A Conversational Interface for Human-Robot Collaborative Manipulation?
Real-Time Conversational Interface for Robot Manipulation Using Edge AI: A Distributed LLM-VLM Coordination Framework?
Web-Based Conversational Interface for Robot Manipulation Using Distributed Edge AI Models?

Arash Ghasemzadeh Kakroudi[1] and Roel Pieters[2]

***Abstract*— This paper presents a human-robot collaboration interface using natural language, distributed LLMs, and vision-language models for real-time manipulation tasks.**

I. INTRODUCTION

TBD

II. RELATED WORK

A. Language-Grounded Robotics

B. Human-Robot Interface Design

C. Distributed Robotics Architectures

III. METHODOLOGY

A. Distributed Hardware Architecture

B. Local Model Deployment and Privacy

C. Multi-Agent Coordination Framework

D. Natural Language Processing Pipeline

E. Vision-Language Grounding

F. Real-Time Distributed Communication

IV. EXPERIMENT

This section presents a comprehensive evaluation of the xyz (system's name) system through quantitative performance metrics, user studies, and challenging real-world scenarios. Our evaluation methodology combines technical benchmarking with human-robot interaction assessment to validate both system capabilities and user experience.

*This work was funded by the PERFORM project under grant agreement no. XXXX.

[1]Arash Ghasemzadeh Kakroudi is a Researcher with the Faculty of Engineering and Natural Sciences, Tampere University, 33720 Tampere, Finland. Email: arash.ghasemzadehkakroudi@tuni.fi

[2]Roel Pieters is Professor of Intelligent Machines and Robotics, Faculty of Engineering and Natural Sciences, Tampere University, 33720 Tampere, Finland. Email: roel.pieters@tuni.fi

A. Performance Evaluation Protocol

We evaluate our system across multiple dimensions to ensure robust validation:

- **Success rate analysis**: Measured across 20? pick-and-place trials
- **Temporal performance**: End-to-end latency from natural language input to task completion
- **Component timing breakdown**: Individual analysis of LLM processing on Jetson AGX Orin, vision detection, and motion execution phases
- **Complex linguistic understanding**: Evaluation using challenging prompts such as *"Pick up the metallic tool specifically designed for driving or tightening screws"*
- **Repeatability assessment**: Consistency testing through 5 repeated trials across 5 different objects (25 total evaluations)

B. Spatial Reasoning and Multi-Object Scenarios

To assess the system's spatial understanding and multi-object manipulation capabilities:

- **Single object manipulation**: Baseline performance with isolated objects
- **Multi-object selection**: Disambiguation between multiple visible objects (*"pick the red cube, not the blue one"*)
- **Complex spatial relationships**: Advanced spatial reasoning tasks (*"place the red cube to the left of the blue one"*)
- **Overlapping object challenges**: Performance degradation analysis in cluttered environments
- **Relative positioning tasks**: Evaluation of spatial preposition understanding (above, below, beside, between)

C. Human Subject Evaluation

We evaluate system usability through a user study with 12? participants (4 novice, 4 intermediate, 4 expert users):

- **Performance metrics**: Task success rate and completion time
- **Usability survey**: System Usability Scale (SUS) questionnaire (need to design the survey if we want to include this)

- **User feedback**: Post-task interviews on system transparency and trust (need to design interview questions if we want to include this)

D. Statistical Analysis

We ensure reliable results through:

- **Adequate sample sizes**: Sufficient trials for meaningful statistical analysis
- **Error reporting**: Confidence intervals for all performance metrics
- **Significance testing**: Standard statistical tests (t-tests, ANOVA) where appropriate

E. Distributed System Performance

We evaluate the distributed architecture's effectiveness across hardware components:

- **Network latency analysis**: ROS2 topic communication delays on Jetson AGX Orin
- **Local inference performance**: LLM processing times on dedicated Jetson hardware
- **Offline operation validation**: System functionality without external network dependencies
- **Multi-device coordination**: Synchronization effectiveness across distributed nodes
- **Resource utilization**: CPU, GPU, and memory usage on Jetson during inference tasks

V. CONCLUSIONS / RESULT

A. Key Findings and Results Summary

B. System Contributions and Novel Aspects

C. Limitations and Future Work

D. Broader Impact and Applications

APPENDIX

- Detailed system configuration files
- Complete ROS2 topic specification
- Hardware setup instructions
- Additional experimental data and plots
- User study questionnaire details

ACKNOWLEDGMENT

- Collaborators and advisors
- Hardware and software providers

REFERENCES

[1] TODO: Add reference here. Probably using Mendeley?
[2] TODO: Add reference here. Probably using Mendeley?
[3] TODO: Add reference here. Probably using Mendeley?
[4] TODO: Add reference here. Probably using Mendeley?
[5] TODO: Add reference here. Probably using Mendeley?
[6] TODO: Add reference here. Probably using Mendeley?
[7] TODO: Add reference here. Probably using Mendeley?
[8] TODO: Add reference here. Probably using Mendeley?
[9] TODO: Add reference here. Probably using Mendeley?
[10] TODO: Add reference here. Probably using Mendeley?
[11] TODO: Add reference here. Probably using Mendeley?
[12] TODO: Add reference here. Probably using Mendeley?
[13] TODO: Add reference here. Probably using Mendeley?
[14] TODO: Add reference here. Probably using Mendeley?
[15] TODO: Add reference here. Probably using Mendeley?
[16] TODO: Add reference here. Probably using Mendeley?
[17] TODO: Add reference here. Probably using Mendeley?
[18] TODO: Add reference here. Probably using Mendeley?
[19] TODO: Add reference here. Probably using Mendeley?
[20] TODO: Add reference here. Probably using Mendeley?
[21] TODO: Add reference here. Probably using Mendeley?
[22] TODO: Add reference here. Probably using Mendeley?