This CVPR paper is the Open Access version, provided by the Computer Vision Foundation.
Except for this watermark, it is identical to the accepted version;
the final published version of the proceedings is available on IEEE Xplore.

RoboGround: Robotic Manipulation with Grounded Vision-Language Priors

Haifeng Huang[1,2*] Xinyi Chen[2*] Yilun Chen[2‡] Hao Li[2] Xiaoshen Han[2]
Zehan Wang[1] Tai Wang[2] Jiangmiao Pang[2] Zhou Zhao[1,2†]
[1]Zhejiang University [2]Shanghai AI Laboratory
{huanghaifeng}@zju.edu.cn

Abstract

Recent advancements in robotic manipulation have highlighted the potential of intermediate representations for improving policy generalization. In this work, we explore grounding masks as an effective intermediate representation, balancing two key advantages: (1) effective spatial guidance that specifies target objects and placement areas while also conveying information about object shape and size, and (2) broad generalization potential driven by large-scale vision-language models pretrained on diverse grounding datasets. We introduce RoboGround*, a grounding-aware robotic manipulation policy that leverages grounding masks as an intermediate representation to guide policy networks in object manipulation tasks. To further explore and enhance generalization, we propose an automated pipeline for generating large-scale, simulated data with a diverse set of objects and instructions. Extensive experiments show the value of our dataset and the effectiveness of grounding masks as intermediate guidance, significantly enhancing the generalization abilities of robot policies. Code and data will be available at robo-ground.github.io.*

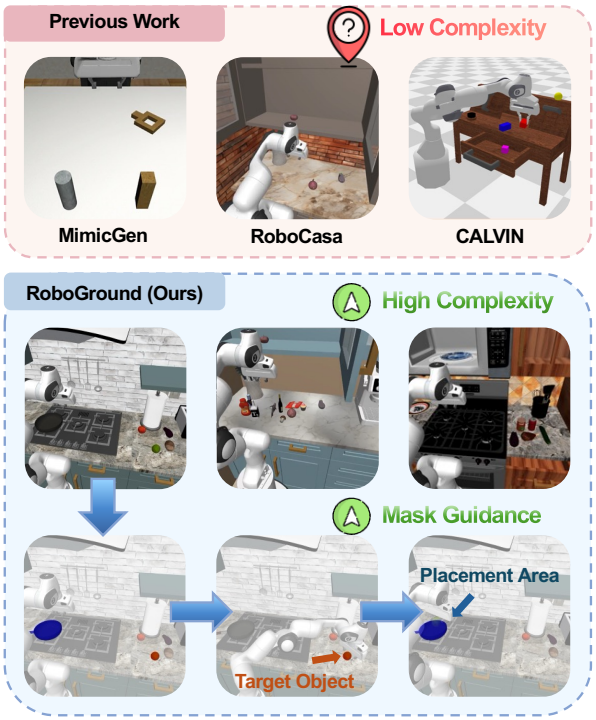

Figure 1. **Examples of generated data and mask guidance for robot policy.** The generated data includes more object distractors in the scene, leading to higher scene complexity. The robot policy is guided by masks to localize the target object and placement area.

1. Introduction

The early low-level policies for manipulation [8, 25, 38, 50] primarily relied on imitation learning from collected demonstrations, focusing on acquiring specific skills within predefined scenes. Consequently, these policies exhibited limited generalization capabilities. More recently, Vision-Language-Action (VLA) models [5, 19] have emerged as a promising approach for learning generalizable robotic policies. These models leverage large-scale robot training data alongside pretrained Vision-Language Models (VLMs) [23, 32, 51] to develop broad manipulation capabilities. However, it is still challenging for these methods to generalize to novel settings without access to extensive datasets and additional fine-tuning, both of which are costly.

Intermediate representations [2, 3, 15, 37] offer an alternative approach to generalized policy learning by providing structured guidance to policy networks across diverse scenarios. Research in this area typically falls into two categories: *accessible yet coarse-grained representations*, such as language instructions [2, 49], which are easy to generate but often lack the spatial precision required for fine-grained object manipulation; and *fine-grained but resource-intensive representations*, such as goal images [4] or point

* Equal contribution.
‡ Project lead.
† Corresponding author.
• Work done during an internship at Shanghai AI Laboratory.

flows [3, 12, 33, 45, 47], which provide detailed spatial guidance but demand extensive training data and computational resources, limiting their scalability.

In this work, we introduce grounding masks as a promising intermediate representation that balances two key aspects: (1) *Effective spatial guidance*, which not only specifies target objects and placement areas but also conveys information about object shape and size, allowing low-level policies to accurately interpret spatial information; and (2) *Broad generalization potential*, facilitated by large-scale vision-language models [21, 32, 44, 48] pretrained on diverse grounding datasets, enhancing adaptability to novel objects and environments. To fully explore grounding mask representations for low-level policy networks, we propose **RoboGround**, a grounding-aware policy guided by object masks for robotic manipulation. Our approach employs a grounded vision-language model to generate masks for both the target object and placement area, which are then seamlessly integrated into the policy network through channel-level and patch-level designs. Specifically, we enhance image inputs via channel concatenation with the masks and introduce a grounded perceiver that attends to mask-guided regions at the patch level, preserving essential spatial information for precise manipulation.

As shown in Figure 1, existing datasets [10, 26–28, 40] often suffer from limited instruction diversity and scene complexity, leading policy networks to overfit to specific conditions and straightforward instruction-scene mappings. In this work, we aim to investigate the effectiveness of grounding mask representations in maintaining consistent performance across diverse scenarios and instructions. To address dataset limitations, we propose **an automated pipeline** for generating simulated manipulation data with a diverse set of objects and instructions. By systematically increasing scene complexity through the inclusion of distractor objects, we create a challenging dataset that promotes generalization across varied instructions and object configurations. Specifically, we generate 24K demonstrations with 112K diverse instructions, covering object appearance, spatial relationships, and commonsense knowledge. Scene complexity is further enhanced by sampling distractor objects from a large pool of 3,526 objects spanning 176 categories. This high-complexity dataset encourages models to leverage intermediate object mask guidance for interpreting instruction-scene relationships, providing a rigorous evaluation of our mask-guided model's generalization potential.

We conduct extensive experiments to evaluate the model's generalization across diverse instructions, unseen objects and categories, and core robotic skills. Our results highlight the effectiveness of grounding masks as intermediate guidance, showing that they substantially enhance the generalization capabilities of robot policies while demonstrating the value of our dataset.

2. Related Work

Intermediate Representations for Robot Policy. Various intermediate guidance strategies have been explored to enhance policy learning, including language instructions [2, 49], 2D trajectories [15, 39, 42], point flows [3, 12, 33, 45, 47], goal images [4, 6], goal sketches [37], and object-centric representations [14, 35, 36]. For instance, RT-Trajectory [15] utilizes rough trajectory sketches to guide policies in performing new tasks, while Im2Flow2Act [45] predicts object flows and maps them to actions. However, acquiring such intermediate guidance during evaluation is often impractical due to the lack of large-scale, diverse datasets and generalizable foundation models. Some approaches leverage object-centric representations to enhance policy robustness. KITE [36] decomposes tasks into keypoints using an additional grounding module, while MOKA [14] employs prompt engineering to derive keypoints that effectively capture affordances in manipulation tasks. Closely related to our mask-based approach, MOO [35] uses pretrained Vision-Language Models (VLMs) to generate coarse bounding boxes, which are then overlaid onto the original image. In contrast, our method focuses on obtaining fine-grained object masks and introduces an efficient Grounded Perceiver to better utilize mask-based intermediate guidance for improved manipulation performance.

Large Vision-Language Models. Large Vision-Language Models (LVLMs) have recently demonstrated remarkable capabilities in image understanding and image-to-text generation, as seen in BLIP-2 [22], LLaVA [23], InstructBLIP [9], and MiniGPT-4 [51]. While these models excel at text generation and holistic image comprehension, they often struggle with fine-grained understanding of local regions within images. To address this, several methods [7, 29, 41, 46] enhance region-level perception by incorporating bounding boxes and spatial location bins, allowing LMMs to focus on specific interactive areas. Furthermore, recent approaches [21, 24, 32, 44, 48] have advanced grounding capabilities, enabling the identification of target objects or regions to support downstream tasks. Our method builds upon this progress by leveraging GLaMM [32] to generate grounding masks for both target objects and placement areas, providing structured guidance for the low-level policy network in robotic manipulation.

3. Data Generation

Existing language-conditioned simulation datasets often suffer from limited object and environment diversity [18, 27, 34] or lack a broad range of instructions and complex scenes [26, 28], making them less effective for train-

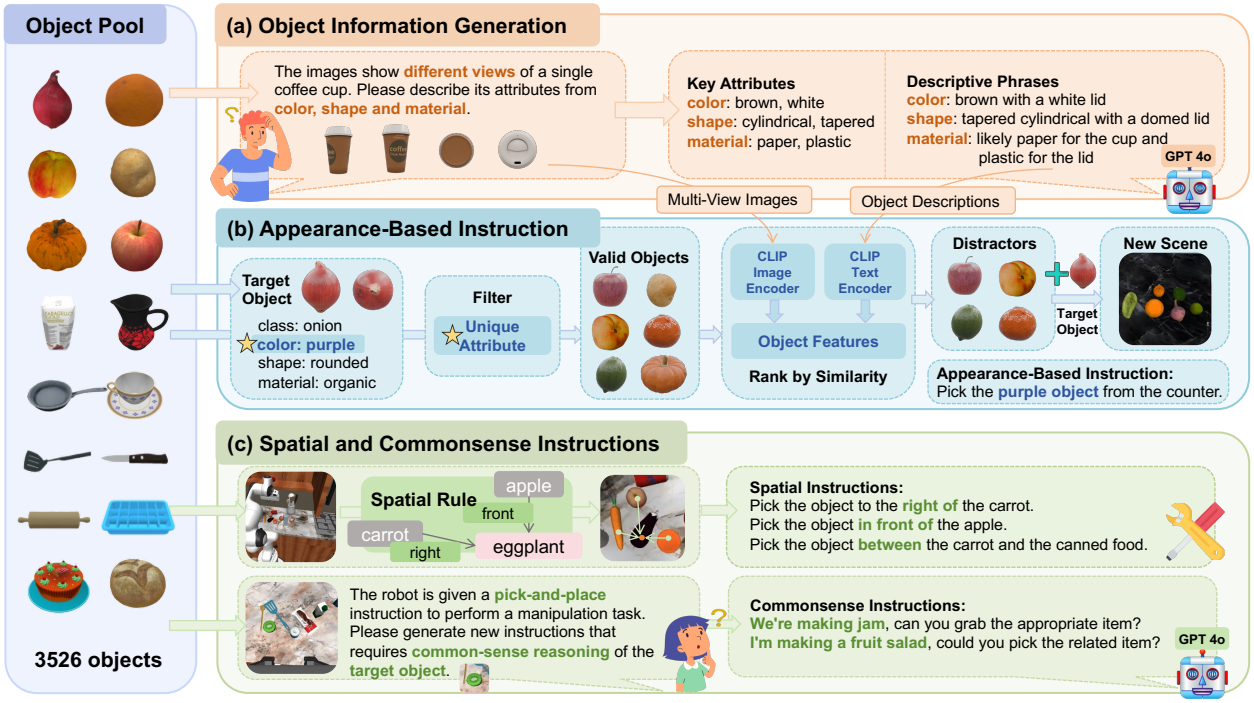

Figure 2. **Data Generation Pipeline.** The pipeline is composed of three key stages: (a) First, we extract informative object attributes in both keyword and descriptive phrase formats; (b) Next, appearance-based instructions are generated using these attributes, where keywords filter objects and descriptive phrases calculate appearance similarity; (c) Finally, spatial and commonsense instructions are generated through rule-based methods and GPT-generated techniques, respectively.

ing robust robotic policies and evaluating model generalization. To address these limitations, we introduce an automated data generation pipeline built upon RoboCasa [28], designed to enhance object variety, instruction diversity, and scene complexity for more challenging manipulation tasks. Our approach produces 24K new demonstrations paired with 112K diverse instructions, covering 176 distinct object categories and a total of 3,526 unique objects.

3.1. Object Set

A diverse object set is essential for increasing scene complexity and enabling varied instruction generation. Expanding on the original RoboCasa dataset, we curated additional 1,017 high-quality tabletop manipulation objects from Objaverse [11], selected from a pool of over 760K items. The initial filtering process, supported by GPT-4 [1], leveraged object tags and descriptions to identify (1) tabletop-suitable items, (2) kitchen-related objects, and (3) exclude multi-item collections. GPT-4 further assisted in categorizing objects based on their characteristics. The final selection underwent manual review to ensure quality. By incorporating distractor objects from this extensive set, we create complex scenes that challenge models to accurately interpret instructions in diverse and dynamic environments.

3.2. Diverse Instructions

Existing language-conditioned datasets often rely on fixed-format instructions that require minimal reasoning, leading models to depend on simple mappings between instructions and tasks rather than developing a deeper understanding of the language's semantics. This can hinder the model's ability to generalize to novel objects or tasks. To address this limitation, we propose designing instructions that focus on discriminative understanding of object appearance, spatial relationships, and commonsense knowledge, as illustrated in Figure 2. We present a target object alongside up to 10 contextually relevant objects (distractors) in the scene, encouraging the model to make precise distinctions between the target and surrounding objects.

Appearance-Based Instructions. To enable detailed visual understanding, we render each object from four perspectives (front, back, left, and right) and combine these views into a composite image that offers a comprehensive visual overview. This image is then processed by GPT-4, which extracts key attributes for each object—such as class, color, shape, and material—both as keywords and full descriptive phrases. When selecting a target object, we randomly assign one attribute as its unique characteristic within the scene. Objects with overlapping keywords for this attribute are filtered out, resulting in a refined set

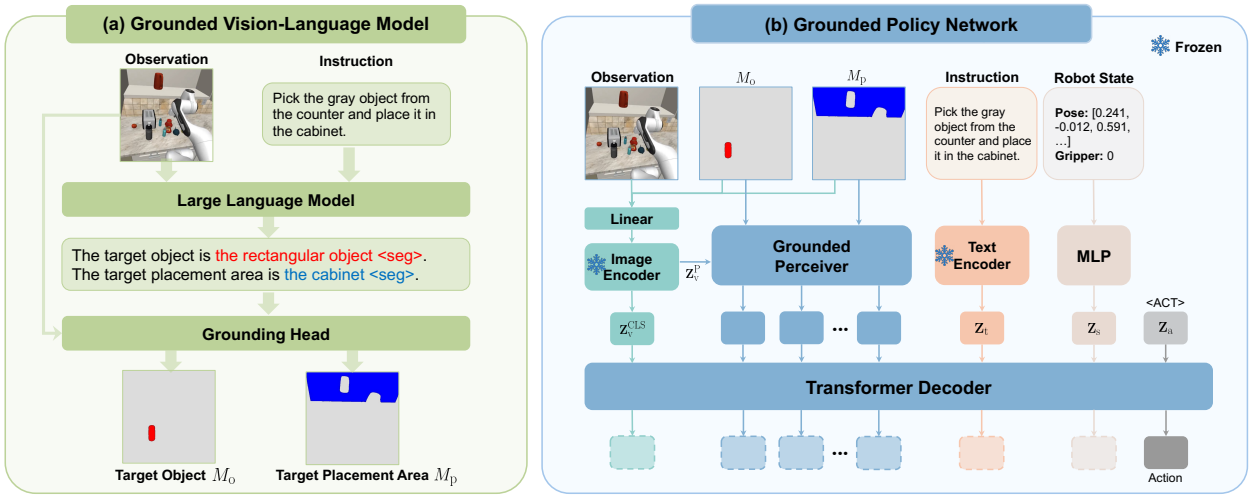

Figure 3. **Overall Architecture of RoboGround.** To enhance policy generalization, we leverage grounding masks as intermediate representations for spatial guidance. Specifically, (a) The grounded vision-language model processes the instruction and image observation to generate target masks. (b) The grounded policy network integrates mask guidance by concatenating masks with the image input and directing attention within the grounded perceiver.

of valid objects. From this set, distractors are sampled by ranking them based on the similarity of their textual and visual embeddings, derived from CLIP [31] text encoder and image encoder, encoded using the full attribute descriptions and multi-view images, respectively.

Appearance-Based Instructions. To enable detailed visual understanding, we render each object from four perspectives (front, back, left, and right) and combine these views into a composite image that provides a comprehensive visual overview. This image is then processed by GPT-4, which extracts key attributes for each object—such as class, color, shape, and material—both as keywords and full descriptive phrases. When selecting a target object, we randomly assign one attribute as its unique characteristic within the scene. Objects with overlapping keywords for this attribute are filtered out, resulting in a refined set of valid objects. From this set, distractors are sampled by ranking them based on the similarity of their textual and visual embeddings, derived from the CLIP [31] text and image encoders, using the full attribute descriptions and multi-view images, respectively.

Spatial Instructions. For randomly generated scenarios specifying a target object and its desired location, we employ a rule-based approach to create instructions that require spatial understanding. First, we extract the coordinates of all objects in the robot's frame. We then generate spatial instructions by identifying the object(s) closest to the target object in a specific direction, ensuring that no other objects of the same type are present in the scene and that the angular deviation in other directions does not exceed 30 degrees. This method produces clear and unambiguous spatial instructions.

Commonsense Instructions. We leverage GPT-4 to generate diverse instructions grounded in commonsense reasoning. For each task, we provide images from three perspectives (robot's left view, right view, and hand view), along with details about the target object and its intended position. Instead of explicitly mentioning the target object, GPT-4 generates pick-and-place task instructions framed within a familiar daily scenario. We prompt GPT-4 to ensure that the scenario is uniquely tailored to the target object, reinforcing the need for commonsense understanding.

4. Method

4.1. Overview

Our motivation is to leverage grounding masks as an intermediate representation to enhance the generalization ability of robotic manipulation policies. This requires a large-scale, pre-trained vision-language model capable of generating grounded information about target objects and placement areas, referred to as the Grounded Vision-Language Model, discussed in Section 4.2. We then incorporate this grounding knowledge into the low-level policy network, where the grounded masks function as both an attention mechanism within the Grounded Perceiver and an additional input channel for vision data, as detailed in Section 4.3. Finally, we outline the training and evaluation procedures for the complete framework in Section 4.4.

4.2. Grounded Vision-Language Model

Recent advances in large vision-language models [21, 32, 44, 48] have enabled generalizable, object-centric ground-

ing. Our approach builds on GLaMM [32], a state-of-the-art model that generates pixel-level segmentation masks, to guide the low-level policy network in accurately localizing target objects and placement areas.

Base Model. The grounded vision-language model takes an image observation and a language instruction as input and outputs binary masks for target objects and/or target placement areas specified by the instruction, as shown in Figure 3(a). Given an image x_{v} and a text instruction x_{t}, the image is encoded by the CLIP vision encoder [31] to obtain the visual feature $\mathbf{F}_{\mathrm{v}} = \mathrm{CLIP}(x_{\mathrm{v}})$. This feature is then projected into the LLM's embedding space via an MLP projector $f_{\mathrm{v}}(\cdot)$, yielding $f_{\mathrm{v}}(\mathbf{F}_{\mathrm{v}}) \in \mathbb{R}^{256\times D_{\mathrm{t}}}$. The LLM $\mathcal{L}(\cdot)$ integrates both the projected visual features and the text instruction to generate the output y_{t}:

$$y_{\mathrm{t}} = \mathcal{L}(f_{\mathrm{v}}(\mathrm{CLIP}(x_{\mathrm{v}})), x_{\mathrm{t}}). \tag{1}$$

The model perceives the image through a prompt formatted as: "`The <IMAGE> provides an overview of the picture`," where the `<IMAGE>` token is replaced by the projected visual feature, represented as a sequence of 256 tokens.

Pixel-level Grounding. To achieve pixel-level grounding, we employ a fine-grained image encoder $\mathcal{E}(\cdot)$ and a pixel decoder $\mathcal{D}(\cdot)$ within the grounding head. The encoder is initialized with a pre-trained SAM [20] encoder, while the decoder follows a SAM decoder-like architecture. A special token `<SEG>` is introduced into the LLM's vocabulary to extract grounding-related features. The last-layer embeddings $\mathbf{F}_{\mathrm{seg}}$ corresponding to the `<SEG>` token are projected into the decoder's feature space using a projector f_{s}. The binary segmentation mask M is then generated as follows:

$$M = \mathcal{D}\left(f_{\mathrm{s}}(\mathbf{F}_{\mathrm{seg}}), \mathcal{E}(x_{\mathrm{v}})\right). \tag{2}$$

For robotic manipulation tasks, the model is prompted to segment both the target object and the placement area (if applicable). As illustrated in Figure 3(a), the model's output specifies these regions using separate `<SEG>` tokens. Each `<SEG>` token is decoded into an individual mask, yielding the target object mask M_{o} and the placement area mask M_{p}, which are subsequently provided to the Grounded Policy Network.

4.3. Grounded Policy Network

After the grounded VLM translates the language instruction into masks for target objects and placement areas, these masks provide useful spatial guidance for the robot's policy. Rather than requiring explicit grounding of semantic descriptions, the policy can focus on leveraging this structured information to improve object localization and action execution. Given the strong generalization ability of the grounded VLM, the masks serve as a bridge between high-level language instructions and low-level manipulation strategies, helping the policy adapt to novel objects and environments more effectively.

Base Model. For the policy network, we employ a language-conditioned transformer architecture, following the GR-1 model [43]. As shown in Figure 3(b), this model processes a sequence of historical image observations, robot states and a language instruction as input to predict future robot actions. Specifically, for each input image x_{v}, along with corresponding masks M_o and M_p, we apply channel concatenation and project the combined channels back to a size of 3 using a linear layer. The result is then fed into a pre-trained ViTMAE encoder [16]. The encoded visual feature $\mathbf{Z}_{\mathrm{v}} \in \mathbb{R}^{197\times D_{\mathrm{v}}}$ is computed as follows:

$$\mathbf{Z}_{\mathrm{v}} = \mathrm{ViTMAE}(\mathrm{Linear}(\mathrm{Concat}(x_{\mathrm{v}}, M_o, M_p))), \tag{3}$$

where D_{v} denotes the hidden dimension of the vision encoder. The encoded feature $\mathbf{Z}_{\mathrm{v}}$ consists of a global representation $\mathbf{Z}_{\mathrm{v}}^{\mathrm{CLS}} \in \mathbb{R}^{1\times D_{\mathrm{v}}}$, obtained from the `CLS` token, and a set of local patch representations $\mathbf{Z}_{\mathrm{v}}^{\mathrm{P}} \in \mathbb{R}^{196\times D_{\mathrm{v}}}$, corresponding to a `14×14` spatial grid. These patch features are then processed by the grounded perceiver, $\mathcal{P}(\cdot)$, which interacts with the grounding masks to reduce the number of tokens, ultimately producing the refined visual token features $\mathcal{P}(\mathbf{Z}_{\mathrm{v}}^{\mathrm{P}})$.

Additionally, the input language instruction x_{t} is encoded using the CLIP text encoder, yielding the text token feature $\mathbf{Z}_{\mathrm{t}}$. The input robot state x_{s} is projected through an MLP to obtain the state token feature $\mathbf{Z}_{\mathrm{s}}$. To enable action prediction, a learnable `ACT` token with feature $\mathbf{Z}_{\mathrm{a}}$ is also included. Consequently, the token sequence for a single timestep input is structured as follows:

$$\left(\mathbf{Z}_{\mathrm{v}}^{\mathrm{CLS}}, \mathcal{P}(\mathbf{Z}_{\mathrm{v}}^{\mathrm{P}}), \mathbf{Z}_{\mathrm{t}}, \mathbf{Z}_{\mathrm{s}}, \mathbf{Z}_{\mathrm{a}}\right). \tag{4}$$

For a history length of N, the complete token sequence for one forward pass is constructed by aggregating token features from the past N timesteps. This sequence is then processed by a transformer decoder, which predicts the next-step action tokens through the output `<ACT>` tokens.

Grounded Perceiver A standard perceiver model [17], used in frameworks like GR-1, functions as a token resampler, reducing the number of tokens derived from the initial visual features. This is achieved through iterative attention layers between a small set of learnable query tokens and the original visual features. However, this token resampling process may lead to information loss, potentially limiting policy learning by failing to capture critical details about the target objects and placement areas. To address this, we propose guiding attention toward regions defined by grounded masks, ensuring that essential information is preserved for effective manipulation.

The perceiver takes as input the `14×14` patch features, denoted as $\mathbf{Z}_{\mathrm{v}}^{\mathrm{P}}$, extracted from the vision encoder, along

with a set of learnable query tokens, $\mathbf{Q}_{\mathrm{g}} \in \mathbb{R}^{k \times D_p}$, where k is the number of query tokens and D_p is the token dimension. These initial queries, referred to as global queries, capture information from the entire image. To integrate grounded masks, we introduce two additional sets of query tokens: $\mathbf{Q}_{\mathrm{o}} \in \mathbb{R}^{k \times D_p}$ for the target object and $\mathbf{Q}_{\mathrm{p}} \in \mathbb{R}^{k \times D_p}$ for the target placement area. During each attention layer, these additional query tokens interact with the patch features, with attention guided by the respective masks, M_{o} and M_{p}, applied via a mask fill operation (see supplementary material for details). This process produces the final set of perceived visual features, $\mathcal{P}(\mathbf{Z}_{\mathrm{v}}^{\mathrm{P}})$, with a total token length of $3 \times k$.

4.4. Training and Inference

VLM Fine-tuning. Although GLaMM was pre-trained on a large-scale grounding dataset, it may not precisely follow the prompt to identify the target object and placement area from a given manipulation instruction in a zero-shot setting. To address this, we construct an instruction-following dataset based on the generated simulation data to fine-tune the VLM. The grounded VLM model is fine-tuned using two types of losses: auto-regressive cross-entropy loss for text generation, and a linear combination of per-pixel binary cross-entropy loss and DICE loss for segmentation.

Policy Training. In each forward pass, the policy network receives image observations, robot states over N consecutive timesteps, and the corresponding language instruction. Based on this input, the network generates N action tokens, which are processed through linear layers to predict arm and gripper actions. Since arm actions are continuous, we use Smooth-L1 loss $\mathcal{L}_{\mathrm{arm}}$ for optimization. For binary gripper actions, we apply Binary Cross Entropy (BCE) loss $\mathcal{L}_{\mathrm{gripper}}$. Thus, the total training loss for the policy network is: $\mathcal{L}_{\mathrm{total}} = \mathcal{L}_{\mathrm{arm}} + \mathcal{L}_{\mathrm{gripper}}$.

Inference. The inference process starts by prompting the grounded VLM to generate segmentation masks for the target objects and placement areas. These initial masks are provided to the grounded policy network to predict the next action tokens. To optimize inference time, segmentation masks are extracted from the grounded VLM only once, at the beginning of the episode. These masks are then used throughout the entire manipulation task as inputs to the policy network. This approach is well-suited for single-turn manipulation tasks, where the target object's position remains stable until the gripper interacts with it, and the placement area remains fixed. Thus, the initial masks provide consistent guidance for the policy network throughout the task.

For multi-view image observations, each view is processed independently by the grounded VLM to obtain the corresponding masks, allowing for parallel processing. In the grounded policy network, each image and its corresponding masks are separately passed through the vision encoder and grounded perceiver to generate visual token features. During each forward pass of the policy network, only the action for the next timestep is predicted.

5. Experiments

5.1. Simulation Setting

Our simulation environment is built upon RoboCasa [28], a large-scale framework offering an automated scene generation pipeline. We leverage this pipeline to create diverse scene layouts and textures for data collection. Our main experiments are conducted within the RoboCasa environment, which includes a range of manipulation configurations. We classify the original RoboCasa data as "Easy," due to its relatively simple scene complexity and straightforward instructions. To introduce more complexity, we generate challenging pick-and-place data that requires advanced semantic understanding and reasoning. The instructions are divided into three categories: "Appearance," "Spatial," and "Common-sense," each representing different types of task-specific knowledge. Additionally, we include fundamental manipulation skills, such as opening and closing doors or drawers, pressing buttons, turning levers, and twisting knobs, to further challenge the model. In these tasks, target masks (e.g., for a drawer handle) are also generated to guide the robot's policy in precise manipulation.

5.2. Main Results

To evaluate our approach, we compare it with three well-established, easy-to-implement methods as baselines. Each method is trained on the same dataset, which includes RoboCasa's 66K demonstrations for foundational skills, alongside our proposed dataset containing 24K demonstrations and 112K instructions with increased diversity and complexity. For evaluation, we generate 400 test samples for each instruction type within the pick-and-place task and for each fundamental skill. The results are presented in Table 1. Metrics for pick-and-place tasks are reported as "a / b", where a is the contact rate (%) and b is the success rate (%). The contact rate refers to the percentage of attempts where the gripper makes contact with the target object, while the success rate indicates the percentage of tasks completed successfully. To ensure a fair comparison, we standardize key settings across methods, including history length, input image views, and image sizes.

Baselines.

- *ACT* [50]: A transformer-based policy network introduced by ALOHA [50]. Trained as a conditional VAE, it performs action chunking when predicting future actions.
- *BC-Transformer* [25]: A standard behavior cloning policy built on a transformer architecture. We use the publicly available BC-Transformer implementation from

Table 1. **Performance Comparison on Simulated Tasks.** Metrics for pick-and-place tasks are reported as "a / b", where a is the contact rate (%) and b is the success rate (%). We also report the success rates for other fundamental skills, including door opening/closing, button pressing, lever turning and knob twisting.

Method	Pick and Place				Open/Close	Press	Turn/Twist
	Easy	Appearance	Spatial	Common-sense			
ACT [50]	47.3 / 18.3	18.5 / 3.8	17.5 / 3.5	15.3 / 2.8	32.0	30.8	26.5
BC-Transformer [25]	79.8 / 34.8	31.3 / 6.8	38.3 / 7.8	26.0 / 5.8	54.8	50.8	45.3
GR-1 [43]	85.3 / 42.8	49.5 / 13.8	54.5 / 16.3	43.0 / 11.5	66.8	58.8	49.5
Ours	89.0 / 43.3	78.5 / 30.5	81.0 / 33.5	76.3 / 30.0	72.0	69.3	54.5

Table 2. **Evaluation Results of Unseen Settings.** Unseen instance denotes evaluation on new objects belonging to classes present in the training data. In contrast, unseen class refers to evaluation on objects from entirely new classes that were not included in the training data.

Unseen Level	Mask	Easy	Appea.	Spatial	Comm.
Instance		71.0 / 35.5	38.0 / 11.5	38.8 / 11.5	30.8 / 7.8
Instance	✓	85.8 / 42.5	75.5 / 29.5	77.8 / 31.8	72.0 / 28.0
Class		53.0 / 17.5	27.5 / 5.3	30.3 / 6.3	28.0 / 5.8
Class	✓	79.5 / 27.3	68.5 / 14.3	69.5 / 14.8	67.3 / 13.8

RoboMimic, which integrates language input by encoding it with CLIP and combining it with vision representations through FiLM [30] layers.

- *GR-1* [43]: A GPT-style model designed for language-conditioned visual robotic manipulation. The original GR-1 model was pre-trained on a large video dataset and predicts both robot actions and future images. Since the pre-trained model is unavailable, we reproduce it here without large-scale pre-training or image prediction.

Analysis. Compared to baseline models, our method consistently outperforms across all tasks. ACT generally performs poorly due to the absence of language input, which is crucial for task-specific language comprehension and reasoning. While BC-Transformer and GR-1 benefit from language input, their performance on more challenging tasks remains relatively low. This limitation likely arises from design shortcomings, as these models encode language input as a single, global text feature, which is inadequate for the nuanced understanding and reasoning required by semantically rich instructions.

Our approach, which leverages grounding masks to guide the policy, shows substantial improvements in challenging tasks, highlighting the effectiveness of incorporating such masks. Interestingly, we observe a consistent gap between the success rate and the contact rate, with the latter being significantly higher. This discrepancy suggests that the model's grasping capability could be further refined. We attribute this to the diverse range of objects in the dataset, which makes learning accurate grasping poses more difficult. Integrating pose-prediction networks, such as AnyGrasp [13], could potentially help address this challenge. Nonetheless, the high contact rate demonstrates the effectiveness of our method.

5.3. Zero-shot Evaluation.

For zero-shot evaluation, we categorize unseen settings into two levels: unseen instance and unseen class. Unseen instances refer to evaluation on new objects within classes present in the training data, while unseen classes involve evaluation on objects from entirely new classes not included in the training data. We exclude 30 classes, comprising 597 objects, as unseen and filter out 1/4 of the objects from each of the remaining classes, resulting in a total of 1,335 unseen instances. For each unseen setting, we generate 400 test samples for each reasoning type within the pick-and-place task. The evaluation results are presented in Table 2.

Analysis. The experimental results on both unseen instances and unseen classes show that incorporating mask information significantly enhances the model's generalization to novel settings, while maintaining strong performance on previously seen tasks. This improved generalization is reflected in better performance across new data distributions. Notably, in more challenging scenarios, mask guidance achieves approximately 100% relative improvement over non-mask baselines, highlighting its crucial role in handling complex, unseen situations.

5.4. Ablation Study.

To accelerate our ablation study, we use a subset of the dataset, as full training and evaluation would require several days on 8 NVIDIA 4090 GPUs. This subset includes 3K original samples and 3K generated reasoning samples for a single pick-and-place scenario in RoboCasa. For evaluation, we generate 50 test samples per instruction type while maintaining the original diversity in scenes and objects to ensure representative and meaningful results.

Training Data and Mask. We perform ablation studies by training models with different datasets and mask configurations. The results, shown in Table 3, indicate that when trained solely on the original simple data, the model eas-

Table 3. **Ablation Study on Training Data and Grounded Masks.** "Ori. Data" refers to original data in RoboCasa, while "New Data" denotes our proposed data with diverse instructions. "Pred" indicates the use of predicted masks, whereas "GT" denotes ground-truth masks.

Ori. Data	New Data	Mask	Easy	Appea.	Spatial	Comm.
✓			80 / 38	26 / 6	20 / 4	26 / 4
	✓		78 / 34	42 / 12	52 / 16	40 / 12
✓	✓		90 / 60	46 / 16	54 / 20	38 / 12
✓		Pred	88 / 44	48 / 14	50 / 16	44 / 12
	✓	Pred	86 / 42	72 / 32	78 / 38	70 / 32
✓	✓	Pred	90 / 64	70 / 40	80 / 38	74 / 36
✓	✓	GT	96 / 68	80 / 48	82 / 44	80 / 42

Table 4. **Ablation Study on Modules for Incorporating Grounding Masks.** "Channel Concat." denotes whether to do the channel concatenation for the image and mask. "Grounded Perceiver" denotes whether to incorporate masks into the perceiver.

Channel Concat.	Grounded Perceiver	Easy	Appea.	Spatial	Comm.
		68 / 24	42 / 12	52 / 16	40 / 12
✓		78 / 36	60 / 26	68 / 30	62 / 26
	✓	80 / 36	56 / 22	64 / 28	64 / 30
✓	✓	86 / 42	72 / 32	78 / 38	70 / 32

ily fits the "Easy" set—even without a mask—but struggles with more challenging sets. This supports our hypothesis that the limited complexity of the original data restricts the model's generalization ability. In contrast, training on our new dataset highlights the critical role of mask guidance in improving generalization. Furthermore, combining both original and new data enhances performance, suggesting that scaling the dataset could further strengthen grounding-aware policies.

Modules for Incorporating Masks. We evaluate two methods for integrating grounding information: concatenating masks as an additional channel and using the grounded perceiver. As shown in Table 4, both approaches effectively incorporate grounding knowledge. While simple mask concatenation allows the model to utilize mask information, the grounded perceiver enables a more comprehensive exploitation of mask features, resulting in improved performance.

Grounding Representations. We evaluate different grounding representations and their impact on model performance. As shown in Table 5, models perform best when using masks as grounding representations. In contrast, lower-dimensional vector representations, such as points and bounding boxes, pose greater challenges for effective learning. Masks provide richer information by accurately capturing the shape and size of the target object, leading to modest improvements over bounding box representations.

Grounded VLM We fine-tune the VLM to improve its instruction-following capability by training it on both simulated and original VLM data. Specifically, we create an instruction-following dataset based on simulated data using the following prompt format: "`Given a robotic manipulation instruction: <Instruction>, identify the target object for manipulation and, if applicable, the target placement area.`" The response is formulated as "`The target object is xxx <SEG>`" to generate the target object mask, and "`The target placement area is xxx <SEG>`" if a placement area is specified.

Table 5. **Ablation Study on Grounding Representations.** "Point" and "Bbox" denote the center pixel point and the 2D bounding box extracted from predicted masks, respectively. "Low-dim" refers to using the point or bbox as a low-dimensional input, while "Image" indicates incorporating them as an additional channel in the image input.

Representation	Easy	Appea.	Spatial	Comm.
Point, Low-dim	72 / 28	44 / 12	56 / 18	42 / 14
Point, Image	76 / 32	60 / 26	68 / 32	64 / 28
Bbox, Low-dim	72 / 30	46 / 14	54 / 18	44 / 16
Bbox, Image	82 / 38	68 / 30	74 / 40	68 / 30
Mask	86 / 42	72 / 32	78 / 38	70 / 32

Table 6. **Ablation Study on Grounded VLM.** "Zero-shot" refers to the zero-shot evaluation of the grounded VLM. "Sim. data" and "VLM data" denotes the use of simulated grounding data and VLM data for fine-tuning.

Method	mIoU
Zero-shot	13.2
Fine-tuned w/ Sim. data	45.5
Fine-tuned w/ Sim. data & VLM data	48.2

For evaluation, we compute the mean Intersection over Union (mIoU) of the predicted simulation results, with a performance comparison shown in Table 6. The initial zero-shot evaluation uses a dependency parsing tool to identify target nouns in the instructions, which are then used to prompt the VLM to generate a mask for each noun. However, this approach leads to suboptimal results due to ambiguities introduced by the dependency parsing. Fine-tuning on simulation data alone significantly improves results but risks losing the knowledge embedded in the original VLM dataset. To mitigate this, we fine-tune the model using both simulated data and the VLM's original grounding data, thereby enhancing simulation performance while preserving the model's prior knowledge.

Acknowledgment

This work was supported by National Key R&D Program of China (2022ZD0162000).

References

[1] Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*, 2023. 3

[2] Suneel Belkhale, Tianli Ding, Ted Xiao, Pierre Sermanet, Quon Vuong, Jonathan Tompson, Yevgen Chebotar, Debidatta Dwibedi, and Dorsa Sadigh. Rt-h: Action hierarchies using language. *arXiv preprint arXiv:2403.01823*, 2024. 1, 2

[3] Homanga Bharadhwaj, Roozbeh Mottaghi, Abhinav Gupta, and Shubham Tulsiani. Track2act: Predicting point tracks from internet videos enables diverse zero-shot robot manipulation, 2024. 1, 2

[4] Kevin Black, Mitsuhiko Nakamoto, Pranav Atreya, Homer Walke, Chelsea Finn, Aviral Kumar, and Sergey Levine. Zero-shot robotic manipulation with pretrained image-editing diffusion models. *arXiv preprint arXiv:2310.10639*, 2023. 1, 2

[5] 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. 1

[6] Qingwen Bu, Jia Zeng, Li Chen, Yanchao Yang, Guyue Zhou, Junchi Yan, Ping Luo, Heming Cui, Yi Ma, and Hongyang Li. Closed-loop visuomotor control with generative expectation for robotic manipulation. *arXiv preprint arXiv:2409.09016*, 2024. 2

[7] Keqin Chen, Zhao Zhang, Weili Zeng, Richong Zhang, Feng Zhu, and Rui Zhao. Shikra: Unleashing multimodal llm's referential dialogue magic, 2023. 2

[8] 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*, page 02783649241273668, 2023. 1

[9] Wenliang Dai, Junnan Li, Dongxu Li, Anthony Meng Huat Tiong, Junqi Zhao, Weisheng Wang, Boyang Li, Pascale Fung, and Steven Hoi. Instructblip: Towards general-purpose vision-language models with instruction tuning, 2023. 2

[10] Murtaza Dalal, Ajay Mandlekar, Caelan Garrett, Ankur Handa, Ruslan Salakhutdinov, and Dieter Fox. Imitating task and motion planning with visuomotor transformers. *arXiv preprint arXiv:2305.16309*, 2023. 2

[11] Matt Deitke, Dustin Schwenk, Jordi Salvador, Luca Weihs, Oscar Michel, Eli VanderBilt, Ludwig Schmidt, Kiana Ehsani, Aniruddha Kembhavi, and Ali Farhadi. Objaverse: A universe of annotated 3d objects. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13142–13153, 2023. 3

[12] Ben Eisner, Harry Zhang, and David Held. Flowbot3d: Learning 3d articulation flow to manipulate articulated objects. *arXiv preprint arXiv:2205.04382*, 2022. 2

[13] Hao-Shu Fang, Chenxi Wang, Hongjie Fang, Minghao Gou, Jirong Liu, Hengxu Yan, Wenhai Liu, Yichen Xie, and Cewu Lu. Anygrasp: Robust and efficient grasp perception in spatial and temporal domains. *IEEE Transactions on Robotics*, 2023. 7, 4

[14] Kuan Fang, Fangchen Liu, Pieter Abbeel, and Sergey Levine. Moka: Open-world robotic manipulation through mark-based visual prompting. *Robotics: Science and Systems (RSS)*, 2024. 2

[15] Jiayuan Gu, Sean Kirmani, Paul Wohlhart, Yao Lu, Montserrat Gonzalez Arenas, Kanishka Rao, Wenhao Yu, Chuyuan Fu, Keerthana Gopalakrishnan, Zhuo Xu, et al. Rt-trajectory: Robotic task generalization via hindsight trajectory sketches. *arXiv preprint arXiv:2311.01977*, 2023. 1, 2

[16] 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*, pages 16000–16009, 2022. 5

[17] Andrew Jaegle, Felix Gimeno, Andy Brock, Oriol Vinyals, Andrew Zisserman, and Joao Carreira. Perceiver: General perception with iterative attention. In *International conference on machine learning*, pages 4651–4664. PMLR, 2021. 5

[18] Stephen James, Zicong Ma, David Rovick Arrojo, and Andrew J Davison. Rlbench: The robot learning benchmark & learning environment. *IEEE Robotics and Automation Letters*, 5(2):3019–3026, 2020. 2

[19] Moo Jin Kim, Karl Pertsch, Siddharth Karamcheti, Ted Xiao, Ashwin Balakrishna, Suraj Nair, Rafael Rafailov, Ethan Foster, Grace Lam, Pannag Sanketi, et al. Openvla: An open-source vision-language-action model. *arXiv preprint arXiv:2406.09246*, 2024. 1

[20] Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C. Berg, Wan-Yen Lo, Piotr Dollár, and Ross Girshick. Segment anything. *arXiv:2304.02643*, 2023. 5

[21] Xin Lai, Zhuotao Tian, Yukang Chen, Yanwei Li, Yuhui Yuan, Shu Liu, and Jiaya Jia. Lisa: Reasoning segmentation via large language model. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9579–9589, 2024. 2, 4

[22] Junnan Li, Dongxu Li, Silvio Savarese, and Steven Hoi. BLIP-2: bootstrapping language-image pre-training with frozen image encoders and large language models. In *ICML*, 2023. 2

[23] Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning, 2023. 1, 2

[24] Chuofan Ma, Yi Jiang, Jiannan Wu, Zehuan Yuan, and Xiaojuan Qi. Groma: Localized visual tokenization for grounding multimodal large language models. In *European Conference on Computer Vision*, pages 417–435. Springer, 2025. 2

[25] Ajay Mandlekar, Danfei Xu, Josiah Wong, Soroush Nasiriany, Chen Wang, Rohun Kulkarni, Li Fei-Fei, Silvio Savarese, Yuke Zhu, and Roberto Martín-Martín. What matters in learning from offline human demonstrations for robot manipulation. *arXiv preprint arXiv:2108.03298*, 2021. 1, 6, 7

[26] Ajay Mandlekar, Soroush Nasiriany, Bowen Wen, Iretiayo Akinola, Yashraj Narang, Linxi Fan, Yuke Zhu, and Dieter Fox. Mimicgen: A data generation system for scalable robot learning using human demonstrations. *arXiv preprint arXiv:2310.17596*, 2023. 2, 4

[27] Oier Mees, Lukas Hermann, Erick Rosete-Beas, and Wolfram Burgard. Calvin: A benchmark for language-conditioned policy learning for long-horizon robot manipulation tasks. *IEEE Robotics and Automation Letters*, 7(3): 7327–7334, 2022. 2

[28] Soroush Nasiriany, Abhiram Maddukuri, Lance Zhang, Adeet Parikh, Aaron Lo, Abhishek Joshi, Ajay Mandlekar, and Yuke Zhu. Robocasa: Large-scale simulation of everyday tasks for generalist robots. *arXiv preprint arXiv:2406.02523*, 2024. 2, 3, 6, 1

[29] Zhiliang Peng, Wenhui Wang, Li Dong, Yaru Hao, Shaohan Huang, Shuming Ma, and Furu Wei. Kosmos-2: Grounding multimodal large language models to the world. *ArXiv*, abs/2306, 2023. 2

[30] 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*, 2018. 7

[31] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PMLR, 2021. 4, 5

[32] Hanoona Rasheed, Muhammad Maaz, Sahal Shaji, Abdelrahman Shaker, Salman Khan, Hisham Cholakkal, Rao M Anwer, Eric Xing, Ming-Hsuan Yang, and Fahad S Khan. Glamm: Pixel grounding large multimodal model. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13009–13018, 2024. 1, 2, 4, 5

[33] Daniel Seita, Yufei Wang, Sarthak J Shetty, Edward Yao Li, Zackory Erickson, and David Held. Toolflownet: Robotic manipulation with tools via predicting tool flow from point clouds. In *Conference on Robot Learning*, pages 1038–1049. PMLR, 2023. 2

[34] Mohit Shridhar, Lucas Manuelli, and Dieter Fox. Cliport: What and where pathways for robotic manipulation. In *Conference on robot learning*, pages 894–906. PMLR, 2022. 2

[35] Austin Stone, Ted Xiao, Yao Lu, Keerthana Gopalakrishnan, Kuang-Huei Lee, Quan Vuong, Paul Wohlhart, Sean Kirmani, Brianna Zitkovich, Fei Xia, et al. Open-world object manipulation using pre-trained vision-language models. *arXiv preprint arXiv:2303.00905*, 2023. 2

[36] Priya Sundaresan, Suneel Belkhale, Dorsa Sadigh, and Jeannette Bohg. Kite: Keypoint-conditioned policies for semantic manipulation. *arXiv:2306.16605*, 2023. 2

[37] Priya Sundaresan, Quan Vuong, Jiayuan Gu, Peng Xu, Ted Xiao, Sean Kirmani, Tianhe Yu, Michael Stark, Ajinkya Jain, Karol Hausman, et al. Rt-sketch: Goal-conditioned imitation learning from hand-drawn sketches. 2024. 1, 2

[38] Yang Tian, Sizhe Yang, Jia Zeng, Ping Wang, Dahua Lin, Hao Dong, and Jiangmiao Pang. Predictive inverse dynamics models are scalable learners for robotic manipulation. *arXiv preprint arXiv:2412.15109*, 2024. 1

[39] Mel Vecerik, Carl Doersch, Yi Yang, Todor Davchev, Yusuf Aytar, Guangyao Zhou, Raia Hadsell, Lourdes Agapito, and Jon Scholz. Robotap: Tracking arbitrary points for few-shot visual imitation. In *2024 IEEE International Conference on Robotics and Automation (ICRA)*, pages 5397–5403. IEEE, 2024. 2

[40] Lirui Wang, Yiyang Ling, Zhecheng Yuan, Mohit Shridhar, Chen Bao, Yuzhe Qin, Bailin Wang, Huazhe Xu, and Xiaolong Wang. Gensim: Generating robotic simulation tasks via large language models. *arXiv preprint arXiv:2310.01361*, 2023. 2

[41] Weiyun Wang, Yiming Ren, Haowen Luo, Tiantong Li, Chenxiang Yan, Zhe Chen, Wenhai Wang, Qingyun Li, Lewei Lu, Xizhou Zhu, et al. The all-seeing project v2: Towards general relation comprehension of the open world. *arXiv preprint arXiv:2402.19474*, 2024. 2

[42] Chuan Wen, Xingyu Lin, John So, Kai Chen, Qi Dou, Yang Gao, and Pieter Abbeel. Any-point trajectory modeling for policy learning. *arXiv preprint arXiv:2401.00025*, 2023. 2

[43] Hongtao Wu, Ya Jing, Chilam Cheang, Guangzeng Chen, Jiafeng Xu, Xinghang Li, Minghuan Liu, Hang Li, and Tao Kong. Unleashing large-scale video generative pre-training for visual robot manipulation. *arXiv preprint arXiv:2312.13139*, 2023. 5, 7, 1

[44] Bin Xiao, Haiping Wu, Weijian Xu, Xiyang Dai, Houdong Hu, Yumao Lu, Michael Zeng, Ce Liu, and Lu Yuan. Florence-2: Advancing a unified representation for a variety of vision tasks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4818–4829, 2024. 2, 4

[45] Mengda Xu, Zhenjia Xu, Yinghao Xu, Cheng Chi, Gordon Wetzstein, Manuela Veloso, and Shuran Song. Flow as the cross-domain manipulation interface. In *8th Annual Conference on Robot Learning*, 2024. 2

[46] Haoxuan You, Haotian Zhang, Zhe Gan, Xianzhi Du, Bowen Zhang, Zirui Wang, Liangliang Cao, Shih-Fu Chang, and Yinfei Yang. Ferret: Refer and ground anything anywhere at any granularity. *arXiv preprint arXiv:2310.07704*, 2023. 2

[47] Chengbo Yuan, Chuan Wen, Tong Zhang, and Yang Gao. General flow as foundation affordance for scalable robot learning. *arXiv preprint arXiv:2401.11439*, 2024. 2

[48] Haotian Zhang, Haoxuan You, Philipp Dufter, Bowen Zhang, Chen Chen, Hong-You Chen, Tsu-Jui Fu, William Yang Wang, Shih-Fu Chang, Zhe Gan, et al. Ferret-v2: An improved baseline for referring and grounding with large language models. *arXiv preprint arXiv:2404.07973*, 2024. 2, 4

[49] Jesse Zhang, Karl Pertsch, Jiahui Zhang, and Joseph J Lim. Sprint: Scalable policy pre-training via language instruc-

tion relabeling. In *2024 IEEE International Conference on Robotics and Automation (ICRA)*, pages 9168–9175. IEEE, 2024. 1, 2

[50] Tony Z Zhao, Vikash Kumar, Sergey Levine, and Chelsea Finn. Learning fine-grained bimanual manipulation with low-cost hardware. *arXiv preprint arXiv:2304.13705*, 2023. 1, 6, 7

[51] Deyao Zhu, Jun Chen, Xiaoqian Shen, Xiang Li, and Mohamed Elhoseiny. Minigpt-4: Enhancing vision-language understanding with advanced large language models. *arXiv preprint arXiv:2304.10592*, 2023. 1, 2