NMM-HRI: Natural multimodal Human-Robot Interaction with Voice and Deictic Posture via Large Language Model

Yuzhi Lai[1], Shenghai Yuan[2*], *Member, IEEE*, Youssef Nassar[3], Mingyu Fan[4], Atmaraaj Gopal[5], Arihiro Yorita[6], Naoyuki Kubota[7] and Matthias Rätsch[3]

***Abstract*—Translating human intent into robot commands is crucial for the future of service robots in an aging society. Existing Human-Robot Interaction (HRI) systems relying on gestures or verbal commands are impractical for the elderly due to difficulties with complex syntax or sign language. To address the challenge, this paper introduces a multimodal interaction framework that combines voice and deictic posture information to create a more natural HRI system. Visual cues are first processed by the object detection model to gain a global understanding of the environment, and then bounding boxes are estimated based on depth information. By using a large language model (LLM) with voice-to-text commands and temporally aligned selected bounding boxes, robot action sequences can be generated, while key control syntax constraints are applied to avoid potential LLM hallucination issues. The system is evaluated on real-world tasks with varying levels of complexity using a Universal Robots UR3e manipulator. Our method demonstrates significantly better HRI performance in terms of accuracy and robustness. To benefit the research community and the general public, we will make our code and design open-source.**

***Index Terms*—Human-robot Interaction, Intent recognition, multimodality perception, Large Language Models**

I. INTRODUCTION

With an aging population, the demand for efficient caregiving solutions is growing, yet labor costs continue to increase. Robotics manipulators have shown promise in alleviating these challenges by automating tasks that traditionally require human intervention [1]. However, one of the most critical and unresolved issues is ensuring effective Human-Robot Interaction (HRI) for elderly users. For HRI systems to truly benefit this population, they must offer interaction methods that are intuitive and natural, resembling everyday human communication. Current systems often rely on humans to memorize complex language syntax or master complex hand gestures [2], which are impractical for the elderly. This highlights the urgent need for a simpler yet highly effective method that allows robots to understand and execute commands from elderly users with ease and reliability.

In the past year, large language models (LLMs) have emerged as promising tools for HRI [3]. Their advanced reasoning and language capabilities make them promising for improving communication between humans and robots. However, directly applying LLMs to HRI presents several challenges. First, LLMs often require users to input detailed and structured text commands, which can be tedious and difficult to understand. Second, without integrated sensing capabilities, LLMs struggle to comprehend the environmental context or specific actions, limiting their effectiveness in real-world applications. Finally, LLMs are prone to hallucinations, generating inaccurate or unsafe responses, which can lead to harmful outcomes when used in control systems without close monitoring. These challenges highlight the need for a more robust integration of LLMs into HRI systems.

To address the challenges, we introduce an **N**atural **M**ulti-**M**odal fusion-based **HRI** framework (NMM-HRI) that recognizes voice and posture, enabling users to convey their intentions to robots. We use simple and intuitive verbal language to compile sets of actions, while deictic postures identify objects or locations for interactions. By combining voice commands with deictic postures, our approach resolves ambiguities in language-based systems, reducing cognitive load in gesture-based systems and providing a more intuitive and natural interaction experience. Additionally, we incorporate an LLM to compile these actions and goals, generating robot action sequences. Unlike rule-based systems or simpler models, LLMs leverage their extensive reasoning ability to handle complex contextual understanding and generate the most reasonable action sequences. The generated action sequences undergo further adjustment of the structure of the output response to ensure structural consistency for safety purposes. Our method allows for the efficient construction of complex sequences of control actions, surpassing the speed of previous benchmarks by almost 50%. Our main contributions are

* Corresponding Author. This work is supported by a grant of the EFRE and MWK ProFö-R&D program, no. FEIH_ProT_2517820 and MWK32-7535-30/10/2. This work is also supported by "CAIpirinha - Conversational AI and Personalized Interaction for Risk-aware Navigation with Human Awareness" Förderkennzeichen: BW7_1030/02, Funding Program "Invest BW - Innovation III". This work is additionally supported by the National Research Foundation, Singapore, under its Medium-Sized Center for Advanced Robotics Technology Innovation.

[1]University of Tuebingen, Geschwister-Scholl-Platz, 72074 Germany. yuzhi.lai@uni-tuebingen.de

[2]Nanyang Technological University, 50 Nanyang Avenue, Singapore 639798, shyuan@ntu.edu.sg

[3]University Reutlingen, Alteburgstraße 150, 72762 Germany. {name.surname}@reutlingen-university.de

[4]Donghua University, 849 Zhongshan West Street, Shanghai 200051, fanmingyu@dhu.edu.cn

[5]Neura Robotics GmbH, 44 Gutenbergstraße, Metzingen 72555, atmaraaj.gopal@neura-robotics.com

[6]Kwansei Gakuin University, 1-155 Uegahara 1bancho, Hyogo 662-8501

[7]Tokyo Metropolitan University, Tokyo, Hachioji, Minamiosawa, 1-ch.-1

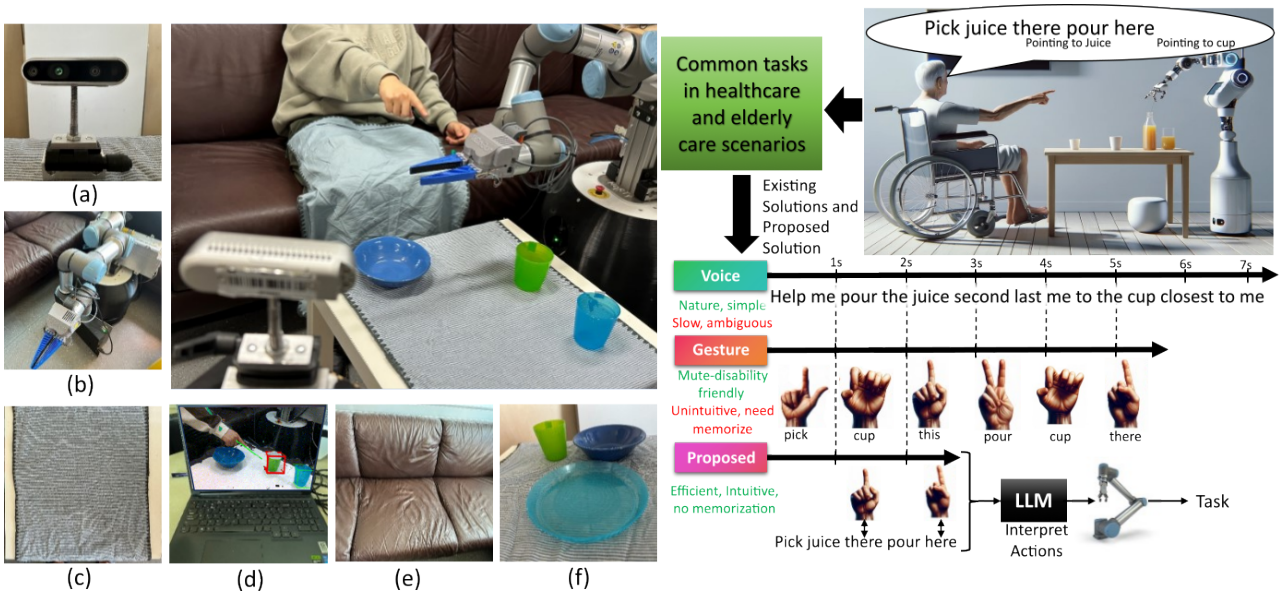

Fig. 1: Proposed voice-posture fusion HRI method has superior efficiency and requires no memorization of key syntax, which is ideal for elderly and healthcare applications. (a) Depth camera, (b) Robot manipulator, (c) Robot operating space, (d) Visual feedback, (e) User space, (f) Objects for experiment.

summarized below:

- We propose NMM-HRI, a parallel multimodal HRI method for robot manipulation. It efficiently constructs complex temporal control sequences using simple parallel inputs, processed by LLM to generate feasible actions.
- Our proposed system seamlessly generates robot control sequences through language, posture, and environmental input. This is achieved by structuring the output response tokens to mitigate LLM hallucination issues in the HRI setting, ensuring safety.
- We benchmark our system against state-of-the-art HRI methods, showcasing strong performance with minimal syntax token memorization and rapid input speed. For the benefit of the community, our system design, algorithms and solutions will be open-source at https://github.com/laiyuzhi/NMM-HRI.

II. RELATED WORK

As perception and navigation solutions advance, robots are increasingly integrated into daily life. However, robots designed for complex tasks, such as surgical [4] and rehabilitation applications [5], face stringent safety requirements and must be validated for many years before widespread use in real world scenarios. In contrast, service robots, particularly for elderly and patient care, demonstrate greater immediate potential. However, current service robot systems often rely on single-modal inputs, such as pure hand gestures [2], voice commands [6], or combinations of body language such as hand gestures and body postures [7]. Hand gesture-based HRI methods [2], for example, frequently use the Leap Motion sensor to detect hand movements and translate them into commands. However, these sensors have a limited field of view, restricting their applicability to broader environments. To address this, some researchers [7] have introduced Kinect cameras to extend the range of the sensor and detect both human posture and gestures. Nonetheless, for elderly users, accurately memorizing and executing the required gestures for commands remains a significant challenge, limiting the effectiveness of these approaches. Voice commands have always been a natural candidate for HRI problems. People often employ various approaches, such as LSTM, hidden Markov models, or an attention-based encoder-decoder network [8], to parse voices into actionable commands. One good example is the Amazon Alexa AI assistant, which can turn on and off lights with voice commands. However, controlling the voice-activated robot with a higher degree of freedom presents several challenges, including the ambiguity in describing the scene and the potential for sentences to have multiple interpretations.

multimodal approaches have great potential in generating accurate HRI responses. Early approaches [9] only allow simple command syntax, which dramatically limits its applications. In recent years, wearable mixed reality (MR) devices [10], such as the Apple Vision Pro, have demonstrated their capabilities as HRI tools. They often come equipped with multimodal inputs like gaming joysticks, gaze detection, head orientation estimation, and voice commands. However, these methods have several issues, including: (1) MR takes a considerable amount of time for humans to learn, with a steep learning curve; (2) Gaze and head-orientation-based HRI often require individualized calibration solutions to infer intentions with reasonable accuracy, thus introducing more

problems. (3) The devices tend to be excessively heavy, posing a challenge for the average user, let alone individuals who are elderly or unwell. (4) They can induce feelings of dizziness and nausea. For elderly individuals or those in need of medical attention, the use of MR devices for cyber-retirement is unlikely to be well-received.

Several other methods have incorporated multimodal inputs as redundant systems to enhance fail-safety. Although some claim to offer universal approaches for multimodality in HRI [11], these are often limited to late fusion frameworks that only partially address failure scenarios. Other modalities, such as electromyography, facial expressions, and voice signals [12], have been combined to control devices like wheelchairs through redundancy. However, these combinations are often impractical due to the variation in electromyographic signals and facial expressions between users, making it impossible to generalize their use effectively.

Existing multimodal approaches have yet to meet expectations for general use, particularly for elderly users or patients who struggle with complex sign or command languages. A major challenge is the ineffective fusion of different information sources. Recent works on Visual Language Models (VLMs) [13] have been proposed to address this issue in HRI. However, these models are often application-specific or tailored to specific environments, lack comprehensive benchmarks against traditional methods, and require substantial GPU resources for processing. In contrast, large language models (LLMs) offer strong reasoning and emergent capabilities while maintaining reasonable computational requirements, making them promising candidates for effectively fusing and processing multimodal information.

Most research in robot action generation involves predefined domain searches, unconstrained exploration, behavior trees, and Bayesian inference. In one study, the authors integrated LLMs into robot action generation, defining a pipeline that converts human intentions into robotic action sequences using prompts and task-relevant APIs [14]. This approach provides a more intuitive and convenient method for user interaction and robot control. However, LLMs cannot directly obtain information from sensors, and Visual Language Models (VLMs) often require significant computational resources [13]. Furthermore, since LLMs are prone to hallucination, combining LLM with VLM increases the likelihood of errors and mistakes, often necessitating multiple trials for successful execution.

III. PROBLEM DEFINITION

The goal of our proposed solution is to derive a parallel multimodal command sequence, which is then translated into robotic action sequences through the use of an LLM. To achieve that, we need to define the problem and the set of mathematical representations.

A. Problem Formulation

Let $\Xi(.)$ denote the object prior information represented by object class κ and object representation $\mathcal{O} \in \mathbb{R}^5$ including 3D position $\xi \in \mathbb{R}^3$, height $h \in \mathbb{R}$ and width $b \in \mathbb{R}$. The $q(t) \in \mathbb{R}^7$ represents the state of the robot end-effector at time t, including the 6D pose and the opening angle of the gripper. The prior observation tuple of the environment $\mathcal{S}$ can be constructed by $\mathcal{S} = (\Xi(\kappa, \mathcal{O}),\ q(t))$.

To control the manipulator $q(t)$ under constraint $\mathcal{S}$, it is essential to infer complete human intention I from a set of sparse keywords $\mathcal{C}$, which consist of object references $I_{\mathbb{O}}$ and action references $I_{\mathbb{A}}$. The NMM-HRI system uses audio and RGBD sensors to generate time series verbal instruction sets $\mathcal{V}$ and human postures $\mathcal{B}$. Object references $I_{\mathbb{O}}$ specify the target object to be interacted with according to $\mathcal{B}$, while action references $I_{\mathbb{A}}$ indicate type of action to be performed with object based on verbal command $\mathcal{V}$.

Therefore, the action intention can be considered as a mapping function from verbal language $\mathcal{V}$ to action intention $I_{\mathbb{A}}$, defined by $\mathcal{M} : I_{\mathbb{A}} = \mathcal{M}(\mathcal{V})$. Similarly, the object intention $I_{\mathbb{O}}$ can be represented by the posture reference $\mathcal{B}$, which interacts with environmental observations $\mathcal{S}$. This representation is defined by the mapping function $\mathcal{P}$, where $I_{\mathbb{O}} = \mathcal{P}(\mathcal{B}, \mathcal{S})$.

B. Parallel multimodal Command Sequence

Action and object intentions are derived from parallel multimodal command sequences and scenes. Below, we outline the components of the multimodal command sequence.

- **Verbal class command:** This command defines the specific class relevant to object intention. An object will only be detected if its class κ corresponds to verbal class command.
- **Verbal action command:** This command is assigned to various intended actions. Two distinct actions can be combined to construct a complex temporal movement, such as first picking up a cup and then pouring water into a bowl.
- **Verbal pronoun command:** This command, such as *this*, *there* or *that*, is employed in conjunction with deictic posture. When the demonstrative pronoun is recognized by the system, it records the location of the object that has been selected through the deictic posture.
- **Verbal metric command:** This optional command enhances input verbal information. This command could include variables such as the angle of inclination for a pouring action or the speed of various actions.
- **Deictic posture:** This specific posture is instrumental in aiding users to select an object within a scene as the object intention. By human skeleton detection, we obtain the deictic posture, r, representing the direction of the user's right forearm. The distance of an object $\Xi_i \in \Xi$ in the scene $\mathcal{S}$ with 3D position ξ_i to the vector r is outlined as $d_i(r, \xi)$:

$$d_i = \sqrt{\frac{\left|(r_2 - r_1) \times (r_1 - \xi_i)\right|^2}{\left|r_2 - r_1\right|^2}}, \tag{1}$$

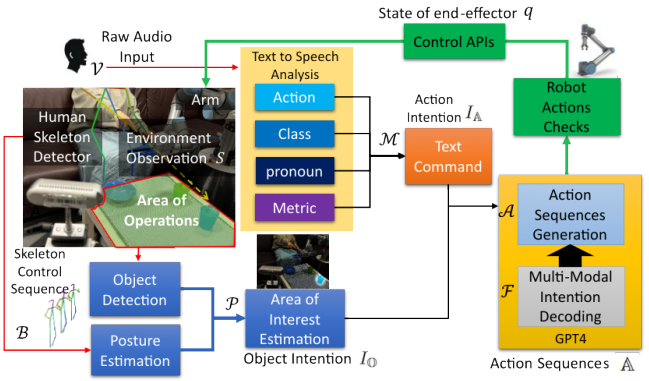

Fig. 2: System Overview. $\mathcal{V}$ represents voice command, $\mathcal{B}$ represents human posture, $\mathcal{M}$ is mapping verbal features to action intention $I_{\mathbb{A}}$, $\mathcal{P}$ is mapping human posture and environment observation $\mathcal{S}$ into object intention $I_{\mathbb{O}}$. GPT4 decodes the multimodal commands and generates the action sequences $\mathbb{A}$. Finally, the state of end-effector q is changed by the control APIs.

where r_1 and r_2 are two random points from r. We specify the object closest to the deictic posture as the object intention, so the mapping $\mathcal{P}$ can be rewritten as:

$$\begin{aligned} I_{\mathbb{O}} &= \mathcal{P}(\mathcal{B},\ \mathcal{S}), \\ &\triangleq \mathcal{P}(r,\ \xi). \end{aligned} \tag{2}$$

Mapping $\mathcal{M}$ converts all inputs from verbal language into text and completes a query task to find the action intention $I_{\mathbb{A}}$ corresponding to the verbal action command. Through adjustment of the multimodal command sequence $\mathcal{C}$, we can derive the human intention I, which encompasses the object intent $I_{\mathbb{O}}$, action intention $I_{\mathbb{A}}$, and metric parameter ω.

$$\begin{aligned} I &\triangleq \mathcal{F}(\mathcal{C}), \\ &= \mathcal{F}(I_{\mathbb{A}},\ I_{\mathbb{O}},\ \omega). \end{aligned} \tag{3}$$

The function $\mathcal{F}$ represents the encoding process from a multimodal command sequence to human intention, as shown in Fig. 2. In our work, GPT4 [15] is utilized to decode command sequence $\mathcal{C}$ into intention I. The robotic action sequence $\mathbb{A}$ is derived from intention I with mapping $\mathcal{A} : \mathbb{A} = \mathcal{A}(I)$. The mapping is also accomplished using GPT4. Finally, GPT4 uses the action sequences $\mathbb{A}$ passed through the check to control the state of the end-effector $q(t)$ to fulfill the human intention I.

C. Construction of Complex Command Sequence

Simple robotic actions require only the specification of the action type, without any additional parameters or object dependencies. An example of this is *go initial position*. For actions like *Pick up a cup*, it is necessary to specify both action and object intention. To achieve more complex temporal control, such as *picking up a cup and then tilting it at a 90-degree angle to pour into a bowl*, our multimodal command sequence allows for the construction of two subordinate commands. In this way, the multimodal command sequence becomes:

$$\mathcal{C} = \{\mathcal{C}_1, \mathcal{C}_2\} = \{I_{\mathbb{A}1},\ I_{\mathbb{O}1},\ I_{\mathbb{A}2},\ I_{\mathbb{O}2},\ \omega\}, \tag{4}$$

where, $\mathcal{C}_1$ and $\mathcal{C}_2$ represent the respective subordinate commands, with each subordinate command encompassing its own action intention $I_{\mathbb{A}}$ and object intention $I_{\mathbb{O}}$. Our system contains the following set of robotic actions with increasing complexity:

- Without object dependency: go initial position, throw, flush, etc.
- With object dependency: pick, put, pour, clean, push, etc.
- Combination of subordinate command sequences: pick + put, pick + pour, pick + throw, pick + go initial, etc.

IV. Methodology

Fig. 2 illustrates the overall system, designed for indoor healthcare or elderly care scenarios where the robot performs tasks based on human input. To enable concise and intuitive interactions, the system integrates several subsystems, including speech-to-text conversion, object detection, posture detection, and action sequence generation and execution. These components work together to ensure that the robot can accurately interpret and carry out the user's commands.

A. Speech-to-Text Conversion

To understand human verbal commands, a speech-to-text module is essential. This module converts the raw audio input into meaningful elements such as class references, pronoun references, intended actions, and metric parameters. Unlike traditional systems that process complete sentences, our approach primarily handles fragmented commands, often consisting of partial words combined with posture cues. This imposes specific constraints on the tools used. After comparing various methods [16], we selected the VOSK [17] for its ability to process partial input and distinguish between different speakers.

B. Object Detection

The object detection model extracts both bounding boxes and object classes using visual cues. We evaluated a few models such as YOLOv5 [18], YOLOv6, YOLOv8, and YOLO-World [19], selecting YOLO-World for its real-time open vocabulary detection capabilities. Our results show that YOLO-World accurately recognizes everyday objects (i.e., shampoo, mug, bottle, scissors) with high reliability. The 2D position of the object within the RGB image is represented by the center of its bounding box. Following object detection, a coordinate transformation step uses the depth map aligned with the RGB image and the 2D object information to determine the object's 3D representation, $\mathcal{R}$, in the camera frame. The 3D position ξ is then used to calculate

the distance, as described in Eq. 1. The width of the object b is used to calculate the opening angle of the gripper. The height of the object is used to calculate the collision-free trajectory.

C. Deictic Posture Detection

The deictic posture represents a distinct type of static gesture used for inferring object reference, as referenced in Eq. 2 and Eq. 1 The system captures the upper body of the human using an RGBD camera placed at an appropriate distance. Subsequently, the 2D human skeletons are tracked from the RGB image using OpenPose [20]. Then, the 3D human skeletons $\mathcal{B}$ are estimated within the camera frame based on the aligned depth map. We define the intention/direction line r of the right forearm as the deictic posture. For cases where individuals cannot move their arms, an additional mobile App connects with RealSense cameras over a local area network to obtain the reference direction directly from camera view, as shown in Fig. 3.

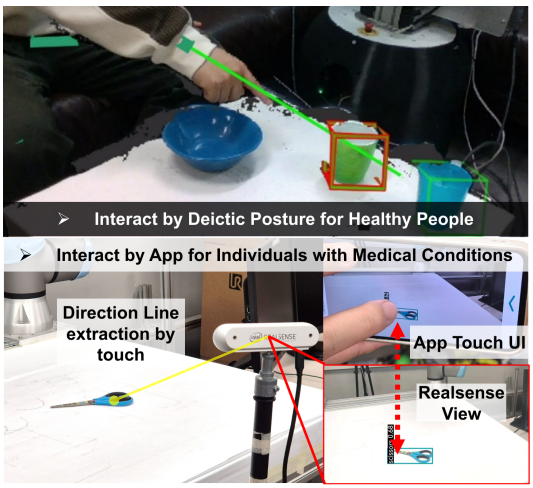

Fig. 3: Alternative ways of finding object reference.

The object reference can be calculated in each frame, but its information is only available when the user points to a detected object, the category of which is defined with a verbal class command and a verbal pronoun command is spoken. The accuracy of the deictic posture is evaluated in a separate experiment later.

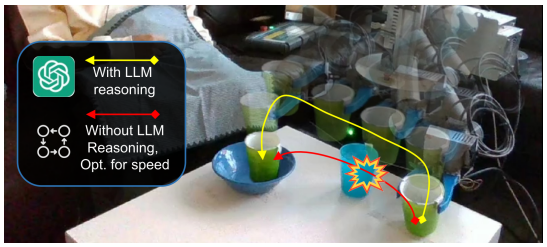

Fig. 4: Collision-free trajectory generation.

D. Action Sequences Generation and Execution

The human intention I is encoded into a robot action sequences $\mathbb{A}$ through the mapping $\mathcal{A}$. Task planning and action sequence generation often require extensive domain knowledge about the state of robots and constraints. To streamline this process, we employ GPT4 to encode this high-level policy. We establish constraints on the output response tokens of the LLM by crafting prompts and setting constraints. This approach helps minimize the issue of hallucination in LLM. As illustrated in Fig. 5, the prompt consists of three parts:

1) Basic API Constraints: outlines the functionalities of APIs that can be utilized in task planning. LLM can only utilize these APIs and basic Python libraries such as numpy to build robot action sequences.
2) Action Definition: delineates the execution methodology for each action. Non-technical users have the flexibility to define new functionalities or modify the execution of each action in prompts using natural language.
3) Example Tasks: demonstrates how similar tasks are executed and guides the strategies for task planning using LLM. With this constraint, LLM will imitate this example task.

Furthermore, we have integrated state feedback from the environment, requiring the robot to determine whether an object is already within its grasp before performing a pick task. Once the prompt and human intentions are processed, the LLM generates the corresponding action sequences and code. The generated action sequence is verified by LLM to ensure collision-free, as shown in Fig. 4. For example, in the water pouring task illustrated in Fig. 5, LLM first understands the multimodal commands and then generates an action sequence based on the gripper state, as well as the height and position of the object. These sequences are restricted to actions defined by the basic API constraints and the action definition.

E. Human-Robot Interaction

Human-robot interaction in our system is enabled through parallel multimodal command sequences, where the user conveys intentions via verbal commands and deictic postures processed by $\mathcal{F}(.)$, with visual feedback (e.g., detected posture, selected object) provided by the graphical user interface. The process of encoding a multimodal command sequence into human intention involves the following steps:

1) The user begins by encoding the action intention $I_{\mathbb{A}}$ through a verbal action command. If this action requires object dependency, the sequence continues; otherwise, it is terminated with the *finish* command.
2) For object-dependent actions, the user encodes the class of the object (verbal class command) followed by a demonstrative pronoun (verbal pronoun command).
3) Simultaneously, the system encodes the target object using deictic posture r and object position ξ, computed by Eq. 1. The object selected during the pronoun command becomes the object intention $I_{\mathbb{O}}$.

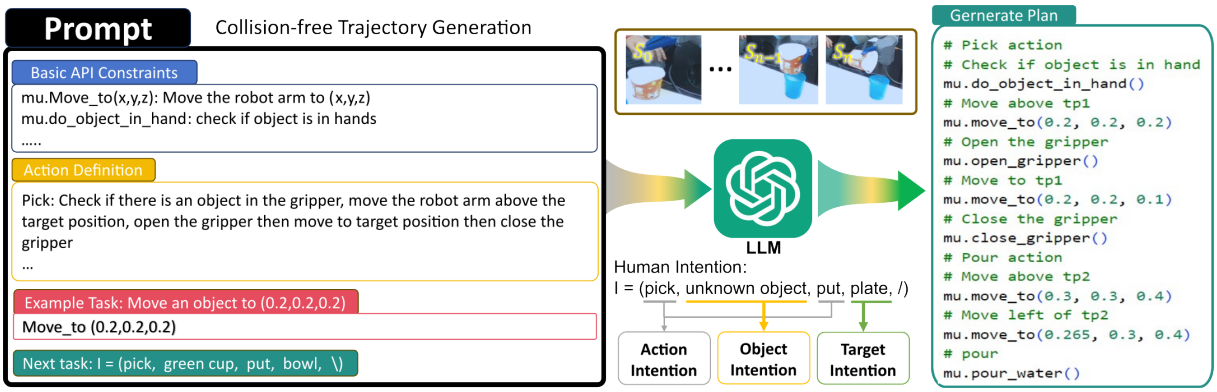

Fig. 5: The prompt is segmented into three sections: basic API constraints, action definition, and example tasks.

4) The metric parameters ω, such as the pour angle, can be encoded verbally by the user to refine the action intention.
5) Once the multimodal sequence is fully encoded, the user can either proceed with further subordinate commands or conclude the process with the *finish* command, resulting in the final encoded human intention.

V. EXPERIMENTAL SETUP

A. Perception and Manipulator Setup

Visual inputs are collected through the Intel Realsense D435i RGBD camera, while auditory signals are captured through a USB microphone. Our system has been trialed with individuals of varying ages in real-world environments, as illustrated in Fig. 1. The scene contains a Universal Robots UR3e robot manipulator and several manipulation objects. An Intel Realsense D435i RGBD camera opposing the robot is used for object detection and deictic posture detection. A laptop with a Nvidia 4060 GPU was used to process the multimodal data and show feedback images in Rviz.

B. Description of Experimental Scenarios

We designed a set of typical manipulation experiments of the increasing human intention complexity according to eq. 3 and eq. 4 with $I = \mathcal{F}(I_{\mathbb{A}1},\ I_{\mathbb{O}1},\ I_{\mathbb{A}2},\ I_{\mathbb{O}2},\ \omega)$:

- $(home,\ -,\ -,\ -,\ -)$, $(throw,\ -,\ -,\ -,\ -)$
- $(pick,\ cup,\ -,\ -,\ -)$
- $(push,\ plate,\ -,\ -,\ near)$
- $(pick,\ cup,\ put,\ bowl,\ -)$, $(pick,\ cup,\ pour,\ cup,\ -)$
- $(pick,\ cup,\ pour,\ bowl,\ ang = 90°)$
- **Multi-step tasks:** pick and throw rubbish, add water and pass, pour muesli and add milk.

In our proposed method, we primarily evaluate the performance by measuring the user interaction time and success rate for each specific scenario. An additional goal is to evaluate the intuitiveness of the HRI system. The execution of all these tasks is depicted in the accompanying videos.

VI. RESULTS AND DISCUSSION

We evaluated the performance of our system through a series of challenging experiments designed to assess criteria such as accuracy, user satisfaction, lighting robustness, and location robustness.

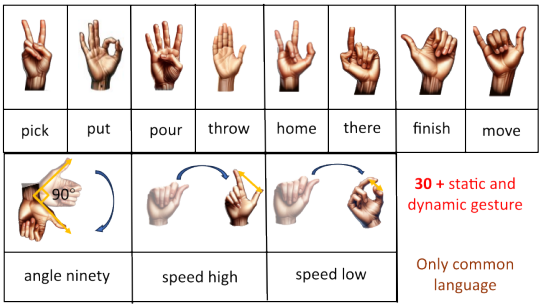

Fig. 6: List of Gestures used in gesture-based HRI system [2] and their corresponding verbal commands in our NMM-HRI experiments.

A. Baseline Selections

In our work, we compare our approach with other open-source SOTA unimodal HRI methods [2], [6] and the multimodal method [13]. The baselines were selected primarily on the basis of similarity in action sequences and intent. The two unimodal benchmarks are gesture-based [2] and NLP-based [6], respectively. The multimodal approach is based on VLM [13], where target objects are selected through dialogue. The gesture-based HRI method [2], which utilizes the Leap Motion sensor, captures hand structure at specific localizations within a narrow field of view. The NLP and VLM approach [6], [13] demonstrated the use of language commands to direct the actions of a robot with a speech-to-text pipeline. NLP-based methods employ word embeddings, attention mechanisms, and probabilistic reasoning to recognize objects described in natural language. In contrast, VLM-based methods utilize the CLIP model to resolve ambiguities in object selection and assist users in forming clearer expressions. However, each baseline method has certain limitations and received numerous complaints from participants:

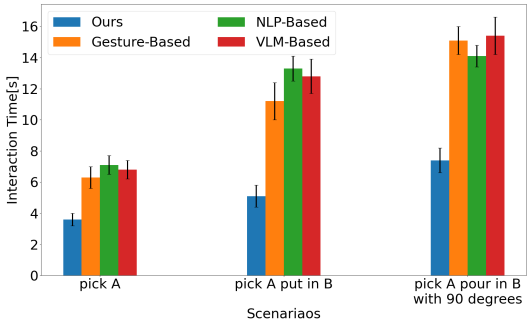

Fig. 7: Compared to other baseline HRI, our system required less time to input the same commands.

1) The Leap Motion's limited field of view requires maintaining the hand consistently above the sensor, which is impractical for the elderly and patients. 2) Natural language commands often struggle to differentiate between similar objects using simple descriptors. 3) Gesture-based HRI requires memorizing a complex set of gestures, which grows increasingly complicated as the command set expands, as shown in Fig. 6. Those are the gesture commands that we ask the participant to memorize. 4) VLM-based methods require gestures to indicate the general direction of a target object, followed by an interactive dialogue to select it. This increases interaction time, and distinguishing between similar objects in the scene remains a challenge.

B. Comparison of Efficiency and Intuitiveness

In the experimental setup, we used two cups of different colors, two bowls of different colors and a plate as manipulated objects. To evaluate the time required for user interaction, we asked participants to perform the same commands using different HRI methods. These tasks included picking up the assigned cup, picking up the assigned cup and placing it on the plate, and picking up the assigned cup and pouring it into the assigned bowl at a 90-degree angle. For the in-house HRI experiment, we recruited 27 participants from the local university, including 6 elderly. All participants were able to speak English and received verbal briefings on the interaction method, but did not undergo any training or trial. Fig. 7 shows the results of the experiment. The findings indicated that our system required 50.6% less time compared to hand gesture-based HRI, 53.3% less time compared to language-based HRI, and 54% less time than VLM-based HRI. This experiment demonstrates that, rather than relying on complex gestures to convey action intentions or ambiguous language (including interactive dialogue) to specify object intentions, our system utilizes simple multimodal commands. This approach significantly improves the efficiency of human-robot interaction (HRI) compared to the three baseline methods. Additionally, the system's wide field of view enables users to interact from a greater range of positions than is possible with the Leap Motion sensor.

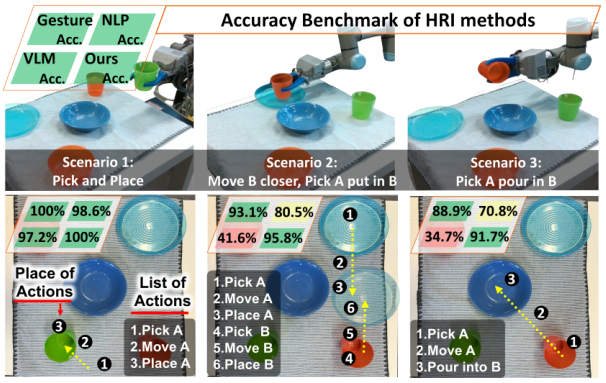

Fig. 8: Accuracy evaluation for various tasks across different approaches..

Overall, our proposed system is more efficient in terms of interaction in a cluttered environment.

C. Comparison of Accuracy

To evaluate the accuracy of action sequences in our proposed method, we designed a set of experiments using various sequences of events and compared them against several baseline methods. Accuracy was defined as the proportion of successfully executed commands (N_{executed}) to the total number of trials (N_{trials}), calculated as Accuracy $= (N_{\text{executed}}/N_{\text{trials}}) * 100\%$.

The actions were categorized into simple commands (e.g., pick and place), and causality commands (tasks involving cause-and-effect), andsequential commands (multi-step actions). The graphical results are presented in Fig. 8 Gesture-based methods often outperformed language-based methods for precise object referencing but struggled with more complex action sequences. Methods relying on VLM or NLP required highly descriptive input, making them less accurate and effective for tasks involving similar objects.

Our multimodal HRI approach simplifies interaction by eliminating complex gestures and reducing language ambiguity. Leveraging LLMs, it corrects misrecognized commands, interprets intent, and generates precise actions. This method effectively combines descriptive language with intuitive commands, ensuring high accuracy across diverse scenarios.

D. Comparison of Robustness

To evaluate the robustness of our solution, we conducted two separate experiments. The first experiment tested robustness in a cluttered environment, while the second evaluated performance under varying lighting conditions. In both cases, we are trying to simulate complex real-world challenges. Robustness, in this context, is defined as the ability of system to correctly interpret and execute human intentions under challenging conditions. It is calculated as the proportion of correctly detected intents (N_{correct}) to the total number of trials (N_{total}), given by Robustness $= (N_{\text{correct}}/N_{\text{total}}) * 100\%$.

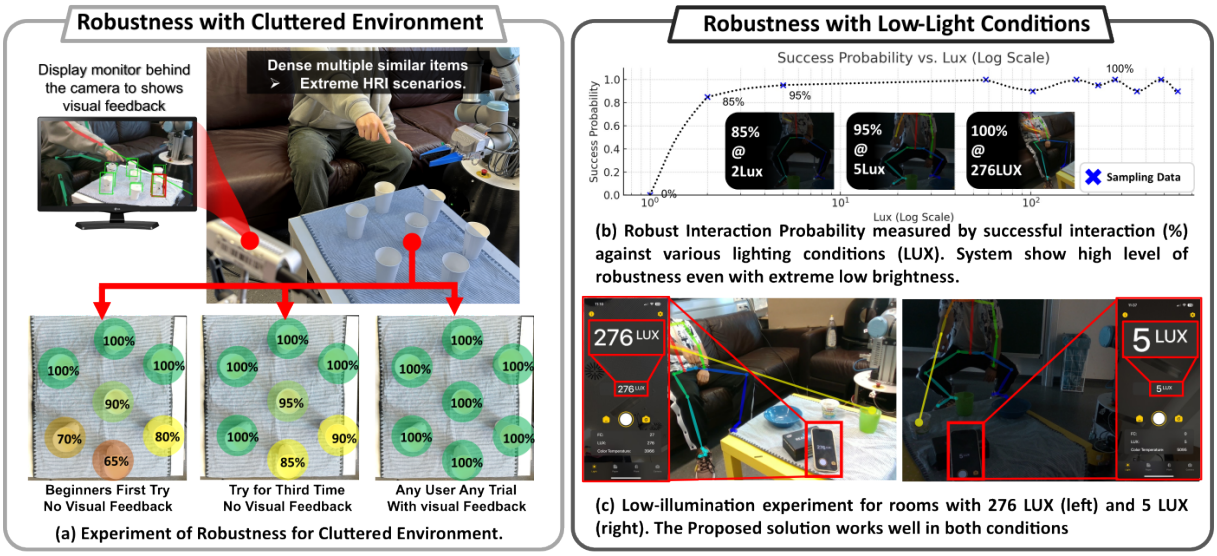

Fig. 9: System robustness evaluation under varying real-world constraints. Lighting affects all baseline methods equally.

Robustness Testing Cluttered Environment: This experiment assessed the robustness of the HRI system in distinguishing between multiple similar objects, a common challenge in real-world scenarios. Six cups, separated by 25 cm, were arranged on a table to evaluate the accuracy intent detection under cluttered conditions, as shown in Fig. 9(a).

To further test the robustness of the system, we conducted experiments with 27 participants divided into two groups. All participants received verbal instructions on using the NMM-HRI system. The first group completed three trials without visual feedback, demonstrating improved accuracy after brief learning sessions. The second group operated with visual feedback, achieving consistently high accuracy, highlighting the system's reliability in providing clear and intuitive interactions. The results reveal that the system's robustness is influenced by the precision of human skeleton detection and the quality of the point cloud generated by the depth camera. These findings underscore the system's capacity for reliable performance in complex, cluttered environments.

Robustness Testing in Low-Light Conditions: To assess the system's robustness under low-light conditions, we conducted tests across a lighting range of 1 to 600 lux, as shown in Fig. 9(b) and Fig. 9(c). The data indicate that the system's HRI accuracy in low-light conditions is comparable to its performance under normal lighting levels.

As illustrated in 9(b), the system effectively captures human intention when the light is above 1 Lux. These findings confirm the adaptability of the system for most typical lighting conditions. However, in extreme low-light scenarios (e.g., At 1 lux or complete darkness), vision-based methods fail unless thermal imaging is used. While thermal imaging could address this issue, its high cost makes it impractical for elderly care applications.

(a) Experiments for household tasks including clean table using towel(left) and flushing a toilet (left).

(b) Perception Experiments in the toilet, living room and kitchen. Deictic postures are estimated by OpenPose and objects are detected by YOLO-World.

Fig. 10: Experiments on adverse tasks and evaluations with diverse elderly participants and environments.

E. Real-world Action and Perception Trials

To evaluate the versatility of the system in real-world scenarios, we conducted a series of action and perception field trials. Beyond generating action sequences for intuitive single tasks (e.g., picking up an object at location A), as shown in Fig. 10(a), the LLM was tested on complex household tasks requiring higher-level reasoning. For example, when tasked with clearing a table using a towel, the LLM successfully generated action sequences, including locating the towel, picking it up, moving the end-effector to the table, and performing a wiping motion to clean the surface. This showcases the advanced reasoning capabilities of LLM in robotic applications.

Additionally, the system was tested in diverse environments common to elderly care centers and homes, demonstrating robust performance even when the lower body of the

user was obscured. In these scenarios, the system reliably fetched daily objects on demand, as shown in Fig. 10(b). These trials highlight the system's adaptability to varying environments and its practical usability in real-world settings.

VII. LIMITATION AND FUTURE WORKS

The method was tested in labs, homes, and elderly care centers with diverse participants, though most were fluent in English, introducing language bias. Further testing in hospitals and with non-English speakers is needed for accessibility. Regulatory delays have impacted hospital trials, but collaboration with healthcare professionals remains a priority.

NMM-HRI struggles in extreme low-light, limiting posture detection and object recognition in darkness. While thermal sensors could help, their high cost is impractical. Developing affordable specialized auxiliary sensors may be necessary.

Our method enhances object detection using YOLO-World, a robust open-vocabulary model. However, its bias toward common knowledge limits accuracy for medical items. A potential solution is a database allowing nursing staff to update the system via verbal identification and online adaptation. Future work will explore this approach.

VIII. CONCLUSIONS

In this work, we introduced a system that handles parallel multimodal inputs for HRI, accommodating diverse verbal features and dynamic postures, and integrates this information to generate action sequences executed through an LLM. Our system demonstrates excellent effectiveness, accuracy, and robustness under various conditions. We will make our code publicly available for the benefit of the community.

REFERENCES

[1] A. Di Lallo, R. Murphy, A. Krieger, J. Zhu, R. H. Taylor, and H. Su, "Medical robots for infectious diseases: Lessons and challenges from the covid-19 pandemic," *IEEE Robotics & Automation Magazine*, vol. 28, no. 1, pp. 18–27, 2021.

[2] P. Vanc, J. K. Behrens, K. Stepanova, and V. Hlavac, "Communicating human intent to a robotic companion by multi-type gesture sentences," in *2023 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2023, pp. 9839–9845.

[3] M. Gramopadhye and D. Szafir, "Generating executable action plans with environmentally-aware language models," in *2023 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2023, pp. 3568–3575.

[4] L. Zhang, M. Ye, P. Giataganas, M. Hughes, A. Bradu, A. Podoleanu, and G.-Z. Yang, "From macro to micro: Autonomous multiscale image fusion for robotic surgery," *IEEE Robotics & Automation Magazine*, vol. 24, no. 2, pp. 63–72, 2017.

[5] J. C. Pulido, C. Suarez-Mejias, J. C. Gonzalez, A. Duenas Ruiz, P. Ferrand Ferri, M. E. Martinez Sahuquillo, C. E. Ruiz De Vargas, P. Infante-Cossio, C. L. Parra Calderon, and F. Fernandez, "A socially assistive robotic platform for upper-limb rehabilitation: A longitudinal study with pediatric patients," *IEEE Robotics & Automation Magazine*, vol. 26, no. 2, pp. 24–39, 2019.

[6] S. Stepputtis, J. Campbell, M. Phielipp, S. Lee, C. Baral, and H. Ben Amor, "Language-conditioned imitation learning for robot manipulation tasks," *Advances in Neural Information Processing Systems*, vol. 33, pp. 13 139–13 150, 2020.

[7] O. Mazhar, S. Ramdani, B. Navarro, R. Passama, and A. Cherubini, "Towards real-time physical human-robot interaction using skeleton information and hand gestures," in *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2018, pp. 1–6.

[8] T. Asfour, M. Waechter, L. Kaul, S. Rader, P. Weiner, S. Ottenhaus, R. Grimm, Y. Zhou, M. Grotz, and F. Paus, "Armar-6: A high-performance humanoid for human-robot collaboration in real-world scenarios," *IEEE Robotics & Automation Magazine*, vol. 26, no. 4, pp. 108–121, 2019.

[9] I. El Makrini, S. A. Elprama, J. Van den Bergh, B. Vanderborght, A.-J. Knevels, C. I. Jewell, F. Stals, G. De Coppel, I. Ravyse, J. Potargent, J. Berte, B. Diericx, T. Waegeman, and A. Jacobs, "Working with walt: How a cobot was developed and inserted on an auto assembly line," *IEEE Robotics & Automation Magazine*, vol. 25, no. 2, pp. 51–58, 2018.

[10] T. R. Groechel, M. E. Walker, C. T. Chang, E. Rosen, and J. Z. Forde, "A tool for organizing key characteristics of virtual, augmented, and mixed reality for human–robot interaction systems: Synthesizing vam-hri trends and takeaways," *IEEE Robotics & Automation Magazine*, vol. 29, no. 1, pp. 35–44, 2022.

[11] S. Rossi, E. Leone, M. Fiore, A. Finzi, and F. Cutugno, "An extensible architecture for robust multimodal human-robot communication," in *2013 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2013, pp. 2208–2213.

[12] I. Moon, M. Lee, J. Ryu, and M. Mun, "Intelligent robotic wheelchair with emg-, gesture-, and voice-based interfaces," in *2003 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, vol. 4, 2003, pp. 3453–3458.

[13] S. Constantin, F. I. Eyiokur, D. Yaman, L. Bärmann, and A. Waibel, "Interactive multimodal robot dialog using pointing gesture recognition," in *Computer Vision - ECCV 2022 Workshops - Tel Aviv, Israel, October 23-27, 2022, Proceedings, Part VI*, ser. Lecture Notes in Computer Science, vol. 13806. Springer, 2022, pp. 640–657.

[14] S. Vemprala, R. Bonatti, A. Bucker, and A. Kapoor, "Chatgpt for robotics: Design principles and model abilities," *Microsoft Auton. Syst. Robot. Res*, vol. 2, p. 20, 2023.

[15] T. Brown, B. Mann, N. Ryder, M. Subbiah, J. D. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell, *et al.*, "Language models are few-shot learners," *Advances in neural information processing systems*, vol. 33, pp. 1877–1901, 2020.

[16] A. Trabelsi, S. Warichet, Y. Aajaoun, and S. Soussilane, "Evaluation of the efficiency of state-of-the-art speech recognition engines," *Procedia Computer Science*, vol. 207, pp. 2242–2252, 2022.

[17] X. Wang, H. Shen, H. Yu, J. Guo, and X. Wei, "Hand and arm gesture-based human-robot interaction: A review," in *Proceedings of the 6th International Conference on Algorithms, Computing and Systems*, 2022, pp. 1–7.

[18] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 779–788.

[19] T. Cheng, L. Song, Y. Ge, W. Liu, X. Wang, and Y. Shan, "Yolo-world: Real-time open-vocabulary object detection," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024, pp. 16 901–16 911.

[20] Z. Cao, T. Simon, S.-E. Wei, and Y. Sheikh, "Realtime multi-person 2d pose estimation using part affinity fields," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 7291–7299.