COMMET: A System for Human-Induced Conflicts in Mobile Manipulation of Everyday Tasks

Dongping Li[1,2*], Shaoting Peng[1*], John Pohovey[1], Katherine Rose Driggs-Campbell[1]

***Abstract*— Continuous advancements in robotics and AI are driving the integration of robots from industry into everyday environments. However, dynamic and unpredictable human activities in daily lives would directly or indirectly conflict with robot actions. Besides, due to the social attributes of such human-induced conflicts, solutions are not always unique and depend highly on the user's personal preferences. To address these challenges and facilitate the development of household robots, we propose COMMET, a system for human-induced COnflicts in Mobile Manipulation of Everyday Tasks. COMMET employs a hybrid detection approach, which begins with multi-modal retrieval and escalates to fine-tuned model inference for low-confidence cases. Based on collected user preferred options and settings, GPT-4o will be used to summarize user preferences from relevant cases. In preliminary studies, our detection module shows better accuracy and latency compared with GPT models. To facilitate future research, we also design a user-friendly interface for user data collection and demonstrate an effective workflow for real-world deployments.**

I. Introduction

Intelligent household robots have been a long-standing goal in robotics. In recent years, advancements in Large Language Models (LLMs) [1], [2] and embodied intelligence [3], [4] have greatly accelerated progress toward this goal, endowing robots with powerful abilities in perception, planning, navigation and manipulation [5], [6], [7], [8], [9], [10], [11]. Besides, considering the social attribute of household robots, various Human-Robot Interaction (HRI) studies [12], [13], [14], [15], [16] aim to facilitate robots better integrate into people's daily lives. Meanwhile, works about adaptive methods and failure detection [17], [18], [19], [20] focus on enabling robots to handle real-world disturbances. However, as opposed to more controllable environments such as laboratories or simulations, **dynamic human activities in the real world will induce more uncontrollable factors, which could cause various conflicts with the robot's actions**. Indirect conflicts like changes in environmental states or object locations; Direct conflicts like obstructing or interrupting the robot's planning and executions. These conflicts will influence robots at both low levels and high levels. Although some works have explored robots in dynamic environments [17], [21], [22], the discussion of the impact of human activities remains limited. Additionally,

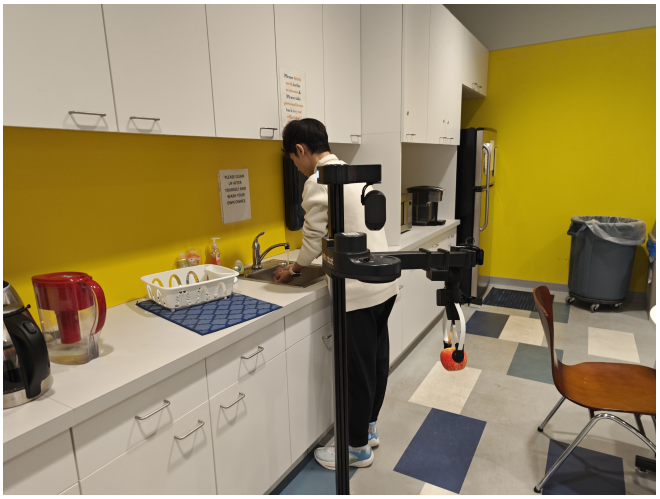

Fig. 1. **An illustration for human-induced conflicts.** The robot is ordered to put the apple into the sink, but another person is washing hands now.

since multiple persons may simultaneously exist in a household environment, the relationships between each individual and the robot would be more complex. Humans can either be the subjects who instruct and collaborate with robots, or they can be the disturbances that are unrelated to the robot's tasks, but current HRI researches mainly focus on the former cases and overlook interactions when humans serve as disturbances.

Moreover, the **solution to the human-induced conflict is not unique and it largely depends on the user's preferences and the current task context**. For a conflict shown in Fig. 1, if the robot wants to finish user tasks, feasible solutions would include *stop and wait*, *go to another sink*, *ask the user for help*, or *tell the person to leave*, and the optimal option depends on the user's demand. If the user is in a hurry, *tell the person to leave* would be the best choice. Therefore, when robots are trying to solve such conflicts, they should also take user's preferences into account.

To address these challenges, we propose *COMMET*, a system designed to detect and resolve various human-induced conflicts. *COMMET* contains both offline and online stages. In the offline stage, we first collect static scenario and dynamic trajectory data that capture normal cases and potential conflict types in real life, then construct retrieval buffers and model training sets. Next, users need to annotate selected data by providing their preferred solution option and an emergency level which reflects the user's concern level of the current conflict. In the online stage, *COMMET* starts with

* Equal Contribution

[1] D. Li is a visiting student of University of Illinois Urbana-Champaign. S. Peng, J. Pohovey and K. Driggs-Campbell are with the Department of Electrical and Computer Engineering at University of Illinois Urbana-Champaign. emails: {dongping, peng33, jpohov2, krdc}@illinois.edu

[2] D. Li is also with ZJU-UIUC Institute, Zhejiang University. emails: dongping.23@intl.zju.edu.cn

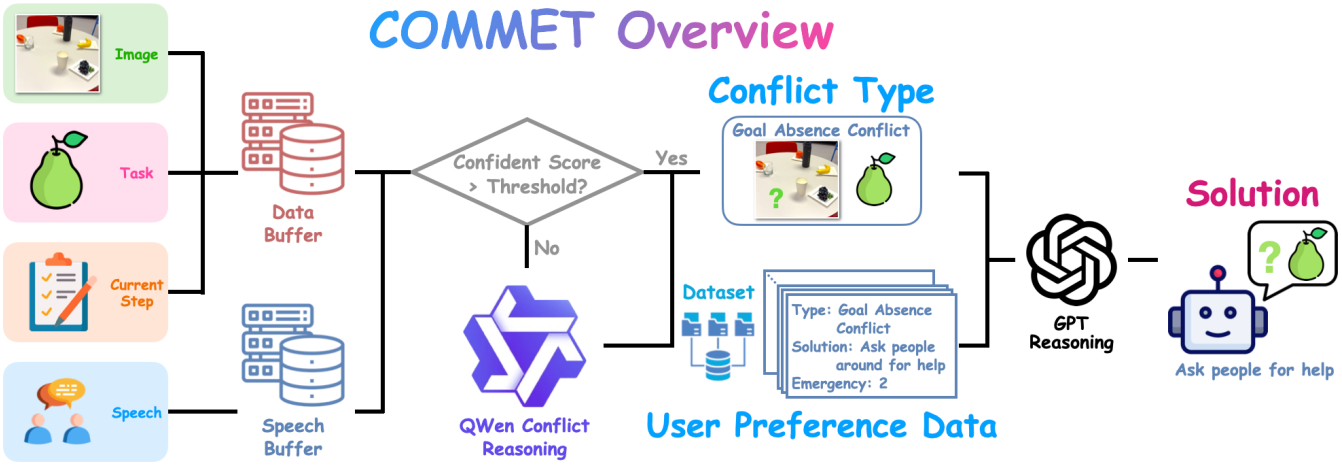

Fig. 2. **System Overview**

the real-time conflict detection, which compares the current robot states and sensor inputs with a speech retrieval buffer and a multi-modal task attribute retrieval buffer. The retrieval output involves the potential conflict type and a confidence score. When the score falls below a pre-set threshold, a fine-tuned LLM will be called for further inference. Once a conflict is detected, *COMMET* will extract corresponding user data based on the detected conflict type, then use GPT-4o [1] to summarize the user's preferences and provide the final prediction. In preliminary studies, we analyze the accuracy and latency of our detection module and compare it against GPT [1] to show its efficiency and effectiveness. We also design a user-friendly interface to collect user preference data and investigate user ratings of preference prediction. Finally, we discuss how to deploy *COMMET* in the real world and show a potential running workflow.

In summary, our paper makes the following contributions:

- **Multi-stage Conflict Detection:** *COMMET* will first conduct real-time speech retrieval and multi-modal task attribute retrieval, and use a fine-tuned model if the retrieval confidence falls below a threshold, the specific conflict type will also be determined simultaneously.
- **Fine-grained User Preference Prediction:** Except solution options, an emergency level is set to reflect the user's concern level. The input user cases will also vary from the conflict type, then a LLM is asked to summarize user's preferences and provide final predictions.
- **System Performance and Feasibility Analysis:** We design a user-friendly interface to facilitate future user studies. We also compare our system's detection performance with GPT-4o, and discuss the possibility and an effective workflow for real-world deployment.

II. COMMET

A. Problem Statement

Our work first makes following assumptions: 1) all conflicts are raised by humans, rather than the robot's planning or execution abilities. 2) all available options are feasible to solve the conflicts. Then the robot needs to detect indirect and direct conflicts brought by humans during the task execution, determine the specific conflict category, and choose a preferred solution based on stored user cases and settings.

B. System Overview

As is shown in Fig.2. The input of *COMMET* includes four components: the real-time visual observation, the final user task, the current step, and the transcribed background speech. We first build two retrieval buffers based on collected user requests, robot trajectories, and anomalous cases. Then we conduct speech retrieval and task attribute retrieval respectively. If the similarity falls below pre-set thresholds, a fine-tuned multi-modal LLM will be used for further inference. Both methods will return a specific conflict type or normal if no conflict is detected. Then we'll input corresponding user options and emergency levels to GPT-4o, which is prompted to summarize user preferences and select a preferred solution. We'll introduce conflict classification in Section II-C, data collection in Section II-D, conflict detection in Section II-E, user preference prediction in Section II-F.

C. Conflict Category

To conduct more fine-grained detection and facilitate the collection of user preferences, we categorize the most common human-induced conflicts during daily task executions into the following four types: 1) **Goal Absence Conflict**: the robot is going to operate but the target is missing. 2) **Human Interaction Conflict**: the robot is executing tasks but another person attempts to command the robot or interact with it. 3) **Human Occupancy Conflict**: Another person's activities occupy the specific space or object, thus hindering the task execution. 4) **Object State Conflict**: current goal states can't satisfy the task needs, or the current step can't continue due to object state changes (e.g., the door is closed or the container is full).

D. Data Collection

We first design static conflict scenarios in 7 different home or indoor environments and include various attributes such

as pets, family members, kitchens, bathrooms, living-rooms and so on. Given that real-world detection is a continuous process, we also simulate a first-person perspective of the robot to collect execution trajectories, which include both normal and anomalous cases. All data will be annotated with the observation, task, current step, speech, and conflict type.

To ensure our system can effectively handle potential disturbances that the robot may encounter in the real world, we process the collected data as follows: 1) The **detection occasion should be time-sensitive**. As there is always a delay between detecting and actually encountering a conflict, it may disappear if the delay is too long. For instance, the robot detects a distant person blocks its path, but the person may move away before the robot actually arrives, thus no actual conflict occurs. Therefore, we maintain a safety range of approximately one meter, conflicts will only be detected when a person or object enters this zone. 2) To **resist the interference of noise in the real world**, we incorporate daily human conversations that are unrelated to the robot into the dataset. In such cases, the robot should ignore noise and continue operating its original tasks. Finally, we collect 1759 data samples, consisting of 134 static scenarios and 1625 trajectory data captured from 21 distinct tasks.

E. Conflict Detection

Since real-time detection demands highly on latency, we choose light-weighted embedding models to perform similarity retrieval by default. Specifically, we use gte-large-en-v1.5 (434M) [23] to calculate text embeddings and clip (428M) [24] to calculate image embeddings. Moreover, we found that retrieving speech and task attributes (consisting of the user task and the current task step) separately performs better than joint retrieval. Therefore, we design two embedding buffers B_s and B_m based on the data collected in Section II-D, and conduct two parallel retrieval processes. B_s is a text-only speech buffer, while B_m is a multi-modal embedding buffer that stores both image embeddings and task attribute embeddings.

For speech retrieval, we first convert the detected background audio into text and calculate its embedding E_s. Then we calculate the cosine similarity between E_s and each embedding E_i in B_s. If the max score S_s surpasses a threshold τ_s, we'll determine that another person is attempting to interact with the robot and directly trigger a *Human Interaction Conflict*. For task attribute retrieval, we calculate both the prompt embedding E_p and image embedding E_{obs} similarity with each E_i in B_m, and utilize a weight parameter w to calculate the final score S_t. Then corresponding conflict or normal information will be extracted.

$$S_s = \max_{E_i \in B_s} (\cos(E_s, E_i)) \tag{1}$$

$$S_t = \max_{E_i \in B_m} \left(w \cdot \cos(E_i^p, E_p) + (1 - w) \cdot \cos(E_i^{obs}, E_{obs}) \right) \tag{2}$$

Although retrieval methods allow us to efficiently leverage existing data with very low latency, manually collected data is limited and often fail to cover all possible conflicts in the real world. It also struggles to generalize to new environments and tasks. These issues are particularly prominent in task attribute retrieval, as the user's demands are various. So we set another threshold τ_t for task attribute retrieval, if S_t falls below τ_t, we assume the current situation may not be represented in B_m, and we will leverage powerful commonsense and reasoning capabilities of LLMs for further detection. In this stage, we use previous data to finetune Qwen 2.5VL-3B and 7B [25], which perform well on multi-modal tasks while maintaining a relatively small parameter size. The model input and output formats are consistent with those in the retrieval stage. We will analyze system latency and hyperparameters w, τ_s and τ_t in Section III.

TABLE I
AVAILABLE SOLUTIONS FOR EACH TYPE OF CONFLICT

Conflict Type	Solution
Goal Absence Conflict	Ask people around for help
	Find another similar spot or object
	Re-calculate the path or make a new task plan
	Inform the user and wait for instructions
Human Occupancy Conflict	Stop execution and wait for the person
	Directly communicate with the person
	Find another similar spot or object
	Inform the user and wait for instructions
Object State Conflict	Ask people around for help
	Find another similar spot or object
	Re-calculate the path or make a new task plan
	Inform the user and wait for instructions
Human Interaction Conflict	Ignore and keep original steps
	Pause current actions and interact with person (chat)
	Switch to new user or task
	Inform the user and wait for instructions

F. User Preference Prediction

As is shown in Table I, we define several general solutions based on the likelihood of real-world scenarios for each type of conflict. Before the system is running, we'll provide users with 20 conflict scenarios (each type contains 5 cases), they need to select one preferred solution and the emergency level for each scenario. The emergency level has three layers and it reflects the user's concern level and the subjective impact of the current situation: the higher level means the current situation is more urgent to the user. In similar cases, the system should prioritize quickly executing the user's current task. Likewise, a lower level indicates that users would agree to make certain compromises in time or task performance to resolve conflicts in a more harmonious or easier way. Once a conflict is detected, the current scenario and user data with the same conflict type will be sent together to GPT-4o. It needs to analyze provided user data to generate a summary of user's preferences, then choose the option that best aligns with the user's preferences.

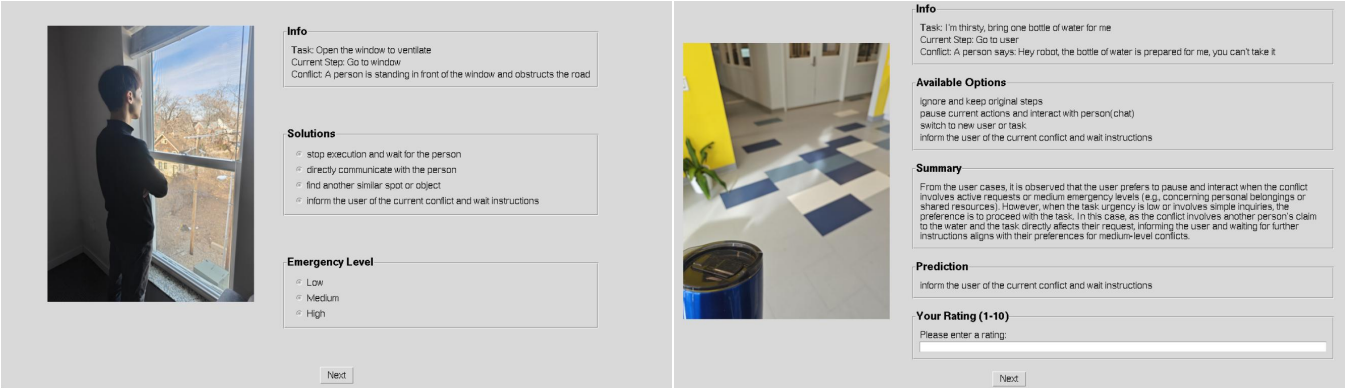

Fig. 3. **User Interface.** The left interface collects user preferences, and the right interface collects user final ratings.

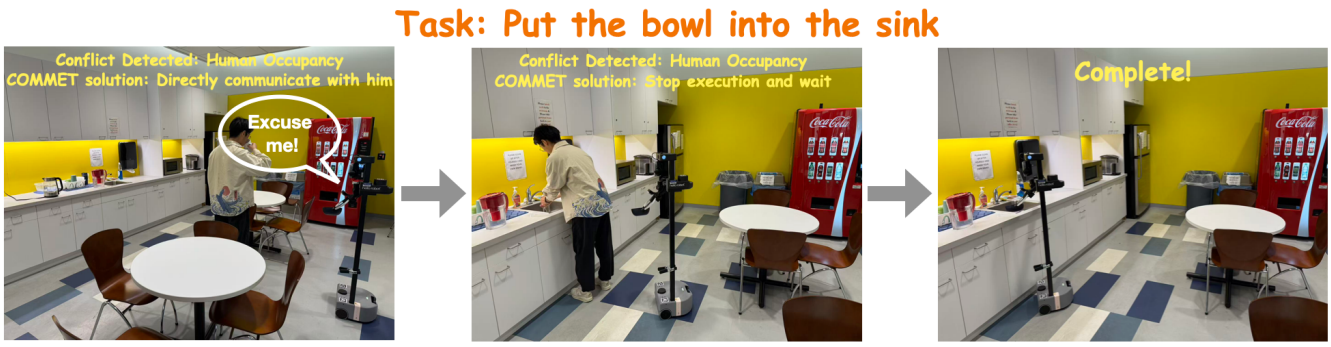

Fig. 4. **Real-world Pipeline Example.** Suppose the task is to put the bowl into the sink. On the robot's way to the sink, a human is blocking the way. After identifying the conflict type, *COMMET* reasons based on the user preference data, successfully completing the task.

TABLE II

PERFORMANCE COMPARISON OF MODELS AND METHODS

Model	Total Acc.	Normal Acc.	Anomaly Acc.	Time (s)
GPT-4o	73.58	91.67	50.00	5.0058
Retrieval(unified)	65.57	75.83	52.17	0.0711
Retrieval(separate)	75.00	90.00	55.43	0.0586
Qwen2.5-VL-3B(ft)	83.96	98.33	65.22	1.5496
Qwen2.5-VL-7B(ft)	87.26	95.00	77.17	2.0107
COMMET(3B)	84.43	98.33	66.30	1.3440
COMMET(7B)	87.26	95.00	77.17	1.8117

III. PRELIMINARY STUDY

A. System Hyperparameter and Performance

For the data collected in Section II-D, 224 data samples will be used to construct the test set, which includes the complete trajectory data for two tasks and 32 static scenarios. The remaining 1535 data samples will be used to construct the embedding buffer for the retrieval module and the training set for model fine-tuning.

Next, we determine the values of w, τ_s, and τ_t in sequence. We begin with unified retrieval where the prompt directly incorporates speech text. By varying w from 0 to 1 in increments of 0.01 and observing the corresponding average detection accuracy, we identify the optimal value of w is 0.87. After that, we conduct separate retrieval while w is fixed. Through the same procedure, we obtain the optimal τ_s of 0.88. Finally, we integrate all modules into *COMMET*, keeping w and τ_s fixed. We find that the optimal τ_t is 0.94 when the system is with 3B model and 0.93 with 7B model.

Finally, we measured the average time per detection. Since our data contains both normal and anomalous cases, we also calculate the individual detection accuracy for each case in addition to the total accuracy. Table II presents performance comparisons across our system, different detection methods, and various detection models.

B. User Preference Prediction

we select 10 samples for each conflict type from the static scenario data. As is shown in Fig. 3, users need to provide their preferred solutions and emergency levels for half of these samples. Then each sample in the remaining half will be sent to GPT-4o, along with five samples of the same conflict type from the user data. GPT-4o will summarize the user's selection preferences for the current conflict type and then predict the user's preferred option. After all predictions are completed, users can rate model predictions to provide feedback for future studies.

C. Real-World Deployment

Since *COMMET* primarily focuses on detecting human-induced conflicts and providing preference-aware solutions, it can be combined with a predefined skill library or an execution system to achieve real-world deployment. An ideal workflow is shown in Fig 4.

REFERENCES

[1] J. Achiam, S. Adler, S. Agarwal, L. Ahmad, I. Akkaya, F. L. Aleman, D. Almeida, J. Altenschmidt, S. Altman, S. Anadkat, *et al.*, "Gpt-4 technical report," *arXiv preprint arXiv:2303.08774*, 2023.

[2] D. Guo, D. Yang, H. Zhang, J. Song, R. Zhang, R. Xu, Q. Zhu, S. Ma, P. Wang, X. Bi, *et al.*, "Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning," *arXiv preprint arXiv:2501.12948*, 2025.

[3] M. Ahn, A. Brohan, N. Brown, Y. Chebotar, O. Cortes, B. David, C. Finn, C. Fu, K. Gopalakrishnan, K. Hausman, *et al.*, "Do as i can, not as i say: Grounding language in robotic affordances," *arXiv preprint arXiv:2204.01691*, 2022.

[4] D. Driess, F. Xia, M. S. Sajjadi, C. Lynch, A. Chowdhery, A. Wahid, J. Tompson, Q. Vuong, T. Yu, W. Huang, *et al.*, "Palm-e: An embodied multimodal language model," 2023.

[5] C. H. Song, J. Wu, C. Washington, B. M. Sadler, W.-L. Chao, and Y. Su, "Llm-planner: Few-shot grounded planning for embodied agents with large language models," in *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 2998–3009, 2023.

[6] M. J. Kim, K. Pertsch, S. Karamcheti, T. Xiao, A. Balakrishna, S. Nair, R. Rafailov, E. Foster, G. Lam, P. Sanketi, *et al.*, "Openvla: An open-source vision-language-action model," *arXiv preprint arXiv:2406.09246*, 2024.

[7] A. Brohan, N. Brown, J. Carbajal, Y. Chebotar, X. Chen, K. Choromanski, T. Ding, D. Driess, A. Dubey, C. Finn, *et al.*, "Rt-2: Vision-language-action models transfer web knowledge to robotic control," *arXiv preprint arXiv:2307.15818*, 2023.

[8] H. Chen, Y. Niu, K. Hong, S. Liu, Y. Wang, Y. Li, and K. R. Driggs-Campbell, "Predicting object interactions with behavior primitives: An application in stowing tasks," in *7th Annual Conference on Robot Learning*, 2023.

[9] N. Yokoyama, S. Ha, D. Batra, J. Wang, and B. Bucher, "Vlfm: Vision-language frontier maps for zero-shot semantic navigation," in *2024 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 42–48, IEEE, 2024.

[10] P. Liu, Y. Orru, J. Vakil, C. Paxton, N. M. M. Shafiullah, and L. Pinto, "Ok-robot: What really matters in integrating open-knowledge models for robotics," *arXiv preprint arXiv:2401.12202*, 2024.

[11] Z. Fu, T. Z. Zhao, and C. Finn, "Mobile aloha: Learning bimanual mobile manipulation with low-cost whole-body teleoperation," *arXiv preprint arXiv:2401.02117*, 2024.

[12] S. Liu, A. Hasan, K. Hong, R. Wang, P. Chang, Z. Mizrachi, J. Lin, D. L. McPherson, W. A. Rogers, and K. Driggs-Campbell, "Dragon: A dialogue-based robot for assistive navigation with visual language grounding," *IEEE Robotics and Automation Letters*, vol. 9, no. 4, pp. 3712–3719, 2024.

[13] D. Song, J. Liang, A. Payandeh, A. H. Raj, X. Xiao, and D. Manocha, "Vlm-social-nav: Socially aware robot navigation through scoring using vision-language models," *IEEE Robotics and Automation Letters*, 2024.

[14] S. Peng, H. Chen, and K. Driggs-Campbell, "Towards uncertainty unification: A case study for preference learning," *arXiv preprint arXiv:2503.19317*, 2025.

[15] H. Wang, N. Chin, G. Gonzalez-Pumariega, X. Sun, N. Sunkara, M. A. Pace, J. Bohg, and S. Choudhury, "Apricot: Active preference learning and constraint-aware task planning with llms," *arXiv preprint arXiv:2410.19656*, 2024.

[16] J. Wu, R. Antonova, A. Kan, M. Lepert, A. Zeng, S. Song, J. Bohg, S. Rusinkiewicz, and T. Funkhouser, "Tidybot: Personalized robot assistance with large language models," *Autonomous Robots*, vol. 47, no. 8, pp. 1087–1102, 2023.

[17] R. Sinha, A. Elhafsi, C. Agia, M. Foutter, E. Schmerling, and M. Pavone, "Real-time anomaly detection and reactive planning with large language models," *arXiv preprint arXiv:2407.08735*, 2024.

[18] H. Xiong, R. Mendonca, K. Shaw, and D. Pathak, "Adaptive mobile manipulation for articulated objects in the open world," *arXiv preprint arXiv:2401.14403*, 2024.

[19] H. Liu, Y. Zhang, V. Betala, E. Zhang, J. Liu, C. Ding, and Y. Zhu, "Multi-task interactive robot fleet learning with visual world models," *arXiv preprint arXiv:2410.22689*, 2024.

[20] D. Li, T. Cai, T. Tang, W. Chai, K. R. Driggs-Campbell, and G. Wang, "Emmoe: A comprehensive benchmark for embodied mobile manipulation in open environments," *arXiv preprint arXiv:2503.08604*, 2025.

[21] P. Liu, Z. Guo, M. Warke, S. Chintala, C. Paxton, N. M. M. Shafiullah, and L. Pinto, "Dynamem: Online dynamic spatio-semantic memory for open world mobile manipulation," *arXiv preprint arXiv:2411.04999*, 2024.

[22] Z. Yan, S. Li, Z. Wang, L. Wu, H. Wang, J. Zhu, L. Chen, and J. Liu, "Dynamic open-vocabulary 3d scene graphs for long-term language-guided mobile manipulation," *arXiv preprint arXiv:2410.11989*, 2024.

[23] X. Zhang, Y. Zhang, D. Long, W. Xie, Z. Dai, J. Tang, H. Lin, B. Yang, P. Xie, F. Huang, *et al.*, "mgte: Generalized long-context text representation and reranking models for multilingual text retrieval," *arXiv preprint arXiv:2407.19669*, 2024.

[24] A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark, *et al.*, "Learning transferable visual models from natural language supervision," in *International conference on machine learning*, pp. 8748–8763, PmLR, 2021.

[25] S. Bai, K. Chen, X. Liu, J. Wang, W. Ge, S. Song, K. Dang, P. Wang, S. Wang, J. Tang, *et al.*, "Qwen2. 5-vl technical report," *arXiv preprint arXiv:2502.13923*, 2025.