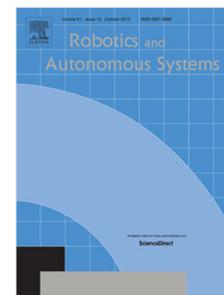


# Journal Pre-proof

A knowledge-driven framework for robotic odor source localization using large language models

Khan Raqib Mahmud, Lingxiao Wang, Sunzid Hassan, Zheng Zhang



PII: S0921-8890(25)00001-6

DOI: <https://doi.org/10.1016/j.robot.2025.104915>

Reference: ROBOT 104915

To appear in: *Robotics and Autonomous Systems*

Received date: 11 September 2024

Revised date: 11 December 2024

Accepted date: 3 January 2025

Please cite this article as: K.R. Mahmud, L. Wang, S. Hassan et al., A knowledge-driven framework for robotic odor source localization using large language models, *Robotics and Autonomous Systems* (2025), doi: <https://doi.org/10.1016/j.robot.2025.104915>.

This is a PDF file of an article that has undergone enhancements after acceptance, such as the addition of a cover page and metadata, and formatting for readability, but it is not yet the definitive version of record. This version will undergo additional copyediting, typesetting and review before it is published in its final form, but we are providing this version to give early visibility of the article. Please note that, during the production process, errors may be discovered which could affect the content, and all legal disclaimers that apply to the journal pertain.

© 2025 Published by Elsevier B.V.

<sup>1</sup> A Knowledge-Driven Framework for Robotic Odor Source  
<sup>2</sup> Localization using Large Language Models

<sup>3</sup> Khan Raqib Mahmud<sup>a</sup>, Lingxiao Wang<sup>b,\*</sup>, Sunzid Hassan<sup>a</sup>, Zheng Zhang<sup>c</sup>

<sup>a</sup>*Department of Computer Science, Louisiana Tech University, Ruston, 71272, Louisiana, USA*

<sup>b</sup>*Department of Electrical Engineering, Louisiana Tech University, Ruston, 71272, Louisiana, USA*

<sup>c</sup>*Department of Aerospace Engineering, Embry-Riddle Aeronautical University, Daytona Beach, 32114, Florida, USA*

<sup>4</sup> **Abstract**

Robotic Odor Source Localization (OSL) technology enables mobile robots to detect and navigate unknown odor sources in diverse environments. Traditional OSL methods, including bio-inspired, engineering-based, and machine learning-based approaches, face limitations of lack of adaptability to varying environments, significant computational resource requirements, and dependence on historical data. To overcome these challenges, we present a knowledge-driven framework that leverages Large Language Models (LLMs) to improve the robot's navigation capabilities through contextual understanding and informed decision-making. A key feature of the proposed work is integrating an LLM agent with a memory module, which stores past experiences and recalls them during the decision-making process, allowing the robotic agent to make decisions based on current sensory inputs and previously acquired knowledge. Compared to traditional deep learning-based methods, such as Deep Q-Network (DQN), both simulation and real-world experiment results demonstrate that our framework significantly outperforms it in terms of accuracy, efficiency, and generalization across different environmental conditions.

<sup>5</sup> *Keywords:* Robotic Odor Source Localization, Large Language Models (LLMs),

<sup>6</sup> Knowledge Driven Framework, Deep Q-Network (DQN).

<sup>7</sup> **1. Introduction**

<sup>8</sup> Olfaction, also known as the sense of smell, provides crucial information about the environment. In nature, animals depend on olfaction to perform life-essential activities, such as foraging [1], homing [2], mate-seeking [3], and navigating in the environment [4] [5]. A mobile robot, integrated with gas or chemical sensors, could detect and trace odors to find an

\*Corresponding author

Email address: [lwang@latech.edu](mailto:lwang@latech.edu) (Lingxiao Wang)

<sup>1</sup>This work is supported by Louisiana Board of Regents with the contract number: LEQSF(2024-27)-RD-A-22.

12 unknown odor source within an environment. The technology of using robots to locate odor  
 13 sources is known as robotic Odor Source Localization (OSL) [6] [7]. In practice, robotic OSL  
 14 finds its applications in air pollutant localization [8], wildfire smoke tracing [9], gas emission  
 15 monitoring [10], and the detection of ocean hydrothermal vents [11], among others.

16 The key to successfully locating an odor source is the design of an effective navigation  
 17 algorithm, i.e., the OSL algorithm, which guides the robot moving toward the odor source  
 18 location based on sensor observations. Present OSL algorithms [12] include bio-inspired  
 19 methods [13, 14, 15, 16, 17, 18, 19] that mimic animal behaviors, engineering-based methods  
 20 [20, 21], that use mathematical models, and machine learning-based methods that rely  
 21 on trained models. Bio-inspired methods, like the moth-inspired algorithm, command a  
 22 mobile robot to find an odor source by mimicking animal odor-searching behaviors [22,  
 23, 24]. However, these methods follow pre-determined behavioral patterns that may not  
 24 adapt well to airflow-varying environments [25]. Engineering-based methods, such as the  
 25 particle filter-based algorithm, rely on mathematical models to derive possible odor source  
 26 locations and direct the robot to the estimated source locations [26]. The downside of  
 27 the engineering-based method is computational complexity [27]. Since most engineering-  
 28 based methods divide the search area into multiple cells, and for each cell, there will be a  
 29 probability calculation to calculate the chance of this cell containing the odor source. The  
 30 computational complexity will increase exponentially if the search area is large and broad.  
 31 Machine learning-based methods, including deep supervised and reinforcement learning (RL)  
 32 techniques, often require extensive training data and computational resources [28]. These  
 33 algorithms aim to model the complex patterns within sensory data collected from various  
 34 environments. However, they frequently face issues like dataset bias, overfitting, and a lack  
 35 of interpretability. Addressing these challenges is crucial for gaining a deeper understanding  
 36 of odor dispersion patterns and making more rational decisions, which could significantly  
 37 improve the effectiveness of OSL systems.

38 Drawing inspiration from the profound capabilities of human cognition, we explore the  
 39 core principles that underlie effective odor localization and raise a pivotal distinction: tra-  
 40 ditional OSL methods are fundamentally data-driven, whereas human olfactory behaviors  
 41 are knowledge-driven [29]. For instance, when faced with a complex odor plume in a turbu-  
 42 lent environment, humans can rely on contextual understanding and reasoning to navigate  
 43 toward the source. Conversely, data-driven methods rely heavily on a large quantity of  
 44 similar data to fit these scenarios, which limits their ability to generalize across diverse con-  
 45 ditions. Additionally, collecting and annotating large datasets for training OSL models is  
 46 labor-intensive and costly.

47 The knowledge-driven approach has emerged as a promising alternative to traditional  
 48 methods in recent years. Unlike data-driven models that rely solely on large datasets,  
 49 knowledge-driven models incorporate contextual understanding, reasoning, and decision-  
 50 making capabilities similar to human cognition. This approach is particularly powerful  
 51 in scenarios where understanding and interpreting complex environments is crucial. Large  
 52 Language Models (LLMs), which have demonstrated exceptional abilities in various domains,  
 53 embody this knowledge-driven paradigm [30, 31]. LLMs leverage vast amounts of pre-  
 54 existing knowledge, enabling them to generalize better across diverse scenarios and make

55 informed decisions with limited contextual data [32, 33]. Recent advancements in LLMs  
 56 with emergent abilities offer an ideal embodiment of human knowledge, providing valuable  
 57 insights toward addressing the challenges of OSL. LLMs possess exceptional human-level  
 58 abilities and show strong performance in areas such as robotics manipulation [34], multi-  
 59 modal understanding [35], and lifelong skill learning [36]. However, like humans need practice  
 60 to master complex tasks, LLMs also require experience and guidance to perform effectively  
 61 in specific applications.

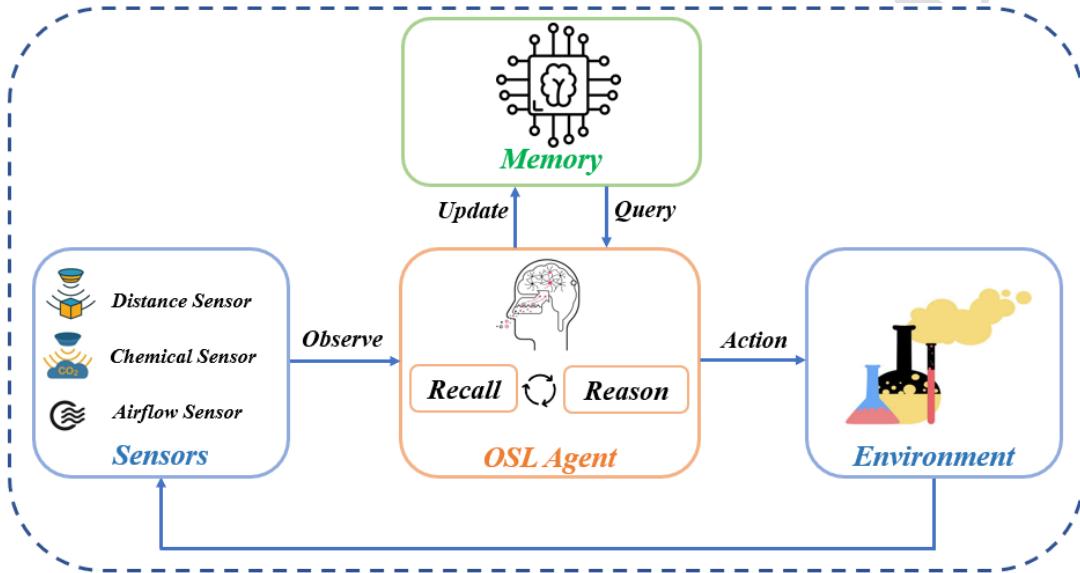


Figure 1: The knowledge-driven paradigm for robotic odor source localization, including an interactive environment, an OSL agent with recall and reasoning abilities, and an independent memory module. The OSL agent continuously processes sensory inputs from the environment, queries and updates experiences from the memory module, and makes informed decisions to navigate toward the odor source.

62 To leverage the potential of LLMs for robotic OSL, we adapt a novel framework that  
 63 integrates these models into a knowledge-driven approach [32]. This framework, illustrated  
 64 in Figure 1, incorporates several key components: a sensors module, an interactive environ-  
 65 ment, an OSL agent with a reasoning module, and a memory module to store and recall  
 66 experiences. The reasoning module uses the LLM to query stored experiences from the  
 67 memory module and apply common-sense knowledge to make informed decisions based on  
 68 current scenarios. This process involves continuous evolution, where the agent observes the  
 69 environment, queries, and updates experiences from memory, and makes decisions.

70 Our research presents a novel framework for robotic OSL that leverages the capabilities of  
 71 LLMs. To the best of our knowledge, we are the first to work to integrate LLM with robotic  
 72 OSL problems. This knowledge-driven approach overcomes the limitations of traditional  
 73 methods by integrating reasoning, contextual understanding, and scenario interpretation  
 74 into the decision-making process. We compare the performance of our LLM-based framework  
 75 with Deep Q-Networks (DQNs), a popular RL method [37]. Both simulation and real-world  
 76 experiment results demonstrate that the LLM-based approach not only outperforms DQNs

77 in terms of accuracy and efficiency but also exhibits superior generalization across different  
 78 environments.

79 The contributions of our article are summarized as follows:

- 80 1. We introduce a knowledge-driven framework that leverages LLMs for robotic OSL,  
 81 emphasizing the advantages of integrating reasoning, contextual understanding, and  
 82 decision-making processes.
- 83 2. We validate the proposed OSL framework under realistic conditions by collecting real-  
 84 world plume data using a wind tunnel and Particle Image Velocimetry (PIV) system.
- 85 3. We compare the performance of the LLM-based OSL framework with traditional DQN  
 86 methods in both simulated and real-world search environments, demonstrating the  
 87 superior adaptability and efficiency of the LLM-based approach.

88 **2. Related Works**

89 *2.1. Robotic Odor Source Localization (OSL)*

90 *2.1.1. Bio-Inspired Methods*

91 Bio-inspired methods for robotic OSL draw inspiration from the natural world, partic-  
 92 ularly from the behaviors and mechanisms employed by various organisms to locate odor  
 93 sources. These methods have been adapted into algorithms and systems to enhance the  
 94 efficiency and adaptability of robots in tasks such as detecting and localizing chemical com-  
 95 pounds or gases in the air, which is crucial for applications such as environmental monitor-  
 96 ing, search and rescue operations, and detecting gas leaks or hazardous substances. One  
 97 approach involves mimicking the adaptability of insect brains to control a robot's movement  
 98 based on sensory feedback, as demonstrated by a brain-machine hybrid system that adjusts  
 99 the robot's velocity in response to neural activities descending from an insect's brain [38].  
 100 Similarly, the flight patterns of moths have inspired algorithms that trigger motion in robots  
 101 upon detecting gas, integrating a repulsion function to navigate around obstacles [39]. The  
 102 behavior of the adult male silk moth, particularly its method of modulating speed based  
 103 on odor detection frequency, has been used to develop a robust moth-inspired algorithm for  
 104 indoor and outdoor odor source localization [40]. This approach leverages the moths' ability  
 105 to use smell and wind direction to locate mates, showing excellent performance in various  
 106 environmental complexities [41] [42]. Other organisms, such as flatworms, have also inspired  
 107 bio-inspired OSL methods. Flatworms' kinesis response and tropotaxis behavior, which do  
 108 not require wind direction or odor concentration information, have led to a bionic odor source  
 109 localization algorithm that improves search efficiency and environmental adaptability [43].  
 110 Additionally, a method mimicking male moths tracking pheromone plumes uses simulations  
 111 of their sensory, behavior, and control systems. Despite advancements, these models still  
 112 fall short of natural efficiency, prompting optimization through genetic algorithms [44]. In-  
 113 corporating biological components, such as moth antennae, into flying robots has provided  
 114 rapid response times and high specificity and sensitivity in chemical detection due to gene  
 115 editing advances [45]. Additionally, the collective behavior observed in nature has led to

116 the development of swarm intelligence algorithms for multi-robot systems, enhancing par-  
 117 allelism, scalability, and robustness in odor source localization tasks [46]. Particle Swarm  
 118 Optimization (PSO), based on the social behavior of birds and fish, has also been adapted  
 119 for odor localization, proving effective in real-world scenarios [47] [48] [49]. Cross-wind Levy  
 120 Walks, spiraling, and upwind surges, inspired by animal search patterns, form a robust 3D  
 121 algorithm for odor localization, outperforming 2D methods [50]. Additionally, biohybrid  
 122 systems combining living materials with synthetic devices have led to robots with insect  
 123 antennae for odor sensing, allowing autonomous navigation and obstacle avoidance [51].

124 However, bio-inspired methods face limitations. The complexity of accurately mimicking  
 125 biological systems can lead to challenges in learning and implementing such methods effec-  
 126 tively. Moreover, the reliance on specific sensor types, such as metal oxide semiconductor  
 127 (MOS) sensors, can limit the sensing capacity of OSL robots, making it difficult to match  
 128 the performance of their biological counterparts [52].

### 129 *2.1.2. Engineering-Based Methods*

130 Robotic OSL employs various engineering-based methods to enable robots to detect and  
 131 locate the sources of odors. These methods range from algorithmic approaches to the utiliza-  
 132 tion of unmanned aerial vehicles (UAVs) for enhanced mobility and efficiency. One primary  
 133 algorithmic approach involves the use of Independent Posteriors (IP) and Dempster–Shafer  
 134 (DS) theory algorithms, which rely on occupancy grid mapping to estimate the probability  
 135 of a location being an odor source. The IP algorithm has shown superior performance in  
 136 minimizing false source attributions in turbulent fluid flow environments [53]. For scenarios  
 137 where constructing a dispersion model of the odor plume is impractical due to complex  
 138 geometries, a combination of Infotaxis and Dijkstra algorithms has been proposed. This  
 139 approach dynamically adjusts the robot's focus between exploration and exploitation, sig-  
 140 nificantly improving success rates and reducing search times [54]. In the realm of UAVs,  
 141 a multi-UAV collaboration based on a collaborative particle filter algorithm and an adap-  
 142 tive path planning algorithm has been developed. This method aims to quickly locate odor  
 143 sources with minimal resource consumption, demonstrating superior performance in simu-  
 144 lation platforms [52]. Olfactory quadruped robots equipped with various sensors have been  
 145 developed for complex environment navigation and odor source localization, offering adapt-  
 146 ability and eco-friendliness [55]. These diverse engineering-based methods highlight the  
 147 interdisciplinary efforts to improve robotic OSL, leveraging algorithmic precision, UAV mo-  
 148 bility, and bio-inspired strategies to address the challenges of locating odor sources in com-  
 149 plex environments. Engineering-based methods, such as those employing UAVs for OSL,  
 150 benefit from flexible deployment and controllable movement in 3D space, offering precise  
 151 source estimation and efficient navigation through collaborative algorithms and adaptive  
 152 path planning [27]. These methods often rely on well-defined computational models and  
 153 algorithms, like Fuzzy inference and Markov decision processes, to optimize search strate-  
 154 gies in turbulent flow environments. While these approaches provide clear frameworks for  
 155 problem-solving and optimization, they may lack the inherent adaptability and robustness  
 156 to environmental variability that bio-inspired methods offer.

157 In summary, bio-inspired methods for robotic OSL offer adaptability, efficiency, and

robustness by leveraging natural strategies, but they may struggle with the complexity of certain algorithms and sensor limitations. Conversely, engineering-based methods provide precise, optimized solutions through computational models but may lack the adaptability to environmental changes seen in bio-inspired approaches.

### 2.1.3. Learning-Based Methods

Learning-based methods for robotic OSL have significantly advanced, leveraging deep learning (DL) and reinforcement learning (RL) to enhance robots' ability to detect and locate odor sources efficiently. DL has emerged as a promising approach for robotic OSL, offering the potential to navigate and identify odor sources with high accuracy. Deep learning methods, particularly, have been instrumental in developing algorithms that enable mobile robots to navigate towards an odor source without predefined search strategies. Two notable deep neural networks (DNNs), feedforward neural networks (FNN) and long short-term memory neural networks (LSTM), have been developed to calculate robot heading commands based on sensor readings, showing promising results in real-world experiments [56]. Additionally, a deep learning-based odor compass has been designed, incorporating a deep learning-based odor attention (DL-OA) model with a separated spatial-temporal attention-based encoder-decoder structure for end-to-end odor source direction estimation (OSDE), demonstrating significant accuracy in indoor environments [57]. Another approach involves a convolutional neural network (CNN) and LSTM modules to improve the accuracy and generalization ability of odor-source direction estimation, further enhancing the performance of OSL robots [58]. On the reinforcement learning front, a multi-continuous-output Takagi–Sugeno–Kang fuzzy system tuned with reinforcement learning has been proposed. This system, designed for dynamic outdoor environments, relies on the robot's observations to guide it towards the odor source, showing comparable success rates and higher efficiency than manually tuned systems [59]. Moreover, the integration of probabilistic models, such as probabilistic gas-hit maps, into the robotic systems provides a higher level of abstraction to model the time-dependent nature of gas dispersion, aiding in source localization in complex indoor environments [60]. The integration of convolutional neural network (CNN) and LSTM modules into the odor compass design further improves accuracy and generalization ability in OSL tasks [61]. Deep learning frameworks also excel in handling sparse and unreliable spatio-temporal chemical sensor data, offering regularized solutions that accurately predict gas leak sources by conforming to the spatio-temporal structure of gas concentration distribution [62]. These advancements underscore the potential of AI-based methods in enhancing the efficiency and reliability of robotic odor source localization.

However, several challenges and limitations are associated with its application, which necessitates further research and development to address. One significant challenge is the complexity of environmental factors, such as signal noise, obstacles, and sparse fingerprints, which can hinder the modeling and localization performance of robots in indoor environments. To overcome this, a novel deep learning framework with a localization attention module and a multi-faceted localization module integrating LSTM and GRU has been proposed, showing efficiency in capturing dynamic spatial and temporal features [63]. Another limitation is the difficulty in learning complex search strategies, such as Bayesian-inference

200 methods, through DL. Experiments have shown that while DL models can imitate simpler  
 201 moth-inspired methods, they struggle with more complex strategies [56]. Addressing this  
 202 requires the development of more sophisticated neural network architectures or training  
 203 methodologies.

204 In summary, addressing the challenges and limitations of using deep learning for robotic  
 205 OSL requires a multifaceted approach, including the development of advanced DL frame-  
 206 works, improved sensor technologies, sophisticated training methodologies, and the integra-  
 207 tion of biological insights into algorithm design [41].

208 *2.2. LLM-based Agent*

209 *2.2.1. Large Language Models*

210 Large language models (LLMs) have significantly transformed the landscape of natural  
 211 language processing (NLP), pushing the boundaries of language understanding and genera-  
 212 tion to new heights. These models, with their vast amount of parameters frequently reaching  
 213 the hundreds of billions, are trained on extensive text datasets. This in-depth training en-  
 214 ables them to grasp natural language and perform various intricate tasks, primarily focusing  
 215 on text generation and comprehension. Notable examples of LLMs include GPT-3 [64],  
 216 PaLM [65], LLaMA [66], and GPT-4 [67]. Constructed using the transformer architecture,  
 217 LLMs have demonstrated significant advancements in performance compared to earlier ver-  
 218 sions due to utilizing extensive data and complex training methods. LLMs stand out from  
 219 smaller language models due to their emergent capabilities, including in-context learning [64],  
 220 following instructions [68, 69], and reasoning with chain-of-thought [70]. The creation and  
 221 implementation of these models have achieved state-of-the-art performance across numerous  
 222 NLP tasks, introducing a paradigm shift in how these tasks are approached. They leverage  
 223 pre-training on large datasets followed by fine-tuning for specific applications [71]. LLMs'  
 224 capacity to produce coherent and contextually appropriate text makes them extremely useful  
 225 for various applications in educational technology, computational social science, and beyond.  
 226 These models have demonstrated exceptional capabilities in generating text, understanding  
 227 complex language nuances, and performing tasks with minimal input through zero-shot or  
 228 one-shot learning settings [72]. The evolution of LLMs, highlighted by their ability to per-  
 229 form tasks zero-shot without specific training data, is transforming computational social  
 230 science by serving as zero-shot data annotators and bootstrapping creative generation tasks,  
 231 showcasing their versatility and broad applicability [73].

232 Recent advancements in LLMs have showcased human-like intelligence and hold the po-  
 233 tential to propel us closer to the realm of Artificial General Intelligence (AGI) [73]. OpenAI's  
 234 pursuit of LLMs has led to significant milestones such as ChatGPT [74] and GPT-4 [67].  
 235 These milestones signify notable advancements in LLMs' capabilities, particularly in natural  
 236 language understanding and generation. The continuous development of LLMs, including  
 237 refining architectures and training strategies, promises further advancements in their ca-  
 238 pabilities and applications. These models have become essential tools in various domains,  
 239 driving innovation and enhancing the ability to tackle complex problems through advanced  
 240 language understanding and generation.

241 *2.2.2. Robotic Transformer*

242 The concept of a “Robotic Transformer” encompasses a range of innovative approaches in  
 243 robotics, focusing on enhancing the capabilities of robots through advanced machine learning  
 244 models, particularly transformers, for tasks such as 3D object manipulation, robotic  
 245 grasping, and skill assessment in robot-assisted activities. In the realm of 3D object manip-  
 246 ulation, the Robotic View Transformer (RVT) represents a significant advancement, offering  
 247 a scalable and accurate multi-view transformer model that outperforms existing methods  
 248 in both training speed and inference speed, demonstrating its effectiveness across a vari-  
 249 ety of real-world tasks with minimal demonstrations required [75]. Similarly, the Robotics  
 250 Transformer model emphasizes the importance of transferring knowledge from large, diverse  
 251 datasets to solve specific downstream tasks, showcasing the potential for generalization in  
 252 robotics through open-ended task-agnostic training [76]. For robotic grasping, extending  
 253 transformer models to 6-Degree-of-Freedom (6-DoF) grasping has shown promising results,  
 254 with methods that efficiently learn both global and local features, significantly improving  
 255 success rates in challenging datasets [77]. Additionally, Act3D introduces a manipulation  
 256 policy transformer that excels in 3D detection for end-effector pose prediction, setting new  
 257 benchmarks in manipulation tasks through its innovative use of 3D feature clouds and adap-  
 258 tive spatial computation [78]. These transformer-based approaches demonstrate significant  
 259 advancements in robotic manipulation tasks, showcasing superior performance, scalability,  
 260 and generalization abilities.

261 *2.2.3. LLM-based Agent in Robotic Tasks*

262 Current advancements in Large Language Model (LLM)-based agent technology have  
 263 significantly enhanced the capabilities of robots in understanding and executing complex  
 264 tasks. These advancements leverage the vast knowledge encoded in LLMs, extending be-  
 265 yond simple prompt engineering to enable more nuanced and situationally aware interactions  
 266 between robots and their environments. A cognitive-agent approach has been developed to  
 267 mitigate the limitations of prompt engineering, allowing robots to acquire new task knowl-  
 268 edge that aligns with their native language capabilities, embodiment, environment, and user  
 269 preferences. This approach has demonstrated the ability of robots to achieve high task  
 270 completion rates in one-shot learning scenarios, with and without human oversight [79].

271 Moreover, the integration of LLMs with reinforcement learning has led to the develop-  
 272 ment of mediator models that optimize the cost and frequency of interactions between agents  
 273 and LLMs. These models enable agents to consult LLMs only when necessary, significantly  
 274 reducing interaction costs and improving performance in complex decision-making tasks  
 275 [80]. LLMs have shown significant promise in enhancing robotic task learning, particularly  
 276 by providing a rich knowledge source that can be tapped into for novel task acquisition. One  
 277 promising approach leverages LLMs for object goal navigation in a zero-shot manner, where  
 278 an embodied agent navigates to a target object described in natural language within an un-  
 279 explored environment. This method has demonstrated substantial improvements in success  
 280 rates over current baselines [81], utilizing the implicit knowledge of LLMs about the semantic  
 281 context of the environment and mapping it into sequential inputs for robot motion plan-  
 282 ning. Furthermore, LLMs integrated with visual and natural language understanding have

283 been explored for incremental decision-making in real-world environments, such as Vision  
 284 and Language Navigation (VLN), where an embodied agent follows navigation instructions  
 285 grounded in real-world observations [82].

286 The concept of using LLMs as a 'robotic brain' to unify memory and control within an  
 287 embodied AI system has been introduced, demonstrating the potential of LLMs in active  
 288 exploration and embodied question answering tasks [83]. This approach not only streamlines  
 289 the interaction between perception, planning, and control but also significantly boosts the  
 290 efficiency and accuracy of robotic tasks. A closed-loop technique, AdaPlanner, has been  
 291 proposed to allow LLM agents to adaptively refine their plans in response to environmental  
 292 feedback, showing improved performance in sequential decision-making tasks [84].

293 In the context of memory utilization, Retrieval-Augmented Generation (RAG) has emerged  
 294 as a highly relevant approach that combines retrieval mechanisms with LLM generation  
 295 tasks. RAG typically retrieves external documents or passages to directly augment LLM  
 296 outputs in a single-stage process. RAG has demonstrated advantages in enhancing knowl-  
 297 edge integration, processing complex queries, and improving retrieval efficiency [85]. Its  
 298 scalability and adaptability to dynamic environments, facilitated by mechanisms such as  
 299 topology-aware retrieval and relevant information gain, make it a strong theoretical founda-  
 300 tion for structured memory-driven tasks [86]. However, RAG's dependency on external data  
 301 sources and the challenges of filtering retrieved information demonstrate the importance for  
 302 domain-specific adjustments to maximize its effectiveness [87].

303 LLM-based agents demonstrate exceptional proficiency in conversational engagement  
 304 and adherence to instructions across various downstream tasks, outperforming rule-based  
 305 systems that require explicit programming for each specific task [88]. These agents can  
 306 process and generate natural language instructions, adapt to new tasks with minimal data,  
 307 and engage in complex decision-making processes. However, LLMs lack the inherent ability  
 308 to engage with and comprehend the complexities of odor source localization as effectively as  
 309 humans. Robotic OSL systems require active interaction with and understanding of their  
 310 environment.

311 To bridge this gap, we propose a novel knowledge-driven framework for robotic OSL that  
 312 enables LLMs to comprehend and navigate odor source localization tasks by incorporating  
 313 human knowledge. By combining the semantic understanding and decision-making capabili-  
 314 ties of LLMs with the sensory data from traditional OSL methods, robots could achieve  
 315 higher levels of accuracy and efficiency in detecting and navigating towards odor sources.  
 316 The integration of LLMs into robotic OSL could revolutionize the field by providing a more  
 317 intuitive and flexible method for robots to understand and navigate their environment. This  
 318 approach has the potential to lead to significant improvements in safety and rescue op-  
 319 erations, pollution control, and environmental monitoring, addressing the urgent need for  
 320 efficient, safe, and accurate localization of hazardous chemical gas leaks.

321 **3. Methodology**

322 *3.1. Problem Statement*

323 The primary objective of Robotic OSL is to develop a mobile robotic system that can  
 324 detect and navigate toward an unknown odor source within a specific environment. This  
 325 task involves determining a sequence of actions that will effectively guide the robot to the  
 326 odor source. Mathematically, we can represent this process as follows:

$$a = F(o, s) \quad (1)$$

327 where  $a$  denotes the robot's action,  $F$  represents the navigation algorithm,  $o$  is the past  
 328 observation, and  $s$  refers to the current observation. The main challenge lies in finding the  
 329 optimal function  $F$  that generates a sequence of actions  $a$  to navigate the robot toward  
 330 the odor source successfully. This function must effectively leverage past experiences  $o$  and  
 331 real-time sensory data  $s$  to make informed decisions that lead the robot to the odor source.

332 To address this challenge, we propose a knowledge-driven paradigm that leverages the ca-  
 333 pabilities of Large Language Models (LLMs) to enhance the robot's navigation and decision-  
 334 making processes. Our approach, illustrated in Figure 1, aims to utilize the generalization  
 335 ability of LLMs, along with their contextual understanding of environmental dynamics, to  
 336 develop a more robust and adaptable function  $F$ . This function integrates past experiences  
 337 stored in memory with real-time sensory inputs to guide the robot in efficiently locating  
 338 odor sources through advanced navigation strategies.

339 *3.1.1. The Search Area and Plume Field*

340 The search area is defined as a two-dimensional search space that contains an odor  
 341 source, the location of which is unknown to the robot. This study focuses on navigating  
 342 within this space to locate the unknown odor source. The odor plume is dispersed within  
 343 this search area, creating a plume field characterized by varying concentrations of odor.  
 344 These concentrations are influenced by environmental factors such as wind and obstacles,  
 345 making the search process more complex.

346 In this study, the plume field was generated by using a wind tunnel and a Particle  
 347 Image Velocimetry (PIV) system, as illustrated in Figure 2(a). The PIV system utilized  
 348 green lasers to identify plume positions and velocities, enabling precise measurements of  
 349 the odor plume distribution. These measurements are crucial for identifying the real-world  
 350 plume distribution. The data collection process involved continuous emission of odor plumes  
 351 (mineral oil) [89] into the wind tunnel through a nozzle, as shown in Figure 2(b). The PIV  
 352 system was employed to record plume positions and velocities in the downstream area, which  
 353 had dimensions of  $2.2 \times 5.0 \text{ m}^2$  ( $x \times y$ ). The resolution of PIV measurements was 220 by 500,  
 354 providing detailed spatial information about the plume dynamics [89]. For this study, the  
 355 search area extends along the x-axis from 0 to 200 resolution units and the y-axis from -110  
 356 to +110 resolution units, corresponding to approximately  $2.2 \times 2.0 \text{ m}^2$ . The odor source is  
 357 located at positions  $x=2$  and  $y=12$ , as illustrated in Figure 2(c).

358 To derive the odor concentration at each point in the plume, we used a mathematical  
 359 model that calculates concentration based on the velocity data obtained from the Particle

360 Image Velocimetry (PIV) system, 2(c). We calculated the concentration at each point in  
 361 the plume field using the formula:

$$c = \sqrt{u^2 + v^2} \quad (2)$$

362 where  $u$  and  $v$  represent the velocity components along the x-axis and y-axis, respectively,  
 363 and  $c$  represents the odor concentration at specific position. This approach assumes that  
 364 higher velocities correspond to denser plume areas, thus indicating higher concentrations of  
 365 the odor. This calculated concentration data and the velocity plots provided by the PIV  
 366 form the basis for creating a detailed concentration map of the plume field. This map is  
 367 crucial for simulating realistic odor dispersion patterns in our experimental setup.

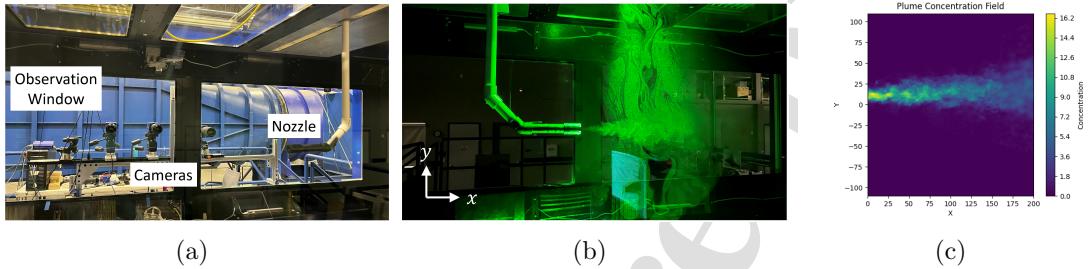


Figure 2: (a) Experiment setup. (b) The nozzle continuously releases odor plumes into the wind tunnel. Green lasers help the PIV identify plume positions and velocities. (c) Diagram of the search area, illustrating the plume field and the locations of the odor source. The color scale ranges from deep purple (low concentration, 0) to bright yellow (high concentration, 16.4), depicting the intensity of the odor plume across the search area.

368 The detailed setup and measurement process ensure high-quality data collection, which is  
 369 essential for conducting realistic simulations and evaluating the effectiveness of the proposed  
 370 OSL framework. This dataset is crucial for testing and validating the effectiveness of the  
 371 OSL algorithms under real-world conditions, ensuring that the developed algorithms are  
 372 robust and applicable in practical scenarios.

### 373 3.1.2. The Robotic Agent

374 The robotic agent is designed to detect chemical compounds in the air and efficiently  
 375 navigate toward the odor source. The key sensors integrated into the robotic agent include  
 376 chemical sensors for detecting the presence and concentration of odors and anemometers for  
 377 measuring wind direction and speed. These sensors enable the robot to gather comprehensive  
 378 environmental data, essential for effective navigation toward the odor source.

Table 1: Definition of Parameters

Parameters	Definitions
$\alpha$	Wind Direction
$\beta$	Speed Factor
$\nu$	Original Speed
$\nu_c$	Robot Speed Command
$\phi_c$	Robot Heading Command
$c_{norm}$	Normalized Concentration

379 As detailed in Table 1, the robot has an original speed  $\nu$ , starting with a robot heading  
 380 command of zero degrees. Only speed and heading commands are required to control a  
 381 robot in a 2D plane. The robot employs an adaptive speed mechanism to calculate its speed  
 382 command ( $\nu_c$ ) that adjusts its speed based on the concentration of the detected odor. The  
 383 adaptive speed mechanism adjusts the robot speed command using the following formula:

$$\begin{aligned}\beta &= \max(0.1, 1 - c_{norm}) \\ \nu_c &= \nu \times \beta,\end{aligned}\tag{3}$$

384 where  $\beta$  is the speed factor, calculated based on the normalized odor concentration  $c_{norm}$  at  
 385 the robot's position and eight adjacent points.  $\nu$  represents the robot's original speed, and  
 386  $\nu_c$  is the robot speed command. This adaptive mechanism ensures that the robot reduces  
 387 its speed as it nears the odor source for better exploitation accuracy while preventing the  
 388 speed from reaching zero to maintain constant movement.

389 We determine the robot's heading command ( $\phi_c$ ) by two search behaviors inspired by the  
 390 mate-seeking behaviors of male moths, including surge and casting behaviors [90]. Surge  
 391 behavior is activated when the moth is inside the plume, which commands the moth to  
 392 move upwind; Casting behavior is activated when the moth moves out of the plume area,  
 393 and the moth will move across the wind to increase the chance of re-detecting plumes. By  
 394 iterating these two search behaviors, a male moth can find a female moth from a considerable  
 395 distance.

396 In this work, we convert these search behaviors into two robot action commands, denoted  
 397 as 0 and 1. Action 0 represents surge behavior, where the robot moves against the wind  
 398 direction, and action 1 represents casting behavior, where the robot moves across the wind  
 399 direction. During the search process, the LLM agent decides which action to perform based  
 400 on the current sensor reading and past sensor observations. If the LLM agent selects action  
 401 0, it moves upwind by adding 180 degrees to the current wind direction, simulating the  
 402 moth's direct approach toward the odor source. Conversely, if action 1 is selected, it moves  
 403 across the wind by adding 90 degrees to the wind direction. Mathematically, we calculated  
 404 the robot heading command as follows:

$$\phi_c = \begin{cases} \alpha + 180 & \text{if } action = 0; \\ \alpha + 90 & \text{if } action = 1, \end{cases}\tag{4}$$

405 This method ensures that the robot can adeptly adjust its trajectory in response to changes  
 406 in wind direction, effectively using bio-inspired maneuvers to locate the odor source. It  
 407 is important to note that the key difference between the moth-inspired method and our  
 408 approach is that we use LLM to decide whether the robot should surge or cast, whereas  
 409 the moth-inspired method makes decisions based on the current sensor reading. The LLM  
 410 makes the decisions based not only on the current sensor reading but also on past sensory  
 411 observations.

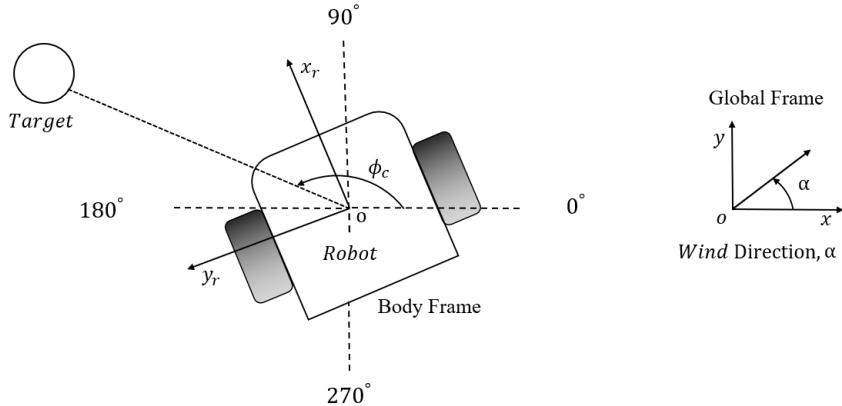


Figure 3: Control of a ground mobile robot.

412 The robot updates its position once the heading command is determined based on the  
 413 obtained heading command. As illustrated in Figure 3, the robot operates within a global  
 414 frame ( $xoy$ ), with its localized adjustments made relative to its body frame ( $x_r o y_r$ ). This  
 415 updating process is determined mathematically in Equation 5, which utilizes trigonometric  
 416 functions to determine the robot's new position:

$$\begin{aligned} x^{k+1} &= x^k + (\nu_c^k \times \cos(\phi_c^k)) \\ y^{k+1} &= y^k + (\nu_c^k \times \sin(\phi_c^k)). \end{aligned} \quad (5)$$

417 Here,  $\phi_c^k$  is the robot's heading command at the  $k$ -th time step, calculated by Equation 4.  
 418 The variables  $(x^k, y^k)$  denote the robot's current coordinates at the  $k$ -th time step within  
 419 the plume field. The adaptive speed,  $\nu_c^k$ , is determined using Equation 3 based on the odor  
 420 concentration detected at the  $k$ -th time step. After executing an action, the robot's updated  
 421 coordinates  $(x^{k+1}, y^{k+1})$  represent its new position at time step  $k + 1$ .

422 The primary goal of the robotic agent is to locate the odor source within the defined  
 423 search area. The robot starts from a random initial position, denoted by  $(x^0, y^0)$ , and uses  
 424 its sensors to detect and follow the odor plume. The robot's task involves making informed  
 425 decisions about its movement direction and distance based on the odor concentration and  
 426 wind direction at its current location. Upon detecting the odor concentration at its current  
 427 position, the robot can choose between moving against the wind or across the wind direction.  
 428 The chosen action and the wind direction determine the robot's heading command using  
 429 Equation 4.

430 *3.2. Proposed Knowledge-Driven OSL Framework*

431 In this section, we present the knowledge-driven framework for robotic OSL, which integrates  
 432 LLMs to enhance robots' decision-making and navigational capabilities. As depicted  
 433 in Figure 4, the framework comprises three interconnected modules: the Environment, the  
 434 Reasoning module, and the Memory module.

435 The Environment module acts as the primary interface between the robot and its sur-  
 436 roundings and captures real-time sensory data. The Reasoning module is the core of the  
 437 decision-making process. It consists of several interdependent components that work to-  
 438 gether to process the sensory data and generate actionable commands for the robot. The  
 439 Memory Module stores a diverse array of past experiences, which the Prompts Generator  
 440 can recall when necessary. This module ensures that the decision-making process is not  
 441 solely reliant on current sensory inputs but also benefits from past experiences, allowing the  
 442 robot to make more informed and nuanced decisions. The following subsections explain the  
 443 reasoning and memory modules in detail.

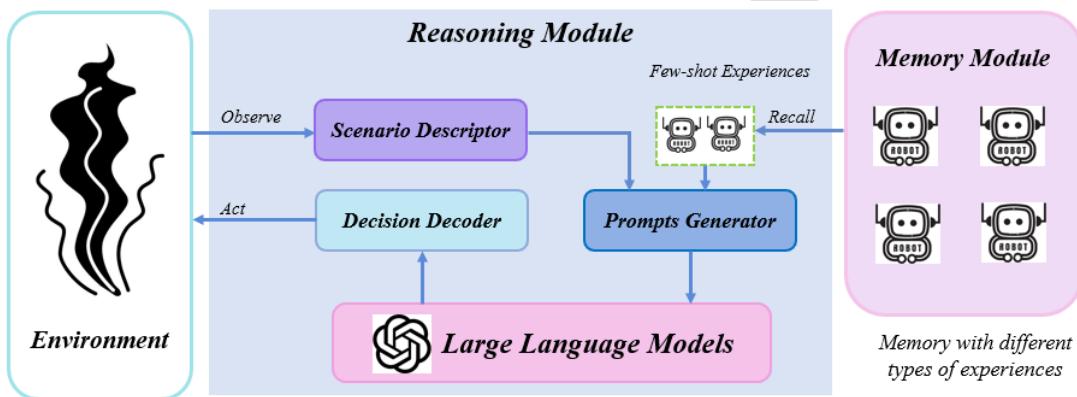


Figure 4: The framework of our knowledge-driven robotic odor source localization system. It consists of three modules: Environment, Reasoning, and Memory. The Reasoning module processes sensory inputs, combines scenario descriptions with experiences from the Memory module to generate prompts, and interprets responses from the LLM to make navigation decisions.

444 The reflection module is designed to evaluate and correct individual actions. For our  
 445 robotic odor source localization (OSL) task, it is impossible to judge the correctness of each  
 446 produced action since the robotic OSL task is a long horizon task, which means the final  
 447 result (i.e., successful finding the odor source or fail finding the odor source) is determined  
 448 by the accumulation of series of actions. The impact of single action is less significant to the  
 449 final result. Thus, we remove the reflection module for the robotic OSL task to accelerate  
 450 the decision-making process.

451 *3.2.1. Reasoning Module*

452 The Reasoning module is the core of our knowledge-driven OSL framework. It uses the  
 453 LLM to analyze sensory inputs, such as odor concentration and wind direction, to make in-  
 454 formed navigation decisions. The reasoning agent can generate effective search strategies by

455 combining current scenario descriptions with stored experiences. In the Reasoning module,  
 456 we utilize experiences derived from the Memory module and the contextual understanding  
 457 of the LLM to make decisions for the current odor localization scenario. The reasoning  
 458 procedure is illustrated in Figure 5, and includes the following steps: (1) scenario encod-  
 459 ing; (2) experience retrieval; (3) prompt generation; (4) Prompt Processing; and (5) action  
 460 decoding.

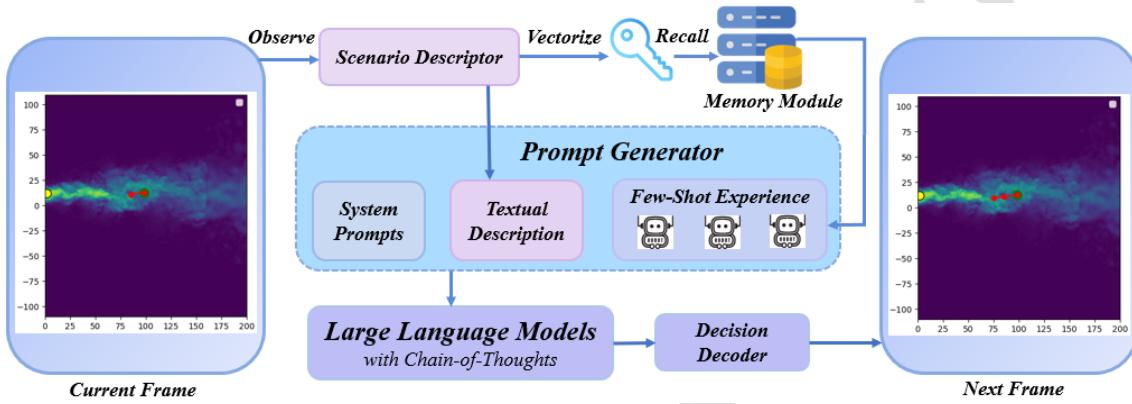


Figure 5: Reasoning module for robotic odor source localization that utilizes the common-sense knowledge of the LLM and retrieves experiences from the Memory module to make informed decisions based on the observed scenario.

461 **Scenario Encoding:** The reasoning module begins by acquiring the sensor data of the  
 462 current environmental scenario into a structured scenario descriptor, as illustrated in Figure  
 463 6. By utilizing the natural language, the scenario descriptor transcribes these sensor data  
 464 into descriptive text that describes the current scenario of the environment. This description  
 465 contains detailed information about the Environment, such as the robot’s position and odor  
 466 concentration within the search area. These descriptions are then fed into the prompt  
 467 generator and used as the keys to retrieve the relevant few-shot experiences from the Memory  
 468 module.

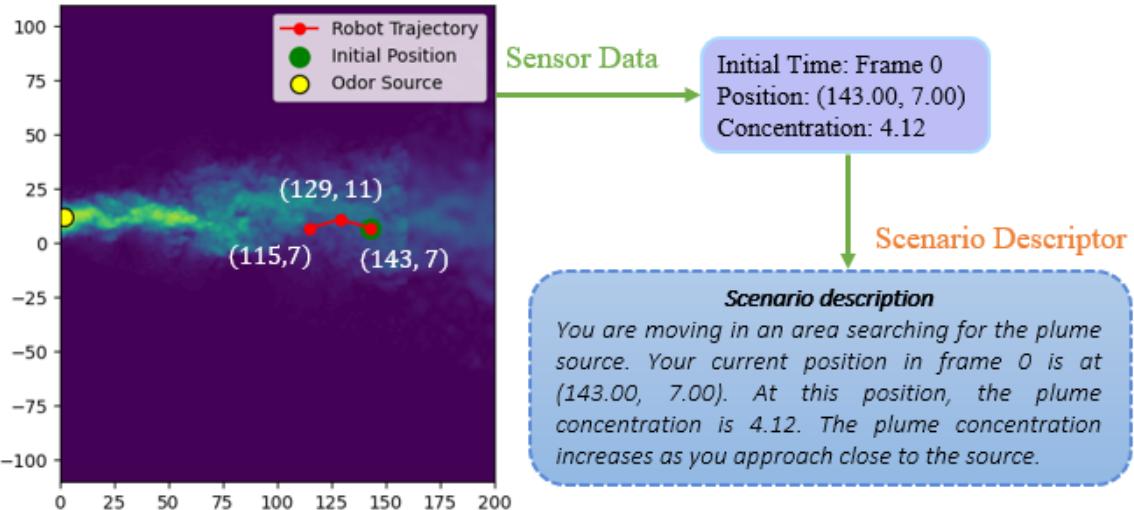


Figure 6: Scenario Descriptor transcribes the current environmental scenario from the sensor data into descriptive text.

469     **Experience Retrieval:** In this step, the current scenario is transformed into a vector  
 470 representation using an embedding model. This vector, derived from a detailed natural  
 471 language description of the current environment (including factors like robot position and  
 472 odor concentration), serves as a query to access our Memory module. By leveraging co-  
 473 sine similarity [91], this query vector is compared against the embeddings of past scenarios  
 474 stored in the database. The system retrieves the top  $k$  most similar entries, representing  
 475 past experiences with contexts closely related to the current scenario. These entries, or “  
 476 few-shot experiences,” as illustrated in Figure 7, are then integrated with the present sce-  
 477 nario description to assist in the reasoning process. This approach ensures that the robot’s  
 478 decision-making is informed by historical data, improving its responses to new yet similar  
 479 conditions.

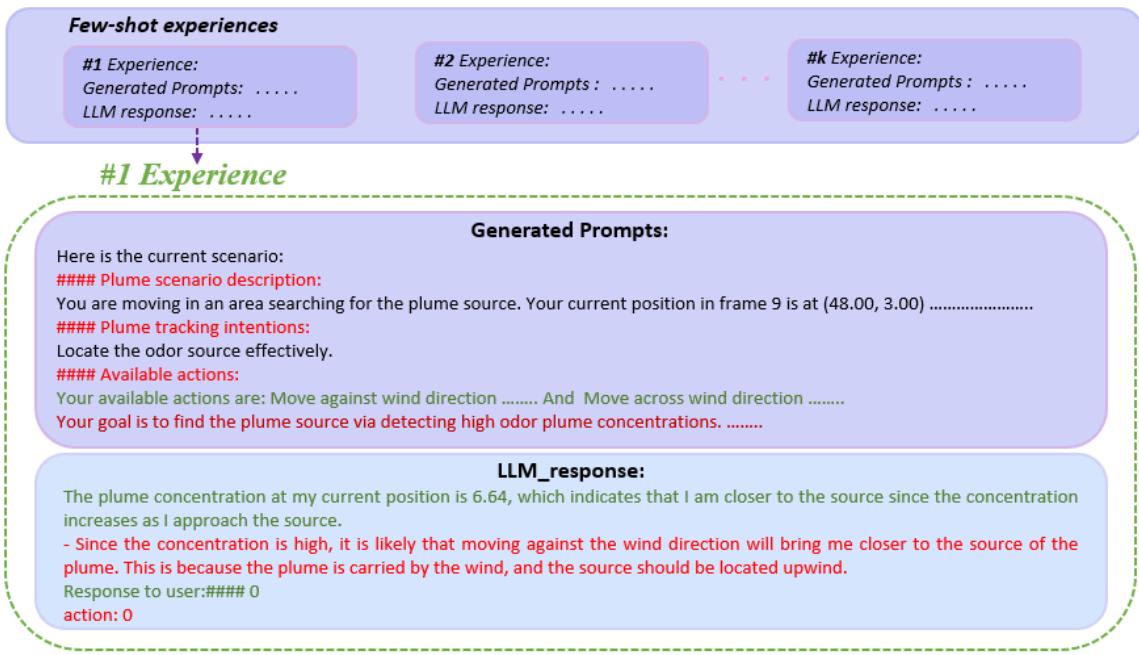


Figure 7: Few-shot experiences where each experience consists of a human-LLM dialogue pair.

480      **Prompt Generation:** This stage involves constructing comprehensive prompts that are  
 481      pivotal for the LLM’s reasoning process. The prompts are generated by combining system  
 482      prompts, textual scene descriptions, and relevant past experiences drawn from few-shot  
 483      experiences. The system prompts summarize the task, detailing the expected inputs and  
 484      outputs and specific constraints governing the reasoning process. These prompts, illustrated  
 485      in Figure 8, tailored specifically for each decision-making instance, capture the specific details  
 486      of the current situation, enabling the LLM to apply its reasoning capabilities effectively to  
 487      deduce the most appropriate navigational actions.

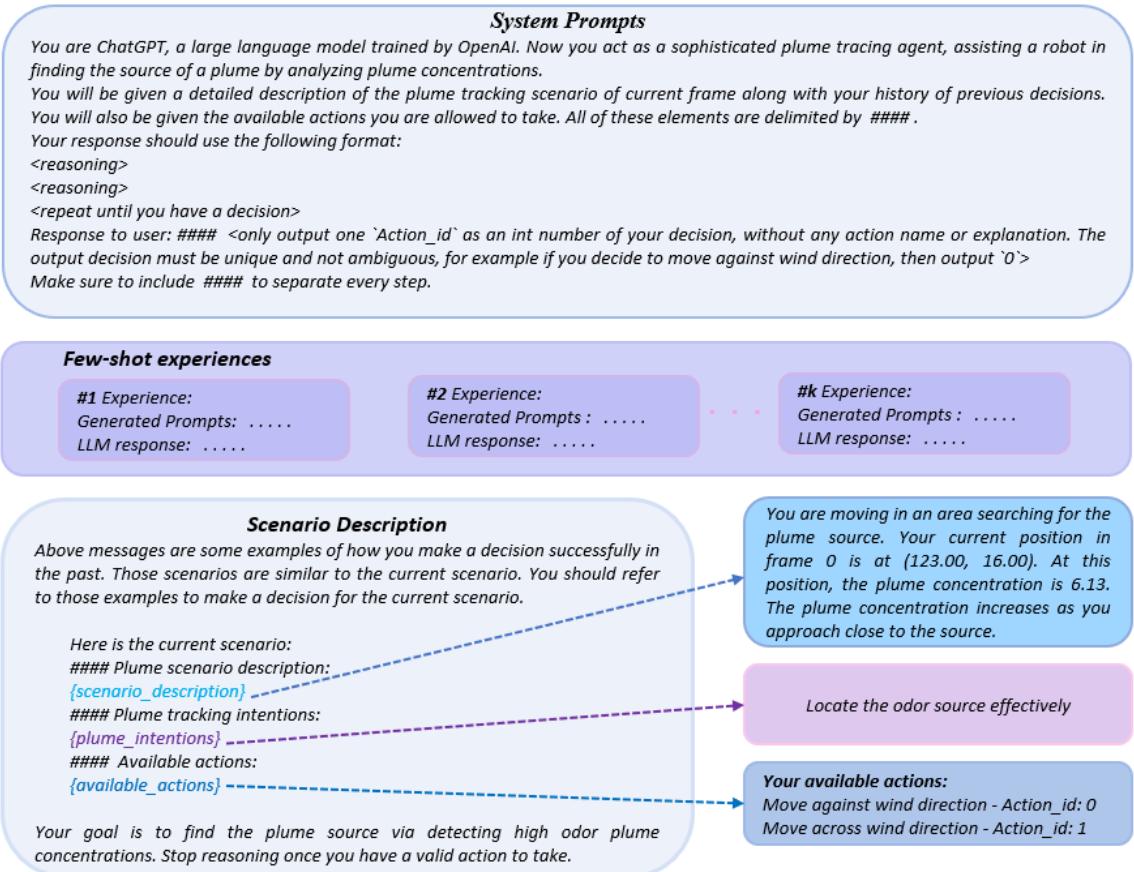


Figure 8: Prompts generator consists of system prompts, textual description, and few-shot experiences.

488     **Prompt Processing:** The complexity and variability of odor localization tasks demand  
489     a detailed reasoning process for accurate decision-making. To address this, we utilize Chain-  
490     of-Thought (CoT) prompting techniques [70], which guide the LLM to articulate its reasoning  
491     step-by-step, helping to clarify the decision-making path and reduce potential inaccuracies.  
492     By structuring the LLM's responses in this manner, we ensure that each decision is based on a  
493     logical progression of thought appropriate for the variable and complex scenarios encountered  
494     in odor localization tasks.

495     **Action Decoding:** The output from the LLM is interpreted by the action decoder  
496     within the reasoning module, translating the model's responses into specific actions for the  
497     robotic agent. The decoder is critical in converting high-level decision outcomes into specific  
498     navigational commands that the robot executes, facilitating interaction with the Environment.  
499     It creates a robust closed-loop system that refines the robot's navigational abilities  
500     through a cycle of continuous feedback and learning, leveraging both new experiences and  
501     past experiences stored in the Memory module. Figure 9 illustrates the functionality of the  
502     reasoning module, showing how environmental data is transformed into a decision-making  
503     prompt for the LLM, and subsequently, how the LLM's output is decoded into actions for

504 the robot. Detailed information on the construction and design of these prompts can be  
 505 found in Appendix Appendix A.1 and Appendix A.2.

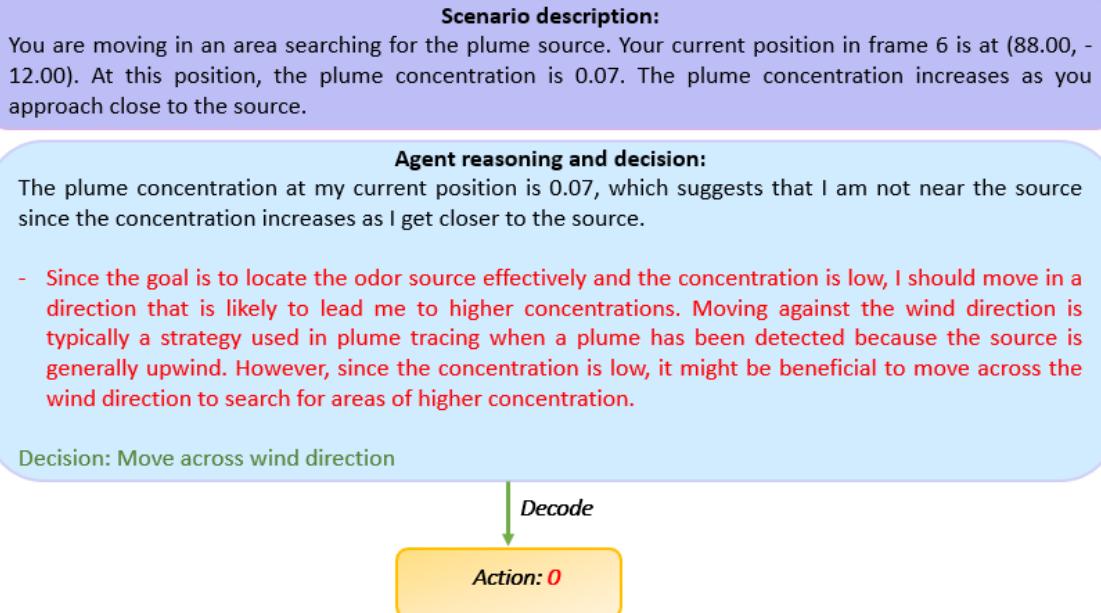


Figure 9: The scenario description provided by the scenario descriptor and the decision decoder determines the action based on the LLM’s reasoning output for robotic odor source localization.

### 506 3.2.2. Memory Module

507 The memory module of our knowledge-driven OSL framework is crucial for the robotic  
 508 agent’s ability to reason effectively in complex odor source localization tasks. It archives  
 509 extensive records of past scenarios, including scene descriptions and corresponding reasoning  
 510 processes, which are instrumental in informed decision-making in new situations. When the  
 511 robotic agent faces a new scenario, it retrieves relevant past experiences to aid decision-  
 512 making. This retrieval process involves transforming the current scenario description into a  
 513 vector. This vector acts as a key, enabling the agent to search through the memory module  
 514 for scenarios with similar conditions and their corresponding successful strategies.

515 To optimize this retrieval process, our framework employs pre-filtering technique based  
 516 on cosine similarity. While providing as many recordings as possible directly in the LLM  
 517 prompt and leaving the LLM to select the best  $n$  candidates may allow the LLM to make  
 518 much more nuanced choices, adopting this in our framework could introduce significant  
 519 challenges for the decision-making process of LLM agent, discussed below:

- 520 • **Real-time Decision-Making:** Adding an extra step for the LLM to dynamically se-  
 521 lect relevant memories would increase computational overhead, slow down the decision-  
 522 making process, and compromise performance in time-sensitive scenarios. The pro-  
 523 posed pre-filtering technique significantly enhances performance by reducing the com-  
 524 plexity of input processing, thereby minimizing inference time.

- 525 • **Finite Token Limits:** LLMs have a finite token limit, which poses significant challenges to process extensive contexts. Providing all memory recordings directly to the  
526 LLM could exceed these limits as the number of experiences grows, which can lead  
527 to the truncation of valuable data. By pre-filtering using cosine similarity, we ensure  
528 that only the most relevant and manageable subset of experiences is included within  
529 the token limit, thereby preserving critical information.
- 531 • **Scaling Memory Module:** Due to token limits and computational overhead, LLMs  
532 face significant challenges in managing large numbers of experiences. As the mem-  
533 ory module grows, providing all recordings directly to the LLM becomes increasingly  
534 unfeasible. A pre-filtering mechanism is essential to ensure scalability while maintain-  
535 ing decision quality. This mechanism allows LLMs to efficiently manage and retrieve  
536 relevant information without overwhelming computational resources.

537 By utilizing pre-filtered past experiences, the robot applies correct strategies to cur-  
538 rent environmental scenarios, ensuring a higher likelihood of successful navigation. This  
539 method of leveraging past experiences enables the robot to maintain consistent and reliable  
540 decision-making, enhancing its operational efficiency without constant updates to the mem-  
541 ory database. This static approach avoids the complexities and potential errors associated  
542 with continuous learning models, focusing instead on applying well-established knowledge  
543 to new situations.

#### 544 4. Experiments and Results

545 This section presents our study’s experimental setup and results on robotic odor source  
546 localization (OSL). We designed the experiments to evaluate the performance of the pro-  
547 posed knowledge-driven OSL framework under different configurations and conditions. We  
548 conducted an ablation study, comparing results across various scenarios, including different  
549 odor source locations and varying levels of memory utilization. Additionally, we compared  
550 our approach to a reinforcement learning method using Deep Q-Networks (DQN).

##### 551 4.1. Experimental Setup

552 We evaluated the performance of our knowledge-driven OSL framework by conducting  
553 experiments in a simulated environment, described in 3.1.1, which incorporates real-world  
554 plume data. We designed the environment to replicate the plume behavior observed in the  
555 wind tunnel experiments, ensuring that the search area and odor dispersion closely mimic  
556 actual conditions.

557 We conducted our experiments using the search area configuration detailed in Section  
558 3.1.1, following the setup of our simulated environment. Initially, we used the original odor  
559 source location at a specific point within the search area, where the nozzle continuously  
560 released odor plumes into the wind tunnel. To further assess the system’s robustness, we  
561 tested the framework by flipping the odor source location along the  $x$ -axis, as shown in  
562 Figure 10. For each experimental setting, we conducted 10 trials to capture the variability  
563 and ensure the robustness of our results. The performance metrics for evaluation included

564 Success Rate (SR), Averaged Travel Distance (TD), and Averaged Search Time (ST). The  
 565 success rate measures the proportion of trials where the robot successfully located the odor  
 566 source. We define the Averaged Travel Distance and the Averaged Search Time as the mean  
 567 values of total distance traveled and time taken across the trials.

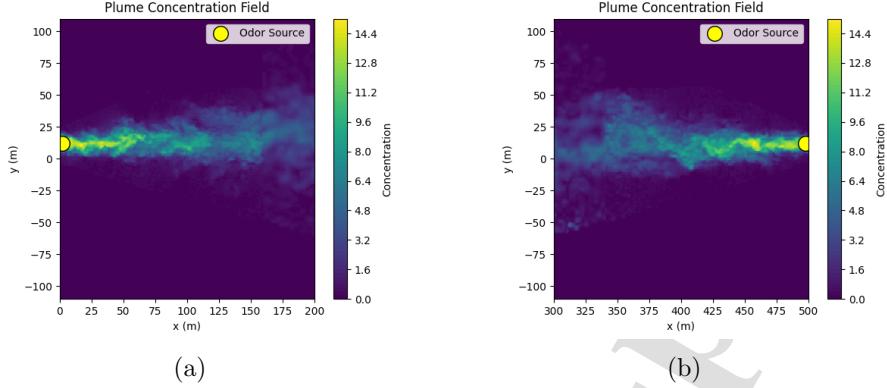


Figure 10: (a) Diagram of the search area, illustrating the plume field and the original location of the odor source. (b) Diagram of the search area after flipping the search area and odor source location along the x-axis.

#### 568 4.2. LLM-Based OSL Agent Configuration

569 We tested the robotic agent's navigation strategy under three different configurations: (i)  
 570 Adaptive-Hint, where the robot adjusts its speed based on odor concentration and receives  
 571 directional hints; (ii) Adaptive-No Hint, where the robot adjusts its speed but does not re-  
 572 ceive any hints; and (iii) Hint Only, where the robot receives hints about movement direction  
 573 but does not adjust its speed. To assess the impact of prior experiences on performance, we  
 574 varied memory utilization across three levels: 0-shot, 3-shot, and 5-shot experiences where  
 575 these experience levels refer to the number of relevant past scenarios retrieved from the  
 576 memory module to inform the robot's decision-making process. In the 0-shot experience,  
 577 the robot makes decisions based solely on the current scenario without using any past ex-  
 578 periences. In the 3-shot experience, the robot retrieves three past experiences to guide its  
 579 current decision, while in the 5-shot experience, the robot retrieves five, providing more  
 580 context to inform its actions.

#### 581 4.3. DQN-Based OSL Agent

582 In our study, to benchmark the performance of our knowledge-driven OSL framework, we  
 583 implemented a Deep Q-Network (DQN) model based on the architecture and methodology  
 584 outlined in [37]. The DQN maps the robot's sensor input states to output actions that guide  
 585 the robot's navigation toward the odor source.

586 We chose DQN, a classic deep reinforcement learning algorithm with a discrete action  
 587 space, as a baseline to compare our framework's performance with a learning-based algo-  
 588 rithm. The purpose of this comparison is not to design a state-of-the-art reinforcement

learning agent but to highlight the generalization capability of our proposed method. The LLM in our framework also outputs discrete actions, including moving against the wind or crosswind. Therefore, for a meaningful comparison, we selected DQN as it aligns with the discrete action space of the LLM-driven framework, which also outputs discrete actions.

The DQN implemented in our project is a fully connected neural network designed to process input states—comprising the robot’s position, velocity components, and concentration of the odor—and output actions that determine the robot’s heading direction. The network architecture, illustrated in Figure 11, consists of the following:

**Input Layer:** The input to the DQN is a state vector that includes the robot’s current x and y positions, the velocity components ( $u$  and  $v$ ), and the odor concentration at the robot’s location. This state vector encapsulates the essential environmental information needed for making navigation decisions.

**Hidden Layers:** The network includes two fully connected layers with 128 neurons each, using ReLU (Rectified Linear Unit) activation functions to introduce non-linearity, allowing the model to capture complex relationships between the input state and the optimal action.

**Output Layer:** This layer comprises four neurons, each representing the Q-value of one of four possible actions: moving left, down, right, or up. The action with the highest Q-value is selected as the robot’s next move.

**Training DQN Network:** We trained the DQN in the same simulated environment used to evaluate our knowledge-driven OSL framework, ensuring a consistent comparison. This environment replicates the plume behavior observed in wind tunnel experiments, with the search area defined as a 2.2x2.0 meter space. Initially, we fixed the odor source at a specific location and later flipped its position along the x-axis to test the model’s robustness. During each training step, the DQN receives a state vector that includes the robot’s position ( $x$ ,  $y$ ), velocity components ( $u$ ,  $v$ ), and odor concentration. Based on this input, the network generates Q-values for the four possible actions: moving left, down, right, or up. The robot then selects and executes the action with the highest Q-value.

We designed the reward system to encourage the robot to locate the odor source efficiently. Mathematically, we define the reward  $r$  as follows:

$$r = \begin{cases} 100 & \text{if } d < \delta; \\ -100 & \text{if } OB \text{ or } t < t_{max}; \\ -0.1 & \text{if } dist(P_4, P_1) < 3; \\ c_{norm} & \text{if } c_{norm} > 0 \text{ and } dist(P_4, P_1) \geq 3; \\ 0 & \text{otherwise,} \end{cases} \quad (6)$$

where  $d$  is the distance of the robot agent to the odor source,  $\delta$  is the distance threshold to consider the goal reached,  $OB$  indicates that the robot moves out of bounds,  $t$  is the current time step,  $t_{max}$  is the maximum allowed time steps,  $P_1$  and  $P_4$  are the robot’s positions at the first and fourth most recent steps,  $dist(P_4, P_1)$  represents the Euclidean distance between the positions  $P_1$  and  $P_4$ ,  $c_{norm}$  is the normalized odor concentration at the robot position.

The robot receives a high positive reward,  $r = 100$  if it successfully reaches the odor source, and a negative reward of  $r = -100$  if the robot either moves outside the designated

625 search area (i.e., moves beyond the boundaries of the environment) or exceeds the maximum  
 626 allowed time for the search. If the robot's movement over the last four steps is minimal,  
 627 indicating a lack of significant progress (i.e., it has not moved much from its position four  
 628 steps ago), we apply a small penalty of  $r = -0.1$  to discourage it from staying stationary.  
 629 When the robot detects an odor concentration and has moved significantly, we reward it  
 630 based on the concentration at its current position, encouraging it to move toward areas with  
 631 higher odor concentrations. If none of these conditions are met, the reward defaults to  $r = 0$ ,  
 632 indicating no progress or penalty. The goal is to maximize cumulative future rewards by  
 633 minimizing the distance to the odor source and reducing travel time.

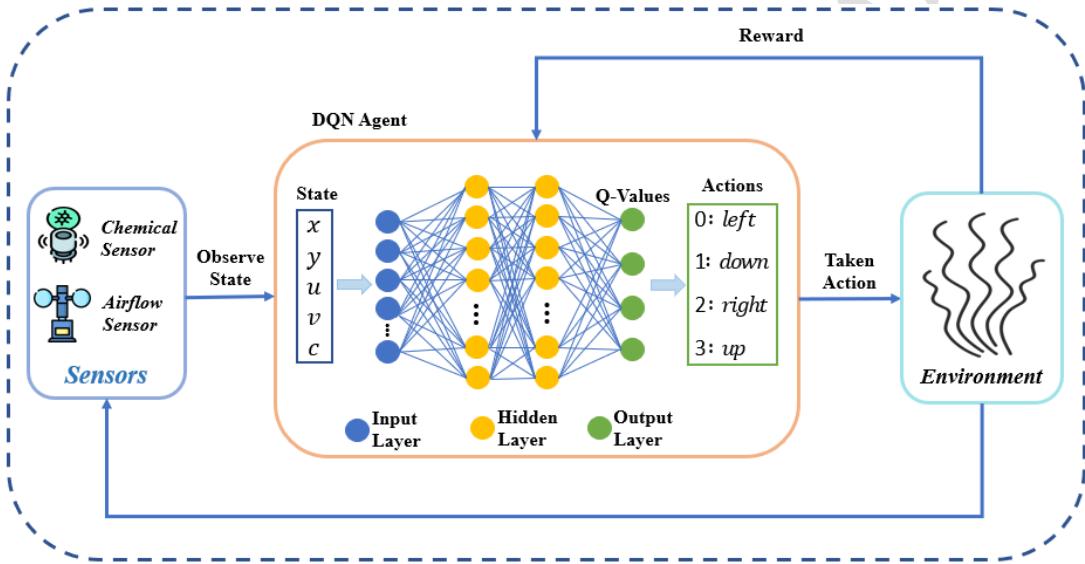


Figure 11: The Deep Q-Network (DQN) architecture for robotic odor source localization. The framework consists of sensors, a DQN agent, and an environment.

634 We conducted training over multiple episodes, each representing a complete search se-  
 635 quence starting from a random initial position until the robot either finds the odor source or  
 636 reaches the episode's time or boundary limits. We trained the DQN with 1,000, 3,000, and  
 637 5,000 episodes using an epsilon-greedy policy, where the robot initially explores randomly  
 638 (high epsilon) and gradually shifts to exploiting the learned policy (epsilon decreases). We  
 639 set the discount factor for future rewards to 0.99, ensuring that the robot values immediate  
 640 rewards while considering long-term benefits. We evaluated the DQN's performance using  
 641 the same metrics as in our knowledge-driven OSL framework: Success Rate (SR), Averaged  
 642 Travel Distance (TD), and Averaged Search Time (ST). These metrics comprehensively  
 643 assess the DQN's ability to navigate the robot to the odor source.

#### 644 4.4. Sample Trials

645 To illustrate the performance of our knowledge-driven OSL framework under various  
 646 memory settings, we demonstrated sample trials using 0-shot, 3-shot, and 5-shot memory

647 configurations. We design the trials to illustrate the impact of different levels of memory  
 648 integration on the robot's ability to navigate the plume field and accurately locate the odor  
 649 source.

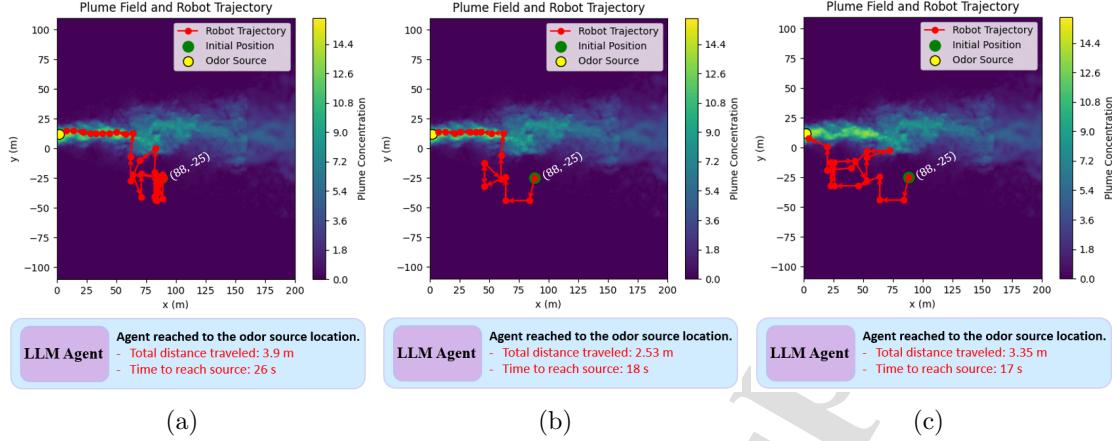


Figure 12: Sample trials of the knowledge-driven OSL framework for past experience: (a) 0-shot memory,  
 (b) 3-shot memory, and (c) 5-shot memory.,

650 The LLM agent can choose one of two actions: moving cross wind or moving against the  
 651 wind. When the action is crosswind, the robot moves in a direction that is 90 degrees to the  
 652 wind direction. In Figure 12, the robot initially performs crosswind movements as no plume  
 653 concentration is detected. When the robot still fails to detect any plume in subsequent posi-  
 654 tions, it continues to choose the crosswind action. This results in back-and-forth movements,  
 655 which are a natural part of the exploratory process during crosswind actions, allowing the  
 656 robot to search for the plume in a broad area. When the robot detects a plume concentra-  
 657 tion that is larger than the threshold, the LLM agent switches to the against-wind action,  
 658 guiding the robot upwind to follow the plume. The observed redundancy in the trajectory  
 659 is expected and reasonable for this exploratory strategy. If the plume had been detected  
 660 during the initial crosswind movements, the robot would have immediately switched to the  
 661 upwind (surge) behavior, avoiding repeated movements. The behavior reflects the LLM's  
 662 decision-making process and the inherent challenges of locating odor sources in dynamic  
 663 environments.

664 In the 0-shot trial (Figure 12(a)), the robot operates without leveraging prior experiences,  
 665 relying solely on the LLM's real-time reasoning to navigate towards the odor source. The  
 666 trajectory shows the robot navigating towards the odor source with some exploration due  
 667 to the lack of memory. Despite this, the robot successfully locates the odor source, demon-  
 668 strating the inherent capability of the LLM to interpret and act on sensory inputs effectively.  
 669 The 3-shot trial (Figure 12(b)) exhibits enhanced navigation efficiency as the robotic agent  
 670 utilizes memory recall to retrieve three similar past experiences from the memory module.  
 671 The trajectory is more direct, with fewer deviations compared to the 0-shot trial, indicat-  
 672 ing the benefits of incorporating relevant past experiences into the decision-making process.

673 This results in quicker and more accurate localization of the odor source. Conversely, the  
 674 5-shot trial (Figure 12(c)) incorporates five past experiences, providing a richer historical  
 675 context for the decision-making process. While this setup allows for a highly informed ap-  
 676 proach, it also introduces the risk of over-reliance on historical data, occasionally leading  
 677 to suboptimal paths that may not perfectly align with the current environmental setup. It  
 678 highlights the potential drawbacks of excessive memory usage, where too much reliance on  
 679 past data can reduce the adaptability and efficiency of navigation.

680 Overall, these sample trials provide valuable insights into how different levels of memory  
 681 integration affect the performance of robotic odor source localization. They illustrate a  
 682 clear trade-off between memory use and navigational efficiency, emphasizing the importance  
 683 of finding an optimal balance to maximize the effectiveness of the knowledge-driven OSL  
 684 framework in dynamic and varied environments.

#### 685 4.5. Ablation Study

686 The ablation study aimed to rigorously evaluate the impact of different memory util-  
 687 ization levels and configuration settings on the performance of our knowledge-driven OSL  
 688 framework. This study provides key insights into how varying the integration of past experi-  
 689 ences and adaptive strategies influences the robot’s ability to locate odor sources effectively.

Table 2: Performance metrics of different levels of memory utilization for OSL.

Memory	Success Rate (SR) ↑	Averaged Travel Distance (TD) ↓	Averaged Search Time (ST) ↓
0-shot	8/10	2.35	15.5
3-shot	<b>10/10</b>	<b>2.19</b>	<b>14.5</b>
5-shot	9/10	2.20	14.7

690 Table 2 presents the performance metrics across different levels of memory utilization  
 691 within the OSL framework: 0-shot, 3-shot, and 5-shot. The 3-shot memory setting demon-  
 692 strated the best overall performance, achieving a high success rate (SR) of 10/10, along  
 693 with the lowest averaged travel distance (TD) and search time (ST). This suggests that  
 694 utilizing a moderate amount of past experiences (3-shot) optimally enhances the robot’s  
 695 efficiency in navigating and locating the odor source. In contrast, the 0-shot setting, which  
 696 depends solely on real-time reasoning without memory recall, resulted in lower success rates  
 697 and longer travel distances. The 5-shot setting, while slightly better than 0-shot, did not  
 698 outperform the 3-shot configuration, indicating that over-reliance on historical data does  
 699 not necessarily lead to improved performance.

700 Our results show that, the 3-shot configuration perform optimally in our OSL framework  
 701 as it retrieves three highly relevant examples with a higher similarity threshold, maintaining  
 702 a concise focus on the most aligned scenarios. This setup enhances the LLM’s reasoning  
 703 process by providing precise, contextually relevant guidance.

704 In contrast, the 0-shot configuration lacks contextual memory, relying solely on general-  
 705 ized reasoning, which often leads to suboptimal navigation and reduced success rates. On

706 the other hand, the 5-shot configuration introduces additional diversity by retrieving two  
 707 more past experiences. While this can offer a broader historical context, it also raises the  
 708 likelihood of including less relevant or conflicting scenarios, especially in environments with  
 709 limited variability like ours. This increased diversity sometimes leads to over-reliance on  
 710 historical data, which can produce suboptimal decision-making and less efficient navigation  
 711 paths. Furthermore, the 5-shot configuration faces several other challenges that impact the  
 712 LLM’s reasoning process. One such challenge is scenario redundancy, which occurs when  
 713 additional memories overlap with existing ones, contributing limited new insights and in-  
 714 troducing unnecessary complexity to the LLM’s reasoning process. Additionally, including  
 715 more memories in low-variability environments increases computational and decision-making  
 716 burden, which overwhelms the LLM and reduces its ability to generalize effectively to the  
 717 current scenario. Lastly, using a larger set of examples increases the risk of overfitting,  
 718 where the reasoning process becomes too focused on specific patterns from the retrieved  
 719 data, which makes it less adaptable to new or varied scenarios.

Table 3: Performance metrics of different settings of configurations for OSL.

Configuration	Success Rate (SR) $\uparrow$	Averaged Travel Distance (TD) $\downarrow$	Averaged Search Time (ST) $\downarrow$
Adaptive-Hint	<b>10/10</b>	<b>1.93</b>	<b>13.5</b>
Adaptive-No Hint	8/10	2.01	13.2
Hint Only	4/10	1.44	6.8

720 Table 3 summarizes the performance metrics for different configuration settings: Adaptive-  
 721 Hint, Adaptive-No Hint, and Hint Only, all utilizing the 3-shot memory setting, which  
 722 showed the best performance. Among these, the Adaptive-Hint configuration demonstrated  
 723 superior performance, achieving the highest success rate (SR) of 10/10 and the most effi-  
 724 cient travel distance (TD) and search time (ST). This configuration adjusts the robot’s speed  
 725 based on the detected odor concentration (adaptive speed) and provides hints or guidance  
 726 on the robot’s movement direction based on this concentration data (hints about the robot’s  
 727 movement). This combined approach of adapting speed and using directional hints proved  
 728 to be the most effective in guiding the robot to locate the odor source efficiently. In con-  
 729 trast, the Adaptive-No Hint configuration showed a slightly lower success rate and efficiency,  
 730 indicating that the absence of hints affects the robot’s ability to navigate optimally. The  
 731 Hint Only configuration, which relied solely on directional hints without adaptive speed  
 732 adjustments, had the lowest success rate and performance metrics. This underscores the  
 733 importance of combining adaptive strategies with memory utilization to enhance the robot’s  
 734 navigation capabilities.

735 The results from the ablation study reveal that the optimal approach for robotic odor  
 736 source localization involves a moderate level of memory utilization (3-shot) combined with  
 737 adaptive strategies (Adaptive-Hint). This balance maximizes the robot’s ability to accu-  
 738 rately and efficiently locate odor sources by effectively combining past experiences with  
 739 real-time environmental feedback.

740 4.6. Comparative Analysis with DQN and Moth-inspired Method

741 In this section, we compared the performance of our knowledge-driven OSL framework  
 742 with a DQN based approach [37] and a Moth-inspired method [92]. The comparison focuses  
 743 on key performance metrics, including success rate (SR), averaged travel distance (TD), and  
 744 averaged search time (ST), across both original and flipped odor source locations.

Table 4: Performance metrics of different OSL algorithms for the original odor source location.

OSL Algorithms	Training Episodes	Success Rate (SR) ↑	Averaged Travel Distance (TD) ↓	Averaged Search Time (ST) ↓
(LLM) Adaptive-Hint (3-shot)	-	<b>10/10</b>	<b>1.93</b>	<b>13.5</b>
Moth-inspired method [92]	-	7/10	2.28	16.2
DQN [37]	1000	8/10	1.45	13.1
	<b>3000</b>	9/10	1.91	16.3
	5000	9/10	3.5	22.8

745 Table 4 presents the performance metrics for the original odor source location. The  
 746 knowledge-driven OSL framework with the Adaptive-Hint (3-shot) configuration achieved  
 747 the highest success rate (10/10). It demonstrated the most efficient navigation, with the  
 748 shortest averaged travel distance (1.93) and averaged search time (13.5). This performance  
 749 underscores the effectiveness of combining memory recall with adaptive strategies in the  
 750 knowledge-driven approach. In contrast, the DQN approach, despite showing competitive  
 751 performance in some metrics, exhibited inconsistencies. While the DQN trained with 3,000  
 752 episodes achieved a success rate of 9/10, its performance in terms of averaged travel distance  
 753 (1.91) and averaged search time (16.3) was less efficient than that of the knowledge-driven  
 754 approach. Additionally, as the training episodes increased to 5,000, the DQN's performance  
 755 declined, indicating the model's limitations in maintaining consistent efficiency and accuracy  
 756 across varying conditions. While showing reasonable performance with a success rate of  
 757 7/10, the Moth-inspired method fell short compared to the knowledge-driven and DQN  
 758 approaches, particularly in terms of average travel distance and search time. It suggests  
 759 that while biologically inspired methods can be effective, they may not fully capture the  
 760 complexities of navigating varied and dynamic environments.

Table 5: Performance metrics of different OSL algorithms for the flipped odor source location.

OSL Algorithms	Training Episodes	Success Rate (SR) $\uparrow$	Averaged Travel Distance (TD) $\downarrow$	Averaged Search Time (ST) $\downarrow$
(LLM) Adaptive-Hint (0-shot)	-	<b>7/10</b>	<b>2.53</b>	<b>15.7</b>
(LLM) Adaptive-Hint (3-shot)	-	<b>8/10</b>	<b>2.65</b>	<b>16.2</b>
(LLM) Adaptive-No Hint (3-shot)	-	<b>4/10</b>	<b>3.28</b>	<b>19.7</b>
Moth-inspired method [92]	-	7/10	3.85	23.8
DQN [37]	1000	0/10	51.8	99
	3000	0/10	46.12	99
	5000	0/10	56.68	99

761 Table 5 summarizes the performance metrics after flipping the odor source location along  
 762 the x-axis. The knowledge-driven OSL framework, particularly the Adaptive-Hint (3-shot)  
 763 configuration, continued to show strong performance with an 8/10 success rate. It highlights  
 764 the framework's adaptability to changes in environmental conditions. In contrast, the DQN  
 765 approach completely failed, with a 0/10 success rate across all training levels, underscoring  
 766 its inability to generalize to new, unseen conditions. While maintaining a 7/10 success rate,  
 767 the Moth-inspired method again showed less efficiency in terms of travel distance and search  
 768 time compared to the knowledge-driven approach. These results indicate that the DQN and  
 769 Moth-inspired methods lack the robustness and adaptability required for effectively handling  
 770 dynamic and altered environments, as evidenced by the significant drop in performance when  
 771 the odor source location was flipped.

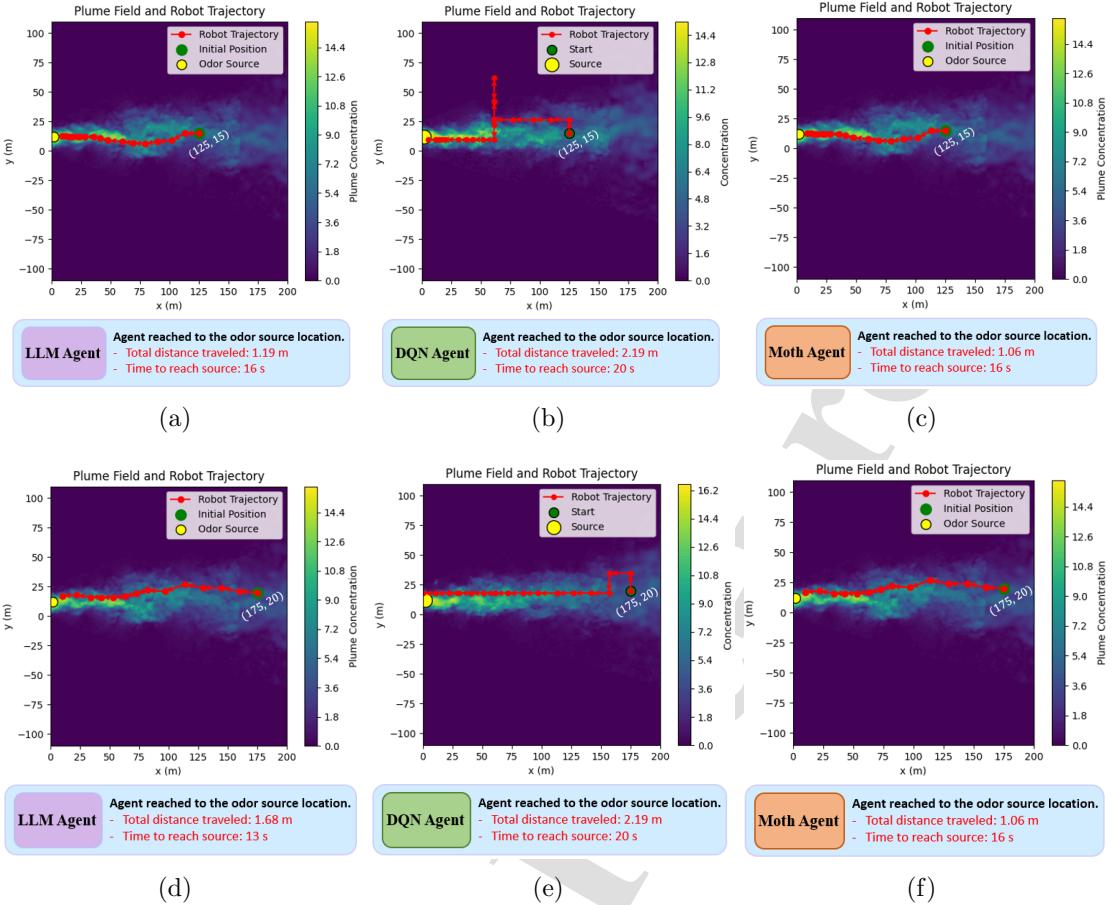


Figure 13: Examples of robotic agent navigation to the odor source location. (a) and (d) show that the robotic agent reaches the odor source location using the LLM’s reasoning, demonstrating efficient navigation and effective decision-making, (b) and (e) show that the robotic agent reaching the odor source location using the DQN approach, which, while sometimes successful, often results in less efficient paths and higher search times, and (c) and (f) show that the robotic agent reaching the odor source location using the Moth-inspired method, demonstrating efficient navigation in which agent is in the plume.

772 Figures 13 and 14 visually depict the trajectories of the robotic agents using the different  
 773 approaches for both the original and flipped odor source locations. For the original odor  
 774 source location, the knowledge-driven agent (Figures 13(a) and 13(d)) demonstrated effi-  
 775 cient and direct navigation to the odor source, showcasing its effective decision-making and  
 776 adaptability. On the other hand, the DQN agent (Figures 13(b) and 13(e)) exhibited less  
 777 consistent and often less direct paths, with a notable decline in performance when training  
 778 episodes were increased, highlighting the limitations in its generalization capabilities. The  
 779 Moth-inspired method (Figures 13(c) and 13(f)) showed a consistent trajectory but was less  
 780 efficient overall compared to the knowledge-driven approach.

781 In the flipped odor source scenario (Figure 14), the knowledge-driven agent (Figure 14(a))  
 782 continued to perform effectively, maintaining adaptability to the new environmental condi-

783 tions. In contrast, the DQN agent failed to locate the odor source entirely (Figure 14(b)),  
 784 reinforcing its struggle with generalization. The Moth-inspired method (Figure 14(c)), al-  
 785 though maintaining some success, again demonstrated inefficiency in navigating the altered  
 786 environment, with the robot often taking longer paths to the odor source.

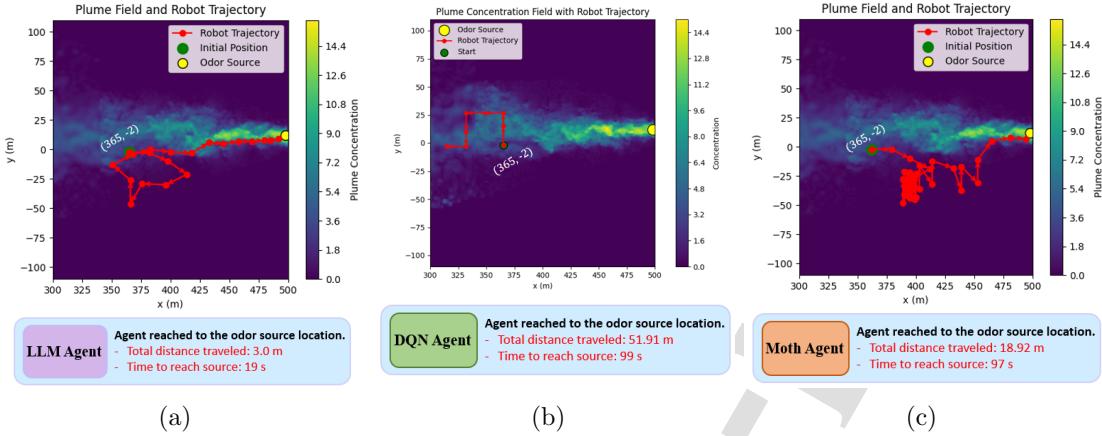


Figure 14: a) Robotic agent reached to the odor source location using LLM’s reasoning: 3-shot memory, (b) shows that DQN agent failed to reach the odor source location where the odor source location is flipped along the  $x$ -axis, and (c) show that the robotic agent reaching the odor source location using the Moth-inspired method, often results in less efficient paths and higher search times.

787 The comparative analysis clearly demonstrates the superior performance of the knowledge-  
 788 driven OSL framework over the DQN and Moth-inspired methods. Integrating large lan-  
 789 guage models (LLMs) with memory-assisted decision-making enhances the robot’s ability to  
 790 locate odor sources efficiently and provides robust adaptability across varied environmental  
 791 scenarios. The DQN approach, while effective under certain conditions, needs to improve  
 792 with generalization, particularly in dynamic or altered environments, making it less reliable  
 793 for real-world applications. The Moth-inspired method, although consistent, lacks the ef-  
 794 ficiency and adaptability needed for optimal performance. Overall, the knowledge-driven  
 795 framework stands out as the most effective solution for robotic odor source localization in  
 796 complex and dynamic environments.

#### 797 4.7. Real World Experiment Results of OSL

##### 798 4.7.1. Search Area

799 In the real-world experimental setup of our odor source localization (OSL) study, we  
 800 defined the search area to test the robot’s navigation and detection capabilities in a controlled  
 801 environment. The search area covered a two-dimensional space of 8.2 meters by 3.3 meters,  
 802 as shown in Figure 15(a). Ethanol vapor, chosen for its non-toxic properties and frequent  
 803 use in OSL research [93], served as the odor source. We employed a humidifier as the plume  
 804 generator to disperse the ethanol vapor throughout the experiments. To ensure a uniform  
 805 airflow direction across the search area, we placed a single fan behind the odor source,  
 806 facilitating the diffusion of the plume, as presented in Figure 15(b).

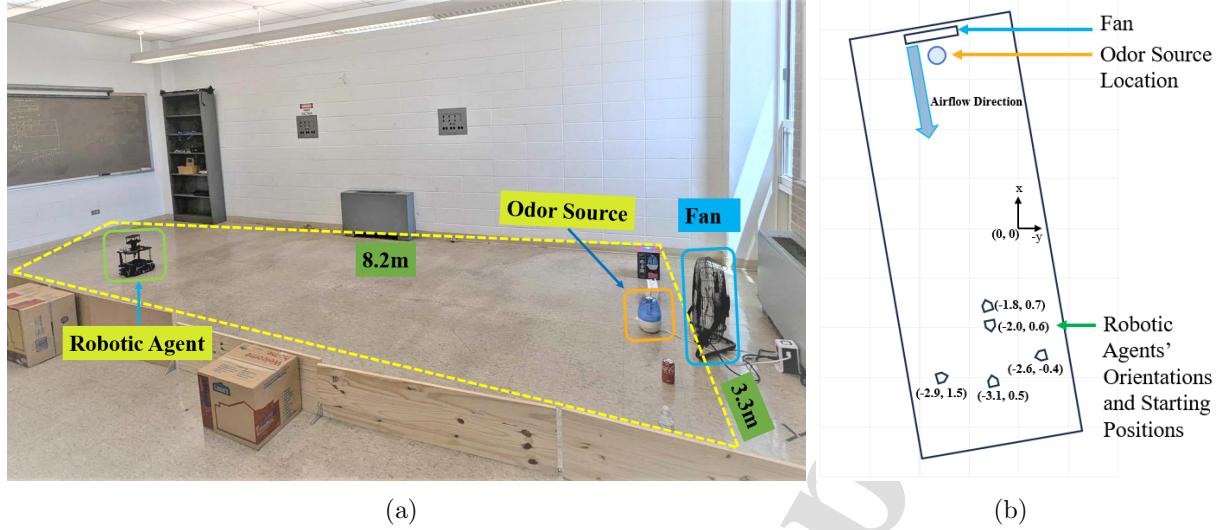


Figure 15: a) An experimental setup shows the robotic agent in the search area, with a humidifier releasing ethanol vapor as the odor source and a fan to establish airflow. The agent starts from a downwind location, navigating toward the ethanol source. (b) A schematic representation of the search area with a fan setup creates a uniform airflow. The diagram includes the odor source location and the initial robot positions, illustrating the experimental conditions for OSL tests.

#### 807 4.7.2. Robotic Agent Specifications

808 The Turtlebot3 mobile robot platform was utilized in our experiments, equipped with  
 809 built-in sensors and components tailored for navigation and sensory tasks in odor source  
 810 localization. The robot's sensor suite includes a 360-degree LDS-02 Laser Distance Sensor for  
 811 measuring the distance, a WindSonic Anemometer for measuring wind speed and direction,  
 812 and an MQ3 alcohol sensor to detect chemical plume concentrations. Figure 16 shows the  
 813 robotic agent used in the experiments.

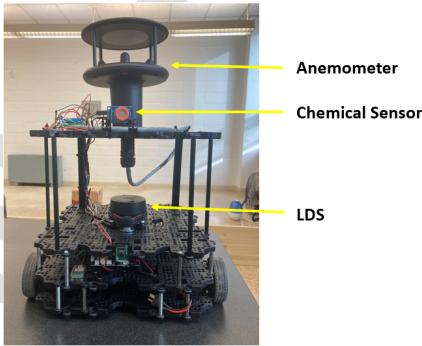


Figure 16: Robotic agent used in the experiments, equipped with various sensors including a chemical sensor and an anemometer to measure odor concentrations and airflow, respectively.

814 Powered by a Raspberry Pi 4 CPU, the Turtlebot3 runs on Ubuntu 20.04 with the

815 Robot Operating System (ROS) Noetic, allowing for seamless integration and real-time  
 816 communication. The robot operates in conjunction with a remote personal computer (PC)  
 817 via a local area network, which facilitates the control and data exchange necessary for  
 818 effective robotic navigation and data processing in complex odor localization scenarios. This  
 819 setup ensures robust performance in dynamic environments, enabling precise control and  
 820 responsiveness during the localization tasks.

821 *4.7.3. OSL Experimental Design*

822 We designed the experiments to evaluate the effectiveness of our proposed Knowledge-  
 823 driven OSL framework, which leverages Large Language Models (LLMs) in a real-world  
 824 setup. Additionally, we assessed the performance of robotic OSL using a Deep Q-Network  
 825 (DQN) model based on Reinforcement Learning (RL). Both approaches were tested under  
 826 controlled conditions to determine their performance and adaptability in real-world scenar-  
 827 ios.

828 We incorporated a dynamic memory setting in the knowledge-driven framework to en-  
 829 hance decision-making across various environmental conditions. We tested three configu-  
 830 rations: Adaptive-Hint, Adaptive-No Hint, and Hint Only, with memory settings varied  
 831 across 0-shot, 3-shot, and 5-shot conditions to assess the impact of accumulated knowledge  
 832 on performance. Each configuration was designed to evaluate the framework's efficiency and  
 833 success rate, with detailed specifics provided in Section 4.1. Trials were conducted under  
 834 standardized conditions, with success defined by the robot's proximity to the odor source—a  
 835 trial was deemed successful if the robot reached within 0.6 meters of the target location. If  
 836 the robot failed to locate the odor source within 120 seconds, the trial was considered a fail-  
 837 ure. The robotic agent's control parameters were set to ensure optimal performance during  
 838 operation. The initial linear velocity ( $l_v$ ) and angular velocity ( $\omega_c$ ) were set at 0.1 m/s and  
 839 0.3 rad/s, respectively, for the Knowledge-driven OSL framework. To improve navigation  
 840 toward the odor source, the robot employed an adaptive speed mechanism, adjusting its  
 841 linear velocity based on the concentration of the detected odor. The adaptive speed was  
 842 determined using the following formula:

$$l_v = \begin{cases} 0.1 & \text{if } c > \delta_c; \\ 0.08 & \text{otherwise,} \end{cases} \quad (7)$$

843 where  $c$  represents the concentration of the odor at the robot's current position, and  $\delta_c$  is  
 844 the threshold concentration level used to determine whether the robot should increase its  
 845 linear velocity.

846 For the DQN algorithm, trained over 5000 episodes, the initial values of  $l_v$  and  $\omega_c$  were  
 847 set at 0.02 m/s and 0.02 rad/s, respectively. This training aimed to optimize the robot's  
 848 ability to navigate toward the odor source based on real-time sensor inputs, without relying  
 849 on a pre-existing knowledge base.

850 Each experimental setup involved five test runs, initialized from the same five starting  
 851 positions. Figure 15(b) shows these starting positions and the airflow setups used during the  
 852 experimental runs. The experiments evaluated each algorithm using predefined performance

853 metrics: Success Rate (SR), averaged Travel Distance (TD), and averaged Search Time  
 854 (ST), as detailed in Section 4.1. The comprehensive testing of both approaches allows  
 855 us to comprehensively assess the practical applicability, efficiency, and adaptability of each  
 856 method in dynamic environments, offering insights into their potential uses in challenging  
 857 OSL tasks.

#### 858 4.7.4. Sample Trials

859 To demonstrate the practical application and validate the performance of our Knowledge-  
 860 driven OSL framework, we conducted specific trial runs and monitor each step and record  
 861 the robot's actions and decisions. In a representative trial run using the Knowledge-driven  
 862 OSL framework with the 3-shot memory setting under the Adaptive-Hint configuration, the  
 863 robot was initialized at a starting point on the edge of the search area. Upon initialization,  
 864 the robot immediately began processing environmental data using its chemical sensor and  
 865 anemometer. At 5 seconds, it detected a significant chemical concentration, indicating the  
 866 presence of an odor plume and triggering its navigation toward the source. The robot  
 867 adjusted its path based on the intensity of ethanol vapor and wind direction data. By 56.2  
 868 seconds, the robot successfully reached within 0.6 meters of the odor source, well within the  
 869 120-second time limit set for the experiment.

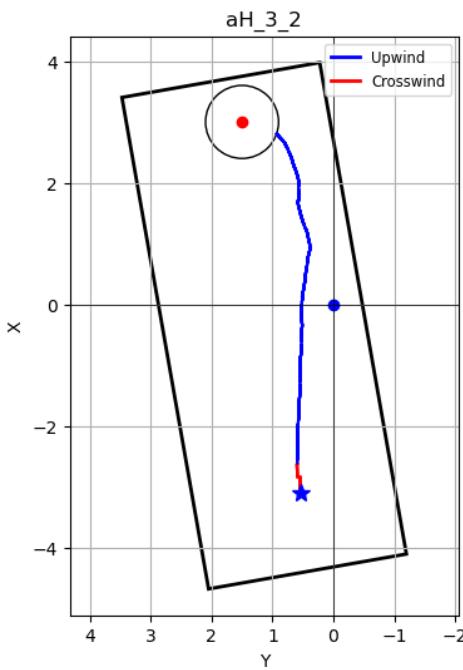
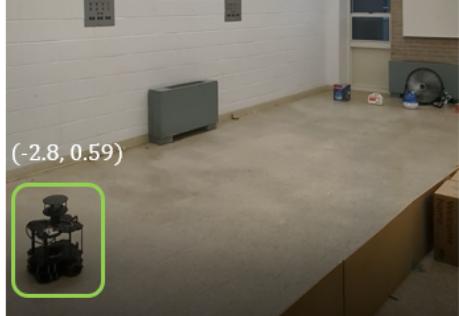


Figure 17: Robot's navigation path during an OSL test under the Adaptive-Hint configuration with a 3-shot memory setting. The robot's trajectory is depicted, showing its approach toward the odor source, marked by the red dot. The blue and red lines represent the robot's movement upwind and crosswind, respectively.

870 The integration of prior experiences allowed the robot to optimize its path efficiently,  
 871 avoiding unnecessary detours and steadily approaching the odor source. The trial conclu-

872 sively demonstrated the framework's effectiveness, with the robot achieving its goal sig-  
 873 nificantly under the allotted time. The robot's trajectory during this trial is presented in  
 874 Figure 17, while snapshots of the robot's navigation at different times are shown in Figure  
 875 18. These images illustrate the robot's decision-making process as it responds to varying  
 876 plume concentrations.

**Scenario description:**

You are moving in an area searching for the plume source. Your current position is at (-2.8, 0.59). At this position, the plume concentration is 649.00.

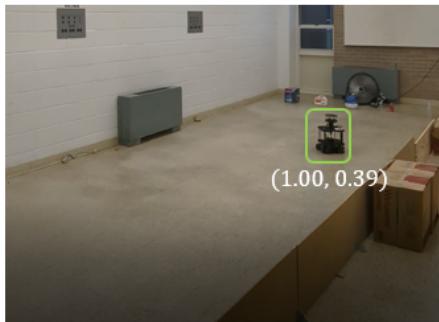
**Agent reasoning and decision:**

The plume concentration at the current position is 649.00, which is considered low. The intention is to locate the odor source effectively, which suggests that I should move to areas with potentially higher concentration.

- Since the concentration is low, it is less likely that moving against the wind direction will lead me to the source. The source should be in the direction where the concentration increases.
- Moving across the wind direction could potentially lead to areas with higher concentration as it suggests exploring new areas where the plume might be more concentrated.

Decision: Move across the wind direction

(a)

**Scenario description:**

You are moving in an area searching for the plume source. Your current position is at (1.00, 0.39). At this position, the plume concentration is 667.00.

**Agent reasoning and decision:**

The plume concentration at the current position is 667.00, which indicates that I am in the presence of the plume since the concentration increases as I approach the source.

- Since the goal is to locate the odor source effectively and the concentration is already high, I should move against the wind direction to trace the plume back to its source.

Decision: Move against the wind direction

(b)

Figure 18: Snapshots of robotic agent navigation to the odor source location in different time. (a) represents the LLM's reasoning and decision-making at position (-2.8, 0.59) of the robotic agent. (b) represents the LLM's reasoning and decision-making at position (1.00, 0.39) of the robotic agent.

877 *4.7.5. Experimental Results*

878 In our real-world experiments, the Knowledge-driven OSL framework under different  
 879 configurations— Adaptive-Hint, Adaptive-No Hint, and Hint Only—each tested with 0-  
 880 shot, 3-shot, and 5-shot memory settings. The performance was primarily measured using

881 Success Rate (SR), averaged Travel Distance (TD), and averaged Search Time (ST) metrics.

882 The Adaptive-Hint configuration with the 3-shot memory setting emerged as the most effective,  
 883 achieving a perfect success rate with optimized travel distance and search time. This  
 884 configuration struck a successful balance between utilizing historical data and adapting to  
 885 real-time environmental changes, leading to efficient navigation and odor source localization.  
 886 In contrast, the Adaptive-No Hint configuration demonstrated moderate success. While it  
 887 maintained adaptability, the absence of hints led to slightly less efficient performance, parti-  
 888 cularly in the 3-shot setting, where longer search times and increased travel distances were  
 889 observed. The Hint Only configuration showed that strategic hints alone could enhance  
 890 decision-making, though it was less effective compared to the adaptive configurations, which  
 891 dynamically adjusted based on environmental inputs. The results of these configurations  
 892 are summarized in Table 6.

Table 6: Performance metrics for the original odor source location.

Configuration	Memory	Success Rate (SR) ↑	Averaged Travel Distance (TD) ↓	Averaged Search Time (ST) ↓
Adaptive-Hint	0-shot	0.8	5.16	64.18
	<b>3-shot</b>	<b>1.0</b>	<b>5.79</b>	<b>53.03</b>
	5-shot	1.0	5.85	53.81
Adaptive-No Hint	0-shot	0.8	5.09	65.59
	3-shot	1.0	6.07	56.85
Hint Only	<b>3-shot</b>	<b>1.0</b>	<b>5.59</b>	<b>51.36</b>

893 Overall, these experiments highlighted the significant impact of integrating memory and  
 894 adaptive strategies into the LLM-driven decision-making process, resulting in more accu-  
 895 rate and effective navigation to the odor source. We visually presented the results of the  
 896 navigation strategies and the robot’s trajectories in Figures 19.

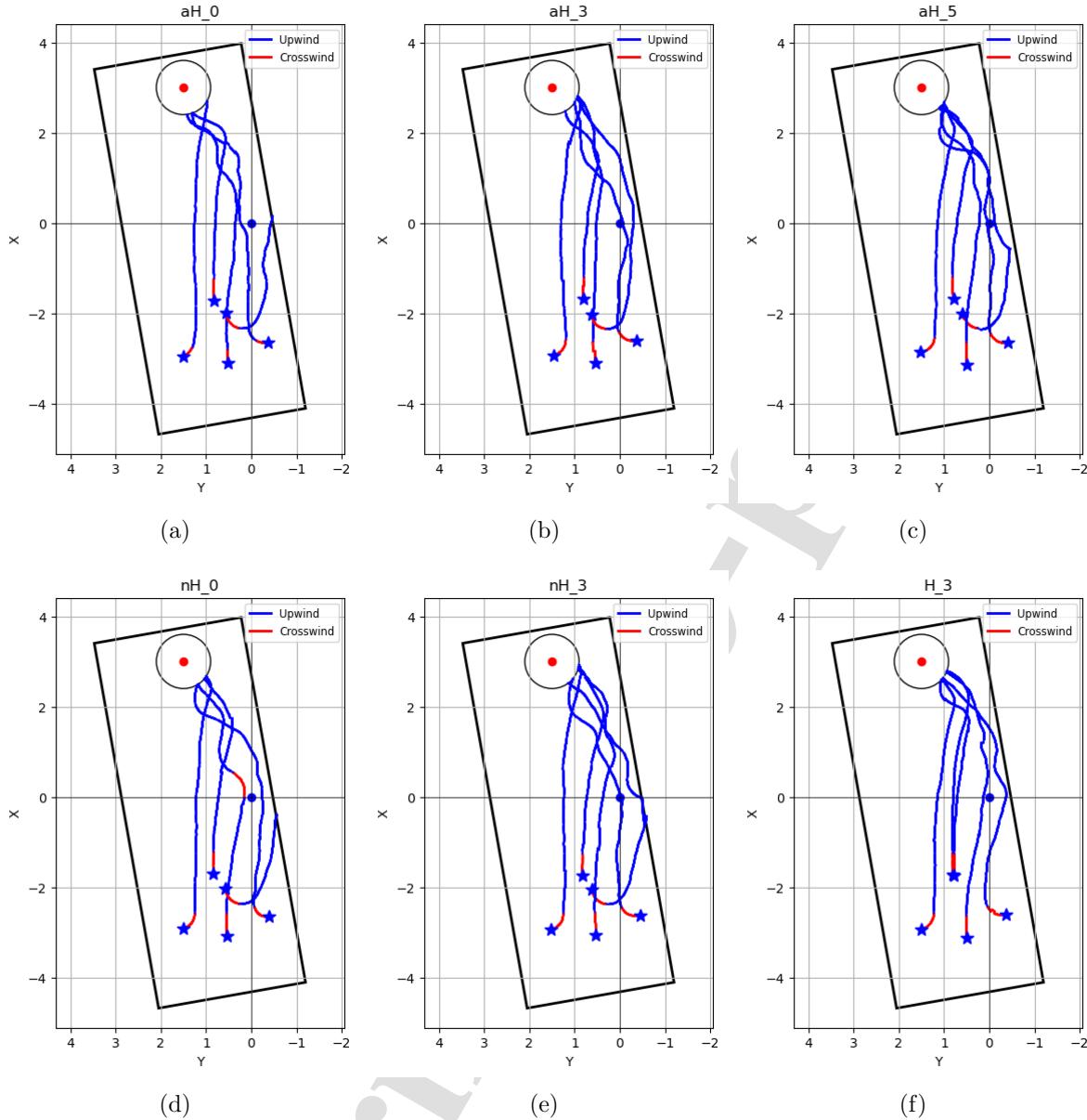


Figure 19: (a), (b) and (c) show the results of LLM within the Adaptive-Hint configuration in 0-shot, 3-shot and 5-shot setting respectively. (d) and (e) show the results of LLM within the Adaptive-No Hint configuration in 0-shot and 3-shot setting respectively. (f) shows the results of LLM within the Only Hint configuration in 3-shot setting.

897 To provide a comparative analysis, we also evaluated the DQN algorithm trained over  
 898 5000 episodes under similar experimental conditions. The DQN demonstrated a lower suc-  
 899 cess rate and less reliable results compared to the Knowledge-driven OSL framework. It  
 900 struggled with longer travel distances and extended search times, underscoring its limita-  
 901 tions in dynamic environments. We presented these results in Figure 20, which illustrates

902 the trajectories of the robotic agent.

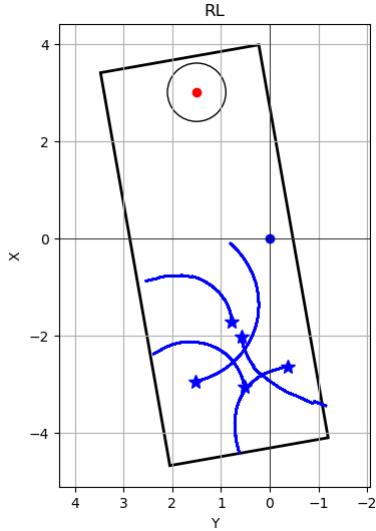


Figure 20: Shows the results of DQN trained with 5000 episodes.

903 The experimental results demonstrate the superior performance of the Knowledge-driven  
 904 OSL framework, particularly within the 3-shot setting of the Adaptive-Hint configuration.  
 905 This configuration not only achieved a perfect success rate but also delivered optimal metrics  
 906 for travel distance and search time. In contrast, the DQN approach lacked consistency and  
 907 adaptability for robust real-world odor localization tasks.

908 We measured the inference time of the LLM during navigation and observed that the  
 909 minimum inference time is 3.59 seconds. While this introduces a delay compared to tra-  
 910 ditional reinforcement learning or sensor-based navigation methods, it did not significantly  
 911 affect navigation accuracy in our experiments. More importantly, our knowledge-driven ap-  
 912 proach demonstrates strong generalization capability, performing effectively even when the  
 913 odor source is flipped during the tests. In contrast, the DQN struggled to generalize across  
 914 different search environments, as illustrated in our experimental results. For instance, Fig-  
 915 ure 14) in subsection 4.6 compares the results of our method with DQN, while Figures 19  
 916 and 20 further emphasize this difference, particularly in real-world adaptation scenarios.

917 The comprehensive analysis underscores the superior adaptability and efficiency of the  
 918 Knowledge-driven OSL framework compared to traditional reinforcement learning approaches  
 919 like DQN. This demonstrates the framework's practical applicability and effectiveness in dy-  
 920 namic real-world odor localization tasks, suggesting promising potential for future enhance-  
 921 ments and applications.

## 922 5. Conclusion and Future Work

923 In this research project, we developed a knowledge-driven framework for robotic odor  
 924 source localization (OSL) by integrating large language models (LLMs) to enhance the

925 robot's navigation capabilities. Our framework leverages the contextual understanding and  
 926 decision-making abilities of LLMs, augmented with a memory module that stores and recalls  
 927 past experiences. Through a series of simulations and real-world experiments, we demon-  
 928 strated the effectiveness of our approach compared to a traditional reinforcement learning  
 929 method using Deep Q-Networks (DQN). The results from both simulated and real-world  
 930 studies indicate that the inclusion of memory, especially with 3-shot experiences, signifi-  
 931 cantly improves the robot's performance in locating the odor source. In real-world tests, the  
 932 adaptive-hint configuration with 0-shot and 3-shot memory settings consistently achieved  
 933 the highest success rates, shortest travel distances, and fastest search times, demonstrating  
 934 the framework's efficiency and robustness across various environments. The adaptive speed  
 935 mechanism further enhances navigation efficiency, leading to shorter travel distances and  
 936 reduced search times in simulations and physical experiments. Additionally, our framework  
 937 exhibited robust generalization capabilities, maintaining high performance even when we al-  
 938 ter the odor source location or change the experimental conditions. Overall, the knowledge-  
 939 driven approach outperformed the DQN-based method in both success rate and efficiency  
 940 and showcased superior adaptability to dynamic environmental conditions. It highlights the  
 941 potential of integrating LLMs and memory modules for complex robotic tasks in varied and  
 942 unpredictable environments.

943 Building on the success of our current framework, we can explore several future research  
 944 directions. First, investigating more advanced memory mechanisms, such as dynamic or  
 945 hierarchical memory networks, could enhance the robot's ability to recall and utilize past  
 946 experiences in more complex scenarios. The memory retrieval process is optimized using  
 947 cosine similarity to identify contextually relevant scenarios. While effective for general se-  
 948 mantic alignment, it does not prioritize decision-critical features such as wind direction,  
 949 odor concentration gradients, or inferred proximity to the odor source. Enhancements like  
 950 weighted embeddings or tagging memories with critical metadata could refine retrieval accu-  
 951 racy and allow the framework to adapt dynamically to varying environmental complexities.  
 952 The framework currently relies on training data from low-variability scenarios, which may  
 953 limit its effectiveness in dynamic or complex environments. Expanding the training data to  
 954 include more diverse and high-variability scenarios, such as through synthetic or augmented  
 955 data, could improve its generalization to unseen conditions.

956 While this study focuses on single-source odor localization to establish a baseline for the  
 957 framework's effectiveness, investigating multiple and distracting odor source localization is  
 958 one of our future research directions. To address the challenges of multi-source environ-  
 959 ments, improvements would be needed in decision-making processes, filtering mechanisms,  
 960 and learning algorithms. Additionally, integrating Kalman filters with LLM pre-filtered in-  
 961 formation offers a promising direction for enhancing real-time tracking and state estimation,  
 962 improving computational efficiency and robustness in dynamic scenarios.

963 Moreover, extending the framework to support multiple collaborative robots could im-  
 964 prove efficiency and accuracy in locating odor sources, especially in more significant or more  
 965 challenging environments. Lastly, additional sensory inputs, such as thermal imaging or  
 966 infrared sensors, could provide a more comprehensive understanding of the environment  
 967 and enhance the robot's decision-making process. Exploring reinforcement learning tech-

968 niques that can learn effectively from sparse rewards could complement our knowledge-  
 969 driven approach, particularly when odor concentrations are infrequent or difficult to detect.  
 970 Finally, enhancing the framework's few-shot learning capabilities by fine-tuning the LLMs  
 971 with domain-specific data could lead to more accurate and contextually relevant decision-  
 972 making.

## 973 References

- 974 [1] G. A. Nevitt, Olfactory foraging by antarctic procellariiform seabirds: Life at high reynolds numbers,  
 Biological Bulletin 198 (2) (2000) 245–253.
- 975 [2] A. D. Hasler, A. T. Scholz, Olfactory Imprinting and Homing in Salmon, New York: Springer-Verlag,  
 1983.
- 976 [3] R. T. Cardé, A. Mafra-Neto, Mechanisms of Flight of Male Moths to Pheromone, Springer US, 1997.
- 977 [4] N. Vickers, Mechanisms of animal navigation in odor plumes, The Biological Bulletin 198 (2) (2000)  
 203–212.
- 978 [5] R. T. Cardé, Odour Plumes and Odour-Mediated Flight in Insects, John Wiley & Sons, Ltd, 1996.
- 979 [6] L. Marques, H. Magalhães, R. Baptista, J. Macedo, Mobile robot olfaction state-of-the-art and research  
 challenges, Institution of Engineering and Technology eBooks, 2022. doi:10.1049/pbce097e\_ch9.
- 980 [7] G. Kowadlo, R. A. Russell, Robot odor localization: A taxonomy and survey, The International Journal  
 of Robotics Research 27 (8) (2008) 869–894. doi:10.1177/0278364908095118.
- 981 [8] M. Dunbabin, L. Marques, Robots for environmental monitoring: Significant advancements and appli-  
 982 cations, IEEE Robotics & Automation Magazine 19 (1) (2012) 24–39.
- 983 [9] L. Wang, S. Pang, M. Noyela, K. Adkins, L. Sun, M. El-Sayed, Vision and olfactory-based wildfire  
 984 monitoring with uncrewed aircraft systems, in: 2023 20th International Conference on Ubiquitous  
 985 Robots (UR), IEEE, 2023, pp. 716–723.
- 986 [10] S. Soldan, G. Bonow, A. Kroll, Robogasinspector-a mobile robotic system for remote leak sensing and  
 987 localization in large industrial environments: Overview and first results, IFAC Proceedings Volumes  
 988 45 (8) (2012) 33–38.
- 989 [11] G. Ferri, M. V. Jakuba, D. R. Yoerger, A novel method for hydrothermal vents prospecting using an  
 990 autonomous underwater robot, in: 2008 IEEE International Conference on Robotics and Automation,  
 991 IEEE, 2008, pp. 1055–1060.
- 992 [12] X. Chen, J. Huang, Odor source localization algorithms on mobile robots: A review and future outlook,  
 993 Robotics and Autonomous Systems 112 (2019) 123–136.
- 994 [13] J. H. Belanger, M. A. Willis, Adaptive control of odor-guided locomotion: Behavioral flexibility as an  
 995 antidote to environmental unpredictability1, Adaptive Behavior 4 (3-4) (1996) 217–253.
- 996 [14] J. H. Belanger, E. A. Arbas, Behavioral strategies underlying pheromone-modulated flight in moths:  
 997 lessons from simulation studies, Journal of Comparative Physiology A 183 (1998) 345–360.
- 998 [15] F. W. Grasso, T. R. Consi, D. C. Mountain, J. Atema, Biomimetic robot lobster performs chemo-  
 999 orientation in turbulence using a pair of spatially separated sensors: Progress and challenges, Robotics  
 1000 and Autonomous Systems 30 (1) (2000) 115–131.
- 1001 [16] F. W. Grasso, Invertebrate-inspired sensory-motor systems and autonomous, olfactory-guided explo-  
 1002 ration, The Biological Bulletin 200 (2) (2001) 160–168.
- 1003 [17] H. Ishida, Y. Kagawa, T. Nakamoto, T. Moriizumi, Odor-source localization in the clean room by an  
 1004 autonomous mobile sensing system, Sensors and Actuators B: Chemical 33 (1) (1996) 115–121.
- 1005 [18] H. Ishida, T. Nakamoto, T. Moriizumi, T. Kikas, J. Janata, Plume-tracking robots: A new application  
 1006 of chemical sensors, The Biological Bulletin 200 (2) (2001) 222–226.
- 1007 [19] Y. Kuwana, S. Nagasawa, I. Shimoyama, R. Kanzaki, Synthesis of the pheromone-oriented behaviour of  
 1008 silkworm moths by a mobile robot with moth antennae as pheromone sensors1this paper was presented  
 1009 at the fifth world congress on biosensors, berlin, germany, 3–5 june 1998.1, Biosensors and Bioelectronics  
 1010 14 (2) (1999) 195–202.

- 1016 [20] S. Pang, J. A. Farrell, Chemical plume source localization, IEEE Transactions on Systems, Man, and  
1017 Cybernetics, Part B (Cybernetics) 36 (5) (2006) 1068–1080.
- 1018 [21] L. Wang, S. Pang, J. Li, Olfactory-based navigation via model-based reinforcement learning and fuzzy  
1019 inference methods, IEEE Transactions on Fuzzy Systems 29 (10) (2021) 3014–3027. doi:10.1109/  
1020 TFUZZ.2020.3011741.
- 1021 [22] W. Li, J. A. Farrell, S. Pang, R. M. Arrieta, Moth-inspired chemical plume tracing on an autonomous  
1022 underwater vehicle, IEEE Transactions on Robotics 22 (2) (2006) 292–307.
- 1023 [23] S. Shigaki, S. Haigo, C. H. Reyes, T. Sakurai, R. Kanzaki, D. Kurabayashi, H. Sezutsu, Analysis of  
1024 the role of wind information for efficient chemical plume tracing based on optogenetic silkworm moth  
1025 behavior, Bioinspiration & biomimetics 14 (4) (2019) 046006.
- 1026 [24] A. M. Matheson, A. J. Lanz, A. M. Medina, A. M. Licata, T. A. Currier, M. H. Syed, K. I. Nagel, A  
1027 neural circuit for wind-guided olfactory navigation, Nature Communications 13 (1) (2022) 4613.
- 1028 [25] R. T. Cardé, A. Mafra-Neto, Mechanisms of Flight of Male Moths to Pheromone, Springer, Boston,  
1029 MA, 1996. doi:10.1007/978-1-4615-6371-6\_25.
- 1030 [26] J. Li, Q. Meng, Y. Wang, M. Zeng, Odor source localization using a mobile robot in outdoor airflow  
1031 environments with a particle filter algorithm, Autonomous Robots 30 (3) (2011) 281–292.
- 1032 [27] L. Wang, S. Pang, J. long Li, Olfactory-based navigation via model-based reinforcement learning and  
1033 fuzzy inference methods, IEEE Transactions on Fuzzy Systems 29 (10) (2021) 3014–3027. doi:10.  
1034 1109/TFUZZ.2020.3011741.
- 1035 [28] S. Hassan, L. Wang, K. R. Mahmud, Robotic odor source localization via vision and olfaction fusion  
1036 navigation algorithm, Sensors 24 (7) (2024). doi:10.3390/s24072309.
- 1037 [29] Y. LeCun, A path towards autonomous machine intelligence, Open Review 0.9. 2, 2022-06-27 (62)  
1038 (2022).
- 1039 [30] D. Driess, F. Xia, M. S. M. Sajjadi, C. Lynch, A. Chowdhery, B. Ichter, A. Wahid, J. Tompson,  
1040 Q. Vuong, T. Yu, W. Huang, Y. Chebotar, P. Sermanet, D. Duckworth, S. Levine, V. Vanhoucke,  
1041 K. Hausman, M. Toussaint, K. Greff, A. Zeng, I. Mordatch, P. Florence, Palm-e: An embodied multi-  
1042 modal language model (2023). arXiv:2303.03378.
- 1043 [31] S. Huang, Z. Jiang, H. Dong, Y. Qiao, P. Gao, H. Li, Instruct2act: Mapping multi-modality instructions  
1044 to robotic actions with large language model (2023). arXiv:2305.11176.
- 1045 [32] L. Wen, D. Fu, X. Li, X. Cai, T. Ma, P. Cai, M. Dou, B. Shi, L. He, Y. Qiao, Dilu: A knowledge-driven  
1046 approach to autonomous driving with large language models, arXiv preprint arXiv:2309.16292 (2023).
- 1047 [33] W. Huang, C. Wang, R. Zhang, Y. Li, J. Wu, L. Fei-Fei, Voxposer: Composable 3d value maps for  
1048 robotic manipulation with language models (2023). arXiv:2307.05973.
- 1049 [34] G. Wang, Y. Xie, Y. Jiang, A. Mandlekar, C. Xiao, Y. Zhu, L. Fan, A. Anandkumar, Voyager: An  
1050 open-ended embodied agent with large language models (2023). arXiv:2305.16291.
- 1051 [35] P. Gao, J. Han, R. Zhang, Z. Lin, S. Geng, A. Zhou, W. Zhang, P. Lu, C. He, X. Yue, H. Li, Y. Qiao,  
1052 Llama-adapter v2: Parameter-efficient visual instruction model (2023). arXiv:2304.15010.
- 1053 [36] X. Zhu, Y. Chen, H. Tian, C. Tao, W. Su, C. Yang, G. Huang, B. Li, L. Lu, X. Wang, Y. Qiao,  
1054 Z. Zhang, J. Dai, Ghost in the minecraft: Generally capable agents for open-world environments via  
1055 large language models with text-based knowledge and memory (2023). arXiv:2305.17144.
- 1056 [37] V. Mnih, K. Kavukcuoglu, D. Silver, A. A. Rusu, J. Veness, M. G. Bellemare, A. Graves, M. A.  
1057 Riedmiller, A. K. Fidjeland, G. Ostrovski, S. Petersen, C. Beattie, A. Sadik, I. Antonoglou, H. King,  
1058 D. Kumaran, D. Wierstra, S. Legg, D. Hassabis, Human-level control through deep reinforcement  
1059 learning, Nature 518 (2015) 529–533.
- 1060 [38] Adaptive control system of an insect brain during odor source localization, IEEE, 2013. doi:10.1109/  
1061 IROS.2013.6696376.
- 1062 [39] 3D Odor Source Localization using a Micro Aerial Vehicle: System Design and Performance Evaluation,  
1063 IEEE, 2020. doi:10.1109/IROS45743.2020.9341501.
- 1064 [40] G. Lu, Q. Zhang, T. Qie, Q. Feng, A robot odor source localization strategy based on bionic behavior,  
1065 IOP Conference Series: Materials Science and Engineering 470 (1) (2019) 012033. doi:10.1088/  
1066 1757-899X/470/1/012033.

- 1067 [41] S. Shigaki, M. Yamada, D. Kurabayashi, K. Hosoda, Robust moth-inspired algorithm for odor source  
1068 localization using multimodal information, Sensors 23 (3) (2023) 1475–1475. doi:10.3390/s23031475.
- 1069 [42] A multi-sensory robot for testing biologically-inspired odor plume tracking strategies. doi:10.1109/  
1070 AIM.2005.1511219.
- 1071 [43] A 3-D bio-inspired odor source localization and its validation in realistic environmental conditions,  
1072 IEEE, 2017. doi:10.1109/IROS.2017.8206252.
- 1073 [44] J.-G. Li, Q.-H. Meng, Y. Wang, M. Zeng, Odor source localization using a mobile robot in out-  
1074 door airflow environments with a particle filter algorithm, Autonomous Robots (2011). doi:10.1007/  
1075 S10514-011-9219-2.
- 1076 [45] M. J. Anderson, J. Sullivan, T. K. Horiuchi, S. B. Fuller, T. L. Daniel, A bio-hybrid odor-guided  
1077 autonomous palm-sized air vehicle., Bioinspiration and Biomimetics (2020). doi:10.1088/1748-3190/  
1078 ABBB81.
- 1079 [46] D. W. D. Muhamad Rausyan Fikri, Palm-sized quadrotor source localization using modified bio-inspired  
1080 algorithm in obstacle region, International Journal of Electrical and Computer Engineering 12 (4) (2022)  
1081 3494–3494. doi:10.11591/ijece.v12i4.pp3494-3504.
- 1082 [47] Biologically-inspired search algorithms for locating unseen odor sources. doi:10.1109/ISIC.1998.  
1083 713672.
- 1084 [48] W. Jatmiko, F. Jovan, R. Y. S. Dhiemas, A. M. Sakti, F. M. Ivan, A. Febrian, T. Fukuda, K. Sekiyama,  
1085 Robots implementation for odor source localization using pso algorithm, WSEAS Transactions on  
1086 Circuits and Systems archive (2011). doi:10.5555/2001161.2001162.
- 1087 [49] K. Gaurav, A. Kumar, R. Singh, Single and multiple odor source localization using hybrid nature-  
1088 inspired algorithm, Sadhana-academy Proceedings in Engineering Sciences 45 (1) (2020) 1–19. doi:  
1089 10.1007/S12046-020-1318-3.
- 1090 [50] Modified particle swarm optimization for odor source localization of multi-robot. doi:10.1109/CEC.  
1091 2011.5949609.
- 1092 [51] Swarm robotic odor source localization using ant colony algorithm, IEEE, 2009. doi:10.1109/ICCA.  
1093 2009.5410516.
- 1094 [52] S. Wang, S. Sun, M. Liu, B. Gao, Y. Wang, Resource-aware probability-based collaborative odor source  
1095 localization using multiple uavs (2023). doi:10.48550/arxiv.2303.03830.
- 1096 [53] M. Staples, C. H. Hugenholz, T. E. Barchyn, M. Gao, A comparison of multiple odor source localization  
1097 algorithms, Sensors 23 (10) (2023) 4799–4799. doi:10.3390/s23104799.
- 1098 [54] L. D. Nhat, D. Kurabayashi, Odor source localization in obstacle regions using switching planning  
1099 algorithms with a switching framework, Sensors 23 (3) (2023) 1140–1140. doi:10.3390/s23031140.
- 1100 [55] Multi-sensory Olfactory Quadruped Robot for Odor Source Localization\*. doi:10.1109/CBS55922.  
1101 2023.10115389.
- 1102 [56] Learn to Trace Odors: Robotic Odor Source Localization via Deep Learning Methods with Real-world  
1103 Experiments. doi:10.1109/SoutheastCon51012.2023.10115175.
- 1104 [57] Z. Yan, Q.-H. Meng, T. Jing, S.-W. Chen, H.-R. Hou, A deep learning-based indoor odor compass, IEEE  
1105 Transactions on Instrumentation and Measurement 72 (2023) 1–10. doi:10.1109/TIM.2023.3238053.
- 1106 [58] A Novel Odor Source Localization Method via a Deep Neural Network-Based Odor Compass, Springer  
1107 International Publishing, 2023. doi:10.1007/978-3-031-21062-4\_16.
- 1108 [59] X. Chen, B. Yang, J. Huang, Y. Leng, C. Fu, A reinforcement learning fuzzy system for continuous con-  
1109 trol in robotic odor plume tracking, Robotica 41 (2022) 1039–1054. doi:10.1017/S0263574722001321.
- 1110 [60] P. Ojeda, J. Monroy, J. Gonzalez-Jimenez, Robotic gas source localization with probabilistic mapping  
1111 and online dispersion simulation (2023). doi:10.48550/arxiv.2304.08879.
- 1112 [61] D. Badawi, I. Bassi, S. Ozev, A. E. Cetin, Deep-learning-based gas leak source localization from sparse  
1113 sensor data, IEEE Sensors Journal 22 (21) (2022) 20999–21008. doi:10.1109/JSEN.2022.3202134.
- 1114 [62] A. S. A. Yeon, A. Zakaria, S. M. M. S. Zakaria, R. Visvanathan, K. Kamarudin, L. M. Kamarudin, Gas  
1115 source localization via mobile robot with gas distribution mapping and deep neural network, in: 2022  
1116 2nd International Conference on Electronic and Electrical Engineering and Intelligent System (ICE3IS),  
1117 2022, pp. 120–124. doi:10.1109/ICE3IS56585.2022.10010251.

- 1118 [63] Y. Shan, H. Lu, W. T. Lou, Multi-faceted deep learning framework for dynamics modeling and robot  
 1119 localization learning, Journal of Intelligent and Fuzzy Systems 45 (2023). doi:10.3233/jifs-230895.
- 1120 [64] T. Brown, B. Mann, N. Ryder, M. Subbiah, J. D. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam,  
 1121 G. Sastry, A. Askell, S. Agarwal, A. Herbert-Voss, G. Krueger, T. Henighan, R. Child, A. Ramesh,  
 1122 D. Ziegler, J. Wu, C. Winter, C. Hesse, M. Chen, E. Sigler, M. Litwin, S. Gray, B. Chess, J. Clark,  
 1123 C. Berner, S. McCandlish, A. Radford, I. Sutskever, D. Amodei, Language models are few-shot learners,  
 1124 in: H. Larochelle, M. Ranzato, R. Hadsell, M. Balcan, H. Lin (Eds.), Advances in Neural Information  
 1125 Processing Systems, Vol. 33, Curran Associates, Inc., 2020, pp. 1877–1901.
- 1126 [65] A. Chowdhery, S. Narang, J. Devlin, M. Bosma, G. Mishra, A. Roberts, P. Barham, H. W. Chung,  
 1127 C. Sutton, S. Gehrmann, P. Schuh, K. Shi, S. Tsvyashchenko, J. Maynez, A. Rao, P. Barnes, Y. Tay,  
 1128 N. Shazeer, V. Prabhakaran, E. Reif, N. Du, B. Hutchinson, R. Pope, J. Bradbury, J. Austin, M. Isard,  
 1129 G. Gur-Ari, P. Yin, T. Duke, A. Levskaya, S. Ghemawat, S. Dev, H. Michalewski, X. Garcia, V. Misra,  
 1130 K. Robinson, L. Fedus, D. Zhou, D. Ippolito, D. Luan, H. Lim, B. Zoph, A. Spiridonov, R. Sepassi,  
 1131 D. Dohan, S. Agrawal, M. Omernick, A. M. Dai, T. S. Pillai, M. Pellat, A. Lewkowycz, E. Moreira,  
 1132 R. Child, O. Polozov, K. Lee, Z. Zhou, X. Wang, B. Saeta, M. Diaz, O. Firat, M. Catasta, J. Wei,  
 1133 K. Meier-Hellstern, D. Eck, J. Dean, S. Petrov, N. Fiedel, Palm: Scaling language modeling with  
 1134 pathways (2022). arXiv:2204.02311.
- 1135 [66] H. Touvron, T. Lavril, G. Izacard, X. Martinet, M.-A. Lachaux, T. Lacroix, B. Rozière, N. Goyal,  
 1136 E. Hambro, F. Azhar, A. Rodriguez, A. Joulin, E. Grave, G. Lample, Llama: Open and efficient  
 1137 foundation language models (2023). arXiv:2302.13971.
- 1138 [67] OpenAI, J. Achiam, S. Adler, S. Agarwal, L. Ahmad, I. Akkaya, F. L. Aleman, D. Almeida, J. Al-  
 1139 tenschmidt, S. Altman, S. Anadkat, R. Avila, I. Babuschkin, S. Balaji, V. Balcom, P. Baltescu, H. Bao,  
 1140 other authors, Gpt-4 technical report (2024). arXiv:2303.08774.
- 1141 [68] L. Ouyang, J. Wu, X. Jiang, D. Almeida, C. Wainwright, P. Mishkin, C. Zhang, S. Agarwal, K. Slama,  
 1142 A. Ray, J. Schulman, J. Hilton, F. Kelton, L. Miller, M. Simens, A. Askell, P. Welinder, P. F. Chris-  
 1143 tiano, J. Leike, R. Lowe, Training language models to follow instructions with human feedback, in:  
 1144 S. Koyejo, S. Mohamed, A. Agarwal, D. Belgrave, K. Cho, A. Oh (Eds.), Advances in Neural Informa-  
 1145 tion Processing Systems, Vol. 35, Curran Associates, Inc., 2022, pp. 27730–27744.
- 1146 [69] J. Wei, M. Bosma, V. Y. Zhao, K. Guu, A. W. Yu, B. Lester, N. Du, A. M. Dai, Q. V. Le, Finetuned  
 1147 language models are zero-shot learners (2022). arXiv:2109.01652.
- 1148 [70] J. Wei, X. Wang, D. Schuurmans, M. Bosma, B. Ichter, F. Xia, E. Chi, Q. Le, D. Zhou, Chain-of-thought  
 1149 prompting elicits reasoning in large language models (2023). arXiv:2201.11903.
- 1150 [71] A. Caines, L. Benedetto, S. Taslimipoor, C. Davis, Y. Gao, O. Andersen, Z. Yuan, M. Elliott, R. Moore,  
 1151 C. Bryant, M. Rei, H. Yannakoudakis, A. Mullooly, D. Nicholls, P. Butterly, On the application of large  
 1152 language models for language teaching and assessment technology (2023). arXiv:2307.08393.
- 1153 [72] J. Göpfert, J. M. Weinand, P. Kuckertz, D. Stolten, Opportunities for large language models and  
 1154 discourse in engineering design, arXiv.org abs/2306.09169 (2023). doi:10.48550/arXiv.2306.09169.
- 1155 [73] W. X. Zhao, K. Zhou, J. Li, T. Tang, X. Wang, Y. Hou, Y. Min, B. Zhang, J. Zhang, Z. Dong, Y. Du,  
 1156 C. Yang, Y. Chen, Z. Chen, J. Jiang, R. Ren, Y. Li, X. Tang, Z. Liu, P. Liu, J.-Y. Nie, J.-R. Wen, A  
 1157 survey of large language models (2023). arXiv:2303.18223.
- 1158 [74] Openai. introducing chatgpt. (2023).  
 1159 URL <https://openai.com/blog/chatgpt/>
- 1160 [75] A. Goyal, J. Xu, Y. Guo, V. Blukis, Y.-W. Chao, D. Fox, Rvt: Robotic view transformer for 3d object  
 1161 manipulation (2023). arXiv:2306.14896.
- 1162 [76] A. Brohan, N. Brown, J. Carbajal, Y. Chebotar, J. Dabis, C. Finn, K. Gopalakrishnan, K. Hausman,  
 1163 A. Herzog, J. Hsu, J. Ibarz, B. Ichter, A. Irpan, T. Jackson, S. Jesmonth, N. J. Joshi, R. Julian,  
 1164 D. Kalashnikov, Y. Kuang, I. Leal, K.-H. Lee, S. Levine, Y. Lu, U. Malla, D. Manjunath, I. Mordatch,  
 1165 O. Nachum, C. Parada, J. Peralta, E. Perez, K. Pertsch, J. Quiambao, K. Rao, M. Ryoo, G. Salazar,  
 1166 P. Sanketi, K. Sayed, J. Singh, S. Sontakke, A. Stone, C. Tan, H. Tran, V. Vanhoucke, S. Vega,  
 1167 Q. Vuong, F. Xia, T. Xiao, P. Xu, S. Xu, T. Yu, B. Zitkovich, Rt-1: Robotics transformer for real-  
 1168 world control at scale (2023). arXiv:2212.06817.

- 1169 [77] Z. Zhao, H. Yu, H. Wu, X. Zhang, 6-dof robotic grasping with transformer (2023). [arXiv:2301.12476](https://arxiv.org/abs/2301.12476).
- 1170 [78] T. Gervet, Z. Xian, N. Gkanatsios, K. Fragkiadaki, Act3d: 3d feature field transformers for multi-task  
1171 robotic manipulation (2023). [arXiv:2306.17817](https://arxiv.org/abs/2306.17817).
- 1172 [79] D. Seamon, Improving knowledge extraction from llms for robotic task learning through agent analysis  
1173 (2023). doi:[10.48550/arxiv.2306.06770](https://doi.org/10.48550/arxiv.2306.06770).
- 1174 [80] B. Hu, C. Zhao, P. Zhang, Z. Zhou, Y. Yang, Z. Xu, B. Liu, Enabling intelligent interactions between  
1175 an agent and an llm: A reinforcement learning approach, arXiv.org abs/2306.03604 (2023). doi:  
1176 [10.48550/arXiv.2306.03604](https://doi.org/10.48550/arXiv.2306.03604).
- 1177 [81] V. S. Dorbala, J. F. Mullen, D. Manocha, Can an embodied agent find your “cat-shaped mug”? llm-  
1178 based zero-shot object navigation, IEEE Robotics and Automation Letters 9 (2023) 4083–4090.
- 1179 [82] R. Schumann, W. Zhu, W. Feng, T.-J. Fu, S. Riezler, W. Y. Wang, Velma: Verbalization embodiment  
1180 of llm agents for vision and language navigation in street view, in: AAAI Conference on Artificial  
1181 Intelligence, 2023.
- 1182 [83] J. Mai, J. Chen, B. Li, G. Qian, M. Elhoseiny, B. Ghanem, Llm as a robotic brain: Unifying egocentric  
1183 memory and control (2023). [arXiv:2304.09349](https://arxiv.org/abs/2304.09349).
- 1184 [84] H. Sun, Y. Zhuang, L. Kong, B. Dai, C. Zhang, Adaplanner: Adaptive planning from feedback with  
1185 language models (2023). [arXiv:2305.16653](https://arxiv.org/abs/2305.16653).
- 1186 [85] S. Yu, M. Cheng, J. Yang, J. Ouyang, Y. Luo, C. Lei, Q. Liu, E. Chen, Multi-source knowledge pruning  
1187 for retrieval-augmented generation: A benchmark and empirical study (2024). [arXiv:2409.13694](https://arxiv.org/abs/2409.13694).
- 1188 [86] X. Li, Application of rag model based on retrieval enhanced generation technique in complex query  
1189 processing, Advances in computer, signals and systems 8 (6) (2023).
- 1190 [87] C. Yao, S. Fujita, Adaptive control of retrieval-augmented generation for large language models through  
1191 reflective tags, Electronics 13 (23) (2024).
- 1192 [88] S. E. Spatharioti, D. Rothschild, D. G. Goldstein, J. M. Hofman, Comparing traditional and llm-  
1193 based search for consumer choice: A randomized experiment, arXiv.org abs/2307.03744 (2023). doi:  
1194 [10.48550/arXiv.2307.03744](https://doi.org/10.48550/arXiv.2307.03744).
- 1195 [89] Z. Zhang, D. Seth, S. Artham, J. G. Leishman, E. P. Gnanamanickam, Time-resolved flowfield mea-  
1196 surements of momentum-driven pulsed transient jets, AIAA Journal 56 (4) (2017) 1434–1446.
- 1197 [90] N. Ando, R. Kanzaki, A simple behaviour provides accuracy and flexibility in odour plume tracking—the  
1198 robotic control of sensory-motor coupling in silkmoths., The Journal of Experimental Biology 218 (23)  
1199 (2015) 3845–3854.
- 1200 [91] W. Huang, P. Abbeel, D. Pathak, I. Mordatch, Language models as zero-shot planners: Extracting  
1201 actionable knowledge for embodied agents (2022). [arXiv:2201.07207](https://arxiv.org/abs/2201.07207).
- 1202 [92] J. Farrell, S. Pang, W. Li, Chemical plume tracing via an autonomous underwater vehicle, IEEE Journal  
1203 of Oceanic Engineering 30 (2) (2005) 428–442.
- 1204 [93] Q. Feng, H. Cai, Z. Chen, Y. Yang, J. Lu, F. Li, J. Xu, X. Li, Experimental study on a comprehensive  
1205 particle swarm optimization method for locating contaminant sources in dynamic indoor environments  
1206 with mechanical ventilation, Energy and Buildings 196 (2019) 145–156.

1207 **Appendix A. Prompts Generation in Knowledge-Driven OSL Framework**

1208 In this section, we detail the specific design of prompts used in the reasoning module.  
 1209 As mentioned in the article, the prompts for the reasoning module primarily consist of three  
 1210 parts: system prompts, scenario description, and few-shot experience. Each part plays a  
 1211 crucial role in guiding the LLM to make informed decisions for the robotic odor source  
 1212 localization (OSL) task.

1213 *Appendix A.1. Reasoning Prompts Generation*

1214 The system prompts section is entirely fixed and includes the foundational information  
 1215 necessary for the LLM to understand the task at hand. This section provides an introduction  
 1216 to the odor source localization task, detailed instructions for input and output formats, and  
 1217 specific formatting requirements for LLM responses to ensure consistency and clarity. Figure  
 1218 A.21 illustrates the prompt template used in our OSL experiment, highlighting the fixed  
 1219 nature of the system prompts and the dynamically generated scenario descriptions.

1220 The scenario description, while mostly fixed, contains dynamically generated parts based  
 1221 on the current decision frame. This section provides information about the robot's current  
 1222 position within the search area, details about wind direction and speed, odor concentrations  
 1223 detected by the robot's sensors, and environmental conditions that may affect navigation.  
 1224 These dynamic elements are crucial as they provide the real-time context necessary for the  
 1225 LLM to generate appropriate navigation strategies. The scenario description is embedded  
 1226 into vectors and used as query inputs to the memory module to retrieve relevant experiences.  
 1227 Available actions include moving against the wind direction or moving across the wind  
 1228 direction. The default navigation intention is: Your navigation intention is to locate the  
 1229 odor source efficiently.

1230 As for the few-shot experience, it is entirely obtained from the memory module. Each  
 1231 experience consists of a human-LLM dialogue pair, where the human question includes the  
 1232 scenario description at that decision frame, and the LLM response represents the correct  
 1233 reasoning and decision made by the robotic agent. The extracted experiences are directly  
 1234 utilized with a few-shot prompting technique to input into the LLM, enabling in-context  
 1235 learning. This approach allows the reasoning module to adapt its decision-making process  
 1236 based on previous experiences stored in the memory module, ensuring continuous improve-  
 1237 ment and adaptability in locating the odor source.

1238 The few-shot experience component is derived entirely from the memory module and con-  
 1239 sists of a human-LLM dialogue pair. The human question includes the scenario description  
 1240 at that decision frame, and the LLM response represents the reasoning and decision made  
 1241 by the robotic agent. These experiences are used with a few-shot prompting technique to  
 1242 input into the LLM, enabling in-context learning. This process allows the reasoning module  
 1243 to adapt its decision-making based on previously encountered scenarios, ensuring continu-  
 1244 ous learning and improvement. Figure A.22 demonstrates the results of a 3-shot experience  
 1245 query, which includes three "move against the wind direction" decisions.

1246 To illustrate, let's consider an example scenario where the robotic agent is tasked with  
 1247 navigating towards an odor source. The prompt generation process would include system

1248 prompts, such as a task description and instructions, a dynamically generated scenario  
 1249 description, and relevant few-shot experiences. The combination of these elements enables  
 1250 the LLM to generate well-informed decisions for the current scenario.

1251 By leveraging LLMs and memory for contextual understanding and decision-making,  
 1252 our framework provides a significant advancement over traditional OSL methods, offering  
 1253 improved adaptability and efficiency in dynamic environments. Figures A.21 and A.22  
 1254 illustrate the detailed structure and examples of the prompts used in the reasoning module,  
 1255 showcasing how they guide the LLM in generating effective navigation strategies.

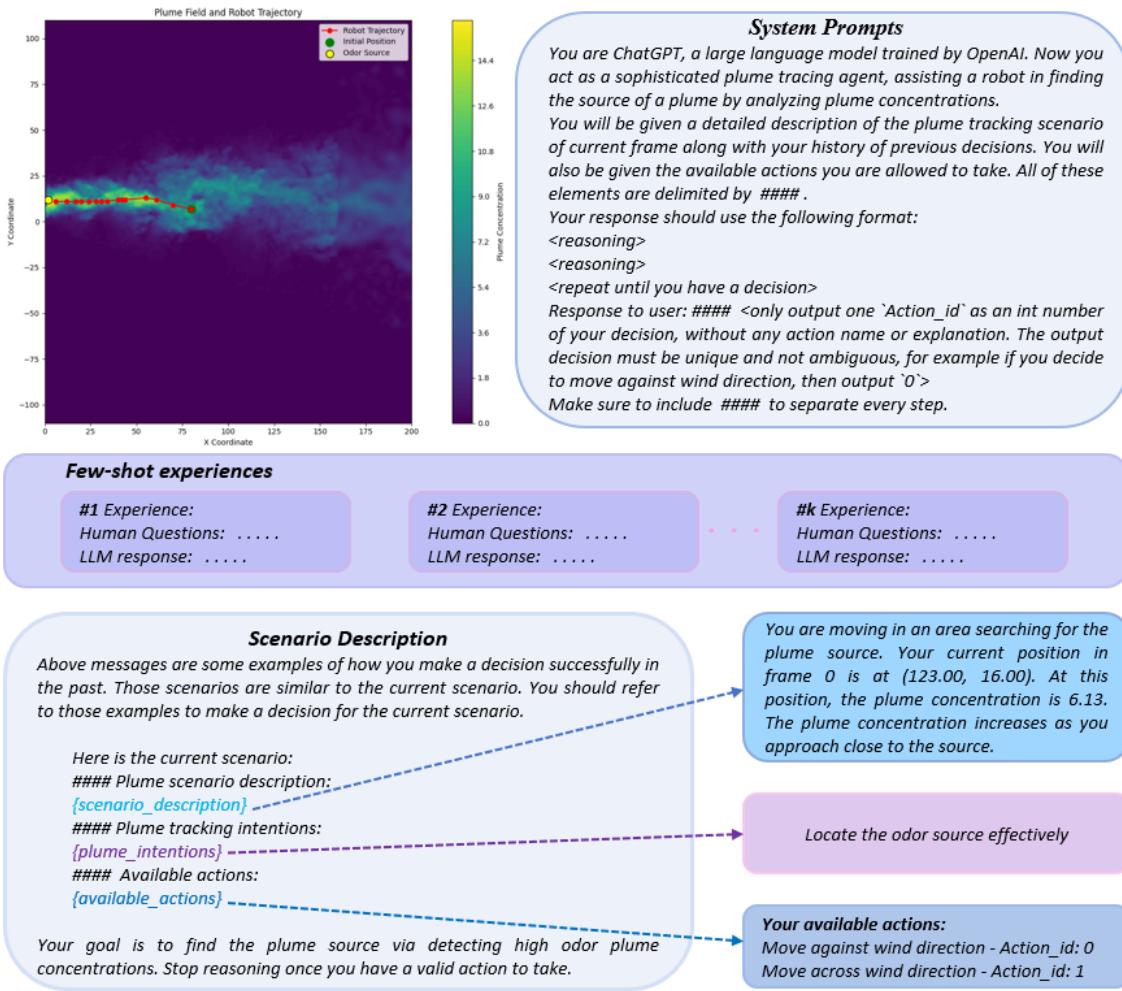


Figure A.21: The prompt template for the reasoning module in our OSL experiment. The prompts: System Prompts box and Scenario Description box are fixed, while the prompts in the colored boxes vary depending on the current scenario.

**Human\_question:**

Here is the current scenario:  
**#### Plume scenario description:**  
 You are moving in an area searching for the plume source. Your current position in frame 9 is at (48.00, 3.00). At this position, the plume concentration is 6.64. The plume concentration increases as you approach close to the source.  
**#### Plume tracking intentions:**  
 Locate the odor source effectively  
**#### Available actions:**  
 Your available actions are:  
 Move against wind direction - target heading direction is the current wind direction plus 180 degrees - Action\_id: 0  
 Move across wind direction - target heading direction is the current wind direction plus 90 degrees - Action\_id: 1  
 Your goal is to find the plume source via detecting high odor plume concentrations. Stop reasoning once you have a valid action to take.

**LLM\_response:**

The plume concentration at my current position is 6.64, which indicates that I am closer to the source since the concentration increases as I approach the source.  
 - Since the concentration is high, it is likely that moving against the wind direction will bring me closer to the source of the plume. This is because the plume is carried by the wind, and the source should be located upwind.  
 Response to user:#### 0  
 action: 0

(a)

**Human\_question:**

Here is the current scenario:  
**#### Plume scenario description:**  
 You are moving in an area searching for the plume source. Your current position in frame 9 is at (93.00, -20.00). At this position, the plume concentration is 0.13. The plume concentration increases as you approach close to the source.  
**#### Plume tracking intentions:**  
 Locate the odor source effectively  
**#### Available actions:**  
 Your available actions are:  
 Move against wind direction - target heading direction is the current wind direction plus 180 degrees - Action\_id: 0  
 Move across wind direction - target heading direction is the current wind direction plus 90 degrees - Action\_id: 1  
 Your goal is to find the plume source via detecting high odor plume concentrations. Stop reasoning once you have a valid action to take.

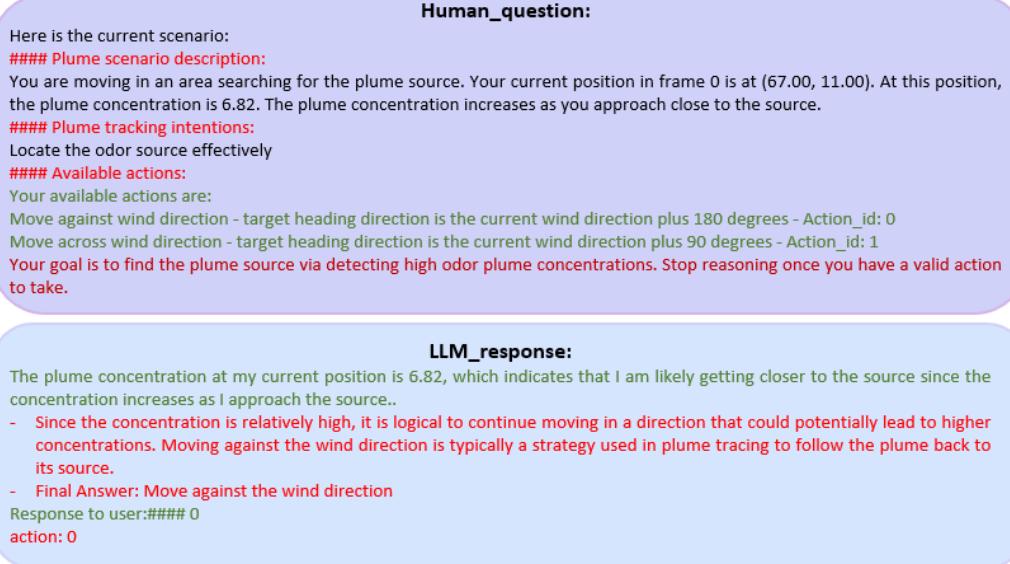
**LLM\_response:**

The plume concentration at my current position is 0.13, which indicates that I am in the presence of the plume since the concentration is above a threshold that might be considered as detecting the plume.  
 - Given that the plume concentration increases as I approach the source, my goal is to move in a direction that will likely increase the concentration further, indicating that I am getting closer to the source.  
 - Since I am currently detecting the plume, the best course of action is to move against the wind direction. This is because in a typical plume tracking scenario, the plume will disperse downwind from the source. By moving against the wind direction, I am likely moving towards the source of the plume.  
 Response to user:#### 0  
 action: 0

(b)

#1 Experience

#2 Experience



(c)

Figure A.22: The 3-shot prompts example for the scenario shown.

1256 *Appendix A.2. Case Study of Reasoning Module*

1257 First, we present the results of the reasoning module for two cases, as shown in Figure  
 1258 A.23. In Case Study I (Figure 23(a)), the robotic agent decides to move across the wind  
 1259 direction based on the current environmental data and stored experiences. The reasoning  
 1260 module processes the sensory inputs, retrieves relevant past experiences, and generates a  
 1261 prompt that guides the robot to move across the wind direction. This decision is based  
 1262 on the logic that moving across the wind direction will likely lead to encountering a higher  
 1263 concentration of the odor plume.

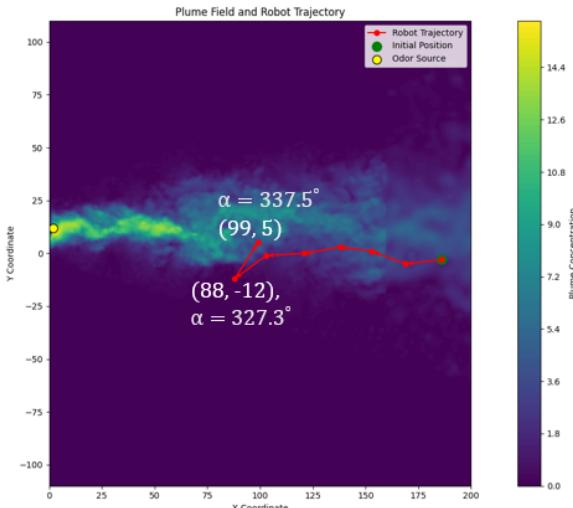
1264 In Case Study II (Figure 23(b)), the scenario is different, prompting the robotic agent  
 1265 to move against the wind direction. The reasoning module again processes the current  
 1266 sensory inputs, retrieves relevant past experiences, and generates a prompt that instructs  
 1267 the robot to move against the wind direction. This decision is made because moving against  
 1268 the wind direction is assessed as the best strategy to locate the odor source based on the  
 1269 environmental conditions and the agent's experiences.

1270 These case studies illustrate how the reasoning module enables the robotic agent to adapt  
 1271 its navigation strategy based on real-time data and stored knowledge, thereby improving the  
 1272 efficiency and accuracy of locating the odor source.

**Case I**

**Scenario description:**

You are moving in an area searching for the plume source. Your current position in frame 6 is at (88.00, -12.00). At this position, the plume concentration is 0.07. The plume concentration increases as you approach close to the source.



**Agent reasoning and decision:**  
The plume concentration at my current position is 0.07, which suggests that I am not near the source since the concentration increases as I get closer to the source.

- Since the goal is to locate the odor source effectively and the concentration is low, I should move in a direction that is likely to lead me to higher concentrations. Moving against the wind direction is typically a strategy used in plume tracing when a plume has been detected because the source is generally upwind. However, since the concentration is low, it might be beneficial to move across the wind direction to search for areas of higher concentration.

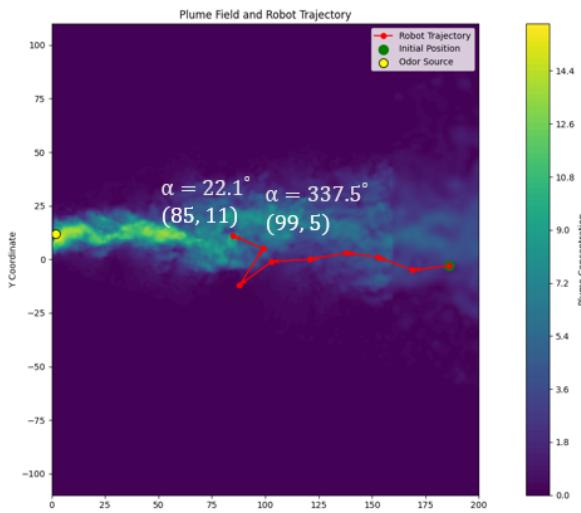
Decision: Move across wind direction

(a)

**Case II**

**Scenario description:**

You are moving in an area searching for the plume source. Your current position in frame 7 is at (99.00, 5.00). At this position, the plume concentration is 4.08. The plume concentration increases as you approach close to the source.



**Agent reasoning and decision:**

The plume concentration at my current position is 4.08, which indicates that I am closer to the source since the concentration is high and increases as I approach the source.

- Since the concentration is high and increases towards the source, the best strategy is to move against the wind direction. This is because in a plume tracing scenario, the source of the odor is typically upwind.

Decision: Move against wind direction

(b)

Figure A.23: (a) Case Study I: Reasoning Module for action 1 - Move across wind direction. (b) Case Study II: Reasoning Module for action 2 - Move against wind direction.

**Biography of Lingxiao Wang:**

Lingxiao Wang received the B.S. degree in Electrical Engineering from the Civil Aviation University of China, Tianjin, China in 2015, M.S. degree in Electrical and Computer Engineering, and Ph.D. in Electrical Engineering and Computer Science from Embry-Riddle Aeronautical University, Daytona Beach, FL, USA in 2017 and 2021, respectively. Currently, he is Assistant Professor of Electrical Engineering at Louisiana Tech University. His current research interests include autonomous systems, robotic applications, and artificial intelligence. He focuses on developing intelligent decision-making models to navigate and control robots using AI methods.

**Biography of Khan Raqib Mahmud:**

Khan Raqib Mahmud is a PhD student of department of Computer Science at Louisiana Tech University in Ruston, Louisiana. He received Master of Science (M. Sc.) degree in Computer Simulation for Science and Engineering, from KTH Royal Institute Technology, Sweden and M. Sc. degree in Computational Engineering from University of Erlangen-Nuremberg, Germany with Erasmus Mundus Scholarship. His research involves Robot Navigation, Computer Vision and Pattern Recognition, Machine Learning, Artificial Intelligence and Autonomous Systems.

**Biography of Sunzid Hassan:**

Sunzid Hassan is a PhD student in the Department of Computer Science at Louisiana Tech University, Ruston, Louisiana 71270, USA. He received his Master's degree in Computer Science from Louisiana Tech University. His current research interests are in the fields of embodied artificial intelligence and robotic odor source localization.

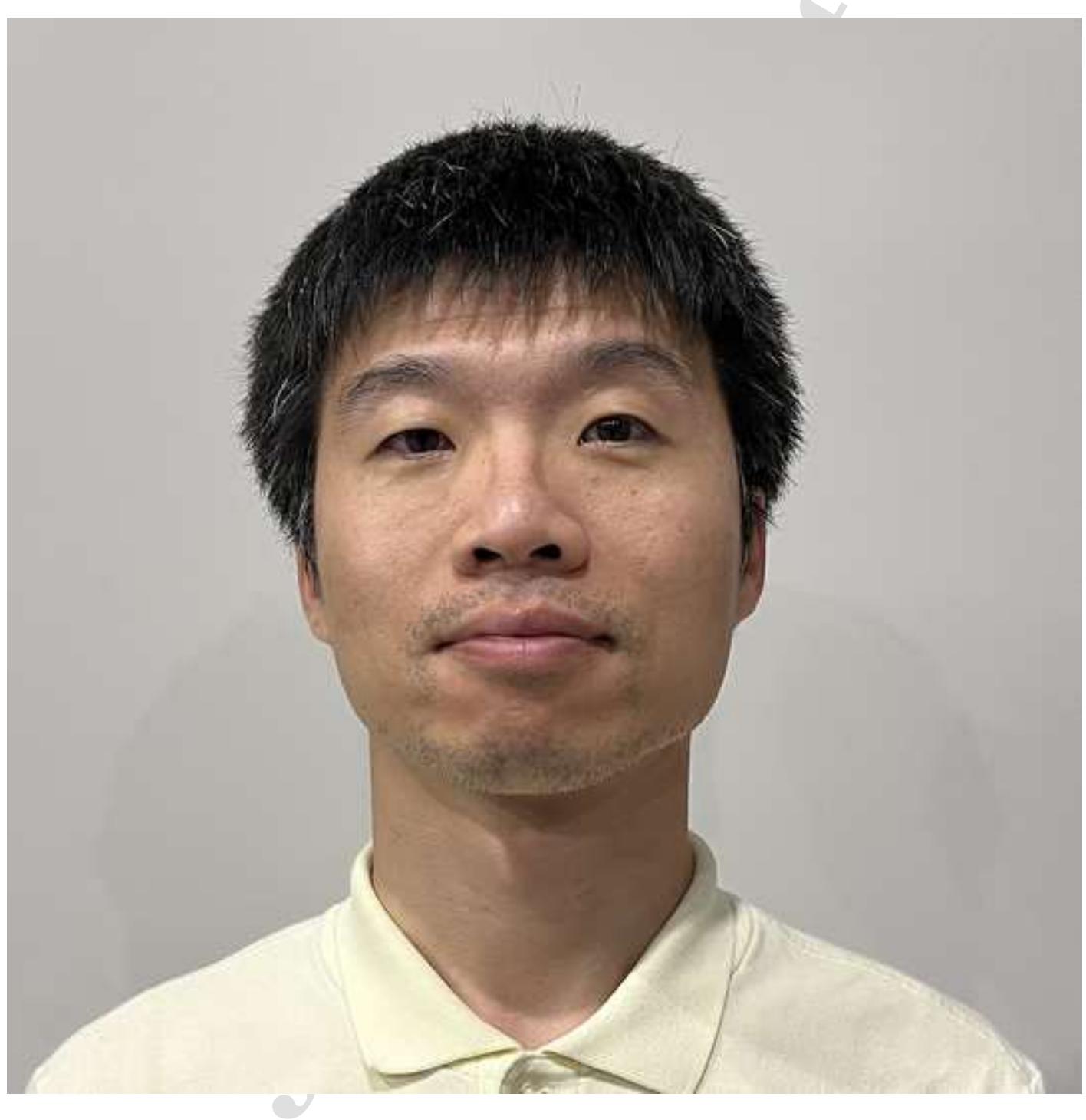
**Biography of Zheng Zhang:**

Dr. Zheng Zhang is an active researcher in experimental aerodynamics, fluid-structure interaction, and developing relevant instrumentation for low-speed wind tunnel testing. He obtained his Ph.D. in Aerospace Engineering and Mechanics from the University of Alabama in 2014. Following his doctorate, he joined Neuroscience Collaborative Center, University of Louisville as an instrumentation developer. Since August 2015, he has been working as a Postdoctoral Scholar, Research Assistant Professor, and then Senior Research Scientist at Department of Aerospace Engineering at Embry-Riddle Aeronautical University, teaching Experimental Aerodynamics and attending developing of ERAU's new low-speed wind tunnel. Dr. Zhang is a Member of AIAA and APS. His researches have been sponsored by funds from DoD, including ONR and ARO.









# A Knowledge-Driven Framework for Robotic Odor Source Localization using Large Language Models

This paper introduces a novel navigation method for controlling a mobile robot to locate a hidden odor source. The key highlights are summarized below:

## 1. Novelty

- We propose a knowledge-driven framework leveraging the reasoning capabilities of Large Language Models (LLMs) for designing the navigation method.
- To the best of our knowledge, this is the first work to utilize LLMs for the odor source localization task.
- The proposed method has been rigorously evaluated in both simulated and real-world environments, demonstrating its effectiveness compared to other learning-based navigation algorithms.

## 2. Technical Advancements

- This study explores the feasibility of employing LLMs as decision-making agents for odor source localization.
- Within the proposed framework, the LLM effectively generates accurate action commands to guide a mobile robot in approaching the hidden odor source.

## 3. Real-World Applications

- The proposed method has broad applicability to solve challenging real-world problems. For instance, it can guide robotic agents to locate chemical gas leaks in hazardous environments, reducing risks for human operators.
- Real-world experimental results show that the proposed approach has the potential for deployment in complex and demanding scenarios.

## 4. Impactful Results

- The method was tested in both simulation and real-world environments. In simulations, realistic odor plume dynamics were modeled using wind tunnel data.
- Real-world experiments employed a ground mobile robot to locate an unknown odor source.
- Results from both scenarios demonstrate that the proposed method outperforms popular learning-based navigation algorithms, such as deep Q-networks, and the traditional moth-inspired navigation method.

**Declaration of interests**

- The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.
- The authors declare the following financial interests/personal relationships which may be considered as potential competing interests:

Lingxiao Wang reports financial support was provided by State of Louisiana Board of Regents. If there are other authors, they declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.