CASPER: Inferring Diverse Intents for Assistive Teleoperation with Vision Language Models

Huihan Liu[1], **Rutav Shah**[1], **Shuijing Liu**[1], **Jack Pittenger**[1], **Mingyo Seo**[1], **Yuchen Cui**[2], **Yonatan Bisk**[3], **Roberto Martín-Martín**[1], **Yuke Zhu**[1]

[1]The University of Texas at Austin [2]The University of California, Los Angeles [3] Carnegie Mellon University

Abstract: Assistive teleoperation, where control is shared between a human and a robot, enables efficient and intuitive human-robot collaboration in diverse and unstructured environments. A central challenge in real-world assistive teleoperation is for the robot to infer a wide range of human intentions from user control inputs and to assist users with correct actions. Existing methods are either confined to simple, predefined scenarios or restricted to task-specific data distributions at training, limiting their support for real-world assistance. We introduce CASPER, an assistive teleoperation system that leverages commonsense knowledge embedded in pre-trained visual language models (VLMs) for real-time intent inference and flexible skill execution. CASPER incorporates an open-world perception module for a generalized understanding of novel objects and scenes, a VLM-powered intent inference mechanism that leverages commonsense reasoning to interpret snippets of teleoperated user input, and a skill library that expands the scope of prior assistive teleoperation systems to support diverse, long-horizon mobile manipulation tasks. Extensive empirical evaluation, including human studies and system ablations, demonstrates that CASPER improves task performance, reduces human cognitive load, and achieves higher user satisfaction than direct teleoperation and assistive teleoperation baselines. More information is available at https://ut-austin-rpl.github.io/casper/

Keywords: Assistive Teleoperation, Mobile Manipulation

1 Introduction

Deploying robots in human-centric settings like households requires balancing robot autonomy with humans' sense of agency [1, 2, 3, 4, 5, 6]. Full teleoperation offers users fine-grained control but imposes a high cognitive load, whereas fully autonomous robots act independently but often misalign their actions with nuanced human needs. **Assistive teleoperation** — a paradigm in which both the human and the robot share control [7, 8, 9, 10] — has thus emerged as an ideal middle ground. By keeping the user in control of high-level decisions while delegating low-level actions to the autonomous robot, this approach both preserves user agency and enhances overall system performance. As such, assistive teleoperation is becoming a desirable paradigm for robots to serve as reliable partners in human-centric environments, such as assisting individuals with motor impairments [11, 12].

While promising, assistive teleoperation in everyday environments remains challenging. A long-standing challenge in assistive teleoperation is to infer human intents from user control inputs and assist users with correct actions [8]. This challenge is amplified in real-world settings, where robots must go beyond closed-set intent prediction [13, 14] to handle diverse, open-ended user goals across different contexts and scenes. As a result, a key capability the robot should possess is to interpret user control inputs within the visual context and infer intent through commonsense reasoning. For

9th Conference on Robot Learning (CoRL 2025), Seoul, Korea.

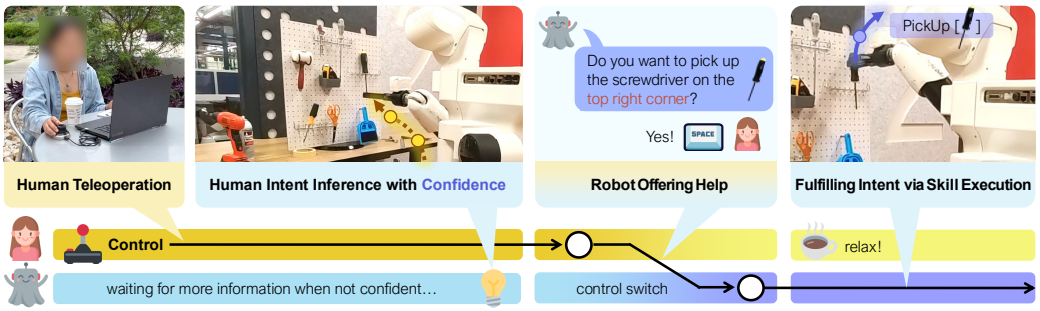

Figure 1: **CASPER infers user intents and offers help when confident.** Given user teleoperation input, CASPER uses VLMs to predict human intent using commonsense reasoning. Upon user confirmation, CASPER performs autonomous execution to fulfill the intent using a skill library. CASPER's background reasoning runs in parallel with foreground human control to minimize disruption.

example, consider a user teleoperating a robot to move a jar of pasta toward both a laptop and a cooking pot. Even if the pasta jar is closer to the laptop, commonsense suggests that the user intends to pour pasta into the pot, not onto the laptop. As another example, some users push an automatic door to open it, while others want to press an accessibility button. These examples illustrate the nuanced and context-dependent nature of human intent, highlighting the level of commonsense reasoning required for robots to provide effective and satisfactory assistance.

Existing assistive teleoperation systems often fall short in inferring diverse intents. Prior methods often limit the problem space to a closed set of objects [14, 9], or to a predefined task like picking up objects, implicitly assuming the intent type is known a priori [14, 13]. These intent inference methods, either based on rule-driven strategies [15, 13] or learned from demonstrations [16, 17, 14, 10], are typically limited to one single skill type or bound by the task distributions at training, struggling to generalize in new scenarios. Critically, these systems usually lack commonsense reasoning, which is essential for interpreting contextual cues and generalizing intent inference to novel scenes and behaviors in real-world environments.

To address the above limitations, we introduce CASPER, an assistive teleoperation system that infers diverse intents from human user control and offers assistance with long-horizon mobile manipulation tasks (Fig. 1). CASPER builds on three core components. First, it features an open-world perception module that uses pre-trained visual language models (VLMs) to provide a generalized understanding of open-world objects and scenes without task-specific training. Second, CASPER leverages VLM-powered commonsense reasoning to infer a diverse range of user intents, significantly expanding the possible intent choices compared with prior systems. Third, to realize task execution, CASPER uses a flexible library of parameterized skills encompassing a range of navigation and contact-rich manipulation behaviors [18]. With this comprehensive and composable skill library, CASPER can execute long-horizon tasks that go beyond the capabilities of traditional assistive teleoperation systems.

Furthermore, deploying the system for long-horizon tasks introduces a user-centric consideration: offering undesirable assistance based on premature intent inference can frustrate or disrupt the user. To avoid this, the system should determine intents only after gathering enough information from user inputs and visual contexts. CASPER addresses this by shadowing the user: it observes foreground human actions and infers user intents in the background. A confidence module based on self-consistency [19] ensures that assistance is triggered only when prediction confidence is high, reducing errors and user disruption. By running VLM-based inference in parallel with user control, CASPER unobtrusively predicts intent and prepares actions.

To evaluate the effectiveness of CASPER in assisting human users, we conduct extensive user studies on a mobile manipulator (TIAGo [20]), involving 10 pilot study participants and 13 study participants, totaling over 80 hours of interaction across 3 long-horizon tasks. Additionally, we conduct offline experiments to test the intent inference module and perform detailed performance analyses and ablation studies. Compared with prior assistive teleoperation baselines without commonsense

reasoning ability and a full teleoperation baseline, CASPER achieves a higher success rate, better user satisfaction, and lower cognitive load of users across all tasks.

2 Related Work

Assistive Teleoperation. Assistive teleoperation offers a promising balance between human control and robotic assistance, enhancing user agency and task efficiency [11, 12, 15, 21, 22, 23]. Assistive teleoperation enables users to share control with the robot, injecting their intent to guide the system toward their goals [8, 7, 10, 24, 25, 26, 27]. Accurately predicting user intent is thus a key challenge [8, 28, 29]. Prior approaches typically select the most probable intent from a fixed set of goals [8, 30, 31, 32, 33], assume a single predefined skill [13], or use data-driven methods to map high-dimensional user inputs to low-dimensional actions within specific tasks [10, 14, 16, 17, 24, 34, 35, 36, 37]. However, both approaches struggle to generalize beyond predefined intents without retraining or reprogramming. Moreover, they also lack the commonsense reasoning capability to interpret human control input within the visual context.

Human Intent Inference. Inferring hidden human states is a critical step toward understanding human behavior for a wide range of downstream tasks [38, 39, 40, 41, 42]. In robotics, intent inference enables robots to operate effectively in human-centered environments [43, 44, 45, 46]. To achieve shared goals, robots must reason about a human collaborator's latent strategy [47, 48], future actions [46, 49, 43], goals [50, 45, 23], and preferences [51, 52] to adjust their behavior accordingly. CASPER advances these efforts by leveraging VLM-based intent inference to facilitate assistance in assistive teleoperation settings.

LLMs and VLMs for Robotics. Foundation models, pretrained on internet-scale data, have gained attention for their strong generalization and adaptability across diverse applications [53]. They hold promise for enhancing the full robotics stack, from perception to decision-making and control [54]. Recent works integrate LLMs and VLMs as high-level planners paired with low-level skills to enable open-vocabulary and open-world robot capabilities [18, 55, 56, 57, 58, 59]. Other studies use LLMs to model humans [60], estimate uncertainty [61], or use language [62, 63, 23]. However, these approaches do not address the interpretation of user control inputs in real-world assistive teleoperation settings. Thus, the potential of LLMs/VLMs for assistive teleoperation remains underexplored.

3 Assistive Teleoperation with CASPER

In this section, we describe CASPER, an assistive teleoperation system that enables robots to infer and execute diverse human intents (Fig. 2). CASPER comprises two key components: an intent inference module that continuously predicts human intent from teleoperation history when shadowing the user in the background, and a skill execution module that executes tasks using a library of skills.

3.1 Problem Formulation

We formulate assistive teleoperation as a sequential decision-making problem defined by the tuple $\langle \mathcal{S}, \mathcal{A}, \mathcal{P}, \mathcal{Z} \rangle$, where $\mathcal{S}$ is the state space, $\mathcal{A}$ is the action space, $\mathcal{P} : \mathcal{S} \times \mathcal{A} \to \mathcal{S}$ is the unobserved transition function, and $\mathcal{Z}$ is the intent space. The state $s \in \mathcal{S}$ comprises the robot's RGB image observation, proprioceptive states (e.g., gripper status, base and end-effector poses), and a list of foreground objects $O = \{o_1, ..., o_n\}$ detected from the open-world perception module. The action $a \in \mathcal{A}$ is either from the human ($a = a_h$) during human teleoperation or from the robot ($a = a_r$) during autonomous execution. We assume that each assistive teleoperation episode is a sequence of one-step subtasks, and users can teleoperate to express their desired goals. We define a human intent for the i-th subtask as $z_i = (l_z^i, o_z^i) \in \mathcal{Z}$, where $l_z^i \in L$ is the intended skill (e.g., "navigate") and $o_z^i \in O$ is the target object (e.g., "the door" in "navigate to the door"). At the start of subtask i, the user provides a teleoperation trajectory snippet $\xi_h^T = (a_h^1, ..., a_h^T)$, where T is the snippet length. The goals of CASPER are to infer the human intent z_i from ξ_h^T, and to fulfill the intent with a trajectory ξ_r. This process repeats until the human indicates the end of the episode.

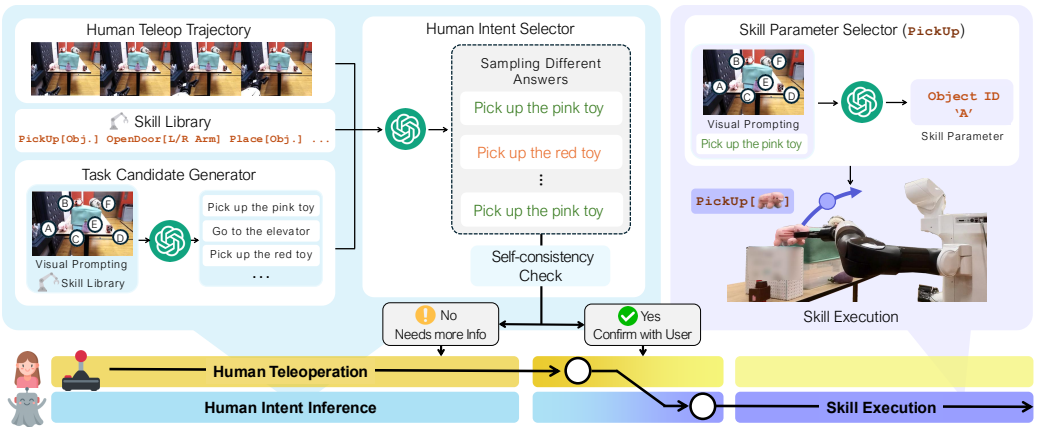

Figure 2: **CASPER architecture.** VLM-based intent inference runs in parallel with human teleoperation. CASPER generates task candidates from observations and infers intent from user inputs among the task candidates, repeating until predictions are self-consistent. Once confirmed by the user, CASPER executes the corresponding skill with estimated parameters.

3.2 Inferring Intents in the Background

CASPER tackles two key challenges in intent inference. To identify intent, it generates possible candidates from open-world observations and selects the most likely one based on commonsense understanding of user inputs. To handle intent ambiguity, it uses confidence estimation to predict only when confident, reducing premature suggestions.

Intent Candidates Generation. To generate an open set of potential intent options, we use a VLM $f_{candidate}$ to analyze the current state s^t and create a set of intent candidates $\{c_1, ..., c_m\}$ (Fig. 2 left). It first identifies actionable objects and then filters feasible object-skill pairs based on how each object is likely to be interacted with. The VLM adapts its predictions to object affordances and the robot's current state (e.g., avoiding "place" actions when the gripper is empty) by reasoning about robot-object interactions in a zero-shot manner. The commonsense-based intent set generation ensures that intent choices are semantically plausible and relevant to the scene.

Human Intent Selection. Given a set of task candidates $\{c_1, ..., c_m\}$, a second VLM f_{intent} predicts the user's intent $\hat{z}$ by analyzing a history of subsampled robot observations, which include downsized images, robot base and end-effector poses (Fig. 2 middle). It chooses the most likely intent $\hat{z}$ among $\{c_1, ..., c_m\}$, and parse the corresponding skill class $l_{\hat{z}}$. To enhance VLM understanding in cluttered scenes, we apply visual prompting [64, 18, 65] to annotate important regions that the VLM should attend to. These annotations include Set-of-Marks (SoM) [66] for segmented objects, gripper masks that highlight gripper position, and arrows indicating gripper motion history.

VLM Confidence Estimation. Real-time intent inference is inherently uncertain due to the ambiguity or incompleteness of human actions. For instance, if a user begins rotating a robot's base in a room with multiple furniture pieces, the intended target remains ambiguous until the user clearly moves the robot toward a specific furniture. Seeking for user confirmation based on a premature guess can disrupt user control and cause frustration. To address this, CASPER employs a confidence-based intent validation mechanism. Inspired by self-consistency methods [19] in LLMs, we run multiple VLM calls in parallel to estimate the confidence of intent predictions. The system only offers assistance when the number of VLM outputs in agreement exceeds a threshold. Formally, let K denote the number of VLM calls and $\hat{z}^k$ the intent predicted by the k-th VLM. The system confirms its prediction with the user if $\sum_{k=1}^{K} \mathbb{I}(\hat{z}^k = \hat{z}^{\text{mode}}) \geq \eta$, where $\mathbb{I}(\cdot)$ is the indicator function, $\hat{z}^{\text{mode}}$ is the most frequent prediction, and η is the agreement threshold. By filtering out low-confidence predictions, this module minimizes disruptions and premature predictions.

Parallel Foreground-Background System Design. Integrating pre-trained VLMs into real-time closed-loop control poses challenges due to the latency in VLM inference. Waiting for VLM out-

Figure 3: **Toy, Shelf, and Door: multi-step mobile manipulation tasks.** At each step, the robot disambiguates user intent among multiple plausible goals, selecting the correct one based on user inputs and visual context.

puts can be frustrating for users, especially when the system is uncertain or incorrect. To mitigate this delay, we adopt a framework where the user operates the robot in the foreground, while the VLM processes inputs simultaneously in the background. If the VLM is still processing or lacks confidence, it remains silent, intervening only when it has a confident prediction. This approach allows the user to operate naturally while the system continuously refines its intent inference.

3.3 Fulfilling Intents with Skill Execution

Once confidence in its prediction, CASPER executes the intent using a library of parametrized skills, with a VLM estimating the skill parameters for execution.

Control Switching. When confident in its prediction, the robot communicates the suggested action via an audible cue. The user can confirm or deny the prediction by pressing different keys on the keyboard. If confirmed, the system signals the transition to autonomous execution with another cue ("Great! I will take over."). If denied, the system prompts the user to continue teleoperation ("Understood, I'll pause here. Feel free to continue.") until the next prediction attempt.

Parametrized Skill Library. In real-world assistive settings, users may require help with long-horizon tasks that involve diverse manipulation and navigation behaviors. CASPER utilizes a library of parameterized skills that cover common mobile manipulation behaviors, including object manipulation skills (e.g., picking, placing, pouring), interactions with the environment (e.g., pushing doors, tapping card readers, pressing buttons, taking elevators), and navigation (e.g., approaching landmarks). Each skill is defined by a behavior primitive (e.g., `PickUp[Obj.]`) and a parameter (e.g., the target object's pose), enabling flexible execution of user intents across diverse environments. Refer to Appendix A.1 for a complete list of skills.

Skill Parameter Selection and Execution. Once a predicted intent $\hat{z}$ is confirmed (e.g., pouring pasta into a pot), the corresponding skill $l_{\hat{z}}$ (e.g., pouring) is called. The parameter estimation VLM f_{skill} identifies parameters such as the target object $o_{\hat{z}}$. Based on the object's pose, the skill execution module executes the skill. After completing the subtask, the robot prompts the user to resume control ("Alright, you can take over now.") for the next intent.

4 Experiments

We seek to answer the following research questions: **RQ1**: Does CASPER improve task performance and user experience compared to existing methods? **RQ2**: Is commonsense VLM reasoning essential for inferring diverse intents? **RQ3**: What is the contribution of each system component to overall performance? We address RQ1 through a user study, RQ2 via offline unit testing of the intent inference module, and RQ3 through ablation experiments.

4.1 User Study: Real-World Mobile Manipulation Tasks

Experiment Setup. We use a TIAGo mobile manipulator equipped with dual arms, a mobile base, and an RGBD camera. Users teleoperate the robot using a 3Dconnexion SpaceMouse while observing livestreamed RGB images. CASPER uses GPT-4o as its VLM backbone. The full teleoperation interface details, sensory setup, and audio/keyboard interaction design are provided in Appendix B.1.

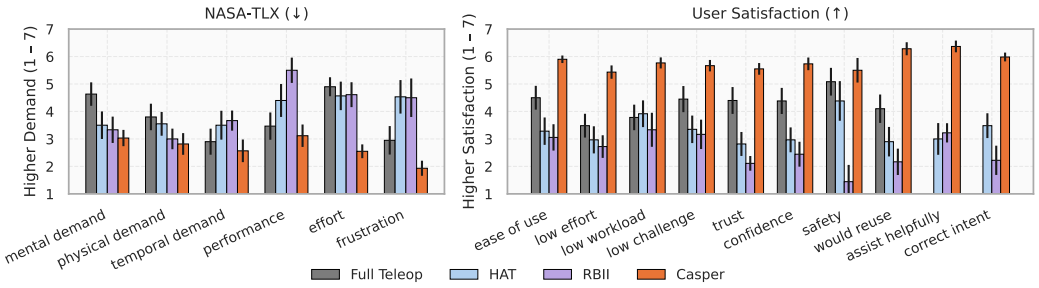

Figure 4: **User study: user workload and user satisfaction.** CASPER consistently outperforms the baselines in terms of user workload (left) and user satisfaction (right) with statistical significance ($p < 0.05$). Detailed per-task results and full questions of user satisfaction can be found in Appendix C. Note that for user satisfaction scores, "assist helpfully" and "correct intent" are not applicable to Full Teleop.

Tasks. We evaluate on 3 tasks (Fig. 3) each requiring multi-step intent inference: *Shelf* (3-step), *Toy* (5-step), and *Door* (2-step, 3-variations). Each step offers multiple plausible choices, requiring the system to use user input to infer intents. More task details are in Appendix B.2.

Participants and Procedures. We conducted an IRB-approved user study with $N = 13$ participants (mean age $= 29.4$; 5 females, 8 males; all able-bodied) , all of whom gave informed consent. Participants completed a practice session before completing each method in randomized order. After each, they answered user satisfaction and NASA-TLX questionnaires.

Independent Variables (Robot Control Methods). We compare CASPER with three baselines: 1) *Full Teleop*: The user manually teleoperates the robot without autonomous robot control. 2) *HAT* [15]: assistive teleoperation that infers human intents using proximity to goal. 3) *RBII* [9]: assistive teleoperation that infers human intents using Bayesian inference using temporal user input history. Since HAT and RBII only support grasping, we use CASPER to predict the skill and let the baselines select the target object, making comparisons conservative in their favor. These baselines test the role of commonsense reasoning in diverse intent inference.

Task Success Rate (% ↑)				
	Full Teleop	HAT	RBII	CASPER
Shelf	75.0	8.3	44.4	**83.3**
Toy	79.2	37.5	33.3	**91.2**
Door	75.0	75.0	57.1	**91.2**
Average	76.4	40.3	45.0	**88.9**

Task Completion Time (s ↓)				
	Full Teleop	HAT	RBII	CASPER
Shelf	256.5	225.0	252.4	**196.1**
Toy	391.2	406.3	388.3	**362.5**
Door	120.4	112.6	118.7	**96.8**
Average	256.0	248.0	253.1	**218.5**

Table 1: **User study: task success rate and completion time.** CASPER outperforms baselines in both task success and completion time.

Dependent Measures (Evaluation Metrics). To evaluate task performance, we measure the binary task success rate (completion in a fixed time limit). We measure human workload with NASA-TLX [67], a standard tool for evaluating subjective cognitive and physical workload. User satisfaction is measured with a questionnaire adapted from prior work [16]. We perform pairwise t-tests between CASPER with baselines to evaluate statistical significance.

Hypotheses. The user study tests the following hypothesis:

- **H1**: CASPER's VLM-driven intent inference and skill execution improve task performance over baselines in real-world assistive tasks;
- **H2**: CASPER reduces user workload and improves user satisfaction compared to baselines.

Results. *Task Performance.* CASPER exhibits significant improvements ($p < 0.05$) in task success rate compared to all baselines (see Table 1). The high success rate reflects the system's ability to infer intents and execute appropriate actions, even in complex scenarios and long-horizon tasks. Full Teleop is the runner-up in terms of success, allowing a portion of participants to succeed with expertise and patience. In contrast, HAT and RBII have lower success rates because they struggle with tasks requiring context or commonsense knowledge. We also report task completion time in

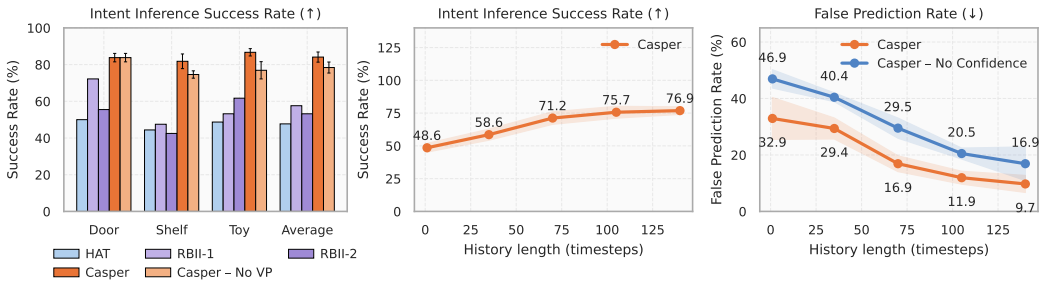

Figure 5: **Quantitative results from unit testing and ablation studies. Left**: CASPER outperforms all baselines in intent inference success rate. Note that no STD is reported for deterministic baselines. The ablation of Casper vs. Capser - No Visual Prompting (VP) highlights the benefit of visual prompting. **Middle**: Success rates improve with longer teleoperation history. **Right**: Removing confidence estimation increases false prediction rates across all history lengths.

Table 1, where CASPER is the lowest across all tasks. Full Teleop is slower due to manual high-precision control (e.g., pouring); the heuristic baselines suffer frequent errors and corrections.

NASA-TLX. Fig. 4 (**Left**) shows that CASPER significantly outperforms ($p < 0.05$) Full Teleop on all NASA-TLX metrics except "performance", indicating that autonomous skill execution lowers cognitive and physical workload. Full Teleop requires continuous user input, resulting in higher workloads. The lack of statistical significance in "performance" suggests that user *perceived* success is sensitive to CASPER occasional inference errors, despite CASPER's *objective* higher success rate. CASPER also significantly outperforms ($p < 0.05$) HAT in all metrics except in "mental demand" and "physical demand," and significantly outperforms ($p < 0.05$) RBII across all metrics. The increased workload in HAT and RBII results from more frequent prediction errors (e.g., predicting to pick up the table), leading to longer time and higher effort. The results indicate that VLM-powered intent inference and skill execution reduce user burden and improve usability. The lack of statistical significance in "mental demand" and "physical demand" likely stems from assistive baselines sharing skill execution module with CASPER, which reduces the difference in these measures.

User Satisfaction. Fig. 4 (**Right**) shows that CASPER has statistically significant improvements ($p < 0.05$) in all 10 user satisfaction metrics over all baselines. The results indicate that CASPER simplifies the assistance process and enhances the user experience. The Full Teleop baseline has lower scores, especially in "effort" and "physical workload" due to the demands of constant manual control. HAT and RBII score lower in "confidence" and "trust", as frequent intent prediction errors reduce user trust, significantly impacting overall user satisfaction.

In summary, the user study confirms that CASPER improves task performance (**H1**), reduces cognitive workload, and increases user satisfaction (**H2**). Beyond the main findings, the user study further reveals several notable insights which we detail in Appendix C, including more *detailed analysis* of results, *participant interviews*, a *demographic breakdown*, and an analysis of *failure cases*.

4.2 Unit Testing: Intent Inference Accuracy

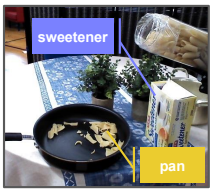

Figure 6: **Unit testing visualization.**

To quantitatively validate CASPER's intent inference accuracy, we conduct unit testing on teleoperation segments collected for each subtask across all three tasks. Each segment serves as an independent data point for evaluating intent inference, where success requires correctly predicting both the intended skill and target object. We prompt the VLM to predict the intended intent for each data point and compute the overall intent inference success rate, isolating intent inference from task execution. In this experiment, we compare CASPER against the HAT and RBII baselines. We evaluate two variants of RBII from the original paper: RBII-1's only uses the gripper-to-goal distance for recursive Bayesian inference, while RBII-2 also uses user joystick inputs with Boltzmann-rational action model. Note that RBII-1 was used in the user study due to the similar average performance between

RBII-1 and RBII-2. As shown in Fig. 5 **(Left)**, CASPER outperforms HAT and RBII baselines. Without commonsense reasoning, the baselines often mispredict targets by relying on gripper motion trends toward nearby objects, e.g., incorrectly pouring pasta into a sweetener box (Fig. 6) because the gripper moved closer to it. In contrast, CASPER's VLM-based inference leverages commonsense knowledge to make accurate predictions, choosing the pan instead.

4.3 Ablation Studies

To assess the impact of CASPER's key components, we perform ablations on the following questions:

How does the VLM input design, like visual prompting, affect intent inference accuracy? To guide the VLM's attention more on user input changes and the manipulated object, we apply visual prompting (VP) by adding a gripper mask and an arrow of the robot gripper motion (rendered from proprioceptive states) on the image. Fig. 5 **(Left)** shows VP yields an average 5.7% boost, especially on Toy ($+9.8\%$). Removing VP hurts the success rate because the VLM must implicitly understand the robot end-effector movement trace. Nonetheless, CASPER's no-VP variant still outperforms all non-VLM baselines by $> 10\%$, confirming that the primary gains come from the VLM's commonsense reasoning; VP enhances its reasoning rather than providing decisive extra information.

How much human teleoperation history is needed for accurate intent inference? CASPER infers intent from a segment of the user's teleoperation trajectory. Short histories risk ambiguity and incorrect predictions, while long histories increase user effort. We investigate the tradeoff by studying the accuracy of intent inference across different trajectory lengths. We vary history length from $T = 4$ to $T = 140$ timesteps and measure intent inference accuracy, defined as correctly predicting both the skill and target object. As shown in Fig. 5 **(Middle)**, longer histories improve accuracy by providing more context. However, gains plateau beyond $T = 100$, offering diminishing returns while adding user burden. Thus, we use $T = 100$ in the user study to balance accuracy and workload.

How does confidence estimation mitigate incorrect intent predictions? We hypothesize that CASPER's confidence estimation module reduces false predictions by filtering out ambiguous cases. To validate this, we ablate the module and compare false prediction rates (defined as incorrect predictions over total predictions). As shown in Fig. 5 **(Right)**, CASPER with uncertainty estimation consistently achieves lower false prediction rates, showing that uncertainty estimation effectively defers predictions when intents are unclear. Fig. 7 also shows qualitative examples. At $T = 40$, CASPER withholds predictions in both tasks due to ambiguity: The viewpoint is still shifting in the "Go to the wooden floor" task, and the gripper movement is still unclear between the basket and bag in the "Place the toy" task. Premature inference could have led to incorrect predictions (e.g., selecting the wrong landmark or container). By $T = 100$, enough context enables correct predictions. These examples illustrate how delaying decisions in uncertain situations improves reliability.

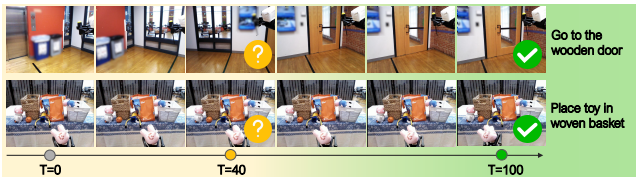

Figure 7: **Confidence estimation visualization.** CASPER predicts until the intent is clearer, ensuring more accurate assistance.

5 Conclusion

We presented CASPER, an assistive teleoperation system that addresses the challenge of intent inference for mobile manipulators in real-world environments. CASPER interprets human intents from teleoperation inputs by leveraging the commonsense reasoning capabilities of pre-trained VLMs, and features an open-world perception module, a flexible library of parameterized skills, and a parallel inference-execution architecture. Extensive user studies and system evaluations demonstrate that CASPER outperforms both direct teleoperation and assistive baselines in success rate, user satisfaction, and mental workload. Future work will explore continual learning of new skills from human

interactions [68, 69] and improving intent inference reliability with uncertainty quantification techniques such as conformal prediction [14, 70].

6 Limitations

CASPER has the following limitations. First, our user study included operators of different ages, skill levels, and teleoperation experience to test CASPER under varied interaction styles. However, to further validate the system's benefits, a necessary next step is to involve users with motor or cognitive disabilities [15, 22] and other underrepresented groups to cover a wider user distribution. Second, while the framework is compatible with the integration of continual learning of new skills [68, 71] from user interaction, that capability is not yet included in the scope of this paper and is left for future work. Lastly, CASPER assumes that the user's intent can be modeled with a combination of skills and target objects; finer-grained intents, such as precise motion paths, styles, or expressive behaviors [72], are not yet supported and will be pursued in future work.

Acknowledgments

We thank all participants in the human and pilot studies for their time and valuable contributions to our experiments. We thank Arpit Bahety, Gu-Cheol Jeong and Luca Macesanu for helping with Tiago hardware. We thank Ruta Desai, Roozbeh Mottaghi, Xavi Puig, Melanie Sclar and Changhao Wang for their fruitful discussions. This work was partially supported by the National Science Foundation (FRR-2145283, EFRI-2318065), the Office of Naval Research (N00014-24-1-2550), the DARPA TIAMAT program (HR0011-24-9-0428), and the Army Research Lab (W911NF-25-1-0065). It was also supported by the Institute of Information & Communications Technology Planning & Evaluation (IITP) grant funded by the Korean Government (MSIT) (No. RS-2024-00457882, National AI Research Lab Project).

References

[1] F. Cantucci and R. Falcone. Collaborative autonomy: Human–robot interaction to the test of intelligent help. *Electronics*, 11(19), 2022. ISSN 2079-9292. doi:10.3390/electronics11193065. URL https://www.mdpi.com/2079-9292/11/19/3065.

[2] S. A. Mostafa, M. S. Ahmad, and A. Mustapha. Adjustable autonomy: A systematic literature review. *Artificial Intelligence Review*, 51, 2019. doi:10.1007/s10462-017-9560-8.

[3] J. Beer, A. Fisk, and W. Rogers. Toward a framework for levels of robot autonomy in human-robot interaction. *Journal of Human-Robot Interaction*, 3:74, 06 2014. doi:10.5898/JHRI.3.2.Beer.

[4] M. A. Collier, R. Narayan, and H. Admoni. The sense of agency in assistive robotics using shared autonomy, 2025. URL https://arxiv.org/abs/2501.07462.

[5] J. D. Loehr. The sense of agency in joint action: An integrative review. *Psychonomic Bulletin & Review*, 29(4):1089–1117, 2022.

[6] W. Wen, Y. Kuroki, and H. Asama. The sense of agency in driving automation. *Frontiers in Psychology*, 10:2691, 2019.

[7] A. D. Dragan and S. S. Srinivasa. A policy-blending formalism for shared control. *The International Journal of Robotics Research*, 32:790 – 805, 2013. URL https://api.semanticscholar.org/CorpusID:18131716.

[8] A. D. Dragan and S. S. Srinivasa. Formalizing assistive teleoperation. In *Robotics: Science and Systems*, pages 73–80. Sydney, Australia, 2012.

[9] B. D. Argall. Autonomy in rehabilitation robotics: An intersection. *Annual Review of Control, Robotics, and Autonomous Systems*, 1(1):441–463, 2018.

[10] S. Chen, J. Gao, S. Reddy, G. Berseth, A. D. Dragan, and S. Levine. Asha: Assistive teleoperation via human-in-the-loop reinforcement learning, 2022. URL https://arxiv.org/abs/2202.02465.

[11] S. W. Brose, D. J. Weber, B. Salatin, G. G. Grindle, H. Wang, J. J. Vazquez, and R. A. Cooper. The role of assistive robotics in the lives of persons with disability. *American journal of physical medicine & rehabilitation*, 89 6:509–21, 2010. URL https://api.semanticscholar.org/CorpusID:38570901.

[12] D. P. Miller. Assistive robotics: An overview. In *Assistive Technology and Artificial Intelligence*, 1998. URL https://api.semanticscholar.org/CorpusID:2877042.

[13] A. Belsare, Z. Karimi, C. Mattson, and D. S. Brown. Toward zero-shot user intent recognition in shared autonomy, 2025. URL https://arxiv.org/abs/2501.08389.

[14] M. Zhao, R. Simmons, H. Admoni, and A. Bajcsy. Conformalized teleoperation: Confidently mapping human inputs to high-dimensional robot actions, 2024. URL https://arxiv.org/abs/2406.07767.

[15] A. Padmanabha, J. Gupta, C. Chen, J. Yang, V. Nguyen, D. J. Weber, C. Majidi, and Z. Erickson. Independence in the home: A wearable interface for a person with quadriplegia to teleoperate a mobile manipulator. In *Proceedings of the 2024 ACM/IEEE International Conference on Human-Robot Interaction*, HRI '24, page 542–551. ACM, Mar. 2024. doi:10.1145/3610977.3634964. URL http://dx.doi.org/10.1145/3610977.3634964.

[16] Y. Cui, S. Karamcheti, R. Palleti, N. Shivakumar, P. Liang, and D. Sadigh. No, to the right: Online language corrections for robotic manipulation via shared autonomy. In *Proceedings of the 2023 ACM/IEEE International Conference on Human-Robot Interaction*, HRI '23. ACM, Mar. 2023. doi:10.1145/3568162.3578623. URL http://dx.doi.org/10.1145/3568162.3578623.

[17] S. Karamcheti, M. Srivastava, P. Liang, and D. Sadigh. Lila: Language-informed latent actions. In A. Faust, D. Hsu, and G. Neumann, editors, *Proceedings of the 5th Conference on Robot Learning*, volume 164 of *Proceedings of Machine Learning Research*, pages 1379–1390. PMLR, 08–11 Nov 2022. URL https://proceedings.mlr.press/v164/karamcheti22a.html.

[18] R. Shah, A. Yu, Y. Zhu, Y. Zhu, and R. Martín-Martín. Bumble: Unifying reasoning and acting with vision-language models for building-wide mobile manipulation, 2024. URL https://arxiv.org/abs/2410.06237.

[19] X. Wang, J. Wei, D. Schuurmans, Q. Le, E. Chi, S. Narang, A. Chowdhery, and D. Zhou. Self-consistency improves chain of thought reasoning in language models, 2023. URL https://arxiv.org/abs/2203.11171.

[20] TIAGo - mobile manipulator robot. https://pal-robotics.com/robot/tiago.

[21] A. Padmanabha, J. Yuan, J. Gupta, Z. Karachiwalla, C. Majidi, H. Admoni, and Z. Erickson. Voicepilot: Harnessing llms as speech interfaces for physically assistive robots. In *Proceedings of the 37th Annual ACM Symposium on User Interface Software and Technology*, UIST '24, page 1–18. ACM, Oct. 2024. doi:10.1145/3654777.3676401. URL http://dx.doi.org/10.1145/3654777.3676401.

[22] R. K. Jenamani, P. Sundaresan, M. Sakr, T. Bhattacharjee, and D. Sadigh. Flair: Feeding via long-horizon acquisition of realistic dishes, 2024. URL https://arxiv.org/abs/2407.07561.

[23] 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*, 9(4):3712–3719, 2024.

[24] S. Karamcheti, A. J. Zhai, D. P. Losey, and D. Sadigh. Learning visually guided latent actions for assistive teleoperation. In A. Jadbabaie, J. Lygeros, G. J. Pappas, P. Parrilo, B. Recht, C. J. Tomlin, and M. N. Zeilinger, editors, *Proceedings of the 3rd Conference on Learning for Dynamics and Control*, volume 144 of *Proceedings of Machine Learning Research*, pages 1230–1241. PMLR, 07 – 08 June 2021. URL https://proceedings.mlr.press/v144/karamcheti21a.html.

[25] E. You and K. K. Hauser. Assisted teleoperation strategies for aggressively controlling a robot arm with 2d input. In *Robotics: Science and Systems*, 2011. URL https://api.semanticscholar.org/CorpusID:17514416.

[26] A. Broad, T. D. Murphey, and B. Argall. Learning models for shared control of human-machine systems with unknown dynamics. *ArXiv*, abs/1808.08268, 2017. URL https://api.semanticscholar.org/CorpusID:2559222.

[27] S. Javdani, H. Admoni, S. Pellegrinelli, S. S. Srinivasa, J. A. Bagnell, and S. J. Carnegie. Shared autonomy via hindsight optimization for teleoperation and teaming. *The International Journal of Robotics Research*, 37:717 – 742, 2017. URL https://api.semanticscholar.org/CorpusID:11336917.

[28] D. E. Gopinath and B. Argall. Active intent disambiguation for shared control robots. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 28:1497–1506, 2020. URL https://api.semanticscholar.org/CorpusID:216029050.

[29] G. Hoffman, T. Bhattacharjee, and S. Nikolaidis. Inferring human intent and predicting human action in human–robot collaboration. *Annual Review of Control, Robotics, and Autonomous Systems*, 7, 2024.

[30] H. Admoni and S. S. Srinivasa. Predicting user intent through eye gaze for shared autonomy. In *AAAI Fall Symposia*, 2016. URL https://api.semanticscholar.org/CorpusID:53307292.

[31] C. Brooks and D. Szafir. Balanced information gathering and goal-oriented actions in shared autonomy. In *2019 14th ACM/IEEE International Conference on Human-Robot Interaction (HRI)*, pages 85–94, 2019. doi:10.1109/HRI.2019.8673192.

[32] S. Nikolaidis, Y. X. Zhu, D. Hsu, and S. Srinivasa. Human-robot mutual adaptation in shared autonomy. In *Proceedings of the 2017 ACM/IEEE International Conference on Human-Robot Interaction*, HRI '17. ACM, Mar. 2017. doi:10.1145/2909824.3020252. URL http://dx.doi.org/10.1145/2909824.3020252.

[33] B. A. Newman, R. M. Aronson, S. S. Srinivasa, K. Kitani, and H. Admoni. Harmonic: A multimodal dataset of assistive human-robot collaboration, 2020. URL https://arxiv.org/abs/1807.11154.

[34] T. Yoneda, L. Sun, G. Yang, B. Stadie, and M. Walter. To the noise and back: Diffusion for shared autonomy, 2023. URL https://arxiv.org/abs/2302.12244.

[35] A. Jonnavittula, S. A. Mehta, and D. P. Losey. Learning to share autonomy from repeated human-robot interaction. *ArXiv*, abs/2205.09795, 2022. URL https://api.semanticscholar.org/CorpusID:248965455.

[36] M. Zurek, A. Bobu, D. S. Brown, and A. D. Dragan. Situational confidence assistance for lifelong shared autonomy. *2021 IEEE International Conference on Robotics and Automation (ICRA)*, pages 2783–2789, 2021. URL https://api.semanticscholar.org/CorpusID:233231464.

[37] C. B. Schaff and M. R. Walter. Residual policy learning for shared autonomy. *ArXiv*, abs/2004.05097, 2020. URL https://api.semanticscholar.org/CorpusID:215737315.

[38] B. Lai, S. Toyer, T. Nagarajan, R. Girdhar, S. Zha, J. M. Rehg, K. Kitani, K. Grauman, R. Desai, and M. Liu. Human action anticipation: A survey, 2024. URL https://arxiv.org/abs/2410.14045.

[39] E. V. Mascaro, H. Ahn, and D. Lee. Intention-conditioned long-term human egocentric action anticipation. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pages 6048–6057, 2023.

[40] H. Girase, H. Gang, S. Malla, J. Li, A. Kanehara, K. Mangalam, and C. Choi. Loki: Long term and key intentions for trajectory prediction, 2021. URL https://arxiv.org/abs/2108.08236.

[41] A. Rasouli, I. Kotseruba, T. Kunic, and J. Tsotsos. Pie: A large-scale dataset and models for pedestrian intention estimation and trajectory prediction. In *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 6261–6270, 2019. doi:10.1109/ICCV.2019.00636.

[42] B. Liu, E. Adeli, Z. Cao, K.-H. Lee, A. Shenoi, A. Gaidon, and J. C. Niebles. Spatiotemporal relationship reasoning for pedestrian intent prediction, 2020. URL https://arxiv.org/abs/2002.08945.

[43] Z. Huang, J. Pohovey, A. Yammanuru, and K. Driggs-Campbell. Lit: Large language model driven intention tracking for proactive human-robot collaboration – a robot sous-chef application, 2024. URL https://arxiv.org/abs/2406.13787.

[44] H. Ali, P. Allgeuer, and S. Wermter. Comparing apples to oranges: Llm-powered multimodal intention prediction in an object categorization task, 2024. URL https://arxiv.org/abs/2404.08424.

[45] Z. Huang, Y.-J. Mun, X. Li, Y. Xie, N. Zhong, W. Liang, J. Geng, T. Chen, and K. Driggs-Campbell. Hierarchical intention tracking for robust human-robot collaboration in industrial assembly tasks, 2023. URL https://arxiv.org/abs/2203.09063.

[46] S. Liu, P. Chang, Z. Huang, N. Chakraborty, K. Hong, W. Liang, D. L. McPherson, J. Geng, and K. Driggs-Campbell. Intention aware robot crowd navigation with attention-based interaction graph. In *IEEE International Conference on Robotics and Automation (ICRA)*, pages 12015–12021, 2023.

[47] C. Wang, C. Pérez-D'Arpino, D. Xu, L. Fei-Fei, C. K. Liu, and S. Savarese. Co-gail: Learning diverse strategies for human-robot collaboration. In *Proceedings of the 5th Conference on Robot Learning (CoRL)*, pages 1279–1290, 2021.

[48] S. Liu, P. Chang, H. Chen, N. Chakraborty, and K. Driggs-Campbell. Learning to navigate intersections with unsupervised driver trait inference. In *IEEE International Conference on Robotics and Automation (ICRA)*, 2022.

[49] H. Wang, K. Kedia, J. Ren, R. Abdullah, A. Bhardwaj, A. Chao, K. Y. Chen, N. Chin, P. Dan, X. Fan, G. Gonzalez-Pumariega, A. Kompella, M. A. Pace, Y. Sharma, X. Sun, N. Sunkara, and S. Choudhury. Mosaic: A modular system for assistive and interactive cooking, 2024.

[50] P. Chang, S. Liu, H. Chen, and K. Driggs-Campbell. Robot sound interpretation: Combining sight and sound in learning-based control. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2020.

[51] 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*, 2023.

[52] 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. In *Conference on Robot Learning*, 2024.

[53] R. Bommasani, D. A. Hudson, E. Adeli, R. Altman, S. Arora, S. von Arx, M. S. Bernstein, J. Bohg, A. Bosselut, E. Brunskill, E. Brynjolfsson, S. Buch, D. Card, R. Castellon, N. S. Chatterji, A. S. Chen, K. A. Creel, J. Davis, D. Demszky, C. Donahue, M. K. B. Doumbouya, E. Durmus, S. Ermon, J. Etchemendy, K. Ethayarajh, L. Fei-Fei, C. Finn, T. Gale, L. Gillespie, K. Goel, N. D. Goodman, S. Grossman, N. Guha, T. Hashimoto, P. Henderson, J. Hewitt, D. E. Ho, J. Hong, K. Hsu, J. Huang, T. F. Icard, S. Jain, D. Jurafsky, P. Kalluri, S. Karamcheti, G. Keeling, F. Khani, O. Khattab, P. W. Koh, M. S. Krass, R. Krishna, R. Kuditipudi, A. Kumar, F. Ladhak, M. Lee, T. Lee, J. Leskovec, I. Levent, X. L. Li, X. Li, T. Ma, A. Malik, C. D. Manning, S. Mirchandani, E. Mitchell, Z. Munyikwa, S. Nair, A. Narayan, D. Narayanan, B. Newman, A. Nie, J. C. Niebles, H. Nilforoshan, J. F. Nyarko, G. Ogut, L. J. Orr, I. Papadimitriou, J. S. Park, C. Piech, E. Portelance, C. Potts, A. Raghunathan, R. Reich, H. Ren, F. Rong, Y. Roohani, C. Ruiz, J. Ryan, C. R'e, D. Sadigh, S. Sagawa, K. Santhanam, A. Shih, K. P. Srinivasan, A. Tamkin, R. Taori, A. W. Thomas, F. Tramèr, R. E. Wang, W. Wang, B. Wu, J. Wu, Y. Wu, S. M. Xie, M. Yasunaga, J. You, M. A. Zaharia, M. Zhang, T. Zhang, X. Zhang, Y. Zhang, L. Zheng, K. Zhou, and P. Liang. On the opportunities and risks of foundation models. *ArXiv*, abs/2108.07258, 2021. URL https://api.semanticscholar.org/CorpusID:237091588.

[54] R. Firoozi, J. Tucker, S. Tian, A. Majumdar, J. Sun, W. Liu, Y. Zhu, S. Song, A. Kapoor, K. Hausman, et al. Foundation models in robotics: Applications, challenges, and the future. *The International Journal of Robotics Research*, 2023.

[55] P. Zhi, Z. Zhang, M. Han, Z. Zhang, Z. Li, Z. Jiao, B. Jia, and S. Huang. Closed-loop open-vocabulary mobile manipulation with gpt-4v. *ArXiv*, abs/2404.10220, 2024. URL https://api.semanticscholar.org/CorpusID:269157231.

[56] J. Liang, W. Huang, F. Xia, P. Xu, K. Hausman, B. Ichter, P. R. Florence, and A. Zeng. Code as policies: Language model programs for embodied control. *2023 IEEE International Conference on Robotics and Automation (ICRA)*, pages 9493–9500, 2022. URL https://api.semanticscholar.org/CorpusID:252355542.

[57] Z. Hu, F. Lucchetti, C. Schlesinger, Y. Saxena, A. Freeman, S. Modak, A. Guha, and J. Biswas. Deploying and evaluating llms to program service mobile robots. *IEEE Robotics and Automation Letters*, 9:2853–2860, 2023. URL https://api.semanticscholar.org/CorpusID:265294597.

[58] M. Ahn, A. Brohan, N. Brown, Y. Chebotar, O. Cortes, B. David, C. Finn, C. Fu, K. Gopalakrishnan, K. Hausman, A. Herzog, D. Ho, J. Hsu, J. Ibarz, B. Ichter, A. Irpan, E. Jang, R. J. Ruano, K. Jeffrey, S. Jesmonth, N. J. Joshi, R. Julian, D. Kalashnikov, Y. Kuang, K.-H. Lee, S. Levine, Y. Lu, L. Luu, C. Parada, P. Pastor, J. Quiambao, K. Rao, J. Rettinghouse, D. Reyes, P. Sermanet, N. Sievers, C. Tan, A. Toshev, V. Vanhoucke, F. Xia, T. Xiao, P. Xu, S. Xu, M. Yan, and A. Zeng. Do as i can, not as i say: Grounding language in robotic affordances, 2022.

[59] W. Huang, F. Xia, T. Xiao, H. Chan, J. Liang, P. R. Florence, A. Zeng, J. Tompson, I. Mordatch, Y. Chebotar, P. Sermanet, N. Brown, T. Jackson, L. Luu, S. Levine, K. Hausman, and B. Ichter. Inner monologue: Embodied reasoning through planning with language models. *ArXiv*, abs/2207.05608, 2022. URL https://api.semanticscholar.org/CorpusID:250451569.

[60] B. Zhang and H. Soh. Large language models as zero-shot human models for human-robot interaction. *2023 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 7961–7968, 2023. URL https://api.semanticscholar.org/CorpusID:257378614.

[61] A. Z. Ren, A. Dixit, A. Bodrova, S. Singh, S. Tu, N. Brown, P. Xu, L. Takayama, F. Xia, J. Varley, Z. Xu, D. Sadigh, A. Zeng, and A. Majumdar. Robots that ask for help: Uncertainty alignment for large language model planners, 2023. URL https://arxiv.org/abs/2307.01928.

[62] P. Chang, S. Liu, and K. Driggs-Campbell. Learning visual-audio representations for voice-controlled robots. In *IEEE International Conference on Robotics and Automation (ICRA)*, 2023.

[63] P. Chang, S. Liu, T. Ji, N. Chakraborty, K. Hong, and K. R. Driggs-Campbell. A data-efficient visual-audio representation with intuitive fine-tuning for voice-controlled robots. In *Conference on Robot Learning (CoRL)*, 2023.

[64] H. Bahng, A. Jahanian, S. Sankaranarayanan, and P. Isola. Exploring visual prompts for adapting large-scale models, 2022. URL https://arxiv.org/abs/2203.17274.

[65] S. Nasiriany, F. Xia, W. Yu, T. Xiao, J. Liang, I. Dasgupta, A. Xie, D. Driess, A. Wahid, Z. Xu, Q. Vuong, T. Zhang, T.-W. E. Lee, K.-H. Lee, P. Xu, S. Kirmani, Y. Zhu, A. Zeng, K. Hausman, N. Heess, C. Finn, S. Levine, and B. Ichter. Pivot: Iterative visual prompting elicits actionable knowledge for vlms, 2024. URL https://arxiv.org/abs/2402.07872.

[66] J. Yang, H. Zhang, F. Li, X. Zou, C. Li, and J. Gao. Set-of-mark prompting unleashes extraordinary visual grounding in gpt-4v, 2023. URL https://arxiv.org/abs/2310.11441.

[67] Wikipedia contributors. NASA-TLX — Wikipedia, The Free Encyclopedia, 2025. URL https://en.wikipedia.org/wiki/NASA-TLX. Accessed: 2025-01-24.

[68] J. Grannen, S. Karamcheti, S. Mirchandani, P. Liang, and D. Sadigh. Vocal sandbox: Continual learning and adaptation for situated human-robot collaboration. In *Conference on Robot Learning (CoRL)*, 2024.

[69] M. Parakh, A. Fong, A. Simeonov, T. Chen, A. Gupta, and P. Agrawal. Lifelong robot learning with human assisted language planners. In *IEEE International Conference on Robotics and Automation (ICRA)*, pages 523–529, 2024.

[70] A. Z. Ren, A. Dixit, A. Bodrova, S. Singh, S. Tu, N. Brown, P. Xu, L. Takayama, F. Xia, J. Varley, Z. Xu, D. Sadigh, A. Zeng, and A. Majumdar. Robots that ask for help: Uncertainty alignment for large language model planners. In *Conference on Robot Learning (CoRL)*, 2023.

[71] W. Wan, Y. Zhu, R. Shah, and Y. Zhu. Lotus: Continual imitation learning for robot manipulation through unsupervised skill discovery, 2024. URL https://arxiv.org/abs/2311.02058.

[72] K. Mahadevan, J. Chien, N. Brown, Z. Xu, C. Parada, F. Xia, A. Zeng, L. Takayama, and D. Sadigh. Generative expressive robot behaviors using large language models. In *Proceedings of the 2024 ACM/IEEE International Conference on Human-Robot Interaction*, HRI '24, page 482–491. ACM, Mar. 2024. doi:10.1145/3610977.3634999. URL http://dx.doi.org/10.1145/3610977.3634999.

[73] T. Ren, S. Liu, A. Zeng, J. Lin, K. Li, H. Cao, J. Chen, X. Huang, Y. Chen, F. Yan, Z. Zeng, H. Zhang, F. Li, J. Yang, H. Li, Q. Jiang, and L. Zhang. Grounded sam: Assembling open-world models for diverse visual tasks, 2024. URL https://arxiv.org/abs/2401.14159.

[74] S. Liu, Z. Zeng, T. Ren, F. Li, H. Zhang, J. Yang, Q. Jiang, C. Li, J. Yang, H. Su, J. Zhu, and L. Zhang. Grounding dino: Marrying dino with grounded pre-training for open-set object detection, 2024. URL https://arxiv.org/abs/2303.05499.

A Methods

A.1 Parametrized Skill Library.

CASPER has 8 parameterized skills, from navigation (`GoToLandmark`, `NavigateToPoint`) to object (`Pour`, `Pick`, `Place`) and environment interactions (`TapCard`, `PushDoor`, `PressButton`). The detailed skills and their descriptions are presented below:

```
skill_name: pick_up_object
arguments: object_of_interest
description: pick_up_object skill moves its arms to pick up the object
    specified in the argument object_of_interest. The pick_up_object
    skill can only pick up objects within the reach of its arms and does
    not control the robot base.

skill_name: place_object
arguments: object_of_interest
description: place_object skill moves its arms to place what it is
    holding to the object specified in the argument object_of_interest.
    The place_object skill can only place objects within the reach of
    its arms and does not control the robot base. The robot should place
    objects onto containers like trash cans, sinks, or items with
    surfaces like chairs, tables, etc.

skill_name: tap_card_open_door
arguments: None
description: Opens the door by tapping the key card access, if key card
    access is needed.

skill_name: goto_landmark
arguments: Selected landmark image from the environment from various
    options.
description: Navigates to the landmark in the environment, example,
    bedroom, kitchen, tool shop, etc.

skill_name: navigate_to_point_on_ground
arguments: object
description: Moves the robot to a point near the selected object. This
    skill can be used to move to a point in the room to perform a task,
    example, navigating near the toaster to make a toast. You should use
    this skill if there is still some distance between the robot and the
    object of interest, e.g. there is some area on the floor in between,
    or there is some hallway. Do not use this if the robot is near and
    can reach the object directly.

skill_name: push_open_door
arguments: None
description: Opens the door if the robot is in front of the door. The
    robot moves forward to open the door.

skill_name: pour_object
arguments: object_of_interest
description: pour_object skill moves its arms to pour whatever it is
    holding to the object (containers) specified in the argument
    object_of_interest. The pour_object skill can only pour to objects
    (containers) within the reach of its arms and does not control the
    robot base.

skill_name: press_button
arguments: button position depending which button you want to press.
description: Equips the robot with the capability of pressing a button.
    The robot will push the button selected in the argument. The subtask
    must indicate which button to press, example, 'Press the
    accessibility button to open the door.'
```

CASPER uses open-world perception module with GSAM [73] and GroundingDINO [74]. Skills are parameterized on object pose. Low-level actions are then generated via Inverse Kinematics for joint configuration (arm) and motion planning on a 2D occupancy map (base). Adding a new skill is straightforward with a text description for VLM, a parameter mapping, and a few optional joint waypoints.

A.2 Intent Inference Details.

VLM Prompts.

1) Prompt for Intent Candidates Generation

First, give a list of possible tasks to perform, using the information of the scene, the relevant objects, and relevant skills.
Note:
- The robot can only manipulate objects that are within 0.7 meters of the robot. If the distance from the robot to the object is greater than 0.7 meters, then you SHOULD NOT include the manipulation skills in the task choices!
- The pick up and place skill should be used on smaller objects, and the navigate skill should be used on furniture, like tables, chairs, etc. You should use the robot history: Eliminate the tasks that the robot has already performed. If the robot has picked up an object, it will not perform the task again!
Formulate your results in the format of multiple-choice questions.
Example 1: Given that I am farther away and the robot is moving, the possible subtasks to perform are:
A) Navigate to the desk with pens on top of it.
B) Navigate to the brown colored door.
Example 2: Given that I am near the table, the possible subtasks to perform are:
A) Place the apple in the pink bowl.
B) Pick up the screwdriver with blue handle.
Example 3: Given that I am near the table, the possible subtasks to perform are:
A) Pick up the blue bowl with pink stripes.
B) Pick up the apple.
D) Pick up the purple bowl.
Example 4: Given that I am in the corridor, the possible subtasks to perform are:
A) Go to the kitchen.
B) Go to the classroom.

2) Prompt for Human Intent Selection

INSTRUCTIONS:
You are given a sequence of images of the scene. The images are taken from the camera on a mobile robot that is moving its base. Your goal is to determine the robot's intent based on this sequence of robot observations. You want to make use of the list of skills, the history of the robot's movement, and the list of task choices to determine the human's goal.
The list of skills that the robot has are below. The tasks are using the skills listed here. {skill description}
HISTORY OF PAST EXECUTIONS: You should make use of this information for decision making.
{history prompt}
Possible task choices:
{subtask prompt}
Think step by step, keep in mind the following points:
1. Consider the given task choices.
2. Focus on the images, and see if there is a change in robot's point of view; see how it is moving and changing its position, or if the gripper is getting closer to one of the objects, or turning towards one of the landmarks.

If the robot gripper is moving, see where the gripper (as masked in the image) is moving towards based on the green arrow, and use that to determine the task choice option.
Then, given the images and the robot's movement, summarize the previous the robot's movement.
3. Then, summarize the previous executions made by the robot and feedback received from the human or environment.
Finally, answer: What is the robot trying to do? Choose from the list of possible task choices.
Example reasoning 1: The robot is moving towards the left, where there is a table with a bowl on it. Since it has already picked up an object, it most likely wants to place the object on the bowl. However, the distance to the table is farther for the robot (greater than 0.7 meters) to place the object. We should first navigate to the table with a bowl on it.
Example reasoning 2: The robot arm is moving closer towards the apple. The apple is already within the reach of the robot, that is, less than 0.7 meters. Therefore, it is likely that the robot will pick up the apple.
Example reasoning 3: The robot is moving towards the bookshelf with a book in its hand. It is most likely trying to place the book on the bookshelf. However, the robot is far away from the bookshelf. We should first navigate to the bookshelf.
Example reasoning 4: The robot is moving towards the table which has a book on it. The robot tried to pick up the book before, but it failed due to IK solver issues. Since the robot is far away from the table, we should first navigate to the table with a book on it using the navigate skill.
Example reasoning 5: The robot is near the book shelf which has one thriller and one comedy book. The robot tried to pick up the comedy book but the human stopped it. It is likely that the robot will try to pick up the thriller book.
Example reasoning 6: The robot is moving towards the bookshelf with a book in its hand. The robot tried to place the book on the book holder, but it failed due to IK solver issues. Since the robot is far away from the book holder, we should first navigate to the bookshelf using the navigate skill.
Example reasoning 7: The robot is holding a bottle in its hand. Given that there are several cups and containers on the table, It is likely that the robot will pour the liquid from the bottle into one of the cups or containers.
Provide the skill name in a valid JSON format. Your answer at the end in a valid JSON of this format: {{'subtask': '', 'skillname': ''}}
Avoid using the object id in the final JSON response. Describe the object(s) involved in the sub-task instead of using the object id in the JSON response. This is very important.
You should only choose from the list of task choices provided! This is very important.
If the arm is moving, you should see where the arm GRIPPER TIPS is moving towards, and use that to determine the task choice!! The gripper tip consists of 2 pointly black parts at the end of the gripper, don't consider the white part.
For example, if the gripper is mostly staying on the table level or below the table, then most likely the user is choosing objects on the table (i.e. the bottom row). Else if the gripper moves above the table and obstructs the objects on the table, then most likely the user is choosing something above the table on the top row.
You should judge the physical distance between the robot and the object. You can tell that by checking if there is some area (like ground, floor) in between robot and the object. If the robot is far away from the object, it will most likely perform NAVIGATION. It is unlikely that the robot will do tasks that involve touching the object if the robot is far away from the object, e.g. pick up, place, pressing button, tapping card, etc.
Pay attention to where the MASKED gripper is moving, and the direction of the arrow of the robot arm's movement!
The arrow means the direction of the gripper's movement. For example, if the arrow is pointing up and right, it means the gripper is moving to something on the top right.
ANSWER: Let's think step by step.

B Experiments Details

B.1 Details on Experiment Setup

We utilize the TIAGo mobile manipulator robot as our platform, which features dual robotic arms and a mobile base. The robot is equipped with the following sensors and components: (1) a 3Dconnexion SpaceMouse, serving as the teleoperation interface that enables users to convey their intents through manual control (2) An RGB-D camera is mounted on the robot's head to capture visual data, providing RGB image streams to both the CASPER system and the user interface. (3) A laptop computer is connected to the robot to display the live RGB image feed to the user. The computer's speaker notifies users when autonomous operation becomes available, using audio prompts generated via the OpenAI text-to-speech API. When the system's intent prediction surpasses a predefined confidence threshold, the user confirms their intent using the computer's keyboard. We employ GPT-4o as the backbone of our visual-language model (VLM) due to its strong multimodal reasoning capabilities, though other VLM architectures may also be suitable.

B.2 Tasks

The three tasks in Fig. 3 are assistive mobile manipulation tasks in a public building, each requiring multi-step intent inference:

- *Shelf* task has three subtasks: Picking up a pasta jar of the user's choice from multiple ones on a multi-layer shelf, navigating to a table, and pouring the pasta into a container selected by the user among multiple containers.
- *Toy* task has five subtasks: Picking up a toy of the user's choice among multiple ones from a table, navigating to a door among multiple exits, opening the door by tapping the card on the card reader, navigating to a table among multiple pieces of furniture, and place the toy in one of the containers on the table. This task is especially challenging since it is a long-horizon task involving five substeps.
- *Door* task has two subtasks: Navigating to a specified door among different landmarks and opening it using a method of the user's choice. This task consists of a suite of three different doors, each of which requires different ways of being opened. The task includes three door types: (1) push to open, (2) press an accessibility button or push to open, (3) tap an access card or pull the handle.

Our experiments span across $20+$ objects, 7 furniture types, and 5 rooms. The diverse objects and skills required in these tasks closely mimic the open-ended possibilities that assistive robots encounter in the real world. Furthermore, the robot must utilize human input to choose the correct action among multiple options in order to successfully complete the task.

C More Human Study Results

Detailed Human Study Results. We present per-task results for NASA-TLX and user satisfaction scores in Figure 8.

Participant Interview: Qualitative Feedback on Methods. We present the qualitative responses from users to gain deeper insights into their experiences with each method. The feedback highlights key aspects of usability, intent inference accuracy, execution reliability, and user workload. Users generally favored CASPER for its ease of use and high success rate, while Full Teleop was perceived as slow and effort-intensive due to full manual control. HAT, despite attempting automated assistance, was often unreliable in execution, leading to frustration. These qualitative findings complement our quantitative results, reinforcing the importance of accurate intent inference, seamless execution, and minimizing user workload in assistive teleoperation systems.

1) CASPER*: most preferred, easiest to use, and most reliable*

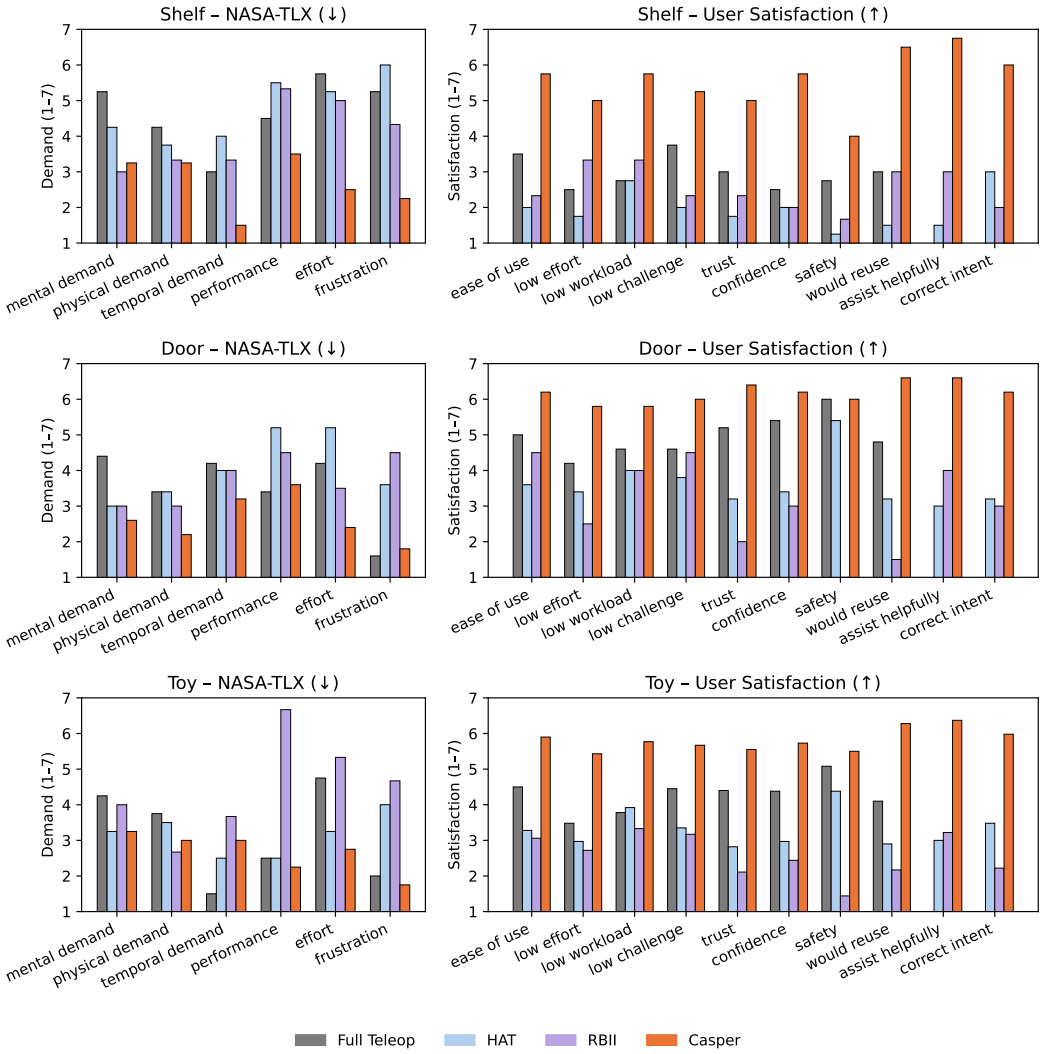

Figure 8: **Per-task results for NASA-TLX and user satisfaction scores.** Note that for user satisfaction scores, "assist helpfully" and "correct intent" are not applicable to Full Teleop.

Users generally praised CASPER for accuracy and low workload.

> "... made it easiest to get the job done and I felt like I had the most success."
> "... is very comfortable."
> "Never misunderstand what I want to do and never make mistakes."

Compared to other methods, CASPER required the least manual effort and had the best success rate in both intent inference and execution. However, some users mentioned that fine-grained control (e.g., pressing buttons) of the skill library could still be improved:

> "The fine-grain control (e.g., press the door open, tap card) is not as smooth."

2) Full Teleop: precise but demanding, slow, and effort-intensive

Full Teleop was seen as high-effort and slow due to its reliance on full teleoperation, making it the most mentally and physically demanding. Users found it precise but burdensome, requiring them to control every aspect of the robot manually:

> "I need to do all the control work by myself for [Full Teleop]."
> " ... takes a lot of human effort and is overall slowest to operate."
> "... was annoying to microcontrol the robot but it worked fine."

Although some appreciated its precision, they found it exhausting and inefficient, often resulting in slow operation:

> "By self-controlling, the robot moves very slow."
> "... is more precise but demanding."

Overall, users acknowledged the controllability it offered but disliked the high workload.

3) HAT: unreliable, sometimes alarming, high failure rate

HAT had mixed to negative feedback, with users describing it as unreliable, inconsistent, and prone to errors in both prediction and execution. While some users found its intent inference acceptable, execution failures made it frustrating:

> "... was a bit alarming."
> "Every time it will make some mistakes or misunderstand what I want to do."

Failures in execution had a particularly strong negative impact on user perception, as they felt that errors led to frustration and loss of trust in the system:

> "[HAT] was terrible."

Despite some users acknowledging its attempt at autonomous assistance, they preferred not to use it again due to its inconsistency:

> "[CASPER] is the best, [Full Teleop] is the worst. [HAT] is ok, but I don't want to try again."

4) RBII: unpredictable, imprecise, and occasionally concerning

RBII also received mixed to negative feedback, primarily due to its unreliable intent inference and unpredictable behavior. Users frequently expressed concerns about safety and control transparency:

> "The inferred intent is always wrong, and the control over the robot actions are always weird and potentially have security issues ... Would it accelerate towards me and hit me?"

While some users acknowledged partial task completion, they noted that it was inconsistent and often required intervention:

> "It was not that bad, it was in the midrange. I was able to complete one task completely and the other one was with difficulties."
> "It does try to do the action, but it performs the wrong actions (e.g., press the wrong spot)."

Overall, while RBII showed some capability, users found it untrustworthy and ineffective for complex tasks.

Participant Interview: Challenges and Failure Cases of CASPER. While CASPER was generally well-received for its accuracy and assistance, users highlighted areas for improvement, mainly focusing on execution reliability, responsiveness, and interaction timing. The key points for improvement include:

1) Execution failures at edge cases

Some users noted that while the robot correctly inferred their intent, it occasionally failed to execute the task properly, making it harder to recover control:

> "Although the robot knew what I was going to do, it sometimes failed in finishing the task and led to a state where it was harder to control the robot."
> "Cannot aim at a specific object well."

2) Slow responsiveness and speed

Several users felt that CASPER was too slow, particularly when transitioning between teleoperation and autonomous assistance:

> "It was a bit slow while I was in control."
> "Waiting for the robot to respond and move because it's slow."
> "The inference for pressing the door open is not very prompt."

More on Human Study Protocol and Design. We conduct a pilot study of group size $= 8$ before the official human study. The pilot study phase aims to evaluate the experimental setup's feasibility, refine task instructions, and identify potential usability issues. Specifically, we aim to assess system usability by ensuring that participants can effectively interact with CASPER and verify that the control of the system is intuitive. Second, we validate intent inference performance by analyzing whether the system reliably predicts user intent on a wide distribution of users and whether adjustments to inference thresholds or timing are necessary. We also gather preliminary user feedback by collecting qualitative insights on user experience, cognitive load, and overall satisfaction to inform improvements before the complete study.