2024 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)
October 14-18, 2024. Abu Dhabi, UAE

Depth Helps: Improving Pre-trained RGB-based Policy with Depth Information Injection

Xincheng Pang[1,2,*], Wenke Xia[1,2,*], Zhigang Wang[2], Bin Zhao[2,3], Di Hu[1,†], Dong Wang[2,†], Xuelong Li[2,4]

***Abstract*— 3D perception ability is crucial for generalizable robotic manipulation. While recent foundation models have made significant strides in perception and decision-making with RGB-based input, their lack of 3D perception limits their effectiveness in fine-grained robotic manipulation tasks. To address these limitations, we propose a Depth Information Injection (DI^2) framework that leverages the RGB-Depth modality for policy fine-tuning, while relying solely on RGB images for robust and efficient deployment. Concretely, we introduce the Depth Completion Module (DCM) to extract the spatial prior knowledge related to depth information and generate virtual depth information from RGB inputs to aid policy deployment. Further, we propose the Depth-Aware Codebook (DAC) to eliminate noise and reduce the cumulative error from the depth prediction. In the inference phase, this framework employs RGB inputs and accurately predicted depth data to generate the manipulation action. We conduct experiments on simulated LIBERO environments and real-world scenarios, and the experiment results prove that our method could effectively enhance the pre-trained RGB-based policy with 3D perception ability for robotic manipulation. The website is released at https://gewu-lab.github.io/DepthHelps-IROS2024.**

I. INTRODUCTION

Building a generalizable manipulation policy is essential for the development of intelligent robotics. Foundation models have achieved remarkable success in various decision-making problems [1]. Leveraging these models offers a promising approach to developing generalizable manipulation capabilities in robotics. To deploy foundation models for generalizable robotic manipulation, recent research has taken different approaches. Some works use the world knowledge embedded in foundation models for instruction decomposition [2]. Others focus on generating generalizable manipulation policies by leveraging extensive robotic datasets [3], [4]. Despite impressive achievements in manipulation tasks, relying solely on RGB perception limits the understanding of 3D environments. This constraint hinders robotic performance in fine-grained manipulation tasks. In response, this work targets to utilize minimal aligned RGB-D data to enhance the 3D scene perception for robotic manipulation in widespread RGB-only scenarios.

To enhance unimodal perception through multimodal integration, two prominent strategies have emerged. These are cross-modal knowledge distillation [5], [6] and missing modality learning [7], [8]. The former approach uses a multi-modal teacher to distill cross-modal knowledge into an unimodal model. The latter method employs data augmentation across different modalities to encourage robust model learning. Despite the success of these methods in various perception tasks [9], [10], their application to fine-grained robotic manipulation policy learning is challenging. This difficulty arises from sequential accumulative errors in multimodal perception prediction.

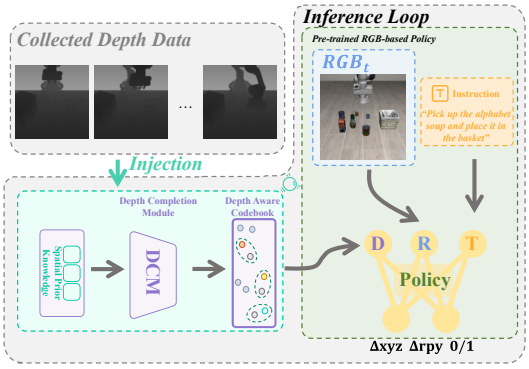

Fig. 1: We propose the Depth Information Injection framework to inject the spatial prior knowledge from the depth information into the RGB-based policy.

To address this challenge and improve the 3D perception capabilities of pre-trained models for fine-grained manipulation tasks, we propose the Depth Information Injection (DI^2) framework. This framework includes the Depth Completion Module (DCM) for integrating spatial prior knowledge with RGB images, and the Depth-Aware Codebook (DAC) to reduce cumulative errors. Specifically, we introduce the DCM to predict depth information from RGB inputs for use in complex environments. This module utilizes the Perceiver Resampler [11] to capture modality-specific biases related to depth, ensuring accurate predictions. We further propose the DAC to refine depth prediction. The DAC transforms input features into discrete quantized vectors, emphasizing crucial information and filtering out modality-specific noise. By aligning quantized vectors from RGB-predicted depth features with those from auxiliary depth data, we accurately represent depth for policy deployment. During inference, the fine-tuned policy uses RGB inputs along with quantized depth features to generate actions. This approach enhances execution efficacy and reliability in complex scenarios.

To validate the effectiveness of our framework, we conducted experiments on the LIBERO [12] benchmark. The experimental results demonstrate excellent performance across a wide range of tasks. Furthermore, we also deploy our

[1]Gaoling School of Artificial Intelligence, Renmin University of China
[2]Shanghai Artificial Intelligence Laboratory
[3]Northwestern Polytechnical University
[4]Institute of Artificial Intelligence, China Telecom Corp Ltd
*Equal contribution. Work is done during internship at Shanghai Artificial Intelligence Laboratory
†Corresponding author

method in real-world scenarios, which proves its effectiveness and reliability in practical applications.

Our main contributions can be summarized as follows:

- We propose the DI^2 framework, which enhances the 3D perception ability of pre-trained RGB-based policy.
- We design the Depth Completion Module and Depth-Aware Codebook to extract modality-specific knowledge for depth prediction on policy deployment.
- Experiments on the LIBERO benchmark and in real-world scenarios validate the effectiveness of our method. We also demonstrate its potential for application in fine-grained manipulation tasks.

II. RELATED WORK

A. Robotic Foundation Models

Foundation models have achieved significant success in perception and decision-making tasks. These include visual grounding [13], visual question answering [11], [14], and more [15]. Recent works [3], [4], [16] have been inspired by the rich world knowledge inherent in foundation models. They aim to equip these models with embodied agents for interactive environments. Early works [2], [17] focused on task planning and decomposed complex instructions into subgoals with a pre-defined skill library. To leverage foundation models for robotic action control, recent work [4] designed an efficient language-conditioned manipulation policy using extensive robotic data. Other research [18] focused on integrating Visual-Language Models (VLM) to utilize their broad world knowledge for robotic manipulation tasks. Despite notable successes in manipulation tasks, the exclusive reliance on RGB perception has limited the potential. In this work, we propose an effective method to enhance the 3D perception ability of the pre-trained manipulation models.

B. Cross-Modal Knowledge Distillation

Cross-modal knowledge distillation is crucial for transferring knowledge between modalities, enhancing the representational capabilities of a target modality. Previous work [5], [6] achieved feature-level knowledge transfer using various loss functions. Other studies [19] developed feature imitation methods that enhance scene perception by integrating knowledge from monocular cameras and depth sensors. Beyond these, some work [9], [20] proposed the adaptive transferring method for robust and effective cross-modal knowledge distillation. However, there is still little research in the field of robotics. Our method addresses the need for extensive robotic RGB-D data by injecting prior depth knowledge into a pre-trained policy. This approach enhances 3D perception for precise manipulation while using minimal data.

C. Missing Modality Learning

Missing modality learning enhances model performance in scenarios with incomplete inputs. Previous work [7], [8] used data augmentation to train models for robust performance in missing modality scenarios, while others [21]–[24] predicted missing modalities during inference using an auxiliary model. In this work, we predict missing modalities with a Depth-Aware Codebook to discretize the predicted features. This method enhances prediction robustness and ensures trajectory accuracy in decision-making.

III. METHOD

In this work, we introduce the DI^2 framework. It uses RGB-Depth data for policy fine-tuning but relies solely on RGB images for robust and efficient deployment. This framework comprises two core modules: the Depth Completion Module and the Depth-Aware Codebook.

A. Overview

The overall framework of the model is shown in Figure 2. When the model takes RGB-D as input during training, the entire model can be expressed as follows:

$$a_t = \pi\left(\mathrm{VLM}\left(f^{text}, f_t^{rgb}, f_t^{depth}\right)\right), \tag{1}$$

where π is the policy model to get the final action. We utilize the VLM to extract features. The f^{text}, f_t^{rgb} and f_t^{depth} are extracted feature.

The RGB-D model, as detailed in Equation (1), effectively utilizes depth images for 3D perception but is limited by its reliance on depth images. To overcome this limitation and enable manipulation in prevalent RGB-only scenarios, we introduce the DCM. This module predicts the depth feature $\hat{f}_t^{depth}$ from the RGB image feature f_t^{rgb}, incorporating spatial prior knowledge P:

$$\hat{f}_t^{depth} = \mathrm{DCM}\left(f_t^{rgb}, P\right). \tag{2}$$

To reduce the cumulative errors, we further propose the Depth-Aware Codebook to discretize the depth features predicted by the DCM:

$$\widetilde{f}_t^{depth} = \mathrm{Codebook}\left(\hat{f}_t^{depth}\right). \tag{3}$$

When we only have RGB as the input during inference, we input the f^{text}, f_t^{rgb}, and $\widetilde{f}_t^{depth}$ together into the VLM to extract features. Ultimately, these features are fed into the policy model π to get the final action.

B. Depth Completion Module

To enable the model to make accurate decisions with only RGB as input and leverage the spatial perception capabilities provided by the depth images during training, we introduce the Depth Completion Module (DCM). This module helps the model extract spatial information from RGB features without depth data and use it in decision-making.

To train the DCM, we utilize the collected trajectory data with RGB-D modalities. Specifically, for each training sample, we use a frozen visual encoder ViT and a specific projection layer Proj^{rgb} to extract the feature of RGB image $f_t^{rgb} = \mathrm{Proj}^{rgb}(ViT(o_t^{rgb}))$. As for the depth image, we make the shape of the depth image o_t^{depth} the same as the RGB image o_t^{rgb}, and then put it into the same frozen visual encoder ViT to extract features. The difference lies in that we use a separate projection layer Proj^{depth} to compress and filter the depth image features $ViT(o_t^{depth})$. Then we can obtain the filtered depth feature representation $f_t^{depth} = \mathrm{Proj}^{depth}(ViT(o_t^{depth}))$. Inspired by Perceiver Resampler [11], we integrate k learnable tokens $P \in \mathbb{R}^{k \times d}$ into the DCM

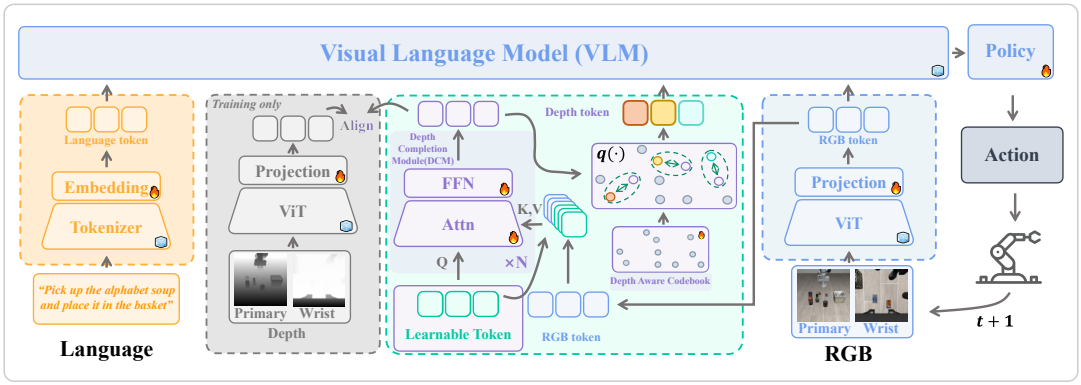

Fig. 2: Overview of our framework. 🔥 denotes the model parameters are updated, while 🧊 indicates the model parameters are frozen. During training, we use the collected depth image to train the DCM and DAC. During inference, we use the DCM together with the RGB token to predict the depth token.

to explicitly capture the spatial prior knowledge. Based on the current timestep's RGB image feature f_t^{rgb} and these learnable tokens P, the DCM estimates the depth feature $\hat{f}_t^{depth}$. In our approach, the parameter P is designated as the query. We concatenate the RGB features f_t^{rgb}, with P to construct both the key and the value components. These are then subjected to processing via cross-attention layers to enhance the feature integration. In essence, P stores spatial prior knowledge based on the similarity between training scenarios and testing scenarios. This enables us to leverage the statistical characteristics within P to predict the depth features related to manipulation. Then the output will be passed into the FFN Layer to get the final feature. The above process can be described as follows:

$$\begin{aligned} \tilde{Q}_{i+1} &= \text{Attn}\left(Q_i, \left[Q_i, \middle| f_t^{rgb}\right], \left[Q_i, \middle| f_t^{rgb}\right]\right), \\ Q_{i+1} &= \text{FFN}\left(\tilde{Q}_{i+1}\right), \end{aligned} \tag{4}$$

where $i \in [0, N)$ means the i-th layer. P is used as Q_0, Q_N is seen as the estimate depth feature $\hat{f}_t^{depth}$.

C. Depth-Aware Codebook

The DCM can obtain an approximate representation of depth features, but there are still errors. Since manipulation involves temporal predictions, errors accumulate over time. This leads to a growing discrepancy between action sequences generated by the DCM and those using depth ground truth. To address this issue, we introduce the Depth-Aware Codebook (DAC).

Mathematically, we define $Z \in \mathbb{R}^{N \times d}$ as the codebook, where N represents the size of the codebook. Given a depth feature $f^{depth} \in \mathbb{R}^{n_d \times d}$, we utilize the quantization operation $\mathbf{q}(\cdot|Z)$ to search for the n_d closest vectors in Z to achieve the discretization of $\hat{f}^{depth}$:

$$\hat{f}_t^{depth} = \mathbf{q}\left(f_t^{depth}|Z\right) = z_k, \text{ where } k = \underset{z_k \in Z}{\text{argmin}} \|f_t^{depth} - e_k\|. \tag{5}$$

Intuitively, the codebook functions as a set of clustering centroids for the depth features in the training dataset, offering higher robustness compared to the original depth features. This characteristic makes the model more reliable when dealing with noise and variations in practical applications.

D. Training Details

Warm-up. We conduct a warm-up training process for manipulation with RGB and depth inputs in the collected trajectories. In this step, we directly train an imitation learning policy with the joint of depth model and frozen pre-trained RGB-based model as shown in Equation (6).

$$L_{warmup} = \sum_t \left|\left|\pi\left(\hat{a}_t | o_t^{rgb}, o_t^{depth}, L\right) - a_t\right|\right|^2. \tag{6}$$

Align. To remove the dependency of depth images, we further train the DCM using the perceptual encoder obtained from the warm-up phase. By utilizing paired RGB-D data in the collected trajectories, our objective is to inject 3D perception knowledge into the policy. We use Equation (7) as the loss to train it.

$$L_{dcm} = \sum_t \left|\left|DCM\left(P, \text{sg}\left(f_t^{rgb}\right)\right) - \text{sg}\left(f_t^{depth}\right)\right|\right|^2, \tag{7}$$

where $\text{sg}(\cdot)$ means the stop gradient operation, $f_t^{\cdot}$ is obtained by the encoder trained in the warm-up phase.

Codebook. This phase is independent of the Align phase. We freeze the depth branch obtained in the warm-up phase and only train the Codebook with MSE loss. However, if we only utilize this loss, it could lead to the codebook collapse problem [25]. Inspired by CVQ-VAE [26], we reinitialize the unoptimized points in each iteration.

Specifically, in each iteration, we perform a running average operation for each codeword c_k in the codebook:

$$\begin{aligned} p_k &= p_k \lambda + \frac{n_k}{B}(1-\lambda), \\ \alpha_k &= \exp\left(-p_k \frac{10N}{1-\lambda}\right), \\ c_k &= c_k(1-\alpha_k) + \bar{z}_k \alpha_k, \end{aligned} \tag{8}$$

where the n_k represents the number of times c_k is used in this iteration. B means the batch size. λ is a hyper-parameter. $\bar{z}_k$ is an anchor vector that is chosen by probabilistic sampling. In our experiments, we set $N = 512, \lambda = 0.99$.

IV. SIMULATION EXPERIMENTS

A. Experiment Setup

To validate the effectiveness of our proposed method, we conduct a series of experiments on the LIBERO benchmark [12]. LIBERO is a large-scale benchmark with 130

robot manipulation tasks. It emphasizes scene diversity and task complexity, enabling fair comparisons and providing comprehensive evaluation results. LIBERO is divided into four sub-suites: Object, Spatial, LongHorizon, and Goal. Each focuses on different manipulation skills, including object interaction, spatial perception, long-term decision-making, and goal-oriented tasks. In terms of training data, we utilize the 50 high-quality human teleoperation demonstration trajectories provided by LIBERO for each task, totaling 6,500 trajectories. We train a multi-task model using these data and evaluate it on the four suites. Regarding the model architecture, we adopt RoboFlamingo [18] pre-trained in Calvin [27] as our model.

B. Preliminary Experiment

To get the upper bound of our method, we first conduct a preliminary experiment where the model could obtain both RGB images and depth images as input. We compare the following methods:

- **RGB-RF** [18]: The standard RoboFlamingo architecture. This baseline uses an LLM to extract text features from task instructions and a ViT to extract visual features from RGB images. These features are then fused using OpenFlamingo [14]. Finally, the fused features are input into the policy network to predict action.
- **RGB-D-RF**: Based on RGB-RF, we add an extra branch to extract features from depth images. We concatenate RGB and depth image features along the channel dimension, then apply the same method as RGB-RF to map the fused features to actions.
- **Data Aug** [7]: This method augments the training data by randomly removing RGB modality or depth modality input with a certain probability p.
- **MM Prompt** [8]: Building upon the Data Aug method, this method introduces an additional learnable token to indicate the current combination type of input modalities. (e.g., RGB-only, Depth-only, or RGB-D)
- **Ours***: To utilize the ground truth depth image, we replace the DCM described in Section III with the depth branch as shown in the gray box in Figure 2.

Method	Avg	Object	Spatial	Long Horizon	Goal
RGB-RF	57.95%	86.20%	69.60%	24.20%	51.80%
RGB-D-RF	61.25%	**88.80%**	69.80%	30.20%	56.20%
Data Aug	58.45%	76.80%	65.80%	35.60%	55.60%
MM Prompt	58.75%	65.40%	**77.40%**	22.00%	**70.20%**
Ours*	**63.95%**	83.00%	69.80%	**37.40%**	65.60%

TABLE I: Preliminary experiment results. We record the success rates of different models under **RGB-D** input (except for RGB-RF, which only utilizes RGB input).

Table I presents success rates of different models on the LIBERO benchmark when provided with RGB-D input. Our method achieves the best overall average success rate of 63.95%, nearly a 6% improvement over the baseline RGB-RF model. This demonstrates the superior performance of our method in effectively utilizing complete RGB-D input. Further analyzing the different task subsets, our model achieves the highest success rate of 37.4% on the Long Horizon suite. This indicates that our method can effectively utilize depth information, exhibiting a clear advantage in complex scenarios that require long-term planning. While the Data Aug and MM Prompt methods perform well on certain subsets, they have lower overall average success rates compared to RGB-D-RF and our method. This is due to the increased learning difficulty from needing to handle different modality combinations. Additionally, we observe that our model achieves better results compared to directly introducing depth in RGB-D-RF. This shows that including the codebook effectively reduces noise-related errors in depth features, enhancing their robustness.

C. Main Experiment

In this section, we verify the task success rate of the model when only RGB images are provided as input. In addition to comparing with the baselines mentioned in Section IV-B, we also compare our results with two cross-modal knowledge distillation methods. These methods distill knowledge from depth features into RGB features, enhancing the spatial perception ability of the RGB model. Specifically, we compare the following two cross-modal knowledge distillation methods:

- **CRD** [5]: This is a cross-modal knowledge distillation method based on contrastive learning loss. In the RGB-D-RF model, we use the depth branch to extract depth features as the teacher model. Additionally, we introduce the CRD contrastive learning loss function during the training of the RGB-RF model.
- **CMKD** [6]: This is a cross-modal knowledge distillation method based on mean squared error (MSE) loss. Similarly, in the RGB-D-RF model, we use the depth branch to extract depth features as the teacher model. We also introduce the MSE loss function during the training of the RGB-RF model.

Method	Avg	Object	Spatial	Long Horizon	Goal
RGB-RF	57.95%	**86.20%**	69.60%	24.20%	51.80%
RGB-D-RF	15.65%	6.60%	22.00%	3.80%	30.20%
Data Aug	54.50%	73.80%	57.20%	28.00%	59.00%
MM Prompt	48.90%	53.40%	62.80%	14.60%	64.80%
CRD	47.60%	62.40%	50.20%	19.60%	58.20%
CMKD	50.60%	63.80%	60.60%	14.60%	63.40%
Ours	**63.15%**	78.60%	**71.20%**	**36.40%**	**66.40%**

TABLE II: Main Experiment Results. We record the success rates of different models in different suites under **RGB-only** input, with each task being tested **50** times. Specifically, in the RGB-D-RF model here, we do not use the Depth branch.

The experimental results are shown in Table II. Comparing the results in Table I and Table II, we observe that in the RGB-only case, the average task success rates for all models have declined to varying degrees. Specifically, the RGB-D-RF model shows the largest performance drop in the RGB-only scene. This model relies heavily on depth information and loses much of its manipulation capability without it. Although the data augmentation methods, Data Aug and MM Prompt, achieve higher average success rates

(54.50% and 48.90%, respectively) than the RGB-D-RF model (15.65%) under RGB-only scene, they are lower than the baseline RGB-RF model (57.95%). This result indicates that although Data Aug and MM Prompt aim to improve generalization and handle missing modalities through data augmentation, they have not succeeded in enhancing RGB-only performance. This has led to an overall decline in performance. The two cross-modal knowledge distillation methods, CRD and CMKD, perform moderately but do not significantly outperform the baseline RGB-RF model. This may be because these methods focus too much on transferring information from depth features in the RGB-RF model to RGB features in the RGB-D-RF model. They neglect the specific requirements of the manipulation task, which leads to performance degradation. In contrast, our method achieves an average success rate of 63.95% with RGB-D input. The success rate decreases only slightly to 63.15% with RGB-only input, showing the smallest drop. This advantage will make our method more applicable in practical application scenarios.

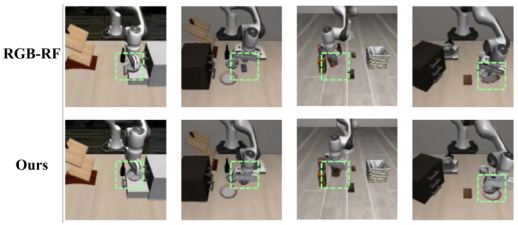

Fig. 3: Qualitative results. This figure illustrates the spatial position perception capability enabled by depth features. The top row shows the effects of the model without depth features. The bottom row shows the results of our method, which can complete the depth features using the DCM.

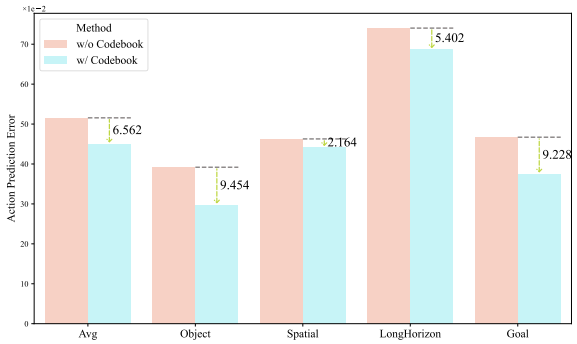

Fig. 4: Action Prediction Error. We evaluate the action sequences generated from predicted depth data against those derived from actual depth images and calculate the Euclidean distance between the actions at corresponding time steps.

Qualitative Results: Figure 3 provides visualized results to illustrate the role of depth information in decision-making. When the robotic arm approaches the target object, depth information plays a critical role. At this stage, accurately perceiving the 3D position and shape of the object is crucial for precise planning. Lacking depth information can lead to a biased understanding of the object's spatial position, affecting the accuracy of action planning.

Prediction Error Analysis: To verify the role of the DAC in our framework, we compare the differences between the action sequences using different predicted depth methods and the action sequences planned using real-depth images. As shown in Figure 4, when using our proposed DAC, the error is smaller than when not using the Codebook. This result verifies that the can effectively improve the quality of depth prediction, generating more accurate and higher-quality depth estimates. With optimized depth representation, our method maintains decision-making and control precision close to that with full RGB-D input, even without real depth data.

DCM	DAC	Avg	Object	Spatial	Long Horizon	Goal
×	×	36.95%	33.80%	42.20%	14.20%	57.60%
✓	×	60.20%	87.00%	67.00%	34.40%	52.40%
✓	✓	63.15%	78.60%	71.20%	36.40%	66.40%

TABLE III: Ablation Experiment Results. × indicates not using that module, while ✓indicates using that module. In particular, we use an MLP to replace the DCM if it is not being used.

Ablation Study: We conducted an ablation study on our model, with results shown in Table III. The results show that the DCM significantly improved performance. It notably enhanced the model's results in the Object, Spatial, and Long Horizon suites. The DAC mainly enhances the model's performance on the Goal test suite, while also bringing some improvements on the Spatial and Long Horizon suites. Notably, adding the DAC led to a performance drop in the Object suite. This result aligns with previous experiments in Table I and Table II and may be related to the suite's characteristics.

V. REAL WORLD EXPERIMENTS

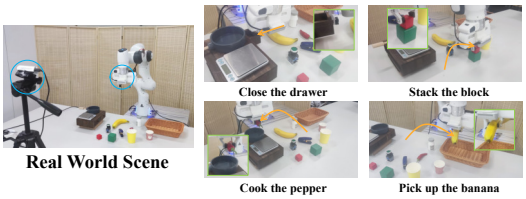

Fig. 5: Illustration of the real-world scene and the four tasks evaluated in our experiments.

Method	Modality	Close drawer	Pick up the banana	Pick up the pepper	Stack the block
RGB-RF	RGB	13.33%	26.67%	10.00%	20.00%
RGB-D-RF	RGB-D	63.33%	83.33%	56.67%	53.33%
RGB-D-RF	RGB	3.33%	20.00%	0.00%	6.67%
Ours	RGB-D	73.33%	76.67%	66.67%	73.33%
Ours	RGB	66.67%	80.00%	63.33%	73.33%

TABLE IV: Real World Experiment Result.

To further verify the effectiveness of the proposed model, we conduct tests in the real world. The experimental setup utilized a Franka Emika Panda robotic arm equipped with two Intel D435 series cameras. We design a complex environment containing four different tasks. The third-person

camera is positioned in front of the robotic arm. This setup makes it difficult for the model to determine the arm's position relative to the target object from texture information. We collect 20 high-quality trajectories for each task using a SpaceMouse. These trajectories are then used to train a single multi-task model. After training, the model is deployed on a host equipped with an NVIDIA 3090 GPU, and real-time inference is performed at a rate of 15 Hz. For each task, we conduct 30 repeated tests to evaluate the model's performance in the real-world environment.

The success rates are recorded in Table IV. The experimental results further validate the effectiveness of our method. Our method performs excellently with complete RGB-D input. More importantly, it also maintains outstanding performance even with just RGB input. More details can be found in the supplementary video.

VI. CONCLUSIONS

In this paper, we propose the Depth Information Injection (DI^2) framework. It enhances the performance of pre-trained robot manipulation models that rely solely on RGB inputs by leveraging minimal aligned RGB-D trajectory data. Our framework centers around two primary modules. The DI^2 framework achieved better results in the LIBERO benchmark. Further, the results of the real-world experiments demonstrate the reliability and applicability of our method in practical application scenarios.

VII. ACKNOWLEDGEMENT

This work is supported by the National Natural Science Foundation of China (NO.62106272), Shanghai AI Laboratory, National Key R&D Program of China (2022ZD0160101), the National Natural Science Foundation of China (62376222), and Young Elite Scientists Sponsorship Program by CAST (2023QNRC001).

REFERENCES

[1] 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.

[2] K. Lin, C. Agia, T. Migimatsu, M. Pavone, and J. Bohg, "Text2motion: From natural language instructions to feasible plans," *Autonomous Robots*, vol. 47, no. 8, pp. 1345–1365, 2023.

[3] 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.

[4] Octo Model Team, D. Ghosh, H. Walke, K. Pertsch, K. Black, O. Mees, S. Dasari, J. Hejna, C. Xu, J. Luo, T. Kreiman, Y. Tan, D. Sadigh, C. Finn, and S. Levine, "Octo: An open-source generalist robot policy," https://octo-models.github.io, 2023.

[5] Y. Tian, D. Krishnan, and P. Isola, "Contrastive representation distillation," *arXiv preprint arXiv:1910.10699*, 2019.

[6] Y. Hong, H. Dai, and Y. Ding, "Cross-modality knowledge distillation network for monocular 3d object detection," in *ECCV*, ser. Lecture Notes in Computer Science. Springer, 2022.

[7] S. Parthasarathy and S. Sundaram, "Training strategies to handle missing modalities for audio-visual expression recognition," in *Companion Publication of the 2020 International Conference on Multimodal Interaction*, 2020, pp. 400–404.

[8] Y.-L. Lee, Y.-H. Tsai, W.-C. Chiu, and C.-Y. Lee, "Multimodal prompting with missing modalities for visual recognition," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 14 943–14 952.

[9] Y. Chen, Y. Xian, A. Koepke, Y. Shan, and Z. Akata, "Distilling audio-visual knowledge by compositional contrastive learning," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 7016–7025.

[10] X. Guo, S. Shi, X. Wang, and H. Li, "Liga-stereo: Learning lidar geometry aware representations for stereo-based 3d detector," in *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, October 2021.

[11] J.-B. Alayrac, J. Donahue, P. Luc, A. Miech, I. Barr, Y. Hasson, K. Lenc, A. Mensch, K. Millican, M. Reynolds, *et al.*, "Flamingo: a visual language model for few-shot learning," *Advances in Neural Information Processing Systems*, vol. 35, pp. 23 716–23 736, 2022.

[12] B. Liu, Y. Zhu, C. Gao, Y. Feng, Q. Liu, Y. Zhu, and P. Stone, "Libero: Benchmarking knowledge transfer for lifelong robot learning," *Advances in Neural Information Processing Systems*, vol. 36, 2024.

[13] S. Wu, H. Fei, L. Qu, W. Ji, and T.-S. Chua, "Next-gpt: Any-to-any multimodal llm," *arXiv preprint arXiv:2309.05519*, 2023.

[14] A. Awadalla, I. Gao, J. Gardner, J. Hessel, Y. Hanafy, W. Zhu, K. Marathe, Y. Bitton, S. Gadre, S. Sagawa, J. Jitsev, S. Kornblith, P. W. Koh, G. Ilharco, M. Wortsman, and L. Schmidt, "Openflamingo: An open-source framework for training large autoregressive vision-language models," *arXiv preprint arXiv:2308.01390*, 2023.

[15] Y. Wei, D. Hu, Y. Tian, and X. Li, "Learning in audio-visual context: A review, analysis, and new perspective," *arXiv preprint arXiv:2208.09579*, 2022.

[16] W. Xia, D. Wang, X. Pang, Z. Wang, B. Zhao, and D. Hu, "Kinematic-aware prompting for generalizable articulated object manipulation with llms," *arXiv preprint arXiv:2311.02847*, 2023.

[17] G. Wang, Y. Xie, Y. Jiang, A. Mandlekar, C. Xiao, Y. Zhu, L. Fan, and A. Anandkumar, "Voyager: An open-ended embodied agent with large language models," *arXiv preprint arXiv:2305.16291*, 2023.

[18] X. Li, M. Liu, H. Zhang, C. Yu, J. Xu, H. Wu, C. Cheang, Y. Jing, W. Zhang, H. Liu, H. Li, and T. Kong, "Vision-language foundation models as effective robot imitators," *arXiv preprint arXiv:2311.01378*, 2023.

[19] Z. Chong, X. Ma, H. Zhang, Y. Yue, H. Li, Z. Wang, and W. Ouyang, "Monodistill: Learning spatial features for monocular 3d object detection," *arXiv preprint arXiv:2201.10830*, 2022.

[20] W. Xia, X. Li, A. Deng, H. Xiong, D. Dou, and D. Hu, "Robust cross-modal knowledge distillation for unconstrained videos," *arXiv preprint arXiv:2304.07775*, 2023.

[21] J. Zhao, R. Li, and Q. Jin, "Missing modality imagination network for emotion recognition with uncertain missing modalities," in *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, 2021, pp. 2608–2618.

[22] M. Ma, J. Ren, L. Zhao, S. Tulyakov, C. Wu, and X. Peng, "Smil: Multimodal learning with severely missing modality," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 35, no. 3, 2021, pp. 2302–2310.

[23] L. Cai, Z. Wang, H. Gao, D. Shen, and S. Ji, "Deep adversarial learning for multi-modality missing data completion," in *Proceedings of the 24th ACM SIGKDD international conference on knowledge discovery & data mining*, 2018, pp. 1158–1166.

[24] Z. Yang, H. Zhang, Y. Wei, Z. Wang, F. Nie, and D. Hu, "Geometric-inspired graph-based incomplete multi-view clustering," *Pattern Recognition*, vol. 147, p. 110082, 2024. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0031320323007793

[25] Y. Takida, T. Shibuya, W. Liao, C.-H. Lai, J. Ohmura, T. Uesaka, N. Murata, S. Takahashi, T. Kumakura, and Y. Mitsufuji, "Sq-vae: Variational bayes on discrete representation with self-annealed stochastic quantization," *arXiv preprint arXiv:2205.07547*, 2022.

[26] C. Zheng and A. Vedaldi, "Online clustered codebook," in *Proceedings of International Conference on Computer Vision (ICCV)*, October 2023.

[27] O. Mees, L. Hermann, E. Rosete-Beas, and W. Burgard, "Calvin: A benchmark for language-conditioned policy learning for long-horizon robot manipulation tasks," *IEEE Robotics and Automation Letters (RA-L)*, vol. 7, no. 3, pp. 7327–7334, 2022.