BEYOND VIEW CONSISTENCY: ROTATION-AWARE CONSISTENT ALIGNMENT FOR VIEWPOINT-ROBUST ROBOTIC MANIPULATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Visual robotic manipulation policies remain highly sensitive to viewpoint variations, which can significantly degrade performance. Existing approaches commonly address this issue by learning view-consistent representations. However, directly enforcing cross-view consistency may overlook viewpoint-dependent task-relevant information for precise manipulation. To address this limitation, we propose Rotation-aware Consistent Alignment (RCA). RCA prioritizes capturing task-relevant features. RCA then accounts for camera rotation to adapt viewpoint-dependent features and align them in a unified latent space. In practice, our framework comprises two stages. The first stage learns task-relevant representations from each observation. The second stage adapts and aligns these representations across viewpoints using a FiLM mapping network conditioned on predicted viewpoint parameters. We evaluate RCA in Meta-World and PandaGym simulations and real-world robot experiments under diverse viewpoint perturbations. RCA improves average success rates over baselines by 19.8, 26.7, 4.2, and 6.9 percentage points in Meta-World azimuth, Meta-World camera shaking, PandaGym azimuth, and real-world azimuth evaluations, respectively.

1 INTRODUCTION

Vision-based robotic manipulation has achieved remarkable progress in recent years (Kalashnikov et al., 2018; Quillen et al., 2018; Agarwal et al., 2020; Chen et al., 2022; Kim et al., 2024; Liu et al., 2024b; Luo et al., 2025; Kim et al., 2026). However, these policies are still highly sensitive to viewpoint variations. Even slight changes in camera viewpoint can lead to substantial performance degradation. In fact, viewpoint variations are difficult to avoid in real-world deployments due to factors such as differences in camera placement or accidental camera movement. Therefore, addressing this sensitivity is a fundamental requirement for real-world deployment.

To this end, recent studies have explored learning view-consistent representations (Sermanet et al., 2018; Seo et al., 2023b; Liu et al., 2024a; Yuan et al., 2024; Pang et al., 2025). These approaches reduce viewpoint-specific variation by enforcing cross-view consistency or performing cross-view reconstruction. However, such view-consistent representations may overlook viewpoint-dependent task-relevant information, thereby limiting downstream policy performance.

As illustrated in Figure 1(a), visual observations differ significantly across viewpoints, and the camera-relative poses of the manipulator and target object also change substantially. Without considering that these changes arise from camera rotation, view-consistent models tend to learn a small subset of shared features across viewpoints, potentially overlooking viewpoint-dependent task-relevant information. To further illustrate this, we present attention maps of ReViWo (Pang et al., 2025), a representative view-consistent method. The model focuses mainly on the robot wrist with minimal cross-view variation, while other task-relevant regions, such as the target object and gripper, are ignored.

In fact, although visual observations differ across viewpoints, the underlying task entities remain the same. Humans remain robust to viewpoint changes by being aware of how these changes affect appearance and adapting their representations accordingly. Inspired by this insight, we propose

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

Rotation-aware Consistent Alignment (RCA). RCA prioritizes capturing task-relevant features. RCA then accounts for camera rotation to adapt viewpoint-dependent features and align them in a unified latent space. As illustrated in Figure 1(b), viewpoint-conditioned adaptation can improve cross-view feature consistency, allowing more task-relevant information to be retained during alignment. Attention maps further show that RCA attends more comprehensively to task-relevant regions, including the target object, gripper, and robot arm.

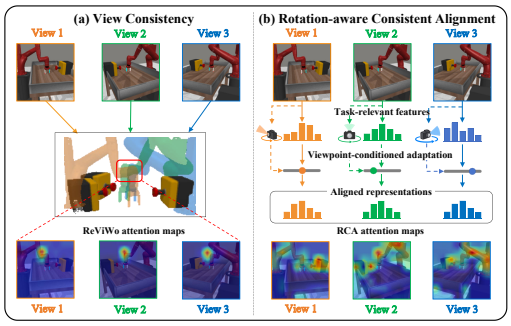


Figure 1: View consistency versus RCA.

In practice, we propose a two-stage framework. First, we learn information-sufficient task-relevant representations from each observation. Second, we use predicted viewpoint parameters to condition the feature mapping, adapting and aligning these representations across viewpoints in a unified latent space. This design improves viewpoint robustness while preserving task-relevant information.

For the first stage, we aim to preserve both motion information and static object information. The inverse-dynamics objective captures motion-related information. For static object information, entity-centric masked reconstruction captures visual information of the manipulator and target objects, while frozen DINOv3 (Siméoni et al., 2025) features provide object semantics. The resulting representations are expected to retain sufficient task-relevant information for subsequent cross-view alignment.

For the second stage, we first predict viewpoint parameters from observations. These parameters condition a FiLM-based (Perez et al., 2018) mapping network that adapts and aligns the task-relevant representations across viewpoints in a unified latent space. The mapping network is trained with three complementary losses: a cross-view contrastive loss aligns mapped features of the same state across viewpoints; a zero-offset preservation loss discourages unnecessary feature distortion at zero viewpoint offset; and a mapping reconstruction loss reduces information loss during mapping.

Our main contributions are summarized as follows:

- We propose Rotation-aware Consistent Alignment (RCA), which first captures task-relevant features and then accounts for camera rotation to adapt and align representations across viewpoints, improving viewpoint robustness while preserving task-relevant information.
- In practice, our framework comprises two stages. The first stage combines inverse-dynamics supervision with entity-centric masked reconstruction to learn task-relevant representations. The second stage uses a FiLM-based feature mapping network conditioned on predicted viewpoint parameters for cross-view alignment.
- Extensive experiments in Meta-World and PandaGym simulations and on a real-world robot demonstrate RCA's improved viewpoint robustness. These evaluations cover azimuth variations, camera shaking, and changes in camera pitch, distance, and height.

2 Related Work

Representation Learning for Visual Manipulation. Visual robotic manipulation requires extracting compact and task-relevant representations from high-dimensional image observations. Self-supervised methods such as CURL (Laskin et al., 2020) and DBC (Zhang et al., 2021) learn visual representations through contrastive learning or bisimulation metrics (Ferns et al., 2004). Masked autoencoders (MAE) (He et al., 2022) have also been widely used for visual representation learning in control tasks (Seo et al., 2023a; Radosavovic et al., 2023). Beyond task-relevant representation learning, recent works leverage general visual priors for manipulation. For instance, PIE-G (Yuan et al., 2022) shows that frozen ImageNet-pretrained encoders can provide generalizable features for robotic tasks, while R3M (Nair et al., 2022) learns frozen representations from diverse human video datasets. Large-scale imitation learning models, such as RT-1 (Brohan et al., 2023), RT-2 (Zitkovich et al., 2023), and RT-X (Vuong et al., 2024), attempt to bypass explicit state representation by leveraging massive datasets and high-capacity models. However, these methods do not explicitly address viewpoint variations, a major challenge for robust visual robotic manipulation.

View-consistent Representation Learning. To mitigate camera viewpoint variations, recent works have explored view-consistent representation learning for robotic manipulation. TCN (Sermanet et al., 2018) learns view-invariant representations from multi-view videos through time-contrastive supervision. Maniwhere (Yuan et al., 2024) leverages multi-view observations and extensive view augmentations, while MVMWM (Seo et al., 2023b) reconstructs target images from different viewpoints with a multi-view masked autoencoder (He et al., 2022). ReViWo (Pang et al., 2025) decomposes visual observations into view-consistent and view-dependent representations for robust control. Although these methods improve viewpoint robustness, direct view-consistency tends to learn a small subset of shared features across viewpoints, which may overlook task-relevant information.

3 PRELIMINARIES

Problem Formulation. We consider visual robotic manipulation under partial observability. At time t, the environment is in state s_t, and the robot executes action a_t based on visual observations. Observations are viewpoint-dependent and generated by $o_t \sim \Omega(\cdot \mid s_t, \xi_t)$, where Ω is the observation distribution and $\xi_t \in \Xi$ denotes the camera viewpoint parameter. During evaluation, ξ_t may differ from the policy-training viewpoint or vary over time, resulting in viewpoint variations in visual observations. Our goal is to learn a control policy π robust to viewpoint variations. We learn the policy through offline reinforcement learning to maximize expected return in simulation, and through imitation learning in real-world experiments.

Data Setting. We use a multi-view dataset $\mathcal{D}_{\text{mv}} = \{(\{(o_t^{(i)}, o_{t+1}^{(i)}, \xi^{(i)})\}_{i=1}^{N}, a_t)\}$ for representation learning, where $o_t^{(i)}$ and $o_{t+1}^{(i)}$ are consecutive observations from viewpoint $\xi^{(i)}$, N is the number of views, and a_t is the executed action. The multi-view dataset can contain data from multiple tasks in the downstream environment and can be supplemented with auxiliary data from other domains.

4 METHOD

We propose Rotation-aware Consistent Alignment (RCA) for viewpoint-robust visual robotic manipulation. RCA first captures task-relevant features and then accounts for camera rotation to adapt viewpoint-dependent features and align them in a unified latent space. This design improves viewpoint robustness while preserving task-relevant information. As shown in Figure 2, we implement RCA through a two-stage framework. The first stage learns task-relevant representations from each observation, aiming to retain sufficient task-relevant information for subsequent cross-view alignment (Section 4.1). The second stage adapts and aligns these representations across viewpoints using a FiLM-based mapping network conditioned on predicted viewpoint parameters (Section 4.2). The aligned representations are then used for downstream policy learning (Section 4.3).

4.1 TASK-RELEVANT REPRESENTATION LEARNING

In Stage 1, we aim to preserve both motion information and static object information. As shown in Stage 1 of Figure 2, we use inverse-dynamics supervision to capture motion-related information and entity-centric masked reconstruction to capture static visual information about the manipulator and target objects. To enrich this entity visual information with object semantics, we concatenate the learned features with frozen DINOv3 (Siméoni et al., 2025) features.

We use a VQ-VAE-style visual encoder E_ϕ to encode each observation $o_t^{(i)}$ into a latent representation $z_t^{(i)} = E_\phi(o_t^{(i)})$, where $o_t^{(i)}$ denotes the observation of state s_t captured from viewpoint $\xi^{(i)}$. The encoder is trained with the following objectives:

Inverse Dynamics Supervision. To encourage the latent representations to capture motion-related information, we train an Inverse Dynamics Model (IDM) I_η to predict the executed action a_t from consecutive latent representations $z_t^{(i)}$ and $z_{t+1}^{(i)}$ by minimizing:

$$\mathcal{L}_{\text{idm}} = \mathbb{E}_{(o_t^{(i)}, o_{t+1}^{(i)}, a_t) \sim \mathcal{D}_{\text{mv}}} \left[\left\| I_\eta(z_t^{(i)}, z_{t+1}^{(i)}) - a_t \right\|_2^2 \right]. \tag{1}$$

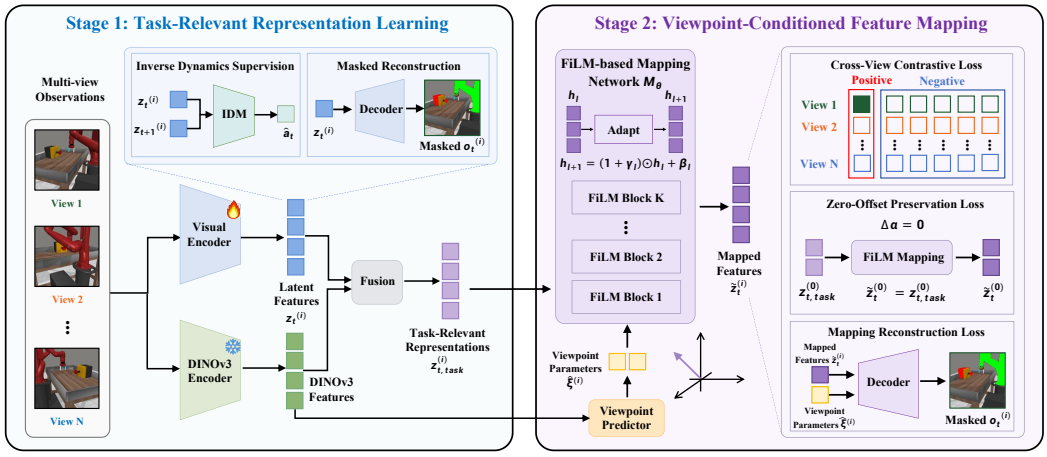


Figure 2: **RCA framework overview.** Our two-stage framework first learns information-sufficient task-relevant representations from multi-view observations, and then aligns them into a unified latent space through a viewpoint-conditioned FiLM-based mapping network.

Entity-Centric Masked Reconstruction. To capture static visual information about the task entities, we prioritize reconstructing manipulator and target-object regions over background pixels. We use Grounded SAM (Ren et al., 2024) with descriptive text prompts to generate binary masks of the robot manipulator and target objects. The union of these masks forms the entity-centric mask $M_t^{(i)} \in \{0,1\}^{H\times W}$. Let $\hat{o}_t^{(i)}$ denote the decoder reconstruction. We use weighted reconstruction to emphasize entity regions:

$$\mathcal{L}_{\text{mask}} = \left\| M_t^{(i)} \odot \left(\hat{o}_t^{(i)} - o_t^{(i)}\right)\right\|_2^2 + \lambda_{\text{bg}} \left\| \left(1 - M_t^{(i)}\right) \odot \left(\hat{o}_t^{(i)} - o_t^{(i)}\right)\right\|_2^2, \tag{2}$$

where $\odot$ denotes the Hadamard product and $0 \le \lambda_{\text{bg}} < 1$ downweights the reconstruction error on background regions.

To enrich the entity visual information with object semantics, we concatenate the learned latent features with features $\text{DINO}(o_t^{(i)})$ from a frozen DINOv3 encoder:

$$z_{t,\text{task}}^{(i)} = \left[z_t^{(i)}; \text{DINO}(o_t^{(i)})\right]. \tag{3}$$

This forms the task-relevant representation of Stage 1.

Overall Training Objective. We train the visual encoder by combining the VQ-VAE quantization objective $\mathcal{L}_{\text{vq-code}}$ (Van Den Oord et al., 2017) with the inverse-dynamics and entity-centric masked reconstruction losses:

$$\mathcal{L}_{\text{task}} = \mathcal{L}_{\text{vq-code}} + \lambda_{\text{idm}}\mathcal{L}_{\text{idm}} + \lambda_{\text{mask}}\mathcal{L}_{\text{mask}}, \tag{4}$$

where λ_{idm} and λ_{mask} weight the two supervision terms.

4.2 Viewpoint-Conditioned Feature Mapping

The task-relevant representations learned in Stage 1 still vary across viewpoints. Directly enforcing view consistency on these representations may overlook viewpoint-dependent task-relevant information. To address this, we first use a viewpoint predictor to estimate the camera rotation. We then use a FiLM-based (Perez et al., 2018) mapping network conditioned on the predicted viewpoint parameters to adapt these features and align them in a unified latent space, as shown in Stage 2 of Figure 2.

Viewpoint Parameter Prediction. To provide viewpoint conditions for feature mapping, we train a viewpoint predictor Q_ν on frozen DINOv3 features. We represent the camera azimuth offset $\Delta\alpha^{(i)}$ relative to the policy-training viewpoint using the sine–cosine encoding $\xi^{(i)} =$

$[\sin(\Delta\alpha^{(i)}), \cos(\Delta\alpha^{(i)})]^\top$. We predict this encoding as $\hat{\xi}^{(i)} = Q_\nu(\mathrm{DINO}(o_t^{(i)}))$. To predict each view's azimuth offset while preserving angular relationships across views, we combine offset regression with relative angular consistency. For a mini-batch $\mathcal{B}_{\mathrm{mv}}$ from $\mathcal{D}_{\mathrm{mv}}$, the prediction loss is:

$$\mathcal{L}_{\mathrm{vp}} = \mathbb{E}_{i\sim\mathcal{B}_{\mathrm{mv}}}\left[\left\|\hat{\xi}^{(i)} - \xi^{(i)}\right\|_2^2\right] + \lambda_{\mathrm{rel}}\mathbb{E}_{i,j\sim\mathcal{B}_{\mathrm{mv}}}\left[\left|\mathrm{wrap}\Big(\Delta\hat{\alpha}^{(i,j)} - \Delta\alpha^{(i,j)}\Big)\right|\right], \tag{5}$$

where λ_{rel} weights the relative angular consistency term, and the second expectation is taken over pairs of different views of the same state within each mini-batch. $\Delta\hat{\alpha}^{(i,j)}$ and $\Delta\alpha^{(i,j)}$ denote the predicted and ground-truth angular differences between view i and view j, and wrap maps angles to $[-\pi, \pi)$. We recover the predicted azimuth offset as $\Delta\hat{\alpha}^{(i)} = \mathrm{atan2}(\hat{\xi}_1^{(i)}, \hat{\xi}_2^{(i)})$.

FiLM-Based Mapping Network. The mapping network M_θ takes the task-relevant representation $z_{t,\mathrm{task}}^{(i)}$ and predicted viewpoint condition $\hat{\xi}^{(i)}$ as inputs. As shown in Figure 2, M_θ is implemented as a stack of FiLM-modulated blocks. For the l-th FiLM-modulated block, a conditioning MLP maps $\hat{\xi}^{(i)}$ to modulation parameters $\gamma_l(\hat{\xi}^{(i)})$ and $\beta_l(\hat{\xi}^{(i)})$. These parameters adapt the intermediate feature $h_l^{(i)}$ through feature-wise scaling and shifting:

$$\mathrm{FiLM}_l(h_l^{(i)}, \hat{\xi}^{(i)}) = \big(1 + \gamma_l(\hat{\xi}^{(i)})\big) \odot h_l^{(i)} + \beta_l(\hat{\xi}^{(i)}), \tag{6}$$

where $\odot$ denotes the Hadamard product and $h_0^{(i)} = z_{t,\mathrm{task}}^{(i)}$. The mapped representation is:

$$\tilde{z}_t^{(i)} = M_\theta(z_{t,\mathrm{task}}^{(i)}, \hat{\xi}^{(i)}). \tag{7}$$

Mapping Network Training Objective. The mapping network is trained to align representations across viewpoints while preserving task-relevant information. We train the mapping network with three complementary losses: a cross-view contrastive loss, a zero-offset preservation loss, and a mapping reconstruction loss. These losses respectively encourage cross-view alignment, discourage unnecessary feature distortion at zero viewpoint offset, and reduce task-relevant information loss during mapping.

Cross-View Contrastive Loss. For downstream control, mapped representations should be consistent across viewpoints for the same state while remaining distinguishable across different states. We therefore adopt an InfoNCE-style contrastive loss (Oord et al., 2018). For an anchor $\tilde{z}_k^{(i)}$ within a mini-batch, features of the same state from other viewpoints $\{\tilde{z}_k^{(j)}\}_{j\neq i}$ are positives, while features of different states from any viewpoint $\{\tilde{z}_l^{(v)}\}_{l\neq k}$ are negatives. The loss is:

$$\mathcal{L}_{\mathrm{cvc}} = \mathbb{E}_{k,i\neq j}\left[-\log\frac{\exp(\mathrm{sim}(\tilde{z}_k^{(i)}, \tilde{z}_k^{(j)})/\tau)}{\sum_{(l,v)\neq(k,i)}\exp(\mathrm{sim}(\tilde{z}_k^{(i)}, \tilde{z}_l^{(v)})/\tau)}\right], \tag{8}$$

where k, l denote state indices, i, j, v denote viewpoint indices, $\mathrm{sim}(\cdot,\cdot)$ is cosine similarity, and τ is a temperature hyperparameter.

Zero-Offset Preservation Loss. To avoid unnecessary feature distortion, we encourage the mapping network to behave like an identity transformation when the viewpoint offset is zero. Let $z_{t,\mathrm{task}}^{(0)}$ denote a task-relevant representation with zero viewpoint offset, and let $\tilde{z}_t^{(0)} = M_\theta(z_{t,\mathrm{task}}^{(0)}, \hat{\xi}^{(0)})$ denote its mapped feature. We define the loss as:

$$\mathcal{L}_{\mathrm{zero}} = \mathbb{E}_{o_t^{(0)}\sim\mathcal{B}_{\mathrm{mv}}}\left[\left\|\tilde{z}_t^{(0)} - \mathrm{sg}(z_{t,\mathrm{task}}^{(0)})\right\|_2^2\right], \tag{9}$$

where $\mathrm{sg}(\cdot)$ denotes the stop-gradient operator.

Mapping Reconstruction Loss. The contrastive loss encourages cross-view alignment but does not ensure that mapped features retain task-relevant information. To reduce information loss during mapping, we reconstruct observations from the mapped features $\tilde{z}_t^{(i)}$ conditioned on the predicted viewpoint parameters $\hat{\xi}^{(i)}$. We apply the same entity-weighted reconstruction loss as in Stage 1:

$$\mathcal{L}_{\mathrm{rec}} = \mathbb{E}_{(t,i)\sim\mathcal{B}_{\mathrm{mv}}}\left[\mathcal{L}_{\mathrm{mask}}\left(D_\psi^{\mathrm{rec}}(\tilde{z}_t^{(i)}, \hat{\xi}^{(i)}), o_t^{(i)}, M_t^{(i)}\right)\right], \tag{10}$$

where D_ψ^{rec} denotes the reconstruction decoder and $\mathcal{L}_{\text{mask}}(\hat{o}, o, M)$ denotes the entity-weighted reconstruction loss defined in Stage 1.

Overall Mapping Objective. The overall mapping objective combines these three losses:

$$\mathcal{L}_{\text{map}} = \lambda_{\text{cvc}}\mathcal{L}_{\text{cvc}} + \lambda_{\text{zero}}\mathcal{L}_{\text{zero}} + \lambda_{\text{rec}}\mathcal{L}_{\text{rec}}, \tag{11}$$

where λ_{cvc}, λ_{zero}, and λ_{rec} are loss weights. The resulting aligned representations are used for downstream policy learning.

4.3 Downstream Policy Learning

After the two stages of RCA, we obtain aligned representations across viewpoints for policy learning, using different downstream learning methods in simulation and real-world experiments.

For a fair comparison in simulation, we use COMBO (Yu et al., 2021), an offline model-based policy method, following ReViWo (Pang et al., 2025). COMBO requires training a world model for policy learning. However, the multi-view dataset contains only a few trajectories per task. We therefore collect additional single-view control data $\mathcal{D}_{\text{pol}}$ for COMBO: 400 trajectories per Meta-World task and 100 for PandaGym Reach. The policy is optimized using the collected data and synthetic rollouts generated by the learned world model.

In real-world experiments, collecting robot trajectories is costly, making offline model-based policy learning difficult with limited data. We therefore adopt ACT (Zhao et al., 2023), which allows us to learn a policy using only 50 single-view demonstrations. Algorithm 1 in the appendix summarizes the training procedure.

5 Experiments

We evaluate the viewpoint robustness of RCA in two simulation benchmarks, **Meta-World** (Yu et al., 2020) and **PandaGym** (Gallouédec et al., 2021), and on a **real-world** tabletop manipulation task using an AgileX Piper arm. Our experiments aim to answer the following research questions: **Q1:** How does RCA perform under viewpoint variations in simulation and the real world (Sections 5.2 and 5.3)? **Q2:** Does RCA better preserve task-relevant information and learn more viewpoint-robust representations than view-consistent methods (Section 5.4)? **Q3:** Can the viewpoint predictor estimate reliable viewpoint parameters for viewpoint-conditioned feature mapping (Section 5.5)? **Q4:** What is the contribution of each component in our framework to the overall performance (Section 5.6)?

5.1 Experiment Setup

Evaluation Environments. In Meta-World (Yu et al., 2020), we evaluate four manipulation tasks with a Sawyer robotic arm: Button Press, Window Close, Door Open, and Drawer Open. In PandaGym (Gallouédec et al., 2021), we evaluate the Reach task with a Franka Emika Panda arm. For real-world evaluation, we use an AgileX Piper arm for a tabletop pick-and-place task, where success requires placing a green can in a basket and releasing it.

Training Datasets. In simulation, we collect a multi-view dataset $\mathcal{D}_{\text{mv}}$ for representation learning, viewpoint prediction, and feature mapping. Following ReViWo (Pang et al., 2025), we sample 20 viewpoints within two 90° azimuth sectors for Meta-World and 20 viewpoints within a single 270° azimuth sector for PandaGym. Viewpoints outside these sampling sectors are out-of-distribution (OOD) in the 360° azimuth evaluation. We also use a subset of Open X-Embodiment (Vuong et al., 2024) as additional data for Stage-1 representation learning. For world-model and policy learning, we collect a separate single-view control dataset $\mathcal{D}_{\text{pol}}$ from a fixed policy-training viewpoint.

For the real-world task, we collect 50 demonstrations. For each demonstration, we replay its recorded action sequence while repositioning a single camera across six azimuths. We temporally align the recordings to construct a paired multi-view dataset. For downstream policy learning, we use only the recordings from a fixed viewpoint in this dataset. Dataset details are provided in Appendix A.2.

Viewpoint Variation Setup. All camera azimuth offsets are relative to the policy-training viewpoint. In simulation, we evaluate 36 azimuth offsets at 10° intervals over the 360° azimuth range. Except for the policy-training viewpoint, all evaluation azimuths are absent from both $\mathcal{D}_{\text{mv}}$ and $\mathcal{D}_{\text{pol}}$, establishing

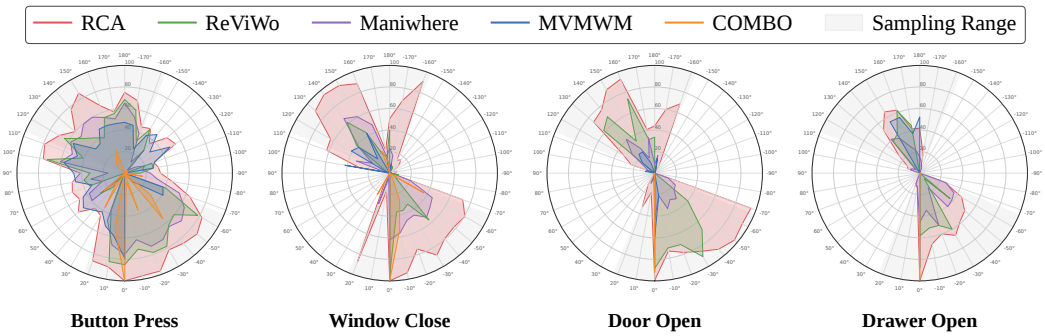


Figure 3: **Meta-World results under viewpoint variations over the full 360° azimuth range.** We report average success rates (%) across three seeds. Angles are relative to the policy-training viewpoint. Shading marks multi-view sampling sectors; unshaded regions are out-of-distribution.

Table 1: Performance under continuous camera shaking on Meta-World tasks. Results are reported as mean $\pm$ standard deviation over the final two evaluation success rates (%) from each of three seeds.

Method	Button Press	Window Close	Door Open	Drawer Open	Average
COMBO	13.33 ± 4.71	16.67 ± 4.22	1.67 ± 2.79	1.67 ± 2.79	8.34
MVMWM	56.11 ± 4.43	18.33 ± 6.58	10.56 ± 1.36	10.00 ± 5.97	23.75
Maniwhere	65.56 ± 13.93	27.23 ± 9.76	20.00 ± 7.60	26.67 ± 20.22	34.87
ReViWo	61.67 ± 3.50	26.67 ± 11.74	32.22 ± 4.04	30.00 ± 6.67	37.64
RCA	$\mathbf{87.78 \pm 5.84}$	$\mathbf{61.67 \pm 4.60}$	$\mathbf{60.00 \pm 18.62}$	$\mathbf{47.78 \pm 15.00}$	**64.31**

a zero-shot viewpoint generalization setting. Evaluation viewpoints outside the multi-view sampling sectors are considered out-of-distribution (OOD). We further evaluate continuous camera shaking on the four Meta-World tasks by perturbing the camera azimuth over time during policy execution.

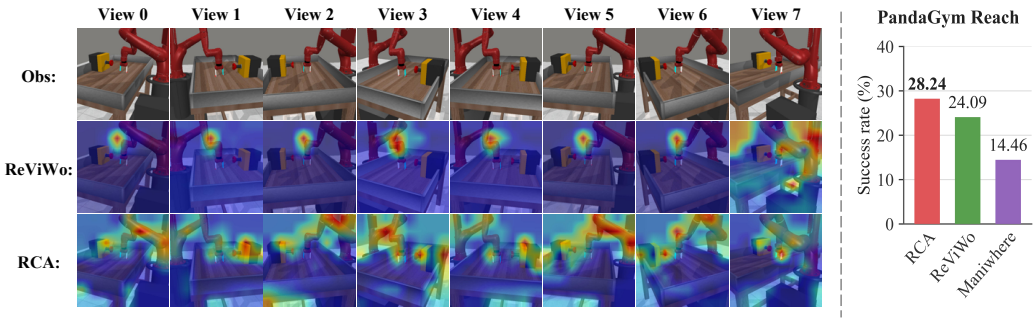
5.2 Simulation Experiments

Baselines. We compare RCA with state-of-the-art baselines for viewpoint-robust robotic manipulation and offline model-based control. **(1) ReViWo** (Pang et al., 2025) decomposes observations into view-consistent and view-dependent representations. **(2) Maniwhere** (Yuan et al., 2024) combines multi-view representation learning with spatial transformer networks, which we adapt to our data setting. **(3) MVMWM** (Seo et al., 2023b) learns multi-view representations with a masked autoencoder by reconstructing randomly masked viewpoints. **(4) COMBO** (Yu et al., 2021) is an offline model-based reinforcement learning baseline; we equip it with a standard VAE (Kingma & Welling, 2014) to encode visual observations. For fair comparison, all baselines learn representations from the same multi-view dataset and learn policies from the same single-view control dataset.

Evaluation in Meta-World. Figure 3 reports average success rates over the full 360° azimuth range. Across the four Meta-World tasks, RCA achieves an average success rate of 42.41%, compared with 22.58% for ReViWo, 22.24% for Maniwhere, and 6.44% for COMBO. RCA generally performs better at viewpoints where the baselines have nonzero success rates, while supporting successful manipulation over a wider angular range. Substantial occlusion near $\pm 90^\circ$ may contribute to performance degradation. ID and OOD results are reported separately in Appendix A.7.

Under continuous camera shaking, as shown in Table 1, RCA achieves the highest average success rate on all four tasks, with a 70.85% relative improvement over ReViWo. To account for other perturbations that may arise during real-world deployment, we additionally evaluate camera pitch, distance, and height changes (Appendix A.8), as well as combined perturbations (Appendix A.9).

Evaluation in PandaGym. We evaluate RCA on PandaGym Reach and compare it with ReViWo and Maniwhere. As shown in Figure 4(b), RCA achieves the highest average success rate over the 36 evaluated viewpoints, with a 17.23% relative improvement over ReViWo. These results further support RCA's improved viewpoint robustness. Angle-wise average success rates for Meta-World and PandaGym under viewpoint variations are provided in Appendix A.6.



(a) Attention Map **(b) PandaGym**

Figure 4: **Attention maps and PandaGym results.** (a) Attention maps of ReViWo and RCA's mapped representations across camera viewpoints. RCA attends more comprehensively to task-relevant regions. (b) Average success rates (%) over all 36 evaluated viewpoints on PandaGym Reach.

Table 2: Real-world success counts under azimuth offsets. The highest value is highlighted in bold.

Method	0°	−10°	−20°	−30°	−40°	−50°	−60°	10°	20°	30°	40°	50°	60°	Average (%)
RCA	**10/10**	**9/10**	**9/10**	**9/10**	**7/10**	**8/10**	**6/10**	7/10	**6/10**	**6/10**	**6/10**	**7/10**	**5/10**	**73.08**
$\pi_{0.5}$	9/10	**9/10**	**9/10**	8/10	6/10	6/10	**6/10**	**8/10**	**6/10**	5/10	5/10	5/10	4/10	66.15
ReViWo	7/10	6/10	5/10	4/10	4/10	3/10	3/10	5/10	4/10	4/10	3/10	4/10	3/10	42.31
Maniwhere	5/10	5/10	3/10	3/10	3/10	3/10	2/10	4/10	2/10	3/10	2/10	3/10	2/10	30.77
ACT	6/10	4/10	1/10	0/10	0/10	0/10	0/10	3/10	1/10	0/10	0/10	0/10	0/10	11.54

5.3 Real-World Experiments

Baselines. We compare RCA with ReViWo, Maniwhere, ACT, and $\pi_{0.5}$ (Physical Intelligence et al., 2025). RCA, ReViWo, and Maniwhere use the same paired multi-view data to learn representations and train identical ACT policy heads from scratch on single-view demonstrations. Standard ACT is trained on the same single-view data, while the pretrained VLA baseline $\pi_{0.5}$ is fine-tuned on them.

Azimuth Generalization. We report successful trials out of ten at each offset and the average success rate across 13 settings. As shown in Table 2, RCA outperforms all baselines in average success rate, achieving 73.08%, compared with 66.15% for $\pi_{0.5}$. Notably, RCA has only 120.87M parameters, whereas $\pi_{0.5}$ has approximately 4B parameters and has undergone large-scale pretraining. Additional evaluations under camera pitch, distance, and height changes are provided in Appendix A.10.

5.4 Qualitative Analysis of Learned Representations

Visualization of Task-Relevant Attention. We compare attention maps of RCA's mapped representations and ReViWo on Meta-World Button Press. As shown in Figure 4(a), RCA attends more comprehensively to task-relevant regions, such as the robot arm, gripper, and target objects, whereas ReViWo focuses more on overlapping regions across views, such as the robot wrist. This suggests that RCA retains task-relevant information while improving viewpoint robustness.

t-SNE Analysis of Viewpoint Robustness. We visualize RCA's mapped representations using t-SNE (Van der Maaten & Hinton, 2008) on Button Press, with 64 observations covering 8 states and 8 viewpoints per state. As shown in Figure 5, RCA forms more compact same-state clusters and better-separated different-state clusters than the baselines. This suggests that RCA improves cross-view alignment while preserving state-discriminative information.

5.5 Reliability of Viewpoint Predictor

We evaluate the viewpoint predictor trained on Meta-World multi-view data from 20 viewpoints sampled within two 90° azimuth sectors. When evaluated at 10° intervals over the full 360° azimuth range on Meta-World Button Press (36 evaluation points), the overall mean angular error is 9.38°. Only four viewpoints exhibit mean angular errors exceeding 20°, while the remaining 32 viewpoints

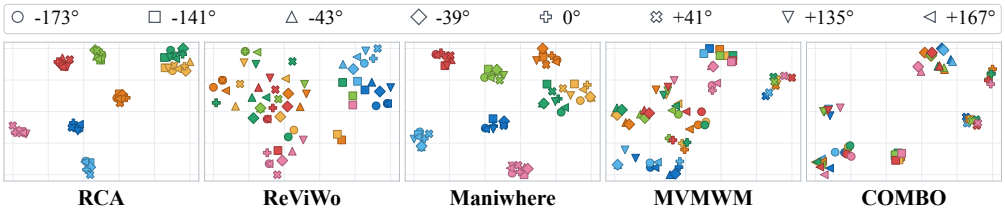


Figure 5: **Cross-view t-SNE visualization.** Colors denote states; marker shapes denote viewpoints.

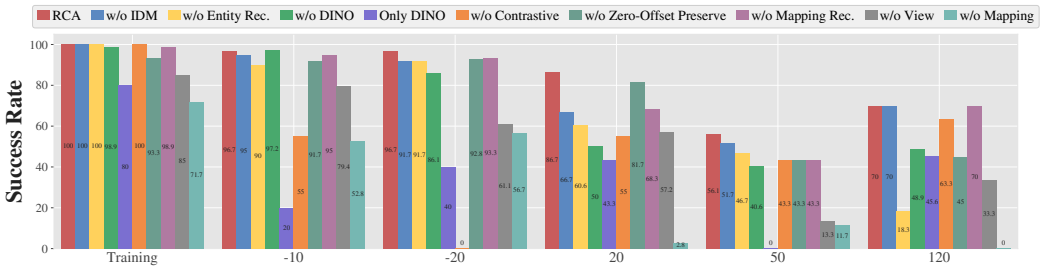


Figure 6: **Ablation study.** We report mean success rates (%) on Meta-World Button Press task. IDM denotes inverse dynamics supervision, Entity Rec. denotes entity-centric masked reconstruction, Contrastive denotes cross-view contrastive loss, Mapping Rec. denotes mapping reconstruction loss.

achieve an average error of 6.27°. As reported in Appendix A.12, downstream policies are relatively robust to this level of prediction error. Detailed angle-wise errors are provided in Appendix A.11.

5.6 Ablation Study

We evaluate component contributions on Meta-World Button Press. As shown in Figure 6, full RCA achieves the highest average success rate across the six viewpoints, including the training viewpoint.

Task-Relevant Representation Learning. (1) w/o Inverse Dynamics or Entity-Centric Masked Reconstruction: These individual ablations reduce performance, supporting the two objectives' roles in preserving motion-related information and visual information about task entities, respectively. (2) Only DINOv3: Using only frozen DINOv3 features with viewpoint-conditioned mapping underperforms full RCA, indicating that frozen visual priors cannot fully replace learned task-relevant features. Both w/o Entity-Centric Reconstruction and w/o DINOv3 Features outperform the baselines on average across the six evaluated viewpoints, suggesting that RCA's gains are not solely attributable to external pretrained models. Appendix A.17 further examines external-model dependence.

Viewpoint-Conditioned Feature Mapping. (1) w/o Contrastive: Removing cross-view contrastive loss substantially reduces average success. (2) w/o Zero-Offset Preservation or Mapping Reconstruction Loss: Each ablation reduces performance, supporting their respective roles in limiting zero-offset distortion and preserving information during mapping. (3) w/o View: Removing viewpoint conditioning substantially reduces performance, highlighting its importance for feature adaptation. (4) w/o Mapping: Applying contrastive supervision directly to task-relevant representations without FiLM mapping causes the largest average success drop. Additional results are provided in Appendix A.13.

6 Conclusion

We propose Rotation-aware Consistent Alignment (RCA) for viewpoint-robust robotic manipulation. RCA prioritizes capturing task-relevant features. RCA then accounts for camera rotation to adapt viewpoint-dependent features and align them in a unified latent space. This design improves cross-view feature consistency while preserving task-relevant information. Extensive experiments in Meta-World and PandaGym simulations and on a real-world robot demonstrate RCA's improved viewpoint robustness to diverse viewpoint perturbations.

AI USE STATEMENT

In this work, we used generative AI tools to aid or polish writing, including revising author-written text and suggesting improvements to clarity and organization. These tools also assisted with LaTeX formatting, terminology consistency checks, and discussions of methodological descriptions.

We did not use generative AI tools for research ideation or execution, or for generating synthetic datasets. AI assistance with proving mathematical claims is not applicable to this work.

We reviewed all AI-assisted work and independently evaluated the suggested revisions, checking that the final text accurately describes our method and experimental findings. We take responsibility for the final content of this work, including text, claims, and artifacts produced with the aid of generative AI.

REPRODUCIBILITY STATEMENT

Section 4 describes the model and learning objectives, and Algorithm 1 summarizes the training procedure, including downstream policy learning. Appendix A documents dataset construction, training viewpoints, model architectures, hyperparameters, evaluation protocols, and computing resources. Appendix C.3 further describes the generation of entity-centric masks used to weight reconstruction losses.

REFERENCES

Rishabh Agarwal, Dale Schuurmans, and Mohammad Norouzi. An optimistic perspective on offline reinforcement learning. In *International conference on machine learning*, pp. 104–114. PMLR, 2020.

Anthony Brohan, Noah Brown, Justice Carbajal, Yevgen Chebotar, Joseph Dabis, Chelsea Finn, Keerthana Gopalakrishnan, Karol Hausman, Alex Herzog, Jasmine Hsu, et al. RT-1: robotics transformer for real-world control at scale. In *Robotics: Science and Systems*, 2023.

Yuanpei Chen, Tianhao Wu, Shengjie Wang, Xidong Feng, Jiechuan Jiang, Zongqing Lu, Stephen McAleer, Hao Dong, Song-Chun Zhu, and Yaodong Yang. Towards human-level bimanual dexterous manipulation with reinforcement learning. *Advances in Neural Information Processing Systems*, 35:5150–5163, 2022.

Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. In *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*, 2021.

Norm Ferns, Prakash Panangaden, and Doina Precup. Metrics for finite markov decision processes. In *Uai*, volume 4, pp. 162–169, 2004.

Quentin Gallouédec, Nicolas Cazin, Emmanuel Dellandréa, and Liming Chen. panda-gym: Open-Source Goal-Conditioned Environments for Robotic Learning. *4th Robot Learning Workshop: Self-Supervised and Lifelong Learning at NeurIPS*, 2021.

Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross Girshick. Masked autoencoders are scalable vision learners. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 16000–16009, 2022.

Dmitry Kalashnikov, Alex Irpan, Peter Pastor, Julian Ibarz, Alexander Herzog, Eric Jang, Deirdre Quillen, Ethan Holly, Mrinal Kalakrishnan, Vincent Vanhoucke, et al. Scalable deep reinforcement learning for vision-based robotic manipulation. In *Conference on robot learning*, pp. 651–673. PMLR, 2018.

Moo Jin Kim, Karl Pertsch, Siddharth Karamcheti, Ted Xiao, Ashwin Balakrishna, Suraj Nair, Rafael Rafailov, Ethan Foster, Grace Lam, Pannag Sanketi, et al. Openvla: An open-source vision-language-action model. In *Conference on Robot Learning*, pp. 2679–2713. PMLR, 2024.

Moo Jin Kim, Yihuai Gao, Tsung-Yi Lin, Yen-Chen Lin, Yunhao Ge, Grace Lam, Percy Liang, Shuran Song, Ming-Yu Liu, Chelsea Finn, et al. Cosmos policy: Fine-tuning video models for visuomotor control and planning. *arXiv preprint arXiv:2601.16163*, 2026.

Diederik P. Kingma and Max Welling. Auto-encoding variational bayes. In *International Conference on Learning Representations*, 2014.

Michael Laskin, Aravind Srinivas, and Pieter Abbeel. Curl: Contrastive unsupervised representations for reinforcement learning. In *International conference on machine learning*, pp. 5639–5650. PMLR, 2020.

Fanfan Liu, Feng Yan, Liming Zheng, Chengjian Feng, Yiyang Huang, and Lin Ma. Robouniview: Visual-language model with unified view representation for robotic manipulation. *arXiv preprint arXiv:2406.18977*, 2024a.

Songming Liu, Lingxuan Wu, Bangguo Li, Hengkai Tan, Huayu Chen, Zhengyi Wang, Ke Xu, Hang Su, and Jun Zhu. Rdt-1b: a diffusion foundation model for bimanual manipulation. *arXiv preprint arXiv:2410.07864*, 2024b.

Jianlan Luo, Charles Xu, Jeffrey Wu, and Sergey Levine. Precise and dexterous robotic manipulation via human-in-the-loop reinforcement learning. *Science Robotics*, 10(105):eads5033, 2025.

Suraj Nair, Aravind Rajeswaran, Vikash Kumar, Chelsea Finn, and Abhinav Gupta. R3M: A universal visual representation for robot manipulation. In *Conference on Robot Learning*, pp. 892–909. PMLR, 2022.

Aaron van den Oord, Yazhe Li, and Oriol Vinyals. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*, 2018.

Jing-Cheng Pang, Nan Tang, Kaiyuan Li, Yuting Tang, Xin-Qiang Cai, Zhen-Yu Zhang, Gang Niu, Masashi Sugiyama, and Yang Yu. Learning view-invariant world models for visual robotic manipulation. In *International Conference on Learning Representations*, 2025.

Ethan Perez, Florian Strub, Harm De Vries, Vincent Dumoulin, and Aaron Courville. Film: Visual reasoning with a general conditioning layer. In *Proceedings of the AAAI conference on artificial intelligence*, volume 32, 2018.

Physical Intelligence, Kevin Black, Noah Brown, James Darpinian, Karan Dhabalia, Danny Driess, Adnan Esmail, Michael Equi, Chelsea Finn, Niccolo Fusai, et al. $\pi_{0.5}$: a vision-language-action model with open-world generalization. *arXiv preprint arXiv:2504.16054*, 2025.

Deirdre Quillen, Eric Jang, Ofir Nachum, Chelsea Finn, Julian Ibarz, and Sergey Levine. Deep reinforcement learning for vision-based robotic grasping: A simulated comparative evaluation of off-policy methods. In *2018 IEEE international conference on robotics and automation (ICRA)*, pp. 6284–6291. IEEE, 2018.

Ilija Radosavovic, Tete Xiao, Stephen James, Pieter Abbeel, Jitendra Malik, and Trevor Darrell. Real-world robot learning with masked visual pre-training. In *Conference on Robot Learning*, pp. 416–426. PMLR, 2023.

Tianhe Ren, Shilong Liu, Ailing Zeng, Jing Lin, Kunchang Li, He Cao, Jiayu Chen, Xinyu Huang, Yukang Chen, Feng Yan, et al. Grounded sam: Assembling open-world models for diverse visual tasks. *arXiv preprint arXiv:2401.14159*, 2024.

Younggyo Seo, Danijar Hafner, Hao Liu, Fangchen Liu, Stephen James, Kimin Lee, and Pieter Abbeel. Masked world models for visual control. In *Conference on Robot Learning*, pp. 1332–1344. PMLR, 2023a.

Younggyo Seo, Junsu Kim, Stephen James, Kimin Lee, Jinwoo Shin, and Pieter Abbeel. Multi-view masked world models for visual robotic manipulation. In *International Conference on Machine Learning*, pp. 30613–30632. PMLR, 2023b.

Pierre Sermanet, Corey Lynch, Yevgen Chebotar, Jasmine Hsu, Eric Jang, Stefan Schaal, Sergey Levine, and Google Brain. Time-contrastive networks: Self-supervised learning from video. In *2018 IEEE international conference on robotics and automation (ICRA)*, pp. 1134–1141. IEEE, 2018.

Oriane Siméoni, Huy V Vo, Maximilian Seitzer, Federico Baldassarre, Maxime Oquab, Cijo Jose, Vasil Khalidov, Marc Szafraniec, Seungeun Yi, Michaël Ramamonjisoa, et al. Dinov3. *arXiv preprint arXiv:2508.10104*, 2025.

Aaron Van Den Oord, Oriol Vinyals, et al. Neural discrete representation learning. *Advances in neural information processing systems*, 30, 2017.

Laurens Van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. *Journal of machine learning research*, 9(11), 2008.

Quan Vuong, Sergey Levine, Homer Rich Walke, Karl Pertsch, Anikait Singh, Ria Doshi, Charles Xu, Jianlan Luo, Liam Tan, Dhruv Shah, et al. Open x-embodiment: Robotic learning datasets and RT-X models : Open x-embodiment collaboration. In *IEEE International Conference on Robotics and Automation, ICRA*, pp. 6892–6903. IEEE, 2024.

Tianhe Yu, Deirdre Quillen, Zhanpeng He, Ryan Julian, Karol Hausman, Chelsea Finn, and Sergey Levine. Meta-world: A benchmark and evaluation for multi-task and meta reinforcement learning. In *Conference on robot learning*, pp. 1094–1100. PMLR, 2020.

Tianhe Yu, Aviral Kumar, Rafael Rafailov, Aravind Rajeswaran, Sergey Levine, and Chelsea Finn. Combo: Conservative offline model-based policy optimization. *Advances in neural information processing systems*, 34:28954–28967, 2021.

Zhecheng Yuan, Zhengrong Xue, Bo Yuan, Xueqian Wang, Yi Wu, Yang Gao, and Huazhe Xu. Pre-trained image encoder for generalizable visual reinforcement learning. *Advances in Neural Information Processing Systems*, 35:13022–13037, 2022.

Zhecheng Yuan, Tianming Wei, Shuiqi Cheng, Gu Zhang, Yuanpei Chen, and Huazhe Xu. Learning to manipulate anywhere: A visual generalizable framework for reinforcement learning. In *Conference on Robot Learning, 6-9 November 2024, Munich, Germany*, Proceedings of Machine Learning Research, pp. 1815–1833. PMLR, 2024.

Amy Zhang, Rowan Thomas McAllister, Roberto Calandra, Yarin Gal, and Sergey Levine. Learning invariant representations for reinforcement learning without reconstruction. In *International Conference on Learning Representations, ICLR*, 2021.

Tony Z. Zhao, Vikash Kumar, Sergey Levine, and Chelsea Finn. Learning fine-grained bimanual manipulation with low-cost hardware. In *Robotics: Science and Systems XIX, Daegu, Republic of Korea, July 10-14, 2023*, 2023.

Brianna Zitkovich, Tianhe Yu, Sichun Xu, Peng Xu, Ted Xiao, Fei Xia, Jialin Wu, Paul Wohlhart, Stefan Welker, Ayzaan Wahid, et al. Rt-2: Vision-language-action models transfer web knowledge to robotic control. In *Conference on Robot Learning*, pp. 2165–2183. PMLR, 2023.

APPENDIX CONTENTS

A Experimental Details and Additional Results **14**

A.1 Robotic Manipulation Environments 14

A.2 Dataset Details 15

A.3 Viewpoints for Representation and RL Training 15

A.4 Architecture and Optimization Details 15

A.5 Evaluation Protocol for Simulated Control Tasks 16

A.6 Angle-wise Success Rates over the Full 360° Azimuth Range 16

A.7 Viewpoint Generalization within and beyond Sampling Sectors 22

A.8 Generalization to Camera Pitch, Distance, and Height Changes in Simulation . . . 22

A.9 Azimuth Variations with Additional Camera Perturbations 25

A.10 Generalization to Camera Pitch, Distance, and Height Changes in the Real World . 27

A.11 Angle-wise Viewpoint Prediction Errors 28

A.12 Policy Sensitivity to Viewpoint Prediction Errors 28

A.13 Additional Ablation Study 30

A.14 Mapping Architecture: FiLM versus MLP 31

A.15 Cross-View Similarity Analysis 31

A.16 Task-Relevant Information Retention 31

A.17 Contribution of External Pretrained Models 33

A.18 Hyperparameters 34

A.19 Compute Resources and Software Environment 34

B Algorithm Description **36**

C Additional Visualizations **36**

C.1 Azimuth Variation Visualization 36

C.2 Additional Attention Map Visualization 36

C.3 Entity-Centric Mask Generation Pipeline 36

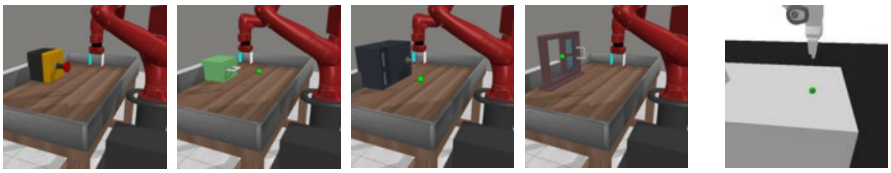
D Limitations and Future Work **40**

A Experimental Details and Additional Results

Overview of Additional Experiments. We provide additional evaluations under camera pitch, distance, and height changes in simulation (Appendix A.8), combined camera perturbations (Appendix A.9), and camera pitch, distance, and height changes in the real world (Appendix A.10). We further examine downstream policy sensitivity to viewpoint prediction errors (Appendix A.12) and conduct additional ablations on Meta-World Window Close task (Appendix A.13). Finally, we evaluate task-relevant information retention (Appendix A.16) and examine whether RCA's performance gains are solely attributable to external pretrained models (Appendix A.17).

A.1 Robotic Manipulation Environments

Simulation Environments. Figure 7 visualizes the two simulated manipulation environments used in our experiments: Meta-World and PandaGym. For Meta-World, we evaluate four tasks: Button Press, Window Close, Door Open, and Drawer Open. These tasks involve precise contact and articulated-object manipulation, requiring policies to preserve robot–object spatial relations under viewpoint variations. For PandaGym, we evaluate the Reach task, where a Franka Emika Panda arm must reach a target position.



(A) Meta-world (B) PandaGym

Figure 7: Simulated manipulation environments. (A) Meta-World: Button Press, Drawer Open, Door Open, and Window Close. (B) PandaGym Reach.

Real-World Environment. Figure 8 shows our real-world tabletop manipulation setup, consisting of an AgileX Piper arm, a single external camera, a green can, and a basket. The robot performs a pick-and-place task with the instruction *"Place the green can into the basket."* A trial is successful when the robot places the can into the basket and releases it. We vary the external camera viewpoint to evaluate viewpoint robustness. The real-world dataset construction is described in Appendix A.2.

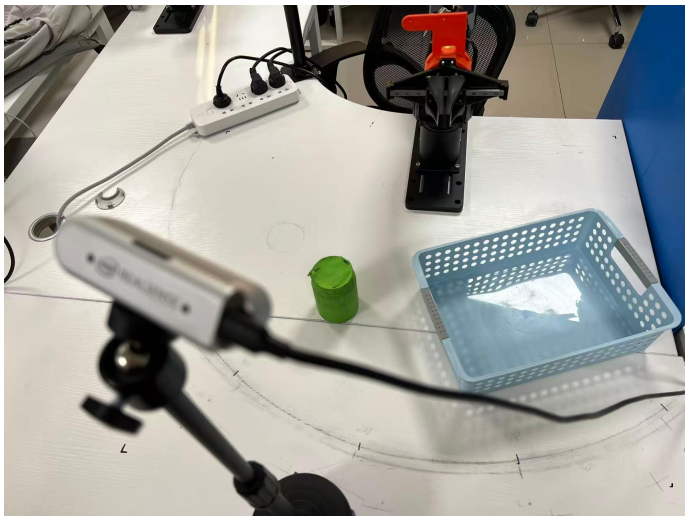


Figure 8: Real-world tabletop manipulation setup with an AgileX Piper arm, a single external camera, a green can, and a basket.

A.2 Dataset Details

Simulation Data. For the multi-view dataset used in representation learning, viewpoint prediction, and feature mapping, we collect 51 Meta-World trajectories across 17 tasks with 20 cameras placed in two 90° azimuth sectors, resulting in 1,020 observation sequences and 112,040 frames. For PandaGym, we collect 36 trajectories across 6 tasks with 20 viewpoints restricted to a single 270° sector. The trajectories are sampled using expert policies with Gaussian noise. The viewpoint predictors for the two simulated environments are trained separately using their corresponding multi-view datasets.

We use a subset of the Routing Primitive Dataset from Open X-Embodiment (Vuong et al., 2024) as auxiliary data for Stage-1 representation learning. This subset contains 101 trajectories recorded from 4 viewpoints, with 10,064 observations in total. For a fair comparison, all baselines use the same auxiliary data in addition to the same task-specific multi-view dataset. For downstream world-model and policy learning, we collect an additional single-view control dataset $\mathcal{D}_{\text{pol}} = \{(o_t, a_t, r_t, o_{t+1})\}$ from a fixed policy-training viewpoint, where r_t denotes the reward. The dataset contains 400 trajectories per task in Meta-World and 100 for PandaGym Reach.

Real-World Data. We collect 50 demonstrations for a tabletop can-placement task using an AgileX Piper arm. Each demonstration is recorded from six camera azimuths: 25°, 50°, 75°, 100°, 125°, and 150°. To reduce camera hardware requirements, we reposition a single camera and replay each recorded demonstration action sequence at each azimuth, while controlling the initial robot, gripper, and object states. The recordings are temporally aligned to construct paired six-view observations with corresponding action labels, avoiding the need for simultaneous multi-camera recording. The dataset contains 22,965 aligned six-view time steps across the 50 demonstrations. These observations are used to train the visual encoder, viewpoint predictor, and FiLM mapping network. For downstream policy learning, we train an ACT policy head from scratch using only the 100° view and its corresponding action labels.

A.3 Viewpoints for Representation and RL Training

In simulation, we collect multi-view data from restricted azimuth sectors to train the Stage-1 encoder, viewpoint predictor, and Stage-2 FiLM mapping network. Apart from the policy-training viewpoint, the evaluation viewpoints over the full 360° azimuth range include both viewpoints within the sampled sectors but absent from the multi-view dataset and out-of-distribution viewpoints outside these sectors.

Specifically, the azimuth angles employed in Meta-World are {0°, 4°, 5°, 16°, 22.5°, 24°, 41°, 135°, 167°, 170°, 179°, 187°, 207°, 219°, 317°, 321°, 330°, 332°, 342°, 350°}. These viewpoints are sampled from two 90° azimuth sectors, $[-45^\circ, 45^\circ]$ and $[135^\circ, 225^\circ]$, with a total angular coverage of 180°. For PandaGym, the azimuth angles are {6°, 21°, 45°, 60°, 69°, 84°, 114°, 126°, 153°, 159°, 162°, 170°, 179°, 206°, 209°, 213°, 217°, 224°, 238°, 244°}, selected from a single 270° azimuth sector, $[0^\circ, 270^\circ]$.

For downstream RL training, we use a fixed training viewpoint: 22.5° for Meta-World and 45° for PandaGym. Thus, the policy is learned only from single-view control data, without access to other viewpoints during policy training.

For viewpoint variation over the full 360° azimuth range, we evaluate at 10° intervals relative to the policy-training viewpoint, which corresponds to the 0° offset. Except for the policy-training viewpoint, all evaluation viewpoints are absent from both the multi-view dataset and the single-view control dataset, establishing a zero-shot viewpoint generalization setting. Moreover, many evaluation viewpoints lie outside the sampled azimuth sectors, enabling out-of-distribution (OOD) evaluation. Visualizations of different viewpoints are provided in Figure 14.

A.4 Architecture and Optimization Details

Stage-1 Encoder. The Stage-1 visual encoder is based on ViT (Dosovitskiy et al., 2021) and operates on RGB observations of size 128×128. Each image is divided into non-overlapping 16×16 patches, yielding 64 visual tokens. Both the encoder and decoder are implemented as 8-layer

transformers with a hidden dimension of 512 and 8 attention heads. Patch embeddings are produced by a strided convolutional patch tokenizer. The encoded features are projected to a 64-dimensional token space and quantized with a VQ bottleneck using a codebook size of 512 and an embedding dimension of 64. The Stage-1 encoder is trained with the VQ-VAE quantization objective, together with the inverse dynamics and entity-centric masked reconstruction losses:

$$\mathcal{L}_{\text{task}} = \mathcal{L}_{\text{vq-code}} + \lambda_{\text{idm}}\mathcal{L}_{\text{idm}} + \lambda_{\text{mask}}\mathcal{L}_{\text{mask}}.$$

Here, $\mathcal{L}_{\text{vq-code}}$ denotes the codebook and commitment losses of the VQ-VAE tokenizer. We set the loss weights to

$$\lambda_{\text{idm}} = 1.0, \qquad \lambda_{\text{mask}} = 1.0.$$

The inverse dynamics head is a two-hidden-layer MLP operating on consecutive latent features.

Stage-2 FiLM Mapping Network. The Stage-2 mapping network maps task-relevant representations into the unified latent space. It is implemented as a stack of FiLM-modulated blocks (Perez et al., 2018). We represent the camera azimuth offset $\Delta\alpha^{(i)}$ relative to the policy-training viewpoint using the sine–cosine encoding $\xi^{(i)} = [\sin(\Delta\alpha^{(i)}), \cos(\Delta\alpha^{(i)})]^\top$. The viewpoint predictor estimates this encoding as $\hat{\xi}^{(i)} = Q_\nu(\text{DINO}(o_t^{(i)}))$, which is used directly as the conditioning input. A conditioning MLP with hidden dimensions $(128, 128)$ maps $\hat{\xi}^{(i)}$ to modulation parameters γ_l and β_l for each FiLM block l, which apply feature-wise scaling by $1 + \gamma_l$ and shifting by β_l. These parameters modulate the intermediate features of the mapping network to align representations across viewpoints. The mapping network uses hidden layers of width 512, and a residual skip connection from the input to the output is added to stabilize optimization and preserve the input task-relevant representation.

A.5 Evaluation Protocol for Simulated Control Tasks

360° **Azimuth Variation.** To assess zero-shot viewpoint generalization over the full 360° azimuth range, we adopt a standardized evaluation protocol. We evaluate at 10° azimuth intervals relative to the policy-training viewpoint, without retraining or fine-tuning. We use the final two evaluation success rates from each training seed. Across our three training seeds, we report the mean and sample standard deviation over the resulting six scores. Each evaluation consists of 30 episodes.

Continuous Camera Shaking. In addition to static azimuth variations, we evaluate RCA under continuous camera shaking to simulate time-varying viewpoint perturbations during test-time policy execution. The initial camera azimuth is shifted by -5° relative to the policy-training viewpoint. During rollout, the camera azimuth changes by 1° at each environment step. The perturbation direction is uniformly sampled from $\{-1, +1\}$ and held fixed for 10 consecutive environment steps. The direction is resampled every 10 steps, and the process continues throughout the rollout. Results follow the same aggregation protocol as in the azimuth evaluation.

A.6 Angle-wise Success Rates over the Full 360° Azimuth Range

In addition to the polar plots shown in the main text, we report detailed angle-wise average success rates and standard deviations over the full 360° azimuth range. In all angle-wise result tables, angles denote azimuth offsets relative to the policy-training viewpoint.

For Meta-World, Tables 3–6 provide the results for Button Press, Window Close, Door Open, and Drawer Open. Table 7 summarizes the task-wise and overall average success rates. RCA achieves the highest average success rate on each task, with an overall average of 42.41%, compared with 22.58% for ReViWo, the strongest baseline by overall average success rate.

For PandaGym, Table 8 and Figure 9 report the angle-wise results on the Reach task. Figure 4(b) summarizes the average success rates over all 36 evaluated viewpoints. RCA achieves the highest average success rate of 28.24%, compared with 24.09% for ReViWo and 14.46% for Maniwhere, outperforming the strongest baseline by 4.15 percentage points.

Table 3: Angle-wise success rates (%) on Meta-World Button Press. Results are reported as mean ± standard deviation over the final two evaluation success rates from each of three training seeds. Offsets are relative to the policy-training viewpoint. Shaded rows indicate viewpoints within the multi-view sampling sectors (ID). Bold marks the highest nonzero mean at each offset.

Offset (°)	RCA	ReViWo	Maniwhere	MVMWM	COMBO
0	**100.00** ± 0.00	85.00 ± 9.61	75.56 ± 5.02	78.89 ± 21.25	**100.00** ± 0.00
10	**88.33** ± 8.10	83.33 ± 7.89	67.77 ± 25.27	58.89 ± 11.29	51.67 ± 1.82
20	**86.67** ± 7.30	36.67 ± 3.65	46.67 ± 20.22	52.22 ± 8.35	0.00 ± 0.00
30	**45.00** ± 8.88	0.00 ± 0.00	3.33 ± 2.98	41.67 ± 6.58	36.67 ± 16.73
40	42.78 ± 12.55	17.23 ± 9.05	6.67 ± 11.55	**45.00** ± 22.49	11.67 ± 3.50
50	**56.11** ± 4.43	29.44 ± 17.31	39.44 ± 11.44	50.00 ± 7.60	25.00 ± 15.02
60	**48.33** ± 5.87	0.00 ± 0.00	37.78 ± 17.85	44.45 ± 2.72	0.00 ± 0.00
70	**51.67** ± 5.87	34.44 ± 2.72	22.23 ± 22.48	32.23 ± 10.26	0.00 ± 0.00
80	**49.44** ± 6.80	38.33 ± 13.29	31.66 ± 17.22	46.11 ± 8.27	0.00 ± 0.00
90	41.11 ± 6.88	32.23 ± 17.47	15.56 ± 20.62	**47.22** ± 2.51	0.00 ± 0.00
100	**76.67** ± 18.38	72.78 ± 6.81	60.56 ± 17.05	57.23 ± 11.04	0.00 ± 0.00
110	**78.89** ± 29.49	41.11 ± 4.04	51.67 ± 26.90	46.67 ± 4.71	0.00 ± 0.00
120	**70.00** ± 19.44	64.44 ± 14.24	57.78 ± 21.88	55.00 ± 6.91	0.00 ± 0.00
130	**60.00** ± 17.39	50.00 ± 13.33	52.78 ± 9.76	35.00 ± 3.50	8.33 ± 4.08
140	**77.22** ± 12.00	42.78 ± 9.05	59.44 ± 27.92	35.55 ± 10.89	11.67 ± 11.69
150	**85.56** ± 10.68	37.78 ± 5.84	60.00 ± 23.85	47.23 ± 7.43	15.00 ± 7.53
160	**66.67** ± 17.64	55.00 ± 5.05	55.00 ± 11.30	47.23 ± 5.34	23.33 ± 6.67
170	**57.78** ± 23.63	56.11 ± 3.90	51.11 ± 26.39	46.67 ± 7.89	0.00 ± 0.00
-10	**96.67** ± 4.22	71.67 ± 5.05	62.78 ± 18.67	0.00 ± 0.00	8.33 ± 5.05
-20	**96.67** ± 4.22	78.33 ± 5.05	72.22 ± 15.30	0.00 ± 0.00	36.67 ± 2.98
-30	**82.22** ± 22.08	78.33 ± 3.50	71.11 ± 14.71	0.00 ± 0.00	0.00 ± 0.00
-40	**85.00** ± 10.05	63.34 ± 3.65	64.45 ± 15.73	10.56 ± 2.51	55.00 ± 1.83
-50	**87.78** ± 10.68	60.00 ± 3.65	68.89 ± 10.68	15.00 ± 4.59	31.67 ± 5.05
-60	**82.78** ± 18.43	78.33 ± 5.05	63.89 ± 20.59	41.11 ± 3.44	0.00 ± 0.00
-70	**64.44** ± 22.77	40.55 ± 11.82	53.89 ± 18.91	37.78 ± 9.11	0.00 ± 0.00
-80	**45.00** ± 22.09	0.00 ± 0.00	30.55 ± 16.52	0.00 ± 0.00	16.67 ± 13.33
-90	**33.34** ± 15.92	14.44 ± 10.03	23.89 ± 21.23	0.00 ± 0.00	0.00 ± 0.00
-100	**31.67** ± 25.97	27.22 ± 5.34	16.67 ± 12.29	26.67 ± 2.11	0.00 ± 0.00
-110	28.89 ± 13.45	26.67 ± 3.65	19.45 ± 14.36	**29.44** ± 4.43	3.33 ± 2.11
-120	**54.44** ± 13.93	0.00 ± 0.00	11.11 ± 6.89	48.34 ± 10.06	1.67 ± 1.82
-130	**51.11** ± 11.86	0.00 ± 0.00	4.44 ± 5.02	30.56 ± 15.41	1.67 ± 1.82
-140	35.55 ± 16.01	27.78 ± 7.20	27.78 ± 13.77	**46.67** ± 8.16	3.33 ± 3.65
-150	**47.78** ± 22.08	46.67 ± 8.69	11.67 ± 13.62	40.56 ± 13.07	0.00 ± 0.00
-160	**45.56** ± 28.02	31.67 ± 5.05	38.89 ± 15.30	23.89 ± 3.90	0.00 ± 0.00
-170	**68.34** ± 7.53	59.44 ± 7.43	52.78 ± 28.23	45.00 ± 6.24	0.00 ± 0.00
-180	**75.00** ± 28.88	68.33 ± 3.50	65.00 ± 26.31	47.22 ± 3.90	0.00 ± 0.00

Table 4: Angle-wise success rates (%) on Meta-World Window Close. Results are reported as mean ± standard deviation over the final two evaluation success rates from each of three training seeds. Offsets are relative to the policy-training viewpoint. Shaded rows indicate viewpoints within the multi-view sampling sectors (ID). Bold marks the highest nonzero mean at each offset.

Offset (°)	RCA	ReViWo	Maniwhere	MVMWM	COMBO
0	**100.00** ± 0.00	95.00 ± 5.05	62.22 ± 24.56	**100.00** ± 0.00	**100.00** ± 0.00
10	**16.66** ± 14.30	0.00 ± 0.00	5.00 ± 7.82	0.00 ± 0.00	0.00 ± 0.00
20	**87.78** ± 11.29	0.00 ± 0.00	30.00 ± 9.43	0.00 ± 0.00	0.00 ± 0.00
30	8.33 ± 2.79	1.67 ± 2.79	0.00 ± 0.00	0.00 ± 0.00	**23.33** ± 2.98
40	0.00 ± 0.00	0.00 ± 0.00	0.00 ± 0.00	0.00 ± 0.00	0.00 ± 0.00
50	0.00 ± 0.00	0.00 ± 0.00	0.00 ± 0.00	0.00 ± 0.00	0.00 ± 0.00
60	0.00 ± 0.00	0.00 ± 0.00	0.00 ± 0.00	0.00 ± 0.00	0.00 ± 0.00
70	0.00 ± 0.00	0.00 ± 0.00	0.00 ± 0.00	0.00 ± 0.00	0.00 ± 0.00
80	0.00 ± 0.00	0.00 ± 0.00	0.00 ± 0.00	0.00 ± 0.00	0.00 ± 0.00
90	0.00 ± 0.00	0.00 ± 0.00	0.00 ± 0.00	0.00 ± 0.00	0.00 ± 0.00
100	28.33 ± 12.07	0.00 ± 0.00	2.78 ± 3.90	**42.23** ± 6.21	0.00 ± 0.00
110	**62.22** ± 3.44	0.00 ± 0.00	32.78 ± 10.20	2.78 ± 1.36	0.00 ± 0.00
120	**63.34** ± 32.73	5.00 ± 1.83	18.34 ± 12.95	41.11 ± 5.02	0.00 ± 0.00
130	**90.00** ± 9.89	40.00 ± 6.67	37.23 ± 30.14	37.22 ± 2.51	0.00 ± 0.00
140	**95.00** ± 4.60	61.67 ± 15.02	66.11 ± 22.74	17.22 ± 4.43	0.00 ± 0.00
150	**93.89** ± 7.72	43.33 ± 6.67	60.56 ± 11.04	42.78 ± 11.04	0.00 ± 0.00
160	**88.33** ± 3.50	18.34 ± 1.82	47.78 ± 23.91	0.00 ± 0.00	21.67 ± 8.37
170	**21.67** ± 17.86	15.00 ± 6.58	15.55 ± 10.26	16.67 ± 2.11	15.00 ± 8.37
-10	**93.89** ± 5.34	36.67 ± 6.67	56.11 ± 25.94	0.00 ± 0.00	53.33 ± 9.89
-20	**75.00** ± 4.60	36.67 ± 6.67	46.67 ± 21.29	0.00 ± 0.00	26.67 ± 9.89
-30	**87.78** ± 9.35	33.33 ± 2.98	45.00 ± 19.52	0.00 ± 0.00	5.00 ± 5.05
-40	**79.44** ± 10.84	56.67 ± 6.32	53.33 ± 17.00	0.00 ± 0.00	0.00 ± 0.00
-50	**77.78** ± 12.94	8.33 ± 5.05	48.89 ± 23.54	0.00 ± 0.00	1.67 ± 1.82
-60	**80.55** ± 24.26	5.00 ± 1.83	45.56 ± 20.08	0.00 ± 0.00	36.67 ± 30.03
-70	**72.22** ± 8.35	0.00 ± 0.00	20.56 ± 19.37	2.78 ± 1.36	0.00 ± 0.00
-80	1.67 ± 2.79	**8.34** ± 1.82	3.33 ± 5.58	0.00 ± 0.00	0.00 ± 0.00
-90	0.00 ± 0.00	0.00 ± 0.00	0.00 ± 0.00	0.00 ± 0.00	0.00 ± 0.00
-100	**6.67** ± 4.71	0.00 ± 0.00	0.00 ± 0.00	0.00 ± 0.00	0.00 ± 0.00
-110	**3.89** ± 1.36	1.67 ± 1.82	0.00 ± 0.00	0.00 ± 0.00	0.00 ± 0.00
-120	**3.33** ± 2.11	0.00 ± 0.00	0.00 ± 0.00	0.00 ± 0.00	0.00 ± 0.00
-130	**11.11** ± 9.59	0.00 ± 0.00	0.00 ± 0.00	0.00 ± 0.00	0.00 ± 0.00
-140	**16.11** ± 12.54	0.00 ± 0.00	0.00 ± 0.00	0.00 ± 0.00	0.00 ± 0.00
-150	**13.89** ± 3.28	5.00 ± 1.83	0.00 ± 0.00	0.00 ± 0.00	0.00 ± 0.00
-160	**90.56** ± 1.36	6.67 ± 3.65	0.00 ± 0.00	0.00 ± 0.00	0.00 ± 0.00
-170	**68.34** ± 26.65	0.00 ± 0.00	16.67 ± 16.60	8.89 ± 1.72	1.67 ± 1.82
-180	**50.56** ± 34.41	40.00 ± 9.89	23.34 ± 3.65	0.00 ± 0.00	3.33 ± 3.65

Table 5: Angle-wise success rates (%) on Meta-World Door Open. Results are reported as mean ± standard deviation over the final two evaluation success rates from each of three training seeds. Offsets are relative to the policy-training viewpoint. Shaded rows indicate viewpoints within the multi-view sampling sectors (ID). Bold marks the highest nonzero mean at each offset.

Offset (°)	RCA	ReViWo	Maniwhere	MVMWM	COMBO
0	**99.45** ± 1.36	88.34 ± 1.82	42.77 ± 19.37	58.89 ± 7.20	91.67 ± 5.05
10	**18.33** ± 12.95	0.00 ± 0.00	4.44 ± 7.79	0.00 ± 0.00	0.00 ± 0.00
20	**33.33** ± 23.57	0.00 ± 0.00	10.00 ± 6.66	0.00 ± 0.00	0.00 ± 0.00
30	**12.22** ± 10.68	0.00 ± 0.00	0.00 ± 0.00	0.00 ± 0.00	0.00 ± 0.00
40	0.00 ± 0.00	0.00 ± 0.00	0.00 ± 0.00	0.00 ± 0.00	0.00 ± 0.00
50	0.00 ± 0.00	0.00 ± 0.00	0.00 ± 0.00	0.00 ± 0.00	0.00 ± 0.00
60	0.00 ± 0.00	0.00 ± 0.00	0.00 ± 0.00	0.00 ± 0.00	**1.67** ± 1.82
70	0.00 ± 0.00	0.00 ± 0.00	0.00 ± 0.00	0.00 ± 0.00	0.00 ± 0.00
80	0.00 ± 0.00	0.00 ± 0.00	0.00 ± 0.00	0.00 ± 0.00	0.00 ± 0.00
90	0.00 ± 0.00	0.00 ± 0.00	0.00 ± 0.00	0.00 ± 0.00	0.00 ± 0.00
100	0.00 ± 0.00	0.00 ± 0.00	0.00 ± 0.00	**8.89** ± 2.72	0.00 ± 0.00
110	**23.33** ± 6.67	0.00 ± 0.00	6.11 ± 4.91	7.22 ± 3.28	0.00 ± 0.00
120	**33.33** ± 6.99	1.67 ± 1.82	9.44 ± 8.28	0.00 ± 0.00	0.00 ± 0.00
130	**73.89** ± 27.11	61.67 ± 3.50	27.22 ± 19.02	15.00 ± 1.83	0.00 ± 0.00
140	**76.67** ± 11.36	68.33 ± 5.05	41.11 ± 8.34	22.78 ± 2.51	0.00 ± 0.00
150	**89.44** ± 4.43	34.44 ± 2.72	26.67 ± 10.95	22.78 ± 2.51	0.00 ± 0.00
160	**92.78** ± 2.51	73.34 ± 8.43	15.00 ± 7.82	10.00 ± 6.67	0.00 ± 0.00
170	**41.11** ± 9.59	31.67 ± 5.05	33.33 ± 23.76	0.00 ± 0.00	0.00 ± 0.00
-10	**73.34** ± 3.65	70.56 ± 12.89	21.11 ± 12.05	21.67 ± 1.82	0.00 ± 0.00
-20	**78.33** ± 9.83	70.56 ± 7.43	35.00 ± 26.06	0.00 ± 0.00	0.00 ± 0.00
-30	83.89 ± 12.00	**89.45** ± 2.51	28.34 ± 21.99	0.00 ± 0.00	0.00 ± 0.00
-40	**92.22** ± 3.44	68.34 ± 18.35	31.11 ± 18.58	0.00 ± 0.00	0.00 ± 0.00
-50	**96.67** ± 5.16	40.00 ± 11.74	23.33 ± 16.05	0.00 ± 0.00	0.00 ± 0.00
-60	**93.89** ± 1.36	0.00 ± 0.00	22.22 ± 20.08	0.00 ± 0.00	0.00 ± 0.00
-70	**95.00** ± 4.60	0.00 ± 0.00	13.34 ± 8.16	0.00 ± 0.00	0.00 ± 0.00
-80	**1.67** ± 1.82	0.00 ± 0.00	0.00 ± 0.00	0.00 ± 0.00	0.00 ± 0.00
-90	0.00 ± 0.00	0.00 ± 0.00	0.00 ± 0.00	0.00 ± 0.00	0.00 ± 0.00
-100	0.00 ± 0.00	0.00 ± 0.00	0.00 ± 0.00	0.00 ± 0.00	0.00 ± 0.00
-110	0.00 ± 0.00	0.00 ± 0.00	0.00 ± 0.00	0.00 ± 0.00	0.00 ± 0.00
-120	0.00 ± 0.00	0.00 ± 0.00	0.00 ± 0.00	0.00 ± 0.00	0.00 ± 0.00
-130	0.00 ± 0.00	0.00 ± 0.00	0.00 ± 0.00	0.00 ± 0.00	0.00 ± 0.00
-140	0.00 ± 0.00	0.00 ± 0.00	0.00 ± 0.00	0.00 ± 0.00	0.00 ± 0.00
-150	**8.33** ± 5.87	0.00 ± 0.00	0.00 ± 0.00	0.00 ± 0.00	0.00 ± 0.00
-160	**68.33** ± 6.91	0.00 ± 0.00	3.33 ± 3.65	2.78 ± 1.36	0.00 ± 0.00
-170	**60.00** ± 11.74	0.00 ± 0.00	16.67 ± 9.43	13.89 ± 3.28	0.00 ± 0.00
-180	**43.89** ± 10.63	33.34 ± 6.67	8.33 ± 14.10	0.00 ± 0.00	0.00 ± 0.00

Table 6: Angle-wise success rates (%) on Meta-World Drawer Open. Results are reported as mean ± standard deviation over the final two evaluation success rates from each of three training seeds. Offsets are relative to the policy-training viewpoint. Shaded rows indicate viewpoints within the multi-view sampling sectors (ID). Bold marks the highest nonzero mean at each offset.

Offset (°)	RCA	ReViWo	Maniwhere	MVMWM	COMBO
0	**100.00** ± 0.00	58.33 ± 5.05	45.56 ± 27.05	87.22 ± 4.43	**100.00** ± 0.00
10	**13.33** ± 9.43	0.00 ± 0.00	12.78 ± 20.38	0.00 ± 0.00	0.00 ± 0.00
20	**12.22** ± 2.72	0.00 ± 0.00	10.56 ± 11.04	0.00 ± 0.00	0.00 ± 0.00
30	0.00 ± 0.00	0.00 ± 0.00	0.00 ± 0.00	0.00 ± 0.00	0.00 ± 0.00
40	0.00 ± 0.00	0.00 ± 0.00	0.00 ± 0.00	0.00 ± 0.00	0.00 ± 0.00
50	0.00 ± 0.00	0.00 ± 0.00	0.00 ± 0.00	0.00 ± 0.00	0.00 ± 0.00
60	0.00 ± 0.00	0.00 ± 0.00	0.00 ± 0.00	0.00 ± 0.00	0.00 ± 0.00
70	0.00 ± 0.00	0.00 ± 0.00	0.00 ± 0.00	0.00 ± 0.00	0.00 ± 0.00
80	0.00 ± 0.00	0.00 ± 0.00	0.00 ± 0.00	0.00 ± 0.00	0.00 ± 0.00
90	0.00 ± 0.00	0.00 ± 0.00	0.00 ± 0.00	0.00 ± 0.00	0.00 ± 0.00
100	0.00 ± 0.00	0.00 ± 0.00	0.00 ± 0.00	0.00 ± 0.00	0.00 ± 0.00
110	**11.67** ± 6.24	0.00 ± 0.00	9.45 ± 15.55	0.00 ± 0.00	0.00 ± 0.00
120	**13.33** ± 9.89	0.00 ± 0.00	8.33 ± 4.08	0.00 ± 0.00	0.00 ± 0.00
130	**25.00** ± 11.69	0.00 ± 0.00	5.00 ± 4.60	0.00 ± 0.00	0.00 ± 0.00
140	**52.22** ± 16.01	38.33 ± 10.05	41.11 ± 39.25	0.00 ± 0.00	0.00 ± 0.00
150	**65.56** ± 11.68	41.67 ± 6.59	37.22 ± 32.28	55.00 ± 7.53	0.00 ± 0.00
160	**62.23** ± 18.22	61.67 ± 1.82	26.67 ± 4.71	55.00 ± 7.53	0.00 ± 0.00
170	41.67 ± 13.94	**45.00** ± 18.35	21.67 ± 24.92	37.23 ± 11.04	0.00 ± 0.00
-10	**66.67** ± 11.74	51.67 ± 5.05	35.00 ± 25.63	0.00 ± 0.00	0.00 ± 0.00
-20	**60.00** ± 17.89	51.67 ± 8.37	51.11 ± 30.23	0.00 ± 0.00	1.67 ± 1.82
-30	**66.67** ± 4.71	60.00 ± 9.89	15.56 ± 13.28	0.00 ± 0.00	1.67 ± 1.82
-40	**58.33** ± 7.23	6.67 ± 6.67	40.56 ± 32.41	0.00 ± 0.00	0.00 ± 0.00
-50	**54.45** ± 21.67	36.67 ± 3.65	38.89 ± 28.65	0.00 ± 0.00	0.00 ± 0.00
-60	**45.00** ± 7.23	0.00 ± 0.00	36.67 ± 20.87	0.00 ± 0.00	0.00 ± 0.00
-70	25.00 ± 21.16	0.00 ± 0.00	**26.11** ± 27.44	0.00 ± 0.00	0.00 ± 0.00
-80	**1.67** ± 2.79	0.00 ± 0.00	0.00 ± 0.00	0.00 ± 0.00	0.00 ± 0.00
-90	0.00 ± 0.00	0.00 ± 0.00	0.00 ± 0.00	0.00 ± 0.00	0.00 ± 0.00
-100	0.00 ± 0.00	0.00 ± 0.00	0.00 ± 0.00	0.00 ± 0.00	0.00 ± 0.00
-110	0.00 ± 0.00	0.00 ± 0.00	0.00 ± 0.00	0.00 ± 0.00	0.00 ± 0.00
-120	0.00 ± 0.00	0.00 ± 0.00	0.00 ± 0.00	0.00 ± 0.00	0.00 ± 0.00
-130	0.00 ± 0.00	0.00 ± 0.00	0.00 ± 0.00	0.00 ± 0.00	0.00 ± 0.00
-140	0.00 ± 0.00	0.00 ± 0.00	0.00 ± 0.00	0.00 ± 0.00	0.00 ± 0.00
-150	0.00 ± 0.00	0.00 ± 0.00	0.00 ± 0.00	0.00 ± 0.00	0.00 ± 0.00
-160	0.00 ± 0.00	0.00 ± 0.00	**5.00** ± 5.48	0.00 ± 0.00	0.00 ± 0.00
-170	**17.78** ± 9.35	0.00 ± 0.00	9.44 ± 7.43	0.00 ± 0.00	0.00 ± 0.00
-180	41.67 ± 20.08	0.00 ± 0.00	15.00 ± 14.72	**52.23** ± 7.20	0.00 ± 0.00

Table 7: Average success rates (%) over the full 360° azimuth range on Meta-World. Each task averages over 36 viewpoints, including the policy-training viewpoint. Average is the mean across the four tasks. The highest value in each column is bold.

Method	Button Press	Window Close	Door Open	Drawer Open	Average
RCA	**63.74**	**44.12**	**38.60**	**23.18**	**42.41**
ReViWo	43.04	14.40	20.33	12.55	22.58
Maniwhere	43.18	20.49	11.64	13.66	22.24
MVMWM	36.39	8.66	5.11	7.96	14.53
COMBO	12.27	8.01	2.59	2.87	6.44

Table 8: Angle-wise success rates (%) on PandaGym Reach over the full 360° azimuth range. Results are reported as mean ± standard deviation over the final two evaluation success rates from each of three training seeds. Angles denote relative azimuth offsets from the policy-training viewpoint. The highest mean at each angle is highlighted in bold.

Method	0°	10°	20°	30°	40°	50°
RCA	**49.44 ± 3.28**	**47.78 ± 4.56**	**43.34 ± 13.33**	**43.34 ± 7.60**	**28.89 ± 2.72**	**26.67 ± 5.58**
ReViWo	34.45 ± 2.72	35.00 ± 3.50	41.66 ± 11.69	41.67 ± 5.05	28.33 ± 5.05	23.34 ± 12.11
Maniwhere	23.33 ± 7.30	20.00 ± 5.97	16.11 ± 5.74	17.22 ± 4.43	16.11 ± 5.74	15.56 ± 8.86
Method	60°	70°	80°	90°	100°	110°
RCA	**26.67 ± 9.43**	**31.67 ± 3.50**	**38.34 ± 5.87**	**27.78 ± 4.56**	**19.45 ± 4.43**	**25.56 ± 4.04**
ReViWo	23.33 ± 2.11	**31.67 ± 15.02**	33.34 ± 3.65	23.34 ± 3.65	11.67 ± 11.69	25.00 ± 15.02
Maniwhere	15.55 ± 5.44	18.89 ± 6.89	16.67 ± 3.65	15.56 ± 1.72	10.55 ± 4.43	14.45 ± 5.84
Method	120°	130°	140°	150°	160°	170°
RCA	**30.00 ± 6.99**	26.67 ± 5.58	**29.44 ± 3.28**	**27.78 ± 7.80**	**31.67 ± 8.37**	**20.00 ± 4.71**
ReViWo	26.67 ± 2.11	**30.00 ± 2.98**	23.33 ± 2.11	21.67 ± 1.82	21.67 ± 1.82	16.67 ± 2.11
Maniwhere	15.00 ± 5.05	13.89 ± 4.91	13.89 ± 6.11	13.89 ± 5.34	16.11 ± 8.01	16.67 ± 7.30
Method	−10°	−20°	−30°	−40°	−50°	−60°
RCA	**45.00 ± 12.43**	22.23 ± 7.20	**33.33 ± 7.60**	**33.33 ± 6.99**	28.33 ± 8.88	**26.67 ± 6.67**
ReViWo	38.33 ± 6.91	**23.33 ± 6.33**	30.00 ± 4.71	27.23 ± 9.76	**28.34 ± 11.88**	**26.67 ± 11.55**
Maniwhere	15.55 ± 8.60	12.78 ± 5.74	15.55 ± 4.55	16.67 ± 6.67	14.45 ± 4.04	13.33 ± 5.96
Method	−70°	−80°	−90°	−100°	−110°	−120°
RCA	25.56 ± 5.02	20.00 ± 6.67	**23.33 ± 5.96**	**18.89 ± 2.72**	**20.00 ± 6.99**	**20.00 ± 6.67**
ReViWo	**26.67 ± 2.98**	**22.78 ± 5.74**	22.78 ± 4.91	16.67 ± 9.89	**20.00 ± 5.58**	3.33 ± 2.98
Maniwhere	15.56 ± 5.02	12.23 ± 5.02	15.00 ± 6.24	16.67 ± 3.65	10.00 ± 5.96	12.22 ± 5.02
Method	−130°	−140°	−150°	−160°	−170°	−180°
RCA	**13.33 ± 9.43**	**16.67 ± 8.69**	**13.33 ± 2.98**	**30.00 ± 5.16**	**24.45 ± 4.04**	**27.78 ± 3.44**
ReViWo	5.00 ± 2.79	11.67 ± 2.79	3.33 ± 2.98	22.78 ± 12.55	20.56 ± 9.05	25.00 ± 5.05
Maniwhere	10.00 ± 6.67	11.11 ± 5.02	3.33 ± 3.65	10.00 ± 5.96	13.33 ± 6.67	13.34 ± 8.94

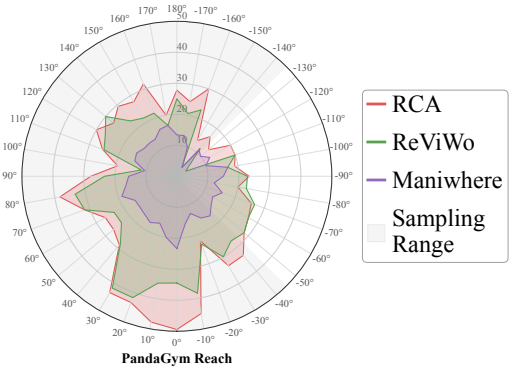


Figure 9: **Success rates over the full 360° azimuth range on PandaGym Reach.** Curves show mean success rates (%) across three seeds at 10° azimuth intervals. Angles denote relative offsets from the policy-training viewpoint. The shaded sector indicates the multi-view sampling range.

Table 9: Success rates (%) within (ID) and outside (OOD) the multi-view sampling sectors on Meta-World. For each task, ID and OOD results are averaged over 18 viewpoints each; ID includes the policy-training viewpoint. Under *Average*, ID and OOD report the corresponding means across the four tasks, while Full 360° reports the overall mean across all four tasks and all 36 viewpoints. The highest value in each column is bold.

	Button Press		Window Close		Door Open		Drawer Open		Average		
Method	ID	OOD	ID	OOD	ID	OOD	ID	OOD	ID	OOD	Full 360°
RCA	**78.46**	**49.01**	**75.59**	**12.65**	**69.38**	**7.81**	**44.23**	**2.13**	**66.91**	**17.90**	**42.41**
ReViWo	61.14	24.94	27.87	0.93	40.65	0.00	25.09	0.00	38.69	6.47	22.58
Maniwhere	60.34	26.02	37.69	3.30	22.19	1.08	25.34	1.98	36.39	8.09	22.24
MVMWM	35.53	37.26	14.66	2.66	9.32	0.90	15.93	0.00	18.86	10.20	14.53
COMBO	18.98	5.56	14.72	1.30	5.09	0.09	5.74	0.00	11.13	1.74	6.44

A.7 Viewpoint Generalization within and beyond Sampling Sectors

In Meta-World, we collect multi-view data from 20 viewpoints sampled within two 90° azimuth sectors, $[-45^\circ, 45^\circ]$ and $[135^\circ, 225^\circ]$. We evaluate at 10° intervals over the full 360° azimuth range relative to the policy-training viewpoint. Except for the policy-training viewpoint, all 36 evaluation viewpoints are absent from the multi-view dataset. We separately report results for the 18 evaluation viewpoints within the sampled sectors (in-distribution, ID) and the 18 outside them (out-of-distribution, OOD). ID evaluation includes the policy-training viewpoint and 17 unseen viewpoints within the sampling sectors. OOD evaluation is particularly challenging because it requires extrapolation beyond the azimuth sectors sampled in the multi-view dataset.

For each task and method, we compute the average success rate over viewpoints in the ID and OOD regions separately. The *Average* columns report averages across the four tasks. Table 9 shows that RCA achieves the highest average success rates in both ID and OOD regions on all four tasks. Across the four tasks, RCA achieves average success rates of 66.91% in ID regions and 17.90% in OOD regions, outperforming the compared baselines in both regions. On Button Press, RCA achieves an OOD average success rate of 49.01%, exceeding the strongest baseline in this region, MVMWM (37.26%), by 11.75 percentage points. These results suggest that RCA improves zero-shot generalization to OOD viewpoints compared with the evaluated baselines.

A.8 Generalization to Camera Pitch, Distance, and Height Changes in Simulation

Beyond azimuth variations and camera shaking, camera viewpoints may also vary in pitch, distance, and height in real-world deployments. To investigate whether RCA's benefits extend beyond 360° azimuth variations and continuous camera shaking, we further evaluate its effectiveness under these additional perturbations.

Evaluation Setup. We evaluate RCA and all baselines on Meta-World Button Press. For each method, we directly use the representation modules trained on multi-view data and the policies trained on fixed-view control data for zero-shot evaluation, without retraining or fine-tuning. The *Average* column reports the mean success rate across all settings in each table.

Camera Perturbations. All offsets are defined relative to the policy-training viewpoint. We evaluate pitch offsets $\Delta p \in \{-20^\circ, -15^\circ, -10^\circ, -5^\circ, -2.5^\circ, 0^\circ, 2.5^\circ, 5^\circ, 10^\circ\}$, camera-distance offsets $\Delta d \in \{-1, -0.5, 0, 0.5, 1\}\,\mathrm{m}$, and camera-height offsets $\Delta h \in \{0, 5, 10, 15, 20\}\,\mathrm{cm}$. Positive and negative offsets indicate increases and decreases in the corresponding camera parameter. Pitch, distance, and height changes are evaluated separately. Figures 10, 11, and 12 illustrate the resulting observations.

Overall Results. Tables 10, 11, and 12 report the results for pitch, distance, and height changes, respectively. RCA achieves the highest average success rate in all three evaluation sets: 83.52% for

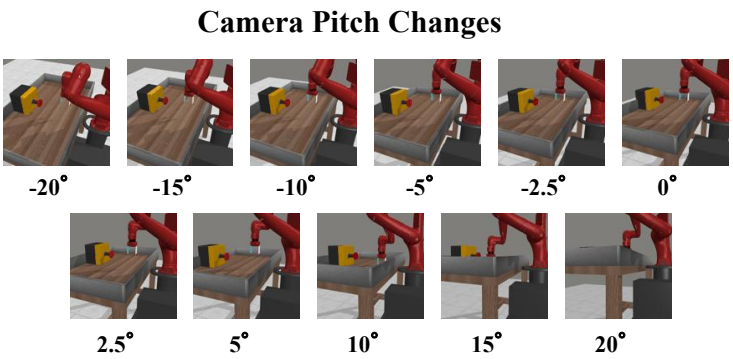


Figure 10: Camera pitch changes on Meta-World Button Press. Angles denote pitch offsets relative to the policy-training viewpoint ($0°$). Negative offsets yield a more top-down perspective, whereas positive offsets lower the viewing angle. At $+15°$ and $+20°$, the table rim severely restricts visibility of the gripper–button interaction region. These two views are shown for illustration only and are excluded from quantitative evaluation.

pitch, 53.00% for distance, and 64.22% for height. The corresponding strongest baseline averages are 60.56% (ReViWo), 46.11% (MVMWM), and 53.33% (MVMWM). RCA also achieves the highest average success rate when the zero-offset reference is excluded: RCA obtains 81.46%, 41.25%, and 55.28%, compared with the strongest baseline averages of 57.50%, 37.92%, and 46.95%, respectively.

Camera Pitch Changes. As shown in Figure 10, positive pitch offsets shift the observation toward a lower viewing angle. At $\Delta p = 15°$ and $20°$, the view approaches or falls below tabletop level, and the raised table rim increasingly occludes the gripper–button interaction region. These configurations substantially limit task observability, making it difficult to assess viewpoint robustness separately from the loss of essential visual cues. We therefore restrict quantitative evaluation on the positive side to $\Delta p \leq 10°$; the $15°$ and $20°$ views are included only to illustrate this visibility limitation. In the opposite direction, $\Delta p = -20°$ produces a more top-down, oblique view of the workspace than the training configuration. Despite this change ($\Delta p = -20°$) in viewing perspective, RCA achieves a mean success rate of 50.56%, the highest among the compared methods at this setting, as reported in Table 10. Across the nine evaluated pitch settings, RCA achieves the highest average success rate of 83.52%, compared with 60.56% for ReViWo, 49.75% for Maniwhere, 54.08% for MVMWM, and 29.63% for COMBO. This corresponds to a 22.96 percentage-point improvement over ReViWo, the strongest baseline.

Camera Distance Changes. Figure 11 illustrates how camera distance affects apparent object size and the extent of the visible workspace. At $\Delta d = -1.0\,\mathrm{m}$, the close-up view contains only portions of the gripper and the button assembly, with much of the robot and surrounding scene outside the frame. This limited spatial context makes the setting particularly challenging, yet RCA achieves the highest average success rate of 35.00%, as shown in Table 11. At $\Delta d = +1.0\,\mathrm{m}$, the camera is farther from the workspace, so the robot and target occupy smaller fractions of the image and fine interaction details become less discernible. At this distance, RCA achieves the highest mean success rate of 43.33% among the compared methods. Across the five evaluated distance settings, RCA achieves the highest average success rate of 53.00%, compared with 33.67% for ReViWo, 27.22% for Maniwhere, 46.11% for MVMWM, and 25.00% for COMBO. This corresponds to a 6.89 percentage-point improvement over MVMWM, the strongest baseline.

Camera Height Changes. Figure 12 illustrates changes in the image positions of the gripper, button, and table boundary at different camera heights. As shown in Table 12, RCA achieves the highest average success rate at all four nonzero height offsets. At the larger offsets of $15\,\mathrm{cm}$ and $20\,\mathrm{cm}$, RCA achieves success rates of 52.78% and 50.00%, respectively. Across the five evaluated height settings, including the zero-offset reference, RCA achieves the highest average success rate of 64.22%, compared with 45.78% for ReViWo, 37.00% for Maniwhere, 53.33% for MVMWM, and

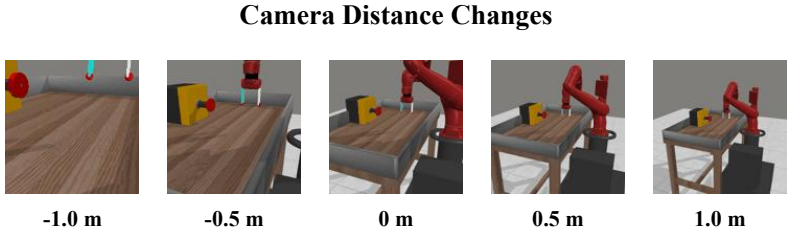


Figure 11: Camera distance changes on Meta-World Button Press. Values denote distance offsets relative to the policy-training configuration, with $0\,\mathrm{m}$ indicating no change. At $-1.0\,\mathrm{m}$, only parts of the gripper and button assembly remain in view. At $+1.0\,\mathrm{m}$, they appear smaller, making fine details harder to see.

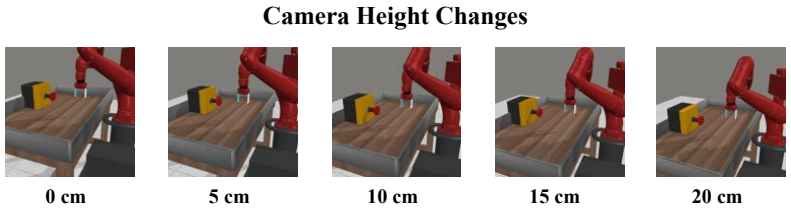


Figure 12: Camera height changes on Meta-World Button Press. Values denote height offsets of 0, 5, 10, 15, and $20\,\mathrm{cm}$ relative to the policy-training configuration. Positive offsets indicate upward camera movement. Other camera parameters remain fixed. Changing the camera height shifts the image positions of the manipulator, target object, and table boundary.

Table 10: Success rates (%) on Meta-World Button-Press under camera *pitch* changes. Offsets are relative to the policy-training viewpoint. For each camera setting, results are reported as mean ± standard deviation over the final two evaluation success rates from each of three training seeds. Average denotes the arithmetic mean of the per-setting means across all nine pitch settings, including the zero-offset reference. The highest mean value in each column is highlighted in bold.

Method	0°	-20°	-15°	-10°	-5°
RCA	$\mathbf{100.00 \pm 0.00}$	$\mathbf{50.56 \pm 9.05}$	$\mathbf{72.22 \pm 18.34}$	$\mathbf{96.67 \pm 5.58}$	$\mathbf{96.67 \pm 5.16}$
ReViWo	85.00 ± 9.61	48.33 ± 8.37	53.33 ± 4.71	55.00 ± 10.05	51.11 ± 11.29
Maniwhere	75.56 ± 5.02	3.33 ± 5.16	5.00 ± 7.82	40.00 ± 15.20	65.00 ± 21.16
MVMWM	78.89 ± 21.25	45.56 ± 7.50	46.67 ± 6.99	53.89 ± 6.47	65.56 ± 6.21
COMBO	$\mathbf{100.00 \pm 0.00}$	21.11 ± 23.91	3.33 ± 5.58	14.44 ± 5.02	27.22 ± 4.91

Method	-2.5°	2.5°	5°	10°	Average
RCA	$\mathbf{97.78 \pm 4.04}$	$\mathbf{96.11 \pm 5.34}$	$\mathbf{95.00 \pm 2.79}$	46.67 ± 6.99	**83.52**
ReViWo	72.78 ± 13.07	63.89 ± 4.43	64.45 ± 7.20	$\mathbf{51.11 \pm 20.30}$	60.56
Maniwhere	75.00 ± 19.30	77.22 ± 5.74	65.56 ± 15.73	41.11 ± 26.72	49.75
MVMWM	72.78 ± 10.20	65.00 ± 7.82	58.33 ± 13.78	0.00 ± 0.00	54.08
COMBO	44.44 ± 12.05	0.00 ± 0.00	33.89 ± 32.83	22.22 ± 18.94	29.63

Table 11: Success rates (%) on Meta-World Button-Press under camera *distance* changes. Offsets are relative to the training viewpoint. For each camera setting, results are reported as mean ± standard deviation over the final two evaluation success rates from each of three training seeds. Average denotes the arithmetic mean of the per-setting means across all five distance settings, including the zero-offset reference. The highest mean value in each column is highlighted in bold.

	Camera distance change (m)					
Method	0	−1	−0.5	0.5	1	Average
RCA	$\mathbf{100.00 \pm 0.00}$	$\mathbf{35.00 \pm 11.69}$	30.55 ± 9.29	$\mathbf{56.11 \pm 6.47}$	$\mathbf{43.33 \pm 14.14}$	**53.00**
ReViWo	85.00 ± 9.61	15.56 ± 11.67	23.33 ± 4.71	44.45 ± 5.44	0.00 ± 0.00	33.67
Maniwhere	75.56 ± 5.02	25.55 ± 16.82	0.00 ± 0.00	5.56 ± 1.72	29.44 ± 18.67	27.22
MVMWM	78.89 ± 21.25	26.67 ± 25.56	$\mathbf{33.89 \pm 10.42}$	50.00 ± 11.16	41.11 ± 14.56	46.11
COMBO	$\mathbf{100.00 \pm 0.00}$	0.00 ± 0.00	0.00 ± 0.00	25.00 ± 21.16	0.00 ± 0.00	25.00

Table 12: Success rates (%) on Meta-World Button-Press under camera *height* changes relative to the training configuration. For each camera setting, results are reported as mean ± standard deviation over the final two evaluation success rates from each of three training seeds. Average is over all five settings, including the zero-offset reference. The highest mean in each column is bold.

	Camera height change (cm)					
Method	0	+5	+10	+15	+20	Average
RCA	$\mathbf{100.00 \pm 0.00}$	$\mathbf{61.66 \pm 14.87}$	$\mathbf{56.67 \pm 6.67}$	$\mathbf{52.78 \pm 10.20}$	$\mathbf{50.00 \pm 4.71}$	**64.22**
ReViWo	85.00 ± 9.61	48.89 ± 4.56	45.56 ± 7.20	49.45 ± 9.29	0.00 ± 0.00	45.78
Maniwhere	75.56 ± 5.02	22.22 ± 18.09	27.78 ± 15.30	24.45 ± 16.01	35.00 ± 29.57	37.00
MVMWM	78.89 ± 21.25	43.89 ± 4.91	50.00 ± 7.60	46.11 ± 7.43	47.78 ± 5.02	53.33
COMBO	$\mathbf{100.00 \pm 0.00}$	13.33 ± 9.89	3.33 ± 2.98	2.78 ± 3.90	0.00 ± 0.00	23.89

23.89% for COMBO. This corresponds to a 10.89 percentage-point improvement over MVMWM, the strongest baseline.

Discussion. Although changes in camera pitch, distance, and height are not seen during training, the visual appearance changes under moderate perturbations may be similar to those induced by some azimuth variations. The feature adaptation and alignment learned across azimuths may also help RCA handle these changes. Therefore, RCA may remain effective under moderate pitch, distance, and height perturbations without retraining or fine-tuning.

A.9 Azimuth Variations with Additional Camera Perturbations

We further evaluate zero-shot viewpoint generalization on Meta-World Button Press by adding camera pitch, height, or distance perturbations to the azimuth variations. We compare RCA, ReViWo, and Maniwhere at azimuth offsets $\Delta\alpha \in \{-10^\circ, -20^\circ, -50^\circ\}$. At each azimuth offset, we evaluate pitch offsets $\Delta p \in \{-5^\circ, -10^\circ, -20^\circ\}$, height offsets $\Delta h \in \{+0.05, +0.1\}\,\mathrm{m}$, or distance offsets $\Delta d \in \{+0.5, +1.0\}\,\mathrm{m}$. Pitch, height, and distance perturbations are evaluated separately at each azimuth offset. Tables 13, 15, and 14 present the corresponding results.

Across the reported settings, RCA achieves average success rates of 55.49%, 52.22%, and 48.06% for pitch, height, and distance perturbations, respectively, compared with 37.59%, 25.93%, and 19.08% for ReViWo and 26.48%, 7.04%, and 16.20% for Maniwhere. These results indicate that RCA maintains higher average success rates than the compared baselines when azimuth variations are combined with camera pitch, height, or distance perturbations.

Table 13: **Camera pitch perturbations at different azimuth offsets on Meta-World Button Press.** Success rates (%) are reported as mean ± sample standard deviation. Average denotes the mean success rate across the listed configurations. Bold marks the highest mean in each row.

Azimuth $\Delta\alpha$	Pitch Δp	RCA	ReViWo	Maniwhere
$-10°$	$-5°$	$\mathbf{60.55 \pm 16.92}$	57.78 ± 10.89	46.11 ± 27.44
	$-10°$	$\mathbf{71.67 \pm 8.63}$	62.78 ± 6.11	38.33 ± 17.73
	$-20°$	$\mathbf{58.33 \pm 10.49}$	45.56 ± 5.84	0.00 ± 0.00
$-20°$	$-5°$	$\mathbf{63.34 \pm 11.15}$	47.78 ± 7.50	23.33 ± 16.19
	$-10°$	$\mathbf{52.78 \pm 14.21}$	27.22 ± 15.97	38.89 ± 26.05
	$-20°$	$\mathbf{27.78 \pm 7.20}$	21.67 ± 6.24	5.00 ± 5.48
$-50°$	$-5°$	$\mathbf{71.11 \pm 19.85}$	32.78 ± 11.43	45.00 ± 26.06
	$-10°$	$\mathbf{60.00 \pm 12.82}$	15.00 ± 11.30	41.11 ± 10.26
	$-20°$	$\mathbf{33.89 \pm 19.14}$	27.78 ± 7.20	0.56 ± 1.36
Average		$\mathbf{55.49}$	37.59	26.48

Table 14: **Camera distance perturbations at different azimuth offsets on Meta-World Button Press.** Success rates (%) are reported as mean ± sample standard deviation. Average denotes the mean success rate across the listed configurations. Bold marks the highest mean in each row.

Azimuth $\Delta\alpha$	Distance Δd (m)	RCA	ReViWo	Maniwhere
$-10°$	$+0.5$	$\mathbf{53.89 \pm 29.09}$	5.00 ± 3.50	11.11 ± 15.44
	$+1.0$	$\mathbf{45.00 \pm 10.70}$	0.00 ± 0.00	20.56 ± 26.70
$-20°$	$+0.5$	$\mathbf{54.45 \pm 5.02}$	43.89 ± 13.73	18.33 ± 19.29
	$+1.0$	$\mathbf{45.56 \pm 10.89}$	7.78 ± 5.84	2.22 ± 2.72
$-50°$	$+0.5$	$\mathbf{46.11 \pm 14.82}$	20.00 ± 8.69	38.33 ± 17.22
	$+1.0$	$\mathbf{43.33 \pm 23.29}$	37.78 ± 10.26	6.67 ± 7.60
Average		$\mathbf{48.06}$	19.08	16.20

Table 15: **Camera height perturbations at different azimuth offsets on Meta-World Button Press.** Success rates (%) are reported as mean ± sample standard deviation. Average denotes the mean success rate across the listed configurations. Bold marks the highest mean in each row.

Azimuth $\Delta\alpha$	Height Δh (m)	RCA	ReViWo	Maniwhere
$-10°$	$+0.05$	$\mathbf{53.33 \pm 12.47}$	48.33 ± 3.50	0.00 ± 0.00
	$+0.10$	$\mathbf{57.22 \pm 9.98}$	47.78 ± 6.89	7.78 ± 11.29
$-20°$	$+0.05$	$\mathbf{29.44 \pm 10.42}$	19.44 ± 15.97	2.22 ± 3.44
	$+0.10$	$\mathbf{51.11 \pm 8.07}$	11.67 ± 4.59	0.56 ± 1.36
$-50°$	$+0.05$	$\mathbf{71.67 \pm 7.23}$	1.11 ± 1.72	15.00 ± 14.72
	$+0.10$	$\mathbf{50.56 \pm 4.43}$	27.22 ± 9.76	16.67 ± 19.21
Average		$\mathbf{52.22}$	25.93	7.04

Table 16: Real-world success counts (out of 10) under camera pitch changes relative to the training configuration. Average reports mean success rate (%) across all three settings. The highest value in each column is highlighted in bold.

	Camera pitch offset			
Method	$0°$	$-5°$	$+5°$	Average (%)
RCA	**10/10**	**9/10**	**7/10**	**86.67**
$\pi_{0.5}$	9/10	8/10	**7/10**	80.00
ReViWo	7/10	5/10	3/10	50.00
Maniwhere	5/10	3/10	2/10	33.33
ACT	6/10	3/10	0/10	30.00

Table 17: Real-world success counts (out of 10) under camera distance changes relative to the training configuration. Average reports mean success rate (%) across all three settings. The highest value in each column is highlighted in bold.

	Camera distance offset (cm)			
Method	0	$+2.5$	$+5$	Average (%)
RCA	**10/10**	**10/10**	**9/10**	**96.67**
$\pi_{0.5}$	9/10	8/10	**9/10**	86.67
ReViWo	7/10	5/10	4/10	53.33
Maniwhere	5/10	3/10	3/10	36.67
ACT	6/10	2/10	0/10	26.67

A.10 Generalization to Camera Pitch, Distance, and Height Changes in the Real World

Evaluation Setup. Beyond the azimuth evaluation in Section 5.3, we further evaluate RCA and the baselines under camera pitch, distance, and height changes on the real-world can-placement task. We reuse the same trained models as in the azimuth evaluation, without retraining or fine-tuning.

Camera Perturbations and Metrics. All offsets are defined relative to the policy-training camera configuration. We evaluate pitch offsets $\Delta p \in \{-5°, 0°, +5°\}$, distance offsets $\Delta d \in \{0, +2.5, +5\}\,\mathrm{cm}$, and height offsets $\Delta h \in \{0, +2.5, +5\}\,\mathrm{cm}$. Positive and negative pitch offsets tilt the camera upward and downward, respectively. Positive distance offsets move the camera farther from the workspace, while positive height offsets raise the camera. Pitch, distance, and height changes are evaluated separately. Each setting is evaluated over 10 trials. Success requires placing the green can inside the basket and releasing it. Tables report successful trials out of 10. For each perturbation type, the *Average* column reports the mean success rate (%) across all settings, including the zero-offset reference.

Camera Pitch Changes. As shown in Table 16, RCA succeeds in 9 out of 10 trials at a pitch offset of $-5°$ and 7 out of 10 trials at $+5°$. RCA achieves the highest average success rate of 86.67% across the three pitch settings, compared with 80.00% for the strongest baseline, $\pi_{0.5}$.

Camera Distance Changes. As shown in Table 17, RCA succeeds in 10 out of 10 trials at a distance offset of $+2.5\,\mathrm{cm}$ and 9 out of 10 trials at $+5\,\mathrm{cm}$. Across the three distance settings, RCA achieves the highest average success rate of 96.67%, compared with 86.67% for the strongest baseline, $\pi_{0.5}$.

Camera Height Changes. As shown in Table 18, RCA succeeds in 10 out of 10 trials at a height offset of $+2.5\,\mathrm{cm}$ and 9 out of 10 trials at $+5\,\mathrm{cm}$. RCA achieves the highest average success rate of 96.67% across the three height settings, followed by $\pi_{0.5}$ at 86.67%.

Table 18: Real-world success counts (out of 10) under camera height changes relative to the training configuration. Average reports mean success rate (%) across all three settings. The highest value in each column is highlighted in bold.

	Camera height offset (cm)			
Method	0	+2.5	+5	Average (%)
RCA	**10/10**	**10/10**	**9/10**	**96.67**
$\pi_{0.5}$	9/10	8/10	**9/10**	86.67
ReViWo	7/10	5/10	4/10	53.33
Maniwhere	5/10	3/10	2/10	33.33
ACT	6/10	6/10	6/10	60.00

A.11 Angle-wise Viewpoint Prediction Errors

Table 19 reports angle-wise viewpoint prediction errors on Meta-World Button Press. We evaluate 36 unique azimuth offsets from -180° to 170° at 10° intervals, covering the full 360° azimuth range.

We compare two training settings. In the limited-sector setting, we train the viewpoint predictor on multi-view data collected from 20 viewpoints sampled within two 90° azimuth sectors, as in the main experiments. For full-360° training, we use multi-view data collected from the same number of viewpoints distributed evenly over the full 360° azimuth range. Gray cells indicate ID viewpoints within the azimuth sectors sampled for limited-sector training.

We compute the overall mean angular error by equally averaging the per-view mean angular errors across all 36 evaluated viewpoints. Under limited-sector training, the overall mean angular error is 9.38°. Only four viewpoints have mean angular errors exceeding 20°, while the remaining 32 viewpoints have an average error of 6.27°.

Figure 13 reports the percentage of evaluated viewpoints whose mean angular errors are within each angular error threshold. Results are shown for ID views, OOD views, and all views under limited-sector training, and for all views under full-360° training. Under limited-sector training, 100% of ID viewpoints (18 of 18) and 77.78% of OOD viewpoints (14 of 18) have mean angular errors within 20°.

Under full-360° training, the overall mean angular error decreases to 5.00°. Only one of the 36 viewpoints has a mean angular error exceeding 20°, while the remaining 35 viewpoints have an average error of 4.57°. These results suggest that broader azimuth coverage in multi-view training data can improve viewpoint prediction accuracy.

A.12 Policy Sensitivity to Viewpoint Prediction Errors

Evaluation Setup. We evaluate RCA's sensitivity to viewpoint-conditioning errors on Meta-World Button Press. We introduce angular errors of 0°, 5°, 10°, and 20° into the ground-truth relative viewpoints used to condition the FiLM-based mapping network. Table 20 reports the corresponding success rates, with ReViWo, Maniwhere, and MVMWM as reference baselines at the same camera azimuth offsets.

Results. As shown in Table 20, RCA's average success rate across the four tested offsets decreases from 94.31% with ground-truth viewpoint conditioning to 81.81%, 78.89%, and 78.20% with angular errors of 5°, 10°, and 20°, respectively. Even at 20° error, RCA achieves a higher average success rate than ReViWo (73.75%), Maniwhere (69.86%), and MVMWM (23.47%). Thus, although viewpoint prediction accuracy affects performance, RCA can tolerate moderate prediction errors.

Relation to Predictor Accuracy. As reported in Appendix A.11, the viewpoint predictor trained on 20 viewpoints sampled within two 90° azimuth sectors achieves an overall mean angular error of 9.38° across 36 evaluated viewpoints on Meta-World Button Press. Only four viewpoints have mean angular errors exceeding 20°, while the remaining 32 have an average error of 6.27°. Together with

Table 19: **Angle-wise viewpoint prediction errors on Meta-World Button Press.** Values are mean angular errors in degrees. Offsets are relative to the policy-training azimuth. Gray cells indicate ID viewpoints within the azimuth sectors sampled for limited-sector training.

Offsets $[-180°, -70°]$			Offsets $[-60°, 50°]$			Offsets $[60°, 170°]$		
Offset	Limited-sector	Full-360°	Offset	Limited-sector	Full-360°	Offset	Limited-sector	Full-360°
$-180°$	1.54	12.34	$-60°$	2.75	2.78	$60°$	1.64	5.12
$-170°$	1.42	6.82	$-50°$	1.37	2.27	$70°$	25.26	4.80
$-160°$	12.39	2.80	$-40°$	4.77	7.77	$80°$	6.34	1.25
$-150°$	16.63	15.38	$-30°$	0.49	1.12	$90°$	21.34	15.29
$-140°$	11.72	3.30	$-20°$	2.21	3.30	$100°$	2.67	0.73
$-130°$	14.66	5.45	$-10°$	4.96	4.03	$110°$	19.22	20.14
$-120°$	9.15	3.22	$0°$	10.85	3.63	$120°$	2.13	2.50
$-110°$	47.43	4.59	$10°$	4.42	4.29	$130°$	2.02	1.56
$-100°$	43.10	0.06	$20°$	3.09	4.74	$140°$	0.29	0.97
$-90°$	15.03	9.33	$30°$	7.53	3.48	$150°$	3.15	3.38
$-80°$	16.66	1.02	$40°$	1.45	4.81	$160°$	0.97	7.42
$-70°$	2.79	4.46	$50°$	7.92	0.58	$170°$	8.37	5.40

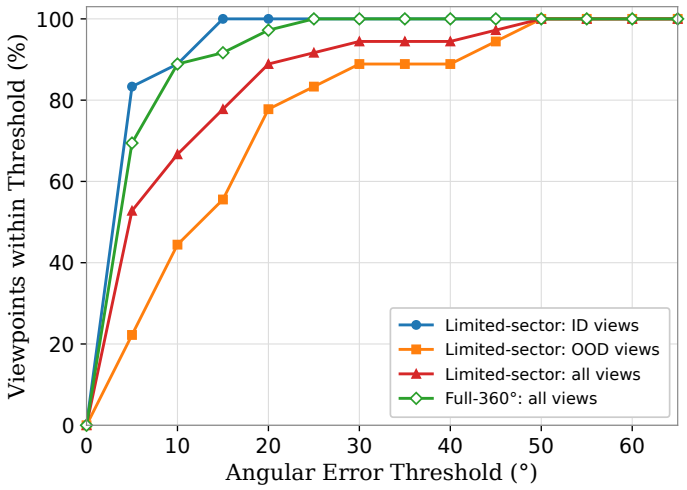


Figure 13: **Viewpoint prediction under limited-sector and full-360° training.** Each curve reports the percentage of evaluated viewpoints whose mean angular errors are within a given threshold. For limited-sector training, results are shown separately for ID views, OOD views, and all views. For full-360° training, results are shown over all views.

Table 20: Success rates (%) on Meta-World Button Press under controlled viewpoint-conditioning errors. Results are reported as mean ± standard deviation over the final two evaluation success rates from each of three training seeds. Camera azimuth offsets are relative to the policy-training viewpoint. *Average* reports the arithmetic mean across the four offsets.

	Camera azimuth offset				
Condition	0°	-10°	-20°	-50°	Average
RCA, 0° error	100.00 ± 0.00	96.11 ± 2.51	93.34 ± 5.58	87.78 ± 11.49	94.31
RCA, 5° error	100.00 ± 0.00	78.33 ± 12.06	78.89 ± 14.55	70.00 ± 12.30	81.81
RCA, 10° error	100.00 ± 0.00	75.56 ± 8.35	76.11 ± 16.39	63.89 ± 16.52	78.89
RCA, 20° error	100.00 ± 0.00	73.33 ± 5.96	77.78 ± 14.56	61.67 ± 17.60	78.20
ReViWo	85.00 ± 9.61	71.67 ± 5.05	78.33 ± 5.05	60.00 ± 3.65	73.75
Maniwhere	75.56 ± 5.02	62.78 ± 18.67	72.22 ± 15.30	68.89 ± 10.68	69.86
MVMWM	78.89 ± 21.25	0.00 ± 0.00	0.00 ± 0.00	15.00 ± 4.59	23.47

Table 21: Success rates (%) on Meta-World Window Close for additional ablations. Results are reported as mean ± standard deviation over the final two evaluation success rates from each of three training seeds. Angles denote azimuth offsets relative to the policy-training viewpoint. Average is over the five offsets, including the training viewpoint. Bold marks the highest mean in each column.

Method	0°	-10°	-20°	$+20^\circ$	$+120^\circ$	Average
RCA	$\mathbf{100.00 \pm 0.00}$	$\mathbf{93.89 \pm 5.34}$	75.00 ± 4.60	$\mathbf{87.78 \pm 11.29}$	$\mathbf{63.34 \pm 32.73}$	**84.00**
w/o Entity-Centric Reconstruction	97.78 ± 3.44	81.67 ± 6.58	$\mathbf{77.23 \pm 21.02}$	51.11 ± 13.93	56.67 ± 10.54	72.89
w/o DINOv3 Features	$\mathbf{100.00 \pm 0.00}$	70.00 ± 15.49	70.56 ± 17.82	50.56 ± 7.12	53.89 ± 12.72	69.00
w/o Inverse Dynamics	$\mathbf{100.00 \pm 0.00}$	89.45 ± 4.43	73.33 ± 16.60	81.67 ± 9.13	42.78 ± 7.43	77.45
w/o Zero-Offset Preservation Loss	$\mathbf{100.00 \pm 0.00}$	92.23 ± 10.89	72.22 ± 28.34	76.67 ± 4.71	58.89 ± 27.14	80.00
w/o Contrastive Loss	96.67 ± 5.16	93.33 ± 5.58	58.89 ± 11.68	1.67 ± 2.79	10.00 ± 4.71	52.11
w/o Mapping Reconstruction Loss	$\mathbf{100.00 \pm 0.00}$	72.23 ± 5.02	61.11 ± 11.68	61.11 ± 6.89	46.67 ± 23.57	68.22
w/o Viewpoint	94.45 ± 9.58	87.78 ± 10.26	68.89 ± 10.89	52.22 ± 12.23	28.89 ± 4.04	66.45
w/o Mapping	96.67 ± 5.16	73.33 ± 11.36	62.78 ± 17.05	49.45 ± 26.28	25.00 ± 14.87	61.45
ReViWo	95.00 ± 5.05	36.67 ± 6.67	36.67 ± 6.67	0.00 ± 0.00	5.00 ± 1.83	34.67
Maniwhere	62.22 ± 24.56	56.11 ± 25.94	46.67 ± 21.29	30.00 ± 9.43	18.34 ± 12.95	42.67
MVMWM	$\mathbf{100.00 \pm 0.00}$	0.00 ± 0.00	0.00 ± 0.00	0.00 ± 0.00	41.11 ± 5.02	28.22

the controlled-error results in Table 20, these observations suggest that RCA can tolerate moderate errors in the viewpoint conditions predicted by the Stage-2 viewpoint predictor.

A.13 Additional Ablation Study

Evaluation Setup. To examine whether the component contributions observed on Button Press extend to another task, we conduct additional ablations on Meta-World Window Close task. We evaluate the eight ablated variants listed in Table 21 and include ReViWo, Maniwhere, and MVMWM as reference baselines. In *w/o Viewpoint*, we retain the mapping network but remove viewpoint conditioning. In *w/o Mapping*, we remove the mapping network and apply the cross-view contrastive loss directly to the task-relevant representations learned in Stage 1. All angles denote azimuth offsets relative to the policy-training viewpoint. We use the same offsets as in the Button Press ablation, except for $+50^\circ$, where all evaluated methods achieve zero success.

Results. Table 21 shows that RCA achieves the highest average success rate of 84.00%, compared with 34.67% for ReViWo, 42.67% for Maniwhere, and 28.22% for MVMWM. Every ablated variant has a lower average success rate than the full model, with the largest reduction occurring when the contrastive loss is removed, from 84.00% to 52.11%. Removing viewpoint conditioning while retaining the mapping network also substantially reduces average success to 66.45%, supporting the role of viewpoint information in feature adaptation. These results further support the contributions of RCA's components on Window Close, consistent with the findings on Button Press.

Table 22: Success rates (%) on Meta-World Button Press with different mapping architectures. Results are reported as mean ± standard deviation over the final two evaluation success rates from each of three training seeds. Camera azimuth offsets are relative to the policy-training viewpoint. *Average* reports the mean across all four offsets, including the training viewpoint. Bold indicates the higher mean in each row.

Camera azimuth offset	MLP	FiLM
$0°$	72.22 ± 6.89	$\mathbf{100.00 \pm 0.00}$
$+10°$	60.00 ± 25.65	$\mathbf{88.33 \pm 8.10}$
$+20°$	51.67 ± 20.63	$\mathbf{86.67 \pm 7.30}$
$+50°$	37.78 ± 21.16	$\mathbf{56.11 \pm 4.43}$
Average	55.42	**82.78**

A.14 Mapping Architecture: FiLM versus MLP

Evaluation Setup. To evaluate the choice of mapping architecture, we compare the FiLM-based mapping network with a viewpoint-conditioned MLP on Meta-World Button Press. The MLP concatenates the predicted viewpoint condition $\hat{\xi}_t$ with the task-relevant representation $z_{t,\mathrm{task}}$ at its input. In contrast, the FiLM-based network uses the same condition to adapt intermediate features through feature-wise scaling and shifting in each FiLM-modulated block. Both variants use the same training data, learning objectives, optimization settings, mapping-network depth, and hidden dimensions. We evaluate camera azimuth offsets of $0°$, $+10°$, $+20°$, and $+50°$ relative to the policy-training viewpoint.

Results and Interpretation. As shown in Table 22, the FiLM-based mapping network achieves higher mean success rates at all four evaluated offsets, increasing the average from 55.42% to 82.78%. With the same viewpoint condition, network depth, hidden dimensions, and training settings, this comparison supports the benefit of adapting intermediate features over concatenating viewpoint information only at the input. Feature-wise scaling and shifting at multiple layers may allow the network to adapt features more effectively to viewpoint changes, facilitating cross-view alignment.

A.15 Cross-View Similarity Analysis

Evaluation Setup. We analyze 316 states drawn from several batches of the Meta-World Button Press training data, each observed from 20 camera viewpoints. We compute cross-view cosine similarities for the task-relevant representations $z^{(i)}_{t,\mathrm{task}}$ before mapping and the mapped representations $\tilde{z}^{(i)}_t$ after mapping, using predicted viewpoint conditions for the FiLM-based mapping network. We consider two view-pair configurations: *Zero-offset vs. others*, comprising the 19 pairs between the policy-training viewpoint and the other viewpoints, and *All view pairs*, comprising all 190 unordered pairs among the 20 viewpoints. For each configuration, we compute mean same-state and different-state similarities using the same view-pair structure. For different-state similarity, each state is compared with the other 315 sampled states. The *similarity margin* is the same-state mean minus the different-state mean.

Results. As shown in Table 23, feature mapping increases same-state cosine similarity from 0.2614 to 0.5790 for the zero-offset viewpoint paired with the other viewpoints, and from 0.2792 to 0.7141 across all view pairs. Different-state similarity also increases, but by a smaller amount, widening the corresponding similarity margins from 0.0362 to 0.1679 and from 0.0363 to 0.2709. On these sampled training states, feature mapping improves same-state cross-view alignment and increases the gap between mean same-state and different-state similarities.

A.16 Task-Relevant Information Retention

To assess whether RCA preserves task-relevant information during feature mapping, we train small predictors to estimate robot–object relative position and manipulation progress from the learned

Table 23: Mean cross-view cosine similarities before and after feature mapping on 316 states from the Meta-World Button Press training data, each observed from 20 viewpoints. Similarity margin is the same-state mean minus the different-state mean.

	Zero-offset vs. others		All view pairs	
Metric	Before	After	Before	After
Same-state similarity	0.2614	0.5790	0.2792	0.7141
Different-state similarity	0.2252	0.4111	0.2429	0.4432
Similarity margin	0.0362	0.1679	0.0363	0.2709

representations. All representation models remain frozen during predictor training and evaluation, so this experiment measures how accurately these task variables can be recovered from their features.

Data Construction. We collect 100 new trajectories for each of Meta-World Button Press and Window Close using scripted experts with action noise. We sample 20 states per trajectory and render each state at 12 azimuth offsets from -180° to 150° in 30° increments relative to the policy-training viewpoint. Labels are obtained from simulator states: relative position is measured from the task-specific object reference point to the gripper center in world coordinates, and manipulation progress is measured by the displacement of the button or window joint from its initial position. For each task, we split the 100 trajectories into 60 training, 20 validation, and 20 test trajectories, keeping all views of each state in the same split. All compared representations use identical images, labels, and data splits.

Prediction Setup. We include ReViWo (Pang et al., 2025) as a representative view-consistent baseline. Stage-1 Encoder denotes the learned latent features z_t, whereas Encoder+DINOv3 denotes the task-relevant representation $z_{t,\mathrm{task}}$ before mapping. RCA denotes the mapped representation $\tilde{z}_t$ obtained using predicted viewpoint conditions. We also evaluate variants without DINOv3 features or viewpoint conditioning. In Direct encoder fine-tuning, we directly fine-tune the Stage-1 visual encoder without a separate FiLM-based mapping network.

For each representation and viewpoint, we train a separate two-layer MLP predictor with hidden width 128. All predictors use the same training settings. Position error is the mean Euclidean distance between predicted and ground-truth 3D relative positions (cm), while progress error is the mean absolute error (mm). Both errors are averaged over test states, viewpoints within each ID/OOD group, and three independently initialized predictors. ID and OOD denote viewpoints within and outside the multi-view sampling sectors used for representation learning, respectively.

Results. Table 24 shows that RCA achieves lower position and progress prediction errors than both ReViWo and direct encoder fine-tuning across ID and OOD viewpoints on both tasks. RCA also maintains prediction errors close to or lower than those of the pre-mapping Encoder+DINOv3 representation.

At ID viewpoints, RCA achieves position errors of 0.85 and $0.98\,\mathrm{cm}$ on Button Press and Window Close, respectively, compared with 1.81 and $2.77\,\mathrm{cm}$ for ReViWo and 5.54 and $10.11\,\mathrm{cm}$ for direct encoder fine-tuning. RCA achieves progress errors of 2.03 and $4.59\,\mathrm{mm}$ on the two tasks, respectively, compared with 3.54 and $14.06\,\mathrm{mm}$ for ReViWo and 13.47 and $53.98\,\mathrm{mm}$ for direct encoder fine-tuning.

At OOD viewpoints, RCA achieves position errors of 1.10 and $1.20\,\mathrm{cm}$ on Button Press and Window Close, respectively, compared with 2.34 and $3.10\,\mathrm{cm}$ for ReViWo and 6.13 and $10.32\,\mathrm{cm}$ for direct encoder fine-tuning. RCA achieves progress errors of 2.86 and $4.68\,\mathrm{mm}$ on the two tasks, respectively, compared with 5.48 and $13.69\,\mathrm{mm}$ for ReViWo and 15.46 and $50.57\,\mathrm{mm}$ for direct encoder fine-tuning.

Compared with the pre-mapping Encoder+DINOv3 representation, RCA achieves lower position prediction errors in both ID and OOD regions on both tasks. Progress prediction errors are also lower, except on Button Press at OOD viewpoints, where the error increases from 2.70 to $2.86\,\mathrm{mm}$. These

Table 24: Task-state prediction errors on Meta-World. A separate two-layer MLP predictor is trained for each representation and viewpoint, with representation models frozen. Results are averaged over three predictor initializations; lower is better. ID/OOD denotes viewpoints within/outside the multi-view sampling sectors used for representation learning. Encoder+DINOv3 provides the pre-mapping reference for assessing task-relevant information retention. Bold marks the lowest error per column.

	Position error (cm)		Progress MAE (mm)	
Representation	ID	OOD	ID	OOD
Button Press				
ReViWo	1.81	2.34	3.54	5.48
Stage-1 Encoder	0.94	1.19	2.33	2.75
Direct encoder fine-tuning	5.54	6.13	13.47	15.46
RCA w/o DINOv3 Features	0.93	1.25	2.33	2.95
Mapping w/o View	0.89	1.22	2.29	2.84
Encoder+DINOv3	0.94	1.16	2.20	**2.70**
RCA	**0.85**	**1.10**	**2.03**	2.86
Window Close				
ReViWo	2.77	3.10	14.06	13.69
Stage-1 Encoder	1.11	1.41	5.12	5.15
Direct encoder fine-tuning	10.11	10.32	53.98	50.57
RCA w/o DINOv3 Features	1.13	1.44	4.98	5.73
Mapping w/o View	1.08	1.40	4.88	5.40
Encoder+DINOv3	1.11	1.31	5.09	4.78
RCA	**0.98**	**1.20**	**4.59**	**4.68**

results support the retention of the measured task-relevant information during feature mapping. They also show that these task variables can be recovered more accurately from RCA representations than from ReViWo or direct encoder fine-tuning.

A.17 Contribution of External Pretrained Models

Evaluation Setup. We examine whether RCA's performance advantage can be explained solely by its use of external pretrained models through two complementary comparisons on Meta-World Button Press and Window Close. First, we augment ReViWo and Maniwhere with the same frozen DINOv3 features used by RCA, keeping the representation-learning data, downstream control data, and evaluation protocol unchanged. Second, we remove DINOv3 feature augmentation, the entity-centric masked reconstruction objective supervised by Grounded SAM-generated masks, or both. The joint ablation is denoted as *w/o Both*. All angles denote azimuth offsets relative to the policy-training viewpoint. We include ReViWo, Maniwhere, and MVMWM as reference baselines for the feature and mask-supervision ablations.

DINOv3-Augmented Baselines. Table 25 reports results at six camera offsets, including the policy-training viewpoint. Adding DINOv3 features improves ReViWo's average success rate from 60.93% to 65.65% on Button Press and from 28.89% to 33.52% on Window Close. For Maniwhere, the corresponding averages increase from 59.08% to 63.98% and from 35.56% to 36.85%, respectively. RCA nevertheless achieves higher averages of 84.35% and 70.00% on the two tasks. Thus, providing both baselines with the same pretrained visual features does not close the performance gap, suggesting that access to DINOv3 features alone does not explain RCA's advantage.

Feature and Mask-Supervision Ablations. Table 26 reports results at five camera offsets, including the policy-training viewpoint, on Button Press and Window Close. Removing entity-centric reconstruction yields average success rates of 83.45% and 76.56%, respectively, while removing DINOv3 features yields 82.89% and 73.11%. Removing both yields 75.89% and 72.33%. Full RCA

Table 25: Success rates (%) with DINOv3-augmented ReViWo and Maniwhere. Results are reported as mean ± standard deviation over the final two evaluation success rates from each of three training seeds. Average is over all six offsets. Bold marks the highest nonzero mean within each task and column.

Method	0°	−10°	−20°	+20°	+50°	+120°	Average
Button Press							
ReViWo	85.00 ± 9.61	71.67 ± 5.05	78.33 ± 5.05	36.67 ± 3.65	29.44 ± 17.31	64.44 ± 14.24	60.93
ReViWo + DINOv3	90.00 ± 4.71	85.00 ± 11.69	80.00 ± 4.71	41.67 ± 16.43	30.56 ± 12.19	66.67 ± 13.34	65.65
Maniwhere	75.56 ± 5.02	62.78 ± 18.67	72.22 ± 15.30	46.67 ± 20.22	39.44 ± 11.44	57.78 ± 21.88	59.08
Maniwhere + DINOv3	76.11 ± 16.52	67.78 ± 18.22	74.44 ± 14.09	59.44 ± 15.55	48.89 ± 8.08	57.22 ± 19.25	63.98
RCA	$\mathbf{100.00 \pm 0.00}$	$\mathbf{96.67 \pm 4.22}$	$\mathbf{96.67 \pm 4.22}$	$\mathbf{86.67 \pm 7.30}$	$\mathbf{56.11 \pm 4.43}$	$\mathbf{70.00 \pm 19.44}$	**84.35**
Window Close							
ReViWo	95.00 ± 5.05	36.67 ± 6.67	36.67 ± 6.67	0.00 ± 0.00	0.00 ± 0.00	5.00 ± 1.83	28.89
ReViWo + DINOv3	95.00 ± 5.05	43.33 ± 14.14	45.56 ± 6.89	1.67 ± 1.82	0.00 ± 0.00	15.56 ± 7.80	33.52
Maniwhere	62.22 ± 24.56	56.11 ± 25.94	46.67 ± 21.29	30.00 ± 9.43	0.00 ± 0.00	18.34 ± 12.95	35.56
Maniwhere + DINOv3	71.66 ± 25.62	51.11 ± 23.06	50.00 ± 30.40	30.56 ± 13.57	0.00 ± 0.00	17.78 ± 17.97	36.85
RCA	$\mathbf{100.00 \pm 0.00}$	$\mathbf{93.89 \pm 5.34}$	$\mathbf{75.00 \pm 4.60}$	$\mathbf{87.78 \pm 11.29}$	0.00 ± 0.00	$\mathbf{63.34 \pm 32.73}$	**70.00**

Table 26: Success rates (%) after removing DINOv3 features, entity-centric reconstruction, or both. "w/o Both" removes both DINOv3 feature augmentation and the entity-centric reconstruction objective. Results are reported as mean ± standard deviation over the final two evaluation success rates from each of three training seeds. Average is over all five offsets, including the training viewpoint. Bold marks the highest mean within each task and column.

Method	0°	−10°	−20°	+20°	−50°	Average
Button Press						
RCA	$\mathbf{100.00 \pm 0.00}$	96.67 ± 4.22	$\mathbf{96.67 \pm 4.22}$	$\mathbf{86.67 \pm 7.30}$	$\mathbf{87.78 \pm 10.68}$	**93.56**
w/o Entity-Centric Reconstruction	$\mathbf{100.00 \pm 0.00}$	90.00 ± 6.66	91.67 ± 2.79	60.56 ± 8.28	75.00 ± 12.78	83.45
w/o DINOv3 Features	98.89 ± 1.72	$\mathbf{97.22 \pm 2.51}$	86.11 ± 10.41	50.00 ± 11.55	82.22 ± 10.68	82.89
w/o Both	$\mathbf{100.00 \pm 0.00}$	90.00 ± 6.66	68.89 ± 13.45	51.11 ± 7.51	69.45 ± 7.72	75.89
ReViWo	85.00 ± 9.61	71.67 ± 5.05	78.33 ± 5.05	36.67 ± 3.65	60.00 ± 3.65	66.33
Maniwhere	75.56 ± 5.02	62.78 ± 18.67	72.22 ± 15.30	46.67 ± 20.22	68.89 ± 10.68	65.22
MVMWM	78.89 ± 21.25	0.00 ± 0.00	0.00 ± 0.00	52.22 ± 8.35	15.00 ± 4.59	29.22
Window Close						
RCA	$\mathbf{100.00 \pm 0.00}$	$\mathbf{93.89 \pm 5.34}$	75.00 ± 4.60	$\mathbf{87.78 \pm 11.29}$	$\mathbf{77.78 \pm 12.94}$	**86.89**
w/o Entity-Centric Reconstruction	97.78 ± 3.44	81.67 ± 6.58	$\mathbf{77.23 \pm 21.02}$	51.11 ± 13.93	75.00 ± 11.88	76.56
w/o DINOv3 Features	$\mathbf{100.00 \pm 0.00}$	70.00 ± 15.49	70.56 ± 17.82	50.56 ± 7.12	74.45 ± 16.82	73.11
w/o Both	$\mathbf{100.00 \pm 0.00}$	76.11 ± 10.20	68.33 ± 15.31	45.56 ± 9.11	71.67 ± 15.88	72.33
ReViWo	95.00 ± 5.05	36.67 ± 6.67	36.67 ± 6.67	0.00 ± 0.00	8.33 ± 5.05	35.33
Maniwhere	62.22 ± 24.56	56.11 ± 25.94	46.67 ± 21.29	30.00 ± 9.43	48.89 ± 23.54	48.78
MVMWM	$\mathbf{100.00 \pm 0.00}$	0.00 ± 0.00	0.00 ± 0.00	0.00 ± 0.00	0.00 ± 0.00	20.00

achieves the highest average on both tasks, at 93.56% and 86.89%. All three ablated variants retain higher average success rates than ReViWo, Maniwhere, and MVMWM on both tasks. These results suggest that RCA's advantage over the reference baselines is not solely attributable to DINOv3 feature augmentation and mask supervision from external pretrained models.

A.18 Hyperparameters

To ensure full reproducibility of our proposed framework, we provide the key hyperparameters used across all stages of training in Table 27.

A.19 Compute Resources and Software Environment

All experiments were conducted on a local workstation equipped with three NVIDIA RTX A5000 GPUs (24GB VRAM each). The machine is powered by dual Hygon C86 7151 16-core processors, providing 64 logical CPU threads in total under the x86_64 architecture. The software environment is built upon Python 3.10.14, PyTorch 2.5.1, and CUDA 12.1. Training times were approximately 45

Table 27: Key hyperparameters for training RCA. Policy settings cover COMBO in simulation and ACT in real-world experiments. World-model settings apply only to COMBO.

Hyperparameter	**Value**
Image size	128×128
Patch size	16×16
Transformer layers	8
Embedding dimension	512
Optimizer for Stage-1 encoder	Adam
Learning rate of Stage-1 encoder	3×10^{-4}
Batch size of Stage-1 encoder	8
Training updates of Stage-1 encoder	100,000
Coef. for inverse dynamics loss (λ_{idm})	1.0
Coef. for entity-centric masked reconstruction loss (λ_{mask})	1.0
Coef. for background reconstruction loss (λ_{bg})	0.5
Optimizer for FiLM mapping network	Adam
Learning rate of FiLM mapping network	10^{-4}
Batch size of FiLM mapping network	64
Training updates of FiLM mapping network	10,000
FiLM mapping network structure	[input dim, 512, 512, output dim]
Number of FiLM-modulated blocks	2
Coef. for zero-offset preservation loss (λ_{zero})	1.0
Coef. for cross-view contrastive loss (λ_{cvc})	0.05
Contrastive temperature (τ)	0.1
Coef. for mapping reconstruction loss (λ_{rec})	0.1
Viewpoint predictor structure	LayerNorm(384) + Linear(384, 2)
Optimizer for viewpoint predictor	Adam
Initial learning rate of viewpoint predictor	3×10^{-4}
Final learning rate of viewpoint predictor	3×10^{-5}
Learning-rate schedule of viewpoint predictor	Exponential decay
Training updates of viewpoint predictor	100,000
Batch size of viewpoint predictor	8
Views per state in predictor mini-batch	8 (simulation), 6 (real world)
Coef. for offset regression loss	1.0
Coef. for relative angular consistency loss (λ_{rel})	0.1
Coef. for predictor linear-weight L_2 penalty	0.001
Optimizer for policy	Adam (COMBO); AdamW (ACT)
Learning rate of policy	10^{-4}
Policy architecture	MLP (COMBO); Transformer-based CVAE (ACT)
COMBO hidden dimensions	[2048, 512, 256, 256]
ACT encoder / decoder layers	4 / 7
ACT hidden / FFN dimensions	256 / 2048
Batch size of policy	256 (COMBO); 64 (ACT)
Action chunk length (H)	16 (ACT)
Optimizer for world model	Adam
Learning rate of world model	10^{-3}
World model structure	[input dim, 200, 200, 200, 200, output dim]
World model ensemble size	7

hours for task-relevant representation learning, 8 hours for mapping network training, 6 hours for viewpoint predictor training, and 3 hours for testing per viewpoint.

B Algorithm Description

Algorithm 1 summarizes the training procedure of RCA. The two representation-learning stages are shared across simulation and real-world experiments. After mapping the single-view control observations into the aligned latent space, downstream policy learning uses COMBO with a learned world model in simulation, or ACT trained directly on demonstration action sequences in the real world. Here, $\mathbf{a}_t = (a_t, \ldots, a_{t+H-1})$ denotes an ACT target action chunk of length H, with $H = 16$ in our real-world experiments. We train ACT for 300 epochs. In simulation, we collect paired multi-view observations by rendering the same environment state from multiple camera viewpoints. In the real world, we reduce hardware costs by repositioning a single camera and replaying the same recorded action sequence at each camera position, then temporally aligning the recordings to construct paired multi-view observations.

C Additional Visualizations

C.1 Azimuth Variation Visualization

Simulation. Figure 14 visualizes sample viewpoints used in our 360° azimuth evaluation on Meta-World and PandaGym. We evaluate the policy at 10° azimuth intervals over the full 360° azimuth range relative to the policy-training viewpoint. Except for the training viewpoint, all downstream evaluation viewpoints are absent from both the multi-view representation dataset and the single-view policy training dataset, forming a zero-shot viewpoint generalization setting. Moreover, many evaluation viewpoints lie outside the sampled azimuth sectors and are out-of-distribution (OOD). As illustrated in Figure 14, the button in Meta-World Button Press is severely occluded at the illustrated azimuth offsets between -150° and -60°, becoming nearly invisible between -150° and -90°. The robot arm exhibits substantial self-occlusion between $+30^\circ$ and $+90^\circ$, with the gripper fully occluded at $+60^\circ$. These occlusions may contribute to performance degradation at these viewpoints.

Real World. Figure 15 shows observations from the 13 camera viewpoints evaluated in Table 2. The azimuth offsets range from -60° to $+60^\circ$ at 10° intervals relative to the 100° policy-training viewpoint. These observations illustrate changes in the apparent spatial layout of the robot, can, and basket across camera viewpoints. At positive offsets, the camera is farther from these entities than at negative offsets, so they occupy a smaller fraction of the image. This may explain why all methods achieve lower average success rates at positive than at negative azimuth offsets (Table 2).

C.2 Additional Attention Map Visualization

In addition to the Meta-World attention maps shown in the main text, we provide visualizations on PandaGym Reach in Figure 16. The first row shows observations from different camera viewpoints. The second row visualizes the attention maps of ReViWo, while the third row shows the attention maps of the mapped RCA representations. Compared with ReViWo, RCA attends more broadly to task-relevant regions, including the robot arm and end-effector across viewpoints. These results further support that RCA preserves richer task-relevant information after viewpoint-conditioned feature mapping under viewpoint variations.

C.3 Entity-Centric Mask Generation Pipeline

Figure 17 illustrates the entity-centric masks for masked reconstruction. We use Grounded SAM to generate fine-grained binary masks with text prompts: “robotic arm” for the manipulator and task-specific object names such as “button”, “drawer”, “door”, and “window” for the target objects. The union of these masks forms the entity-centric mask. These masks highlight the manipulator and target objects in Meta-World and PandaGym tasks across viewpoints. These masks allow the reconstruction loss to emphasize task-relevant entity regions over background pixels, encouraging the learned representations to preserve static visual information about the manipulator and target objects.

Algorithm 1 Integrated Training Pipeline of RCA

Require: Multi-view dataset $\mathcal{D}_{\text{mv}}$ with viewpoint annotations and actions; for simulation, an additional reward-labeled single-view control dataset $\mathcal{D}_{\text{pol}}$ from the policy-training viewpoint
Ensure: Trained control policy π_Φ

Initialize encoder E_ϕ, viewpoint predictor Q_ν, FiLM mapping network M_θ, and auxiliary training modules; keep DINOv3 frozen
// Stage 1: Task-Relevant Representation Learning
for each Stage-1 training iteration **do**
    Sample multi-view observation transitions and actions from $\mathcal{D}_{\text{mv}}$
    Extract $z_t^{(i)} \leftarrow E_\phi(o_t^{(i)})$ and $z_{t+1}^{(i)} \leftarrow E_\phi(o_{t+1}^{(i)})$
    Update E_ϕ and auxiliary modules using $\mathcal{L}_{\text{task}}$
end for
Freeze E_ϕ
// Stage 2: Viewpoint-Conditioned Feature Mapping
for each viewpoint predictor training iteration **do**
    Sample $\mathcal{B}_{\text{mv}}$ with paired views of the same states and viewpoint labels from $\mathcal{D}_{\text{mv}}$
    Predict $\hat{\xi}^{(i)} \leftarrow Q_\nu(\text{DINO}(o_t^{(i)}))$
    Update Q_ν using $\mathcal{L}_{\text{vp}}$
end for
Freeze Q_ν
for each FiLM mapping training iteration **do**
    Sample a mini-batch of paired multi-view observations from $\mathcal{D}_{\text{mv}}$
    Build $z_{t,\text{task}}^{(i)} \leftarrow [E_\phi(o_t^{(i)}); \text{DINO}(o_t^{(i)})]$
    Obtain viewpoint conditions $\hat{\xi}^{(i)}$ from Q_ν
    Map $\tilde{z}_t^{(i)} \leftarrow M_\theta(z_{t,\text{task}}^{(i)}, \hat{\xi}^{(i)})$
    Update M_θ using $\mathcal{L}_{\text{map}}$
end for
// Downstream Policy Learning
Freeze M_θ
if the experiment is in simulation **then**
// Simulation: Model-Based Offline RL
    Obtain $z_{t,\text{task}}$ and $\hat{\xi}_t$ for each observation in $\mathcal{D}_{\text{pol}}$
    Map each observation to $\tilde{z}_t = M_\theta(z_{t,\text{task}}, \hat{\xi}_t)$, retaining the remaining trajectory data
    Initialize latent world model f_ω, policy π_Φ, and value networks
    Train f_ω on aligned recorded transitions $(\tilde{z}_t, a_t, r_t, \tilde{z}_{t+1})$
    for each policy training iteration **do**
        Generate synthetic rollouts using f_ω and π_Φ
        Update π_Φ and value networks with COMBO using recorded and synthetic transitions
    end for
else
// Real World: Imitation Learning
    Use the 100° recordings in $\mathcal{D}_{\text{mv}}$ for policy learning
    Obtain $z_{t,\text{task}}$ and $\hat{\xi}_t$ for each observation in these recordings
    Map each observation to $\tilde{z}_t = M_\theta(z_{t,\text{task}}, \hat{\xi}_t)$, retaining the corresponding action sequences
    Initialize an ACT policy π_Φ from scratch
    for each policy training iteration **do**
        Sample aligned representations $\tilde{z}_t$ and corresponding target action chunks $\mathbf{a}_t$ from the mapped 100° demonstrations
        Update π_Φ with the ACT objective using $\tilde{z}_t$ as input and $\mathbf{a}_t$ as the target
    end for
end if
return trained policy π_Φ

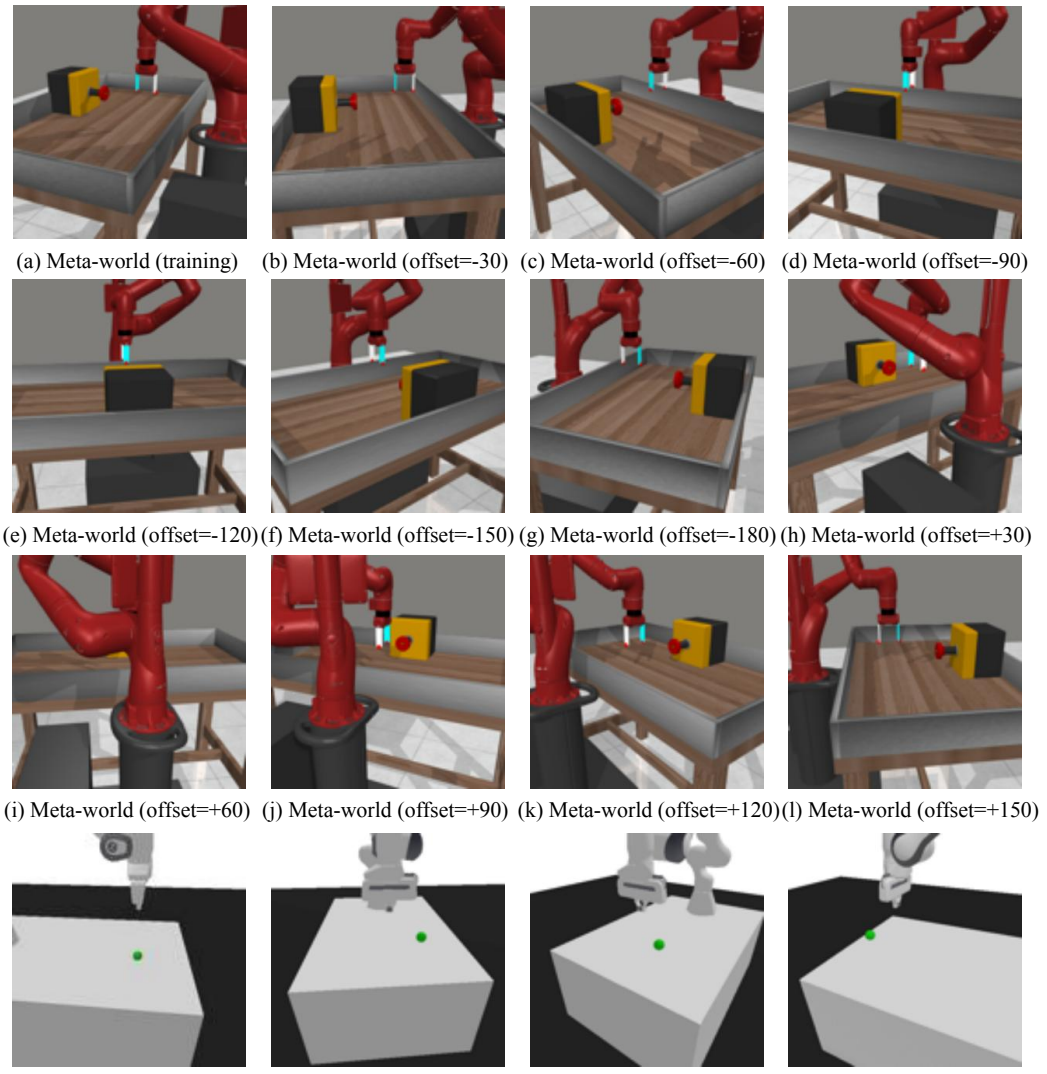


Figure 14: **Visualization of** 360° **azimuth variations in Meta-World and PandaGym.** The first three rows show representative Meta-World viewpoints, where (a) is the policy-training viewpoint and (b–l) are evaluation viewpoints with different azimuth offsets. The last row shows PandaGym viewpoints, including the policy-training viewpoint (I) and representative evaluation viewpoints (II–IV). These examples illustrate the substantial visual shifts and self-occlusions encountered during zero-shot viewpoint generalization over the full 360° azimuth range.

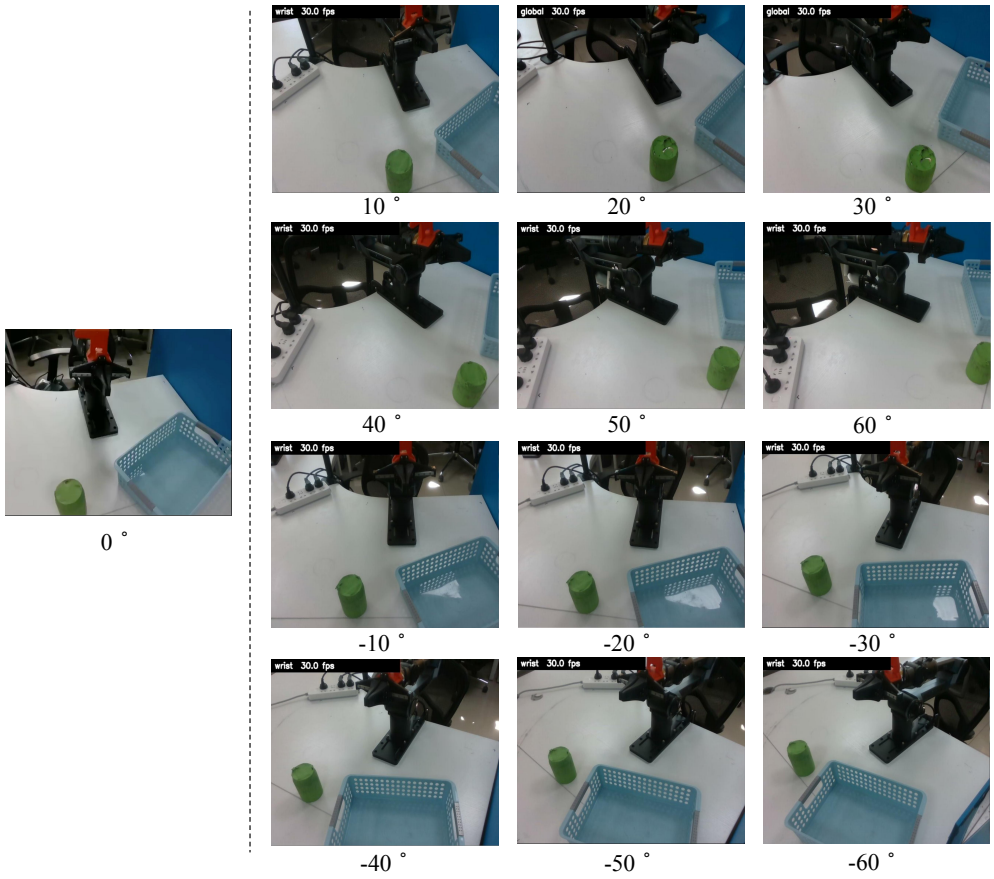


Figure 15: **Real-world observations under camera azimuth changes.** Angles indicate azimuth offsets from the policy-training viewpoint at 100°, which corresponds to a 0° offset.

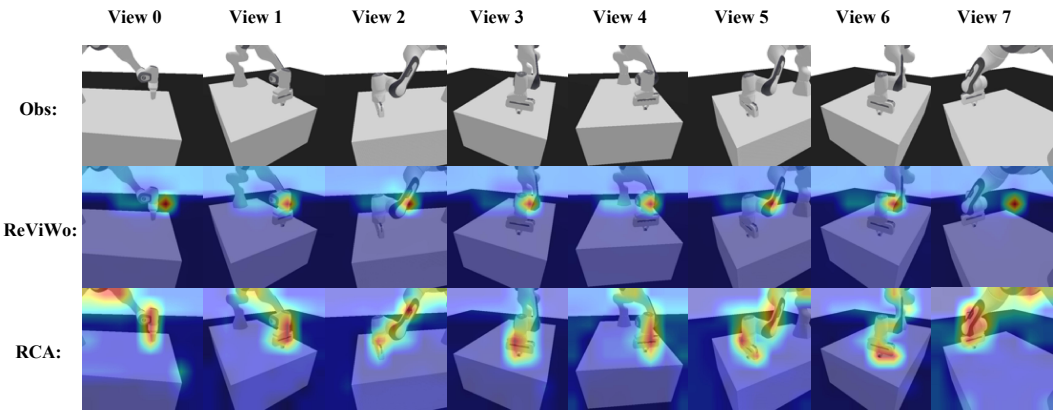


Figure 16: **Additional attention map visualization on PandaGym.** Top row: observations from different viewpoints. Middle row: attention maps of ReViWo. Bottom row: attention maps of the mapped RCA representations. Compared with ReViWo, RCA attends more consistently to the robot end-effector and task-relevant object regions across viewpoints, indicating better preservation of task-relevant information after viewpoint-conditioned feature mapping.

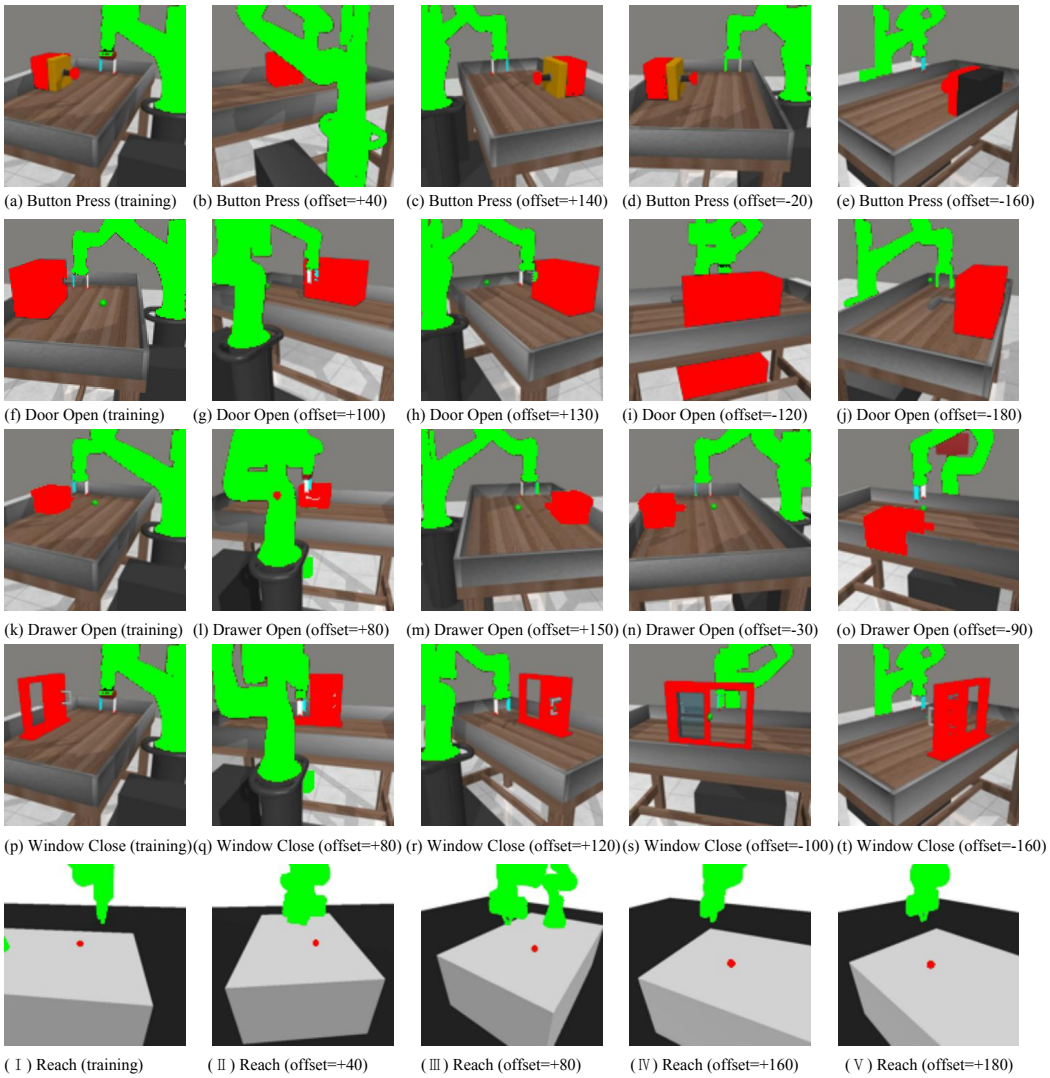


Figure 17: **Visualization of entity-centric masks for masked reconstruction.** This figure illustrates the entity-centric masks generated by Grounded SAM and used to weight the reconstruction loss in Stage 1. The **robot manipulator is highlighted in green**, while the **target objects are highlighted in red**. Rows 1–4 show the Meta-World tasks Button Press, Door Open, Drawer Open, and Window Close, respectively, and the final row shows the PandaGym Reach task. The examples cover the policy-training viewpoint and evaluation viewpoints with relative azimuth offsets ranging from $-180°$ to $+180°$. These masks emphasize manipulator and target-object regions over background pixels during reconstruction, encouraging the learned representations to preserve static visual information about these task entities.

D Limitations and Future Work

RCA currently learns and aligns representations from individual observations, without explicitly incorporating observation histories. Future work could introduce temporal information to retain task-relevant visual cues from earlier observations when these cues become temporarily occluded. Such temporal context could provide more complete information for cross-view alignment and further improve viewpoint robustness under partial occlusion.