2024 IEEE International Conference on Robotics and Automation (ICRA 2024)
May 13-17, 2024. Yokohama, Japan

Object-Centric Instruction Augmentation for Robotic Manipulation

Junjie Wen[1,*], Yichen Zhu[2,*], Minjie Zhu[1], Jinming Li[3], Zhiyuan Xu[2], Zhengping Che[2], Chaomin Shen[1,†], Yaxin Peng[3], Dong Liu[2], Feifei Feng[2], and Jian Tang[2,†]

***Abstract*— Humans interpret scenes by recognizing both the identities and positions of objects in their observations. For a robot to perform tasks such as "pick and place", understanding both what the objects are and where they are located is crucial. While the former has been extensively discussed in the literature that uses the large language model to enrich the text descriptions, the latter remains underexplored. In this work, we introduce the *Object-Centric Instruction Augmentation (OCI)* framework to augment highly semantic and information-dense language instruction with position cues. We utilize a Multi-modal Large Language Model (MLLM) to weave knowledge of object locations into natural language instruction, thus aiding the policy network in mastering actions for versatile manipulation. Additionally, we present a feature reuse mechanism to integrate the vision-language features from off-the-shelf pre-trained MLLM into policy networks. Through a series of simulated and real-world robotic tasks, we demonstrate that robotic manipulator imitation policies trained with our enhanced instructions outperform those relying solely on traditional language instructions.**

I. Introduction

A fundamental challenge in deciphering the intelligence of embodied agents lies in the representation and interpretation of the rich, continuous sensory data obtained from our environment. The "what-where" framework in cognitive science [1], [2] posits that our brain employs separate neural channels to encode two primary categories of information [3]. Specifically, the "what" pathway [4] identifies detailed attributes of objects, like their identity and characteristics (e.g., color and shape). In contrast, the "where" pathway [5] discerns spatial details, such as position, trajectory, and closeness. Modern advancements in the domain of object-centric representations in artificial intelligence are probing these concepts [6], with a growing emphasis on simultaneously training both the "what" and "where" facets within specific contexts.

The recent advancement of the Large Language Model (LLM) has attracted increasing attention, with many initiatives leveraging human-like language comprehension for robotics. It revolutionized the pipeline for embodied agents and resolved the challenge in the previous literature where human instruction can be hard to interpret by robots [7], [8]. However, these methodologies primarily focus on augmenting the text with more detailed instructions, i.e., task planning [9], [8], [10]. The presumption is that by learning from the input observation, the policy networks should comprehend the objects' position internally and store this information as the form of parameters, and subsequently generate a corresponding action trajectory. While this is a prevalent approach, it often demands a vast number of demonstrations for the policy network to genuinely understand an object's position and produce a valid trajectory.

Our work is built upon the recent development of Multi-modal Large Language Models (MLLM) to augment language instruction with object-centric information. The novelty of our framework is how we inform the robot with visual information via transferred language instruction that allows us to infer high-level features around affordances and intents.

To start, we aim to develop a position-aware MLLM capable of representing the desired objects' locations. Our investigation delves into two kinds of positions: absolute and relative. The absolute position signifies an object's exact location. Taking the user instruction "pick up the apple and place it on the plate" as an example, we can directly extract the bounding boxes of both the apple and the plate, then supplement the instruction with these bounding boxes. This provides the policy network with a clear understanding of object locations based on a first-person perspective. Furthermore, any distractors, i.e., objects not pertinent to the action, are excluded, sharpening the policy network's focus on the task at hand. On the other hand, relative positions place the robot at the world's center, and define the direction of target objects relative to the robot. Using the previous example, our system might indicate that the "apple is to the left" and the "plate is to the right". This mirrors the human-centric approach to perceiving object positions. These language augmentations can be perceived as visual cues presented in text, allowing the policy network to derive positional knowledge from natural language. This circumvents the challenges associated with learning about objects directly from images.

Moreover, considering the computational intensity of MLLM calculations, we have devised a method that repurposes MLLM features, marrying its vision-language insights with policy networks to enhance robotic manipulation performance. By caching the MLLM's pre-computed embedding at the onset of the inference phase, we enable policy networks to operate seamlessly and efficiently, leveraging MLLM knowledge without incurring high computational costs.

We assessed our proposed Object-Centric Instruction aug-

[1]School of Computer Science, East China Normal University, China {51255901019, 51255901028}@stu.ecnu.edu.cn, cmshen@cs.ecnu.edu.cn
[2]Midea Group, China {zhuyc25, xuzy70, chezp, liudong13, feifei.feng, tangjian22}@midea.com
[3]Department of Mathematics, School of Science, Shanghai University, China {ljm2022, yaxin.peng}@shu.edu.cn
*Equal contributions. This work was done during Junjie Wen, Minjie Zhu, and Jinming Li's internship at Midea Group.
†Corresponding authors: Chaomin Shen and Jian Tang.

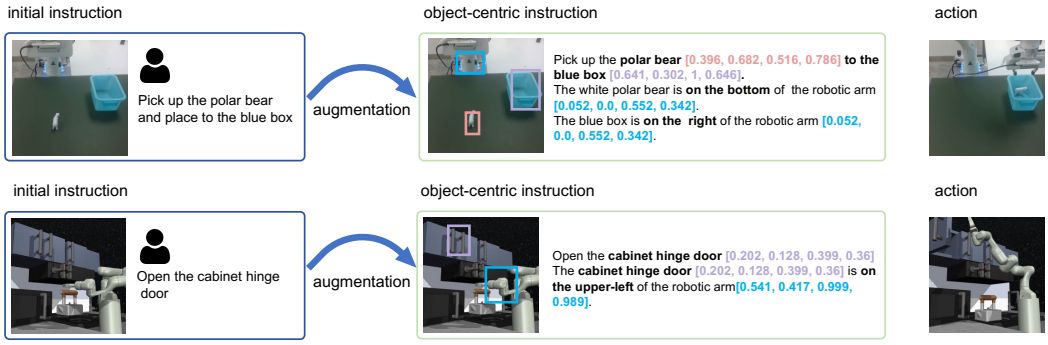

Fig. 1: Two examples of object-centric instruction augmentation for simulation and real robots, respectively. Given an initial instruction from the left figure, we augment them by providing the object's absolute position and relative position to the robotics and obtain the action eventually.

mentation (OCI) on unseen simulated and real-world robotic manipulation settings. Our findings suggest that OCI not only facilitates superior policy learning in contrast to flat instructions, but also enhances the understanding of the pivotal influence of an object's position on the success rate of manipulation. Through detailed ablation studies, we emphasize the importance of using pretrained models in "where" dimensions as vital for achieving remarkable enhancements.

To summarize, our contributions are the follows:

- We've established a pipeline that equips MLLM with position-aware knowledge, enabling it to augment incoming instructions with object-centric information.
- We introduce a Feature Reuse Mechanism that incorporates feature embedding from MLLM into the policy networks.
- Empirical evaluations and ablations on both simulated and real robots validate the superiority of our framework over existing baselines.

II. Related Work

Natural Language Instruction for Robotic Manipulation. The rise of large foundation models, specifically the Large Language Model (LLM) and Vision-Language Models [11], [12], [13], [14], [15], [16], [17], has opened a new era for embodied agents to understand instruction and generalize to unseen domains. These works include task planning [18], [19], navigation [20], code generation for actions [21], multi-robot collaboration [22], human-robot interaction [23], etc,. Another line of work utilize vision-language models for robotics [24]. Some of them use pre-trained VLM as instruction encoder [25], [26], [27], [28], [29], [30], [31] or visual encoder [32], [33], [34], [15], and others for visual state representations [35], object navigation [36], [37], high-level planning [24], [9], [38], [39], or providing supervision or success detection [40], [10], [41], [42], and open-vocabulary object localization [37]. RT-2 [43] develop an end-to-end framework that outputs actions with images and instructions directly.

Unlike prior applications of foundational models to downstream tasks like planning and navigation, our goal is to enhance language instructions to boost the generalizability of robotic manipulation, emphasizing positional cues for objects in text structure.

Object-Centric Representation Learning. The fields of robotics and vision have delved deeply into the use of object-centric representations, recognizing their value in facilitating modular reasoning about visual scenes. Within the robotics domain, it is common to use poses [44], [45] and bounding boxes [46], [47] to represent objects presented in a scene. However, these methods are typically limited to predefined object categories or specific instances. The development of the vision foundation model, i.e., SAM [48], motivates the researcher to extend the object-centric learning [49], [50] to unseen objects. Some works leverage unsupervised learning [51], [52] approaches to endow the manipulation policies with object awareness [53], [54], but their works are limited to simulation. Unlike prior works, we present a novel approach that translates object positions into natural language descriptions, enhancing visual representations for more effective policy learning in manipulation tasks. This method aligns with recent trends where LLM are integrated with vision systems [55], [56], resulting in enriched scene comprehension through a multi-modal approach.

III. Methodology

This section discusses how to chain what and where into a unified and useful instruction for manipulation policy learning. There are two key components to our OCI framework: 1) a fine-tuned MLLM that is adept at comprehending language and the environment, with the ability to correlate an object's location to its identity, and 2) a feature reuse mechanism that leverages the features embedding from MLLM to improve policy learning. The following section will address these two challenges in detail.

A. Position-Aware Multi-modal Large Language Models

Finetune Position-Aware MLLM. We use the pretrained weights [55] that were trained on a combined dataset of Conceptual Caption, SBU [57], and LAION [58]. The entire pre-training stage undergoes 20,000 training steps with a batch size of 256, covering approximately 5 million image-text pairs. To enhance its visual comprehension of text queries,

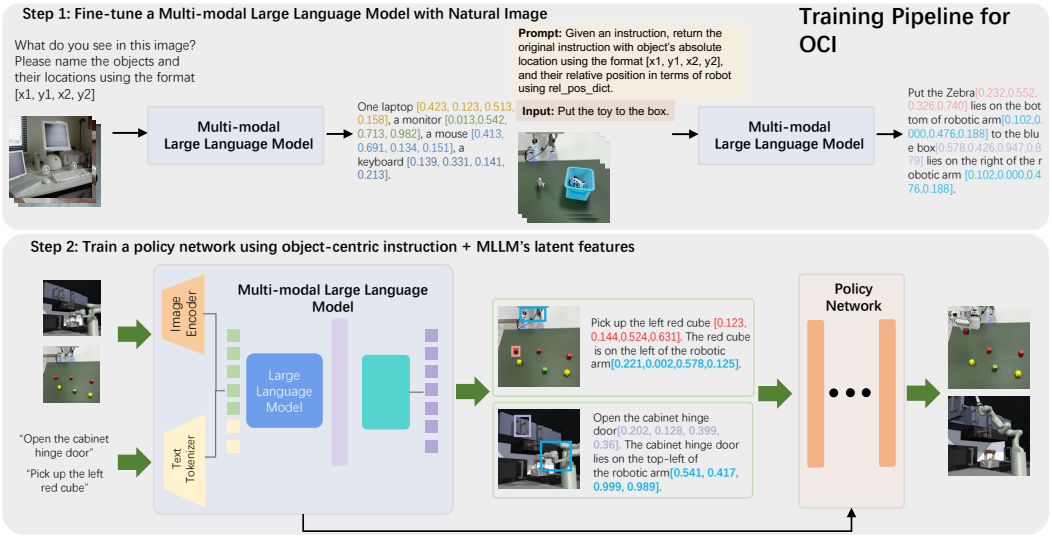

Fig. 2: An overview of the training process in OCI. We first fine-tune a MLLM with general detection datasets and our collected datasets. The fine-tuned MLLM enables automatically augmenting instruction object-centric information. Subsequently, this is integrated with a policy network to develop a model capable of generating specific actions.

we fine-tuned the pre-trained models using the annotated LLaVA-Instruct-150K dataset [56]. While the VLM currently demonstrates competent vision-language integration at the image level, this is inadequate for our purposes. We aim for scene comprehension that transcends mere image-level analysis, emphasizing object-centric representation from a holistic standpoint.

To facilitate the MLLM with object-level understanding and comprehend objects in terms of location, we use referring detection datasets such as RefCOCO, RefCOCO+, and RefCOCOg [59] to fine-tune the model. Notice that the RefCOCO datasets contain object's bounding boxes, accompanied by questions and answers. These datasets identify bounding boxes of mentioned objects within given instructions. This closely aligns with our use-case, which requires understanding the absolute position of objects.

Now, the MLLM can return the bounding boxes of referred objects but still cannot re-write the given instruction. We further make an instructing tuning dataset that allows the model to know how to rewrite the incoming command into our desired format, which preserves the instruction, adds bounding boxes to the target objects, and provides the relative position of the objects. In particular, we formulate the format of prompt, input, and output. We use a set of data that contains these three components, along with the image for the current scene, to fine-tune the MLLM. We collected 200 number of high-quality data for this fine-tuning step. Each image is collected with a random setup on the table. The initial input language is manually set, and we use the GPT-3.5-turbo [60] to enrich the instruction format. We use o_1 and o_2 to represent two objects for simplicity. In practice, we use a concrete object's name. We ask the GPT to "Use three different expressions to rewrite the instruction of 'pick up o_1 and place to o_2'.", it returns "Grab o_1 and set it on o_2.", "Take o_1 and position it on o_2.", and "Lift o_1 and put it onto o_2". In our experiment, each instruction is augmented with ten more different expressions, which are then used for the training.

Representation of Position. We represent the absolute position of objects using bounding boxes. These positions are articulated with numerical values, expressed in a manner that's both intuitive and aligned with natural language conventions. A bounding box is characterized by the format $[x_{\min}, y_{\min}, x_{\max}, y_{\max}]$. Here, both x and y are normalized in relation to the image's dimensions. By default, coordinates are presented with a precision of three decimal places. Such coordinates can appear at any point within the model's input or output sequence.

While bounding boxes define the absolute position of objects, a description of their relative positioning remains essential. To articulate the relative position of an object, a reference point is necessary. In the context of robotic manipulation, understanding movement in relation to the robot's own position is pivotal. As such, we designate the robot as the central reference in the image and detail objects' positions relative to it. For this purpose, we provide eight directional options: (a) Left, (b) Right, (c) Top, (d) Bottom, (e) Upper-left, (f) Upper-right, (g) Bottom-left, (h) Bottom-right.

Overall, when providing linguistic instructions to the robot, our pipeline facilitates object-centric commands. These commands guide the policy network with specific positional expressions. For instance, as shown in Figure 1, when the user instructs the robot to "pick up the polar bear and place it in the blue box", the following instructions with coordinates are given: "Pick up the polar bear [0.396, 0.682, 0.516, 0.786] to the blue box [0.641, 0.302, 1, 0.646]. The white polar bear is on the bottom of the robotic arm [0.052,

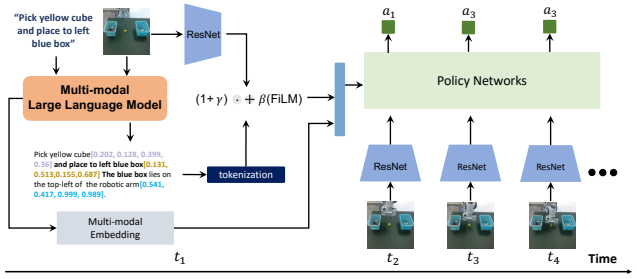

Fig. 3: This figure illustrates how we connect the policy network with MLLM. During the fine-tuning phase, MLLM is only utilized at the initial time step t_1, and its parameters are frozen.

0.0, 0.552, 0.342]. The blue box is on the right of the robotic arm [0.052, 0.0, 0.552, 0.342]." While preserving the original language of the user's request, our system provides bounding boxes for identified objects in the scene. Additionally, with the robot positioned at the center of the image, we detail the relative positions of the referred objects, highlighting that the polar bear is below the robot and the blue box is to its right. The overview can be found at the top of Figure 2.

Architecture. We have selected the pre-trained ViT-L/14 [61] from CLIP [12] as our visual encoder and adopted LLaMA-2-7B [62], [63] as our LLM. To ensure modal alignment and provide the appropriate input dimension for the LLM, a fully connected layer is implemented to transform the ViT's 16×16 output embedding $V \in \mathbb{R}^{16\times16\times1024}$ to $V^{'} \in \mathbb{R}^{256\times4096}$. We tap into the robust vision-language capabilities inherent to the text-image alignment [55]. Additionally, we fine-tune the associated networks while keeping the language and visual embeddings static, with only the alignment layers being adjustable.

B. Feature Reuse Mechanism

This section gives a detailed description of how we reuse the features from MLLM to improve the generalization of manipulation policy learning.

Policy networks. To formulate an efficient multi-task robot policy, we employ policy networks designed with a multi-task decoder architecture. Concretely, our objective is to learn a robot policy represented by $\pi(a_t|P, H)$, where $H := \{o_1, a_1, o_2, a_2, \cdots, o_t\}$ captures the historical trajectory of past interactions. Within this framework, the $o_t \in \mathbb{O}$ and $a_t \in \mathbb{A}$ respectively represent observations and actions taken at each interaction step. These policy networks ingest multi-modal tokens. For encoding, we deploy multi-modal prompts: the image undergoes processing via a vision backbone and is subsequently combined with the tokenized, augmented instruction using straightforward concatenation. The next step for the policy networks is to delineate the action space. Our policy network consists of three MLP layers with ReLU non-linear activation.

Feature Reuse Method. In our setup, we primarily retrieve the augmented instruction at the first frame and then abandon the model. Yet, discarding the computations made within the MLLM once a new instruction is acquired is inefficient. This is because the MLLM provides not just vision-language comprehension of the original instruction but also intuitively recognizes an object's identity and position. Finding a method to harness this wealth of information is both challenging and rewarding.

To implement this idea, we architected a feature reuse mechanism. In specific, we denote the feature embedding by the final layers of the MLLM as $E_{mllm} \in \mathbb{R}^D$ and define the multi-modal token (fusion of image tokens and text tokens) as $M \in \mathbb{R}^{D'}$, where D' and D are its corresponding dimensions. Herein, we add a sequence of operations with a LayerNorm-MLP (LN-MLP) [64] ($\mathbb{R}^D \rightarrow \mathbb{R}^{D'}$), such that $E^{'}_{mllm} = \text{LN-MLP}(E_{mllm} \in \mathbb{R}^{D'})$. The LN-MLP introduces nonlinearity and allows more flexible transformations for the E_{mllm}. Here, the $E^{'}_{mllm}$ carried prior knowledge on vision-language understanding. Therefore, it is paramount to align this prior vision-language information with the current instruction-observation pair. We use cross-attention to serve as a bridge to connect these two types of multi-modal embedding.

Particularly, M is projected into a query (Q) and key (K), and $E^{'}_{mllm}$ is projected into value (V). The keys K and values V are down-sampled to different sizes for different heads indexed by i. Thus, we formulate our multi-scale cross-attention (MSC) as $Q_i = E^{'}_{mllm}W^Q_i$, $K_i = MSC(M, r_i)W^K_i, V_i = MSC(M, r_i)W^V_i$, and $V_i = V_i + P(V_i)$. The $MSC(\cdot, r_i)$ is an MLP layer for aggregation in the i^{th} head with the down-sampling rate of r_i, and $P(\cdot)$ is a depth-wise convolutional layer for projection. Compared with the standard cross-attention, more fine-grained and low-level details that are beneficial to manipulation tasks are preserved. Finally, we calculated the attention tensor by:

$$h_i = Softmax(\frac{Q_i K^T_i}{\sqrt{d_h}} V_i) \tag{1}$$

where d_h is the dimension. Intuitively, the instruction-observation queries the useful knowledge in vision-language knowledge embedding and passes it into the policy network to improve the successful rate on unseen domains.

Note that the MLLM is frozen during the policy learning. At test time, we extract both the augmented instruction and feature embedding at the task's initial frame. These instructions and feature embedding are then stored for subsequent frame inferences, allowing the MLLM to be purged from the processor's memory. This optimizes computational speed by freeing up resources.

Overview of the Framework. In our context, the text includes not only natural language but also regions of interest that are represented by a set of floating points. Inputting regions of interest into the model includes various approaches, such as direct concatenation with cropped image patches [65], encoder-decoder structure for bounding boxes representation [66], [67], and utilization of Gaussia map [68], [69], [70]. In our experiment, we find it is sufficient to directly incorporate bounding boxes as a natural language

Fig. 4: The experimental results on Franka Kitchen. On all sub-tasks, our proposed OCI beats existing approaches, where our methods lead for a large margin on some tasks.

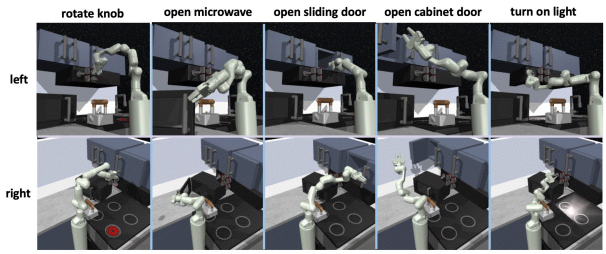

Fig. 5: The example of Franka Kitchen for five tasks on two camera views.

TABLE I: Ablation on two types of augmented instruction. Experiments are conducted on Franka Kitchen. "w/o" indicates "without". We report the average success rate over five tasks and two camera views per task.

Model	10 Demos	25 Demos
OCI (Ours)	61.7	69.4
w/o absolute position	54.2	63.5
w/o relative position	49.6	57.3

without additional pre-processing. Initially, a pre-trained image encoder processes each image. This is then tokenized, similar to RT-1 [28]. The language undergoes tokenization first, after which a T5-small model extracts features. It is then concatenated with the image token. A policy network is followed up to generate the action space. A comprehensive overview of this framework can be found in Figure 3.

IV. Experiments

Our experiments aim to answer the following questions: 1) Does our method enable better policy learning than using naive language instruction? 2) Is our method effective in real-world environments? We initiate our discussion by outlining the simulation environments tailored to address these queries. Subsequently, we present in-depth experimental results that positively affirm answers to both questions.

A. Simulation Experiments

Experiments Setup. Franka Kitchen benchmark focuses on tasks like sliding open the right door, opening the cabinet, turning on the light, turning the stovetop knob, and opening the microwave. The example of each task is

TABLE II: Ablation on reusable feature mechanism. Experiments are conducted on Franka Kitchen. "w/o" indicates "without". We report the average success rate over five tasks and two camera views per task.

Model	10 Demos	25 Demos
OCI (Ours)	61.7	69.4
w/o reusable features	53.9	62.3

present in Figure 5. For Franka Kitchen tasks, the length of a demonstration is 50, which contains 50 state-action pairs. For MLLM, we fixed the weights when training the policy networks. There are two views in the Franka Kitchen simulator; each is conducted three times, and the average success rate over both views is reported. All Franka tasks include proprioceptive data of the arm joint and gripper positions. The horizon for all Franka tasks is 50 steps, and our imitation experiments use either 10 or 25 demos.

Baselines. We compare our model with R3M [71], which is the state-of-the-art method and widely applicable method in Franka Kitchen. We also compare with BLIP-2 [65], a SOTA vision-language model. We replace our MLLM with BLIP-2 in our method and retain the FRM. In all experiments, the policy network is learned using few-shot learning on a small amount of demonstration data. There are two settings, one of which utilizes 25 demonstrations, and the other utilizes ten demonstrations. We report the success rate in five tasks per benchmark and two different camera views for each set respectively. Each experiment is run five times, and we report the average performance.

Main Experimental Results. We demonstrate the experimental results in Figure 4. It is obvious that for all tasks on two different camera views, our approach achieves superior performance over both R3M and BLIP-2. On some difficult tasks, such as opening the cabinet, opening the microwave, and turning the stovetop knob, the performance gap between our proposed OCI is even larger compared to R3M and BILP-2. Compared to BILP-2, which also uses a large-scale pre-trained vision-language model to align the instruction and observation, then map to the policy network, our OCI achieves stronger. performance over five tasks, showing the effectiveness of the object-centric instruction augmentation.

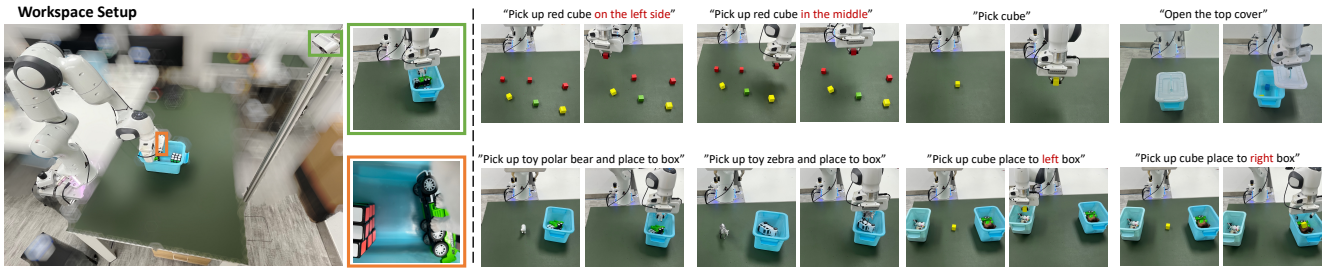

Fig. 6: **Left:** The setup of our Franka real robot. **Right:** The example of some tasks that we collected.

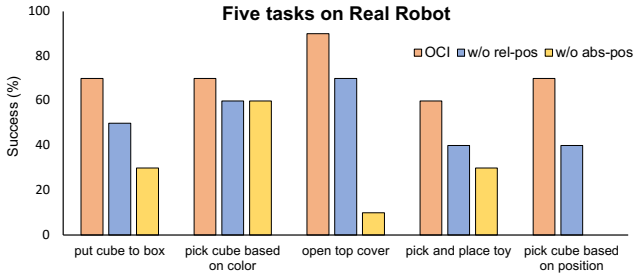

Fig. 7: Results from real-world experiments indicate that both relative and absolute positions are crucial for the successful execution of manipulation tasks.

B. Ablation Study

How Important for Absolute and Relative Position in Instruction? We evaluate the utility of instruction augmentation and present our findings in Table I. Our experiments indicate that both relative and absolute positions are critical, confirming the efficacy of our framework. We conduct another ablation study is conducted on real-world experiments.

The Effectiveness of Feature Reuse Mechanism. We explore the efficacy of our proposed FRM. The experimental results are detailed in Table II. Observing the data, we find that omitting the FRM results in an average drop of 7.1% in the success rate, highlighting the importance of FRM.

C. Real-world Experiments

Given our encouraging simulation experiments, it is natural to ask whether our algorithm can work on real-world robotic manipulation tasks, which present the additional challenges of noisy image observations from imperfect camera sensors and increased object quantity and diversity.

We use the Franka robot with a 7-DOF arm, which is equipped with a parallel jaw gripper (see Figure 6, left). Our workspace uses two D435i RealSense RGBD cameras. We only use the RGB information in our experiments. One egocentric camera is attached to the robot's hand, and one exocentric camera is positioned at the robot's front.

Experiment Setup. To realize the stated challenges above, we design a real-world environment (referred to as Real-Robot), in which a Franka robot is tasked with 1) picking up a cube to either left or right box, 2) picking up a cube based on the color, 3) open the top cover of a box, 4) pick up a toy and place to the box, 5) pick up a cube based on its position (left, middle, or right). We provide a sample of the data we gathered from real-world trajectories in Figure 6 (right). The tasks are numerically represented for simplicity; for example, Task 1 corresponds to "picking up a cube and placing it in either the left or right box".

Experimental Results. An evaluation of our approach was conducted using a real robot setup, with each task repeated in ten trials. Additionally, we conducted an ablation study, which is presented in Figure 7. In this study, we progressively removed the relative position (denoted as "w/o rel-pos") and the absolute position (denoted as "w/o abs-pos"). Across all five tasks, our OCI consistently outperformed the baselines, achieving the highest success rates. Particularly notable were the results for Task 1 and Task 5. In these tasks, the robot needed to discern direction, such as picking up the cube from the left, right, or middle side or placing the cube on the left or right box. Plain language instructions struggled with Task 5, failing entirely, and achieved a mere 30% success rate for Task 1. However, introducing absolute position information boosted success rates by 40% and 20% for the two tasks, respectively. Incorporating relative position data further enhanced performance by 20% and 30%, underscoring the effectiveness of our OCI.

V. Conclusion

This work contributes a novel perspective on language instruction for robotics manipulation. Motivated by the concept of visual understanding in human intelligence, we augment language instruction by adding an object's absolute and relative position into the text format. Such augmented language alleviates the burden of the visual encoder, which was previously responsible for localizing objects on its own. We conduct experiments on both simulation and real-world scenarios and show the superior performance of our methods over conventional language instruction. We believe our approach presents a fresh perspective on the kinds of instructions best suited for versatile robotic manipulation.

Acknowledgment

This work was supported by the Large-scale Numerical Simulation Computing Sharing Platform of Shanghai University, and the Shanghai Science and Technology Innovation Action Plan under Grant 22511105400.

REFERENCES

[1] L. G. Ungerleider and J. V. Haxby, "'what'and 'where'in the human brain," *Current opinion in neurobiology*, vol. 4, no. 2, pp. 157–165, 1994.

[2] E. H. de Haan and A. Cowey, "On the usefulness of 'what'and 'where'pathways in vision," *Trends in cognitive sciences*, vol. 15, no. 10, pp. 460–466, 2011.

[3] M. N. Hebart and G. Hesselmann, "What visual information is processed in the human dorsal stream?" *Journal of Neuroscience*, vol. 32, no. 24, pp. 8107–8109, 2012.

[4] M. A. Goodale and A. D. Milner, "Separate visual pathways for perception and action," *Trends in neurosciences*, vol. 15, no. 1, pp. 20–25, 1992.

[5] B. R. Sheth and R. Young, "Two visual pathways in primates based on sampling of space: exploitation and exploration of visual information," *Frontiers in integrative neuroscience*, vol. 10, p. 37, 2016.

[6] E. Freud, J. C. Culham, D. C. Plaut, and M. Behrmann, "The large-scale organization of shape processing in the ventral and dorsal pathways," *elife*, vol. 6, p. e27576, 2017.

[7] S. Vemprala, R. Bonatti, A. Bucker, and A. Kapoor, "Chatgpt for robotics: Design principles and model abilities," *Microsoft Auton. Syst. Robot. Res*, vol. 2, p. 20, 2023.

[8] M. Ahn, A. Brohan, N. Brown, Y. Chebotar, O. Cortes, B. David, C. Finn, C. Fu, K. Gopalakrishnan, K. Hausman *et al.*, "Do as i can, not as i say: Grounding language in robotic affordances," *arXiv preprint arXiv:2204.01691*, 2022.

[9] Y. Mu, Q. Zhang, M. Hu, W. Wang, M. Ding, J. Jin, B. Wang, J. Dai, Y. Qiao, and P. Luo, "Embodiedgpt: Vision-language pre-training via embodied chain of thought," *arXiv preprint arXiv:2305.15021*, 2023.

[10] T. Xiao, H. Chan, P. Sermanet, A. Wahid, A. Brohan, K. Hausman, S. Levine, and J. Tompson, "Robotic skill acquisition via instruction augmentation with vision-language models," *arXiv preprint arXiv:2211.11736*, 2022.

[11] J. Wei, X. Wang, D. Schuurmans, M. Bosma, F. Xia, E. Chi, Q. V. Le, D. Zhou *et al.*, "Chain-of-thought prompting elicits reasoning in large language models," *Advances in Neural Information Processing Systems*, vol. 35, pp. 24 824–24 837, 2022.

[12] A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark *et al.*, "Learning transferable visual models from natural language supervision," in *International conference on machine learning*. PMLR, 2021, pp. 8748–8763.

[13] T. Kojima, S. S. Gu, M. Reid, Y. Matsuo, and Y. Iwasawa, "Large language models are zero-shot reasoners," *Advances in neural information processing systems*, vol. 35, pp. 22 199–22 213, 2022.

[14] L. Fan, G. Wang, Y. Jiang, A. Mandlekar, Y. Yang, H. Zhu, A. Tang, D.-A. Huang, Y. Zhu, and A. Anandkumar, "Minedojo: Building open-ended embodied agents with internet-scale knowledge," *Advances in Neural Information Processing Systems*, vol. 35, pp. 18 343–18 362, 2022.

[15] M. Shridhar, L. Manuelli, and D. Fox, "Cliport: What and where pathways for robotic manipulation," in *Conference on Robot Learning*. PMLR, 2022, pp. 894–906.

[16] Y. Zhu, M. Zhu, N. Liu, Z. Ou, X. Mou, and J. Tang, "Llava-phi: Efficient multi-modal assistant with small language model," *arXiv preprint arXiv:2401.02330*, 2024.

[17] X. Liu, Y. Zhu, Y. Lan, C. Yang, and Y. Qiao, "Query-relevant images jailbreak large multi-modal models," *arXiv preprint arXiv:2311.17600*, 2023.

[18] W. Huang, P. Abbeel, D. Pathak, and I. Mordatch, "Language models as zero-shot planners: Extracting actionable knowledge for embodied agents," in *International Conference on Machine Learning*. PMLR, 2022, pp. 9118–9147.

[19] R. Ma, L. Lam, B. A. Spiegel, A. Ganeshan, R. Patel, B. Abbatematteo, D. Paulius, S. Tellex, and G. Konidaris, "Skill generalization with verbs."

[20] C. Huang, O. Mees, A. Zeng, and W. Burgard, "Visual language maps for robot navigation," in *2023 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2023, pp. 10 608–10 615.

[21] J. Liang, W. Huang, F. Xia, P. Xu, K. Hausman, B. Ichter, P. Florence, and A. Zeng, "Code as policies: Language model programs for embodied control," in *2023 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2023, pp. 9493–9500.

[22] Z. Mandi, S. Jain, and S. Song, "Roco: Dialectic multi-robot collaboration with large language models," *arXiv preprint arXiv:2307.04738*, 2023.

[23] Y. Cui, S. Karamcheti, R. Palleti, N. Shivakumar, P. Liang, and D. Sadigh, "No, to the right: Online language corrections for robotic manipulation via shared autonomy," in *Proceedings of the 2023 ACM/IEEE International Conference on Human-Robot Interaction*, 2023, pp. 93–101.

[24] D. Driess, F. Xia, M. S. Sajjadi, C. Lynch, A. Chowdhery, B. Ichter, A. Wahid, J. Tompson, Q. Vuong, T. Yu *et al.*, "Palm-e: An embodied multimodal language model," *arXiv preprint arXiv:2303.03378*, 2023.

[25] Y. Jiang, A. Gupta, Z. Zhang, G. Wang, Y. Dou, Y. Chen, L. Fei-Fei, A. Anandkumar, Y. Zhu, and L. Fan, "Vima: General robot manipulation with multimodal prompts," *arXiv preprint arXiv:2210.03094*, 2022.

[26] F. Hill, S. Mokra, N. Wong, and T. Harley, "Human instruction-following with deep reinforcement learning via transfer-learning from text," *arXiv preprint arXiv:2005.09382*, 2020.

[27] E. Jang, A. Irpan, M. Khansari, D. Kappler, F. Ebert, C. Lynch, S. Levine, and C. Finn, "Bc-z: Zero-shot task generalization with robotic imitation learning," in *Conference on Robot Learning*. PMLR, 2022, pp. 991–1002.

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

[29] S. Nair, E. Mitchell, K. Chen, S. Savarese, C. Finn *et al.*, "Learning language-conditioned robot behavior from offline data and crowd-sourced annotation," in *Conference on Robot Learning*. PMLR, 2022, pp. 1303–1315.

[30] C. Lynch and P. Sermanet, "Language conditioned imitation learning over unstructured data," *arXiv preprint arXiv:2005.07648*, 2020.

[31] M. Shridhar, L. Manuelli, and D. Fox, "Perceiver-actor: A multi-task transformer for robotic manipulation," in *Conference on Robot Learning*. PMLR, 2023, pp. 785–799.

[32] Y. J. Ma, S. Sodhani, D. Jayaraman, O. Bastani, V. Kumar, and A. Zhang, "Vip: Towards universal visual reward and representation via value-implicit pre-training," *arXiv preprint arXiv:2210.00030*, 2022.

[33] M. Laskin, K. Lee, A. Stooke, L. Pinto, P. Abbeel, and A. Srinivas, "Reinforcement learning with augmented data," *Advances in neural information processing systems*, vol. 33, pp. 19 884–19 895, 2020.

[34] R. Shah and V. Kumar, "Rrl: Resnet as representation for reinforcement learning," *arXiv preprint arXiv:2107.03380*, 2021.

[35] S. Karamcheti, S. Nair, A. S. Chen, T. Kollar, C. Finn, D. Sadigh, and P. Liang, "Language-driven representation learning for robotics," *arXiv preprint arXiv:2302.12766*, 2023.

[36] S. Y. Gadre, M. Wortsman, G. Ilharco, L. Schmidt, and S. Song, "Clip on wheels: Zero-shot object navigation as object localization and exploration," *arXiv preprint arXiv:2203.10421*, vol. 3, no. 4, p. 7, 2022.

[37] A. Stone, T. Xiao, Y. Lu, K. Gopalakrishnan, K.-H. Lee, Q. Vuong, P. Wohlhart, B. Zitkovich, F. Xia, C. Finn *et al.*, "Open-world object manipulation using pre-trained vision-language models," *arXiv preprint arXiv:2303.00905*, 2023.

[38] B. Chen, F. Xia, B. Ichter, K. Rao, K. Gopalakrishnan, M. S. Ryoo, A. Stone, and D. Kappler, "Open-vocabulary queryable scene representations for real world planning," in *2023 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2023, pp. 11 509–11 522.

[39] M. Zhu, Y. Zhu, J. Li, J. Wen, Z. Xu, Z. Che, C. Shen, Y. Peng, D. Liu, F. Feng *et al.*, "Language-conditioned robotic manipulation with fast and slow thinking," *arXiv preprint arXiv:2401.04181*, 2024.

[40] Y. Du, K. Konyushkova, M. Denil, A. Raju, J. Landon, F. Hill, N. de Freitas, and S. Cabi, "Vision-language models as success detectors," *arXiv preprint arXiv:2303.07280*, 2023.

[41] X. Zhang, Y. Ding, S. Amiri, H. Yang, A. Kaminski, C. Esselink, and S. Zhang, "Grounding classical task planners via vision-language models," *arXiv preprint arXiv:2304.08587*, 2023.

[42] T. Sumers, K. Marino, A. Ahuja, R. Fergus, and I. Dasgupta, "Distilling internet-scale vision-language models into embodied agents," *arXiv preprint arXiv:2301.12507*, 2023.

[43] A. Brohan, N. Brown, J. Carbajal, Y. Chebotar, X. Chen, K. Choromanski, T. Ding, D. Driess, A. Dubey, C. Finn *et al.*, "Rt-2: Vision-language-action models transfer web knowledge to robotic control," *arXiv preprint arXiv:2307.15818*, 2023.

[44] J. Tremblay, T. To, B. Sundaralingam, Y. Xiang, D. Fox, and S. Birch-

field, "Deep object pose estimation for semantic robotic grasping of household objects," *arXiv preprint arXiv:1809.10790*, 2018.

[45] S. Tyree, J. Tremblay, T. To, J. Cheng, T. Mosier, J. Smith, and S. Birchfield, "6-dof pose estimation of household objects for robotic manipulation: An accessible dataset and benchmark," in *2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2022, pp. 13 081–13 088.

[46] T. Migimatsu and J. Bohg, "Object-centric task and motion planning in dynamic environments," *IEEE Robotics and Automation Letters*, vol. 5, no. 2, pp. 844–851, 2020.

[47] D. Wang, C. Devin, Q.-Z. Cai, F. Yu, and T. Darrell, "Deep object-centric policies for autonomous driving," in *2019 International Conference on Robotics and Automation (ICRA)*. IEEE, 2019, pp. 8853–8859.

[48] A. Kirillov, E. Mintun, N. Ravi, H. Mao, C. Rolland, L. Gustafson, T. Xiao, S. Whitehead, A. C. Berg, W.-Y. Lo *et al.*, "Segment anything," *arXiv preprint arXiv:2304.02643*, 2023.

[49] Y. Zhu, Z. Jiang, P. Stone, and Y. Zhu, "Learning generalizable manipulation policies with object-centric 3d representations," in *7th Annual Conference on Robot Learning*, 2023.

[50] J. Shi, J. Qian, Y. J. Ma, and D. Jayaraman, "Plug-and-play object-centric representations from "what" and "where" foundation models."

[51] F. Locatello, D. Weissenborn, T. Unterthiner, A. Mahendran, G. Heigold, J. Uszkoreit, A. Dosovitskiy, and T. Kipf, "Object-centric learning with slot attention," *Advances in Neural Information Processing Systems*, vol. 33, pp. 11 525–11 538, 2020.

[52] C. P. Burgess, L. Matthey, N. Watters, R. Kabra, I. Higgins, M. Botvinick, and A. Lerchner, "Monet: Unsupervised scene decomposition and representation," *arXiv preprint arXiv:1901.11390*, 2019.

[53] C. Wang, R. Wang, A. Mandlekar, L. Fei-Fei, S. Savarese, and D. Xu, "Generalization through hand-eye coordination: An action space for learning spatially-invariant visuomotor control," in *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2021, pp. 8913–8920.

[54] N. Heravi, A. Wahid, C. Lynch, P. Florence, T. Armstrong, J. Tompson, P. Sermanet, J. Bohg, and D. Dwibedi, "Visuomotor control in multi-object scenes using object-aware representations," in *2023 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2023, pp. 9515–9522.

[55] D. Zhu, J. Chen, X. Shen, X. Li, and M. Elhoseiny, "Minigpt-4: Enhancing vision-language understanding with advanced large language models," *arXiv preprint arXiv:2304.10592*, 2023.

[56] H. Liu, C. Li, Q. Wu, and Y. J. Lee, "Visual instruction tuning," *arXiv preprint arXiv:2304.08485*, 2023.

[57] V. Ordonez, G. Kulkarni, and T. Berg, "Im2text: Describing images using 1 million captioned photographs," *Advances in neural information processing systems*, vol. 24, 2011.

[58] C. Schuhmann, R. Beaumont, R. Vencu, C. Gordon, R. Wightman, M. Cherti, T. Coombes, A. Katta, C. Mullis, M. Wortsman *et al.*, "Laion-5b: An open large-scale dataset for training next generation image-text models," *Advances in Neural Information Processing Systems*, vol. 35, pp. 25 278–25 294, 2022.

[59] L. Yu, P. Poirson, S. Yang, A. C. Berg, and T. L. Berg, "Modeling context in referring expressions," in *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11-14, 2016, Proceedings, Part II 14*. Springer, 2016, pp. 69–85.

[60] L. Ouyang, J. Wu, X. Jiang, D. Almeida, C. Wainwright, P. Mishkin, C. Zhang, S. Agarwal, K. Slama, A. Ray *et al.*, "Training language models to follow instructions with human feedback," *Advances in Neural Information Processing Systems*, vol. 35, pp. 27 730–27 744, 2022.

[61] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly *et al.*, "An image is worth 16x16 words: Transformers for image recognition at scale," *arXiv preprint arXiv:2010.11929*, 2020.

[62] H. Touvron, T. Lavril, G. Izacard, X. Martinet, M.-A. Lachaux, T. Lacroix, B. Rozière, N. Goyal, E. Hambro, F. Azhar *et al.*, "Llama: Open and efficient foundation language models," *arXiv preprint arXiv:2302.13971*, 2023.

[63] H. Touvron, L. Martin, K. Stone, P. Albert, A. Almahairi, Y. Babaei, N. Bashlykov, S. Batra, P. Bhargava, S. Bhosale *et al.*, "Llama 2: Open foundation and fine-tuned chat models," *arXiv preprint arXiv:2307.09288*, 2023.

[64] J. L. Ba, J. R. Kiros, and G. E. Hinton, "Layer normalization," *arXiv preprint arXiv:1607.06450*, 2016.

[65] L. Bracha, E. Shaar, A. Shamsian, E. Fetaya, and G. Chechik, "Disclip: Open-vocabulary referring expression generation," *arXiv preprint arXiv:2305.19108*, 2023.

[66] Y. Huang, X. Liu, Y. Zhu, Z. Xu, C. Shen, Z. Che, G. Zhang, Y. Peng, F. Feng, and J. Tang, "Label-guided auxiliary training improves 3d object detector," in *European Conference on Computer Vision*. Springer, 2022, pp. 684–700.

[67] P. Zhang, Z. Kang, T. Yang, X. Zhang, N. Zheng, and J. Sun, "Lgd: label-guided self-distillation for object detection," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 36, no. 3, 2022, pp. 3309–3317.

[68] Z. Lin, Z. Zhang, L.-Z. Chen, M.-M. Cheng, and S.-P. Lu, "Interactive image segmentation with first click attention," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2020, pp. 13 339–13 348.

[69] Z. Lin, Z. Zhang, L.-H. Han, and S.-P. Lu, "Multi-mode interactive image segmentation," in *Proceedings of the 30th ACM International Conference on Multimedia*, 2022, pp. 905–914.

[70] M. Hao, Y. Liu, X. Zhang, and J. Sun, "Labelenc: A new intermediate supervision method for object detection," in *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XXV 16*. Springer, 2020, pp. 529–545.

[71] S. Nair, A. Rajeswaran, V. Kumar, C. Finn, and A. Gupta, "R3m: A universal visual representation for robot manipulation," *arXiv preprint arXiv:2203.12601*, 2022.