ECE 445

SENIOR DESIGN LABORATORY

PROJECT PROPOSAL

Long-horizon Task Completion with Robotic Arms by Human Instructions

BINGJUN GUO (bingjun3)
QI LONG (qilong2)
QINGRAN WU (qingran3)
YUXI CHEN (yuxi5)
(alphabetically)

Sponsor: Gaoang Wang, Liangjing Yang
TA: Tielong Cai, Tianci Tang

March 14, 2025

1 Introduction

1.1 Problem

The application of robotic arms for complex, long-horizon tasks such as assembling, cooking, and packing is rapidly expanding due to their potential to improve efficiency and reduce human labor. However, executing these multi-step operations consistently remains challenging. Such tasks require robots to reason about the interdependencies between subtasks, adapt to dynamic environmental conditions, and integrate continuous feedback effectively. Current robotic manipulation methods struggle with decomposing and chaining task into manageable actions and maintaining robustness throughout execution. Moreover, many existing robotic systems are limited to predefined scenarios with known object interactions, making them unsuitable for dynamic environments where conditions frequently change. To overcome these limitations and enable robotic arms to autonomously manipulate objects based on real-time feedback, there is a critical need for a comprehensive framework that integrates perception, planning, and acting intelligence effectively.

1.2 Solution

Our proposed solution addresses the challenges associated with long-horizon robotic manipulation tasks by integrating Perception, Planning, and Acting Intelligence into a cohesive framework. At a high level, our approach leverages advanced sensing technologies alongside intelligent planning algorithms to enable a robotic arm (specifically UR3) to autonomously execute complex tasks based on human instructions.

In detail, our solution begins with perception: an RGB or RGB-D camera mounted on the robotic arm captures images of the environment and objects involved in the task. Computer vision techniques process these images to accurately identify and localize objects. Next, in the planning stage, we utilize advanced language models capable of interpreting semantic instructions provided by human users alongside processed visual data to generate logical sequences of subtasks. These plans consider dependencies between subtasks and environmental constraints. Finally, during the acting stage, the robotic arm executes planned movements guided by continuous feedback from visual and tactile sensors integrated into a closed-loop control system.

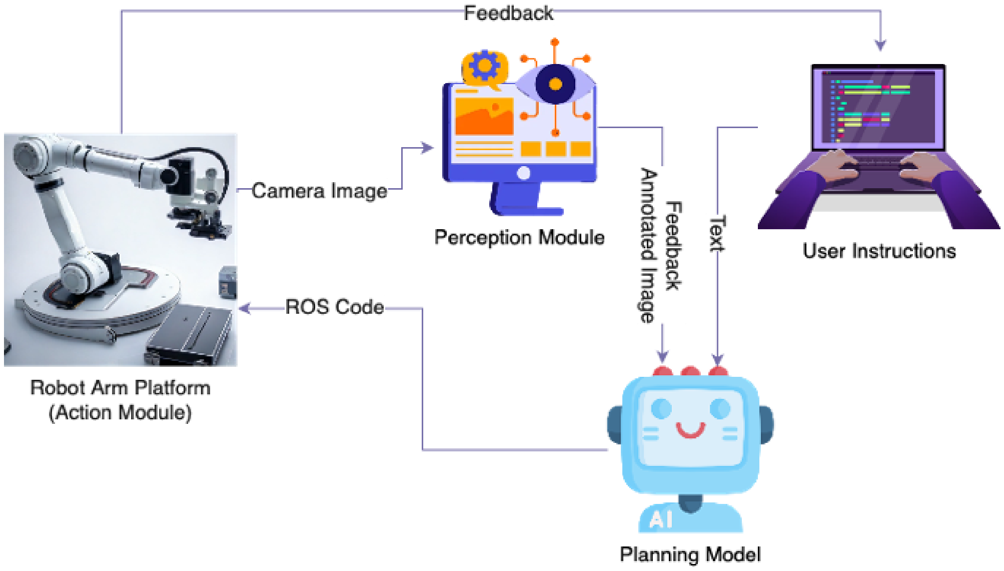

Figure 1: Visual Aid

1.3 High-level Requirements List

1.3.1 Perception Accuracy

The system must achieve at least 90% accuracy in identifying and localizing target objects within its operating environment.

1.3.2 Planning Efficiency

The robot must generate actionable multi-step operation plans within 5 seconds after receiving human instructions.

1.3.3 Execution Robustness

The robotic arm must successfully complete at least 90% of attempted long-horizon tasks without collisions or critical errors under varying environmental conditions.

2 Design

2.1 Mechanics

We are using the UR3 robotic arm as our base platform and will design a custom two-finger parallel gripper for object manipulation. The suction-based end effector currently available in our lab is limited to objects with regular shapes or flat surfaces, making it unsuitable for items like apples and bananas. To overcome this limitation, we plan to build a two-finger gripper driven by a motor and gear sets to handle a wider variety of objects.

A preliminary model of this gripper is shown is the figure below:

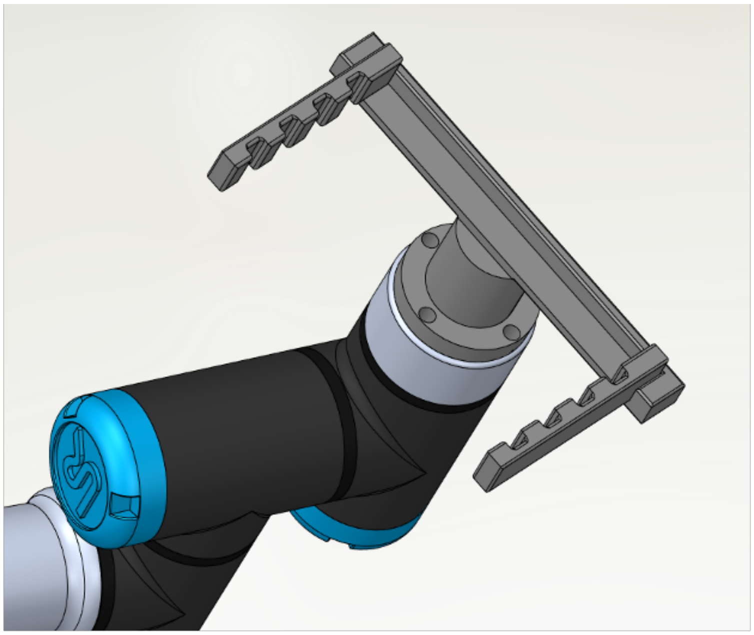

Figure 2: Preliminary Model of Gripper

In this design, we will develop a PCB to control the motor, which in turn actuates the gripper to manipulate objects. The main structure will be 3D-printed, with supplementary parts made from acrylic sheets. To ensure a secure grip on objects, we plan to use silicone or other high-friction materials in the gripper.

2.2 Abstracted Subsystem Overview & Requirements

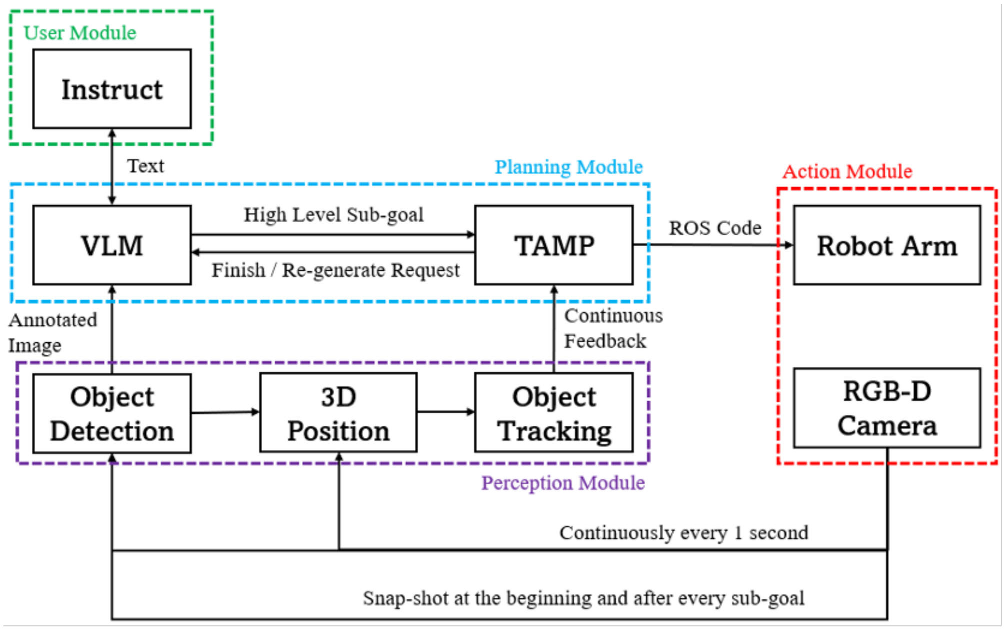

Figure 3: Block Diagram for Whole System

The overall system includes four modules: User, Perception, Planning and Action.

2.2.1 User Module

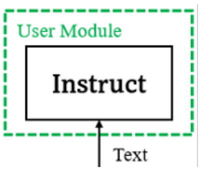

Figure 4: User Module

a) **Overview:** This is the user interface for interaction. User can input a high-level instruction in natural language, for example, “Clean the table”. When the task is finished, the message will be prompted to the user module.

b) **Requirements:** The user input should be a high-level task that cannot be solved within smaller than 10 unit steps.

2.2.2 Perception Module

a) **Overview:** It helps to capture useful information from the working environment, which then will be used for planning and acting. It includes three components: Object Detection, 3D Positioning and Object Tracking.

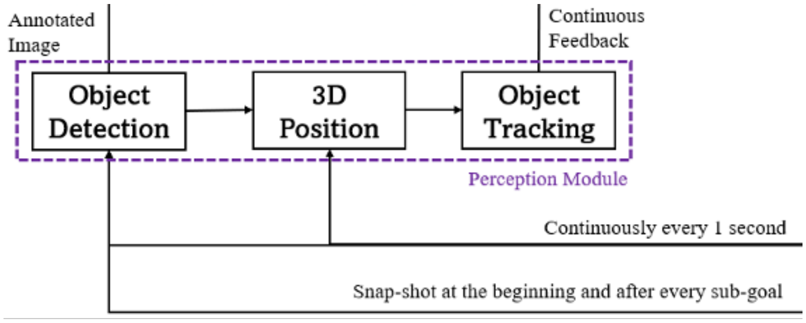

Figure 5: Perception Module

b) **Functionalities:** Object Detection is responsible for identifying the objects of interest in a scene and labelling them with semantic classification (e.g., a cup) and region (a bounding box). 3D Positioning is responsible for labelling each object identified by Object Detection with object pose (e.g., location, object shape) in 3D coordinates. Object Tracking is responsible for continuously keeping track of the object's location in 3D coordinates.

c) **Workflow:** When a user input is received or a sub-goal is achieved, a snapshot of the scene will be sent (by Action Module) to Object Detection, whose annotation result will be sent to VLM (Planning Module) for planning. During the process of robot arm's taking actions, real-time capturing RGB image and depth (from Action Module) will be sent to Object Detection, 3D Positioning and Object Tracking for continuous tracking, which will provide feedback for modifying plans and actions (Planning Module) if needed.

d) **Requirements:** All components need to be able to identify at least 30 categories of daily life objects, including 3D shapes, finishing a single computation within 1 second. The input should include at least RGB images and depth.

2.2.3 Planning Module

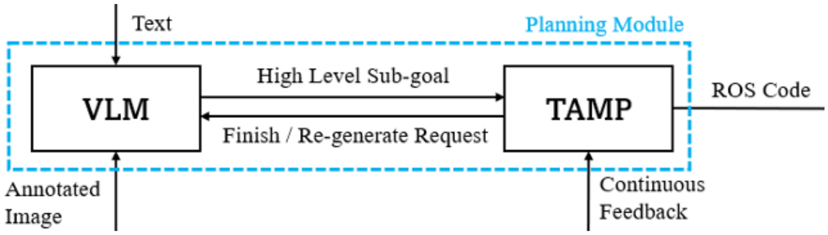

Figure 6: Planning Module

a) **Overview:** It does the reasoning or analysis work to transform the user instruction into sub-task planning, which composes the whole long-horizon high-level task, and then transform the sub-tasks from natural language to ROS coding. It includes two components: Vision Language Model (VLM) and Task and Motion Planner (TAMP).

b) **Functionalities:** VLM can generate text answers to questions in image and text format, which includes reasoning and planning. TAMP can generate motion trajectories that consider both sub-goal and robot arm's reality constraints, which refine the plan and transform it into robot-understandable language.

c) **Workflow:** Given a user instruct (from User Module) or a finish signal (reaching the end of the plan), it will query the Perception Module for an annotated image of current working environment. Then VLM generates a plan which includes the next single object of interest for interaction, the sub-goal related to this object and detailed action steps. With this information, TAMP generates motion trajectories for the robot arm, coded in ROS, which will guide the robot arm's action (Action Module). During the process of acting, TAMP will also receive feedback from Perception Module continuously, which will then adjust its plan and motion trajectories accordingly.

d) **Requirements:** VLM should be pre-trained to have a high performance, such as having 70B parameters. The TAMP should be able to generate ROS codes that can move the robot arm.

2.2.4 Action Module

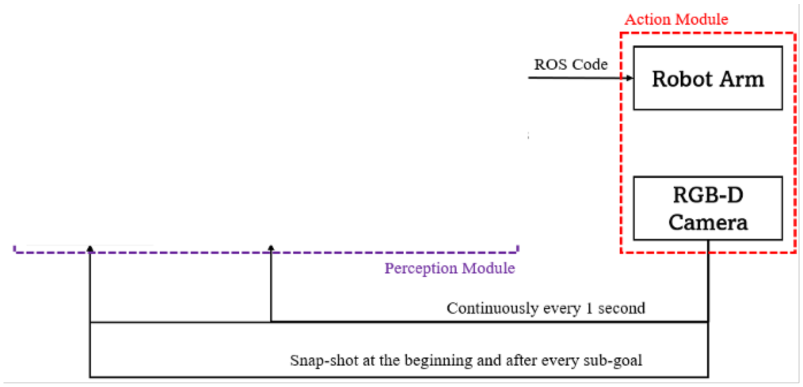
Action Module
ROS Code
Robot Arm
RGB-D Camera
Perception Module
Continuously every 1 second
Snap-shot at the beginning and after every sub-goal

Figure 7: Action Module

a) **Overview:** It interacts with the real environment to execute the tasks. It includes two components: Robot Arm and RGB-D Camera, which together enable the robotic system to manipulate objects and provide real-time visual feedback to the Perception Module.

b) **Functionalities:** The Robot Arm receives motion commands (ROS codes) from the TAMP (Planning Module) and moves accordingly to complete sub-goals such as picking up or placing objects. The RGB-D Camera captures RGB images and depth information and sends them to the Perception Module for Objection Detection, 3D Positioning, and Object Tracking.

c) **Workflow:** When the TAMP (Planning Module) generates motion trajectories, it sends these commands to the Robot Arm, which adjusts its joints and end effector position according to achieve the desired positions and orientations. Specifically, the TAMP sends these commands to the PCB in the specially designed hand, which controls the motor's motion and subsequently controls the gripper's motion. As the robot operates, the RGB-D Camera continuously streams RGB images and depth information to the Perception

Module for real-time 3D Positioning and Object Tracking. At the start and completion of each sub-goal, the camera captures a snapshot that is processed by the Perception Module for updated object detection, which will be used in the Planning Module.

d) **Requirements:** The Robot Arm should achieve a high degree of positional accuracy (for example, ± 1 mm at the end effector) to ensure reliable manipulation of objects. It should be able to lift objects up to 1 kg. The RGB-D Camera should be able to capture RGB images and depth information in real time.

2.3 Tolerance Analysis

2.3.1 Perception Module

For Perception Module, by training, the Object Detection is capable of correctly recognizing at least 30 types of objects, given current open-source dataset COCO contains 80 categories, VOC contains 20 categories and YOLO algorithm can achieve high accuracy (68.9 mAP at [0.5]).

2.3.2 Planning Module

For Planning Module, current Large Language Models (e.g., DeepSeek, ChatGPT, Doubao) are capable of reasoning about images for text generation.

2.3.3 Action Module

For Action Module, 3D-printed and acrylic structures can withstand substantial forces without damage, and collision-free algorithms help prevent severe mechanical damage. Additionally, using silicone with a patterned surface increases the coefficient of friction, which enables the gripper to lift heavier objects with reduced risk of slipping.

2.3.4 Remainings

The critical part is that it is unknown how we can transform natural language instructions into motion trajectories and ROS codes efficiently, which poses a risk.

3 Ethics and Safety

3.1 Ethical Issues

3.1.1 Development-Stage Ethical Concerns

The ACM Code of Ethics[1] emphasizes honesty about the capabilities and limitations of a system. To avoid misrepresenting the capabilities of our robot, we will specify its application scenarios, operational boundaries, and potential failing cases. In the meantime, since the planning stage of our robot is data-based, bias in training data that could result in unfair or unpredictable behavior is possible. Following IEEE and ACM guidelines, we will take fairness in consideration when determining our planning model (VLM) and test under diverse circumstances to mitigate bias. The IEEE Code of Ethics [2] also stresses that developers should accept responsibility for their technology's consequences, as the ACM Code emphasize on the robustness and usability of the system. To promote the sustainable development of our project, we will include logging and traceability features during the development to allow both developers and users to diagnose errors and attribute responsibility.

3.1.2 Misuse and Unintended Consequences

The project could be misused for unsafe or malicious tasks such as unauthorized modifications or weaponization. Responding to both the IEEE Code's concern [2] about physical abuse and ACM Code's[1] valuing on the public good, we will strictly restrict our robot's capability to conduct physical harm through e.g. limit the maximum operation speed or rejecting malicious language instructions. Another risk is that users might overestimate the robot's ability, leading to dangerous reliance on automation. We will provide clear user guidelines and training that ensure the users remaining awareness of the robot's limitations, responding to the ACM Code's requirement to foster public awareness of our technology.

3.2 Safety and Regulatory Standards

Safety is to be attached with primary significance during the process of both development and utilization. During the process, the robot arm could accidentally harm developers or users due to unexpected movements. In addition, bugs or adversarial commands caused by illusions of LLM could lead to unsafe robot actions. Below are several notable federal and industry safety standards to consider.

3.2.1 ISO 10218-1/2[3]

This standard mandates safety requirements for robotic arms, including emergency stop mechanisms, protective barriers, and collaborative safety features. The project will comply with these safety measures to ensure operational safety.

3.2.2 ISO/TS 15066[4]

As the project involves a robot interacting with human instructions, this standard provides guidelines on safe human-robot interaction, ensuring safe speeds, forces, and workspace conditions. Our kinetic and dynamic interpreter will comply to a preset safe constraint for motion output, and we will ensure that the instructions are given from a safe distance with respect to the robot's workspace.

3.2.3 ANSI/RIA R15.06[5]

This U.S. standard aligns with ISO 10218 and includes risk assessments, safety interlocks, and safe operation zones, all of which will be incorporated into the project's design.

3.2.4 Robot Manipulator General Safety Procedures[6]

Additional to the social standards, Illinois Robotics Group impose necessities to check the damage condition of robot arms and strict clothing regulations. Also, before the testing or instructing start, there should be loud and clear announcement that raise awareness and mental alert in the vicinity of everyone.

References

[1] ACM. “”ACM Code of Ethics”.” (2018), [Online]. Available: https://www.acm.org/code-of-ethics (visited on 03/14/2025).

[2] IEEE. “”IEEE Code of Ethics”.” (2016), [Online]. Available: https://www.ieee.org/about/corporate/governance/p7-8.html (visited on 03/14/2025).

[3] ISO. “”Industrial robot safety bundle”.” (2025), [Online]. Available: https://www.iso.org/publication/PUB200102.html.

[4] ISO. “”Robots and robotic devices — Collaborative robots”.” (2016), [Online]. Available: https://www.iso.org/obp/ui/#iso:std:iso:ts:15066:ed-1:v1:en.

[5] ISHN. “”ANSI/RIA R15.06-2012- The industrial robot safety standard”.” (2018), [Online]. Available: https://www.ishn.com/articles/107815-ansiria-r1506-2012--the-industrial-robot-safety-standard.

[6] I. R. Group. “”Robot Manipulator Safety Rules”.” (), [Online]. Available: https://robotics.illinois.edu/lab/robot-manipulator-safety-rules/ (visited on 03/14/2025).