LLM-Driven Robotic Indoor Air Quality Monitoring: Advances in Natural Language Interaction and Decision-Making

Lin He, Saravanan Kanagaratnam, Lorena A. Zanandreis, and Min Deng

***Abstract*— We propose a novel framework for building indoor air quality monitoring that seamlessly integrates multiple large language models (LLMs) with a mobile robotic platform. A dedicated bridge node converts natural language commands into structured JSON messages, enabling non-experts to control navigation and sampling without manual coding. In addition, a mid-level verification and scheduling module—powered by a second LLM—ensures safe and reliable decision-making. Our findings highlight the potential of LLM-driven robotics for intelligent building management and environmental monitoring.**

I. Introduction

Maintaining a comfortable indoor environment is regarded as a critical function of buildings, as it can enhance the well-being and comfort of occupants [1]. Traditionally, indoor environment inspections involve manual data collection at various locations within a building. This process is labor-intensive, time-consuming, and prone to errors, resulting in inefficiency and fragmented data that complicates analysis and decision-making [2]. To address labor shortages and improve data collection efficiency on construction sites, there has been a strong push for the development of robotic technologies in the construction sector [3]. Despite rapid advances in robotic intelligence, construction sites remain complex and unstructured. Depending solely on robots for open-ended tasks like on-site data collection is often deemed an "impossible mission," highlighting the need for human-robot collaboration [4]. This highlights the urgent need for intuitive communication interfaces that enable seamless collaboration between data collectors and robotic assistants [5]. Nevertheless, the incompatibility of conventional building data collection workflows with highly mechanized robotic processes has significantly curtailed the research and development of robotic solutions [6].

Although human-robot collaboration in construction is well-studied, most existing research focuses on robot-centric tasks, providing limited flexibility in dynamic environments [7]. In human-robot collaborative construction teams, most robots currently operate at a relatively low level of autonomy [8]. Therefore, certain problems remain unaddressed with regard to handling unexpected events in on-site data collection: (1) How can human researchers directly query the robot in natural language about current air conditions (e.g., "What is the $PM_{2.5}$ concentration at this point?" or "Why is the concentration in the living room higher than in the corridor?"); (2) How can the robot instantly comprehend human-issued natural language commands and take action accordingly?; (3) Can the robot, when faced with unforeseen circumstances, provide feedback to humans and propose optional autonomous decision-making plans? Integrating LLMs and multimodal sensors into robots offers a precise, immediate approach to data collection tasks. The main challenge is creating a system that interprets language instructions, controls robots, and streams sensor data in real time for further analysis and action—redefining building monitoring and maintenance.

Although several studies focus on enhancing data collection and IAQ inspections by improving robotic navigation, robot-assisted IAQ monitoring still faces limitations—primarily its reliance on digital navigation [9], which hampers inspection efficiency and intelligence. Digital navigation, which relies on precise numerical inputs and geometric maps, requires measuring and inputting accurate coordinates for each object. Though accurate, this method is time-consuming and labor-intensive [10]. In contrast, Semantic navigation is more intuitive and efficient, letting inspectors use natural language commands. Voice interaction is considered the most natural form of human-computer communication [11]. Voice-based natural language instructions improve communication accuracy and efficiency. Instead of specifying exact coordinates, semantic navigation accepts high-level directives, reducing manual effort and boosting efficiency. It relies on object recognition, semantic reasoning, and real-time sensor data to remain accurate without coordinating input [12]. On the sensor side, Robotic systems with advanced sensing capabilities have proven versatile and effective in various domains, including occupancy detection, floor cleaning, surface defect defection, and indoor air monitoring [13]. Integrating specific sensors and enabling real-time data provision are therefore essential.

To address the above-mentioned constraints, this study develops a method for integrating LLMs and multimodal functions on a robot to facilitate real-time human-robot interaction and decision-making, aiming to achieve more intelligent and efficient IEQ inspections. The layered architecture comprises three module layers *"Fig. 1"*:

1) High-Level Decision Layer (LLM): Uses a Large Language Model (LLM) to parse human natural language commands and generate comprehensive task plans in the context of environmental information. The LLM receives human input, carries out language understanding, semantic inference, and knowledge retrieval (optionally interfacing with specialized knowledge bases or APIs), then outputs commands formatted in JSON.

Lin He is with the Department of Civil Engineering, Texas Tech University, Lubbock, TX 79409, USA (e-mail: lynn.he@ttu.edu).

Saravanan Kanagaratnam is with the Department of Civil Engineering, Texas Tech University, Lubbock, TX 79409, USA (e-mail: sakanaga@ttu.edu).

Lorena A. Zanandre is with the Department of Civil Engineering, Texas Tech University, Lubbock, TX 79409, USA (e-mail: lzanandr@ttu.edu).

Min Deng is with the Department of Civil Engineering, Texas Tech University, Lubbock, TX 79409, USA (fax: 806-742-3446; e-mail: mindeng@ttu.edu).

2) Mid-Level Safety Verification and Task Scheduling Layer: After receiving JSON commands from the LLM, a safety verification or planning-and-scheduling module checks logical validity and feasibility. This layer can integrate real-time multimodal data to repair or modify any “hallucinatory” or unreasonable commands generated by the LLM. Simultaneously, it can implement a multi-task scheduling mechanism to decide on priorities or queue tasks when multiple commands conflict or resources are limited.

3) Low-Level Robot Control Layer (RC): This layer handles the execution of specific motions and actions such as navigation or sensor operation. It receives the “approved/modified” JSON commands from the upper layer and invokes the relevant motion control algorithms (for instance, ROS navigation stack or an autonomous controller) for execution.

With a large language model, this framework enables non-programmer human workers to interact intuitively with robotic assistants, forming a three-way collaboration among the human operator, the LLM, and the robot. Not only can humans issue commands to robots, but the robots can proactively prompt the operator with questions or suggestions—relayed and refined by the LLM—to form a pipeline of continuous interaction. Mid-level verification and scheduling mechanisms significantly reduce the “uncertainty and potential risks” that arise from LLM-issued commands, thereby making the system more reliable. Robotic technology can therefore be safely and effectively integrated into building data collection tasks, enabling faster, better decisions in unexpected situations.

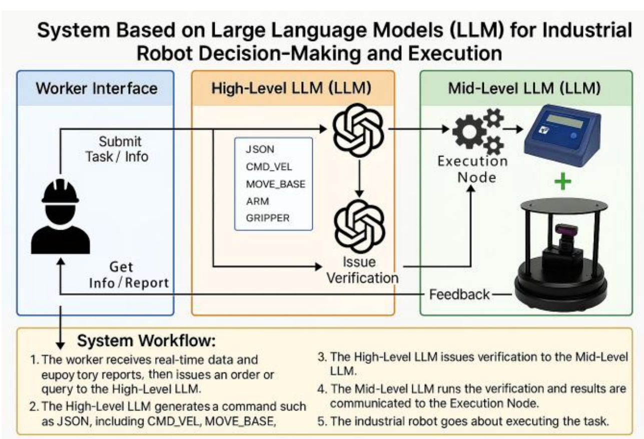

Figure 1. Frame diagram

II. System Architecture and Methods

A. Overall System Design

To implement an end-to-end pipeline for indoor air quality (IAQ) monitoring via natural language interactions, we developed a layered system architecture. The primary hardware platform is a TurtleBot4, equipped with (i) a built-in RGB-D camera for visual sensing and navigation, (ii) a DustTrak (DT) particle counter, and (iii) two CO_2 sensors, S1 and S2, providing real-time environmental data. For each measurement point, sampling was performed over a one-minute duration, with DT readings recorded at a resolution of one second.

At the software level, the system integrates a GPT-based LLM interface through a dedicated service node within ROS 2. This node mediates between high-level language instructions and the robot’s control framework. Specifically, the node translates GPT-generated commands (in JSON format) into actionable ROS 2 messages for navigation, sensor activation, and data collection. In turn, sensor measurements and robot status information are fed back into the LLM node or logged for subsequent analysis. This bidirectional information flow ensures that environmental data and user requests jointly inform how the robot selects its next measurement point or adjusts its path during IAQ inspections.

B. LLM Integration

1) GPT-based Service Node. We employed GPT as our primary large language model, interfacing with it through an API that is encapsulated in a custom ROS 2 service node. User input arrives as natural language text, which the GPT node processes via carefully crafted prompts. The node outputs JSON-formatted commands containing fields such as action type (e.g., “navigate,” “measure,” “report”), target coordinates, sensor parameters, or duration. The ROS 2 node then passes these JSON commands to dispatch corresponding instructions to lower-level controllers.

2) Prompt Design and Command Generation. To ensure robust and flexible command generation, we designed prompt templates that instruct GPT to produce JSON messages aligned with our robot’s control logic. For example, an operator could say, “Check the air quality near the main entrance,” prompting GPT to generate a JSON command specifying.

Such a structure allows us to map high-level linguistic tasks to specific operational sequences in ROS 2. The “main entrance” identifier can be translated into a set of coordinates if preconfigured in a map, and the “duration”: 60 parameter triggers a one-minute sampling routine using the DustTrak and CO_2 sensors. This modular approach ensures that GPT can flexibly generate commands for any recognized location or sensor usage scenario.

C. Natural Language Parsing & Task Planning

Upon receiving a user command, the system performs the following steps:

1) Semantic Parsing: GPT interprets the user’s natural language request, leveraging its language model to produce a semantic representation of the task.

2) Task Decomposition: Based on the parsed context, the GPT-based node decomposes the task into sub-actions such as navigation, measurement, and data reporting. These sub-actions are encoded in the JSON message.

3) Action Dispatch: The JSON commands are then forwarded to the motion control modules within ROS 2. If a multi-layer control structure is used, the high-level plan from GPT is reconciled with existing navigation stacks (e.g., ROS 2 Navigation) for path planning and obstacle avoidance. The robot’s embedded controllers handle real-time motion control,

while GPT remains responsible for updating the plan if new user instructions arrive or sensor data changes significantly.

This hierarchical approach enables the LLM to manage high-level decision-making and ensures reliable low-level execution through standard robotic frameworks.

D. Air Quality Sensing & Data Processing

We equipped TurtleBot4 with a DustTrak (DT) sensor, sampling particulate matter at a 1-second interval, and two CO_2 sensors (S1, S2), each operating over a one-minute sampling window at designated measurement points. Sensor calibration was performed prior to deployment, and basic outlier rejection was implemented to mitigate noise in raw measurements.

During each measurement cycle, sensor readings are stored with their corresponding robot pose (obtained from the onboard odometry or SLAM). This positional tagging allows us to correlate IAQ data with specific locations within the indoor environment. Additionally, these datasets can be processed offline or in real time to generate spatial distributions or heatmaps of particulate concentrations and CO_2 levels. Visualizing these data patterns facilitates targeted investigations of potential problem areas or anomalies.

E. Decision-Making Loop

After the robot gathers new air quality data, the following adaptive decision-making loop is triggered:

1) Data Analysis: The latest sensor readings—particulate matter and CO_2 levels—are passed to either GPT or an auxiliary algorithm for further examination.

2) Adaptive Task Planning: If elevated concentrations are detected in a certain region, the system can prompt GPT to suggest a refined plan. For instance, GPT may generate a JSON instruction to perform additional measurements around the hotspot or extend the sampling duration.

3) Exception Handling: In cases where sensor readings appear inconsistent or if the robot encounters unexpected obstacles, GPT can either consult the user for clarification (e.g., "Should I skip the current measurement point?") or adjust the plan automatically, subject to operator approval.

4) Execution Feedback: The mid-level ROS 2 nodes constantly monitor execution status, providing real-time progress updates and sensor feedback to the GPT service node, closing the loop for continual refinement of the IAQ inspection strategy.

Through this iterative process, the LLM-driven framework remains context-aware and capable of responding to unforeseen conditions, which is especially critical in dynamic indoor environments. By integrating natural language interaction, JSON-based command generation, and multi-sensor data processing, our system offers an intuitive and adaptive solution for performing IAQ monitoring tasks with minimal human intervention.

III. EXPERIMENTAL SETUP

A. Environment and Hardware

Our experiments were conducted in a laboratory *"Fig. 2"*, designed to simulate a typical office environment. The space has minimal obstructions aside from standard office furniture, providing ample room for robot navigation. As depicted in the layout, we arranged 39 sampling stations (labeled P1 to P39) at intervals of approximately 1–3 ft to capture a range of air quality data points room *"Fig. 3"*; the room door was kept open throughout the experiment to allow for natural airflow.

We used Turtlebot4 as the base robotic platform. The robot is equipped with:

• An on-board Intel® CPU (Quad-core, 1.8 GHz) and 4 GB RAM for local computation.

• A built-in RGB-D camera for environmental sensing and navigation.

• A DustTrak (DT) particle counter and two CO_2 sensors (S1 and S2), positioned so as not to obstruct the robot's mobility or camera field of view.

Figure 2. Experimental environment

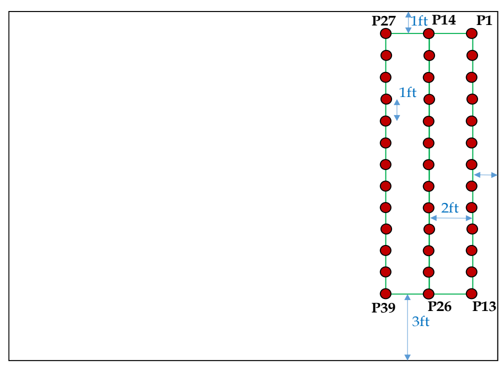

Figure 3. Sampling point setting

Sensor placement was configured to ensure minimal interference with the robot's center of gravity while still allowing air intake without obstruction. All sensor data were timestamped and logged with associated robot poses, enabling subsequent correlation of IAQ measurements with location information.

B. *Evaluation Protocol*

To assess the effectiveness of our LLM-driven robotic system, we designed several tasks reflecting common indoor inspection scenarios:

1) Localized Sampling: The robot receives a natural-language command (e.g., "Navigate to the window area and measure the air quality, pausing every minute for a sampling period"). The LLM generates a structured command containing the action type (e.g., "navigate," "measure"), the target location, and sensor parameters. This JSON command is then interpreted by the ROS2 nodes for execution.

The system evaluates whether it can parse and execute each subtask——navigating, sampling, and reporting accurately and on schedule. This figure shows CO_2 concentration measured by two sensors (S1 in blue and S2 in pink) between 11:15 AM and 12:00 PM. The robot collected these readings while navigating and performing intermittent sampling, illustrating successful execution of natural language commands for environmental monitoring *"Fig. 4"*.

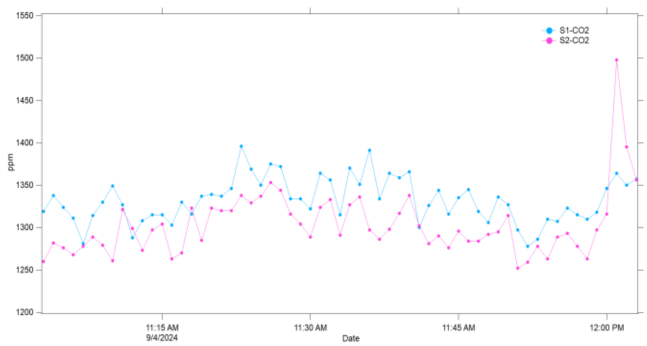

Figure 4. The results of our experiment

2) Room Patrol: The robot is instructed to traverse the entire room, stopping at predetermined stations to collect particulate and CO_2 data. Once completed, it identifies potential "hotspots" of elevated pollution levels.

3) Baseline Comparison:

We compare our approach to a baseline system that uses manually programmed waypoint navigation without natural language capabilities. Key evaluation metrics include instruction parsing accuracy, task completion time, and context awareness. These metrics quantify the efficiency and flexibility of LLM-based guidance compared to conventional scripted approaches.

IV. Results Expectations and Discussion

In this work, we introduced a multi-LLM driven approach for real-time IEQ inspections, showcasing how natural language commands can flexibly coordinate robotic navigation, sensing, and decision-making. Unlike traditional sensor networks, which are often static and lack adaptive path planning, our LLM-driven system provides several key advantages:

Flexibility: Allows operators to dynamically adjust inspection tasks without pre-programming specific routes or sensor triggers.

Context Integration: Seamlessly integrates multimodal sensor data with semantic information, enabling real-time decision-making based on location, sensor readings, and task priorities.

Efficiency and Precision: Reduces overall task completion time by integrating decision-making directly into the command pipeline, minimizing the need for manual intervention.

Looking ahead, we plan to integrate additional sensor modalities, explore on-device LLM inference for enhanced responsiveness, and investigate multi-robot collaboration to further expand the system's coverage and efficiency in complex indoor environments.

References

[1] M. Han et al., "A novel reinforcement learning method for improving occupant comfort via window opening and closing," Sustain. Cities Soc., vol. 61, p. 102247, Oct. 2020, doi: 10.1016/j.scs.2020.102247.

[2] J. Teizer, "Status quo and open challenges in vision-based sensing and tracking of temporary resources on infrastructure construction sites," Adv. Eng. Inform., vol. 29, no. 2, pp. 225–238, Apr. 2015, doi: 10.1016/j.aei.2015.03.006.

[3] M. Gharbia, A. Chang-Richards, Y. Lu, R. Y. Zhong, and H. Li, "Robotic technologies for on-site building construction: A systematic review," J. Build. Eng., vol. 32, p. 101584, Nov. 2020, doi: 10.1016/j.jobe.2020.101584.

[4] W. Lu, J. Chen, Y. Fu, Y. Pan, and F. A. Ghansah, "Digital twin-enabled human-robot collaborative teaming towards sustainable and healthy built environments," J. Clean. Prod., vol. 412, p. 137412, Aug. 2023, doi: 10.1016/j.jclepro.2023.137412.

[5] S. Park, C. C. Menassa, and V. R. Kamat, "Integrating Large Language Models with Multimodal Virtual Reality Interfaces to Support Collaborative Human–Robot Construction Work," J. Comput. Civ. Eng., vol. 39, no. 1, p. 04024053, Jan. 2025, doi: 10.1061/JCCEE5.CPENG-6106.

[6] J. M. Davila Delgado et al., "Robotics and automated systems in construction: Understanding industry-specific challenges for adoption," J. Build. Eng., vol. 26, p. 100868, Nov. 2019, doi: 10.1016/j.jobe.2019.100868.

[7] P. 'Takis' Mitropoulos and G. Cupido, "The role of production and teamwork practices in construction safety: A cognitive model and an empirical case study," J. Safety Res., vol. 40, no. 4, pp. 265–275, Aug. 2009, doi: 10.1016/j.jsr.2009.05.002.

[8] C.-J. Liang, X. Wang, V. R. Kamat, and C. C. Menassa, "Human–Robot Collaboration in Construction: Classification and Research Trends," J. Constr. Eng. Manag., vol. 147, no. 10, p. 03121006, Oct. 2021, doi: 10.1061/(ASCE)CO.1943-7862.0002154.

[9] Y. Geng et al., "Robot-based mobile sensing system for high-resolution indoor temperature monitoring," Autom. Constr., vol. 142, p. 104477, Oct. 2022, doi: 10.1016/j.autcon.2022.104477.

[10] I. Kostavelis and A. Gasteratos, "Learning spatially semantic representations for cognitive robot navigation," Robot. Auton. Syst., vol. 61, no. 12, pp. 1460–1475, Dec. 2013, doi: 10.1016/j.robot.2013.07.008.

[11] Z. Peng, X. Li, Z. Zhu, M. Unoki, J. Dang, and M. Akagi, "Speech Emotion Recognition Using 3D Convolutions and Attention-Based Sliding Recurrent Networks with Auditory Front-Ends," IEEE Access, vol. 8, pp. 16560–16572, 2020, doi: 10.1109/ACCESS.2020.2967791.

[12] I. B. de A. Santos and R. A. F. Romero, "A Deep Reinforcement Learning Approach with Visual Semantic Navigation with Memory for Mobile Robots in Indoor Home Context," J. Intell. Robot. Syst., vol. 104, no. 3, p. 40, Feb. 2022, doi: 10.1007/s10846-021-01566-0.

[13] M. Jin, S. Liu, S. Schiavon, and C. Spanos, "Automated mobile sensing: Towards high-granularity agile indoor environmental quality monitoring," Build. Environ., vol. 127, pp. 268–276, Jan. 2018, doi: 10.1016/j.buildenv.2017.11.003.