Voice-Driven Instruction to Symbolic Task Execution for Automation (VISTA)

Khyat Sharma (RBE)
Advisor: Professor Jing Xiao (RBE), Professor Constantinos Chamzas (RBE)

Project Overview

VISTA is a voice-driven framework for robotic block manipulation that converts spoken natural language instructions into robot actions. Rather than manually programming new goal states, a user can describe a target structure in natural language, enabling more accessible robot task specification for manufacturing, warehousing, and assistive robotics.

Objectives

- Interpret robotic block-manipulation goals in natural language
- Convert language into a procedural world-state representation
- Generate valid PDDL problems for symbolic planning
- Execute plans through collision-free motion planning on a UR10
- Benchmark local LLM accuracy across models and complexity
- Identify key bottlenecks in language-driven task execution

System Components

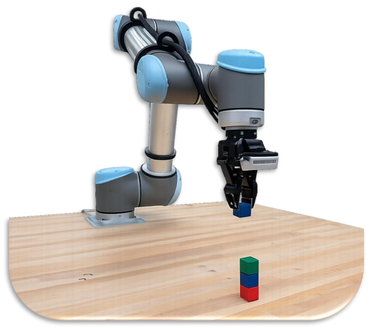
Hardware:

- Robot: UR10 robotic manipulator
- Input: Microphone (16 kHz continuous audio)
- Compute: NVIDIA RTX 4090 workstation

Software:

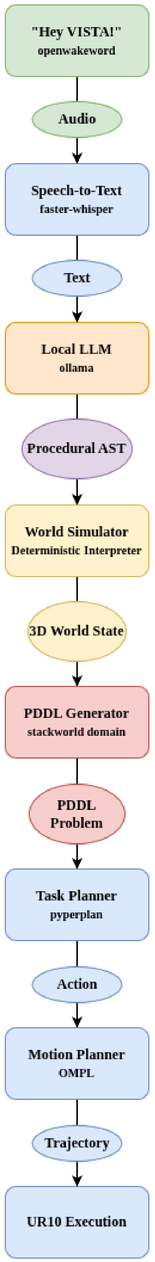
- Wake Word: openwakeword ("hey vista")
- Speech-to-Text: faster-whisper (tiny model, CPU, int8)
- LLM Runtime: ollama (local inference)
- Symbolic Planner: pyperplan (A* search)
- Motion Planning: OMPL
- Domain: Custom PDDL ("stackworld")

UR10 Execution & Task Plan



```
pick-up(blue_cube_0)
stack(blue_cube_0, red_cube_0)
pick-up(green_cube_0)
stack(green_cube_0, blue_cube_0)
pick-up(blue_cube_1)
stack(blue_cube_1, green_cube_0)
```

System Architecture



"Hey VISTA!" Wake Word

On detection, records up to **10s of speech**, then transcribes locally with faster-whisper (tiny model, CPU, int8). **Fully hands-free**, no keyboard interaction needed!

Local LLM

Carefully **engineered system prompt** constrains the output to only valid procedural AST JSON, defining the schema, 9 permitted operations, parameters, and coordinate conventions.

Procedural AST

Intermediate representation with 9 operations (BUILD_LINE, BUILD_RECT, PLACE_SINGLE, STACK_ON, MOVE, etc.)

```
{
  "steps": [
    {
      "op": "PLACE_SINGLE",
      "label": "red_base",
      "color": "red",
      "origin": {"x": 0, "y": 0}
    },
    {
      "op": "BUILD_STACK",
      "label": "alt_tower",
      "count": 3,
      "relative_to": {"ref": "red_base", "anchor": "top"},
      "color_pattern": {"mode": "alternating", "colors": ["blue", "green"]}
    }
  ]
}
```

3D World State

Fully deterministic, rule-based interpreter executes the AST step by step, tracking each block's **x/y/z** position, **color**, and **positional relationships**. The resulting state is converted into PDDL predicates.

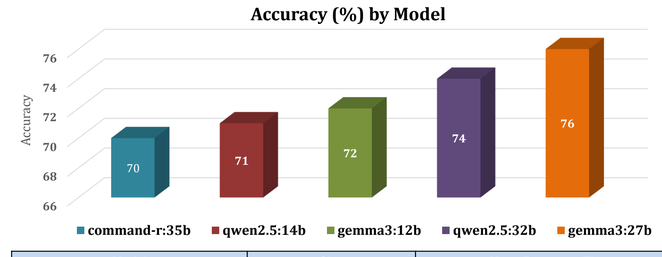
stackworld PDDL Domain

Custom domain with 3 types (block, position, color) and 4 actions (Pickup, Putdown, Stack, Unstack). Predicates encode stacking, grid placement, color, availability, and gripper state.

Benchmark Dataset

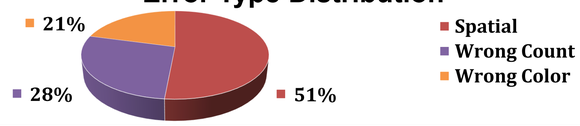
100 natural-language block rearrangement tasks drawn from the **ManipulationNet** challenge.
Tasks span 4 difficulty levels: **Entry** (10), **Easy** (70), **Medium** (10), **Hard** (10): from **single-block placements** to **multi-step constructions** with **color changes** and **spatial transformations**.
Accuracy = exact match between generated and ground-truth block configuration (**Ground truth** was **human-annotated**).

Multi-Model Benchmark Results



Model	Size	Mean Response Time
gemma3:12b	8.1GB	1.65s
qwen2.5:14b	9.0GB	2.13s
gemma3:27b	17GB	3.43s
qwen2.5:32b	20GB	12.22s
command-r:35b	19GB	12.99s

Error Type Distribution



Future Directions

Broader Impact:

- Voice-driven robot interfaces **lower the barrier to automation** for non-expert users in manufacturing, warehousing, and assistive fields
- Fully local inference keeps speech + task data on-device, **preserving user privacy**

Future Work:

- **Spatial fine-tuning** to address errors caused by coordinate and axis confusion
- Vision-language model integration
- Extend beyond block rearrangement to **general-purpose** task-level robot instruction