel, B.; Tedrake, R.; Song, S. Diffusion policy: Visuomotor policy learning via action diffusion. *Int. J. Robot. Res.* **2023**, *44*, 1684–1704. [CrossRef]
15. Lipman, Y.; Chen, R.T.; Ben-Hamu, H.; Nickel, M.; Le, M. Flow matching for generative modeling. *arXiv* **2022**, arXiv:2210.02747.
16. Burgess, C.P.; Matthey, L.; Watters, N.; Kabra, R.; Higgins, I.; Botvinick, M.; Lerchner, A. Monet: Unsupervised scene decomposition and representation. *arXiv* **2019**, arXiv:1901.11390. [CrossRef]
17. Greff, K.; Kaufman, R.L.; Kabra, R.; Watters, N.; Burgess, C.; Zoran, D.; Matthey, L.; Botvinick, M.; Lerchner, A. Multi-object representation learning with iterative variational inference. In Proceedings of the International Conference on Machine Learning, Long Beach, CA, USA, 9–15 June 2019; pp. 2424–2433.
18. Locatello, F.; Weissenborn, D.; Unterthiner, T.; Mahendran, A.; Heigold, G.; Uszkoreit, J.; Dosovitskiy, A.; Kipf, T. Object-centric learning with slot attention. *Adv. Neural Inf. Process. Syst.* **2020**, *33*, 11525–11538.
19. Zeng, A.; Florence, P.; Tompson, J.; Welker, S.; Chien, J.; Attarian, M.; Armstrong, T.; Krasin, I.; Duong, D.; Sindhwani, V.; et al. Transporter networks: Rearranging the visual world for robotic manipulation. In Proceedings of the Conference on Robot Learning, London, UK, 8–11 November 2021; pp. 726–747.
20. Shridhar, M.; Manuelli, L.; Fox, D. Perceiver-actor: A multi-task transformer for robotic manipulation. In Proceedings of the Conference on Robot Learning, Atlanta, GA, USA, 6–9 November 2023; pp. 785–799.
21. Yang, J.; Zhang, H.; Li, F.; Zou, X.; Li, C.; Gao, J. Set-of-mark prompting unleashes extraordinary visual grounding in gpt-4v. *arXiv* **2023**, arXiv:2310.11441.
22. He, Y.; Ma, G.; Chen, S. Autonomous decision-making of welding position during multipass GMAW with T-joints: A Bayesian network approach. *IEEE Trans. Ind. Electron.* **2021**, *69*, 3909–3917. [CrossRef]
23. He, Y.; Cai, R.; Dai, F.; Yu, Z.; Deng, Y.; Deng, J.; Wang, Z.; Ma, G.; Zhong, W. A unified framework based on semantic segmentation for extraction of weld seam profiles with typical joints. *J. Manuf. Processes* **2024**, *131*, 2275–2287. [CrossRef]
24. Achiam, J.; Adler, S.; Agarwal, S.; Ahmad, L.; Akkaya, I.; Aleman, F.L.; Almeida, D.; Altenschmidt, J.; Altman, S.; Anadkat, S.; et al. Gpt-4 technical report. *arXiv* **2023**, arXiv:2303.08774. [CrossRef]
25. Liu, S.; Zeng, Z.; Ren, T.; Li, F.; Zhang, H.; Yang, J.; Jiang, Q.; Li, C.; Yang, J.; Su, H.; et al. Grounding dino: Marrying dino with grounded pre-training for open-set object detection. In *Proceedings of the European Conference on Computer Vision*; Springer: Cham, Switzerland, 2024; pp. 38–55.
26. Radford, A.; Kim, J.W.; Hallacy, C.; Ramesh, A.; Goh, G.; Agarwal, S.; Sastry, G.; Askell, A.; Mishkin, P.; Clark, J.; et al. Learning transferable visual models from natural language supervision. In Proceedings of the International Conference on Machine Learning, Virtual, 18–24 July 2021; pp. 8748–8763.
27. Chen, G.; Zheng, Y.D.; Wang, J.; Xu, J.; Huang, Y.; Pan, J.; Wang, Y.; Wang, Y.; Qiao, Y.; Lu, T.; et al. Videollm: Modeling video sequence with large language models. *arXiv* **2023**, arXiv:2305.13292. [CrossRef]
28. Ravi, N.; Gabeur, V.; Hu, Y.T.; Hu, R.; Ryali, C.; Ma, T.; Khedr, H.; Rädle, R.; Rolland, C.; Gustafson, L.; et al. Sam 2: Segment anything in images and videos. *arXiv* **2024**, arXiv:2408.00714. [PubMed]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.