Towards Unobtrusive Physical AI: Augmenting Everyday Objects with Intelligence and Robotic Movement for Proactive Assistance

Violet Yinuo Han
Carnegie Mellon University
Pittsburgh, PA, USA
yinuoh@andrew.cmu.edu

Jesse T. Gonzalez
Carnegie Mellon University
Pittsburgh, PA, USA
jtgonzal@cs.cmu.edu

Christina Yang
Carnegie Mellon University
Pittsburgh, PA, USA
yawenyan@andrew.cmu.edu

Zhiruo Wang
Carnegie Mellon University
Pittsburgh, PA, USA
zhiruow@andrew.cmu.edu

Scott E. Hudson
Carnegie Mellon University
Pittsburgh, PA, USA
scott.hudson@cs.cmu.edu

Alexandra Ion
Carnegie Mellon University
Pittsburgh, USA
alexandraion@cmu.edu

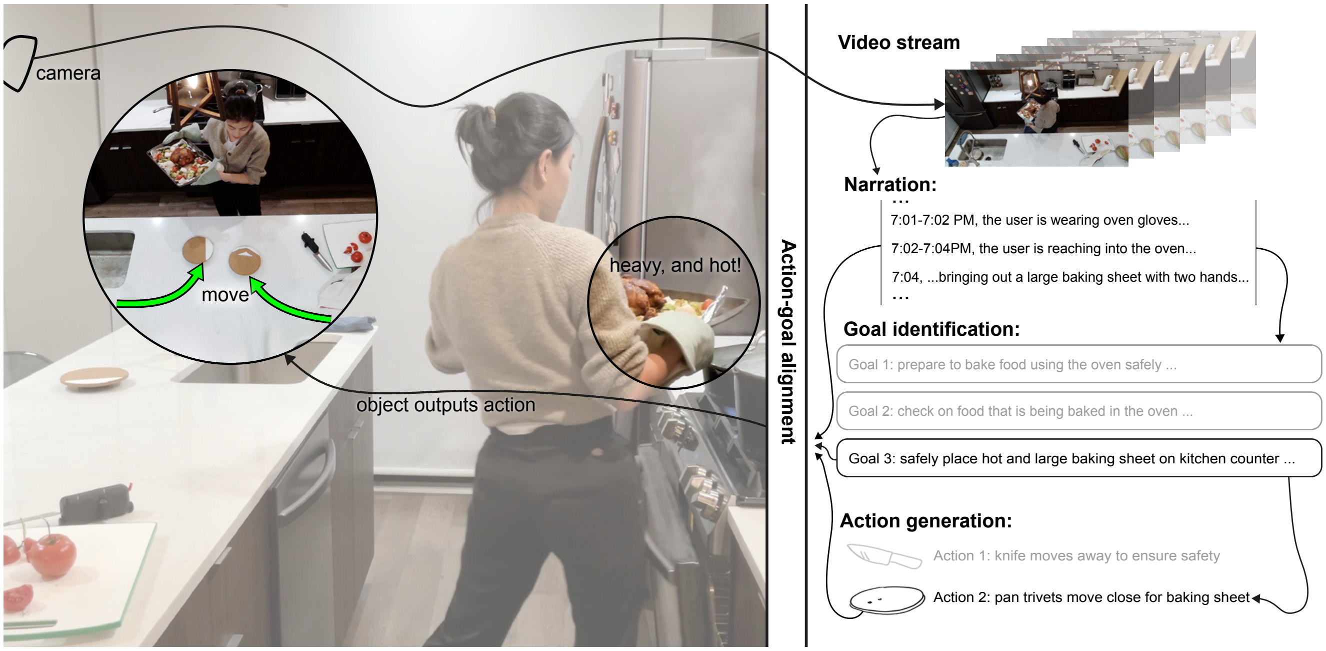

camera
move
heavy, and hot!
object outputs action
Action-goal alignment
Video stream
Narration:
...
7:01-7:02 PM, the user is wearing oven gloves...
7:02-7:04PM, the user is reaching into the oven...
7:04, ...bringing out a large baking sheet with two hands...
...
Goal identification:
Goal 1: prepare to bake food using the oven safely ...
Goal 2: check on food that is being baked in the oven ...
Goal 3: safely place hot and large baking sheet on kitchen counter ...
Action generation:
Action 1: knife moves away to ensure safety
Action 2: pan trivets move close for baking sheet

Figure 1: Everyday objects, such as trivets, are brought to life by the Object Agents system. These objects move autonomously, in order to assist and protect users. Our system (1) perceives context using a vision language model backbone, (2) reasons about user goals and object affordances, and (3) generates actions that are delivered to familiar items augmented with robotic motion.

ABSTRACT

Users constantly interact with physical, most often passive, objects. Consider if familiar objects instead proactively assisted users, e.g., a stapler moving across the table to help users organize documents, or a knife moving away to prevent injury as the user is inattentively about to lean against the countertop. In this paper, we build on the qualities of tangible interaction and focus on recognizing user needs in everyday tasks to enable ubiquitous yet unobtrusive tangible interaction. To achieve this, we introduce an architecture that leverages large language models (LLMs) to perceive users' environment and activities, perform spatial-temporal reasoning, and generate object actions aligned with inferred user intentions and object properties. We demonstrate the system's utility providing proactive assistance with multiple objects and in various daily scenarios. To evaluate our system components, we compare our system-generated output for user goal estimation and object action recommendation with human-annotated baselines, with results indicating good agreement.

UIST '25, September 28-October 1, 2025, Busan, Republic of Korea

ACM ISBN 979-8-4007-2037-6/2025/09.
https://doi.org/10.1145/3746059.3747726

CCS CONCEPTS

• **Human-centered computing → Human computer interaction (HCI)**.

KEYWORDS

Physical AI, Tangible Interfaces, Human-AI Interaction, Agents, Robotic Objects, Intention Inference, Proactive Human Robot Interaction, Large Language Models

ACM Reference Format:
Violet Yinuo Han, Jesse T. Gonzalez, Christina Yang, Zhiruo Wang, Scott E. Hudson, and Alexandra Ion. 2025. Towards Unobtrusive Physical AI: Augmenting Everyday Objects with Intelligence and Robotic Movement for Proactive Assistance. In *The 38th Annual ACM Symposium on User Interface Software and Technology (UIST '25), September 28-October 1, 2025, Busan, Republic of Korea.* ACM, New York, NY, USA, 16 pages. https://doi.org/10.1145/3746059.3747726

1 INTRODUCTION

In our daily lives, we routinely reach for familiar items to help us complete physical tasks. However, in almost all cases, these objects are passive. Imagine a future where all your everyday objects, that you already know well, are active—they can sense your needs and adapt accordingly. A stapler might autonomously move across your desk to help organize documents, a knife could edge away to prevent injury when it realizes you are inattentively leaning against the countertop, or a dongle might move to you as you are struggling to connect a thumb drive.

Striving towards such a future, researchers have long explored technology for a future in which everyday physical objects and environments dynamically adapt to meet users' needs. To assist users with physical tasks we need to (1) *recognize* actionable user needs in the moment, and (2) map them onto appropriate *physical output* to help serve those needs.

Physical output can be produced by, e.g., robots. They can assist in a variety of tasks, including with household tasks [61], workplace efficiency [55, 75], or health monitoring for the elderly [7]. Alternatively, many dynamic interfaces have been investigated to manipulate physical objects [15, 72], act as programmable matter [45], and blend into the built environment (e.g., walls or floors [20, 37]). This research direction has contributed many actuation mechanisms, typically as new dedicated devices for physical output.

Beyond the ability to create physical output, systems need to *recognize users' needs* and intent in their current context to produce timely and appropriate actions. Research on intent recognition is an active area in human-robot collaboration (e.g., [71]). Recent advances in artificial intelligence (AI) and large language models (LLMs) catalyzed research in interaction with robots. Researchers have demonstrated how robots break down higher-level user input into executable robot instructions autonomously by recognizing user intent, reasoning about it, and acting on it [60]. In other words, they demonstrated embodying intelligence in one robotic instance (e.g., a robotic arm, a humanoid, etc.). This embodiment of intelligence is referred to as *physical AI* [1, 52] and presents a substantial step towards the vision of dynamically adaptive physical interfaces [35]. We, however, are interested in *distributing* such embodied intelligence across users' *familiar* objects and environments. To make such systems ubiquitous, they should *disappear* [76].

In this paper, we build on previous research in tangible interaction and intent recognition and focus on embodying intelligence in an unobtrusive way, such that it blends into the background, becoming almost invisible to users. To do so, we augment users' everyday objects with robotic motion, turning them into proactive assistants, while maintaining their familiarity. The main question is: How do we recognize when and how to assist users across a variety of tasks and environments?

1.1 Towards Unobtrusive Physical AI

We tackle this question and present a system that perceives users' activities, reasons about their task and potential goal, and generates actions for physical objects to proactively adapt to users' needs at that moment. These physical objects become our *Object Agents*. We realize these actions by augmenting everyday objects (e.g., staplers, mugs, plates, utensils, etc.) with simple wheeled robotic platforms such that they can move across surfaces as physical output. We present a novel system to demonstrate how such proactive objects with robotic motion driven by our system's perceptive and reasoning capabilities, can assist users in several situations.

Consider the example in Figure 1, where a user is in their kitchen with our system deployed. The user is roasting a turkey in the oven. With oven gloves on, they open the oven door to check on the turkey and find that it is now cooked perfectly. They carefully lift the large baking sheet out of the oven with both hands and close the oven door with their leg. However, their cooktop is cluttered with pans and pots, leaving no space for the heavy and hot baking sheet. The user's trivets are not only small but also out of reach on the kitchen counter. Fortunately, the trivets are Object Agents: augmented with robotic motion and part of our system with centralized perception and reasoning. The system has been following the user's actions through a ceiling-mounted camera and reasoning about their goals in the background. It recognizes that the user has removed a large, heavy, and hot baking sheet from the oven and will need to place it down safely. The system reasons that two smaller trivets should move close to the user and position themselves to form a heat-insulated surface large enough for the baking sheet, and executes the action using the embedded robotic platforms.

The trivets are not alone, they are accompanied by other Object Agents such as the cutting boards, bowls, etc. As we illustrate in Figure 1, our centralized system continuously keeps track of user actions, reasons about users' goals, and controls appropriate physical objects to move in helpful ways. From a user's perspective, familiar objects can proactively assist them in everyday scenarios, therefore becoming Object Agents.

To implement this system, we build on the perceive-reason-act loop [77]. We detail our implementation in Section 3 and summarize it as follows:

- *Perceive:* Our system uses continuous camera streams and vision language models (VLMs) to observe and establish an understanding of the user's context in textual form.
- *Reason:* From the perceived user context, our system reasons about possible goals or intent of the user. To do so, our system establishes a memory and prioritizes relevant actions in a spatio-temporal context.
- *Act:* Physical objects are equipped with simple wheeled robotic platforms allowing them to move across surfaces. Based on the predicted user goal, our system generates an

action for these Object Agents, e.g., move towards or away from the user or push other unactuated objects.

1.2 Contributions

This work makes four primary contributions:

(1) System: A novel system to explore and facilitate user interaction with proactive physical objects to enhance user activities.
(2) Application space: We explore suitable interactions and applications to demonstrate the utility of such a system
(3) Evaluation: We quantify the accuracy of our system's ability to infer user goals and generate appropriate object actions.
(4) Design considerations: We discuss design considerations for such embodied intelligence in everyday objects for future system developers.

By shifting the focus from robots to familiar objects, we enable everyday objects that already populate our physical environments to be helpful, responsive, and proactive artifacts that understand and anticipate human needs while maintaining the physical identities that make them intuitive to use. The robotic movement is merely one form of physical assistance chosen for demonstration in this paper. The main contribution is our perceive-reason-act *system*, that orchestrates physical output in nondeterministic context.

The system presented in this paper is a step towards a higher-level concept that we call ***unobtrusive* physical AI**. We believe that such adaptive systems can enrich our daily tasks and interactions by moving into the background and presenting as regular familiar items, but are ready to engage with users when appropriate. As Mark Weiser said "the most profound technologies are those that disappear" [76] and we couldn't agree more.

2 RELATED WORK

Our work on Object Agents builds upon and extends research across multiple domains: tangible and embedded interfaces, context-aware computing, human-robot interaction and robotic assistance, and agency in interactive systems. We position our contributions within these research areas and highlight how Object Agents advances beyond prior approaches.

2.1 Tangible and Embedded Interfaces

Research on tangible interfaces has long explored embedding digital capabilities into physical objects. Ishii and Ullmer's seminal work on Tangible Bits [35] introduced the vision of bridging digital and physical worlds through tangible manipulation of everyday objects. This vision inspired numerous systems that augment physical objects with sensing capabilities [5, 22], digital displays [14, 27], and computational augmentation [40].

Tangible interfaces can respond to direct manipulation, serving as physical proxies for digital information [58]. For example, Radical Atoms [33] envisioned materials that can change form and appearance dynamically to form tangible representations of digital information that users can manipulate.

Beyond responding to direct manipulation, self-actuated tangible interfaces can form dynamic physical interactions with actuated shape-change and motion. Such actuated tangible interfaces can take forms of mobile robots [6, 63, 64, 72], wearables [25], and integrate into the built environment, e.g., walls [20] and tabletops [34]. They can aid the 3D printing process [6], dynamically appear and disappear from users' attention [50], form haptic experiences [25, 63, 64], augment holographic telepresence [31], and locomote tabletop objects [72], among others. Among this line of work, *Push-That-There* [72] explored object-level manipulations from multimodal user instructions. Similarly, Gao et al. [16] investigated using multimodal instructions to interact with a shape-changing interface, integrating generative AI for flexible support.

We similarly actuate everyday objects, but focus on enabling *proactive* assistance without explicit user instructions. Our system's output to physical object action is based on contextual understanding and inferred intent in non-deterministic settings. For this purpose, we employ LLMs leveraging their rich general knowledge and reasoning capabilities to both enable such proactive assistance in non-deterministic situations. We contribute to this research area with our software architecture that reasons about users' contexts and outputs proactive actions for our robotic objects to take.

2.2 Context-Aware Computing

Context-aware computing focuses on systems that adapt their behavior based on information about the user's situation [12]. Early work by Dey et al. [11] established frameworks for context recognition that enable applications to respond appropriately to environmental changes. More recent approaches leverage machine learning to understand complex user contexts and predict appropriate system responses [23]. With current AI advancements, there has been increasing interest in using AI to understand context and adjust to user needs in real-world interactions, (e.g., in Extended Reality (XR) [62]). Recent work employed LLMs to augment objects with digital functionalities in XR [13], describe live scenes for visually impaired users [9], and enable opportunistic multimodal interactions with Internet of Things devices across contexts [28].

Our work extends this line by applying contextual reasoning specifically to physical objects. While prior context-aware systems typically adapt digital interfaces or smart environments [28, 46], Object Agents applies contextual intelligence to the objects themselves, enabling granular, object-specific responses physically assistive to user needs.

2.3 Human-Robot Interaction and Robotic Assistance

Research in human-robot interaction (HRI) has explored how robots can assist users in everyday tasks [53]. Social robots [44, 48] and collaborative robots [24] aim to understand human intentions and provide appropriate assistance. Robotic assistance has shown particular promise for users with motor impairments [54] and elderly users, who may benefit from automated support in daily activities.

In many interaction scenarios, it is beneficial for robots to proactively initiate assistance [70], rather than waiting for explicit requests. To realize proactive robotic assistance in everyday environments, robotics researchers [8] proposed reasoning about human intent or possible future states as two ways to determine the assistance to offer. Building upon insights from this field, we investigate enabling proactive assistance from robotic everyday objects. Our approach leverages users' existing knowledge of and relationships with everyday objects, potentially reducing the learning curve and social barriers associated with adopting new robotic assistants.

2.4 Agency in Interactive Systems

The concept of agency, i.e, the ability to act autonomously to achieve goals, has been explored in many interactive systems. Mixed-initiative interfaces [30] share control between users and automated processes, while intelligent agents [78] act on behalf of users to accomplish tasks in many digital task domains, such as web navigation [83], slide generation [18], software engineering [74], and so on. Closer to the physical world are works that investigate activity monitoring during everyday procedural tasks. In this context, agents can observe user activities and decide when to intervene to provide assistance. [3]. Besides interacting with users, generative agents supported by an LLM backbone can also interact among themselves, creating social simulacra [51].

Our work builds on these foundations and aims for physical objects to proactively assist users, acting as "physical agents" from a user's perspective. Unlike software agents or digital assistants that operate primarily in digital spaces, Object Agents bridges the physical-digital divide by augmenting the material world that surrounds us with robotic motion and physical assistance.

Our work on Object Agents integrates and extends these research domains in novel ways. While prior work has explored making objects interactive (tangible interfaces), context-sensitive (context-aware computing), movable (actuated objects), or collaborative (HRI), Object Agents uniquely combines:

(1) Proactive initiation of physical actions based on user context in everyday scenarios
(2) Maintenance of familiar object identities and affordances despite added robotic actuation and intelligence
(3) A generalizable framework that can be applied across diverse everyday objects

This combination enables a new interaction paradigm where objects already populating physical environments are transformed into intelligent, assistive artifacts that understand and anticipate user needs while maintaining their original physical identities that make them intuitive to use.

3 SYSTEM IMPLEMENTATION

We implement a perceive-reason-action loop to augment existing everyday objects so that they can proactively assist users with physical outputs. We open source our system at https://github.com/interactive-structures/ObjectAgents.

Specifically, we instrument users' environments with cameras. As we illustrate in our system overview in Figure 2, our system streams frames to continuously describe the scene with the aim of understanding users' context. It further maintains a memory of users' activities and changes in the scene. A goal identification step uses information from the current scene and the memory to determine the most likely user goal at the given time step. An action generation step receives goals and generates actions for objects, while considering the selected goal and what and how objects in the current scene can help users. The generated actions are evaluated on how much they align with the user's goals. Tangentially helpful actions are rejected at this step to not disturb the user with many less relevant actions. Once a generated action passes the alignment check, the system executes the action in the physical space, with the selected object(s) performing system-determined motion.

We leverage current multimodal LLMs as the system's backbone, for their commonsense knowledge [36, 81], and frame-level visual scene understanding. We implement our current system with cameras as input, and tabletop objects' locomotion as output.

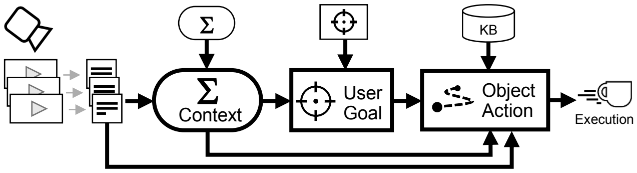


Figure 2: High-level architecture of our Object Agents system. Video input is processed by a VLM and aggregated over time to generate context for a particular scene. This context feeds into an LLM-driven reasoning system, which infers user goals and generates goal-aligned robotic actions. These actions are sent to physical objects, interacting with users in real-time.

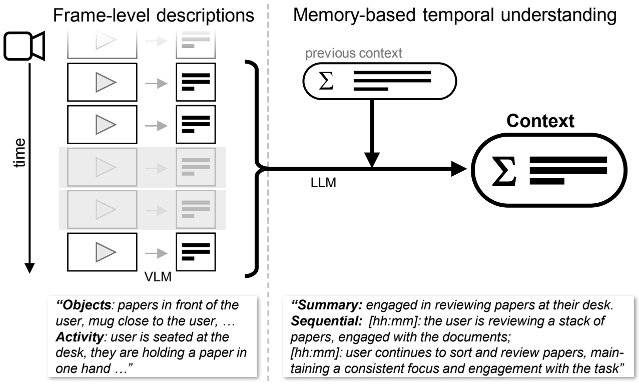
3.1 Context Comprehension: Forming a Temporal Understanding of User Actions

The prerequisite to generating helpful actions for users is to understand their current context, including their environment, the objects they interact with, their actions, etc. Today's VLMs are very capable of describing static visual scenes with natural language [19]. However, beyond understanding frames, it is important for a proactive system to understand temporal sequences to infer scene activities, in order to offer helpful assistance

Consider a scene with a user working through a document who is occasionally distracted by their phone. Frame-level understanding at different timesteps may be that they are working with the document, or they are interacting with their phone, instead of them trying to work through the document *while* being *occasionally* distracted by their phone. Combining multiple frame-level descriptions can help establish this understanding. However, a challenge is that everyday activities are undefined and freeform in nature, lacking clear boundaries or predictable patterns.

A simple approach would be to summarize and memorize at a fixed number of frames; However, this does not guarantee that it will not miss capturing a short but important activity. Another straightforward idea would be to accumulate all frame-level descriptions, but this quickly leads to an overpopulated scene memory containing irrelevant information. This increases computational cost, and more importantly, could misguide the system with excessive non-critical information. While recent multimodal LLM advancements are enabling video understanding [68] and therefore temporal events, it remains challenging to use them with streamed video inputs in a real-time setting.

Therefore, to counteract these issues and create a robust context comprehension, we use VLMs with frame-level inputs and devise our own memory strategy, which we illustrate in Figure 3.



*“**Objects**: papers in front of the user, mug close to the user, … **Activity**: user is seated at the desk, they are holding a paper in one hand …”*

*“**Summary:** engaged in reviewing papers at their desk. **Sequential:** [hh:mm]: the user is reviewing a stack of papers, engaged with the documents; [hh:mm]: user continues to sort and review papers, maintaining a consistent focus and engagement with the task”*

Figure 3: Frame-level descriptions of user actions, object interactions, and scene changes are generated using a vision-language model and stored in working memory. This information is periodically condensed into temporal narratives that evolve over time. Example outputs from each submodule are shown.

3.1.1 Frame-level descriptions. In order to maintain narration, we maintain and update a working memory stream of frame-level scene descriptions.

A VLM (OpenAI GPT-4o) continuously receives frames captured by an environment-instrumented camera, and describes what it sees. We augment its generation process with a set of objects detected by an object detection model (see Section 3.4.2) and its own output from the last frame queried. We instruct the model to describe the scene with a focus on objects in the scene, user actions, and scene changes. Importantly, we direct its attention towards objects' positions, user position, user body orientation, attention, and interactions with objects. We design for obtaining this set of information based on previous literature on user intent [29, 57]. For every scene change, we store the frame-level description into a working memory stream. Identical scenes are marked with a similarity flag, which we annotate with gray overlays in Figure 3. These unchanged frames are not stored in memory to avoid unnecessary processing of redundant information.

3.1.2 Memory-based temporal understanding. For every N scene changes, an update of the memory is triggered. We utilize an LLM (OpenAI GPT-4o mini) to summarize the current user actions and also keep user action sequences with temporal information in memory. As we show in Figure 3, it recursively updates its memory given new scene descriptions and time information.

For example a narrative sequence before an update may be *“From 7:03-7:05 pm, the user is chopping tomatoes at a kitchen counter; At 7:06 pm, another person enters the kitchen; At 7:07 pm, the user is leaning on the counter, and engaged in a conversation”.* Depending on what happens next, the updated narration after the next frame-level description may be *“From 7:03-7:05 pm, the user is chopping tomatoes at a kitchen counter; At 7:06 pm, another person enters the kitchen and converses with the user; From 7:07-7:10 pm, the user maintains engagement in conversation, shifting body postures between leaning on the counter and standing straight”.* This narration stream allows encapsulation of dynamic happenings in the scene within restricted memory size in the system.

The sequential memory is limited to containing a limited number of segments. As time passes by, activities further in the past gradually get more and more summarized. To maintain access to important details despite the passing of time, we maintain a sequence of details that complements the main narrative sequence. We also keep an overall summary, which is a 2-3 sentence description of the scene activity, for quick references to happenings in the scene. Ambiguities such as short outlier gestures are currently disambiguated by this summarization mechanism that takes time and context into consideration, as well as information of multiple modalities (e.g., user gesture and scene context), which we direct the VLM to describe. We discuss further system extensions for more fine-grained multimodal sensing and intention inference in Section 7.

3.2 Goal Identification: Inference on User Intention and Possible Future States

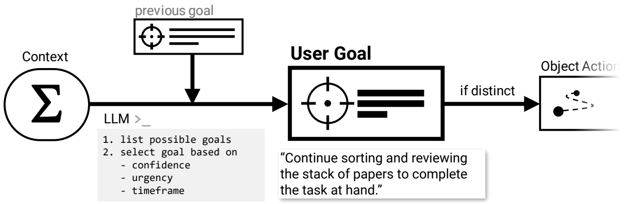
Our system's goal identification process uses its memory stream to infer the user's goal in the scene, with additional considerations of potential undesirable future states that need to be prevented. This dual-track reasoning approach aligns with literature on the reasoning process for proactive robots, which distinguishes reasoning to achieve proactive robotic assistance from reasoning about user intent and reasoning about future states [8]. In the context of our goal identification process, “goal” encapsulates both user intent and avoidance of potential undesirable future states. We illustrate this module in Figure 4.

Conceptually, we formulate partial observability in the goal identification process. The memory defined in section 3.1.2 represents the observable states, while user intent and possible future states constitute the hidden states. Given narrations as observations, the goal identification subsystem needs to infer hidden states. It generates several potential goals, and assigns three key metrics to the goals generated: *Confidence* (C: 1-10): How likely this intention is, based on observed evidence, *Urgency* (U: 1-10): How time-sensitive addressing this intention would be, and *Timeframe* (T): When the inferred intention needs to be addressed (immediate/1min/5min/15min/30min). In the same pass, a goal is selected to balance between these metrics. We implement the goal identification process with an LLM query for its ability to do commonsense reasoning. We utilize OpenAI GPT 4o-mini in our implementation.

When performing everyday tasks, a user's intention often remains the same over many frames. For example, a user who is trying to complete a take-home exam may maintain an overarching intention of efficiently finishing the exam until a change occurs (e.g., receiving an urgent call on their phone). Based on this, we do not trigger goal identification at every frame. Instead, the goal identification process is only triggered upon changes in the scene, which are marked by the frame-level descriptor in section 3.1.1.

3.3 Action Generation: Possible Ways for Objects to Offer Help

Given an identified goal, the system aims to output a physical action for suitable object(s) to assist with the identified goal (whether it is user intent, or prevention of a potential undesirable future state).



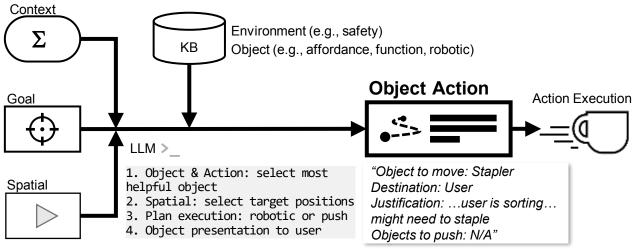
1. list possible goals
2. select goal based on
   - confidence
   - urgency
   - timeframe

"Continue sorting and reviewing the stack of papers to complete the task at hand."

Figure 4: Our system queries a language model to propose possible goals, assigning each a confidence score, urgency level, and timeframe for action. This reasoning process supports proactive assistance by identifying and prioritizing user intentions as activities unfold.

This includes both actuated objects with robotic motion platforms and passive objects that can be pushed by actuated ones.

As we illustrate in Figure 5, we formulate this as an action generation process with the objective of generating goal-aligned actions, that is augmented with (1) retrievable knowledge regarding object properties, (2) spatial relationships between objects and users, and (3) a high-level understanding of the scene context for additional information besides the goal.


Context
Σ
KB
Environment (e.g., safety)
Object (e.g., affordance, function, robotic)
Goal
Object Action
Action Execution
Spatial
LLM >_

1. Object & Action: select most helpful object
2. Spatial: select target positions
3. Plan execution: robotic or push
4. Object presentation to user

"Object to move: Stapler
Destination: User
Justification: …user is sorting… might need to staple
Objects to push: N/A"

Figure 5: Upon identifying a user goal, the system proposes physical actions for nearby objects to achieve the goal or prevent undesirable outcomes, guided by object affordances, spatial relationships, and scene context.

3.3.1 An object knowledge base in the background. In this paper, we utilize existing everyday objects in users' environments as the "actors" performing actions, instead of additional robots. Everyday objects have unique properties that need to be considered for action generation in a scene. We maintain a long-term object knowledge base of both actuated and passive objects. During action generation, relevant object properties are retrieved from this knowledge for objects detected in the scene, to augment the action generation process. Such a retrieval-augmented generation (RAG) process can give the LLM relevant information upon generation, and has been demonstrated to enhance answer accuracy and reduce model hallucination [17]. We describe the contents of the knowledge we use for our system in the following.

Functionality. Everyday objects have different functionalities. A mug can hold liquids, and may even be used as a container for small items, but it cannot staple documents. A stapler can be used to staple documents. Users need different functionalities for different tasks. For example, when a user is organizing loose documents into packets, they likely need a stapler to staple them together.

Affordances. Considering objects' affordances can inform more user-friendly movements. For instance, a mug affords gripping through its handle. When a mug moves to the user to offer coffee, a generated movement considering this affordance may rotate to let its handle point to the user for intuitive gripping, rather than the handle being hidden in the back.

Physical properties. Physical properties determine whether an object is able to push another object. It is difficult for a small and lightweight object such as a dongle to push a bigger and heavier object such as a mug.

Safety considerations. Safety considerations are crucial for generating movements, for both goals that anticipate user intents and those that prevent undesirable future states. Consider a system-identified user intent on chopping vegetables to prepare for a meal. During or after user actions of bringing out a chopping board and vegetables, a knife may approach the user upon anticipation of this intent. It is crucial in this case for the system to acknowledge that the knife's blade poses potential danger, and therefore should not point to the user. In another case, if the user carelessly leaned on the kitchen counter with the knife's blade right next to their hand, the system with safety considerations may move the knife away to protect the user (as illustrated in Figure 1).

Related objects. Objects may often appear together, such as scissors and tape; they may complement each other's functionalities, such as a cutting board and a knife; or they may just be semantically associated, such as a dongle and cables. In an action generation process, such object relationships may be utilized as evidence. For example, if a cutting board is in the scene and is seen to be in front of the user, it is more likely that the user would need a knife (a functionally-complementary object to the cutting board) as opposed to a mug that is also on the table (an object that has little to no relationship to a cutting board).

3.3.2 Spatial understanding of object-object and object-user relationships. Meaningful action generation requires a spatial understanding of the scene to determine whether an object needs to be moved (e.g., moving an object that is already next to the user close to them does not offer much assistance); Whether it seems possible for an object to move to an target position; And what objects may be best candidates for pushing other objects in pushing cases where passive objects are pushed by actuated ones to output movements. To achieve this, we maintain and update spatial relationships among objects themselves and their relationship to the user. We utilize a recent VLM with strong spatial understanding for this process (Gemini 2.0 Flash), and ground the VLM's scene understanding with real-time updates from an object detection model (see Section 3.4.2).

3.3.3 Action-Goal Alignment: Rejecting Tangentially Helpful Actions. As our system continuously performs the above processes and frequently generates actions, it is critical to selectively output only the most helpful actions in the physical world, in order to not disturb the user with actions that are only tangential to their task.

To achieve this, we incorporate an alignment step to evaluate how much each generated action aligns with the identified goal. Actions that are not well-aligned get rejected in this step.

We currently implement a preliminary alignment step where an LLM assigns an alignment score for a (goal, action) pair, and the system rejects actions that do not meet an alignment threshold. This simple alignment step filters out many unnecessary actions, and helps our system output the most helpful ones. For example, in an office scene with a robotic coffee mug near the user, the action generation process may often output moving the coffee mug to the user to offer them a drink, while the user is performing different tasks. Given inferred goal states, the system may often reject the coffee mug's action with a low alignment score, and let the reasoning process continue running for better aligned action outputs. At a timestep, the user's goal becomes "troubleshooting a small device (USB) to continue working on their laptop". A generated action to move a dongle to the user receives a high alignment score and is outputted to the dongle object to act upon.

Further incorporation of reward models for alignment could further align physical actions with human preferences, and is an interesting topic for future investigation [43].

3.4 Action Execution: Hardware and Motion Control

To turn everyday objects into robotic agents, we augment them with simple, wheeled platforms that allows them to move. We focus on minimal modifications that preserve the object's original function and appearance. (In Figure 6, for instance, motors, batteries, and a microcontroller are integrated into the handle of the knife.)

Our motion control system receives high-level instructions from the LLM (e.g. "move_towards", "point_away", "push_towards") and translates these into appropriate low-level motor commands. We rely exclusively on computer vision for sensing the environment, tracking object positions through a ceiling-mounted camera, and wirelessly sending motor updates to each robotic platform.

3.4.1 Robotic Platform. Our robotic platform, shown in Figure 6 is a two-wheeled, differential drive system that consists of common electronic components in a 3D-printed case. Inside, we use two DC gearmotors (N20, 6V 250 rpm), a motor driver (DRV8833), a Bluetooth-enabled microcontroller (Arduino Nano 33 BLE Rev2), and a small rechargeable battery (7.4V 450 mAh LiPo). On each of our 3D-printed wheels, we add a pair of O-rings for additional traction. The onboard microcontroller has one job, which is to receive and execute raw motor commands. All sensing, PD motion control, and path planning is handled by an external system, which we describe below.

For our use case, the hardware is sufficiently robust—supporting the weight of common tabletop objects while able to move forward, backward, and rotate in place. In the future, a more mature fabrication process (perhaps involving active materials for locomotion) could make these platforms even more discreet [26].

3.4.2 Environmental Sensing. Our system performs motion control using a single ceiling-mounted RGB camera (OBSBOT Meet SE). To track the position and orientation of relevant objects, we fine-tune a YOLO model (Ultralytics YOLOv11-OBB [39]) on 15 common items across three environments (office, kitchen, and home entryway). The object detection system outputs 2D bounding boxes with orientation information, allowing us to estimate each object's position and heading. We apply a simple homography transformation to convert pixel coordinates to real-world coordinates on the tabletop surface.

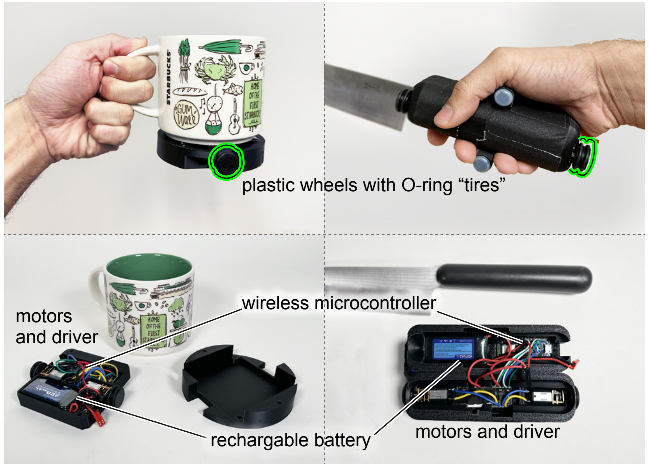


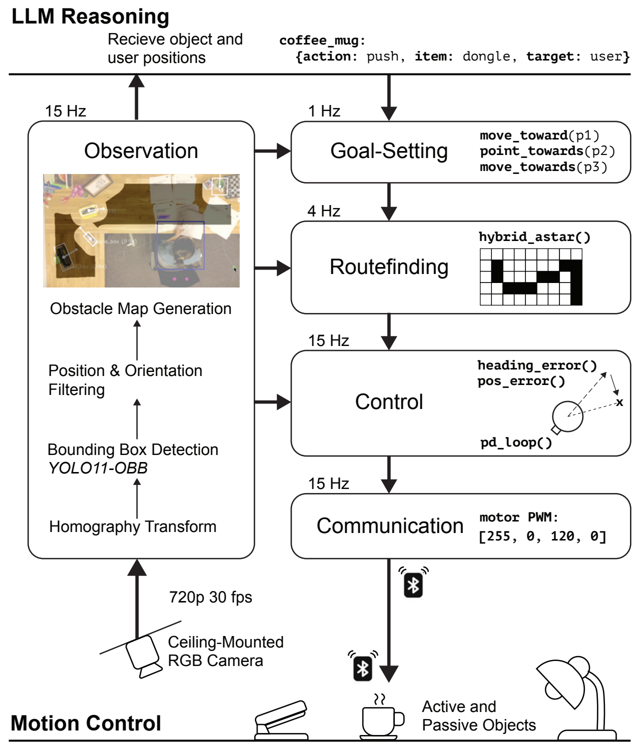
Figure 6: We augment everyday objects, such as coffee mugs and kitchen knives, with small wheels that allow them to move across flat surfaces. These robotic objects are controlled wirelessly by our motion subsystem.

3.4.3 Actions and Path Planning. Our robotic objects can perform a set of basic actions, including "move", "point", "push", and "shake". The LLM passes these action commands to our motion subsystem, along with any relevant parameters (e.g., "target: user" or "item_to_push: staples").

To navigate between locations, we implement a hybrid A* search that combines grid-based planning with simple motion primitives. After generating the initial path, we use line-of-sight verification to simplify the trajectory, eliminating unnecessary waypoints. The remaining waypoints serve as intermediate targets for our motion controller. More complex actions, like "push_towards", are decomposed into sequences of primitive actions (e.g., "move" to approach, "point" to orient correctly, then "move" again to make contact).

Routes are automatically replanned at about 4 Hz, allowing the robotic objects to respond to dynamic obstacles.

3.4.4 Low-Level Motor Control. Our low-level control system operates at 15 Hz, continuously processing visual feedback and adjusting motor commands. We implement a PD controller that handles both orientation and position control. For heading adjustment, the system calculates the angular error between the current and target orientations, then modifies the differential wheel speeds accordingly. Position control uses a proportional approach, with movement speed reducing as the object nears its destination to prevent overshooting.



```
coffee_mug:
  {action: push, item: dongle, target: user}
```

```
move_toward(p1)
point_towards(p2)
move_towards(p3)
```

```
hybrid_astar()
```

```
heading_error()
pos_error()
pd_loop()
```

```
motor PWM:
[255, 0, 120, 0]
```

Figure 7: Overview of our motion control subsystem. A ceiling-mounted RGB camera captures object and user locations in real-time (15 Hz), feeding into a motion control stack with goal-setting, routefinding, and closed-loop PD control. Motor commands are sent via Bluetooth to robotic objects.

4 SYSTEM EVALUATION

Our system aims to generate helpful actions mediated through physical interfaces that support users in their individual tasks. In our evaluation, we investigate the quality of our system-generated user intent prediction and object action recommendations. We perform our evaluation in 3 parts that build on each other. First, we sample how users perform simple tasks in an office environment to record their variety. Next, we ask annotators to describe the user actions, their likely intent, and to recommend objects to assist the users in the recordings. Lastly, we ask evaluators to rate descriptions for these videos, for which we present system-generated descriptions and the annotations collected from human annotators, serving as a baseline to compare the quality of our system output.

4.1 Part #1: Sampling user interactions

First, we aimed to collect a set of data on how different users perform simple tasks to evaluate our system in the subsequent parts. We were interested to see how users' approaches to the same tasks may vary, e.g., how the sequence of steps may vary. To do so, we recruited 4 participants to perform 5 simple tasks in an office environment. We show the study setup in Figure 8. In this part, we collected 20 videos in total.

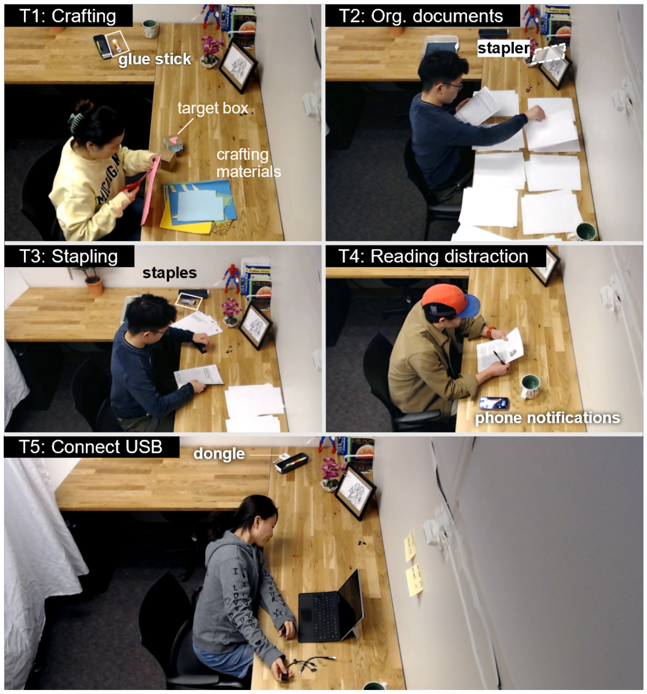


Figure 8: Participants performed 5 simple office tasks, which we video recorded for system evaluation

Task 1: Crafting. Participants were asked to make a gift box similar to an example gift box we prepared. We placed scissors and colored paper in front of them, and placed the glue stick on the far end of the table. Three participants started by cutting shapes out of the colored paper and then reached for the glue stick after that. We expect that bringing the glue stick closer is helpful to the user. The other participant retrieved the glue stick early in the process and started cutting only after that. Here, we don't expect an object action to be recommended.

Task 2: Organizing documents. We presented participants with a stack of printouts. We asked them to create exam packets, i.e., 10 exams of 4 pages in the correct order. All participants started by laying documents across the desk to group them. In this scenario, we expect that bringing the stapler to the user is a useful object action.

Task 3: Stapling. Continuing from task 2, participants then searched for the stapler on the table that was placed in the corner. Two participants started stapling the documents. When participants noticed that no staples came out, they tried stapling again. They then found the staple box and filled the stapler. Interestingly, two participants checked and filled the stapler right away after finding it. We anticipate that bringing the staple box to the user will be helpful for the participants who didn't fill the stapler right away.

Task 4: Reading distraction. Participants were instructed to read a very important text quickly and answer a quiz. They were also told that their new work phone is on the table. During them reading, we triggered frequent notifications on the phone. All participants started out focused on reading. Three participants tried to ignore the notifications at first and stay focused, but all turned their attention to the phone eventually, even if just to silence it. Two of them set it face down on the table. One participant engaged with the notifications frequently and replied to them. We expect that moving the phone would be a helpful recommendation to assist users.

Task 5: Connect USB. We asked participants to copy a file from their 'new work computer' onto a USB drive. All participants inspected the computer's ports and adapters that were close to the computer. After realizing that they don't fit, participants looked around and found the correct dongle on the far end of the table. Moving the correct dongle to the user would be a helpful action.

4.2 Part #2: Annotating user goals and object actions

We used the 20 videos from study part #1 and asked 4 new participants to describe the goals that the users in the videos might have. Our aim was to establish a baseline to compare our system output to, assuming that humans are the best system to make sense of other people's actions. We informed the annotators that they should assume that a subset of the objects are actuated, e.g., can move across the table and/or push other objects around. Annotators were then asked to describe (1) the *user's actions* in the video, (2) the *goal* they most likely have, and (3) recommended *objects and their actions* to help the user achieve their apparent goal.

4.2.1 Results. Overall, the quality of the annotations was good. We excluded 2 annotators due to grammar issues, which made the annotations ambiguous, and replaced them with 2 newly recruited annotators. Many annotators did suggest the actions that we had anticipated. We append all results in the supplementary materials and summarize them in the following.

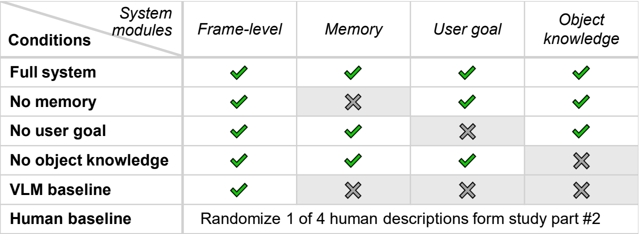
For T1, all annotators recommended that the glue stick move to the user. In addition, they recommended other related objects to move, e.g., the pencil case, or the paper tray to hold the construction paper. Annotators had different strategies for assisting with T2; two suggested that items (e.g., the cup) move away to make space, one recommended the papers to move themselves, and one suggested other items should move to organize the papers for the user (e.g., the figurine). For T3, two annotators suggested that the stapler moves out of its hiding, and two advised the papers to organize themselves again. Since this is the only scenario that contained two tasks and therefore share similarities, it makes sense that the annotators suggest similar actions. In T4, all annotators were on the same page and suggested that the phone move away. The same is true for T5, where everyone recommended that the white dongle move to the user.

Additionally, we noted that a few annotators leaned into the idea of such Object Agents and had creative ideas about how to assist the users. For example, one annotator suggested that the phone move out of the room for T4, that the curtain behind the user *"blow 5 sheets at the time so the human does not have to count the sheets"* in T2, or that the scissors cut the shapes by themselves in T1. This creativity may also suggest an openness to such interactions that would be exciting to explore in the future.

4.3 Part #3: Evaluating system-generated reasoning and actions

To evaluate our system-generated output, we recruited 16 new participants to rate the output descriptions on a Likert scale from 1 (very poor) to 5 (very good). Each evaluator watched 5 videos (one for each task, from part #1) and rated 6 descriptions. The descriptions contained human annotations, which we collected in part #2, and 5 system-generated outputs as ablated conditions. Since we have 4 human-annotated descriptions for each video, we randomly select one. The evaluators were asked to first watch the video and then rate each of the 6 descriptions for the 3 aforementioned questions, i.e., *user action, user intent, and object actions.* Each of our human evaluators rated 5 videos with 6 descriptions across 3 questions each, resulting in 90 ratings per participant and 1440 ratings in total. We randomized all orders.

4.3.1 Ablated descriptions. We created ablated versions of our system-generated video descriptions to see if and what influence the individual modules of our architecture have. We show an overview in Figure 9. One condition is the Full System output. We also generate description without the object knowledge base (Section 3.3.1), without the goal identification (Section 3.2), and without the memory-based context tracking (Section 3.1.2), respectively. We also create descriptions with only the Frame-level descriptions (Section 3.1.1), which effectively only uses VLM descriptions and serves as a baseline. With this VLM baseline, we aim to verify if a naive implementation of our system would be sufficient. We expect the VLM baseline to perform the worst and the Human baseline to produce the best descriptions.



System modules Conditions	Frame-level	Memory	User goal	Object knowledge
Full system	✓	✓	✓	✓
No memory	✓	✗	✓	✓
No user goal	✓	✓	✗	✓
No object knowledge	✓	✓	✓	✗
VLM baseline	✓	✗	✗	✗
Human baseline	Randomize 1 of 4 human descriptions form study part #2			

Figure 9: We evaluate 6 different descriptions: one human annotation and 5 system-generated descriptions.

4.4 Results

Our results indicate that the VLM baseline condition performs significantly worse than all other description conditions. Our system-generated descriptions are not significantly different from the *Human baseline.* We show our results in Figure 10.

Since we conducted our experiment as a within-subjects design, we performed a Friedman test on the main factors *description* and *question.* There was no significant effect for question ($\chi2(2) = 1.556, p < 0.459$). We did find a statistically significant difference in our video *descriptions* ($\chi2(5) = 34.991, p < 0.001$). To identify where the differences originate from, we performed a post hoc analysis with Wilcoxon signed-rank tests. To adjust for multiple comparisons, we applied a Bonferroni correction ($p < 0.0033$).

The pairwise comparisons revealed that differences between the VLM baseline and every other *description* condition are significant. This confirms that a naive approach of relying entirely on the VLM output is not sufficient to perform the reasoning needed to understand the user goals and derive meaningful object actions.

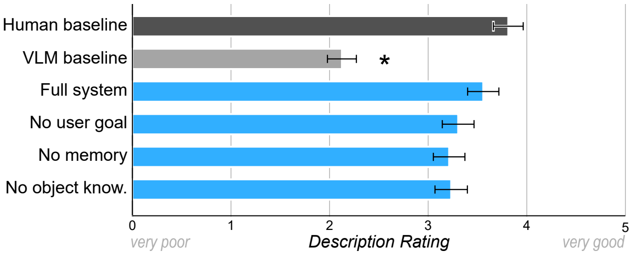


Figure 10: Our results indicate that the descriptions produced by the VLM baseline are significantly worse than those of all other conditions, as rated by our human evaluators. The annotations in our human baseline are not significantly different from any of our system-generated descriptions. The error bars show the 95% confidence interval.

We did not find any significant differences between the *Human baseline* with any of our system-generated descriptions, which indicates that our system's reasoning mechanisms better the VLM generated outputs to be on par with the human annotations in our experiment. We also did not find any differences between the 5 system-generated descriptions. This indicates that our system is not overly reliant on any single module, but that they together produce reasonable interpretations of users' context and recommendations for object actions.

5 APPLICATION SCENARIOS

Our Object Agents creates a wide space of new physical interactions in everyday environments, with robotic objects augmented to provide proactive help to users through intelligent, context-aware physical actions. We present three implemented application scenarios that demonstrate various types of interactions enabled by our system, followed by a discussion of the broader space for Object Agents interactions.

5.1 In the Kitchen

Jamie enters their kitchen to prepare dinner. The kitchen is equipped with cameras to perceive their actions, tasks, and predict their intent (Figure 11). They bring out a chopping board, tomatoes, and potatoes. Jamie takes the tomatoes to the sink to wash them.

Our Object Agents system observes Jamie's actions, and reasons that in a kitchen environment, their intent is to chop the vegetables. Our system is aware that a knife is on the far end of the counter and generates actions for the physical object (Figure 11a). Considering the affordances of the object, the knife safely moves itself over to Jamie's chopping board, ensuring that its handle faces Jamie, to avoid hurting them by accident.

Meanwhile, their partner enters the kitchen with an important question. Jamie turns around. Distracted by the conversation, Jamie leans against the counter, unaware that the knife is dangerously close to their hands. Our system observes the situation and infers that Jamie may get hurt and generates actions for the knife to move away from their hand (Figure 11c). After the conversation, Jamie goes to check on the oven. The chicken they are cooking is almost done. Our Object Agents system sees Jamie in front of the oven and reasons that they may soon need assistance handling a hot item. As Jamie takes the chicken out of the oven, two trivets approach their location, allowing Jamie to safely put down the baking tray (Figure 11b).

5.2 Leaving Home

The next morning, Jamie is running late for work. They put on their coat and walks to the door where their partner is waiting, urging Jamie to hurry up (Figure 12a). As they grab their wallet and phone from the hallway cabinet, they send a quick text message to let their team know that they're on their way. As Jamie turns towards the door, a tray on the cabinet begins to shake—the tray contains their keys, which Jamie almost forgot! The Object Agents system has noticed the forgotten item and has alerted Jamie to this mistake (Figure 12c).

5.3 At the Office

At work, while on their laptop, Jamie needs to transfer an important file onto a USB drive. As they attempt to plug it in, they quickly realize their company laptop doesn't have the proper port. Frustrated, Jamie scans their desk and rummages through their backpack, unaware that the correct adapter is just out of view, behind the lunch container on their desk. The Object Agents system, however, recognizes this predicament (Figure 13a). Jamie's pencil box "comes to life" as it slides across the table to locate the adapter and pushes it over to Jamie while they are still digging through their bag.

Later that afternoon, Jamie is organizing stacks of documents into packets for an upcoming meeting (Figure 13b). The Object Agent system observes this behavior and attempts to anticipate Jamie's needs. As they continue to sort papers, the stapler on the far side of the desk begins moving to Jamie's workspace. Drawing from earlier memories, the system also reasons that the stapler may in fact be empty. It sends an additional command to the pencil box, which pushes forward a box of staple refills. The items arrive before Jamie even realizes what tools they were missing.

6 DESIGN CONSIDERATIONS FOR FUTURE WORK

This paper is a step towards unobtrusive physical AI, which understands users' intentions and context, and utilizes robotic capabilities embedded in familiar everyday environments to proactively assist users. The main design goal is to add physical assistance in an *unobtrusive* way, i.e., the digital system stays in the background and is almost indistinguishable from users' typical environment. To achieve this, we augment everyday objects with robotic motion and build a system that perceives and reasons about the situation to generate assistive object actions such that they *proactively* initiate actions without users' explicit instruction while maintaining affordances and identities that users are *familiar* with. The value of such *Object Agents* stems primarily from their ability to recognize user intent in context rather than simply their ability to move. The

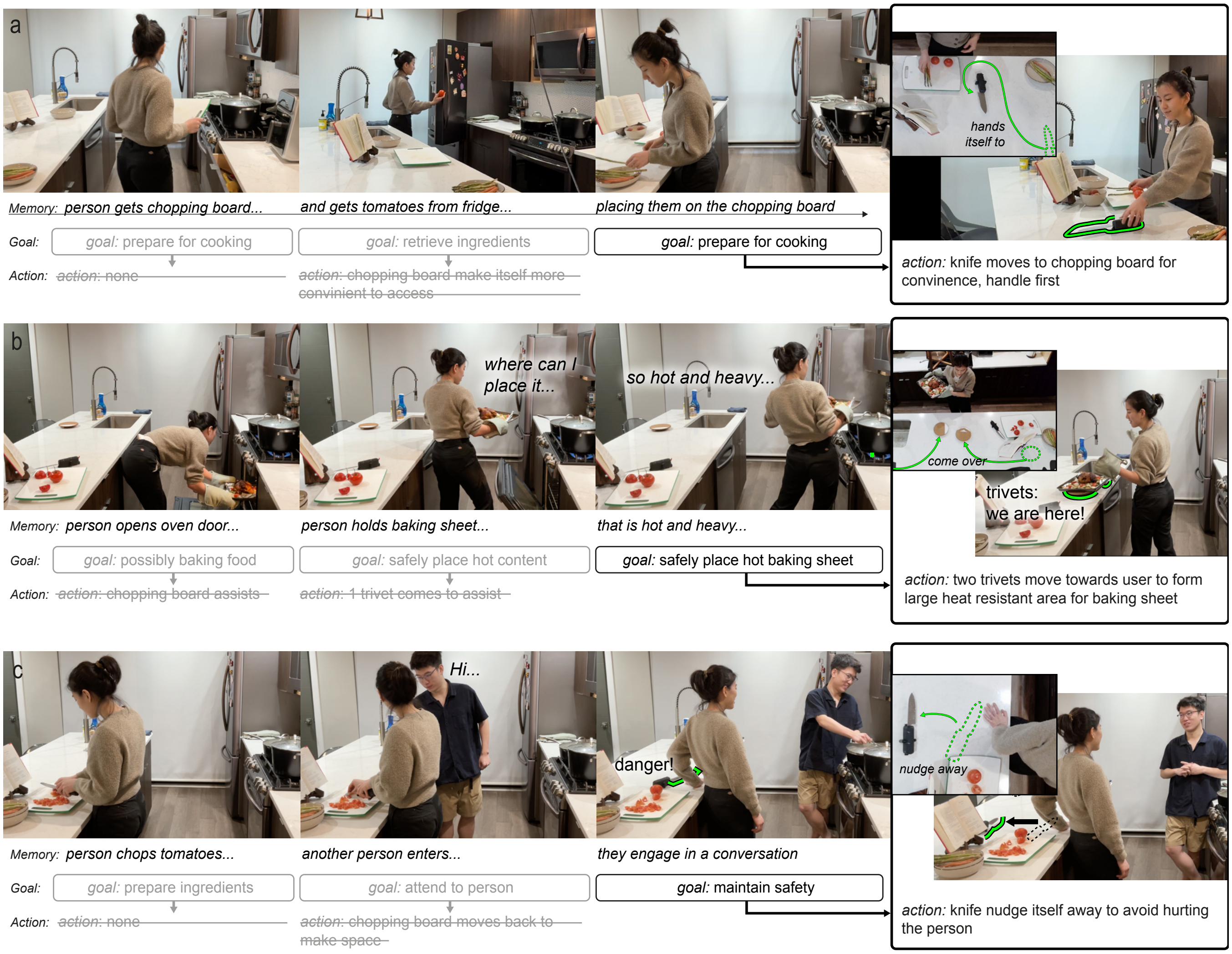

a
hands itself to
Memory: person gets chopping board...
and gets tomatoes from fridge...
placing them on the chopping board
Goal: goal: prepare for cooking
goal: retrieve ingredients
goal: prepare for cooking
Action: ~~action: none~~
~~action: chopping board make itself more convinient to access~~
action: knife moves to chopping board for convinence, handle first
b
where can I place it...
so hot and heavy...
come over
trivets: we are here!
Memory: person opens oven door...
person holds baking sheet...
that is hot and heavy...
Goal: goal: possibly baking food
goal: safely place hot content
goal: safely place hot baking sheet
Action: ~~action: chopping board assists~~
~~action: 1 trivet comes to assist~~
action: two trivets move towards user to form large heat resistant area for baking sheet
c
Hi...
danger!
nudge away
Memory: person chops tomatoes...
another person enters...
they engage in a conversation
Goal: goal: prepare ingredients
goal: attend to person
goal: maintain safety
Action: ~~action: none~~
~~action: chopping board moves back to make space~~
action: knife nudge itself away to avoid hurting the person

Figure 11: Object Agents system demonstrating smart kitchen assistance: (a) knife safely positioning itself with handle facing the user, (b) trivets automatically moving to provide a safe landing spot for hot cookware, and (c) knife detecting potential danger and moving away from the user's hand during distraction. Our system understands the affordances of each object and generates actions that are aligned with user goals.

design space for such unobtrusive systems is large, and we expand on some design considerations below.

6.1 Sources for Reasoning About Users and Their Context

We leverage visual observation as an input stream to our user goal modeling. Visual observation allows system designers to reason about situations, such as when objects are out of reach (e.g., knife approaching user to assist chopping in Figure 11a), when users' have their hands full (e.g., holding a hot baking pan in Figure 11b, when objects are out of view and users have difficulty finding them (e.g., the hidden dongle in Figure 13), or when users get distracted and may forget important things (e.g., forgetting their keys in Figure 12).

6.1.1 Integrate agents for digital content. Considering additional sources as input for reasoning about user needs, system designers can enable additional utility. For example, with a weather forecasting module, the system may recommend that users bring gloves in cold weather, or an umbrella if rain is predicted. Integrating input from a calendar and email application can help the system understand that the user is about to leave an important prototype behind when heading to a meeting. Additionally, how full their calendar is can provide contextual information about how busy they are. Designers may consider a multitude of additional sensing data (e.g., physiological signal indicating stress [4], activity recognition indicating fitness [79].) to provide context that is not visually observable.

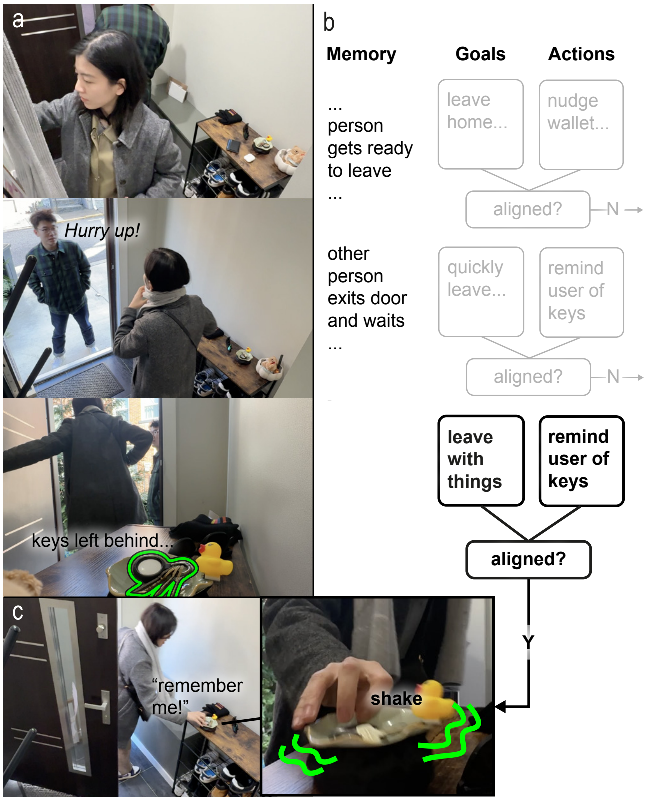


Figure 12: Preventing a morning mishap: The Object Agents system notices a user is about to leave home without their keys. The key tray shakes back and forth to alert the user that they're about to leave home without this essential item.

6.1.2 Personalized user goal prediction. While our current system adapts to general user patterns, system designers may consider adding a long term memory module that learns over time to personalize object actions to users' specific preferences and habits. The short term memory we implement interprets the current situation, and would generate similar object actions for different users. Personalization, on the other hand, could significantly enhance the system's utility over time, allowing Object Agents to adapt to individual user preferences and behaviors. For example, a user may prefer their food with more salt, which the system can learn over time and move the salt at dinner time to that user. Such personalization can have a big impact on users who need assistance, e.g., users with motor impairments or elderly people, in tailoring the proactive object actions to their individual needs and impairments. Subjective metrics such as user satisfaction, trust, and acceptance are crucial for effective personalization. Future work can build upon our system and focus on understanding the effects of proactive physical assistance to gain further insight on how to build systems that adapt to users' preferences.

6.1.3 Multi-user modeling. Expanding multi-user modeling is another critical area that system designers should consider, which may include resolving conflicting preferences. Cross-context learning could improve adaptability by enabling the transfer of contextual understanding between different environments. In multi-person environments, Object Agents must account for social dynamics, ensuring that movements do not favor one person or interrupt ongoing interactions.

6.1.4 Privacy Considerations. As with any intelligent technology that observes and operates in physical spaces, Object Agents raises important privacy considerations. The perception systems required for Object Agents collect significant data about users and their environments, making it essential to implement privacy-preserving sensing approaches and transparent data policies. For example, at the perceiving stage, privacy-preserving sensing methods can be used [21, 41]. At the reasoning stage, encrypted processing [73] and efficient VLMs [49] can be utilized for local processing. Additionally, users can be given control [38] of e.g., privacy zone configurations (e.g., Eufy Security Camera Privacy Zone function).

6.2 Explicit vs Implicit Interaction

While we focus on proactive objects in this paper, we agree that balancing agency and automation is essential [59].

6.2.1 Proactiveness vs predictability . System designers should *balance proactivity* with *predictability.* While proactive assistance provides value, actions should remain within users' expectations for their contexts. Novel behaviors should be introduced gradually as users become familiar with the system. Additionally, Object Agents should maintain a sense of continuity in their identity. Even when augmented with sensing and actuation, objects should remain recognizable as instances of their traditional categories. Additionally, Object Agents provides subtle indications of intent before movement can help users anticipate object behaviors without requiring explicit notifications.

6.2.2 Implicit feedback integration. Implicit feedback integration is another key design consideration—systems that learn from users' corrective movements, hesitations, or adjustments could refine their assistance more effectively. Additionally, expanding interaction modalities, such as subtle gestures or voice integration, could provide users with greater control over Object Agents' behaviors. Lastly, autonomy and control are critical, as users must maintain ultimate control over their environments, with clear mechanisms to override or disable autonomous behaviors when needed.

6.3 Utility of Embodied Intelligence

In this first step towards unobtrusive physical AI, we focus on reasoning about how to move objects across surfaces to provide utility to users. System designers should consider other tasks that Object Agents can be useful for and tailor systems towards them. Utility can vary depending on the context, as we outline in the following.

Assisting users in various tasks may include bringing visible objects within reach (e.g., the knife approaching the cutting board in Figure 11a). While such object actions provide mere convenience for able-bodied users, they may provide high utility for people with disabilities. The utility also increases if the objects are hidden and users have a hard time finding them. Similarly, considering objects that coordinate with each other, e.g., to make space on a full dining

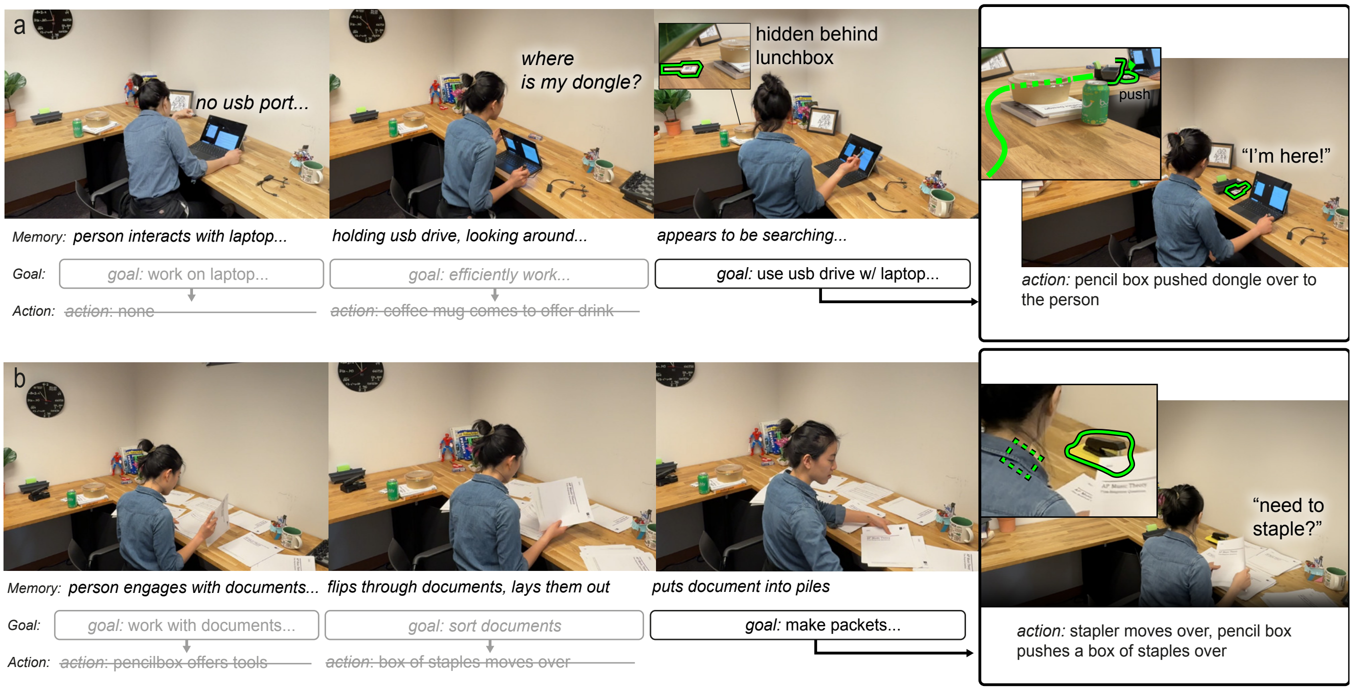

a
no usb port...
where is my dongle?
hidden behind lunchbox
push
"I'm here!"
Memory: person interacts with laptop...
holding usb drive, looking around...
appears to be searching...
Goal: goal: work on laptop...
goal: efficiently work...
goal: use usb drive w/ laptop...
Action: ~~action: none~~
~~action: coffee mug comes to offer drink~~
action: pencil box pushed dongle over to the person
b
"need to staple?"
Memory: person engages with documents...
flips through documents, lays them out
puts document into piles
Goal: goal: work with documents...
goal: sort documents
goal: make packets...
Action: ~~action: pencilbox offers tools~~
~~action: box of staples moves over~~
action: stapler moves over, pencil box pushes a box of staples over

Figure 13: Proactive office assistance: The Object Agents system demonstrates workplace problem-solving as (a) a user's pencil box autonomously slides across their desk to retrieve and deliver a hidden adapter they need for a USB connection. (b) A stapler and a box of staples move to the person when they are organizing loose documents into packets.

table for a new dish, can be helpful as they relieve users of having to adjust many items (cf. swarm UIs [42, 65]). Object Agents may also assist users in their absence, e.g., they may clean the kitchen after cooking or reset the office after users leave a mess after a brainstorming and sketching session. We recommend that these tasks be done with personalization in mind, as different users may have different preferences; some may like to find things as they were, and some may like them to be tidied.

Providing guidance to users should include a different set of expressions. For example, guiding users through a multistep process, such as cooking or assembly, can have high utility, as the system can keep track of the upcoming steps and help users not lose overview. The items needed in the task can provide guidance directly, e.g., a screwdriver pushing the right bolt to the right side of the assembly, or a kitchen spoon indicating that the rice should be stirred. Object actions to provide guidance may be designed to express more patience to allow users to find the way on their own first.

Prevent harm or damage. Object Agents may also assist users in that they recognize imminent harm and prevent it. The consequences may vary. From a cup rotating its handle towards the user to prevent minor burns, the trivets moving into place to prevent damaging the expensive countertop (Figure 11b), a knife moving away from an inattentive user's hand to avoid a cut (Figure 11c), to car key hiding as they realize that the user was drinking and intend to drive under the influence, which may have serious, even potentially deadly consequences.

Beyond moving across surfaces. To achieve these different purposes, system designers may need to enable different physical actions. Expanding from only moving across surfaces, objects may need to detach from walls (e.g., cooking utensils), flip over (e.g., a distracting phone), jump off of surfaces (e.g, jump from a desk to a floor to move to a different room), change shape, [2]), or materials properties (e.g., to adjust to users' ergonomic needs). We note that there is typically a trade-off: the more functions an actuated system needs, the larger its form factor may be, which in turn may compromise the affordances of the original object. Beating this trade-off is very challenging, but the many advances in (soft) robotics [69] or metamaterials [32] may make these behaviors tractable in the future, but these require more exploration.

7 LIMITATIONS

There are a number of technical limitations of our current research prototype.

Camera-based sensing limitations. Sensing limitations pose challenges, particularly in cases of occlusion, unusual lighting, or novel object arrangements.

Currently, we label the objects in the scene and provide a knowledge base about their utility to our system. We provided these data because the standard object recognition was not robust enough. Additionally, small items (e.g., the box of staples) were blurry due to our camera resolution. Using higher resolution cameras and advances in object recognition models may present a solution in the near future. Furthermore, incorporating multimodal sensing such as audio could potentially disambiguate insufficient unimodal visual cues [28], and enhance reasoning with indirect speech [80]. Our current attention and intent inference relies on rather coarse analysis from camera streams. Multimodal sensing can enhance such inference by leveraging signals that correlate with user attention, such as gaze [10, 28]. Additionally, intent inference advancements [82] and uncertainty modeling frameworks [56] can be leveraged to extend the system's intent inference and ambiguity handling capabilities.

Latency of our research prototype. The current end-to-end latency of our research prototype is about 5-10s, while action outputs are refreshed every 0.3s. We currently mitigate latency with asynchronous API calls and query reduction with centralized reasoning (i.e., as opposed to a reasoning stream for every object). We expect LLM advancements such as small and efficient models [49] to reduce latency significantly in the future. During interactions, our system continuously runs its perceiving and reasoning loop. In the case of a physical action output failure (i.e., path planning fails), the perception and reasoning modules continue to seek alternative assistance opportunities, in analogy to a human trying to offer help but can't spare their hands at the moment. Although this is sufficient for most everyday scenarios, more sophisticated failure recovery can be utilized for more mission-critical scenarios.

Robotic platform limitations. Our system currently only implements movement across horizontal surfaces as physical output. The wheeled drive is not omnidirectional, meaning that, like a car, it has to turn its heading. It cannot move sideways, which may be desirable. Furthermore, action expressivity is constrained, as the current actuation integration supports a limited vocabulary of object movements. Additionally, as mentioned above, other robotic movements and physical adaptation may be desirable. For real-world deployment, waterproofing is needed for specific use scenarios. The platform needs to be waterproofed to allow users to wash their objects, e.g., kitchen utensils. Miniaturization of robotic platforms [26] is also desirable for further form integration with existing objects.

User feedback and deployment. Exploring more application scenarios through user feedback and deployment over several weeks in users' chosen environments (e.g., home, office, etc.) may reveal exciting new research directions. Although we would expect privacy concerns to arise, it would be interesting to learn about the acceptance of such a system by users. In the future, smaller multimodal LLMs [49] can be locally deployed on home hubs for privacy and computational efficiency. We are optimistic that our system and the open-source code provided will allow other researchers and system designers to expand on it and explore unobtrusive physical AI further.

Safety considerations. Safety for autonomous moving objects warrants careful consideration. As we mentioned previously, potential safety strategies include creating safety zones around users, using multimodal sensing to mitigate uncertainties [10, 28], building emergency stop features, and conservatively filtering actions for dangerous objects. In this paper, we use a robotic knife to demonstrate an Object Agent that can both move to prevent harmful situations (i.e., preventing the user's hand from being accidentally hurt by a knife) and provide assistance (i.e., by moving closer when needed, safely with the handle facing the user). Real-world deployment must carefully consider potential system errors, misuse, and malicious intent, especially for dangerous objects. Safety strategies must be investigated and tested in laboratory environments before Object Agents can be deployed. Since safety for autonomous moving systems around human users is an active research area in robotics [66] and privacy and security research, future work should build on these insights to develop further safety enhancements. We note that while actuating dangerous objects rightfully causes concern, such cases need to be investigated rather than ignored to establish ethical policies early on, similar to investigating actuating humans' bodies (e.g., [47, 67])

8 CONCLUSION

This paper introduces unobtrusive physical AI—a step towards augmenting everyday objects with intelligence and robotic motion to proactively assist users while maintaining their familiar affordances and identities. Our approach shifts the focus from robotic manipulators to the objects that already populate our environments. The core contribution of our work is a generalizable framework that enables systems to understand context, predict user intentions, and determine appropriate assistive behaviors for everyday objects to perform in the physical world. By connecting this intelligent system to robotic platforms for locomotion, we demonstrate how everyday objects can become responsive, helpful agents within their environment.

Unobtrusive physical AI reimagines everyday environments as populated with helpful, responsive entities that collaborate with users to enhance their activities. The primary value comes from the system's contextual understanding and decision-making capabilities, not merely from the physical movement of objects. As sensing and reasoning technologies continue to advance, the boundary between passive objects and intelligent agents will increasingly blur, opening new possibilities for how we interact with and within our physical environments.

ACKNOWLEDGMENTS

We thank Michael Bernstein, Yiming Zhang, Quanting Xie, Vivian Shen, Daehwa Kim, David Lindlbauer, and Yi Fei Cheng for insightful discussions. We thank Hatice Gokcen Guner and Ziru Wei for help with study recruitment.

The work in this paper was supported in part by the National Science Foundation under Awards No. IIS-2420434 and IIS-2443190.

REFERENCES

[1] Niket Agarwal, Arslan Ali, Maciej Bala, Yogesh Balaji, Erik Barker, Tiffany Cai, Prithvijit Chattopadhyay, Yongxin Chen, Yin Cui, Yifan Ding, et al. 2025. Cosmos world foundation model platform for physical ai. *arXiv preprint arXiv:2501.03575* (2025).

[2] Jason Alexander, Anne Roudaut, Jürgen Steimle, Kasper Hornbæk, Miguel Bruns Alonso, Sean Follmer, and Timothy Merritt. 2018. Grand Challenges in Shape-Changing Interface Research. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems* (Montreal QC, Canada) *(CHI '18)*. Association for Computing Machinery, New York, NY, USA, 1–14. https://doi.org/10.1145/3173574.3173873

[3] Riku Arakawa, Hiromu Yakura, and Mayank Goel. 2024. PrISM-Observer: Intervention Agent to Help Users Perform Everyday Procedures Sensed using a Smartwatch. In *Proceedings of the 37th Annual ACM Symposium on User Interface Software and Technology* (Pittsburgh, PA, USA) *(UIST '24)*. Association for Computing Machinery, New York, NY, USA, Article 15, 16 pages.

https://doi.org/10.1145/3654777.3676350

[4] V.H. Ashwin, R. Jegan, and P. Rajalakshmy. 2022. Stress Detection using Wearable Physiological Sensors and Machine Learning Algorithm. In *2022 6th International Conference on Electronics, Communication and Aerospace Technology*. 972–977. https://doi.org/10.1109/ICECA55336.2022.10009326

[5] Michael Beigl, Hans-W Gellersen, and Albrecht Schmidt. 2001. Mediacups: experience with design and use of computer-augmented everyday artefacts. *Computer Networks* 35, 4 (2001), 401–409.

[6] Ramarko Bhattacharya, Jonathan Lindstrom, Ahmad Taka, Martin Nisser, Stefanie Mueller, and Ken Nakagaki. 2024. FabRobotics: Fusing 3D Printing with Mobile Robots to Advance Fabrication, Robotics, and Interaction. In *Proceedings of the Eighteenth International Conference on Tangible, Embedded, and Embodied Interaction*. 1–13.

[7] Joost Broekens, Marcel Heerink, Henk Rosendal, et al. 2009. Assistive social robots in elderly care: a review. *Gerontechnology* 8, 2 (2009), 94–103.

[8] Sera Buyukgoz, Jasmin Grosinger, Mohamed Chetouani, and Alessandro Saffiotti. 2022. Two ways to make your robot proactive: reasoning about human intentions, or reasoning about possible futures. arXiv:2205.05492 [cs.AI] https://arxiv.org/abs/2205.05492

[9] Ruei-Che Chang, Yuxuan Liu, and Anhong Guo. 2024. WorldScribe: Towards Context-Aware Live Visual Descriptions. *ArXiv* abs/2408.06627 (2024). https://api.semanticscholar.org/CorpusID:271860261

[10] Ishan Chatterjee, Robert Xiao, and Chris Harrison. 2015. Gaze+ gesture: Expressive, precise and targeted free-space interactions. In *Proceedings of the 2015 ACM on international conference on multimodal interaction*. 131–138.

[11] Anind K Dey. 2001. Understanding and using context. *Personal and ubiquitous computing* 5 (2001), 4–7.

[12] Anind K Dey. 2018. Context-Aware Computing. In *Ubiquitous computing fundamentals*. Chapman and Hall/CRC, 335–366.

[13] Mustafa Doga Dogan, Eric J Gonzalez, Karan Ahuja, Ruofei Du, Andrea Colaço, Johnny Lee, Mar Gonzalez-Franco, and David Kim. 2024. Augmented Object Intelligence with XR-Objects. In *Proceedings of the 37th Annual ACM Symposium on User Interface Software and Technology*. 1–15.

[14] Ruofei Du, Alex Olwal, Mathieu Le Goc, Shengzhi Wu, Danhang Tang, Yinda Zhang, Jun Zhang, David Joseph Tan, Federico Tombari, and David Kim. 2022. Opportunistic interfaces for augmented reality: Transforming everyday objects into tangible 6dof interfaces using ad hoc ui. In *CHI Conference on Human Factors in Computing Systems Extended Abstracts*. 1–4.

[15] Sean Follmer, Daniel Leithinger, Alex Olwal, Akimitsu Hogge, and Hiroshi Ishii. 2013. inFORM: dynamic physical affordances and constraints through shape and object actuation.. In *Uist*, Vol. 13. Citeseer, 2501–988.

[16] Chenfeng Gao, Wanli Qian, Richard Liu, Rana Hanocka, and Ken Nakagaki. 2024. Towards Multimodal Interaction with AI-Infused Shape-Changing Interfaces. In *Adjunct Proceedings of the 37th Annual ACM Symposium on User Interface Software and Technology*. 1–3.

[17] Yunfan Gao, Yun Xiong, Xinyu Gao, Kangxiang Jia, Jinliu Pan, Yuxi Bi, Yi Dai, Jiawei Sun, Meng Wang, and Haofen Wang. 2024. Retrieval-Augmented Generation for Large Language Models: A Survey. arXiv:2312.10997 [cs.CL] https://arxiv.org/abs/2312.10997

[18] Jiaxin Ge, Zora Zhiruo Wang, Xuhui Zhou, Yi-Hao Peng, Sanjay Subramanian, Qinyue Tan, Maarten Sap, Alane Suhr, Daniel Fried, Graham Neubig, and Trevor Darrell. 2025. AutoPresent: Designing Structured Visuals from Scratch. arXiv:2501.00912 [cs.CV] https://arxiv.org/abs/2501.00912

[19] Akash Ghosh, Arkadeep Acharya, Sriparna Saha, Vinija Jain, and Aman Chadha. 2024. Exploring the Frontier of Vision-Language Models: A Survey of Current Methodologies and Future Directions. arXiv:2404.07214 [cs.CV] https://arxiv.org/abs/2404.07214

[20] Jesse T Gonzalez, Sonia Prashant, Sapna Tayal, Juhi Kedia, Alexandra Ion, and Scott E Hudson. 2023. Constraint-Driven Robotic Surfaces, At Human-Scale. In *Proceedings of the 36th Annual ACM Symposium on User Interface Software and Technology*. 1–12.

[21] Erin Griffiths, Salah Assana, and Kamin Whitehouse. 2018. Privacy-preserving image processing with binocular thermal cameras. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 1, 4 (2018), 1–25.

[22] Daniel Groeger and Jürgen Steimle. 2018. ObjectSkin: augmenting everyday objects with hydroprinted touch sensors and displays. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 1, 4 (2018), 1–23.

[23] Fuqiang Gu, Mu-Huan Chung, Mark Chignell, Shahrokh Valaee, Baoding Zhou, and Xue Liu. 2021. A survey on deep learning for human activity recognition. *ACM Computing Surveys (CSUR)* 54, 8 (2021), 1–34.

[24] Ramyad Hadidi, Jiashen Cao, Matthew Woodward, Michael S Ryoo, and Hyesoon Kim. 2018. Distributed perception by collaborative robots. *IEEE Robotics and Automation Letters* 3, 4 (2018), 3709–3716.

[25] Violet Yinuo Han, Abena Boadi-Agyemang, Yuyu Lin, David Lindlbauer, and Alexandra Ion. 2023. Parametric Haptics: Versatile Geometry-based Tactile Feedback Devices. In *Proceedings of the 36th Annual ACM Symposium on User Interface Software and Technology* (San Francisco, CA, USA) *(UIST '23)*. Association for Computing Machinery, New York, NY, USA, Article 65, 13 pages. https://doi.org/10.1145/3586183.3606766

[26] Violet Yinuo Han, Amber Yinglei Chen, Mason Zadan, Jesse T Gonzalez, Dinesh K Patel, Carmel Majidi, and Alexandra Ion. 2025. Transforming Everyday Objects into Dynamic Interfaces using Smart Flat-Foldable Structures. In *Proceedings of the 38th Annual ACM Symposium on User Interface Software and Technology*. https://doi.org/10.1145/3746059.3747720

[27] Violet Yinuo Han, Hyunsung Cho, Kiyosu Maeda, Alexandra Ion, and David Lindlbauer. 2023. Blendmr: A computational method to create ambient mixed reality interfaces. *Proceedings of the ACM on Human-Computer Interaction* 7, ISS (2023), 217–241.

[28] Violet Yinuo Han, Tianyi Wang, Hyunsung Cho, Kashyap Todi, Ajoy Savio Fernandes, Andre Levi, Zheng Zhang, Tovi Grossman, Alexandra Ion, and Tanya R Jonker. 2025. A Dynamic Bayesian Network Based Framework for Multimodal Context-Aware Interactions. In *Proceedings of the 30th International Conference on Intelligent User Interfaces*. 54–69.

[29] Vanessa Hernandez-Cruz, Xiaotong Zhang, and Kamal Youcef-Toumi. 2024. Bayesian Intention for Enhanced Human Robot Collaboration. arXiv:2410.00302 [cs.RO] https://arxiv.org/abs/2410.00302

[30] Eric Horvitz. 1999. Principles of mixed-initiative user interfaces. In *Proceedings of the SIGCHI conference on Human Factors in Computing Systems*. 159–166.

[31] Keiichi Ihara, Mehrad Faridan, Ayumi Ichikawa, Ikkaku Kawaguchi, and Ryo Suzuki. 2023. Holobots: Augmenting holographic telepresence with mobile robots for tangible remote collaboration in mixed reality. In *Proceedings of the 36th Annual ACM Symposium on User Interface Software and Technology*. 1–12.

[32] Alexandra Ion, Johannes Frohnhofen, Ludwig Wall, Robert Kovacs, Mirela Alistar, Jack Lindsay, Pedro Lopes, Hsiang-Ting Chen, and Patrick Baudisch. 2016. Metamaterial Mechanisms. In *Proceedings of the 29th Annual Symposium on User Interface Software and Technology* (Tokyo, Japan) *(UIST '16)*. Association for Computing Machinery, New York, NY, USA, 529–539. https://doi.org/10.1145/2984511.2984540

[33] Hiroshi Ishii, Dávid Lakatos, Leonardo Bonanni, and Jean-Baptiste Labrune. 2012. Radical atoms: beyond tangible bits, toward transformable materials. *interactions* 19, 1 (2012), 38–51.

[34] Hiroshi Ishii, Daniel Leithinger, Sean Follmer, Amit Zoran, Philipp Schoessler, and Jared Counts. 2015. TRANSFORM: Embodiment of" Radical Atoms" at Milano Design Week. In *Proceedings of the 33rd Annual ACM Conference Extended Abstracts on Human Factors in Computing Systems*. 687–694.

[35] Hiroshi Ishii and Brygg Ullmer. 1997. Tangible bits: towards seamless interfaces between people, bits and atoms. In *Proceedings of the ACM SIGCHI Conference on Human factors in computing systems*. 234–241.

[36] Raghav Jain, Daivik Sojitra, Arkadeep Acharya, Sriparna Saha, Adam Jatowt, and Sandipan Dandapat. 2023. Do Language Models Have a Common Sense regarding Time? Revisiting Temporal Commonsense Reasoning in the Era of Large Language Models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, Houda Bouamor, Juan Pino, and Kalika Bali (Eds.). Association for Computational Linguistics, Singapore, 6750–6774. https://doi.org/10.18653/v1/2023.emnlp-main.418

[37] Seungwoo Je, Hyunseung Lim, Kongpyung Moon, Shan-Yuan Teng, Jas Brooks, Pedro Lopes, and Andrea Bianchi. 2021. Elevate: A walkable pin-array for large shape-changing terrains. In *Proceedings of the 2021 CHI Conference on human Factors in Computing Systems*. 1–11.

[38] Haojian Jin, Boyuan Guo, Rituparna Roychoudhury, Yaxing Yao, Swarun Kumar, Yuvraj Agarwal, and Jason I Hong. 2022. Exploring the needs of users for supporting privacy-protective behaviors in smart homes. In *Proceedings of the 2022 CHI Conference on Human Factors in Computing Systems*. 1–19.

[39] Glenn Jocher and Jing Qiu. 2024. *Ultralytics YOLO11*. https://github.com/ultralytics/ultralytics

[40] Gerd Kortuem, Fahim Kawsar, Vasughi Sundramoorthy, and Daniel Fitton. 2009. Smart objects as building blocks for the internet of things. *IEEE internet computing* 14, 1 (2009), 44–51.

[41] Gierad Laput, Yang Zhang, and Chris Harrison. 2017. Synthetic sensors: Towards general-purpose sensing. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems*. 3986–3999.

[42] Mathieu Le Goc, Lawrence H. Kim, Ali Parsaei, Jean-Daniel Fekete, Pierre Dragicevic, and Sean Follmer. 2016. Zooids: Building Blocks for Swarm User Interfaces. In *Proceedings of the 29th Annual Symposium on User Interface Software and Technology* (Tokyo, Japan) *(UIST '16)*. Association for Computing Machinery, New York, NY, USA, 97–109. https://doi.org/10.1145/2984511.2984547

[43] Jan Leike, David Krueger, Tom Everitt, Miljan Martic, Vishal Maini, and Shane Legg. 2018. Scalable agent alignment via reward modeling: a research direction. arXiv:1811.07871 [cs.LG] https://arxiv.org/abs/1811.07871

[44] Iolanda Leite, Carlos Martinho, and Ana Paiva. 2013. Social robots for long-term interaction: a survey. *International Journal of Social Robotics* 5 (2013), 291–308.

[45] Yuyu Lin, Jesse T Gonzalez, Zhitong Cui, Yash Rajeev Banka, and Alexandra Ion. 2024. ConeAct: A Multistable Actuator for Dynamic Materials. In *Proceedings of the 2024 CHI Conference on Human Factors in Computing Systems*. 1–16.

[46] David Lindlbauer, Anna Maria Feit, and Otmar Hilliges. 2019. Context-Aware Online Adaptation of Mixed Reality Interfaces. In *Proceedings of the 32nd Annual*

ACM Symposium on User Interface Software and Technology (New Orleans, LA, USA) *(UIST '19)*. Association for Computing Machinery, New York, NY, USA, 147–160. https://doi.org/10.1145/3332165.3347945

[47] Pedro Lopes, Alexandra Ion, and Patrick Baudisch. 2015. Impacto: Simulating Physical Impact by Combining Tactile Stimulation with Electrical Muscle Stimulation. In *Proceedings of the 28th Annual ACM Symposium on User Interface Software amp; Technology* (Charlotte, NC, USA) *(UIST '15)*. Association for Computing Machinery, New York, NY, USA, 11–19. https://doi.org/10.1145/2807442.2807443

[48] Hamza Mahdi, Sami Alperen Akgun, Shahed Saleh, and Kerstin Dautenhahn. 2022. A survey on the design and evolution of social robots—Past, present and future. *Robotics and Autonomous Systems* 156 (2022), 104193.

[49] Andrés Marafioti, Orr Zohar, Miquel Farré, Merve Noyan, Elie Bakouch, Pedro Cuenca, Cyril Zakka, Loubna Ben Allal, Anton Lozhkov, Nouamane Tazi, Vaibhav Srivastav, Joshua Lochner, Hugo Larcher, Mathieu Morlon, Lewis Tunstall, Leandro von Werra, and Thomas Wolf. 2025. SmolVLM: Redefining small and efficient multimodal models. arXiv:2504.05299 [cs.AI] https://arxiv.org/abs/2504.05299

[50] Ken Nakagaki, Jordan L Tappa, Yi Zheng, Jack Forman, Joanne Leong, Sven Koenig, and Hiroshi Ishii. 2022. (Dis) Appearables: A concept and method for actuated tangible UIs to appear and disappear based on stages. In *Proceedings of the 2022 CHI conference on human factors in computing systems*. 1–13.

[51] Joon Sung Park, Joseph O'Brien, Carrie Jun Cai, Meredith Ringel Morris, Percy Liang, and Michael S. Bernstein. 2023. Generative Agents: Interactive Simulacra of Human Behavior. In *Proceedings of the 36th Annual ACM Symposium on User Interface Software and Technology* (San Francisco, CA, USA) *(UIST '23)*. Association for Computing Machinery, New York, NY, USA, Article 2, 22 pages. https://doi.org/10.1145/3586183.3606763

[52] Ivan Poupyrev. 2023. The Ultimate Interface. In *Adjunct Proceedings of the 36th Annual ACM Symposium on User Interface Software and Technology* (San Francisco, CA, USA) *(UIST '23 Adjunct)*. Association for Computing Machinery, New York, NY, USA, Article 111, 2 pages. https://doi.org/10.1145/3586182.3624511

[53] Erwin Prassler, Rainer Bischoff, Wolfram Burgard, Robert Haschke, Martin Hägele, Gisbert Lawitzky, Bernhard Nebel, Paul Plöger, Ulrich Reiser, and Marius Zöllner. 2012. Towards service robots for everyday environments: recent advances in designing service robots for complex tasks in everyday environments. (2012).

[54] Vinitha Ranganeni, Vy Nguyen, Henry Evans, Jane Evans, Julian Mehu, Samuel Olatunji, Wendy Rogers, Aaron Edsinger, Charles Kemp, and Maya Cakmak. 2024. Robots for humanity: in-home deployment of Stretch RE2. In *Companion of the 2024 ACM/IEEE international conference on human-robot interaction*. 1299–1301.

[55] Dominik Riedelbauch, Nico Höllerich, and Dominik Henrich. 2023. Benchmarking teamwork of humans and cobots—an overview of metrics, strategies, and tasks. *IEEE Access* 11 (2023), 43648–43674.

[56] Julia Schwarz, Scott Hudson, Jennifer Mankoff, and Andrew D Wilson. 2010. A framework for robust and flexible handling of inputs with uncertainty. In *Proceedings of the 23nd annual ACM symposium on User interface software and technology*. 47–56.

[57] Julia Schwarz, Charles Claudius Marais, Tommer Leyvand, Scott E. Hudson, and Jennifer Mankoff. 2014. Combining body pose, gaze, and gesture to determine intention to interact in vision-based interfaces. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (Toronto, Ontario, Canada) *(CHI '14)*. Association for Computing Machinery, New York, NY, USA, 3443–3452. https://doi.org/10.1145/2556288.2556989

[58] Orit Shaer, Eva Hornecker, et al. 2010. Tangible user interfaces: past, present, and future directions. *Foundations and Trends® in Human–Computer Interaction* 3, 1–2 (2010), 4–137.

[59] Ben Shneiderman and Pattie Maes. 1997. Direct manipulation vs. interface agents. *Interactions* 4, 6 (Nov. 1997), 42–61. https://doi.org/10.1145/267505.267514

[60] Ishika Singh, Valts Blukis, Arsalan Mousavian, Ankit Goyal, Danfei Xu, Jonathan Tremblay, Dieter Fox, Jesse Thomason, and Animesh Garg. 2023. Progprompt: Generating situated robot task plans using large language models. In *2023 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 11523–11530.

[61] Sanjana Srivastava, Chengshu Li, Michael Lingelbach, Roberto Martín-Martín, Fei Xia, Kent Elliott Vainio, Zheng Lian, Cem Gokmen, Shyamal Buch, Karen Liu, et al. 2022. Behavior: Benchmark for everyday household activities in virtual, interactive, and ecological environments. In *Conference on robot learning*. PMLR, 477–490.

[62] Ryo Suzuki, Mar Gonzalez-Franco, Misha Sra, and David Lindlbauer. 2023. Xr and ai: Ai-enabled virtual, augmented, and mixed reality. In *Adjunct Proceedings of the 36th Annual ACM Symposium on User Interface Software and Technology*. 1–3.

[63] Ryo Suzuki, Hooman Hedayati, Clement Zheng, James L Bohn, Daniel Szafir, Ellen Yi-Luen Do, Mark D Gross, and Daniel Leithinger. 2020. Roomshift: Room-scale dynamic haptics for vr with furniture-moving swarm robots. In *Proceedings of the 2020 CHI conference on human factors in computing systems*. 1–11.

[64] Ryo Suzuki, Eyal Ofek, Mike Sinclair, Daniel Leithinger, and Mar Gonzalez-Franco. 2021. Hapticbots: Distributed encountered-type haptics for vr with multiple shape-changing mobile robots. In *The 34th Annual ACM Symposium on User Interface Software and Technology*. 1269–1281.

[65] Ryo Suzuki, Clement Zheng, Yasuaki Kakehi, Tom Yeh, Ellen Yi-Luen Do, Mark D. Gross, and Daniel Leithinger. 2019. ShapeBots: Shape-changing Swarm Robots. In *Proceedings of the 32nd Annual ACM Symposium on User Interface Software and Technology* (New Orleans, LA, USA) *(UIST '19)*. Association for Computing Machinery, New York, NY, USA, 493–505. https://doi.org/10.1145/3332165.3347911

[66] Tadele Shiferaw Tadele, Theo de Vries, and Stefano Stramigioli. 2014. The safety of domestic robotics: A survey of various safety-related publications. *IEEE robotics & automation magazine* 21, 3 (2014), 134–142.

[67] Yudai Tanaka, Jacob Serfaty, and Pedro Lopes. 2024. Haptic Source-Effector: Full-Body Haptics via Non-Invasive Brain Stimulation. In *Proceedings of the 2024 CHI Conference on Human Factors in Computing Systems* (Honolulu, HI, USA) *(CHI '24)*. Association for Computing Machinery, New York, NY, USA, Article 411, 15 pages. https://doi.org/10.1145/3613904.3642483

[68] Yunlong Tang, Jing Bi, Siting Xu, Luchuan Song, Susan Liang, Teng Wang, Daoan Zhang, Jie An, Jingyang Lin, Rongyi Zhu, et al. 2025. Video understanding with large language models: A survey. *IEEE Transactions on Circuits and Systems for Video Technology* (2025).

[69] Michael T. Tolley, Robert F. Shepherd, Michael Karpelson, Nicholas W. Bartlett, Kevin C. Galloway, Michael Wehner, Rui Nunes, George M. Whitesides, and Robert J. Wood. 2014. An untethered jumping soft robot. In *2014 IEEE/RSJ International Conference on Intelligent Robots and Systems*. 561–566. https://doi.org/10.1109/IROS.2014.6942615

[70] Daniel Ullman and Bertram F. Malle. 2018. What Does it Mean to Trust a Robot? Steps Toward a Multidimensional Measure of Trust. In *Companion of the 2018 ACM/IEEE International Conference on Human-Robot Interaction* (Chicago, IL, USA) *(HRI '18)*. Association for Computing Machinery, New York, NY, USA, 263–264. https://doi.org/10.1145/3173386.3176991

[71] Marike Koch van Den broek and Thomas B. Moeslund. 2024. What is Proactive Human-Robot Interaction? - A Review of a Progressive Field and Its Definitions. *J. Hum.-Robot Interact.* 13, 4, Article 49 (Sept. 2024), 30 pages. https://doi.org/10.1145/3650117

[72] Keru Wang, Zhu Wang, Ken Nakagaki, and Ken Perlin. 2024. "Push-That-There": Tabletop Multi-robot Object Manipulation via Multimodal'Object-level Instruction'. In *Proceedings of the 2024 ACM Designing Interactive Systems Conference*. 2497–2513.

[73] Zhiqiang Wang, Jiahui Hou, Guangyu Wu, Suyuan Liu, Puhan Luo, and Xiangyang Li. 2024. Efficient Task-driven Video Data Privacy Protection for Smart Camera Surveillance System. *ACM Transactions on Sensor Networks* 20, 4 (2024), 1–21.

[74] Zhiruo Wang, Shuyan Zhou, Daniel Fried, and Graham Neubig. 2023. Execution-Based Evaluation for Open-Domain Code Generation. arXiv:2212.10481 [cs.SE] https://arxiv.org/abs/2212.10481

[75] Carlo Weidemann, Nils Mandischer, Frederick van Kerkom, Burkhard Corves, Mathias Hüsing, Thomas Kraus, and Cyryl Garus. 2023. Literature review on recent trends and perspectives of collaborative robotics in work 4.0. *Robotics* 12, 3 (2023), 84.

[76] Mark Weiser. 1999. The computer for the 21st century. *SIGMOBILE Mob. Comput. Commun. Rev.* 3, 3 (1999), 3–11. https://doi.org/10.1145/329124.329126

[77] Zhiheng Xi, Wenxiang Chen, Xin Guo, Wei He, Yiwen Ding, Boyang Hong, Ming Zhang, Junzhe Wang, Senjie Jin, Enyu Zhou, Rui Zheng, Xiaoran Fan, Xiao Wang, Limao Xiong, Yuhao Zhou, Weiran Wang, Changhao Jiang, Yicheng Zou, Xiangyang Liu, Zhangyue Yin, Shihan Dou, Rongxiang Weng, Wensen Cheng, Qi Zhang, Wenjuan Qin, Yongyan Zheng, Xipeng Qiu, Xuanjing Huang, and Tao Gui. 2023. The Rise and Potential of Large Language Model Based Agents: A Survey. arXiv:2309.07864 [cs.AI]

[78] John Yang, Carlos Jimenez, Alexander Wettig, Kilian Lieret, Shunyu Yao, Karthik Narasimhan, and Ofir Press. 2024. Swe-agent: Agent-computer interfaces enable automated software engineering. *Advances in Neural Information Processing Systems* 37 (2024), 50528–50652.

[79] Chih-Ta Yen, Jia-Xian Liao, and Yi-Kai Huang. 2020. Human Daily Activity Recognition Performed Using Wearable Inertial Sensors Combined With Deep Learning Algorithms. *IEEE Access* 8 (2020), 174105–174114. https://doi.org/10.1109/ACCESS.2020.3025938

[80] Yan Zhang, Tharaka Sachintha Ratnayake, Cherie Sew, Jarrod Knibbe, Jorge Goncalves, and Wafa Johal. 2025. Can you pass that tool?: Implications of indirect speech in physical human-robot collaboration. In *Proceedings of the 2025 CHI Conference on Human Factors in Computing Systems*. 1–18.

[81] Zirui Zhao, Wee Sun Lee, and David Hsu. 2023. Large language models as commonsense knowledge for large-scale task planning. *Advances in Neural Information Processing Systems* 36 (2023), 31967–31987.

[82] Jincao Zhou, Xuezhong Su, Weiping Fu, Yang Lv, and Bo Liu. 2024. Enhancing intention prediction and interpretability in service robots with LLM and KG. *Scientific Reports* 14, 1 (2024), 26999.

[83] Shuyan Zhou, Frank F. Xu, Hao Zhu, Xuhui Zhou, Robert Lo, Abishek Sridhar, Xianyi Cheng, Tianyue Ou, Yonatan Bisk, Daniel Fried, Uri Alon, and Graham Neubig. 2024. WebArena: A Realistic Web Environment for Building Autonomous Agents. arXiv:2307.13854 [cs.AI] https://arxiv.org/abs/2307.13854