Instruct2Subtask: A Language Parsing Framework for Sequential Robotic Manipulation

Tribikram Dhar, Rebanta Dey, Pandya Naisarg Jayantkumar, Samrat Dutta
TCS Research, Bangalore, India.
(tribikram.d, dey.rebanta, naisarg.pandya, d.samrat)@tcs.com

***Abstract*—We present a modular robotic manipulation framework, robot agnostic in nature that executes long-horizon tasks specified through natural language instruction. The system integrates a Vision-Language Supervisory Planner (VLM-SP), a Grasp-Pose Estimator (GPE), and a structured skill repository containing the robot's executable skills. Given a natural-language instruction, the VLM planner decomposes the task into grounded subtasks-an essential step for reducing planning complexity, enabling skill reuse, and ensuring robust execution. Each subtask is then mapped to appropriate skills and paired with object-specific grasp predictions for reliable manipulation.**

***Index Terms*—Vision Language Models, Robotic Manipulation, Task and Motion Planning**

I. Introduction and Motivation

Long-horizon robotic manipulation tasks require a robot to reason over multi-step objectives, diverse sets of objects and varying scene contexts. Recent advances in vision–language–action models like RT-1 [1] and RT-2 [2], have shown that large multimodal models can provide powerful grounding and generalization for real-world manipulation. These systems can translate high-level instructions into actionable behaviors, but they typically rely on massive training datasets and specialized robot.

To address these constraints, we propose a robot-agnostic, modular framework that leverages the reasoning capabilities of VLMs for task decomposition while preserving classical control for reliable execution without end-to-end training. Our approach integrates a fine-tuned VLM supervisory planner, a grasp estimator, and a repository of robot motor skills to transform natural language instructions into executable subtasks.

To summarise, we propose

1) We propose a modular approach to solve long-horizon task decomposition by integrating VLM-SP, GPE and a skill repository to perform required robot manipulation.
2) Our framework is hardware agnostic and is scalable to different arm manipulator with varying kinematics.
3) Our framework can be easily deployable to any environment with minimal fine-tuning.

II. System Overview

The proposed framework consists of three main modules: the VLM Supervisory Planner (VLM-SP), the Grasp Point Estimator (GPE), and a skill repository. As illustrated in Fig. 1, the user provides a task command either through text or speech. The VLM-SP uses this command along with a task environment RGB-D image to decompose the long-horizon task into a sequence of subtasks and an associated list of relevant objects. Each object in this list is then processed by the GPE to compute its corresponding grasp pose. Subsequently, each subtask is matched with an appropriate skill from the skill repository, allowing the system to generate the optimal robot actions required to execute the subtasks in order. The complete pseudocode for this framework is provided in Algorithm 1

Algorithm 1 Instruct2Subtask pseudocode

Input: User Command (text or speech)
Output: Robot actions

```
object_list, pose_list ← ∅                                  ▷ empty list
rgbd_image ← ENV_CAMERA
subtask_list, object_list ← VLM_SP(user_command, rgb_image)
for all (subtask, object) in (subtask_list, object_list) do
    pose ← GPE(object, rgbd_image)
    pose_list.APPEND(pose)
end for
while subtask_list is not empty do
    subtask ← POP_FRONT(subtask_list)
    pose ← POP_FRONT(pose_list)
    robot_proprio ← GET_ROBOT_PROPRIOCEPTION
    π ← SKILL_REPOSITORY(subtask)
    robot_action ← π(pose, robot_proprio)
    SEND_TO_ROBOT(robot_action)
end while
```

III. Modules

The following subsections explain the working principle of each of the 3 modules in our framework.

A. VLM Supervisory Planner

The VLM Supervisory Planner is an implementation of a fine-tuned version of MiniGPT-4 [3], which leverages a Vision Transformer (ViT) for visual feature extraction, a Q-former for vision-language cross modal learning, a projection layer to map Q-former embeddings into text space embedding and a Vicuna LLM for text generation. Task environment image read from the camera is converted into meaningful visual feature embeddings using ViT. These embeddings are processed by the Q-former to perform cross-modal attention with language tokens of the user command generated from

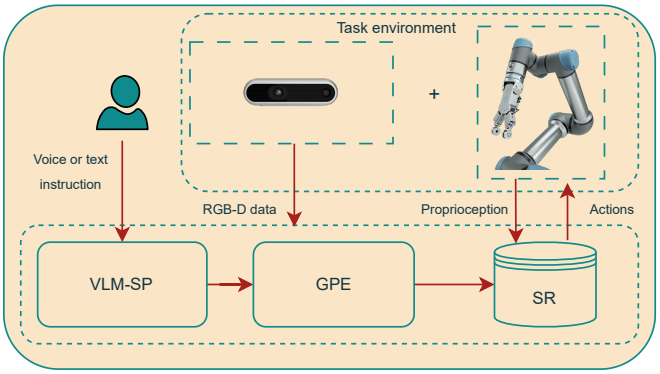

Fig. 1. Overview of our framework. The task environment contains a stationary robot arm manipulator operating over the task table, with an RGB-D camera capturing live images of the scene. SR: Skill Repository.

BLIP and the resultant embedding is transformed using the trained projection layer into text space embedding. The text space embeddings are passed into the Vicuna LLM to generate meaningful subtasks for the given task by the user. The VLM-SP was fine-tuned on a laboratory dataset containing 1000 `(image, text)` pairs.

B. Grasp Points Estimator

The Grasp Point Estimator is designed to determine optimal grasp locations for object manipulation. It operates using a sequential architecture called GroundingSAM, integrating GroundingDINO [4] and MobileSAM [5].

GroundingDINO is a promptable object detection framework capable of identifying objects in an image based on natural language descriptions. It employs a Swin Transformer backbone to extract visual embeddings and a BERT encoder to process textual inputs. These initial `(image, text)` features are refined through a Feature-Enhancer module that performs cross-modal fusion between visual and textual representations. The enhanced features are then passed to the Language-Guided Query Selection module—modeled after DETR [6]—and subsequently decoded to produce multiple `(noun, bounding box)` predictions.

MobileSAM, a lightweight version of Segment Anything (SAM) [7], generates precise segmentation masks from user prompts. It can accept either bounding box coordinates or single-point prompts to guide the segmentation. The model uses a ViT-Tiny backbone, a distilled version of ViT-Large to extract dense visual features, which the decoder then combines with the prompt to produce the final segmentation mask. In this setup, the bounding box coordinates from GroundingDINO serve as prompts to MobileSAM, enabling it to generate segmentation masks corresponding to the user-specified objects.

Finally, a heuristic-based method analyzes the resulting mask to compute a stable and reliable grasp point.

C. Skill Repository

The skill repository comprises a library of robot motor skills where each skill is composed of multiple robot primitive behaviours. The current implementation includes two fundamental skills which are pick and place. Each skill is composed of different primitives for example the skill "pick" is a composition of primitives 1. Open gripper 2. Move to source object with predicted pose 3. Close gripper 4. Move end-effector.

Each skill is a motor policy $\pi : (S, P) \rightarrow A$ that maps the robot's proprioceptive state $s \in S$ and target pose $p \in P$ to an action $a \in A$. During inference, each subtask from the VLM-SP is associated with its corresponding skill, which uses the GPE-predicted poses and current robot state to generate real-time joint-space actions using the Orocos Kinematics and Dynamics Library [8].

IV. Hardware Setup

We implemented our framework on a workstation equipped with Intel Xeon Gold processor (64 GB system memory) and Nvidia RTX A5500 (24 GB VRAM). The hardware for our experiments included an Intel Realsense Camera, a UR10-CB3 robot with a mounted RG2 gripper. ROS Noetic was used for communication.

V. Conclusion

We introduced a modular framework for long-horizon task decomposition that combines a VLM-based planner, a grasp estimator, and a skill library. Our experiments demonstrate strong performance with minimal fine-tuning, along with easy adaptation to different environments. The approach scales to arm manipulators with varying kinematics and can incorporate additional knowledge modules to handle new objects and skills. While our current evaluation focuses on static settings, we plan to extend the system to dynamic environments in future work.

References

[1] 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.

[2] B. Zitkovich, T. Yu, S. Xu, P. Xu, T. Xiao, F. Xia, J. Wu, P. Wohlhart, S. Welker, A. Wahid *et al.*, "Rt-2: Vision-language-action models transfer web knowledge to robotic control," in *Conference on Robot Learning*. PMLR, 2023, pp. 2165–2183.

[3] 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.

[4] S. Liu, Z. Zeng, T. Ren, F. Li, H. Zhang, J. Yang, Q. Jiang, C. Li, J. Yang, H. Su *et al.*, "Grounding dino: Marrying dino with grounded pre-training for open-set object detection," in *European conference on computer vision*. Springer, 2024, pp. 38–55.

[5] C. Zhang, D. Han, Y. Qiao, J. U. Kim, S.-H. Bae, S. Lee, and C. S. Hong, "Faster segment anything: Towards lightweight sam for mobile applications," *arXiv preprint arXiv:2306.14289*, 2023.

[6] N. Carion, F. Massa, G. Synnaeve, N. Usunier, A. Kirillov, and S. Zagoruyko, "End-to-end object detection with transformers," in *European conference on computer vision*. Springer, 2020, pp. 213–229.

[7] 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," in *Proceedings of the IEEE/CVF international conference on computer vision*, 2023, pp. 4015–4026.

[8] H. Bruyninckx, P. Soetens, and B. Koninckx, "The real-time motion control core of the Orocos project," in *IEEE International Conference on Robotics and Automation*, 2003, pp. 2766–2771.