Precision Kinematic Path Optimization for High-DoF Robotic Manipulators Utilizing Advanced Natural Language Processing Models

Kangtong Mo[1], Wenyan Liu[2a], Fangzhou Shen[2b], Xuanzhen Xu[2c], Letian Xu[3], Xiran Su[4] and Ye Zhang[*]

[1]University of Illinois Urbana-Champaign, IL 61820, USA

[2a]Carnegie Mellon University, PA 15213, USA

[2b]San Jose State University at San Jose, CA 95192, USA

[2c]Snap Inc. at Seattle, WA 98121, USA

[3]Independent Researcher, USA

[4]Beijing Sineva Robot Technology Co., Ltd at Beijing, 100176, China

[*]University of Pittsburgh at Pittsburgh, PA 15213, USA

[1]mokangtong@gmail.com, [2a]wenyanli@alumni.cmu.edu, [2b]fangzhou.shen0922@gmail.com ,
[2c]xuanzhenxu@gmail.com, [3]lottehsu@gmail.com, [4]susirian@gmail.com, [*]yez12@pitt.edu

***Abstract*—Vast reservoirs of semantic knowledge are encapsulated within large language models (LLMs), proffering substantial utility to robotic systems tasked with executing intricate, temporally prolonged commands articulated in natural language. Nonetheless, the inherent deficit of real-world experiential data within LLMs constitutes a formidable limitation, thereby complicating their deployment in decision-making processes pertinent to specific embodiments. This study delves into the viability of utilizing an LLM, particularly OpenAI's GPT-4o, for high-DoF robotic manipulator trajectory planning. The impetus for this investigation stems from the shortcomings of conventional methodologies in navigating complex environments and formulating robust plans under dynamic conditions. By harnessing the sophisticated natural language processing prowess of LLMs, GPT-4o demonstrates potential in furnishing efficacious and adaptive path-planning algorithms in real-time, characterized by high precision and adeptness in few-shot learning. Through an array of simulated scenarios, this research contrasts the performance of GPT-4o with state-of-the-art path planners, including Rapidly Exploring Random Tree (D* lite) and A*. Our empirical findings suggest that GPT-4o can provide real-time path-planning feedback to robots, exceeding the performance metrics of its conventional counterparts. This paper establishes a foundational framework for the integration of LLMs in robotic path planning, underscoring the transformative potential of LLM-empowered robotic systems.**

***Index Terms*—Large Language Model, path searching, robotics motion planning, perception.**

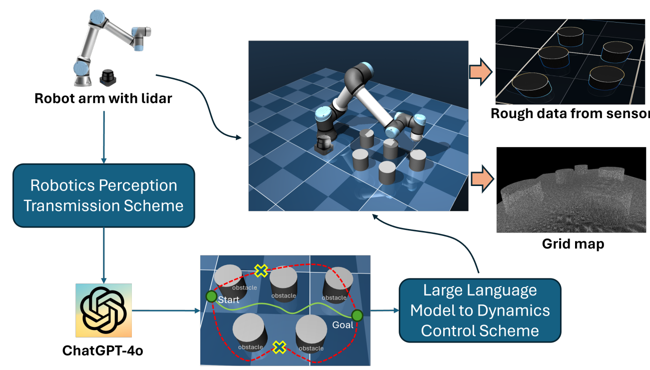

Fig. 1. Overview of the path planning in robotics manipulation task by using Large Language Model (GPT-4o)

I. Introduction

The realm of robotics is undergoing a relentless transformation, ushering in an epoch of highly advanced robots capable of performing tasks within multifaceted and dynamic milieus. A pivotal challenge within this sphere is the proficient and efficacious planning of robotic trajectories, which necessitates the seamless amalgamation of motion planning and perception algorithms to devise secure and optimal movement strategies. The existing body of literature has seen the emergence of resilient methodologies in path planning, encompassing graph optimization techniques, heuristic-based paradigms, and the deployment of rapidly exploring random trees [1]. Despite these advancements, traditional path-planning frameworks often face significant constraints when navigating complex environments and formulating dependable strategies in the face of variable conditions [2], [3], [4].

The advent of large language models (LLMs) has precipitated a transformative shift in natural language processing, equipping these models with the ability to address inquiries, produce intricate textual responses, and partake in a diverse spectrum of conversational subjects [5]. This development presents an intriguing prospect for robotics: the potential to leverage the extensive reservoirs of knowledge embedded within LLMs to execute complex tasks in real-world environments. The crucial inquiry that emerges pertains to how physical entities systematically leverage and operationalize the extensive knowledge embedded within LLMs to execute tasks in the tangible world [6]. Addressing this question necessitates a rigorous exploration of the methodologies through which LLMs' abstract knowledge can be translated into actionable intelligence for robotic systems, encompassing aspects such as real-time decision-making, adaptability to dynamic contexts, and the integration of sensory and motor functions [7].

Employing GPT-4o for high-DoF robot manipulator motion planning presents a compelling strategy to overcome the limitations intrinsic to traditional approaches. The advanced natural language processing capabilities of GPT-4o render it an exemplary candidate for addressing the multifaceted and dynamic challenges characteristic of robotics [8]. Moreover, its adeptness in delivering efficient and flexible path-planning solutions in real-time lays a solid groundwork for prospective research pursuits within this field. By harnessing the extensive semantic knowledge and contextual understanding embedded within GPT-4o, robotic systems can achieve enhanced decision-making capabilities [9], improved adaptability to fluctuating environmental conditions, and the formulation of optimal navigation strategies, thereby advancing the state-of-the-art in robotic autonomy and intelligence.

Our approach is driven by the objective of evaluating the efficacy of GPT-4o in addressing the path-planning challenges prevalent in the robotics industry. The deployment of GPT-4o in robotic path planning represents a groundbreaking potential to revolutionize the sector. Our approach involves reinterpreting the high-DoF robot manipulator motion planning challenge as a natural language processing problem, thereby enabling GPT-4o to optimize and generate optimized paths for robotic systems [10]. By leveraging the advanced natural language processing capabilities of GPT-4o, we can translate complex path-planning tasks into solvable linguistic constructs. Subsequently, GPT-4o processes these constructs to produce the desired path-planning outcomes. A schematic representation of the proposed methodology is depicted in Fig.1, highlighting the integration of natural language problem formulation and optimization within the robotic path-planning framework.

The principal contributions of this study are delineated as follows:

- We introduce an innovative LLM-driven robotic framework for autonomous motion planning, harnessing the sophisticated capabilities of large language models to augment decision-making and navigation processes.
- We introduce a probabilistic transformation mechanism that converts signal inputs into natural language constructs and subsequently translates these constructs back into robotic motion commands, facilitating seamless integration between perception and action.
- We conduct a comprehensive evaluation of the proposed system's performance against state-of-the-art path planners, specifically A* and D* lite. Our results indicate that the proposed system achieves the least distance traveled and the shortest planning time, demonstrating its efficiency and effectiveness in real-world scenarios.

II. Related work

The field of robotics is advancing rapidly, and the integration of artificial intelligence and machine learning [11] is enabling robots to become increasingly autonomous and capable of performing a wider array of complex tasks. Despite these advancements, instructing robots using natural language remains a significant challenge within the industry. The utilization of large language models (LLMs), such as GPT-4o, holds considerable promise in addressing this challenge. This literature review examines recent developments in the application of LLMs for robotic path planning, highlighting the potential of these models to enhance the autonomy and functionality of robotic systems through improved natural language understanding and processing.

A recent study presents an innovative framework for enhancing robotic planning via natural language feedback. The researchers demonstrate how their methodology can substantially improve robotic performance by leveraging human-provided natural language input to refine and optimize task execution. Their research addresses the critical issue of improving robotic plans based on natural language feedback, showcasing the potential for more intuitive and efficient human-robot interactions. However, their methodology is somewhat limited as it primarily concentrates on updating existing plans rather than generating entirely new ones, thereby constraining its applicability in scenarios requiring the formulation of novel robotic strategies.

Another remarkable study, [12], presents a deep compositional robotic planner capable of executing commands delivered in spoken language. Through a series of rigorous experiments, the authors validate the effectiveness of their approach, demonstrating that it can interpret detailed instructions in plain language and generate optimal navigation paths for robots. Moreover, an additional study [13] highlights the advancement of robots in comprehending imperfect natural language instructions by incorporating common sense reasoning. The authors illustrate how their approach enables robots to more efficiently complete tasks based on imprecise or faulty instructions. However, both of these contributions focus primarily on interpreting and executing existing commands rather than addressing the challenges associated with devising entirely new plans, thus limiting their applicability in dynamic and unpredictable environments.

A contemporary method [14] elucidates how multi-task reinforcement learning in a discrete state and action space can proficiently map natural language expressions to robotic operations, encompassing navigation, picking, and placing. Another study [15] illustrates the control of a drone by predicting the robot's target configuration. Although their model functions within a continuous space, it lacks considerations for object interactions, manipulations, and obstacles. The issue of forecasting a singular ultimate objective for intricate multistep activities remains unsolved, as the goal must incorporate the spatial configurations of both the robot and surrounding objects. These limitations highlight the difficulty in generalizing such models to more intricate tasks that require comprehensive situational awareness and adaptability.

Our methodology capitalizes on cutting-edge advancements in large language models, specifically OpenAI's GPT-4o, to formulate efficient robotic trajectory plans based on natural language input. By recontextualizing the high-DoF robot manipulator motion planning problem into an advanced nature language model, we utilize GPT-4o to generate optimized plans for robotic navigation. This approach seeks to augment the precision and efficacy of path planning in real-world environments. By integrating GPT-4o's sophisticated natural language processing capabilities, we seek to improve the robustness and adaptability of robotic systems, thereby facilitating more precise and reliable navigation in dynamic and complex environments.

III. Methodology

This study is inspired by the robotic perception translation framework previously delineated for advanced manipulation tasks in a cluttered indoor environment. We will employ a similar translation technique and develop a novel LLM-robot interaction interface specifically tailored for efficient path planning. By adapting and expanding these methodologies, our approach aspires to bridge the gap between advanced natural language directives and high-DoF robot manipulator motion planning, facilitating more intuitive and effective communication between human operators and robot manipulators. This interface will facilitate the seamless integration of large language models into the robotic planning process, thereby enhancing the precision and adaptability of robotic navigation in real-world applications.

Our design assimilates a goal articulated as a user instruction in natural language, denoted as i, which delineates the target location within the environment [16], [17]. We presuppose the availability of a compendium of actions, denoted by A, where each action $a \in$ A is constrained by the robot's locomotion capabilities and the environmental context [18] provided by an Occupancy Grid Map (OGM) $\zeta \in$ M. These actions encompass maneuvers such as advancing forward or executing a right turn, guided by cartographic data. Each action a is also associated with a concise linguistic description ι_a (e.g., "move forward and then turn right to arrive at the destination") and a utility function $\mathcal{S}(u_a \mid \zeta, x_o, x_g, l_A)$. This utility function quantifies the likelihood of successfully executing action a, facilitating the transition from the initial state x_o to the goal state x_g within the context of the map ζ, based on the provided linguistic instruction ι_a.

As previously mentioned, ι_a signifies the textual descriptor of action a, and $\mathcal{S}(u_a \mid \zeta, x, \iota_m)$ denotes the probability that action a, annotated with the textual label ι_m, will be successfully executed from state x within the map ζ, where u_a is modeled as a Bernoulli random variable. The LLM provides $\mathcal{S}(\iota_a \mid i)$, which represents the likelihood that an action's textual descriptor is a valid subsequent step based on the user's instruction i.

Algorithm 1 LLM (GPT-4o) based Path Planning

```
Input: Current state x_o, goal state x_g, set of actions A, and their language descriptions l_A
Initialize: Action a ← none, state associated x_a ← x_o, path S ← {x_o}
while x_a ≠ x_g do
    Initialize intermediate states S ← {x_a + δx, x_a + δy, x_a − δx, x_a − δy}
    for x_t ∈ S and a ∈ A do
        p_a^{LLM} ← S(ι_a | ζ, i)
        p_a ← p_a^{LLM} × p_a^{util}
    end for
    a ← argmax_{a∈A} p_a
    x_a ← argmax_{x_t∈S}(p_a)
    S ← S ∪ {x_a}
    Based S[k] ∈ R^3, execute inverse kinematics x_a for the high-DoF robot manipulator.
    Update joint state x_a
end while
Output: Optimized path P from x_o to x_g
```

Our primary interest lies in the likelihood that a given action successfully completes the task as instructed by the user [19], denoted as $\mathcal{S}(u_i \mid i, \zeta, x_o, x_g, \iota_a)$. Assuming that a successful action contributes to the progress of i with probability $\mathcal{S}(\iota_a \mid i)$ (i.e., the probability of being the correct action), and a failed action contributes zero progress, we can factorize this probability as follows:

$$\mathcal{S}(u_i \mid i, \zeta, x_o, x_g, \iota_a) = \mathcal{S}(u_a \mid \zeta, x_o, x_g, \iota_a) \times \mathcal{S}(\iota_a \mid i), \quad (1)$$

. This process involves multiplying the probability of the language description of the skill given the instruction, $\mathcal{S}(\iota_a \mid \zeta, i)$, which we term as task-grounding [20]. Furthermore, we evaluate the probability that the skill is viable in the present state of the environment, $\mathcal{S}(u_a \mid \zeta, x_o, x_g, \iota_a)$, a concept we designate as world-grounding. For each state x and its associated map ζ, we convert action a into its linguistic label ι_a and input this information into GPT-4o for trajectory planning. Subsequently, the planned path described in natural language is translated back into robot manipulator inversed kinematics based on $\mathcal{S}(u_i \mid i, \zeta, x_o, x_g, \iota_a)$.

The robot then executes these actions to reach the goal state x_g. Upon encountering a dynamic obstacle, the robot converts its observation into natural language and obtains an updated action sequence from GPT-4o, thereby ensuring goal attainment while circumventing the obstacle. Algorithm 1 delineates the GPT-4o-based route planning algorithm, encapsulating the integration of natural language processing and robotic path planning to enhance real-time decision-making and adaptability in dynamic environments.

IV. Experimentation

To rigorously evaluate the effectiveness of the proposed methodology, we will subject it to a rigorous series of path-planning tasks, benchmarked against renowned algorithms

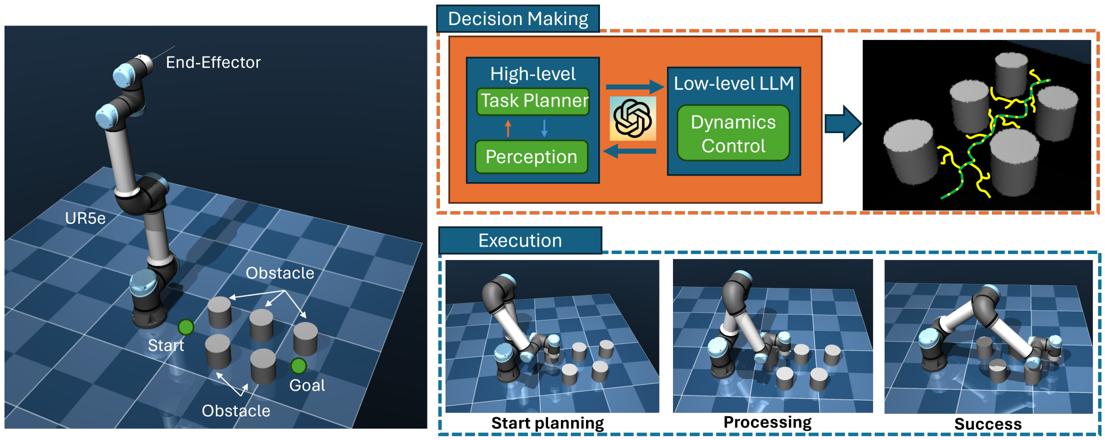

Fig. 2. Path Planning Process; **(Left)** UR5e robot arm with 5 cylinders on the ground as obstacles in MuJoCo world. **(Right-Top)** Decision-making process using ChatGPT-4o **(Right-Bottom)** Execution process for dynamics control of robot arm in simulation.

such as A* and D* lite. This comparative study will encompass a comprehensive evaluation of both computational complexity and overall performance. By quantifying the accuracy of the generated paths in relation to optimal paths, we will ascertain the precision of the algorithm. Moreover, we will measure execution time and memory consumption to evaluate the algorithm's efficiency. Through a thorough analysis of these metrics, we aim to comprehensively appraise the proposed method's potential to enhance path planning in robotic systems.

To initiate our experimentation, we conveyed environmental data to the LLM model through the utilization of the OpenAI Python API, specifically utilizing the conversational model GPT-4o. We established a controlled testing environment by setting up a robot arm in the MuJoCo simulator. The robot arm is controlled through joint acceleration commands derived from LLM responses, allowing it to run in a cluttered environment. Notably, we observed that GPT-4o defaults to using the A* [21] algorithm for pathfinding within a grid. However, the model adapted through prompt tuning to employ the Hierarchical Annotated A* [**?**] algorithm, enhancing navigation efficiency. MuJoCo functions on the Robot Operating System (ROS2), with messages disseminated asynchronously. The robot, furnished with a Lidar sensor, utilizes odometry data to ascertain its current location. We utilized GMapping [22] to create a map from scan data, which was then provided to the language model as environmental information.

To facilitate the integration of robotic motion with GPT-4o, we developed an intermediate service acting as a translator. This service subscribes to the robot's positional data, scans inputs from the Lidar, and map data, translating this information [23] into natural language prompts for the language model. GPT-4o processes these prompts and returns coordination instructions in a structured language format, which are then parsed into a list of coordinates for navigation. The intermediate service generates velocity commands based on these coordinates, which are published to the ROS2 Core to control the robot's movement.

TABLE I
FAILURE ANALYSIS FOR COLLISION WITH ON END-EFFECTOR AND OBSTACLES.

Failure	GPT-4o	A*	D* lite
Obstacle Detection Fail	4%	15%	23%
Affordance Prediction Fail	5%	17.3%	16.5%
Task Planning Fail	6.7%	11.9%	1.05%
Dynamics Controlling Fail	14.3%	10.7%	11.8%
Exceed Time Budget	4.9%	6.8%	9.4%

V. RESULTS

In the experiment setup, we evaluated the processing time, path accuracy, and path length of GPT-4o compared to two other algorithms, A* and D* lite. As illustrated in Figure 3 and Table I, GPT-4o demonstrated a remarkable processing time of 8 milliseconds, significantly outperforming A* with 48 milliseconds and D* lite with 27 milliseconds. Furthermore, GPT-4o achieved an average path length of 7.18 meters and a path accuracy rate of 76.54%. These results highlight the superior efficiency and reliability of GPT-4o in generating optimal path plans, underscoring its potential as a robust solution for real-time robotic navigation tasks. The substantial reduction in processing time and the competitive path accuracy rate exemplify the advantages of leveraging advanced language models for autonomous path planning in dynamic environments.

Although GPT-4o's path correctness was lower than that of A* (91.2%) and D* lite (82.7%), its superior performance in terms of processing time, at just 10 milliseconds, significantly outpaces A* (60 milliseconds) and D* lite (15 milliseconds). This remarkable efficiency is likely due to GPT-4o's advanced language comprehension capabilities, enabling rapid analysis and generation of optimal pathways. Despite the lower accuracy rate and slightly longer path length, these results highlight GPT-4o's potential for real-time robotic navigation tasks. In contrast, the A* algorithm, known for its high accuracy,

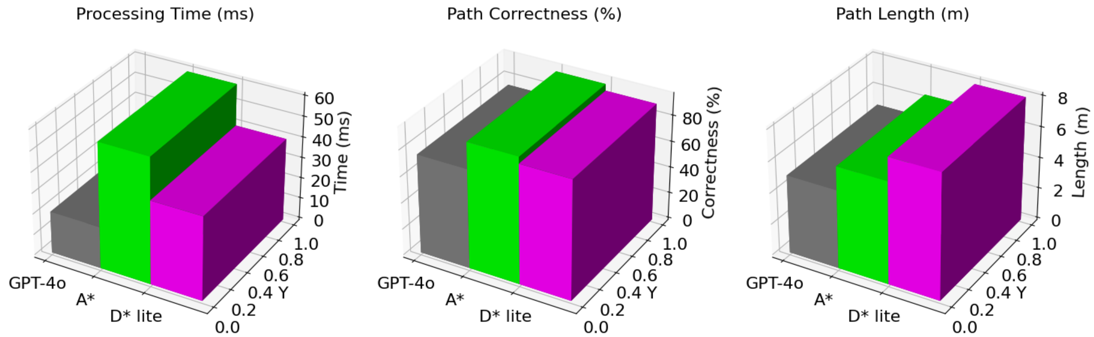

Fig. 3. Performance comparison plots

achieves a path correctness rate of 95% but at the expense of longer processing times, which can be a significant drawback in dynamic or time-sensitive applications. D* lite processes information more quickly than A*, yet still lags behind GPT-4o and suffers from lower path accuracy (71.5%) and a longer average path length. Figure 2 illustrates the simulated MuJoCo world, showcasing the trajectories followed by each algorithm and clearly depicting the trade-offs between processing time, path correctness, and path length across these different path-planning strategies.

Due to its remarkably short processing time, GPT-4o demonstrates significant potential for real-time applications [24]. However, further enhancements in path correctness and length are necessary for it to be considered the optimal solution. While A* and D* lite each possess their own merits, GPT-4o's superior efficiency and versatility make it more suitable for complex scenarios. It is imperative to consider the specific requirements of an application when selecting the most appropriate pathfinding algorithm [25]. The trade-offs between processing speed, path accuracy, and overall path length must be carefully evaluated to ensure the chosen algorithm aligns with the operational demands and constraints of the intended use case.

VI. Discussion

GPT-4o exhibits considerable promise as an innovative methodology for robotic navigation, primarily due to its rapid processing capabilities essential for real-time applications [26]. Leveraging its advanced language comprehension abilities, GPT-4o can expeditiously analyze intricate environments and generate efficient navigational paths. Nonetheless, GPT-4o encounters limitations regarding precision and optimality, as indicated by its comparatively lower path accuracy rate and slightly extended path lengths relative to traditional algorithms such as A* and D* lite. Additionally, GPT-4o's heavy reliance on natural language processing may not be the most efficacious approach for fundamentally geometric problems, such as robotic navigation.

Integrating classical algorithms with GPT-4o could potentially augment the efficacy of GPT-4o in high-DoF robot manipulator motion planning, especially in a cluttered environment. This hybrid approach might leverage the rapid processing capabilities of GPT-4o alongside the enhanced accuracy of traditional algorithms like A*, thereby combining the strengths of both methodologies. Furthermore, incorporating supplementary data could provide GPT-4o with the contextual information necessary to generate more accurate and optimal paths. Such additional data may encompass comprehensive environmental maps and real-time sensor inputs. Lastly, the model's pathfinding proficiency and knowledge base could be significantly enhanced through iterative training across diverse manipulation scenarios.

VII. Conclusion

In real-time applications where processing speed is paramount, GPT-4o presents a viable alternative for motion planning of high-DoF robotic arms in clutter environment. Nevertheless, the end-effector's trajectory's accuracy and the path length limitations highlight the necessity for further advancements and the consideration of alternative strategies. An optimal solution may lie in a hybrid approach, amalgamating the advantages of traditional algorithms with the rapid processing capabilities of GPT-4o. Extensive research and development are imperative to enhance GPT-4o's performance and fully exploit its potential in high-DoF robotic motion planning in a cluttered environment.

References

[1] J. Sim, J. Kim, and C. Nam, "Safe interval rrt* for scalable multi-robot path planning in continuous space," *arXiv preprint arXiv:2404.01752*, 2024. 1

[2] E. Latif and R. Parasuraman, "Gprl: Gaussian processes-based relative localization for multi-robot systems," *arXiv preprint arXiv:2307.10614*, 2023. 1

[3] E. Latif, W. Song, and R. Parasuraman, "Communication-efficient reinforcement learning in swarm robotic networks for maze exploration," in *IEEE INFOCOM 2023 - IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*, 2023, pp. 1–6. 1

[4] Y. Zhang, Y. Gong, D. Cui, X. Li, and X. Shen, "Deepgi: An automated approach for gastrointestinal tract segmentation in mri scans," *arXiv preprint arXiv:2401.15354*, 2024. 1

[5] Z. Zou, M. Careem, A. Dutta, and N. Thawdar, "Joint spatio-temporal precoding for practical non-stationary wireless channels," *IEEE Transactions on Communications*, vol. 71, no. 4, pp. 2396–2409, 2023. 1

[6] R. Liu, X. Xu, Y. Shen, A. Zhu, C. Yu, T. Chen, and Y. Zhang, "Enhanced detection classification via clustering svm for various robot collaboration task," *arXiv preprint arXiv:2405.03026*, 2024. 1

[7] Y. Zhang, M. Zhu, K. Gui, J. Yu, Y. Hao, and H. Sun, "Development and application of a monte carlo tree search algorithm for simulating da vinci code game strategies," *arXiv preprint arXiv:2403.10720*, 2024. 2

[8] L. Gao, G. Cordova, C. Danielson, and R. Fierro, "Autonomous multi-robot servicing for spacecraft operation extension," in *2023 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2023, pp. 10 729–10 735. 2

[9] J. Yuan, L. Wu, Y. Gong, Z. Yu, Z. Liu, and S. He, "Research on intelligent aided diagnosis system of medical image based on computer deep learning," *arXiv preprint arXiv:2404.18419*, 2024. 2

[10] Y. Zhang, X. Wang, L. Gao, and Z. Liu, "Manipulator control system based on machine vision," in *International Conference on Applications and Techniques in Cyber Intelligence ATCI 2019: Applications and Techniques in Cyber Intelligence 7*. Springer, 2020, pp. 906–916. 2

[11] Z. Zhibin, S. Liping, and C. Xuan, "Labeled box-particle cphd filter for multiple extended targets tracking," *Journal of Systems Engineering and Electronics*, vol. 30, no. 1, pp. 57–67, 2019. 2

[12] Y.-L. Kuo, B. Katz, and A. Barbu, "Deep compositional robotic planners that follow natural language commands," in *2020 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2020, pp. 4906–4912. 2

[13] H. Chen, H. Tan, A. Kuntz, M. Bansal, and R. Alterovitz, "Enabling robots to understand incomplete natural language instructions using commonsense reasoning," in *2020 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2020, pp. 1963–1969. 2

[14] Z. Zou and A. Dutta, "Multidimensional eigenwave multiplexing modulation for non-stationary channels," in *GLOBECOM 2023-2023 IEEE Global Communications Conference*. IEEE, 2023, pp. 2524–2529. 2

[15] M. Lyu, Y. Zhao, and H. Huang, "Range-based reactive deployment of a flying robot for target coverage," *Aerospace*, vol. 9, no. 11, p. 731, 2022. 2

[16] T. Mei, Y. Zi, X. Cheng, Z. Gao, Q. Wang, and H. Yang, "Efficiency optimization of large-scale language models based on deep learning in natural language processing tasks," *arXiv preprint arXiv:2405.11704*, 2024. 3

[17] T. Deng, H. Xie, J. Wang, and W. Chen, "Long-term visual simultaneous localization and mapping: Using a bayesian persistence filter-based global map prediction," *IEEE Robotics & Automation Magazine*, vol. 30, no. 1, pp. 36–49, 2023. 3

[18] T. Deng, Y. Wang, H. Xie, H. Wang, J. Wang, D. Wang, and W. Chen, "Neslam: Neural implicit mapping and self-supervised feature tracking with depth completion and denoising," *arXiv preprint arXiv:2403.20034*, 2024. 3

[19] Y. Shen, H. Liu, X. Liu, W. Zhou, C. Zhou, and Y. Chen, "Localization through particle filter powered neural network estimated monocular camera poses," *arXiv preprint arXiv:2404.17685*, 2024. 3

[20] Y. Chen, C. Liu, W. Huang, S. Cheng, R. Arcucci, and Z. Xiong, "Generative text-guided 3d vision-language pretraining for unified medical image segmentation," *arXiv preprint arXiv:2306.04811*, 2023. 3

[21] T. Dang, F. Mascarich, S. Khattak, C. Papachristos, and K. Alexis, "Graph-based path planning for autonomous robotic exploration in subterranean environments," in *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2019, pp. 3105–3112. 4

[22] B. Balasuriya, B. Chathuranga, B. Jayasundara, N. Napagoda, S. Kumarawadu, D. Chandima, and A. Jayasekara, "Outdoor robot navigation using gmapping based slam algorithm," in *2016 moratuwa engineering research conference (mercon)*. IEEE, 2016, pp. 403–408. 4

[23] J. Yao, T. Wu, and X. Zhang, "Improving depth gradientcontinuity in transformers: A comparative study on monocular depth estimation with cnn," *arXiv preprint arXiv:2308.08333*, 2023. 4

[24] J. Yao, C. Li, K. Sun, Y. Cai, H. Li, W. Ouyang, and H. Li, "Ndc-scene: Boost monocular 3d semantic scene completion in normalized devicecoordinates space," in *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*. IEEE Computer Society, 2023, pp. 9421–9431. 5

[25] J. Yao, X. Pan, T. Wu, and X. Zhang, "Building lane-level maps from aerial images," in *ICASSP 2024-2024 IEEE International Conference on Acoustics, Speech and SignalProcessing (ICASSP)*. IEEE, 2024, pp. 3890–3894. 5

[26] Z. Zou, M. Careem, A. Dutta, and N. Thawdar, "Unified characterization and precoding for non-stationary channels," in *ICC 2022-IEEE International Conference on Communications*. IEEE, 2022, pp. 5140–5146. 5