CHANGE2ACT: OBJECT-CENTRIC VISUAL REASONING FOR IMAGE-GOAL MANIPULATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Robot manipulation is a fundamental capability for embodied intelligence in real-world environments. Recent generalist robot policies typically specify task goals through natural language and generate continuous actions from visual observations. However, specifying tasks in complex scenes requires precise instructions about objects and their spatial relationships, and writing such instructions demands substantial human effort. To this end, we introduce Image-Goal Manipulation (IGM), where a goal image directly specifies the desired environment state and the policy generates continuous manipulation actions from the current visual observation to realize that state. IGM poses two key challenges. First, the policy must identify task-relevant objects and their desired state transitions despite substantial visual differences between the current observation and goal image. Second, these transitions must be converted into effective action conditions that guide object interaction and continuous control. To address these challenges, we propose Change2Act, a framework that connects explicit visual change reasoning with lightweight policy adaptation. Specifically, we design an Object-Centric Change Reasoning module that establishes semantic correspondences between objects in the initial and goal images and encodes their spatial differences into a structured change descriptor. We further design a Change-Guided Action Conditioning module that maps this descriptor into compact goal tokens and combines them with the original goal image to guide a pretrained robot policy. A goal-role embedding distinguishes the desired state from current observations. Experiments demonstrate that Change2Act achieves more accurate offline action predictions than the compared baselines while requiring fewer trainable parameters.

1 INTRODUCTION

Robot manipulation is a fundamental capability of embodied intelligence, enabling agents to move beyond passive environment understanding and actively modify the physical world Bagnell et al. (2012); Kroemer et al. (2021); Ai et al. (2025). Reliable manipulation requires not only recognizing objects and their semantics, but also understanding spatial relationships, anticipating interaction outcomes, and generating precise continuous control actions Brohan et al. (2023). With advances in large-scale robot datasets, vision-language pretraining, and generative policy learning, robot policies have made substantial progress in complex scene understanding, cross-scene knowledge transfer, and action generation Vuong et al. (2023); Chi et al. (2025); Zhou et al. (2026), providing an important foundation for building robotic systems that can accommodate diverse manipulation requirements.

Prior studies have explored various task interfaces and spatial reasoning mechanisms for representing and executing manipulation objectives. Transporter Networks model pick-and-place operations through spatial correlations between visual features Zeng et al. (2021). CLIPort combines semantic representations from vision-language models with spatial manipulation modules to localize and manipulate target objects according to language instructions Shridhar et al. (2022). VIMA further specifies manipulation tasks through multimodal prompts composed of text, images, and object-level visual information Jiang et al. (2023). These studies demonstrate the importance of both semantic and spatial visual information for robot manipulation.

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.

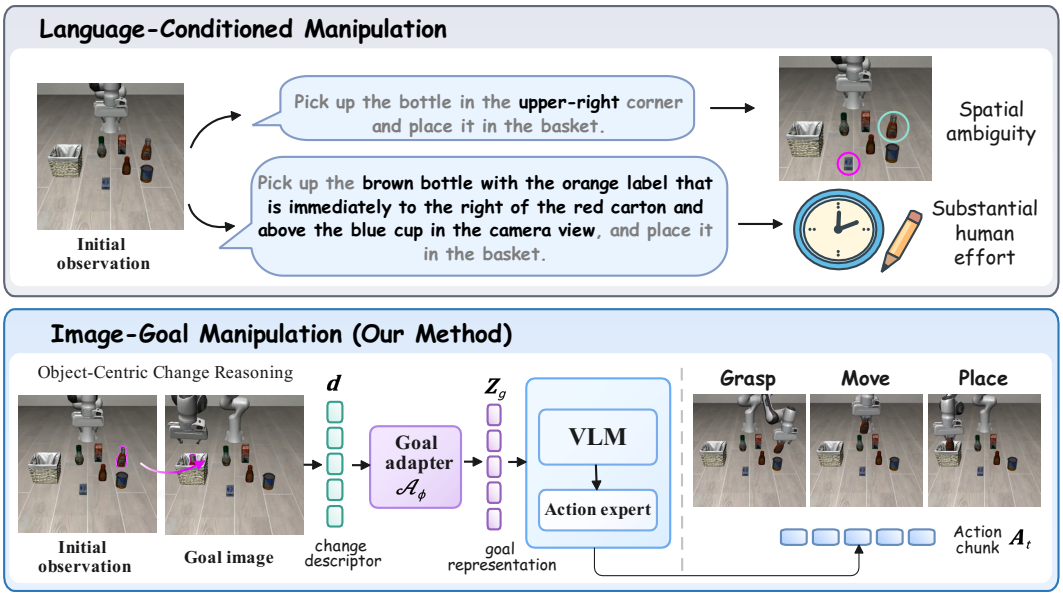


Figure 1: Motivation of Change2Act for visual-goal manipulation. Language-conditioned manipulation relies on language instructions to specify goals, which can introduce spatial ambiguity and require substantial human effort for fine-grained descriptions. Change2Act identifies object-level changes between the initial and goal images and maps them into compact goal tokens to condition a pretrained policy for continuous manipulation.

However, tasks such as scene restoration, object repositioning, and layout imitation require precise goal specifications. The policy must know which object to manipulate and what its final position, orientation, and spatial relationships should be. Language is often insufficient for describing such details Jiang et al. (2023). Fine-grained descriptions are costly to annotate and may introduce ambiguity. A goal image provides a more direct representation of the desired scene. Nevertheless, simply adding it as another visual input does not clearly reveal the required manipulation. The current and goal images may also differ in robot pose, occlusion, and background appearance. Therefore, the policy must identify task-relevant object changes while ignoring unrelated visual differences. It must then determine which object to manipulate and how to change its state.

To this end, we introduce Image-Goal Manipulation (IGM), where a goal image directly specifies the desired environment state and the policy generates continuous manipulation actions from the current visual observation to realize that state. IGM poses two key challenges. First, the policy must identify task-relevant objects and their desired changes despite substantial visual differences between the current observation and goal image. Differences in robot pose, occlusion, and background appearance can obscure the object changes that define the manipulation objective. The policy therefore needs to determine which object should be manipulated and how its configuration should change. Second, the policy must translate the identified object changes into actionable objectives, enabling the visual goal to guide object localization, interaction, and continuous action generation.

To address these challenges, we propose Change2Act, a framework that connects explicit visual change reasoning with lightweight policy adaptation. Specifically, we design an Object-Centric Change Reasoning module that establishes semantic correspondences between objects in the initial and goal images. It organizes the matched objects' semantic features, spatial states, and their differences into a structured change descriptor, highlighting task-relevant changes. We further design a Change-Guided Action Conditioning module that encodes object changes and provides them alongside the original goal image as inputs to a pretrained policy. The goal image depicts the desired scene layout, while the change information highlights the object to manipulate and its initial and target positions, providing concrete cues for action generation.

To evaluate Change2Act, we construct IGM-Restore, a visual-goal manipulation dataset for scene restoration derived from LIBERO demonstrations. The dataset pairs current observations with goal

images specifying the desired scene configurations. We then conduct comparative experiments and ablation studies to evaluate action prediction accuracy and parameter efficiency. Our contributions are summarized as follows:

- We formulate Image-Goal Manipulation (IGM) and propose Change2Act, a framework that connects explicit visual change reasoning with parameter-efficient policy adaptation.
- We design Object-Centric Change Reasoning to identify task-relevant object correspondences and spatial changes, and Change-Guided Action Conditioning to integrate this information with the goal image for action generation.
- We construct IGM-Restore from LIBERO demonstrations for visual-goal scene restoration. Comparative experiments and ablation studies demonstrate that Change2Act improves offline action prediction accuracy with fewer trainable parameters and validate the contributions of its components.

2 Related Work

Robot Manipulation. Robot manipulation requires policies to generate continuous control actions from environmental observations and task goals. Recent vision-language-action models combine vision-language representations with learning from robot demonstrations, advancing multi-task manipulation. Research in this area primarily focuses on visual processing, action modeling, and policy adaptation. AVA-VLA Xiao et al. (2026) maintains execution history through recurrent states and uses active visual attention to strengthen task-relevant representations. OpenVLA-OFT Kim et al. (2025) systematically studies action decoding and fine-tuning strategies, combining parallel decoding, action chunking, and continuous action regression to improve policy performance and inference efficiency. ResVLA Zhong et al. (2026b) decomposes actions into low-frequency intent anchors and high-frequency residuals, refining local dynamics through a residual diffusion bridge. $\pi_{0.5}$ Black et al. (2025) combines heterogeneous robot data, vision-language knowledge, and high-level semantic prediction through co-training to improve open-world generalization. Octo Mees et al. (2024) follows a similar task-conditioned action generation paradigm, jointly processing language instructions or goal images with current observations to predict actions. Its architecture does not explicitly establish object-level correspondences between current and goal images.

Visual Change Detection. Visual change detection aims to localize content changes between observations of the same scene while suppressing nuisance variations in illumination and viewpoint. Early deep learning methods primarily extract and compare image features through dual-branch networks. Daudt et al. (2018) introduce fully convolutional Siamese networks for change prediction from co-registered images. Subsequent studies incorporate attention mechanisms to improve cross-image association and contextual modeling. DR-TANet Chen et al. (2021) models local dependencies between street-view images through temporal attention with a dynamic receptive scope. ChangeFormer Bandara & Patel (2022) combines a hierarchical Transformer encoder with a multilayer perceptron decoder to capture multi-scale features and long-range relationships for remote sensing change detection. More recently, visual foundation models have provided transferable region and semantic representations for change detection. SAM Kirillov et al. (2023) enables promptable region segmentation, Grounding DINO Liu et al. (2024) supports open-set object localization, and DINOv2 Oquab et al. (2023) learns general-purpose image and dense visual features through self-supervision. Building on these capabilities, AnyChange Zheng et al. (2024) performs zero-shot change detection through bitemporal matching in the SAM latent space without additional training. Lin et al. (2025) combine frozen DINOv2 features with full-image cross-attention to learn correspondences and non-correspondences, improving robustness to illumination and viewpoint variations. Such region and semantic representations have also been applied to robotic manipulation. DexGraspVLA Zhong et al. (2026a) uses SAM to obtain target object masks and combines frozen DINOv2 features with a diffusion policy for dexterous grasping. ReKep Huang et al. (2025) generates keypoint candidates by clustering DINOv2 features within SAM segmentation masks, then uses a vision-language model to formulate relational keypoint constraints for multi-stage manipulation.

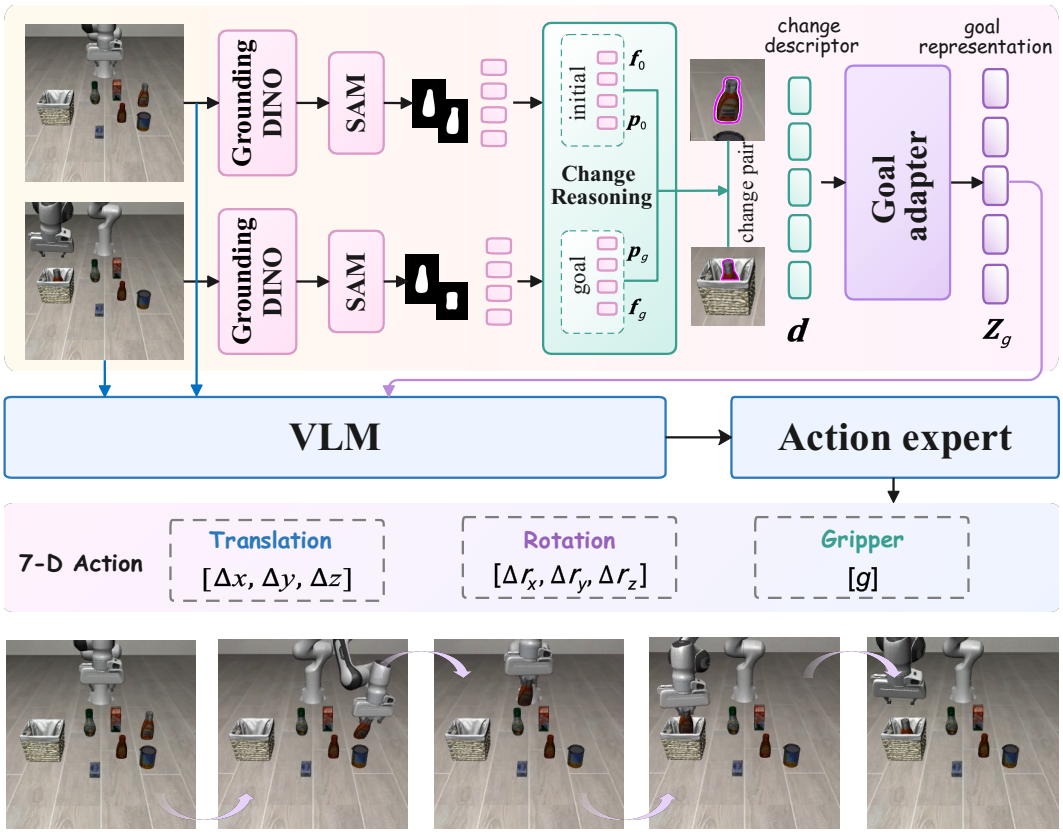


Figure 2: Overview of Change2Act. Object-Centric Change Reasoning establishes object correspondences between the initial and goal images and encodes their semantic and spatial changes into a structured change descriptor $\mathbf{d}$. Change-Guided Action Conditioning maps $\mathbf{d}$ into compact goal tokens $\mathbf{Z}_g$, which complement the goal image to condition the frozen policy. The action expert predicts action chunks, with each action comprising three translation components, three rotation components, and one gripper command. The bottom row illustrates a manipulation sequence toward the visual goal.

3 METHOD

3.1 OVERVIEW

We study Image-Goal Manipulation (IGM), where a robot generates continuous actions to reach the scene configuration specified by a goal image. As shown in Figure 2, given the current scene image I_t, wrist-view image I_t^w, robot state s_t, and goal image I_g, we propose Change2Act to predict an action sequence $\mathbf{A}_t = (a_t, \ldots, a_{t+H-1})$, where H denotes the prediction horizon. Each action is represented as $\mathbf{a}_t = [\Delta x_t, \Delta y_t, \Delta z_t, \Delta r_{x,t}, \Delta r_{y,t}, \Delta r_{z,t}, g_t]^\top \in \mathbb{R}^7$, where the first three components specify translation commands, the next three specify rotation commands, and g_t denotes the gripper command. Change2Act consists of two components. The Object-Centric Change Reasoning module establishes object-level semantic correspondences between the initial scene image I_0 and goal image I_g, and organizes the matched objects' semantic features, spatial states, and their differences into a fixed change descriptor $\mathbf{d}$. The Change-Guided Action Conditioning module maps $\mathbf{d}$ into a compact goal representation $\mathbf{Z}_g$, which is provided alongside I_g and the current observations to condition the pretrained policy.

3.2 OBJECT-CENTRIC CHANGE REASONING

Identifying object changes requires reducing interference from variations in robot pose, occlusion, and background appearance. We design an Object-Centric Change Reasoning module to establish

object correspondences between the initial scene image I_0 and goal image I_g, and construct a representation of the matched object's changes. We first extract object candidates using a pretrained segmentation model, yielding candidate sets $\mathcal{O}^0$ and $\mathcal{O}^g$. Robot regions are estimated through detection and segmentation, and candidates with excessive overlap with the robot masks are filtered out to reduce interference from robot motion. For each remaining candidate, we extract a local visual semantic feature $\mathbf{f}$ and a spatial feature $\mathbf{p}$ encoding its image location and scale.

For the initial and goal images captured from aligned viewpoints, we obtain a change mask M through RGB pixel differencing, adaptive thresholding, and morphological processing. This mask provides spatial cues for object matching. We then establish cross-image correspondences by combining semantic similarity, change association, and geometric information. For a candidate pair o_i^0 and o_j^g, we compute the cosine similarity between their semantic features as s_{ij}^{sem}. We measure the fraction of each candidate region covered by the change mask and take the geometric mean of these fractions to obtain the change association score s_{ij}^{chg}. The geometric score s_{ij}^{geo} multiplies the distance between normalized centroids by a scale penalty derived from their area ratio, favoring displaced objects with similar scales. The joint scoring function Ψ multiplies the semantic similarity, clipped at zero, by the other two scores. We first filter candidate pairs using semantic, change-overlap, and geometric criteria to form a valid set $\mathcal{V}$, then select the highest-scoring pair:

$$(i^*, j^*) = \underset{(i,j)\in\mathcal{V}}{\arg\max}\ \Psi\left(s_{ij}^{\text{sem}}, s_{ij}^{\text{chg}}, s_{ij}^{\text{geo}}\right). \tag{1}$$

For the selected pair (i^*, j^*), let $\mathbf{f}_0$ and $\mathbf{f}_g$ denote the semantic features of the matched objects in the initial and goal images, and let $\mathbf{p}_0$ and $\mathbf{p}_g$ denote their spatial features. We construct the change descriptor as

$$\mathbf{d} = \text{Concat}(\mathbf{f}_0, \mathbf{f}_g, \Delta\mathbf{f}, \mathbf{p}_0, \mathbf{p}_g, \Delta\mathbf{p}), \tag{2}$$

where $\Delta\mathbf{f} = \mathbf{f}_g - \mathbf{f}_0$ and $\Delta\mathbf{p} = \mathbf{p}_g - \mathbf{p}_0$. This descriptor preserves the object's initial and desired configurations while explicitly encoding their differences.

3.3 Change-Guided Action Conditioning

To guide action generation with the identified object changes, we design a Change-Guided Action Conditioning module that incorporates the output of Object-Centric Change Reasoning into a pretrained robot policy. The object-level change representation describes the selected object and its desired spatial change. The policy needs to combine this information with the current scene and robot state to predict manipulation actions. A learnable goal adapter $\mathcal{A}_\phi$ maps the change descriptor $\mathbf{d}$ into K compact goal tokens $\mathbf{Z}_g \in \mathbb{R}^{K\times D}$, where D is the policy's multimodal embedding dimension. The adapter learns this mapping through action supervision, allowing object semantics and spatial differences to contribute to action prediction. Its output provides the policy with information about the selected object's initial configuration, target configuration, and their differences.

We retain the original goal image alongside $\mathbf{Z}_g$ as inputs to the policy. The goal image depicts the complete desired scene, while the change representation highlights the object to manipulate and its spatial changes. This combination provides context about surrounding objects and the scene layout that is absent from the selected object's representation, together with a focused cue about the region requiring manipulation. To distinguish the desired scene from current observations, we add a learnable goal-role embedding $\mathbf{e}_g \in \mathbb{R}^D$ to each goal-image token. This embedding marks the goal features as describing the configuration to be reached, helping the policy distinguish their role from features describing the current environment. Denoting the goal-image embeddings by $\mathbf{X}^g$, the two conditioning representations are constructed as

$$\mathbf{Z}_g = \mathcal{A}_\phi(\mathbf{d}), \qquad \widetilde{\mathbf{X}}^g = \mathbf{X}^g + \mathbf{e}_g. \tag{3}$$

The policy jointly processes these goal conditions with the current third-person and wrist-view representations $\mathbf{X}_t^c$ and $\mathbf{X}_t^w$, robot state s_t, and generic task-prompt representation $\mathbf{X}^{l_0}$. The third-person observation provides the current scene configuration, while the wrist-view observation supplies a local view of the manipulation area. Together with the robot state, these observations help the policy determine how to achieve the desired object change from its current state. The generic prompt is shared across examples and specifies the scene-restoration task, while the goal image and change

representation provide the specific manipulation objective. For the flow-based action model, the conditioned prediction is expressed as

$$\mathbf{v}_t^\tau = F_\theta\Big(\mathbf{A}_t^\tau, \tau \;\Big|\; \mathbf{X}_t^c, \mathbf{X}_t^w, s_t, \widetilde{\mathbf{X}}^g, \mathbf{Z}_g, \mathbf{X}^{l_0}\Big), \tag{4}$$

where F_θ denotes the pretrained action model, τ denotes the flow time, $\mathbf{A}_t^\tau$ is the noisy action sequence at time τ, and $\mathbf{v}_t^\tau$ is the predicted velocity field used to generate the action sequence $\mathbf{A}_t$.

For a given initial–goal image pair, the change representation and goal image provide a fixed task condition, while the current observations and robot state are updated across prediction steps. The policy therefore receives a consistent description of the desired outcome while adapting its action predictions to the current environment state. Through this conditioning process, the object changes identified by the preceding module become inputs to continuous action prediction, connecting visual goal understanding with the pretrained policy's control capabilities.

3.4 Training Strategy

Change2Act adopts a lightweight adaptation strategy with pretrained $\pi_{0.5}$ as the action backbone, learning visual goal conditions to guide action generation. During training, both the visual models used for object-level change reasoning and the pretrained policy backbone remain frozen. Only the goal adapter and goal-role embedding are optimized. The goal adapter converts object-change information into a goal representation that the policy can use, while the goal-role embedding identifies the role of the goal image within the multimodal input. This design restricts training to the additional goal-conditioning components, totaling approximately 1.649M trainable parameters.

Training follows the flow-matching objective of the action expert. Given a demonstration action sequence $\mathbf{A}_t$, we sample standard Gaussian noise $\boldsymbol{\omega}$ and a flow time τ, and construct the noisy action sequence as

$$\mathbf{A}_t^\tau = (1-\tau)\mathbf{A}_t + \tau\boldsymbol{\omega}. \tag{5}$$

Here, τ controls the noise level, and the corresponding target velocity is $\boldsymbol{\omega} - \mathbf{A}_t$. We denote the conditioning context by $\mathcal{C}_t$, which comprises the current third-person and wrist-view observations, robot state, goal-image features with the goal-role embedding, compact goal representation, and generic task prompt. The model predicts the velocity field from the noisy action sequence, flow time, and conditioning context. The training objective is

$$\mathcal{L}_{\mathrm{FM}} = \mathbb{E}_{\mathbf{A}_t,\tau,\boldsymbol{\omega}}\left[\left\|F_\theta(\mathbf{A}_t^\tau, \tau \mid \mathcal{C}_t) - (\boldsymbol{\omega} - \mathbf{A}_t)\right\|_2^2\right]. \tag{6}$$

Here, θ denotes the frozen pretrained policy parameters. Within $\mathcal{C}_t$, the goal representation depends on the adapter parameters ϕ, while the goal-image features incorporate the learnable role embedding $\mathbf{e}_g$. Gradients propagate through the pretrained policy to these conditioning components, updating only ϕ and $\mathbf{e}_g$. Through action supervision, the goal adapter learns to encode object semantics and spatial changes into effective action conditions, while the role embedding learns to identify the desired scene. Together, these components enable visual goals to guide continuous action prediction with a small number of trainable parameters.

4 Experiments

4.1 Experimental Setup

Datasets. We construct IGM-Restore from successful LIBERO demonstrations Liu et al. (2023), a visual-goal manipulation dataset for scene restoration that supports the evaluation of offline action prediction. For each trajectory, we define the initial scene image I_0 as the first valid third-person observation before the first expert action, and the goal image I_g as the final stable frame after task completion. Both images come from the same demonstration and camera view, so their differences primarily reflect changes in object configurations, robot pose, and occlusion. The goal image remains fixed throughout the trajectory and specifies the desired object positions, orientations, and scene layout. Training samples are constructed from all valid timesteps within the retained trajectories. At each timestep t, the current third-person image I_t, synchronized wrist-view image, and robot state are paired with the fixed goal image I_g, with the expert action sequence starting at t serving as

supervision. For Change2Act, the object-level change representation $\mathbf{d}$ is precomputed from I_0 and I_g and shared across all timesteps of the same trajectory, while the current observations and robot state vary over time. Each demonstration therefore yields multiple action prediction samples. Expert actions use a seven-dimensional representation comprising three translation components, three rotation components, and one gripper component. All samples share the generic instruction, *"Restore the scene to match the reference image."* Specific task goals are provided by the goal image, without object names or task-specific language descriptions.

We split the trajectories into training, validation, and test sets at the episode level, stratified by scene group, with an approximate ratio of $8 : 1 : 1$. All images, actions, and derived samples from the same trajectory are assigned to a single split, preventing overlap of adjacent frames or samples from the same manipulation sequence across splits. The validation set is used exclusively for model selection and hyperparameter tuning, and final performance is reported on a fixed held-out test set. The test set contains 2,765 expert action targets, each counted exactly once during evaluation.

Baselines. We compare Change2Act with representative robot manipulation policies, including AVA-VLA Xiao et al. (2026), OpenVLA-OFT Kim et al. (2025), ResVLA Zhong et al. (2026b), Octo Mees et al. (2024), and $\pi_{0.5}$ Black et al. (2025). All baselines are fine-tuned on IGM-Restore using the same data splits and generic instruction as Change2Act. AVA-VLA, OpenVLA-OFT, and $\pi_{0.5}$ are adapted via LoRA, while ResVLA and Octo are fully fine-tuned. For models supporting multiple image inputs, we provide both the current and goal images through their native visual input interfaces. All methods are evaluated on the same held-out test set using identical offline action prediction metrics.

Implementation Details. We use the pretrained $\pi_{0.5}$ LIBERO model as the policy backbone, with an action prediction horizon of 10. Object candidates, robot masks, and local visual features are obtained using SAM, Grounding DINO, and DINOv2, with all three models kept frozen. The matched objects' visual features and geometric information from the initial and goal images are precomputed to form a 1,161-dimensional object-level change representation, which is mapped to four goal tokens through a lightweight goal adapter. We freeze the $\pi_{0.5}$ backbone and optimize only the goal adapter and goal-role embedding, yielding 1.649M trainable parameters. The model is trained on a single NVIDIA RTX 4090 for 2,000 optimization steps with a batch size of 1. We use AdamW with a gradient clipping threshold of 1.0. The learning rate is increased to 1×10^{-4} during the first 100 warm-up steps and then reduced to 1×10^{-5} following a cosine decay schedule. At inference, we generate action sequences using 10 flow-matching sampling steps and denormalize the predictions to the original LIBERO action space.

Metrics. We evaluate offline action prediction after denormalizing predictions to the original seven-dimensional LIBERO action space. Action MAE measures the mean absolute error between predicted and ground-truth actions across all dimensions, while Action RMSE measures the root mean squared error and is more sensitive to large deviations. Translation MAE and Rotation MAE are computed over the first three translation components and the subsequent three rotation components, respectively, to assess prediction accuracy for each action component. Gripper Accuracy measures agreement between predicted and expert gripper commands at each evaluated timestep. We binarize both predicted and ground-truth gripper control values using a threshold of 0.5 and compute the proportion of matching binary states. Lower values indicate better performance for all error metrics, whereas higher values indicate better Gripper Accuracy.

4.2 Performance Comparison

We evaluate Change2Act against representative robot manipulation policies in terms of offline action prediction accuracy and parameter efficiency. As reported in Table 1, Change2Act achieves the best results across all five evaluation metrics. Compared with the strongest baseline, $\pi_{0.5}$, it reduces Action MAE from 0.061 to 0.038 and Action RMSE from 0.131 to 0.081, corresponding to relative reductions of 37.7% and 38.2%, respectively. Translation MAE and Rotation MAE decrease to 0.069 and 0.014, while Gripper Accuracy increases by 4.05 percentage points to 98.59%. These gains indicate improved prediction of both continuous motion and gripper states. Both methods are trained for 2,000 optimization steps, but Change2Act updates only 1.649M parameters, approxi-

Table 1: Comparison with representative robot manipulation policies on the IGM-Restore test set.

Method	Trainable Params.	Train. Steps	Action MAE ↓	Action RMSE ↓	Trans. MAE ↓	Rot. MAE ↓	Gripper Acc. ↑
AVA-VLA	27.71M	7401	0.073	0.149	0.113	0.028	92.44%
OpenVLA-OFT	27.71M	7401	0.071	0.147	0.114	0.028	93.06%
ResVLA	3.95B	5000	0.252	0.434	0.394	0.047	56.06%
Octo	202.46M	49340	0.093	0.199	0.146	0.028	88.57%
$\pi_{0.5}$	466.96M	2000	0.061	0.131	0.100	0.023	94.54%
Change2Act (Ours)	1.65M	2000	**0.038**	**0.081**	**0.069**	**0.014**	**98.59%**

Table 2: Ablation results on IGM-Restore.

Change Representation	Goal-Role Embedding	Action MAE ↓	Action RMSE ↓	Trans. MAE ↓	Rot. MAE ↓	Gripper Acc. ↑
×	✓	0.046	0.106	0.082	0.017	97.36%
✓	×	0.040	0.090	0.072	0.015	97.94%
✓	✓	**0.038**	**0.081**	**0.069**	**0.014**	**98.59%**

mately 0.35% of those updated by the $\pi_{0.5}$ baseline. These results demonstrate the effectiveness and parameter efficiency of the proposed visual-goal conditioning.

4.3 Ablation Studies

To examine the contributions of the conditioning components, we evaluate two ablations: removing the change descriptor together with its corresponding goal tokens, and removing the goal-role embedding. The results are summarized in Table 2. Removing the descriptor and goal tokens increases Action MAE from 0.038 to 0.046 and Action RMSE from 0.081 to 0.106, while reducing Gripper Accuracy to 97.36%. This indicates that explicit object-level transition information provides useful conditioning beyond the goal image alone. Removing the goal-role embedding increases Action RMSE to 0.090 and reduces Gripper Accuracy to 97.94%, with degradation in the remaining error metrics. This supports the benefit of distinguishing the semantic roles of the current observation and goal image. The full model performs best across all metrics, supporting the effectiveness of structured transition conditioning and goal-role encoding.

4.4 Parameter Efficiency Analysis

We evaluate baseline variants with different trainable parameter budgets to examine the relationship between adaptation size and offline action prediction accuracy. As shown in Figure 3, rank-1 LoRA applied to all configured target layers of AVA-VLA uses 3.463M trainable parameters and achieves an Action MAE of 0.076. Restricting LoRA to the query, key, value, and output projections reduces the parameter count to 1.049M, while increasing Action MAE to 0.082. Head-only fine-tuning of Octo uses 1.802M parameters and yields an Action MAE of 0.275. Change2Act achieves the lowest Action MAE of 0.038 with 1.649M trainable parameters, together with the lowest translation and rotation errors among the evaluated variants. Compared with the 3.463M-parameter AVA-VLA variant, Change2Act halves Action MAE while using fewer than half as many trainable parameters. These results support the effectiveness of combining explicit object-level change information with lightweight goal conditioning for parameter-efficient policy adaptation.

4.5 Case Study

To illustrate how Change2Act translates visual goals into manipulation actions, we compare its execution with AVA-VLA on three scene-restoration tasks, as shown in Figure 4. These tasks require identifying the relevant object and its desired configuration among multiple scene objects, despite differences in robot pose and occlusion between the initial and goal images. AVA-VLA fails to

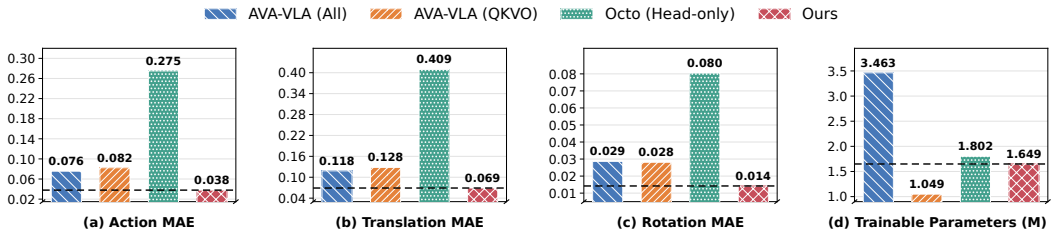


Figure 3: Offline action prediction errors and trainable parameter counts of different adaptation strategies on IGM-Restore. Dashed lines indicate Change2Act's values.

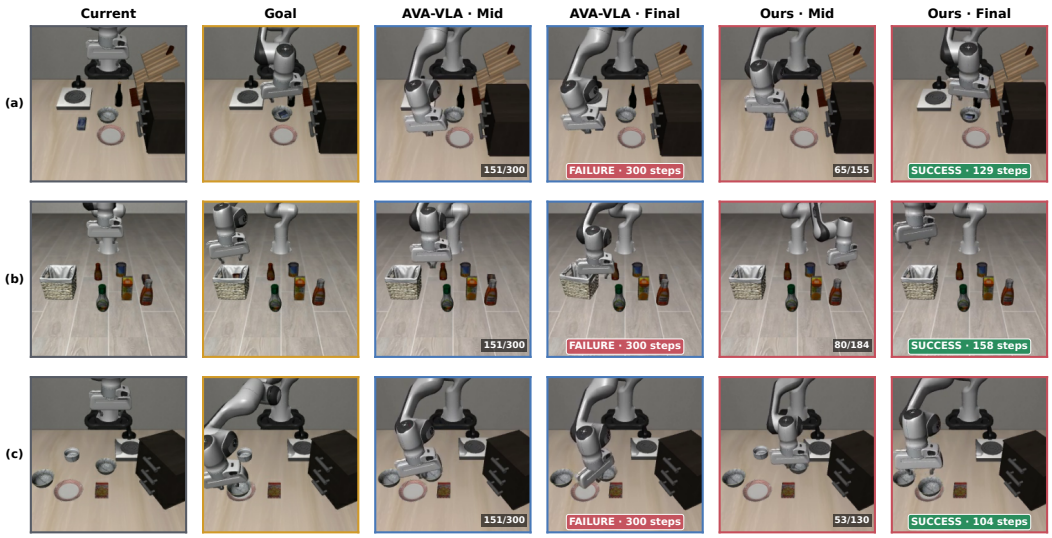


Figure 4: Qualitative execution comparisons on three visual-goal scene-restoration tasks. Each row shows the initial observation, goal image, and intermediate and final observations of AVA-VLA and Change2Act. Change2Act reaches the desired object configurations in the three examples, whereas AVA-VLA does not complete the tasks within the 300-step limit. Final-frame annotations report execution outcomes and control-step counts.

complete the tasks within 300 control steps, whereas Change2Act succeeds in 129, 158, and 104 steps, respectively. The intermediate and final observations show Change2Act progressively moving the relevant objects toward their target configurations. These examples complement the offline results and illustrate the effectiveness of object-level change reasoning and goal conditioning in the depicted tasks.

5 Conclusion

We presented Change2Act, a framework for Image-Goal Manipulation that connects visual change reasoning with pretrained policy adaptation. Object-Centric Change Reasoning establishes semantic correspondences and encodes object-level spatial changes into a structured descriptor. Change-Guided Action Conditioning maps this descriptor into compact goal tokens and combines them with the goal image, using a goal-role embedding to distinguish the desired state from current observations. With a frozen policy backbone and only 1.649M trainable parameters, Change2Act achieves the best offline action prediction results among the compared methods on IGM-Restore. Ablation studies and qualitative execution examples further support the effectiveness of explicit object-level change information for visual-goal manipulation.

AI USE STATEMENT

We used generative AI tools for language editing to improve the clarity and readability of the manuscript, and for literature searches to better understand developments and prior work in the field. We also used these tools to discuss research ideas and obtain suggestions for conceptual illustrations, helping organize the research content and improve the presentation of our method. The authors reviewed and revised the AI-assisted content, verified the accuracy and relevance of the references, and checked that the text and figures were consistent with the implemented method and experimental results. The authors take full responsibility for the accuracy, originality, and integrity of the final manuscript.

ETHICS STATEMENT

This study does not involve human subjects, personally identifiable information, or sensitive personal data. The experimental data used in this work are constructed based on the existing LIBERO robot manipulation benchmark and its successful demonstration trajectories. The dataset is constructed and used solely for research and evaluation in robotic visual understanding and manipulation policies, without any additional collection of human-subject data or processing of personal information. All experiments and analyses in this work are conducted in accordance with principles of research integrity and responsible data use.

REPRODUCIBILITY STATEMENT

We provide the methodological and experimental details necessary to reproduce our results throughout the paper. Section 3 describes the proposed Change2Act framework, including object-centric change reasoning, change-guided action conditioning, and the training objective. Section 4.1 details the construction and episode-level split of IGM-Restore, baseline configurations, evaluation metrics, and implementation settings, including the pretrained models, trainable components, optimization protocol, and inference procedure. IGM-Restore is constructed from the publicly available LIBERO benchmark, and the pretrained models used in our framework are also publicly available. These descriptions are intended to support reproducible implementation and evaluation of the proposed method.

REFERENCES

Bo Ai, Stephen Tian, Haochen Shi, Yixuan Wang, Tobias Pfaff, Cheston Tan, Henrik I Christensen, Hao Su, Jiajun Wu, and Yunzhu Li. A review of learning-based dynamics models for robotic manipulation. *Science Robotics*, 10(106):eadt1497, 2025.

J Andrew Bagnell, Felipe Cavalcanti, Lei Cui, Thomas Galluzzo, Martial Hebert, Moslem Kazemi, Matthew Klingensmith, Jacqueline Libby, Tian Yu Liu, Nancy Pollard, et al. An integrated system for autonomous robotics manipulation. In *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 2955–2962. IEEE, 2012.

Wele Gedara Chaminda Bandara and Vishal M Patel. A transformer-based siamese network for change detection. In *IGARSS 2022-2022 IEEE international geoscience and remote sensing symposium*, pp. 207–210. IEEE, 2022.

Kevin Black, Noah Brown, James Darpinian, Karan Dhabalia, Danny Driess, Adnan Esmail, Michael Robert Equi, Chelsea Finn, Niccolo Fusai, Manuel Y. Galliker, Dibya Ghosh, Lachy Groom, Karol Hausman, brian ichter, Szymon Jakubczak, Tim Jones, Liyiming Ke, Devin LeBlanc, Sergey Levine, Adrian Li-Bell, Mohith Mothukuri, Suraj Nair, Karl Pertsch, Allen Z. Ren, Lucy Xiaoyang Shi, Laura Smith, Jost Tobias Springenberg, Kyle Stachowicz, James Tanner, Quan Vuong, Homer Walke, Anna Walling, Haohuan Wang, Lili Yu, and Ury Zhilinsky. $\pi_{0.5}$: a vision-language-action model with open-world generalization. In Joseph Lim, Shuran Song, and Hae-Won Park (eds.), *Proceedings of The 9th Conference on Robot Learning*, volume 305 of *Proceedings of Machine Learning Research*, pp. 17–40. PMLR, 27–30 Sep 2025. URL https://proceedings.mlr.press/v305/black25a.html.

Anthony Brohan, Noah Brown, Justice Carbajal, Yevgen Chebotar, Xi Chen, Krzysztof Choromanski, Tianli Ding, Danny Driess, Avinava Dubey, Chelsea Finn, et al. Rt-2: Vision-language-action models transfer web knowledge to robotic control. *arXiv preprint arXiv:2307.15818*, 2023.

Shuo Chen, Kailun Yang, and Rainer Stiefelhagen. Dr-tanet: Dynamic receptive temporal attention network for street scene change detection. In *2021 IEEE Intelligent Vehicles Symposium (IV)*, pp. 502–509. IEEE, 2021.

Cheng Chi, Zhenjia Xu, Siyuan Feng, Eric Cousineau, Yilun Du, Benjamin Burchfiel, Russ Tedrake, and Shuran Song. Diffusion policy: Visuomotor policy learning via action diffusion. *The International Journal of Robotics Research*, 44(10-11):1684–1704, 2025.

Rodrigo Caye Daudt, Bertr Le Saux, and Alexandre Boulch. Fully convolutional siamese networks for change detection. In *2018 25th IEEE international conference on image processing (ICIP)*, pp. 4063–4067. IEEE, 2018.

Wenlong Huang, Chen Wang, Yunzhu Li, Ruohan Zhang, and Li Fei-Fei. Rekep: Spatio-temporal reasoning of relational keypoint constraints for robotic manipulation. In Pulkit Agrawal, Oliver Kroemer, and Wolfram Burgard (eds.), *Proceedings of The 8th Conference on Robot Learning*, volume 270 of *Proceedings of Machine Learning Research*, pp. 4573–4602. PMLR, 06–09 Nov 2025. URL https://proceedings.mlr.press/v270/huang25g.html.

Yunfan Jiang, Agrim Gupta, Zichen Zhang, Guanzhi Wang, Yongqiang Dou, Yanjun Chen, Li Fei-Fei, Anima Anandkumar, Yuke Zhu, and Linxi Fan. Vima: Robot manipulation with multimodal prompts. 2023.

Moo Jin Kim, Chelsea Finn, and Percy Liang. Fine-tuning vision-language-action models: Optimizing speed and success. *arXiv preprint arXiv:2502.19645*, 2025.

Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C Berg, Wan-Yen Lo, et al. Segment anything. In *2023 IEEE/CVF international conference on computer vision (ICCV)*, pp. 3992–4003. IEEE, 2023.

Oliver Kroemer, Scott Niekum, and George Konidaris. A review of robot learning for manipulation: Challenges, representations, and algorithms. *Journal of Machine Learning Research*, 22(30): 1–82, 2021. URL http://jmlr.org/papers/v22/19-804.html.

Chun-Jung Lin, Sourav Garg, Tat-Jun Chin, and Feras Dayoub. Robust scene change detection using visual foundation models and cross-attention mechanisms. In *2025 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 8337–8343. IEEE, 2025.

Bo Liu, Yifeng Zhu, Chongkai Gao, Yihao Feng, Qiang Liu, Yuke Zhu, and Peter Stone. Libero: Benchmarking knowledge transfer for lifelong robot learning. *Advances in Neural Information Processing Systems*, 36:44776–44791, 2023.

Shilong Liu, Zhaoyang Zeng, Tianhe Ren, Feng Li, Hao Zhang, Jie Yang, Qing Jiang, Chunyuan Li, Jianwei Yang, Hang Su, et al. Grounding dino: Marrying dino with grounded pre-training for open-set object detection. In *European conference on computer vision*, pp. 38–55. Springer, 2024.

Oier Mees, Dibya Ghosh, Karl Pertsch, Kevin Black, Homer Rich Walke, Sudeep Dasari, Joey Hejna, Tobias Kreiman, Charles Xu, Jianlan Luo, et al. Octo: An open-source generalist robot policy. In *First workshop on vision-language models for navigation and manipulation at ICRA 2024*, 2024.

Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, et al. Dinov2: Learning robust visual features without supervision. *arXiv preprint arXiv:2304.07193*, 2023.

Mohit Shridhar, Lucas Manuelli, and Dieter Fox. Cliport: What and where pathways for robotic manipulation. In *Conference on robot learning*, pp. 894–906. pmlr, 2022.

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. In *Towards Generalist Robots: Learning Paradigms for Scalable Skill Acquisition@ CoRL2023*, 2023.

Lei Xiao, Jifeng Li, Juntao Gao, Feiyang Ye, Yan Jin, Jingjing Qian, Jing Zhang, Yong Wu, and Xiaoyuan Yu. Ava-vla: improving vision-language-action models with active visual attention. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 13453–13463, 2026.

Andy Zeng, Pete Florence, Jonathan Tompson, Stefan Welker, Jonathan Chien, Maria Attarian, Travis Armstrong, Ivan Krasin, Dan Duong, Vikas Sindhwani, and Johnny Lee. Transporter networks: Rearranging the visual world for robotic manipulation. In Jens Kober, Fabio Ramos, and Claire Tomlin (eds.), *Proceedings of the 2020 Conference on Robot Learning*, volume 155 of *Proceedings of Machine Learning Research*, pp. 726–747. PMLR, 16–18 Nov 2021. URL https://proceedings.mlr.press/v155/zeng21a.html.

Zhuo Zheng, Yanfei Zhong, Liangpei Zhang, and Stefano Ermon. Segment any change. *Advances in Neural Information Processing Systems*, 37:81204–81224, 2024.

Yifan Zhong, Xuchuan Huang, Ruochong Li, Ceyao Zhang, Zhang Chen, Tianrui Guan, Fanlian Zeng, Ka Nam Lui, Yuyao Ye, Yitao Liang, et al. Dexgraspvla: A vision-language-action framework towards general dexterous grasping. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pp. 18836–18844, 2026a.

Yiming Zhong, Yaoyu He, Zemin Yang, Pengfei Tian, Yifan Huang, Qingqiu Huang, Xinge Zhu, and Yuexin Ma. From noise to intent: Anchoring generative vla policies with residual bridges, 2026b. URL https://arxiv.org/abs/2604.21391.

Pei Zhou, Wanting Yao, Qian Luo, Xunzhe Zhou, and Yanchao Yang. Hyper-goalnet: Goal-conditioned manipulation policy learning with hypernetworks. *Advances in Neural Information Processing Systems*, 38:83438–83469, 2026.

A Appendix

A.1 Controlled Interventions on the Change Descriptor

To examine whether action prediction relies on the identified object-level changes, we perform controlled interventions on the change descriptor d while keeping the model weights, visual inputs, robot states, and sampling noise fixed. The semantic features characterize the selected object's visual semantics, while the spatial information includes its position and scale features p_0, p_g in the initial and goal images, together with their difference Δp. We replace the semantic features, spatial information, or both. Semantic replacement uses features from a static object in the same scene while preserving the original spatial information. Spatial replacement preserves the semantic features but replaces the corresponding spatial components with information inconsistent with the true object transition. Joint replacement modifies both. As shown in Figure 5, semantic replacement increases Action MAE, Action RMSE, and Translation MAE by 2.99%, 3.81%, and 2.55%, respectively, while spatial replacement increases these errors by 0.67%, 0.74%, and 0.60%. Joint replacement causes the largest degradation, with respective increases of 4.51%, 5.70%, and 4.37%. For the Action MAE increases under these three interventions, one-sided paired Wilcoxon tests yield p-values of 0.0273, 0.0180, and 0.0012, respectively. Reversing the displacement direction also increases Action MAE by 1.86% ($p = 0.0017$). These results support the policy's effective use of object semantics and spatial changes encoded in the descriptor.

A.2 Direct Evaluation of Object-Centric Change Reasoning

We directly evaluate the Object-Centric Change Reasoning module using instance segmentation ground truth provided by LIBERO, considering object selection, cross-image correspondence, and spatial localization. The three recognition metrics are computed at the episode level: Target Selection Accuracy measures the correctness of task-object selection in the initial image, Cross-image

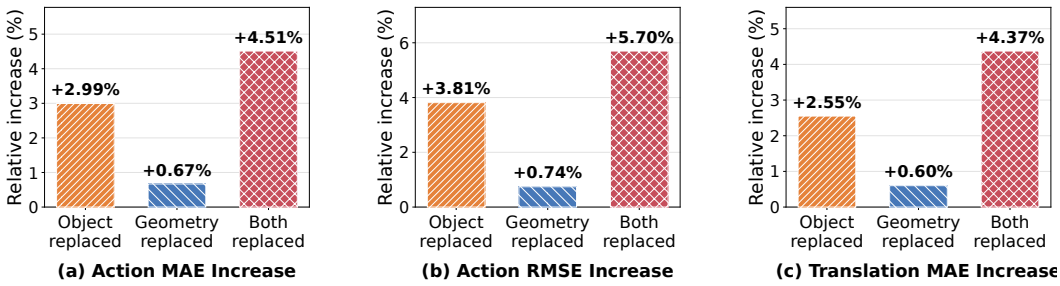


Figure 5: Relative increases in offline action prediction errors after replacing semantic features, spatial information, or both in the change descriptor, compared with the original descriptor.

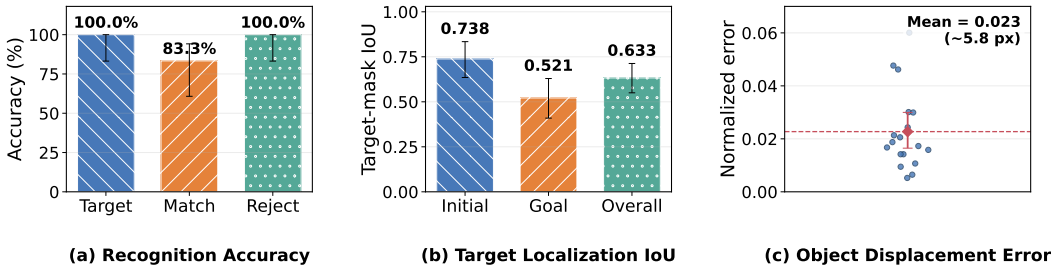


Figure 6: Direct evaluation of Object-Centric Change Reasoning. (a) Target selection, cross-image matching, and distractor rejection rates. (b) Target-mask IoU across initial, goal, and all visible views. (c) Normalized object displacement error.

Matching Accuracy measures whether the selected regions correspond to the same task object across images, and Distractor Rejection Rate measures the proportion of final selections that do not correspond to the robot or other task-irrelevant distractors. As shown in Figure 6(a), Target Selection Accuracy and Distractor Rejection Rate both reach 100% on the fixed test split, while Cross-image Matching Accuracy is 83.33% for samples with a visible target object in the goal image. We further use target-mask IoU to measure the overlap between predicted regions and instance segmentation ground truth. As shown in Figure 6(b), the mean IoU values for the initial images, goal images, and all visible views are 0.738, 0.521, and 0.633, respectively. Figure 6(c) reports the discrepancy between predicted and ground-truth object displacements. For samples where the target is visible in both images, the mean normalized displacement error is 0.0227, corresponding to approximately 5.8 pixels in a 256×256 image. Although mask overlap is lower in the goal images than in the initial images, the low displacement error indicates that the module can still estimate object position changes with reasonable accuracy, providing useful spatial information for the change descriptor.

A.3 Qualitative Analysis of Object-Centric Change Reasoning

Figure 7 presents qualitative results of the Object-Centric Change Reasoning module. In Figure 7 (a–d), the module identifies the task object and establishes correspondences between the initial and goal images, with substantial overlap between the predicted and ground-truth masks. Figure 7 (b) further illustrates correspondence under partial occlusion after the object is placed in the basket. Figure 7 (e) shows a more challenging case in which container occlusion reduces the visible target region, leading to a noticeable discrepancy between the predicted and ground-truth masks.

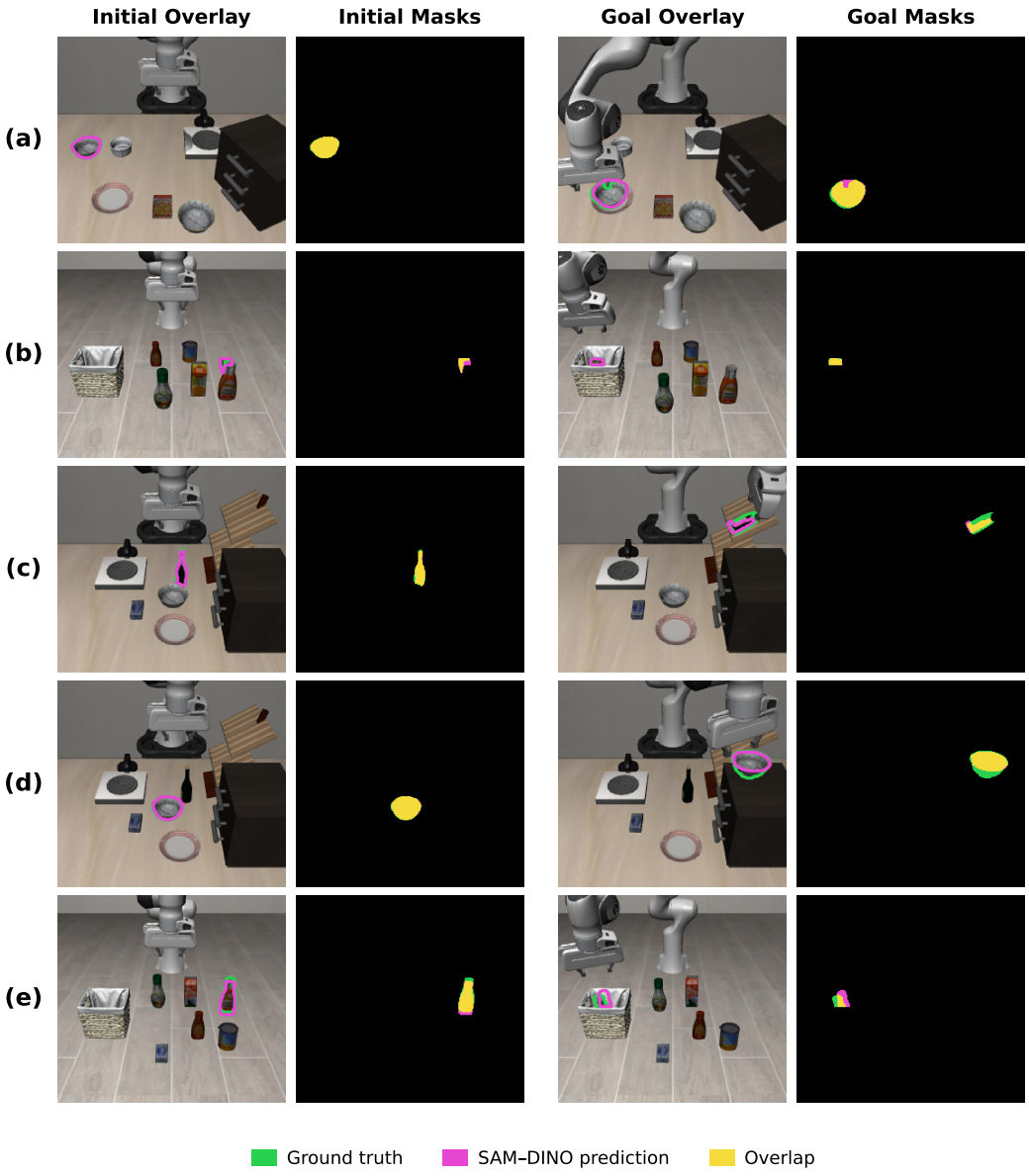


Figure 7: Qualitative results of Object-Centric Change Reasoning: (a–d) successful cases and (e) a challenging case with container occlusion. Green, magenta, and yellow indicate ground-truth masks, predicted masks, and their overlap, respectively.