Priors of U: Dynamic Preference Learning for Embodied Interaction

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

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

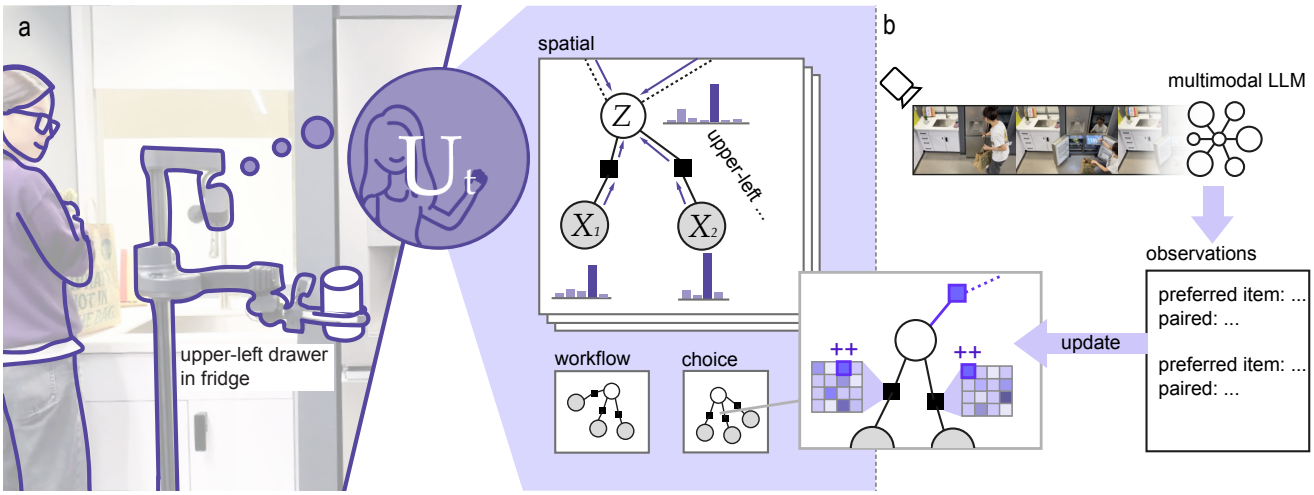

Figure 1: We propose learning user models from passive observation to personalize embodied interactions. (a) Our user model enables embodied agents to personalize assistance across three preference dimensions (spatial, choice, and workflow), such as predicting preferred storage locations for grocery items through structured probabilistic inference. (b) Our framework pairs a multimodal LLM with factor graphs to dynamically update the user model from video observations.

Abstract

People have their own habits; from how they organize their workspace, over which mug they reach for in the morning, to how they prepare for cooking. Such preferences are context-dependent, dynamic, and often difficult to articulate. As embodied AI agents, such as robots, enter personal spaces, effective interaction requires understanding these implicit and evolving preferences; otherwise, such systems risk being more hindering than helpful. We propose an architecture that learns user preferences from real-world behavioral observation without requiring explicit instruction or feedback. We introduce *a factor-graph-based user model* that captures preferences across spatial arrangement, tool selection, and workflow, and *continuously adapts* as tasks and contexts change. The model expands with new observations, enabling open-ended, in situ learning of user-specific preference patterns. Our evaluation shows accurate, calibrated predictions across domains and robust adaptation over time. We demonstrate personalized robot assistance scenarios illustrating seamless and context-aware human–agent interaction.

UIST '26, Detroit, MI, USA

ACM ISBN 979-8-4007-2856-3/26/11
https://doi.org/10.1145/3830398.3830688

CCS Concepts

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

Keywords

Personalization, Embodied Agents, User Modeling, Probabilistic Graph Models

ACM Reference Format:
Violet Yinuo Han and Alexandra Ion. 2026. Priors of U: Dynamic Preference Learning for Embodied Interaction. In *The 39th Annual ACM Symposium on User Interface Software and Technology (UIST '26), November 02–05, 2026, Detroit, MI, USA.* ACM, New York, NY, USA, 14 pages. https://doi.org/10.1145/3830398.3830688

1 Introduction

Everyone has their own way of doing things: from how they organize their closet, to the coffee mug they reach for every morning, or their preparation habits during cooking. These preferences shape how our everyday tasks flow and how our spaces feel. They are context-dependent, dynamic, and often difficult to articulate, and when someone gets them wrong, we notice immediately.

Building an understanding of users is not a new challenge. HCI research has a long history of adapting *digital* interfaces and interactions to individual users for better user experience, from adaptive interfaces that adjust to individual capabilities [14], to recommender

systems that uncover preferences from interaction traces [29], and more recently, constructing general user models from screen-based activities across applications [55]. A consistent effort has been to build user understanding that enables dynamic adaptation to users, without requiring much explicit user input.

As intelligent agents move from digital interfaces into physical and personal spaces (e.g., home robots), this challenge becomes both more critical and more complex. Embodied agents must not only complete tasks, but do so in ways that align with individual preferences. A robot that organizes a closet in a way that the user can no longer find their favorite shirts, brings a drink the user dislikes, or cleans up in a way that disrupts the user's workflow becomes *hindrance rather than help*. In the physical space, preferences span a richer space, covering the many objects a user owns, the tasks they perform, and the diverse conditions under which they do them.

Robotics research has explored learning user preferences through learning reward functions from demonstrations [42], active querying [42, 60], and explicit specification of goals and constraints [19]. These approaches rely heavily on user involvement and typically target narrow task-level adaptations, but explicit preferences are *difficult to articulate* and *burdensome to provide for every task*.

Instead, we propose passive, behavior-driven preference learning from observations. Humans routinely infer such preferences through observation [27]. We propose that embodied agents should do the same. A preference is commonly defined as favoring one option over another. Our key insight is that *everyday actions already encode preferences*: repeated tool choices, consistent object placements, and habitual workflows provide implicit signals of how users prefer tasks to be performed. Because preferences are internal and not directly observable, we draw on principles from behavioral economics [53] and cognitive science [27], which infer preferences from observed behavior. We adapt this idea to model user preferences by treating user actions and choices within observed contextual factors as *probabilistic evidence* of *context-dependent preferences* in *dynamic settings*.

We present Priors of U, a framework for learning *multidimensional, dynamic, and conditional user preferences* from *passive behavioral observation*. Our framework pairs a perception module that watches users and extracts structured observations with a factor-graph-based user model that accumulates these observations into calibrated probabilistic beliefs. We explicitly separate perception and modeling: a multimodal LLM extracts structured observations of behavioral and contextual factors from videos, while factor graphs model beliefs about user preferences, updating dynamically as observations accumulate. *The user model lives in the factor graphs rather than LLM context*, so preference beliefs stay interpretable, carry contextual dependencies and calibrated uncertainty, and support probabilistic inference. Our user model captures preferences across spatial arrangements, tool selection, and workflow, continuously adapts as tasks and contexts change, and expands as new patterns emerge, enabling open-ended in situ learning of how individual users prefer to do things.

We evaluate Priors of U in a controlled desktop setting with six participants performing tasks across three domains (assembly, office, and crafting). The system achieves 89% prediction accuracy with well-calibrated confidence, substantially outperforming LLM-based and count-based alternatives. It dynamically adapts when users change their behaviors, expands over time and domains, and learns different preference profiles across participants. We demonstrate personalized robot assistance scenarios to illustrate how preference-aware agents enhance user experience in daily tasks, yet extending this model to unconstrained real-world environments remains future work (Section 8). Specifically, this paper **contributes**:

(1) A framework for learning multidimensional user preferences from passive behavioral observation, combining vision-language models for perception with factor graphs for structured probabilistic modeling, updating, and inference.
(2) A system instantiation that learns spatial, choice, and workflow preferences.
(3) An evaluation showing accurate, calibrated, and adaptive preference learning across six participants and three domains.
(4) Application scenarios showing how our user model benefits use.

2 Related Work

Our work builds on user modeling and personalization, embodied and physical interfaces, and preference learning in robotics.

2.1 User Modeling and Personalization

HCI has a long tradition of building computational models of users to adapt interactive systems to individual needs. Early work established core approaches including stereotype-based models [52] and Bayesian inference over user goals [25]. On the systems side, SUPPLE demonstrated that user models built from interaction traces could drive personalized interface generation through constraint optimization [14], and Gajos et al. extended this to accessibility by adapting interfaces to individual motor and vision capabilities [15].

More recently, user modeling and personalization have expanded across interaction domains: spatial layout adaptation in mixed reality [58], graspable interface preferences [8], context-dependent recommendations [24], touchscreen accessibility [43], systems that adapt to individual user paces [16], and real-time intent inference from multimodal input via a dynamic Bayesian network paired with an LLM [22]. A consistent theme is minimizing user effort, leveraging signals from implicit sources such as interaction history [14, 39]. To build user models that can be shared across digital applications, Shaikh et al. introduced General User Models that use large language models to generate propositions about users from their computer use, enabling agent assistance grounded with personalized context [55].

These systems operate primarily in *digital environments*, where interaction traces are structured and directly observable. Extending user modeling to embodied interaction introduces distinct challenges: physical environments are *cluttered and open-ended*, behavioral signals are *less structured and harder to extract*, and stubbornly incorrect predictions carry *physical consequences* that *diminish users' experiences in their everyday spaces*.

2.2 Embodied and Physical Interfaces

HCI research has long explored how computational interaction can be embodied by physical interfaces rather than confined to screens. Tangible interfaces pioneered this vision by coupling digital information with physical interfaces [26], and subsequent work extended computational interaction to shape-changing interfaces [13], architectural spaces [18], furniture [56], and robotically augmented everyday objects that locomote [20] and manipulate [23].

More recently, researchers have brought agency into these physical interfaces. Han et al. explored how everyday objects augmented with intelligence and robotic movement can proactively assist users [21]. Kari and Abtahi designed robotic assistance that feels invisible, letting users enjoy help without attending to the robot [32]. Jeung et al. enabled end-user authoring of interactions for tabletop swarm robots [28]. Mahadevan et al. studied implicit and explicit methods for allocating tasks between humans and robots [37].

These systems demonstrate that embodied agents can perceive context and assist people in *shared physical spaces*. Yet the ability to act in shared physical spaces also carries stakes, as robots that assist in ways the user does not prefer may become a hindrance rather than a help. Current systems lack the ability to *learn what a user likes* with *minimal user burden* and to *continuously adapt* to them in everyday life.

2.3 Adapting Robot Behaviors to Users

A growing body of robotics research addresses how robots can adapt their behavior to individual users, from low-level actions such as how to hand over objects [44] and organize users' things [1, 31, 49, 63], to higher-level decisions such as how to allocate tasks in collaboration [40, 41] and personalized task planning [19].

Researchers have explored a range of learning signals to adapt robots to humans, including physical corrections [2], diverse feedback [7], active querying via language [60], and language-based goal specification before tasks [19].

More recently, LLM-based approaches have attempted to leverage accumulated interaction memories for personalized embodied assistance, though Kwon et al. [35] found that even frontier LLMs struggle with this, particularly for sequential user patterns. Closest to our philosophy, Shah et al. [54] argue that environments already encode human preferences since humans have been optimizing them. We share this insight but extend it from *static world states* to *dynamic behavioral streams*, learning not just what users prefer but how those preferences *shift with context* and *evolve over time*.

3 Framework

Our framework enables embodied agents to learn user preferences from passive behavioral observation. It is built on two core ideas: First, that *people's natural behaviors already contain evidence for their preferences*. Second, *we separate perception* (observations of behaviors and context) *and modeling* (using observations as evidence to dynamically update beliefs in a structured probabilistic manner).

We pair a perception module with a modeling module (Figure 2). The perception module watches the user and extracts structured behavioral and contextual observations, such as what items were chosen, where things were placed, and how tasks were sequenced (Figure 2a). The modeling module takes these observations and maintains a structured probabilistic user model using factor graphs (Figure 2b). This dual approach combines the strengths of state-of-the-art vision foundation models, including spatial understanding [33, 59], video captioning [64], activity recognition [67], and open-vocabulary generalization [48], with factor graphs' capacity for structured probabilistic inference [34], efficient belief updates from sparse evidence, and explicit representation of uncertainty over inferred preferences.

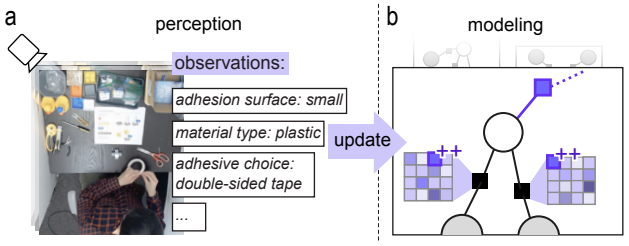

Figure 2: We pair a perception module with a modeling module. (a) The perception module captures behavioral and contextual observations, (b) updating the modeling module that consists of factor graphs dynamically encoding users' dynamic and conditional preferences.

The perception module watches user actions and extracts two things: the choices a user makes among alternatives (e.g., tools, workflow), and the contextual factors present when each choice occurs (Figure 2a). We take such an observation as *evidence* for a situational preference and store it in the factor graph with its contextual factors, strengthening or weakening the corresponding belief conditioned on the factors (Figure 2b). Beliefs form and update dynamically as evidence accumulates.

3.1 Background: Factor Graphs

Central to our framework are factor graphs that encode, update, and infer user preferences. A factor graph is a bipartite probabilistic graphical model that factorizes a joint distribution over variables into a product of local functions.

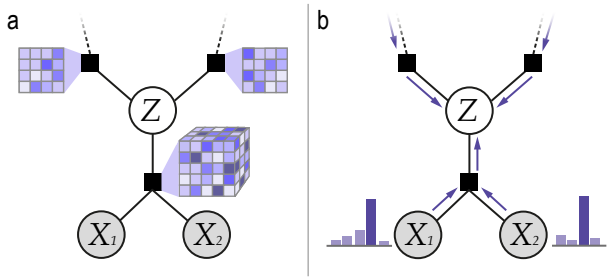

Figure 3: (a) A factor graph consists of variables (round nodes) and factors (black squares), local functions over the variables they connect. During inference, belief propagation takes observed variables' (grey) distributions as evidence and computes the marginal distribution of the latent variable (white).

The graph contains two types of nodes: variable nodes (circles) and factor nodes (squares), as shown in Figure 3. Variable nodes represent quantities of interest, either latent variables whose values we want to infer (e.g., a user's preferred cutting tool) or observable variables whose values we can see (e.g., the material being cut). Factors encode local relationships between connected variables (Figure 3a).

At query time, belief propagation [34] passes messages along edges between factors and variables, combining local evidence to produce a globally coherent prediction, turning the graph from a static representation into an inference engine (Figure 3b).

Rather than treating a user's preferences as a single opaque estimate, we decompose them into *local contextual relationships*, connecting a user's tool choice to, e.g., the material being worked on, and let belief propagation combine these to arrive at contextual preference predictions.

Factor graphs have been widely used in robotics and computer vision for SLAM [12], optimal control [65], and motion tracking [46]. We adapt them for *user preference modeling*, where their support for incremental updates, modular structure, and tractable inference over sparse evidence allows for maintaining evolving, multidimensional preference representations from behavioral observations.

3.2 Preference Dimensions

To determine what our user model should capture, we draw on prior work in human-robot interaction and identify seven preference dimensions for physical assistance, organized into four categories, that we illustrate in Figure 4.

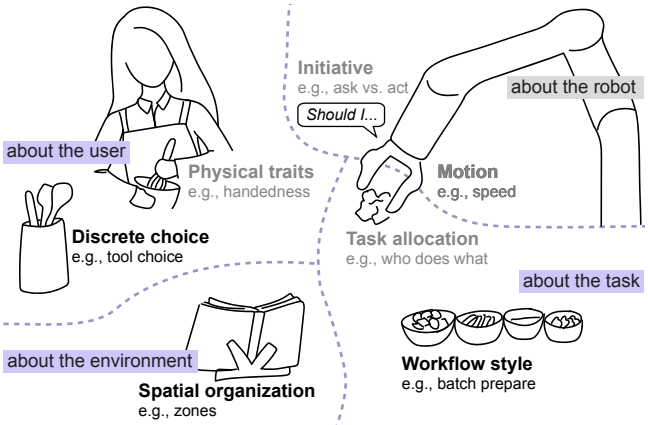

Figure 4: Seven preference dimensions for physical assistance across four categories. We instantiate the three that are observable from passive video: discrete choice, spatial organization, and workflow style.

Preferences about the user include **physical traits** such as handedness and motion capabilities [10], which may be used to adjust actions such as handing an object to the user's dominant hand. **Discrete choices** include selecting among alternative items that serve the same purpose, such as choosing one tool over another [35, 57].

Preferences about the environment include **spatial organization** preferences such as rules for organization [31, 63] and preferred storage locations [1]. Prior work has highlighted spatial preferences as a recurring theme for personalization in home robotics [1, 31, 49, 63]. We extend this dimension to capture spatial preferences that vary with task and context.

Preferences about the task include **workflow style** preferences concerning how a user executes a task [17], such as preparation and cleanup patterns, and **task allocation preferences** concerning which subtask a user prefers to delegate to a robot [41, 66].

Preferences about the robot include **initiative** preferences capturing whether and when a robot should act proactively [3] and **motion preferences** capturing desired movement trajectories [9] and speed [4].

In this work, we instantiate our framework on three dimensions that are directly observable from passive video: discrete choice, spatial organization, and workflow style. The integration of other preference dimensions would require observing human-robot interactions and is interesting future work that our system can subsequently be extended to (Section 8).

3.3 User Model Life Cycle

Our user model attunes to a user over time. It accumulates evidence from observations, expands its representation as new patterns emerge, adapts when user behavior changes or it makes a mistake, and recognizes when it does not yet know enough to act.

3.3.1 Factor graph initiation. As shown in Figure 5, each factor graph is organized around a question the system will need to answer, such as, given the observed current context, "Where does this item go?", "What tool is preferred among the alternatives?", and "Does this user prefer to work sequentially, or batch repetitive actions?"

Identifying such a question determines a latent variable Z and observed context variables $X_1, \ldots, X_m$. The graph connects Z to these variables through factors (Figure 5).

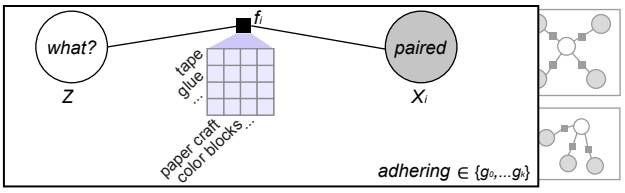

Figure 5: Each factor graph is initiated with a latent variable encoding a question, e.g., what adhesive is preferred (under context), connected to observable variables with factors.

3.3.2 Learning from observation. As the system observes the user making choices in context, each factor updates as a smoothed co-occurrence count (Figure 6):

$$f_i(z, x_i) = n_{z,x_i} + \alpha,$$

where x_i and z are the values taken by X_i and Z respectively, n_{z,x_i} is the number of times the user chose z in such context, and α is a Laplace smoothing pseudocount ensuring the model never assigns zero probability to unseen combinations.

3.3.3 Growing with new context. A fixed set of context variables cannot capture what matters for every user in every situation. Our framework allows the perception module to propose new context

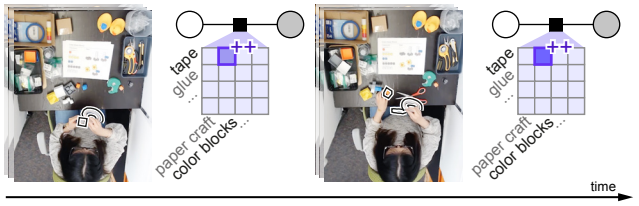

Figure 6: Observations of user behaviors update factor graphs, forming and updating beliefs.

variables at runtime. When it notices a pattern not yet represented, such as material type influencing tool choice, it proposes a new variable X_{m+1}. The factor graph expands by adding the new observable variable X_{m+1} and a new factor f_{m+1} connecting Z to X_{m+1}, initialized with uniform pseudocounts (Figure 7a). Subsequent observations populate this factor, and the new variable participates in inference alongside existing ones. Growth also occurs within a context variable, by adding values not seen before, such as a new material, to the variable's domain (Figure 7b).

If two context variables encode the same concept, their factors multiply the same signal during belief propagation, counting the same evidence twice and risking making the system overconfident. Therefore, we constrain the perception module to only propose variables capturing distinct concepts.

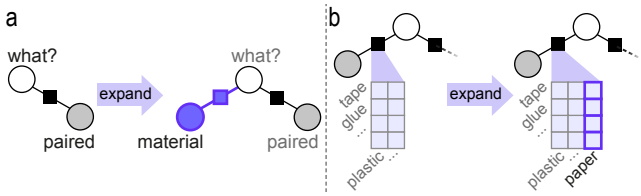

Figure 7: To capture dynamic context, our factor graphs expand with (a) new factors and observable variables, and (b) expansion of existing factors.

3.3.4 Adapting to change. Users change their preferences over time. A previously preferred coffee mug may be replaced by a new favorite. If our system has observed the user selecting the old mug many times, it will have accumulated strong factor weights favoring that mug and would need a comparable number of new observations to shift toward the replacement under naive updating. To enable quick adaptation in these cases, we design a contrastive update mechanism that, when a new item z^* is selected in context $\mathbf{x} = (x_1, \ldots, x_m)$, penalizes the alternative z' previously preferred in that same context $\mathbf{x}$ by a step size δ, floored at a minimum value α, allowing the user model to adapt quickly to shifting preferences (Figure 8):

$$f_i(z', x_i) \leftarrow \max(\alpha,\ f_i(z', x_i) - \delta)$$

where x_i is the value X_i takes in $\mathbf{x}$, and z''s weight is reduced in every factor f_i.

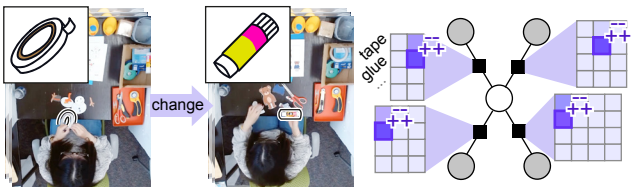

Figure 8: To quickly adapt to a change in the same observed context, our system penalizes a previously preferred item in favor of the newly preferred item.

3.3.5 Error-driven adaptive updates. When the user model produces a confident prediction $\hat{z}$ that disagrees with the user's observed choice z_{obs}, it needs to recover quickly (Figure 9a). We take an incorrect prediction as another observation opportunity (Figure 9b), as the system can see what the user actually does, which reveals the correct preference. We exploit this by comparing each prediction against the subsequent observation and, when $\hat{z} \neq z_{\text{obs}}$ with high confidence, applying an amplified penalty λ to the incorrectly predicted item and reinforcement ω to the observed one (Figure 9c), both scaled by $2/K$ (where K is the number of alternatives) to prevent the model from oscillating between alternatives in larger domains.

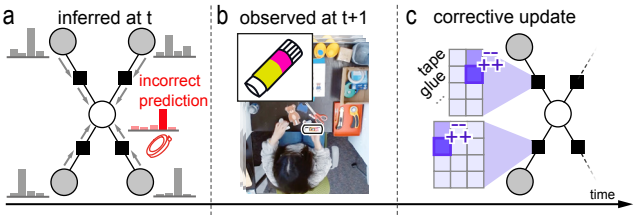

Figure 9: (a) When the user model confidently outputs an incorrect preference prediction, (b) we use the next observation to (c) correct relevant factors.

3.3.6 Knowing when it does not know. Embodied agents should ask for help when uncertain [51]. Our use of structured probabilistic representations allows the system to recognize when it lacks sufficient evidence to make a reliable prediction. First, we gate factors by evidence, letting a factor f_i participate in belief propagation only if the observed variables in its scope have been updated by evidence. Without this, unevidenced factors would contribute their marginal counts to the posterior, creating false certainty from a factor's overall marginal rather than the current context.

Second, each system output is a full marginal distribution over the latent variable given observed evidence. After belief propagation, the system computes the normalized entropy of this distribution:

$$\bar{H} = \frac{-\sum_z P(z \mid \mathbf{x}) \log P(z \mid \mathbf{x})}{\log K}$$

where K is the number of possible values of the latent variable Z (i.e., the size of its domain).

When uncertainty is detected (entropy exceeding a threshold $\bar{H} > \theta_H$), the system abstains rather than guessing (Figure 10).

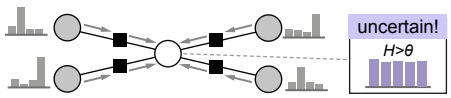

Figure 10: Our structured probabilistic modeling approach allows us to detect when our user model is uncertain via entropy of the latent variable.

For example, this is critical during cold start, when only a few observations are available; after an abrupt preference change, before the model has caught up; or when the available context is genuinely ambiguous (e.g., a new material the user has not cut before, unlike anything previously observed).

4 System

We instantiate our framework on three preference dimensions observable from passive video: spatial organization, discrete choice, and workflow style.

4.1 Perception Pipeline

We implement our observation module using Gemini 2.5 Flash, a multimodal LLM, to process video chunks and extract structured contextual and behavioral observations. To ground the LLM's outputs, we provide labeled photographs of items and zones in the scene, produced either by hand annotation or open-vocabulary segmentation models [11].

The LLM processes each chunk with a structured prompt defining what constitutes an evident preference observation for each dimension (e.g., actively selecting one tool when alternatives are available). It also proposes new observation variables that capture contextual factors not yet represented in the current factor graph when necessary (e.g., noticing that the user's adhesive choice varies with the area for adherence).

The LLM outputs a list of observations covering each chunk. To filter hallucinated observations, we apply self-consistency sampling [62]: we run N=3 independent samples per chunk and retain only observations where at least 2 samples agree, exploiting the finding that real events are stable across samples while hallucinations vary [38].

4.2 Modeling Preferences

These observations are the evidence that populates our factor graphs. Each preference dimension is a *plate*: a factor-graph template instantiated once per instance (e.g., one graph per item for spatial organization, one per tool use goal for discrete choice), organized around the question it answers and linking a latent preference to observed context (Figure 11). Beyond the initial contextual variables, the perception module proposes additional variables at runtime (Section 3.3.3) to adapt to new contexts dynamically.

For *spatial organization* (Figure 11 spatial), the question is "where does this item go?" The latent variable is the preferred location across workspace zones and receptacles. Observed variables include the item's use state, whether it was recently used or is idle, allowing the system to learn patterns such as which zones active tools stay in and which containers or zones idle ones are stored in.

For *discrete choice* (Figure 11 choice), the question is "which item does the user prefer among alternatives?" The latent variable is the preferred item (e.g., which cutting tool). Observed variables include the paired item being worked on. Separate plates keep goals (e.g., adhering, cutting) independent.

For *workflow style* (Figure 11 workflow), the question is "when does the user perform an action?" The latent variable is the preferred timing (e.g., handling something immediately vs. deferring it). Observed variables include the type and count of items involved, letting the system learn patterns such as deferring an action when items are few but performing it when they accumulate.

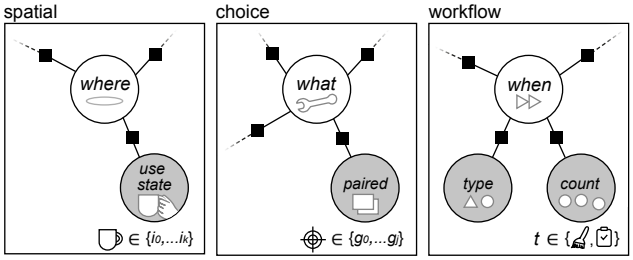

Figure 11: The three preference dimensions, each a plate replicating one factor graph per instance. Each asks a question (Where? What? When?) linking a latent preference to observed context, and updates from observations.

4.3 Continuous Update

As new observations arrive, each updates the relevant factor's weight as a co-occurrence count (Section 3.3.2). An observation may also expand a variable's domain when a new state appears (e.g., a new tool enters the workspace) or add a new context variable proposed by the perception module (Section 3.3.3). Factor graphs evolve as the system observes. For example, an adhesive factor graph starts with one contextual factor, "paired item" (Figure 11). When the perception module sees the user applying double-sided tape to small plastic color blocks, it proposes new factors such as "target surface area" and "material type," and stores this as evidence for a preference in that context. If the user switches to glue for paper, the system learns a separate "glue for paper" while keeping the original "tape for plastic", since each belief is conditioned on its own context.

This design provides two benefits: (1) preferences remain *context-specific* rather than overgeneralized (e.g., a blanket preference for tape), and (2) infrequently observed contexts accumulate little evidence, preventing interpreting rare contextual actions as stable preferences.

Beyond these updates, contrastive and error-driven updates let the model adapt quickly when preferences shift or a prediction is wrong (Section 3).

4.4 Querying the User Model

A query arrives either from a user instruction ("hand me a cutting tool") or from a proactive agent anticipating a next step. The system first routes the query to the relevant preference dimension using a local small language model (Qwen2.5-1.5B [47]) via

multiple-choice classification. It then selects the appropriate factor graph plate within that dimension using cosine similarity between sentence embeddings (all-mpnet-base-v2 [50]) of the query and plate names. Runtime observations become evidence: when an observation exactly matches a known variable state, it enters as hard evidence. When the match is approximate (e.g., "craft scissors" versus the known "scissors"), the system embeds the observation and computes cosine similarity against known domain values:

$$\text{sim}(a, b) = \frac{\mathbf{e}_a \cdot \mathbf{e}_b}{\|\mathbf{e}_a\| \, \|\mathbf{e}_b\|}$$

This becomes a soft evidence distribution through a temperature-scaled softmax (scores are divided by a temperature τ before softmax, concentrating probability on the closest match while keeping some weight on similar alternatives). This lets the model apply learned associations even to inputs it has not seen before.

The system runs belief propagation and returns a prediction and a confidence estimate. When the posterior entropy is too high (Section 3.3.6), the system signals uncertainty, enabling the downstream agent to ask the user rather than act on a weak prediction.

5 Evaluation

We evaluate our system's ability to learn user preferences from passive observation and adapt as behavior and contexts evolve. Our evaluation asks three questions. First, are our user models *accurate and well-calibrated?* Second, can the system *dynamically adapt to shifting preferences and new contexts?* Third, how does our approach *compare to simpler alternatives* a system builder might try, such as an LLM with textual memory of past observations?

5.1 Data Collection

We collect video of participants performing everyday desktop tasks across multiple domains, each designed to surface preference decision points across the targeted three preference dimensions that our system can observe and learn from. Our study obtained IRB approval from our institution (protocol STUDY2025_00000100).

5.1.1 Participants and Setup. We recruited 6 participants at our local university by word of mouth to perform desktop tasks at a standard workspace. We opted for a smaller number of participants since we don't look at trends across participants, but only model one individual's preference in isolation. An overhead camera captured each participant's actions throughout the session (Figure 12). Each session lasted approximately 1.5 hours, yielding 9 hours of video data. Participants were compensated with $18 for their time. We obtained informed consent prior to the study.

5.1.2 Tasks. Each participant completed tasks across three domains, i.e., assembly (Figure 12a), office work (Figure 12b), and crafting (Figure 12c), in randomized order. In the assembly session, participants built two figures from small color blocks. In the crafting session, participants made two paper crafts from pre-cut components. In the office work session, participants assembled two document packets. For each domain, participants were asked to organize the workspace, they received boxed packages, and used available tools to complete the task following instruction sheets while keeping the workspace as tidy as they preferred.

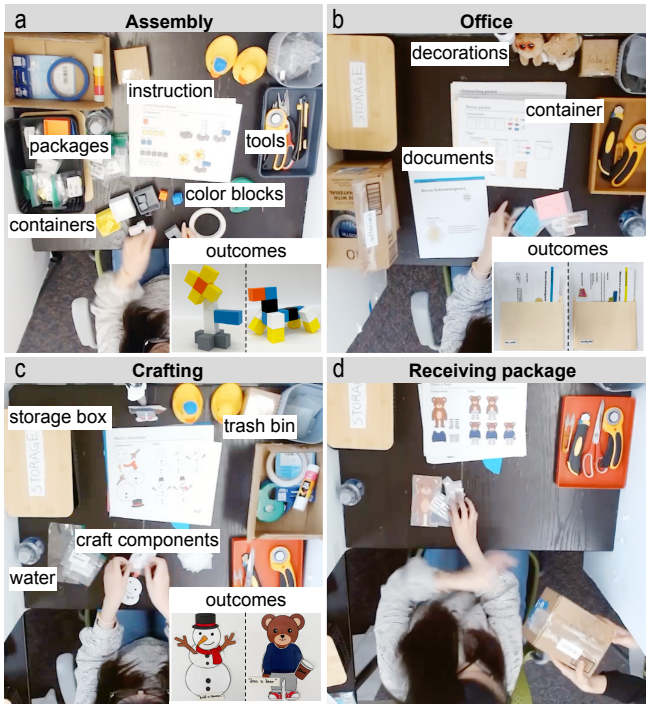

Figure 12: Participants performed (a) assembly, (b) office, and (c) crafting tasks at three workspaces in randomized order. (d) Participants received packages mid-session to simulate changing environments.

All three domains shared the same set of cutting tools and adhesives, as well as similar receptacles (a storage box, an open container, and four small containers) and comparable sets of objects (e.g., two decorative objects, two to five domain-relevant items such as pens in office), as depicted in Figure 12. In each session, we left some tools out in the initial setup and gave them to participants during the session (Figure 12d), to introduce variance that resembles real-world environments.

Each task surfaced decision points across three preference dimensions. Discrete choice preferences are revealed when participants select one tool over alternatives. Spatial preferences are revealed through workspace organization and subsequent tidy-ups during the task. Workflow preferences emerge through preparation steps (e.g., whether participants sort components before beginning assembly) and cleanup timing (e.g., discarding packaging immediately vs. saving it until the end). Participants were instructed to do everything however it felt natural and were not informed of what or how their preferences were being tracked until the end of the study.

5.2 Evaluation Procedure

5.2.1 Method. To simulate a continuously learning system in real-world deployment, we evaluate it using an incremental leave-one-out procedure. We segment each participant's video data across 3 sessions into 2-minute segments (yielding about 45 learning steps per participant). At each timestep t, the system has been updated with all segments from 0 to $t-1$, and we use it to predict what

happens at t: e.g., which item is chosen among alternatives, where objects are placed, or whether an action is done now or deferred.

5.2.2 Ground Truth Annotation. To obtain ground truth, we densely annotated all video data for each preference dimension using a custom annotation tool. We reviewed each VLM-extracted observation against the corresponding video frame, correcting errors where needed. For discrete choice, we labeled every tool selection event (which cutting tool, adhesive, or attachment was chosen at each decision point). For spatial preferences, we labeled object placement actions (which objects were moved where during different phases). For workflow preferences, we labeled preparation behaviors (e.g., whether participants sorted components before assembly) and cleanup timing (e.g., immediate vs. deferred disposal). These human-verified labels are *only used to score accuracy, not as model input*. Our evaluated system uses raw VLM observations, with self-consistency sampling [62] to mitigate errors.

5.2.3 Conditions. We compare our system against three design alternatives.

LLM + Memory receives dynamically updated textual descriptions from a video LLM across all three preference dimensions, leveraging in-context learning [6] to dynamically update over time. Observations are accumulated as text descriptions and updated in prompts incrementally. This condition receives the same text and visual grounding (item dictionaries, zone labels) as our system and follows the same incremental procedure. To obtain calibrated confidence estimates, we pose all predictions as multiple-choice questions and capture confidence as next-token log-probabilities over the answer options [30].

LLM Commonsense predicts from task context alone with no observation history, representing a generic prior without personalization. This condition also receives textual grounding of item, receptacle, and zone inventories. Confidence extraction is the same as *LLM+Memory*.

Frequency Count predicts the most frequently observed choice so far, testing whether structured probabilistic inference adds value beyond simply counting past behavior.

5.2.4 Metrics. We report prediction accuracy and Brier score [5]. Brier score is widely used to evaluate calibration in probabilistic systems [55]. It measures the mean squared error between predicted probabilities and actual binary outcomes, jointly capturing accuracy and calibration in a single metric (lower is better).

5.3 Results

Our user models are accurate, well-calibrated, and dynamically adaptive to changing behavior.

5.3.1 Accuracy and Calibration. As we show in Figure 13, our system achieves 92% prediction accuracy (Brier: 0.09) for discrete choice, 90% for spatial organization (Brier: 0.10), and 84% for workflow style (Brier: 0.09), reaching 89% overall accuracy (Brier: 0.09), averaged across all six participants and three domains. Accuracy variability is low, with per-participant overall accuracies of 86.9%, 88.6%, 86.4%, 90.5%, 88.1%, and 91.5%.

LLM Commonsense, which predicts from task context alone using commonsense reasoning, reaches 42% accuracy for choice, 10% for spatial, and 33% for workflow. Predicting without any personal history, it performs poorly across all three dimensions, showing how much personalization matters.

LLM + Memory, which receives the same observations as our system and accumulates them as textual descriptions via in-context learning, performs at 44% accuracy for choice, 18% for spatial, and 33% for workflow. Surprisingly, textual memory of personal history adds little over *LLM commonsense*. Its high Brier score shows that it produces confident yet incorrect predictions, since it accumulates observations but cannot reliably condition on context.

Frequency Count reaches 68% accuracy for choice, 61% for spatial, and 39% for workflow. This shows that frequency counting captures some signal, but by predicting the most common choice regardless of context, it misses the contextual variation our model captures.

Brier scores show a calibration difference. The LLM conditions produce confidently incorrect predictions, assigning high probabilities to wrong outcomes (overall Brier: 0.69 for commonsense, 0.65 for memory). Our system maintains well-calibrated uncertainty (overall Brier: 0.09), enabling a downstream agent to ask rather than act on a weak prediction.

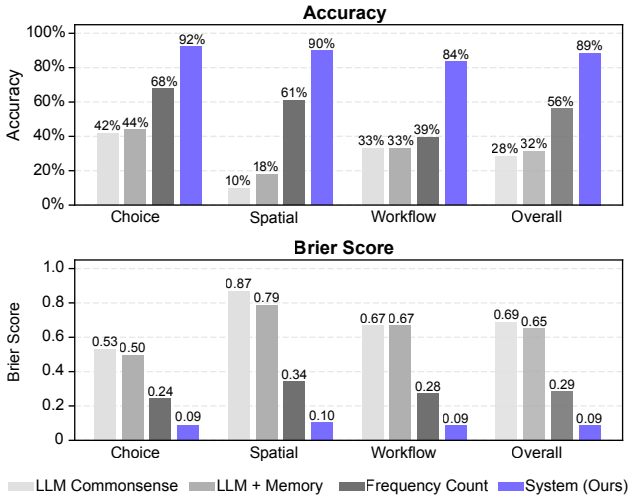

Figure 13: Our system (purple) achieves higher accuracy and lower Brier score than other design alternatives across dimensions.

5.3.2 Dynamic Adaptation over Behavior Adjustments. People dynamically adjust how they prefer to do things. For example, in our study, participants switched tools for different reasons, reorganized their workspaces between domains, and changed workflows as they discovered more efficient strategies. An effective system should adapt quickly when behavior shifts and maintain stable predictions when behavior is consistent.

In Figure 14, we show P1's selection preferences that *continuously update across the three domains*. Consider P1's adhesive preference during crafting (Figure 14b). While making the first paper craft, the system converges on P1's contextual adhesive preference (i.e., glue stick) given LLM proposed observation variables capturing relevant context: paired with craft component, permanence, small scale, and

paper material (Figure 14b, stable learning). The four contextual factors consistently favor glue stick.

While making an identical second craft, P1 switches from glue stick to double-sided tape. The system, still modeling glue stick as preferred, predicts wrong (Figure 14b, behavior change). This confident mismatch triggers our error-driven mechanism. The system penalizes the previously preferred glue stick and shifts its preference toward double-sided tape under the current context (Figure 14b, corrective update), correcting within 1–2 observations.

LLM + Memory baseline, on the other hand, does not recover within the crafting session, as its declining accuracy curve shows (Figure 14a). An embodied agent relying on it would keep applying glue for P1 even after P1 has shifted to double-sided tape.

Across participants, our system adapts to two kinds of change. Users change their behavior within a domain, as P1 did with adhesives, and contexts change between domains as new materials, objects, and task structures appear. Behavior changes and incorrect predictions trigger contrastive and error-driven updates, while new contexts trigger variable expansion and factor additions. These adaptations allow our system to maintain accuracy under changing behaviors, conditional preferences, and contexts.

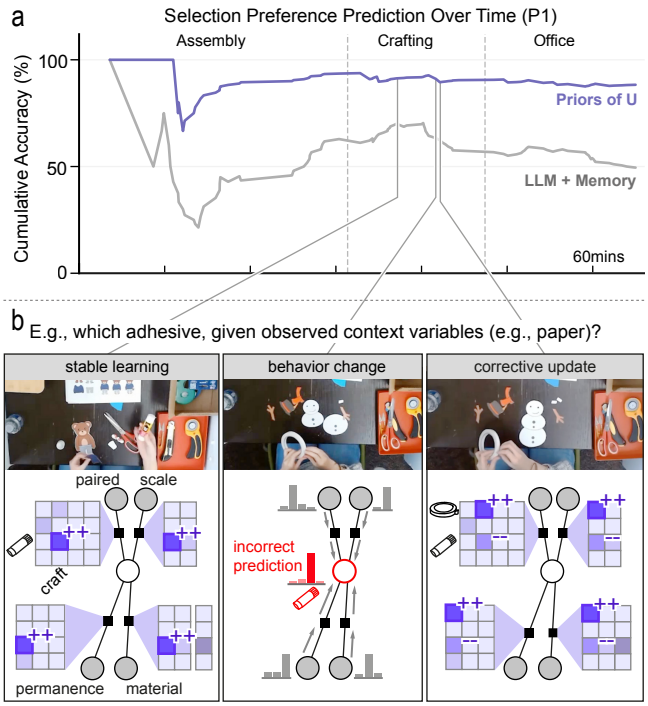

Figure 14: (a) Our system (purple) recovers quickly after P1's behavior change; LLM + Memory (gray) does not. (b) When behavior changes, a corrective update quickly revises the learned preference.

5.3.3 How Different Are Users? Our system discovers a rich set of conditional preferences for each participant. After three sessions, each participant's model contains around 150 learned conditional predictions (e.g., "prefers scissors when cutting paper and when cutting is precise" or "stores the utility knife in a receptacle when idle"). These conditionals emerge from the factor graph structure, where combinations of discovered observation variables (e.g., material, subtask) define distinct contexts for conditional preferences.

We show specific examples of this variation across the three preference dimensions in Figure 15. For spatial organization (Figure 15 Where?), when empty plastic packages are no longer in use, P1 stores them in a storage box, P3 places them in an open container, P4 leaves them on the table, and P2, P5, and P6 throw them away. For adhering paper craft components (Figure 15 What?), P1 prefers double-sided tape, shifted from the previously preferred glue stick (Figure 14); P2, P3, and P6 prefer a glue stick, and P4 reaches for the tape dispenser. For workflow of preparing and using adhesives (Figure 15 When?), P1, P4, P5, and P6 batch the action by preparing many tape strips before applying them, while P2 and P3 cut one strip and switch to applying it before cutting the next. Such variation is also seen in daily life and has no single correct answer, so embodied agents should learn and adapt to each user rather than assume one way.

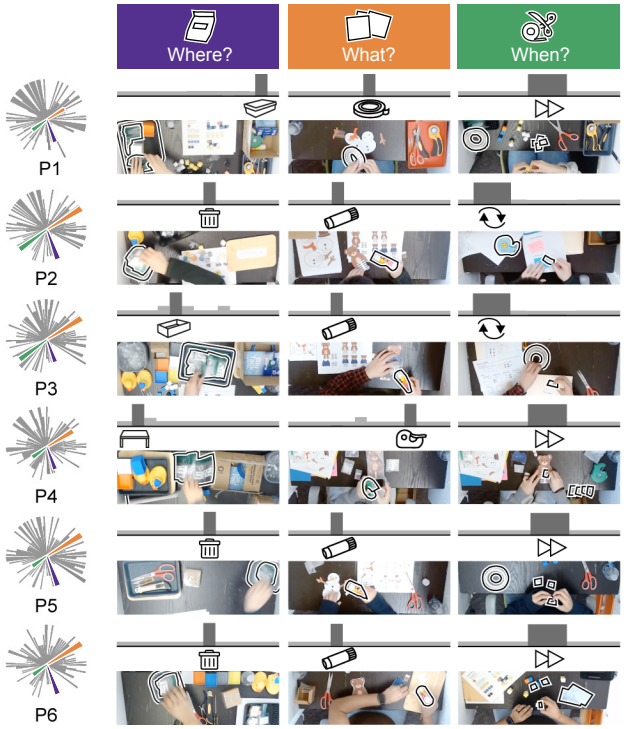

Figure 15: After three sessions, each participant's user model has discovered many conditional preferences, and these user models differ across participants. We show specific examples of how participants differ from each other.

6 Robot Demonstration

To illustrate how our user model enables personalized embodied assistance, we instantiate our system in a kitchen scenario. A team member performed cooking-related tasks in a lab kitchen setup

populated with their own objects and groceries, captured by an environmentally instrumented camera. Our full pipeline processes the video observations to dynamically build a user model. We then use a teleoperated robot to carry out the model's predictions at moments where the user would naturally benefit from assistance.

First, the system observes the user's natural kitchen routines: how they organize groceries (Figure 16a), arrange their workspace (Figure 16b), and clean up (Figure 16c), building up a user model.

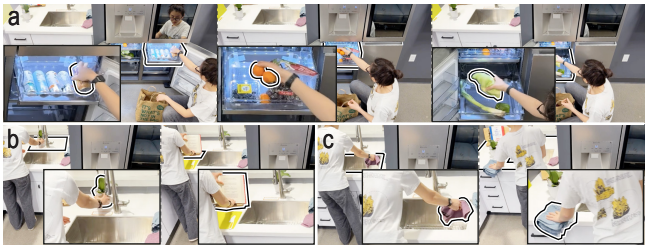

Figure 16: The system observes the user's natural activities in the kitchen through environment cameras, including (a) loading groceries, (b) organizing the countertop for cooking, and (c) cleaning up.

When the user returns with new groceries, the robot helps unload them according to learned spatial preferences: fruit goes to the top-left drawer in the fridge (Figure 17 red), vegetables to the middle-left (Figure 17 orange), and drinks to the top-right (Figure 17 green). Though the specific items differ from earlier trips, the learned preferences generalize across items in the same category.

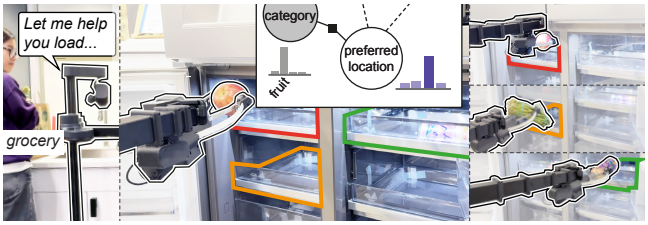

Figure 17: The robot unloads groceries to user-preferred locations.

When the user asks the robot to arrange the space for cooking, the user model knows to push the plant to the back, set up the bookstand in front of them for the cookbook, and prepare the workspace accordingly (Figure 18), saving the user the burden of specifying all these details in their instruction.

Finally, when the user asks the robot to clean up, it reaches for the blue rag for the countertop and the purple rag for the sink, following the user's cleaning habits rather than treating the two rags as interchangeable (Figure 19), which may otherwise result in undesirable help without personalization: e.g., using a user's *floor-cleaning* rag for their *countertop*.

7 Discussion

Personalizing embodied assistance can bring both interaction benefits and tension, which we discuss in turn.

Figure 18: When asked to arrange for cooking, the robot sets up the workspace to the user's preferences.

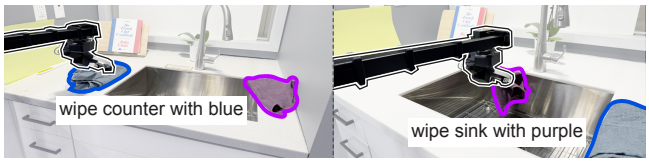

Figure 19: When asked to clean up, the robot uses the correct rag for each surface.

7.1 Interaction Benefits of Personalized Embodied Agents

Our physical living experiences are deeply personal. Our spaces are set up differently, we have different habits, and we prefer different things. As agents move out of screens and into our physical experiences, it is crucial for them to do things *the way we want*. An agent that organizes your fridge wrong or hands you the wrong tool is not just unhelpful, it gets in the way. We outline four interaction benefits that a personalized user model can provide (Figure 20), spanning a spectrum from user-initiated to agent-initiated interaction.

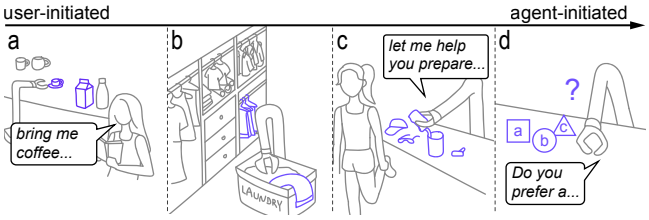

Figure 20: Personalization brings interaction benefits across the spectrum from user-initiated to agent-initiated assistance. (a) Users needn't specify everything when instructing; (b) agents personalize long-horizon task completion to the user; (c) agents proactively help based on known habits and routines; (d) agents know when to ask for more information.

At the user-initiated end, everyday instructions are often underspecified. *When a user asks* for coffee, they may not state their preferred creamer and mug every time. An agent that knows the user can fill in the gap from context and prior observations, for instance a small espresso with a splash of sweet cream when they are reading in the afternoon (Figure 20a).

Further along, *when an agent automates a long-horizon task* such as organizing a closet, personalized knowledge lets it finish in a state the user prefers (e.g., pants hung on the lower right) rather than a merely tidy state that is unfamiliar and may force the user to redo it (Figure 20b).

Moving toward the agent-initiated end, *an agent that knows a user's routines and habits* can *proactively offer help*. When a user begins their morning pre-run stretch in summer, the agent might prepare their electrolyte drink, hat, and running glasses, anticipating a familiar routine (Figure 20c).

Finally, knowing the user also *lets an agent recognize when it lacks the information it needs*, and ask targeted questions rather than act on a weak guess (Figure 20d).

7.2 How Much Should Agents Know About Us?

A system that learns preferences from passive observation raises a natural tension. The same capability that enables a robot to silently set up your workspace the way you like it also means an agent is continuously watching and building a model of your habits. Some of these inferences may surprise users: the system might learn that you always reach for a specific mug on stressful mornings, or that your cleanup habits change when you are tired. These patterns are useful for personalization, but they are also intimate.

We do not think there is a clean technical fix for this. Privacy-preserving measures like on-device processing and local model storage (Section 8) address data security, but they do not resolve the deeper question: *what kind of relationship do people want with the agents that share their spaces?* Some users may welcome an agent that anticipates their needs without being asked. Others may find the same behavior unsettling, preferring to stay in control even at the cost of convenience.

This is a design question, instead of merely a technical one. We believe that the right approach involves giving users visibility into what the system has learned (our factor graphs are inherently interpretable), control over what gets retained or forgotten, and the ability to set boundaries on what the agent should act on. How to design these controls in a way that is lightweight enough for everyday use is an interesting next step.

8 Limitations and Future Work

We discuss the limitations of our current system implementation and evaluation, and opportunities for future work.

Privacy. **Observation data is deeply personal, and it should stay in users' control.** Our proof-of-concept system's observation step currently uses a proprietary multimodal LLM (Gemini 2.5 Flash). For a system that passively observes daily life, sending video to a cloud API is untenable for real-world deployment. However, open-source video-capable LLMs are rapidly closing the gap with proprietary models [36, 61], and local inference on consumer hardware is increasingly feasible. We plan to test our full pipeline locally using an open-source video LLM.

Observation Window and Factor Discovery. Our current system processes video in fixed 2-minute chunks, which determines the granularity at which the LLM can observe and propose new context variables. Some preference patterns may only become apparent over longer windows (e.g., workflow preferences that span an entire task phase), while others may require finer resolution (e.g., momentary hesitations between tool choices). We saw initial benefits of varying window sizes during the implementation of our system. Adaptive observation windows that expand or contract based on activity density could improve both the efficiency and completeness of factor discovery.

Study Scope. Our evaluation involves six participants performing structured tasks in a controlled desktop setting. This validates our approach as a valuable direction to pursue further. Real-world evaluation is an important next step.

Moving into the Messy Real World. Real homes are messy: occlusion, poor lighting, cluttered backgrounds, unfamiliar objects, and limited camera angles would all degrade perception quality. Our system partially addresses this through self-consistency filtering and soft evidence propagation, which prevent any single uncertain observation from dominating the model, and as foundation models improve, our perception pipeline improves with them. Still, how robust preference learning can be under sustained partial observability remains an open question. An exciting next step is evaluating our system on egocentric and multi-view datasets that capture real daily activities at scale [45], which would test our pipeline under naturalistic conditions with diverse viewpoints, cluttered environments, and longer time horizons. Our contribution is the groundwork for that step, a dynamic, interpretable, and uncertainty-aware user model that can extend to real-world environments.

Richer and Longer-Term Learning. Our system learns entirely from passive observation, currently over a 1.5-hour session per participant. Both the observation modality and the time horizon can expand. In practice, users will likely give explicit feedback as well, whether volunteered ("not that mug," "I prefer the other drill") or prompted by the system. These directed signals could accelerate learning. Over longer time horizons, preferences drift, seasons change, and life circumstances shift. Over months rather than hours, the system could surface patterns a short evaluation misses, such as gradual drift or routines tied to time of day or week, and could track how habits evolve.

Extending to Other Preference Dimensions. Our framework identifies seven preference dimensions for physical assistance. We currently implement only three that are observable from passive video. Dimensions like initiative preferences, motion parameters, and task allocation would require observing human-robot interaction rather than human behavior alone. Our factor graph architecture may be extended to these dimensions in future work.

9 Conclusion

We presented a framework for learning user preferences from behavioral observation, enabling embodied agents to understand how people like things to be done without requiring explicit instruction. Our system combines vision foundation models for open-vocabulary observation with factor graphs for structured, interpretable, and efficient probabilistic inference. It learns across spatial arrangement, tool selection, and workflow preferences, and continuously adapts as behaviors and contexts change. Our evaluation showed accurate, well-calibrated predictions that outperform LLM baselines, and robust adaptation to behavioral shifts, new conditions, and new domains. Through robot demonstrations in kitchen scenarios, we showed how a personalized user model enables qualitatively

different interaction benefits for embodied agents. We believe that as embodied agents enter our personal spaces, the ability to know and adapt to each user will be critical to a good user experience.

Acknowledgments

We thank Prof. Andrea Bajcsy and Yiming Zhang for insightful discussions, Ziru Wei and Yitao Fan for pilot testing our study, Prof. Zackory Erickson for lending his lab's robot, and Daehwa Kim for help with video shooting. This work was partially supported by the National Science Foundation under Award No. IIS-2443190.

References

[1] Nichola Abdo, Cyrill Stachniss, Luciano Spinello, and Wolfram Burgard. 2015. Robot, organize my shelves! Tidying up objects by predicting user preferences. In *2015 IEEE International Conference on Robotics and Automation (ICRA)*. 1557–1564. https://doi.org/10.1109/ICRA.2015.7139396

[2] Andrea Bajcsy, Dylan P. Losey, Marcia K. O'Malley, and Anca D. Dragan. 2017. Learning Robot Objectives from Physical Human Interaction. In *Proceedings of the 1st Annual Conference on Robot Learning (Proceedings of Machine Learning Research, Vol. 78)*, Sergey Levine, Vincent Vanhoucke, and Ken Goldberg (Eds.). PMLR, 217–226. https://proceedings.mlr.press/v78/bajcsy17a.html

[3] Jimmy Baraglia, Maya Cakmak, Yukie Nagai, Rajesh Rao, and Minoru Asada. 2016. Initiative in robot assistance during collaborative task execution. In *2016 11th ACM/IEEE International Conference on Human-Robot Interaction (HRI)*. 67–74. https://doi.org/10.1109/HRI.2016.7451735

[4] Sandra Bedaf, Patrizia Marti, and Luc De Witte. 2019. What are the preferred characteristics of a service robot for the elderly? A multi-country focus group study with older adults and caregivers. *Assistive Technology* 31, 3 (2019), 147–157. https://doi.org/10.1080/10400435.2017.1402390 PMID: 29125807.

[5] Glenn W Brier. 1950. Verification of Forecasts Expressed in Terms of Probability. *Monthly Weather Review* 78, 1 (1950), 1–3.

[6] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel Ziegler, Jeffrey Wu, Clemens Winter, Chris Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. Language Models are Few-Shot Learners. In *Advances in Neural Information Processing Systems*, H. Larochelle, M. Ranzato, R. Hadsell, M.F. Balcan, and H. Lin (Eds.), Vol. 33. Curran Associates, Inc., 1877–1901. https://proceedings.neurips.cc/paper_files/paper/2020/file/1457c0d6bfcb4967418bfb8ac142f64a-Paper.pdf

[7] Erdem Bıyık, Dylan P. Losey, Malayandi Palan, Nicholas C. Landolfi, Gleb Shevchuk, and Dorsa Sadigh. 2022. Learning reward functions from diverse sources of human feedback: Optimally integrating demonstrations and preferences. *The International Journal of Robotics Research* 41, 1 (2022), 45–67. https://doi.org/10.1177/02783649211041652 arXiv:https://doi.org/10.1177/02783649211041652

[8] Arthur Caetano, Yunhao Luo, Adwait Sharma, and Misha Sra. 2025. GraspR: A Computational Model of Spatial User Preferences for Adaptive Grasp UI Design. In *Proceedings of the 38th Annual ACM Symposium on User Interface Software and Technology (UIST '25)*. Association for Computing Machinery, New York, NY, USA, Article 96, 16 pages. https://doi.org/10.1145/3746059.3747744

[9] Maya Cakmak, Siddhartha S. Srinivasa, Min Kyung Lee, Jodi Forlizzi, and Sara Kiesler. 2011. Human preferences for robot-human hand-over configurations. In *2011 IEEE/RSJ International Conference on Intelligent Robots and Systems*. 1986–1993. https://doi.org/10.1109/IROS.2011.6094735

[10] Gerard Canal, Guillem Alenyà, and Carme Torras. 2017. A taxonomy of preferences for physically assistive robots. In *2017 26th IEEE International Symposium on Robot and Human Interactive Communication (RO-MAN)*. 292–297. https://doi.org/10.1109/ROMAN.2017.8172316

[11] Nicolas Carion, Laura Gustafson, Yuan-Ting Hu, Shoubhik Debnath, Ronghang Hu, Didac Suris, Chaitanya Ryali, Kalyan Vasudev Alwala, Haitham Khedr, Andrew Huang, Jie Lei, Tengyu Ma, Baishan Guo, Arpit Kalla, Markus Marks, Joseph Greer, Meng Wang, Peize Sun, Roman Rädle, Triantafyllos Afouras, Effrosyni Mavroudi, Katherine Xu, Tsung-Han Wu, Yu Zhou, Liliane Momeni, Rishi Hazra, Shuangrui Ding, Sagar Vaze, Francois Porcher, Feng Li, Siyuan Li, Aishwarya Kamath, Ho Kei Cheng, Piotr Dollár, Nikhila Ravi, Kate Saenko, Pengchuan Zhang, and Christoph Feichtenhofer. 2025. SAM 3: Segment Anything with Concepts. *CoRR* abs/2511.16719 (2025). https://doi.org/10.48550/ARXIV.2511.16719 arXiv:2511.16719

[12] Frank Dellaert. 2012. Factor Graphs and GTSAM: A Hands-On Introduction. *Georgia Institute of Technology, Tech. Rep* 2, 4 (2012).

[13] Sean Follmer, Daniel Leithinger, Alex Olwal, Akimitsu Hogge, and Hiroshi Ishii. 2013. inFORM: dynamic physical affordances and constraints through shape and object actuation. In *Proceedings of the 26th Annual ACM Symposium on User Interface Software and Technology* (St. Andrews, Scotland, United Kingdom) *(UIST '13)*. Association for Computing Machinery, New York, NY, USA, 417–426. https://doi.org/10.1145/2501988.2502032

[14] Krzysztof Gajos and Daniel S. Weld. 2004. SUPPLE: automatically generating user interfaces. In *Proceedings of the 9th International Conference on Intelligent User Interfaces* (Funchal, Madeira, Portugal) *(IUI '04)*. Association for Computing Machinery, New York, NY, USA, 93–100. https://doi.org/10.1145/964442.964461

[15] Krzysztof Z. Gajos, Jacob O. Wobbrock, and Daniel S. Weld. 2007. Automatically generating user interfaces adapted to users' motor and vision capabilities. In *Proceedings of the 20th Annual ACM Symposium on User Interface Software and Technology* (Newport, Rhode Island, USA) *(UIST '07)*. Association for Computing Machinery, New York, NY, USA, 231–240. https://doi.org/10.1145/1294211.1294253

[16] Alix Goguey, Carl Gutwin, Zhe Chen, Pang Suwanaposee, and Andy Cockburn. 2021. Interaction Pace and User Preferences. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems* (Yokohama, Japan) *(CHI '21)*. Association for Computing Machinery, New York, NY, USA, Article 195, 14 pages. https://doi.org/10.1145/3411764.3445772

[17] Matthew Gombolay, Anna Bair, Cindy Huang, and Julie Shah. 2017. Computational Design of Mixed-Initiative Human–Robot Teaming That Considers Human Factors: Situational Awareness, Workload, and Workflow Preferences. *The International Journal of Robotics Research* 36, 5-7 (2017), 597–617.

[18] 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* (San Francisco, CA, USA) *(UIST '23)*. Association for Computing Machinery, New York, NY, USA, Article 53, 12 pages. https://doi.org/10.1145/3586183.3606740

[19] Jennifer Grannen, Siddhart Kamcheti, Blake Wulfe, and Dorsa Sadigh. 2025. ProVox: Personalization and Proactive Planning for Situated Human-Robot Collaboration. *IEEE Robotics and Automation Letters (RA-L)* (2025).

[20] Violet Yinuo Han, Amber Yinglei Chen, Mason Zadan, Jesse T Gonzalez, Dinesh K Patel, Wendy Fangwen Yu, 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 (UIST '25)*. Association for Computing Machinery, New York, NY, USA, Article 98, 15 pages. https://doi.org/10.1145/3746059.3747720

[21] 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 *Proceedings of the 38th Annual ACM Symposium on User Interface Software and Technology (UIST '25)*. Association for Computing Machinery, New York, NY, USA, Article 173, 16 pages. https://doi.org/10.1145/3746059.3747726

[22] 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 (IUI '25)*. Association for Computing Machinery, New York, NY, USA, 54–69. https://doi.org/10.1145/3708359.3712070

[23] Violet Yinuo Han, Ziru Wei, Aiden Yiliu Li, Chris Wu, and Alexandra Ion. 2026. Objects with Arms: How Robot Embodiment Shapes Perception in Physical Collaboration. In *2026 35th IEEE International Conference on Robot and Human Interactive Communication (RO-MAN)*.

[24] Keita Higuchi, Hiroki Tsuchida, Eshed Ohn-Bar, Yoichi Sato, and Kris Kitani. 2020. Learning Context-dependent Personal Preferences for Adaptive Recommendation. *ACM Trans. Interact. Intell. Syst.* 10, 3, Article 23 (Nov. 2020), 26 pages. https://doi.org/10.1145/3359755

[25] Eric Horvitz, Jack Breese, David Heckerman, David Hovel, and Koos Rommelse. 1998. The Lumiere Project: Bayesian User Modeling for Inferring the Goals and Needs of Software Users. In *Proceedings of the Fourteenth Conference on Uncertainty in Artificial Intelligence, Madison, WI, July 1998, Morgan Kaufmann: San Francisco*. 256–265.

[26] 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* (Atlanta, Georgia, USA) *(CHI '97)*. Association for Computing Machinery, New York, NY, USA, 234–241. https://doi.org/10.1145/258549.258715

[27] Alan Jern, Christopher G Lucas, and Charles Kemp. 2017. People Learn Other People's Preferences through Inverse Decision-Making. *Cognition* 168 (2017), 46–64.

[28] Matthew Jeung, Anup Sathya, Wanli Qian, Steven Arellano, Luke Jimenez, and Ken Nakagaki. 2025. Shape n' Swarm: Hands-on, Shape-aware Generative Authoring with Swarm UI and LLMs. In *Proceedings of the 38th Annual ACM Symposium on User Interface Software and Technology (UIST '25)*. Association for Computing Machinery, New York, NY, USA, Article 175, 16 pages. https://doi.org/10.1145/3746059.3747781

[29] Michael Jugovac and Dietmar Jannach. 2017. Interacting with Recommenders—Overview and Research Directions. 7, 3, Article 10 (Sept. 2017), 46 pages. https://doi.org/10.1145/3001837

[30] Saurav Kadavath, Tom Conerly, Amanda Askell, Tom Henighan, Dawn Drain, Ethan Perez, Nicholas Schiefer, Zac Hatfield-Dodds, Nova DasSarma, Eli Tran-Johnson, et al. 2022. Language Models (Mostly) Know What They Know. *arXiv preprint arXiv:2207.05221* (2022).

[31] Ivan Kapelyukh and Edward Johns. 2021. My House, My Rules: Learning Tidying Preferences with Graph Neural Networks. In *Conference on Robot Learning, 8-11 November 2021, London, UK (Proceedings of Machine Learning Research, Vol. 164)*, Aleksandra Faust, David Hsu, and Gerhard Neumann (Eds.). PMLR, 740–749. https://proceedings.mlr.press/v164/kapelyukh22a.html

[32] Mohamed Kari and Parastoo Abtahi. 2025. Reality Promises: Virtual-Physical Decoupling Illusions in Mixed Reality via Invisible Mobile Robots *(UIST '25)*. Association for Computing Machinery, New York, NY, USA, Article 172, 17 pages. https://doi.org/10.1145/3746059.3747660

[33] Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C. Berg, Wan-Yen Lo, Piotr Dollár, and Ross Girshick. 2023. Segment Anything. In *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*. 3992–4003. https://doi.org/10.1109/ICCV51070.2023.00371

[34] F.R. Kschischang, B.J. Frey, and H.-A. Loeliger. 2001. Factor graphs and the sum-product algorithm. *IEEE Transactions on Information Theory* 47, 2 (2001), 498–519. https://doi.org/10.1109/18.910572

[35] Taeyoon Kwon, Dongwook Choi, Hyojun Kim, Sunghwan Kim, Seungjun Moon, Beong woo Kwak, Kuan-Hao Huang, and Jinyoung Yeo. 2026. Embodied Agents Meet Personalization: Investigating Challenges and Solutions Through the Lens of Memory Utilization. In *The Fourteenth International Conference on Learning Representations*. https://openreview.net/forum?id=E5L43l5EIu

[36] Bin Lin, Yang Ye, Bin Zhu, Jiaxi Cui, Munan Ning, Peng Jin, and Li Yuan. 2024. Video-LLaVA: Learning United Visual Representation by Alignment Before Projection. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, Yaser Al-Onaizan, Mohit Bansal, and Yun-Nung Chen (Eds.). Association for Computational Linguistics, Miami, Florida, USA, 5971–5984. https://doi.org/10.18653/v1/2024.emnlp-main.342

[37] Karthik Mahadevan, Mauricio Sousa, Anthony Tang, and Tovi Grossman. 2021. "Grip-that-there": An Investigation of Explicit and Implicit Task Allocation Techniques for Human-Robot Collaboration. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems* (Yokohama, Japan) *(CHI '21)*. Association for Computing Machinery, New York, NY, USA, Article 215, 14 pages. https://doi.org/10.1145/3411764.3445355

[38] Potsawee Manakul, Adian Liusie, and Mark Gales. 2023. SelfCheckGPT: Zero-Resource Black-Box Hallucination Detection for Generative 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, 9004–9017. https://doi.org/10.18653/v1/2023.emnlp-main.557

[39] Abhinav Mehrotra, Robert Hendley, and Mirco Musolesi. 2016. PrefMiner: mining user's preferences for intelligent mobile notification management. In *Proceedings of the 2016 ACM International Joint Conference on Pervasive and Ubiquitous Computing* (Heidelberg, Germany) *(UbiComp '16)*. Association for Computing Machinery, New York, NY, USA, 1223–1234. https://doi.org/10.1145/2971648.2971747

[40] Heramb Nemlekar, Neel Dhanaraj, Angelos Guan, Satyandra K. Gupta, and Stefanos Nikolaidis. 2023. Transfer Learning of Human Preferences for Proactive Robot Assistance in Assembly Tasks. In *Proceedings of the 2023 ACM/IEEE International Conference on Human-Robot Interaction* (Stockholm, Sweden) *(HRI '23)*. Association for Computing Machinery, New York, NY, USA, 575–583. https://doi.org/10.1145/3568162.3576965

[41] Ali Noormohammadi-Asl, Stephen L. Smith, and Kerstin Dautenhahn. 2025. To Lead or to Follow? Adaptive Robot Task Planning in Human–Robot Collaboration. *IEEE Transactions on Robotics* 41 (2025), 4215–4235. https://doi.org/10.1109/TRO.2025.3582816

[42] Malayandi Palan, Nicholas C Landolfi, Gleb Shevchuk, and Dorsa Sadigh. 2019. Learning Reward Functions by Integrating Human Demonstrations and Preferences. *Robotics: Science and Systems (RSS)* (2019).

[43] Yi-Hao Peng, Muh-Tarng Lin, Yi Chen, TzuChuan Chen, Pin Sung Ku, Paul Taele, Chin Guan Lim, and Mike Y. Chen. 2019. PersonalTouch: Improving Touchscreen Usability by Personalizing Accessibility Settings based on Individual User's Touchscreen Interaction. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems* (Glasgow, Scotland Uk) *(CHI '19)*. Association for Computing Machinery, New York, NY, USA, 1–11. https://doi.org/10.1145/3290605.3300913

[44] Gojko Perovic, Francesco Iori, Angela Mazzeo, Marco Controzzi, and Egidio Falotico. 2023. Adaptive Robot-Human Handovers With Preference Learning. *IEEE Robotics and Automation Letters* 8, 10 (2023), 6331–6338. https://doi.org/10.1109/LRA.2023.3306280

[45] Toby Perrett, Ahmad Darkhalil, Saptarshi Sinha, Omar Emara, Sam Pollard, Kranti Kumar Parida, Kaiting Liu, Prajwal Gatti, Siddhant Bansal, Kevin Flanagan, Jacob Chalk, Zhifan Zhu, Rhodri Guerrier, Fahd Abdelazim, Bin Zhu, Davide Moltisanti, Michael Wray, Hazel Doughty, and Dima Damen. 2025. HD-EPIC: A Highly-Detailed Egocentric Video Dataset. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 23901–23913. https://doi.org/10.1109/CVPR52734.2025.02226

[46] Johannes Pöschmann, Tim Pfeifer, and Peter Protzel. 2020. Factor Graph based 3D Multi-Object Tracking in Point Clouds. In *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. 10343–10350. https://doi.org/10.1109/IROS45743.2020.9340932

[47] Qwen, :, An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, Huan Lin, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiaxi Yang, Jingren Zhou, Junyang Lin, Kai Dang, Keming Lu, Keqin Bao, Kexin Yang, Le Yu, Mei Li, Mingfeng Xue, Pei Zhang, Qin Zhu, Rui Men, Runji Lin, Tianhao Li, Tianyi Tang, Tingyu Xia, Xingzhang Ren, Xuancheng Ren, Yang Fan, Yang Su, Yichang Zhang, Yu Wan, Yuqiong Liu, Zeyu Cui, Zhenru Zhang, and Zihan Qiu. 2025. Qwen2.5 Technical Report. arXiv:2412.15115 [cs.CL] https://arxiv.org/abs/2412.15115

[48] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. 2021. Learning Transferable Visual Models From Natural Language Supervision. In *Proceedings of the 38th International Conference on Machine Learning, ICML 2021, 18-24 July 2021, Virtual Event (Proceedings of Machine Learning Research, Vol. 139)*, Marina Meila and Tong Zhang (Eds.). PMLR, 8748–8763. http://proceedings.mlr.press/v139/radford21a.html

[49] Kartik Ramachandruni and Sonia Chernova. 2025. Personalized Robotic Object Rearrangement from Scene Context. In *2025 34th IEEE International Conference on Robot and Human Interactive Communication (RO-MAN)*. 259–266. https://doi.org/10.1109/RO-MAN63969.2025.11217903

[50] Nils Reimers and Iryna Gurevych. 2019. Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, Kentaro Inui, Jing Jiang, Vincent Ng, and Xiaojun Wan (Eds.). Association for Computational Linguistics, Hong Kong, China, 3982–3992. https://doi.org/10.18653/v1/D19-1410

[51] Allen Z. Ren, Anushri Dixit, Alexandra Bodrova, Sumeet Singh, Stephen Tu, Noah Brown, Peng Xu, Leila Takayama, Fei Xia, Jake Varley, Zhenjia Xu, Dorsa Sadigh, Andy Zeng, and Anirudha Majumdar. 2023. Robots That Ask For Help: Uncertainty Alignment for Large Language Model Planners. In *Conference on Robot Learning, CoRL 2023, 6-9 November 2023, Atlanta, GA, USA (Proceedings of Machine Learning Research, Vol. 229)*, Jie Tan, Marc Toussaint, and Kourosh Darvish (Eds.). PMLR, 661–682. https://proceedings.mlr.press/v229/ren23a.html

[52] Elaine Rich. 1979. User Modeling via Stereotypes. *Cognitive Science* 3, 4 (1979), 329–354.

[53] Paul A. Samuelson. 1948. Consumption Theory in Terms of Revealed Preference. *Economica* 15, 60 (1948), 243–253. http://www.jstor.org/stable/2549561

[54] Rohin Shah, Dmitrii Krasheninnikov, Jordan Alexander, Pieter Abbeel, and Anca D. Dragan. 2019. Preferences Implicit in the State of the World. In *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*. OpenReview.net. https://openreview.net/forum?id=rkevMnRqYQ

[55] Omar Shaikh, Shardul Sapkota, Shan Rizvi, Eric Horvitz, Joon Sung Park, Diyi Yang, and Michael S. Bernstein. 2025. Creating General User Models from Computer Use. In *Proceedings of the 38th Annual ACM Symposium on User Interface Software and Technology (UIST '25)*. Association for Computing Machinery, New York, NY, USA, Article 35, 23 pages. https://doi.org/10.1145/3746059.3747722

[56] David Sirkin, Brian Mok, Stephen Yang, and Wendy Ju. 2015. Mechanical Ottoman: How Robotic Furniture Offers and Withdraws Support. In *Proceedings of the Tenth Annual ACM/IEEE International Conference on Human-Robot Interaction* (Portland, Oregon, USA) *(HRI '15)*. Association for Computing Machinery, New York, NY, USA, 11–18. https://doi.org/10.1145/2696454.2696461

[57] Paul Slovic. 1975. Choice between Equally Valued Alternatives. *Journal of Experimental Psychology: Human perception and performance* 1, 3 (1975), 280.

[58] Yao Song, Christoph Gebhardt, Yi-Chi Liao, and Christian Holz. 2025. Preference-Guided Multi-Objective UI Adaptation. In *Proceedings of the 38th Annual ACM Symposium on User Interface Software and Technology (UIST '25)*. Association for Computing Machinery, New York, NY, USA, Article 120, 13 pages. https://doi.org/10.1145/3746059.3747645

[59] Gemini Robotics Team, Saminda Abeyruwan, Joshua Ainslie, Jean-Baptiste Alayrac, Montserrat Gonzalez Arenas, Travis Armstrong, Ashwin Balakrishna, Robert Baruch, Maria Bauza, Michiel Blokzijl, et al. 2025. Gemini Robotics: Bringing AI into the Physical World. *arXiv preprint arXiv:2503.20020* (2025).

[60] Huaxiaoyue Wang, Nathaniel Chin, Gonzalo Gonzalez-Pumariega, Xiangwan Sun, Neha Sunkara, Maximus Adrian Pace, Jeannette Bohg, and Sanjiban Choudhury. 2025. APRICOT: Active Preference Learning and Constraint-Aware Task Planning with LLMs. In *Proceedings of The 8th Conference on Robot Learning (Proceedings of Machine Learning Research, Vol. 270)*, Pulkit Agrawal, Oliver Kroemer, and Wolfram Burgard (Eds.). PMLR, 1590–1642. https://proceedings.mlr.press/v270/wang25e.html

[61] Peng Wang, Shuai Bai, Sinan Tan, Shijie Wang, Zhihao Fan, Jinze Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, et al. 2024. Qwen2-VL: Enhancing

Vision-Language Model's Perception of the World at Any Resolution. *arXiv preprint arXiv:2409.12191* (2024).

[62] Xuezhi Wang, Jason Wei, Dale Schuurmans, Quoc V. Le, Ed H. Chi, Sharan Narang, Aakanksha Chowdhery, and Denny Zhou. 2023. Self-Consistency Improves Chain of Thought Reasoning in Language Models. In *The Eleventh International Conference on Learning Representations, ICLR 2023, Kigali, Rwanda, May 1-5, 2023*. OpenReview.net. https://openreview.net/forum?id=1PL1NIMMrw

[63] Jimmy Wu, Rika Antonova, Adam Kan, Marion Lepert, Andy Zeng, Shuran Song, Jeannette Bohg, Szymon Rusinkiewicz, and Thomas Funkhouser. 2023. TidyBot: Personalized Robot Assistance with Large Language Models. In *2023 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. 3546–3553. https://doi.org/10.1109/IROS55552.2023.10341577

[64] Antoine Yang, Arsha Nagrani, Paul Hongsuck Seo, Antoine Miech, Jordi Pont-Tuset, Ivan Laptev, Josef Sivic, and Cordelia Schmid. 2023. Vid2Seq: Large-Scale Pretraining of a Visual Language Model for Dense Video Captioning. In *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 10714–10726. https://doi.org/10.1109/CVPR52729.2023.01032

[65] Shuo Yang, Gerry Chen, Yetong Zhang, Howie Choset, and Frank Dellaert. 2021. Equality Constrained Linear Optimal Control With Factor Graphs. In *2021 IEEE International Conference on Robotics and Automation (ICRA)*. 9717–9723. https://doi.org/10.1109/ICRA48506.2021.9562000

[66] Michelle D Zhao, Reid Simmons, and Henny Admoni. 2023. Learning Human Contribution Preferences in Collaborative Human-Robot Tasks. In *Proceedings of The 7th Conference on Robot Learning (Proceedings of Machine Learning Research, Vol. 229)*, Jie Tan, Marc Toussaint, and Kourosh Darvish (Eds.). PMLR, 3597–3618. https://proceedings.mlr.press/v229/zhao23b.html

[67] Yue Zhao, Ishan Misra, Philipp Krähenbühl, and Rohit Girdhar. 2023. Learning Video Representations from Large Language Models. In *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 6586–6597. https://doi.org/10.1109/CVPR52729.2023.00637