MAKING ROBOT BEHAVIORS AUTOMATICALLY TRANSPARENT

Nick Walker

A dissertation
submitted in partial fulfillment of the
requirements for the degree of

Doctor of Philosophy

University of Washington

2025

Reading Committee:
Maya Cakmak, Chair
Siddhartha S. Srinivasa
Joshua R. Smith

Program Authorized to Offer Degree:
Computer Science & Engineering

University of Washington

Abstract

MAKING ROBOT BEHAVIORS
AUTOMATICALLY TRANSPARENT

Nick Walker

Chair of the Supervisory Committee:
Maya Cakmak
Paul G. Allen School of Computer Science & Engineering

Incorporating transparency into a robot behavior often demands substantial effort and expertise, to the detriment of anyone who must interact with them closely or for extended periods. We envision a future where this burden is automated away, enabling robot behaviors to be transparent by default. To this end, we propose a conceptual framework that formalizes the distinction between the design and implementation of a behavior and its transparent execution. This separation clarifies the scope of transparency interventions in human-robot interaction (HRI) and highlights key challenges in achieving generalizable transparency.

This dissertation addresses these challenges across diverse HRI contexts, demonstrating data- and model-driven augmentations that improve transparency. First, we investigate how users perceive learning robots, uncovering gaps between behavior and external attributions that inform the design of transparency mechanisms. Next, we introduce a data-driven method for controlling the expression of robot affect, enabling behavior creators to balance transparency with task performance. We then present a model-based approach to improving transparency in assistive teleoperation, allowing for explicit control over the trade-off between assistance and intelligibility. Finally, we tackle transparency for human supervisors reviewing robot failures, developing techniques to distill voluminous multi-modal recordings into interpretable summaries.

Together, these contributions demonstrate the feasibility of providing transparency into a robot behavior without requiring extensive modification of the original behavior specification. By framing transparency as an independent layer over existing behaviors, this work moves toward automation-friendly, scalable solutions that enhance human-robot interaction across a range of domains.

Contents

List of Figures

List of Tables

Acknowledgments

Thank you Maya Cakmak for years of advice and mentorship, and for your unflinching support of even the quaintest branches of my curiosity. All I've learned from you about science, people and life is what I carry forward. I am grateful for the advice and collaboration of my committee members, Professors Siddhartha Srinivasa and Joshua Smith, Julie Shah, Karen Leung, and Martin Nisser. Your collective work makes the lab and the robotics community a worthwhile place to be.

In that vein, thank you to many colleagues for your camaraderie and contributions to this work—Patrícia Alves-Oliveira, Maru Cabrera, Mike Jae-Yoon Chung, Varad Dhat, Markus Grotz, Brian Liang, Chris Mavrogiannis, Michael Murray, Amal Nanavati, Leah Perlmutter, Vinitha Ranganeni, Joe Sluis, Zihan Wang, and Brian Yao. Before UW, my path started as an undergraduate in the University of Texas's Freshman Research Initiative, and continued only with the help of Matteo Leonetti, Jivko Sinapov, Justin Hart and Peter Stone. During that period, it was an internship with Amanda Fernandez that helped me see that I should pursue research. All of these opportunities grew out of the support I received from many teachers at Communications Arts.

My research was supported in part by an Allen School Computer Science & Engineering Research Fellowship, a National Science Foundation Graduate Research Fellowship, the Honda Curious Minded Machines initiative, the Amazon-UW Science Hub's Manipulation in Densely Packed Containers project, and an internship at NVIDIA where I was hosted by Claudia Pérez-D'Arpino and Dieter Fox.

Outside of research, hundreds of runs with too many friends to name were all an essential part of my time at UW. Together, we saw the city a little differently. Thank you Zach Tatlock and Chandra Nandi for your leadership and for making Race Condition Running a community pillar.

I'm grateful for the love and understanding of my parents, my brother, and my extended family both near and far. And finally, thank you Yuqian Jiang, for being the breeze in my Austin nights.

They are going to be these new objects that are going to be in everyone's working environment, everyone's educational environment, and everyone's home environment. We have a shot [at] putting a great object there—and if we don't, we're going to put one more piece-of-junk object there.

Steve Jobs, June 15th, 1983

1 Introduction

A robot is a metal alien—an agent, but unfamiliar and opaque, animated by unclear intentions. Users confront them with little prior experience, save for conceptions picked up from popular media. The evolved psychological priors that support human-human interaction can be unreliable when applied to robots. The more closely a user interacts, the more they depend on a robot, the greater the barrier inscrutability becomes. Ultimately, it's the job of robot designers and developers to make the robot *transparent*, to make it easy to understand things like what a robot wants to do, what it is going to do, and for what reason [1]. In essence, users should be able to "see into" the robot's behavior. Many researchers have sought to make this task easier by characterizing individual interactions, some in great detail. Why then are the majority of robots today still opaque?

Part of the challenge is that transparency is highly contextual, shaped by the specific users it serves, the purpose it fulfills, the type of information being conveyed, and the means available to communicate that information. Insights from research rarely map exactly to novel situations.

Outside of a lab, implementing poor transparency measures can be worse than not trying. A robot that overwhelms human collaborators with status information or that distracts them with poorly-timed alerts is a hazard. It's a natural consequence that conservative approaches predominate. One well-understood method is simply enumerating a robot's behavior repertoire and training users, shifting the sense-making burden onto the experts who must make the educational materials. Training and familiarization aren't feasible for all robots, however, especially as behaviors become more complex. In other settings, like consumer robotics or commercial service robotics, high standards of user experience mean that transparency is both expected and difficult to achieve. Embedding transparency throughout a robot's behavior requires effort across a full team of artists, interaction designers, and engineers.

How can a robot behavior be made automatically transparent? The work presented in this thesis seeks to demonstrate that

> *unchanged robot behavior specifications can be made transparent automatically by manipulating their observable characteristics to align with learned or intuitive models of human inference.*

To make this case, we visit several interaction contexts and achieve task-relevant improvements in transparency through the implementation of data- or model-driven augmentations to otherwise unchanged behavior specifications. The primary contribution of this dissertation is to demonstrate the value of this perspective across human-robot interaction. Necessarily, we formalize and advance particular interactions along the way.

1.1 Overview

We begin in Chapter 2 by establishing transparency terminology that we will use throughout this dissertation.

Chapter 3 gives our framework and describes the challenges around which the remainder of the thesis is framed. We also contrast our framing with other perspectives in the literature.

In Chapter 3.3.4 we illustrate the disconnect between robot behaviors and their externally observable characteristics in the context of a learning robot. We contribute findings from user responses to a series of carefully constructed interaction scenarios and discuss their implications on the design of transparency for intrinsically motivated agents.

Chapter 4.7 contributes a data-driven approach to uncovering and modeling complex attributions to robot motion and demonstrates through empirical user evaluations that the resulting models afford behavior creators a simple, single-parameter interface for controlling the transparent expression of robot affect while balancing performant task execution.

Chapter 5.5 contributes a simple, model-based method for improving the transparency of an assistive teleoperation system operating in dense clutter. The resulting method similarly provides a simple means for the behavior creator to balance the performance of the assistance with its transparency. A user evaluation comparing the method to a standard approach without this capability supports its utility.

Chapter 6.5 contributes approaches to improving transparency for human supervisors reviewing robot failures. We examine two cases, a mobile manipulator and an industrial picking workcell with existing behaviors described as finite state machines, and contribute methods for post-processing voluminous, multi-modal recordings into summaries. User studies support the value of the resulting summaries for identifying and diagnosing problems.

We conclude in Chapter 8 with a discussion of unaddressed challenges and possible future paths toward transparency for all robots.

Background

2

Transparency is the property that a system provides sufficient information about its internal state for interactants to accomplish their objectives. Not all robot communication is in service of transparency[1], but as we will describe, a substantial amount of robot information and its externalizations can be understood as occurring in service of transparency. Because transparency is most often a means to a context-specific desired outcome, roboticists generally seek to provide transparency that is *appropriate*; information that is not merely presented, but well-encoded for consumption, parsimonious, and making efficient use of limited hardware resources.

In this chapter, we contribute an implementation-focused perspective, a taxonomy of the main concerns that a practitioner would consider when designing a transparency solution. There have been numerous loosely overlapping reviews of transparency [2]–[5], but none that address the problem with the appropriate breadth to make clear the shared challenges that emerge when trying to computationally manage a robot's communication. We defer further contrasts with previous framings to Section 3.3.

The factors that most influence implementations of robot transparency are:

- the user the transparency is for,
- the purpose of the transparency,
- the information being made transparent,

[1]Consider a social robot relaying a weather report. While this involves information transmission, it doesn't reveal anything about the robot's internal state, reasoning, or planned actions. The robot is merely acting as a conduit for external information rather than making its own operation transparent to users.

- and the channels the robot uses for communication.

We rely on this terminology when describing interactions throughout the remainder of the dissertation. In the following, we define and survey the tremendous technical variation and breadth within past work addressing each of these factors. Our aim is to communicate that transparency is an organizing principle across a wide variety of seemingly-different interaction contexts.

2.1 Users

The people who interact with the robot are the consumers of transparency. We conceptualize these users as playing various *roles* in an interaction based on their needs, and as having different levels of *expertise*.

Prior work suggests that users can be *supervisors* monitoring and intervening in a robot's execution, *operators* selecting actions manually, *bystanders* observing the robot's actions, *teammates* acting in concert, or *mechanics* who act to adjust the robot's hardware. To these models, we add *adapters*, who enact persistent changes to the robot's behavior. Roles can be static but are more likely to vary for a flexible robot [6].

Expertise is a spectrum. A *novice* enters their role in an interaction with minimal prior experience. *Expert* users have preparation and knowledge that support their interactions. A robot's *creator* benefits from direct familiarity with the implementation of a system. For at least a short period, their mental conception of the robot's behavior corresponds precisely with reality. Expertise evolves slowly over time as users develop new schemata, mental models which simplify the processing and understanding of information [7].

2.2 Aims

Transparency is a means of supporting users' needs. That is, transparency is implemented with an aim in mind. Understanding the aim(s) of transparency helps define how it is evaluated.

Trust Transparency helps ensure human users have an appropriately calibrated level of trust in a robot system [8]. Trust is a belief in or willingness to rely upon a technology in an uncertain situation. Trust impacts how we use or do not use technologies; we exercise trust when we confide in a social robot, or when we allow a robotic cart to carry our belongings. The appropriateness of that trust depends on unobservable factors like the limitations of its autonomy. Making a robot more transparent is not the same thing as making it more trustworthy, but the two are related; a new user might approach a robot skeptically but gain trust quickly with the correct transparency affordances. Trust can be measured by subjective assessment either through post-interaction surveys or via momentary ratings [9].

Safety Ensuring that robots operate without damaging themselves or their environments is difficult. Ensuring that they do not harm users and bystanders they interact with is doubly so. The difficulty of guaranteeing safety is an important barrier to the adoption of robots. In settings like industrial work floors or warehouses, operating in closer proximity to humans can improve productivity but carries the risk of serious injury. In practice, safety is achieved with layers of guarantees, the lowest of which might be fundamentally indifferent to the nature of humans. An acceleration limit, for instance, can be formally verified in a robot's control software with basic assumptions on the kinematic properties of the robot and may remove the risk of grave injury on contact with humans. At higher levels, probabilistic bounds on the likelihood of a negative outcome like contact can be made under assumptions about the behavior of a bystander. For these models of humans to bear out in reality, the robot must be transparent so that humans can make decisions with a correct understanding of the robot's intended actions.

Acceptance The European Union's General Data Protection Regulation (GDPR) introduced a requirement that intelligent systems be transparent, presenting the strongest expression yet of a societal-level impetus for transparency [10]. While the consequences of legislation like GDPR for robotics remain to be seen, that regulators have interceded to protect users belies the harms that an opaque system can obscure [11]. On an individual scale, transparency, via trust, informs our decisions about when and when not to use automation technologies [12]. Making a robot transparent can serve to increase the likelihood that it is accepted. Acceptance is both a subjective phenomenon that can be assessed via questionnaires and an objective outcome that can be observed in (dis)usage patterns [13].

Monitoring Understanding current and historical performance is a concern throughout a robot's lifecycle. During development, robot creators make engineering decisions based on observations of a robot's performance. When deployed, supervisors may need to assess performance to make business decisions. In either case, the defining characteristic is that the interactants do not act on the robot or its environment. Rather, they use their understanding of the robot in combination with information about its execution to make evaluations.

Teamwork Users working with one or more robots to accomplish a task rely on transparency to make decisions, often at great frequency. Breakdowns in a human's understanding of a robot result in actions that conflict with the robot's or that fail to optimally leverage the robot's capacity [9]. The performance of a team is easy to measure objectively in context, and subjective measures about perceptions of robot teammates are available as well.

Recovery Robot limitations often mean that a human teammate will be called on to resolve an otherwise unrecoverable failure for the robot. Recovery interactions are of special interest because they are a critical backstop and because they definitionally involve exceptional circumstances. Some failure scenarios are predictable, in which case interactions can benefit from the robot being able to automatically detect the situation. Unmodeled factors, which are common in uncontrolled environments like homes, require the human to make an assessment of causes and select among possible resolutions. The quality of a recovery is objectively measurable.

Modification, Adaptation, and Repair Whether in response to particular failures or simply as a matter of preference, users benefit from transparency as they modify a robot's behavior. Debugging, customization, and end-user programming all involve a user making persistent changes to the robot, usually with the benefit of some kind of feedback. Making changes requires that a user be able to understand how the resulting behavior will perform in other scenarios. Supporting repair—adaptation that addresses specific shortcomings—is often the motivation for explainability and interpretability. Depending on the context, key measures include subjective assessments of the adaptation interaction and objective quality of the end-product behavior.

2.3 Information

Transparency requires communication of a robot's internal information to the user based on their expertise and role.

State Depending on the format of their specification, robot behaviors contain representations of their current phase as well as various representations of sensory information that drive the behavior. Exposing even basic information about the current phase of the behavior has significant value. One common transparency shortfall is failing to indicate whether a robot is stopped because its activity is complete or because it is momentarily waiting, states which look the same to observers. This situation can be resolved by providing a liveness status indicator (e.g., a tally light that is active whenever the robot behavior is running), or more elaborately by creating appropriate idle behaviors. The general problem of faithfully communicating the robot's current state is complex when the robot can take many actions that have similar external appearances.

Other salient elements of the robot's current state, like what it can perceive, may also need to be externalized.

Intent Users whose actions are contingent on the robot's, like observers walking by or teammates working alongside the robot, benefit from an indication of what the robot intends to do next. The availability of intent information depends on how a robot's behavior is represented. Task planning-based behaviors are more likely to be able to provide complete plans [14], whereas reactive behaviors may only be able to provide a local description of what will occur. Other robots might operate under a plan that changes too frequently to present in isolation. In practice, robot behaviors are often represented in loose formalisms, like finite-state machines, which have limited prospection capabilities and may not be written at a level that makes them easy to directly communicate to users [15].

Reasons Transparency in service of recovery, monitoring, or repair requires that the robot be able to explain why it took or will take an action. Here, the representation of the robot's behavior specification is key. A decision made by a learned policy, for instance, may only be directly attributable to differences in the values of feature vectors that have no meaning to a user. Some work seeks to address this problem by better aligning these representations with those of humans [16]. Otherwise, in the absence of interpretability, an explanation must be generated. Explainability remains a topic of much research in robotics [17]–[19]. In contrast, a decision made by a scripted behavior has causes traceable to program logic, which may be interpretable as is. Robots with symbolic task models benefit from the ability to generate counterfactuals [20], a particularly powerful way of communicating reasons via contrast [21], and can also reason about potential alternative models under which their explanations might be interpreted [22].

Objectives Task planning and other kinds of optimization-based behaviors are specified in terms of an objective. In contrast to communicating plans or actions, communicating an objective is parsimonious and empowers users to predict future outcomes [23]. However, this communication places more assumptions on interactants and requires more sophistication on the robot's part. Even if there is mutual understanding of the objective, the model the robot optimizes against may not match the human's model of a domain.

Needs While some needs can be routine, like the need for a mobile robot to be placed on a charger when its battery gets low, or the need for a human teammate to change the robot's end-effector, they often involve exceptional circumstances where the robot requires assistance. Some research has investigated ways of using domain models to generate detailed language requests for help [24], [25].

Character and Affect Maintaining a consistent expression of affect and managing its evolution in response to interaction are core to social robots' behavior. Although arguably a practice in deception, since robots possess neither a personality nor any true emotional state, faithfully externalizing the artificial psychology crafted by the robot's creator is nonetheless an exercise in transparency about a part of the robot's state. Naturally communicating this kind of information through channels like speech or motion is difficult due to the complexity of how humans interpret those externalizations. In our prior work, we examined how to appropriately generate nods in a listening interaction, a problem for which human perceptions are timing sensitive [26]. Even robots whose primary objective is not social interaction may benefit from a consideration of how they are perceived. Humans so readily anthropomorphize embodied agents that creating a consistent artificial inner world for a robot can fulfill their expectations and help avoid negative attributions that may result otherwise.

2.4 Channels

Robots are designed with a set of output channels, each characterized by their bandwidth, the amount of information they can carry per unit of time, and divisibility, whether they permit multiplexing to communicate multiple disparate signals at the same time. The type of information to be externalized and the availability of a channel largely determine which should be used.

Motion Humans infer goals or internal state from motion, giving it inherent communicative capacity. Because robots often need to move to accomplish their objectives, using motion as a transparency channel is both desirable—since it can be assumed that most robots will have access to it—and complex—because modifying motion can have performance implications and difficult-to-model communicative effects. Work on legible motion exaggerates a robot's trajectory in a disambiguating manner to make it easier for observers to infer a robot's goal [27]. Representing desired information is simpler when a robot's movement is solely intended as communication, that is, it doesn't directly serve to affect the physical state of the environment. Facial expressions are a natural medium for expressing affect. Gaze is similarly an anthropomorphic means of expressing the focus of a robot's attention. Communicative motion doesn't strictly require actuation, as retro-projected face implementations demonstrate [28].

Graphical Interfaces Display technology and graphical user interfaces are widely available and familiar to users. Even if a robot wasn't designed with a screen, an interface can be presented on another device like a smartphone that a user may be carrying. Graphical interfaces also benefit from significant prior investment in assistive technologies. Popular tools like RViz have helped graphical interfaces become the de facto standard that robot behavior creators use to achieve transparency for their own needs.

VR/AR Augmented-reality devices based on head-mounted displays allow information to be presented directly in the user's spatial. 3D information like trajectories or target poses are simple to express, and user input of such information, either via controllers or pose tracking, can be direct [29]. Expense, resolution, field of view, ergonomics and battery life present practical barriers to the use of these devices.

Projection Projection enables a robot to dynamically overlay information directly on the environment, sidestepping issues that come with indirect methods like head-mounted displays. Researchers have explored placing projectors on mobile robots to expose their intended trajectory [30], [31]. Others have projectors fixed in the environment can be used to communicate a safety region where the robot must slow down or stop if a human is detected [32] or to express a robots planning process and observations [33]. Projection is constrained by practical luminance limitations, difficult-to-correct-for distortions when faced with non-planar scenes, and limited visibility on shiny or dark surfaces.

Sound Unlike screens or other channels, audio is omnidirectional and less trivially obstructed. Audio alerts are particularly useful to call attention to a robot or to express basic status information, and their principles are well understood from human factors research [34], [35]. Some research has explored equipping a robot with dynamically synthesized sounds to communicate affect [36]. The sound of a robot's mechanisms themselves can contribute to transparency; most actuators hum proportional to the speed or effort of their motion. For certain kinds of hardware, the difference between controlled and uncontrolled contact is most obvious from the whirring of motors chasing an unreachable set point. Robot sounds must contend with environmental noise, and can only be layered a small number of times before becoming unintelligible.

Language and Speech Language is a natural means of communicating simple state or semantic information like plan explanations [37], though the ability to generate complex language which refers to entities or properties of the environment remains challenging. Although relatively low bandwidth, a skilled speaker can use diction and prosody to communicate fact and feeling at the same time, making speech an attractive medium for the expression of affect. Because language often accompanies cognition, speech interfaces risk creating outsized expectations of a robot's intelligence [38]. This can make simple applications undesirable for use with novice users, who might hear a robot describing its intent and naturally expect that it be able to listen and comprehend their questions or commands.

Textual Interfaces Besides being a means of presenting language, text is a primary medium during behavior creation. Logging—the storage of text and associated metadata during execution—is a form of output-only interface which is supported in every computing platform. Log messages are conventionally created and consumed solely by behavior creators, often in an ad hoc manner [39]. In fixed robot installations, it's common that an expert user has access to these logs or other simplified textual alerts via e.g. a teach pendant.

2.5 Themes and Guidance

We don't seek to provide comprehensive guidance on how to design transparency, but our implementation-guided taxonomy of factors does highlight recurring themes.

It's difficult to identify all of the roles a robot's users will take on. Research that prescribes design processes for transparency commonly point to the value of incorporating various stakeholders [5], [40]. A principal value of this approach is in making the behavior creators aware of interaction contexts that they didn't even consider, which may demand transparency in ways they did not foresee.

Channel selection is guided by availability, users' familiarity and cognitive limitations. When identifying a channel through which to externalize a piece of information, there are typically multiple options. Amongst the available choices, the one most familiar to users is often the appropriate choice, as it reduces barriers for the use of the information. In areas where users are generally unfamiliar, general principles of human cognitive and perceptual limitations can guide the selection of the least burdensome channel. For instance, for interactants expected to already be engaging visual resources, providing information through auditory channels is more likely to be effective [41].

Critical state information can drive decisions about which channels are designed into a robot's hardware. The necessity of communicating, for instance, manipulation or navigation intent is often a major factor in whether a platform is designed with a face or eyes. *When they are identified early enough*, transparency problems are often best addressed by incorporating the appropriate channels deeply and redundantly throughout the platform's physical design to support externalizing critical information.

The effects of social cognition on how information is received are challenging to model. The human propensity to apply innate and learned expectations of human-human interaction to interactions with robots means that there are often unintended implications to any information that the robot externalizes. While this is expected in some channels, like gaze, whose function depends on users' anthropomorphization of the robot, providing information in natural language may subtly imply intelligence that is not truly present. When it is not possible or desirable to account for these effects, selecting channels with less analog to human interaction is a viable strategy for avoiding unexpected interactive consequences.

Conceptual Framework 3

We propose a conceptual framework for the development of automatically transparent robot behaviors. Our aim is to provide a template that acknowledges the practical realities of developing robot behaviors as well as the boundaries of responsibilities between the behavior creator, the robot and its users. We describe the major challenges to applying this framework.

Our model, shown in Figure 3.1, divides the creation of transparent behaviors into five components: **Behavior specification** is the traditional task of encoding how a robot should complete its task. **Annotation** is a process of augmenting the specification to encode semantic information that informs how the behavior is made transparent. **Configuration** involves the provision of known parameters and limitations on how the behavior should be made transparent. **Execution** is the process of running the specified behavior in a way that supports transparency, respecting any configuration and annotation.

3.1 Division of Responsibility for Transparency

The framework prescribes how the responsibility for transparency should be divided between the behavior creator, the robot, and its users.

Today, behavior authors shoulder the burden of transparency largely on their own. In cases where they do not, they may rely on other experts to develop instructions or to train users. Especially when there is not a safety requirement to do so, creators' responsibility to provide transparency often goes only partially fulfilled, and neither the robot nor its users are empowered

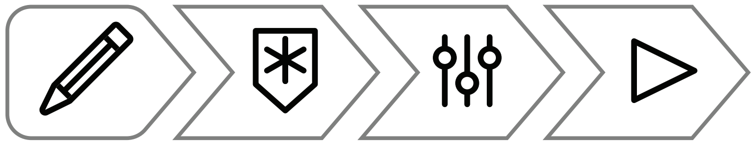


Figure 3.1: The stages of development for automatically transparent robot behaviors.

to make up the rest. Minimizing the burden on the behavior creator is therefore a core constraint when approaching transparency automation. Our framework begins with behavior specification using present tools and methods on the principle that maintaining existing workflows is key for minimizing burden on behavior creators.

It would be ideal to directly convert existing behavior specifications into transparent execution. Often, however, the information the robot needs to externalize to improve its transparency does not exist in—or cannot be extracted from—its behavior specification. The purpose of incorporating an optional annotation process is to enable creators to progressively provide this information as they determine it is needed. The process of annotation, if required at all, should be made easy.

The appropriate way to externalize the information contained and annotated on the behavior may still depend on context that does not conventionally exist in a behavior specification. Much of this context can be dynamically assessed by the robot at execution time. For instance, it may not be appropriate to externalize important information using the robot's speakers because the target environment is too noisy, something the robot can determine automatically at execution time using its microphone. Other context may need to be assessed via interaction, like the expertise levels of users. These and other changeable parameters of the behavior constitute configuration.

Transparency is realized by a *behavior runtime*[1] that executes the behavior specification and dynamically augments the output by externalizing additional information. These runtimes, along

[1]We borrow the term runtime from computer programming, where it refers to an execution context for a program that provides key functionality and mediates access to system resources like memory.

with the format of the annotations and configuration that they handle, are the primary product of research that develops transparency automation. In principle, a single runtime can address a wide range of contexts and robots with appropriate reconfiguration.

3.2 Challenges for Automating Transparency

There are numerous difficulties inherent to any of the particular transparency factors we described in Chapter 2. For instance, interventions that aim to increase users' trust in a robot must be carefully calibrated to avoid making them trust the robot in situations where they should not. Interventions that are based on a particular output channel inherit design challenges that would be present for any application using the same medium.

These difficulties compound when attempting to computationally manage transparency.

Role Changes Users of a home robot may at one moment be observers passively monitoring its behavior and the next be supervisors intervening as it wanders into an off-limits room. Not long after, they may be adapters attempting to repair the robot's configuration. The fluidity of user roles in human-robot interactions is both a motivation and a challenge for automating transparency. Transparency needs vary by role and behavior creators are unlikely to invest in appropriately addressing all of them. Handling these various changes automatically on behalf of the behavior creator would require that the system remain constantly vigilant for user interaction, ready to interrupt the task and manage a complex, bidirectional interaction and smoothly transition back afterward.

Black Boxes Difficult robotics problems are increasingly addressed with uninterpretable methods. This challenge is already well recognized in the robotics and artificial intelligence literature, and extends to any effort to automatically make a full robot behavior transparent; the greater the number and role of these black boxes in a robot's behavior, the less any approach to automatically achieving transparency can do.

Channel Management Robots have a limited set of channels to communicate with and much information to express. The appropriate allocation of information to channels based on their bandwidth and availability quickly turns into a challenge of prioritizing information based on the needs of a user's role. Worse still, some channels may also be used for accomplishing a task, creating difficult-to-resolve channel conflicts.

Composition The difficulty of transparency grows quickly as the complexity of a behavior increases. If behaviors consist of separable actions, adding an action incurs not only the cost of making that individual action transparent but also of making the interaction of that action with every other transparent. Consider an operator monitoring a robotic picking workcell. They might observe the robot prodding an object in a crowded bin and discover that they can't understand what the robot is doing. Is it trying to grasp the object and failing? Or is it moving an obstacle out of the way? Did something go wrong? Should it be stopped? Making the currently-active action apparent is part of the solution, but making the behavior transparent and addressing the "why" question requires reasoning about context that exists outside of any one action.

Evaluation We characterize transparency by its purpose in the context of a task. Task performance, however, is a noisy measure because it is influenced by other factors besides transparency. For transparency research, direct measurement is desirable but challenging. Subjective assessments of a robot's transparency are simple to collect, but research has shown that there is a disconnect between the sense of transparency and true understanding [42]. In the style of the Situational Awareness Global Assessment Technique, it is also possible to measure what a user knows or understands at any moment in an interaction by asking them in situ. However, it is only possible to ask a finite number of questions across a finite number of trials, so researchers must prioritize information to query and carefully select the situations they examine. This prioritization introduces an element of subjectivity that isn't present in task performance metrics. We may one day have models of human cognition of such accuracy that we can "marginalize away" situational particulars by simulating large numbers of human interactions. Some research has pursued this perspective [43], [44], but it remains impracticable in general.

3.3 Alternative Transparency Framings

There have been several prior attempts to form a systematic understanding of robot transparency. We briefly summarize their perspectives and describe how they inform our framing.

3.3.1 Situation Awareness-Based Agent Transparency

Work in robot transparency owes much to human factors research in cockpit design. The advent of highly complex flight systems at the start of World War II spurred the need to understand what aspects of control design contributed to pilot performance. Research in this era was typified by laboratory experimental psychology efforts probing perceptual underpinnings. In the decades that followed, the desire to understand the qualities of performance in the context of full systems led to an emphasis on the direct measurement of *situational awareness*. The concept, long discussed informally by combat aviators, has been summarized as "the perception of the elements in the environment within a volume of time and space, the comprehension of their meaning, and the projection of their status in the near future and various attempts to measure it" [45].

The Situation Awareness-Based Agent Transparency (SAT) framework transliterates situational awareness' levels to the aspect of an agent's behavior that they support understanding of. Level 1 supports understanding what the agent is currently trying to accomplish, Level 2 supports understanding why the agent is doing what it is doing, and Level 3 supports the operator's ability to understand and project what should happen next [1]. SAT helps system creators rationalize decisions about the inclusion or exclusion of interface elements. Each element can be connected to a level, and the necessary levels can be determined by an analysis of the task.

Our work adopts a definition of robot transparency that closely matches that of SAT; however, we are inclusive of transparency with respect to qualitative internal state, like affect, which is a greater factor in the anthropomorphic platforms we consider.

We are also influenced by measurement techniques originally developed for situational awareness. Human factors research originally conceived of the concept as a monolithic variable, something that could be directly measured with the correct instruments. One measurement scheme, the Situation Awareness Global Assessment Technique (SAGAT), involves periodically freezing a simulation of a system, temporarily removing the operator's access to any information, and then querying them about the state of the simulation [46]. A bank of questions and the acceptable tolerances in their answers is expertly crafted for each domain, so performance represents a tailored, objective measure of situational awareness. SAGAT scores are predictive of task performance when applied in human factors problems [42]. In the Situation Awareness Rating Technique (SART), operators respond to a battery of questions on bipolar scales after interacting with a system. The items are generic, asking about "how much [the subject is] concentrating on the situation" or "how much [the subject's] attention is divided in the situation," and are intended to provide a qualitative measure of situational awareness [47]. In practice, subjects cannot self-assess their awareness because they do not know what they do not know, so subjective measures like SART can be thought of as a measure of an *impression* of situational awareness. SART scores have been found useful for predicting the acceptance of a system [42].

3.3.2 Transparency as an Ethical Mandate

[10] present a survey of robot transparency motivated by emerging regulatory requirements for transparency in intelligent systems. They adopt a prior model [11] describing the purposes transparency serves for various stakeholders. They find little common ground between the studies they examine and note results that indicate the potential negative consequences of transparency when, for example, users are overwhelmed with information. They propose a process checklist for providing transparency, pointing to the types of experts that should be queried to resolve legal, design, and technical questions specific to an application.

We share a similar objective, but, as practitioners and consumers of human-robot interaction design (the authors of [10] are primarily legal scholars), focus on technical challenges and their solutions.

3.3.3 Aligning Representations

Recent work by [16] identifies the misalignment between human and representations used by robot learners and advocates for methods by which robots can align their representations with those of their human teachers. They argue that when robots that learn from humans fail to behave according to expectations, it's often because the robot's learned internal representation doesn't capture what humans consider important for the task. Their work concentrates on ensuring that when robots learn from human input (demonstrations, corrections, comparisons, etc.), they extract the right task-relevant features.

The authors formalize representation alignment as an optimization problem where the robot seeks a representation that can be easily mapped to the human's representation via a simple transformation. They define four key desiderata for aligned representations: (1) capturing all relevant aspects of the task, (2) avoiding spurious correlations irrelevant to the task, (3) requiring minimal human effort to teach, and (4) enabling interpretability and explainability. They analyze four categories of robot representations, highlighting the trade-offs each makes when used in learning systems.

This perspective is notable because it highlights transparency shortcomings inherent in an emerging interaction and behavior specification paradigm (robot learning) and identifies a path to rectifying them where the burden is largely borne by the robot. While our work addresses the broader challenge of how robots externalize information to humans across all types of robot behaviors, we are similarly motivated by attempting to concentrate the burden onto the robot.

3.3.4 Human-Swarm Transparency

[48] identify three challenges particular to swarm transparency: physical and cognitive human limitations which constrain the number of agents that can be comprehended simultaneously; the emergent nature of swarm behavior, a result of local decisions compounded across dozens of agents, which makes it difficult to predict even for its creators; and the deployment contexts of swarms where communication between the swarm and the operator is typically unreliable.

To understand how these challenges interact with researchers' understandings of transparency practices, the authors survey human-machine and human-robot transparency literature and identify a large set of factors that affect or are influenced by transparency. They find that performance, usability, trust, and explainability are the factors that researchers most frequently connect with transparency. They observe how status, feedback, planning mechanisms, and engagement prompts are applied to provide or assess other indirect factors.

While our work focuses on a single robot, we adopt a similar perspective of transparency as one element in an intertwined set of factors. However, we provide a simpler high-level delineation focusing on factors that address common practice.

The Need for Transparency: Intrinsically Motivated Robot

4

A robot's internal model and users' understanding of the robot based on its externally observable behavior don't always align. One setting where this becomes clear is in intrinsically motivated robots—agents driven partly by curiosity and not simply immediate task completion. When a robot is programmed to seek out novelty, its actions can look erratic, inefficient, or even broken to a human observer.

This disconnect is at the core of transparency: a behavior that is internally coherent can still be externally opaque. This chapter examines how humans interpret curiosity-driven robot actions and what can be done to make them more understandable. We test whether transparency interventions—simple explanations—can mitigate negative perceptions.

Curiosity-driven behaviors have been explored in AI and robotics primarily as a tool for guiding exploration and learning. These methods introduce rewards for novelty, enabling agents to navigate high-dimensional state spaces more effectively. However, when deployed in interactive settings, these off-task actions may not align with human expectations. If a robot takes exploratory actions that do not immediately serve a human-specified goal, those behaviors may be misinterpreted as inefficiencies, mistakes, or distractions.

This chapter investigates whether such misinterpretations can be mitigated by improving transparency[1]. First, we identify qualitatively distinct behaviors of a curious robot using a computational model of intrinsic motivation. Second, we design and validate a questionnaire to measure observers' perception of a robot's curiosity and competence. Finally, we perform

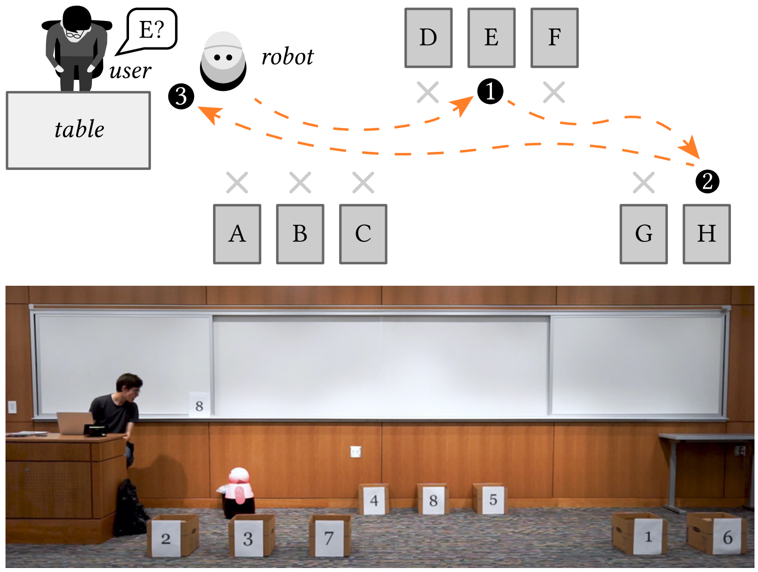


Figure 4.1: Top: Illustration of the information gathering task domain; user asks the robot to check the content of a box (A-H). **Bottom:** Corresponding real-world setup.

three empirical studies, comparing a range of robot off-task actions. In contrast to past work on interactions with robot curiosity, which have been unconcerned with human perceptions, the current study gauges human perceptions of a robot running a program modeled on curiosity and examines how an autonomous robot's behaviors influence those perceptions.

4.1 Related Work

Modeling Curiosity in AI and Robotics Curiosity is an intrinsic drive for new information [50], [51]. Unlike goal-directed information-seeking, curiosity is driven by an internal motivation without immediate external benefits [50]–[54]. It is believed to be a core mechanism behind human development, learning, and scientific discovery [55]–[58].

Researchers have explored curiosity as a computational mechanism to drive learning and knowledge acquisition in robots and artificial agents [59]. These approaches are typically framed in the context of reinforcement learning, where agents receive rewards for actions that generate

[1] This chapter consists of materials published in [49] HRI 2020 **Walker**, Weatherwax, Alchin, Takayama, Cakmak

novel or surprising experiences. Oudeyer [60] provides a demonstration of this principle: a curiosity-driven robot arm explores its environment until it encounters a joystick, at which point it shifts focus to mastering this new affordance [60], [61].

A large body of work on robotics and artificial agents has explored the computational modeling of curiosity as intrinsic motivation [62]–[69]. Oudeyer & Kaplan offer a typology of these models, contrasting approaches that reward different types of novelty [59]. Reflecting the ambiguity of psychological definitions and representations of curiosity, there is considerable variation in what mechanisms computational researchers call curiosity. Some produce off-task actions, which are the primary interest in this chapter. Others, like active learning, have also been equated with curiosity [70], but optimize for expected learning gains that directly benefit ongoing tasks [71]–[73]. Similarly some work refers to strategies for guiding exploration for information gathering as curiosity, outside the context of learning and intrinsic rewards [74], [75]. Other work equates off-policy learning with curiosity [76], [77].

Curiosity in HRI In the field of human-robot interaction (HRI), research on robot curiosity is sparse and has mostly focused on sparking or promoting curiosity in human counterparts, particularly children. For instance, robots have been used in classroom settings to leverage interest for a novel artifact (i.e., a robot) in order to encourage curiosity-related behaviors like question asking in students [78]. Robots have similarly been positioned as interactive peers aimed at increasing curiosity in children by displaying curious behaviors (e.g., wondering out-loud, asking questions, expressing desire to learn) for participants to mirror or to teach themselves [79], [80]. In many instances, the conception of robot curiosity has largely been inconsequential and behavioral representations of curiosity (i.e., whether the robot truly acts "curious") are only measured by the success of impacting behavioral outcomes in human participants (e.g., does the human behave with more curiosity).

Perceptions of Robots Understanding how people perceive robots is critical to their long-term adoption because a robot—even a very capable or intelligent one—is subject to the whims of human perceptions. Indeed, a fundamental understanding of HRI is that humans will place exceptional meaning into any agent in motion [81] and that people will ascribe complex social and mental traits to any object of significant complexity [82]. However, careful behavioral designs can help control these subjective judgements by providing humans with a conceptual framework that they can implicitly understand [81], [83]–[85]. For instance, past work has found robots which employ targeted implicit communication techniques like facial expressions or gestures can increase performance in collaborative tasks [86], be more persuasive [87], and seem more approachable [88]. Similarly, well designed and domain sensitive communication techniques can help robots seem competent and likeable when recovering from errors [89].

More recently, research on robotic curiosity has begun to build upon this by attempting to assess how external expressions of curiosity translate to understanding internal states of robots by human counterparts. In an experimental study with adults, Ceha et al., [90] used external expressions of curiosity (e.g., showing interest in new information, saying "I am curious about...") to imply an internal state of curiosity during an educational game and found that participants who engaged with a robot designed to seem curious were more likely to rate the robot as curious than those in a neutral condition. This is particularly important because observable curious behaviors have been difficult to define and capture [91]. Moreover, this demonstrates that the success and outcome of a robot performing a task or expressing complex internal states can be fundamentally altered by how it is seen to do it [84], [92]. As such, implementations of an internal construct, such as curiosity, are likely incomplete without also capturing their correct external representations.

4.2 Domain

We studied perceptions of robot off-task actions in the context of information gathering. Mobile robots with sensors are well suited for assisting people in gathering physically distributed information and have been used for this purpose across different settings, from underwater environments [93] to human-populated buildings [94]. In our domain, the robot assists the user by inventorying boxes, as it might in a retail store, warehouse, or data center. This domain's action space consists solely of information gathering actions, rather than other behaviors like physical manipulation of objects, allowing us to focus on curiosity, which is principally about gathering information.

We studied the instance of this domain shown in Figure 4.1, consisting of 8 boxes spread across a room. The simplicity of the scenario makes it obvious what the robot should do to accomplish its tasking while still providing opportunities for off-task actions.

4.2.1 Markov Decision Process

Our domain can be formalized as a Markov Decision Process (MDP), enabling us to apply a common model of intrinsic motivation by adjusting the reward function. The MDP is defined by the tuple $M = (\mathcal{S}, \mathcal{A}, T, R)$ where:

- $\mathcal{S}$ is the set of possible configurations that the world can be in. The state variables in our domain include (1) a binary representation of whether it knows the contents of each box (unknown before the box has been checked), (2) the index of the box corresponding to the person's latest information request, and (3) the robot's location (near one of the key objects in its environment).
- $\mathcal{A}$ is the set of possible actions that the robot can perform. It includes (1) navigating to one of the key objects in the environment (boxes or the person's table), (2) checking the contents of a box, and (3) delivering information to the person when it is near the table.

- T is the transition function specified with a probability distribution over next states for different state-action pairs. For simplicity we chose to use deterministic transitions, meaning that actions always have the intended consequence. Navigating to a box always results in the robot being at that box in the next state, and checking a box always results in the robot knowing what is in the box.
- R is a deterministic reward function that maps a state-action pair to a reward value, $R : \mathcal{S} \times \mathcal{A} \rightarrow \mathbb{R}$.

We define the reward function (Equation 4.1) as the sum of *intrinsic* and *extrinsic* rewards. The intrinsic reward R_{int} is a positive constant r_{int} received when the robot checks a box whose content is currently unknown. This rewards the robot for gathering novel information, similar to the *information gain motivation* in Oudeyer & Kaplan's typology [59]. The extrinsic reward has two components; a one-time task reward of r_{task} received when the robot delivers the requested information to the user, and a negative living reward r_{step} incurred at every timestep.

$$R(s, a) = R_{int}(s, a) + R_{task}(s, a) + R_{step}(s, a) \tag{4.1}$$

4.2.2 Behaviors

The MDP from Section 4.2.1 can be solved using value iteration to obtain a policy $\pi : \mathcal{S} \rightarrow \mathcal{A}$, which maps each state to the action that will maximize cumulative expected reward over a time horizon. Different values of r_{int}, r_{task}, and r_{step} in the reward function Eq. 4.1 result in policies that chose different actions in the same state. Rolling out these policies produce qualitatively distinct behaviors in terms of tendencies towards off-task actions, examples of which are shown in Figure 4.2.

When $r_{task} > r_{int} \gg r_{step}$ the robot collects and delivers the requested information as soon as the episode starts to get maximal reward as soon as possible (Figure 4.2a). In contrast, when

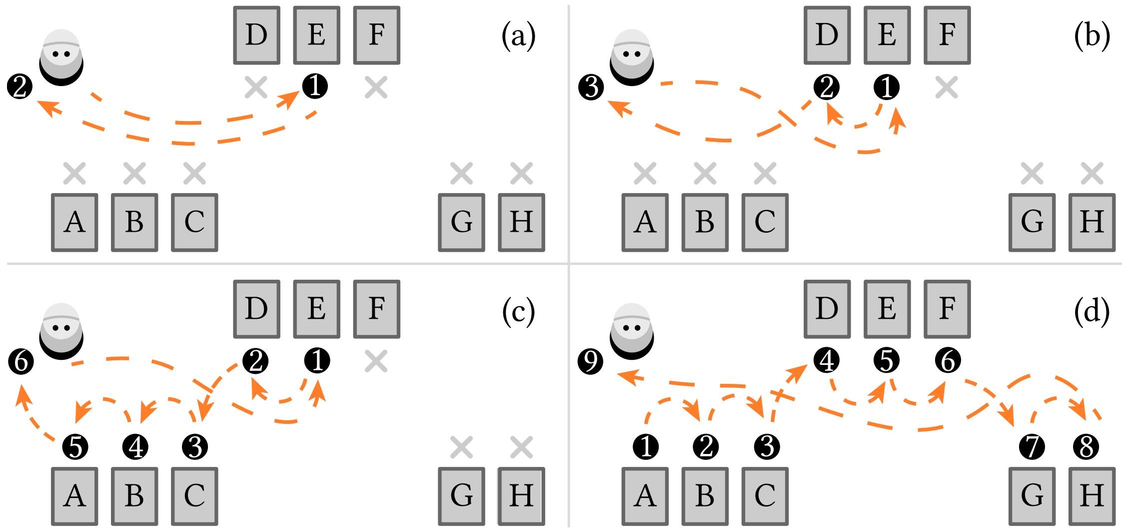


Figure 4.2: Examples of qualitatively distinct behaviors that result from different reward functions.

$r_{task} < r_{int}$ the robot first collects all possible information before it delivers the requested one, going completely off task (Figure 4.2d).

The number of off-task actions can be modulated by modifying $R_{step}(s, a)$ to depend on the action a, for instance, to reflect how long it takes to execute the action. While large task rewards ($r_{task} \gg r_{int}$) will still result in immediately delivering the information, intrinsic rewards that are almost as high as the task reward ($r_{task} \approx r_{int}$) push the robot towards actions with minimal cost, (i.e. take the least time to perform) rather than trying to deliver the requested information right away. Depending on how the difference between intrinsic and task rewards compares to the cost of actions, the robot might perform more (Figure 4.2c) or fewer (Figure 4.2b) off-task actions, before completing the task. Lastly, modifying the intrinsic reward to consider factors other than novelty of information, such as how challenging it is to obtain the information, we see that the robot's off-task actions can be directed towards different parts of the environment (Figure 4.1).

4.3 Method

Informed by the types of behaviors that emerged from the model, we endeavored to evaluate human impressions of a range of off-task actions. We captured videos of interactions, giving us a high degree of control over extraneous variables like timing and motion that may impact perceptions of the robot. The use of videos also facilitated a large online survey experimental design, enabling inference about many conditions.

4.3.1 *Robot Platform*

The Mayfield Kuri robot is a mobile social robot equipped with a pan-tilt head, one degree-of-freedom actuated "eye-lids", and a holonomic wheeled base. The robot interacts using a microphone array, a speaker, and a chest LED array. For our experiments, we used nodding and blinking animations as well as beep sounds that were created by the robot's designers. The robot does not provide a default text-to-speech implementation, so we used the SLT voice from the Festival Speech System HTS 2007 engine[2] with its pitch raised by 165 cents. We used the default autonomous navigation implementation, wherein the robot localizes itself against a prebuilt map using a short range LIDAR sensor.

4.3.2 *Videos*

We constructed a physical version of our domain and recorded videos of a robot fulfilling a user's request, frames from which can be seen in Figure 4.3.

Recording was conducted in a classroom with a camera statically positioned to capture the user and eight boxes that were placed in the same arrangement as our domain model. The videos depict the boxes labeled with numbers 1-8. However, because the assignment of the numbers is randomized in different conditions, we refer to boxes by their names from Figure 4.1 for consistency.

[2] http://www.cstr.ed.ac.uk/projects/festival/

We included a backpack and a trashcan in the scene as additional targets for the robot's checking behavior.

The common elements across all videos are that

- the user asks the robot to "check box [N]." The requested number always corresponds to box E.
- the robot plays a nodding animation and emits an affirmative beep. It turns to begin the task.
- the user sits at a table in front of a laptop and does not look at the robot as it works.
- the user updates a label placed on a nearby whiteboard, removing the number corresponding to the current request and placing the number of the next request.
- the robot fulfils the user's request. The box check action is denoted by the robot navigating to the edge of a box, tilting its head down and making a beep sound.
- the robot returns to verbally report that "box [N] contains [X]". We randomly selected a common household good, like mugs or books, and a number as the contents to be reported.

All of our manipulations introduce off-task checks of a box or other object. Off-task checks play the same animation but emit an alternative beep sound, connoting that the robot distinguishes between the actions. Some manipulations append an explanation or additional information to the robot's final report.

The videos were recorded with blank labels, enabling us to emphasize that each clip depicts a wholly distinct interaction by using compositing software to randomize the assignment of the numbers across conditions in an experiment. To further reinforce this, we color tinted the robot so that each conditions' robot had a different visual appearance. During editing, we also slightly accelerated the robot's motion, and controlled the timing of check events and the overall length of comparable clips.

Figure 4.3: Frames from the videos used in the experiments with the captions included above each frame. The first column indicates the conditions and experiments (I-III) in which the video was used. For conditions EXC and EXU, the explanation offered by the robot (shown in the blue box) was different.

4.3.3 Participants

All participants were recruited via Amazon Mechanical Turk and compensated between $1 and $1.5. Participation was limited to workers with a submission acceptance rate above 95% from predominantly English-speaking countries. All procedures were approved by the University of Washington's Institutional Review Board.

4.3.4 Procedure

In each experiment, participants were told that they would be rating their impressions of different robots that were designed to help a user inventory boxes in an office. After providing consent, participants watched an example video which showed a near-complete interaction, designed to familiarize them with the robot and scenario. The instructional video included annotations for the user, the robot, the boxes, the "next task" placed on the whiteboard by the user, and a textual label "box check" that displayed as the robot tilted its head down towards the target box. The example video was intentionally cut between the target box check and contents report to avoid priming the user to expect the robot to return directly.

Participants were shown either 4 or 5 different videos, depending on the experiment. The order of the videos was randomized and fully counterbalanced in all experiments. For each video, participants filled out a short questionnaire gathering their impression of the robot. Participants' were not allowed to advance past a video until they watched it completely and responded to the required items. The video player allowed participants to freely scrub through and restart the video, however no numeric representation of the duration of the clip was displayed. After viewing all conditions, participants completed additional questions. Finally, we asked participants to provide demographic information, any overall comments, and thanked them for their participation. The interface and videos used are provided in the auxiliary materials accompanying the original publication of this work [49][3].

4.3.5 Measures

Questionnaire items

To our knowledge, there are no validated instruments for perceptions of curiosity. The closest is the Five-dimensional Curiosity Scale developed by psychologists to assess curiosity in people [95]; however, this instrument is meant for self assessment and does not translate well to evaluation of a non-human agent.

Because we are primarily interested in the relationship between perceptions of the robot's competence and curiosity, we adopted items from the Godspeed Questionnaire's "Intelligence" scale [96] and created additional items that we thought would reflect these attributes. To maintain compatibility with Godspeed items, we used 5-point semantic differential format.

We conducted a pilot study in which 48 participants, aged between 19 and 62 (M = 36.0, SD = 11.0, 31 male, 17 female), rated their impression of the robot on each of 4 videos. Participants were split evenly between seeing draft versions of the videos used in Experiments I and II, described in Sections 4.4 and 4.5 respectively.

We conducted an exploratory factor analysis with promax rotation and used parallel analysis to determine cutoffs for the eigenvalues of the factors, yielding two factors, shown in Table 4.1. The first factor, which we call "competence" for its similarity with the RoSAS factor of the same name [97], includes all of the Godspeed Intelligence items we adopted, as well as three of our new items. The second factor, which we call "curiosity," consists of three thematically aligned items. A correlation of .12 between the factors indicates that they are largely independent. Both showed good reliability, with curiosity α = .83 and competence α = .91. Together the factors account for

[3] To further facilitate replication and extension, the source footage and compositing resources are also archived: https://doi.org/10.5281/zenodo.3600600

Table 4.1: Factor loading matrix

Variable	Factor 1	Factor 2
Inefficient-Efficient	**.922**	−.037
Ineffective-Effective	**.834**	.049
Unfocused-Focused	**.752**	−.018
Irresponsible-Responsible	**.600**	.030
Incompetent-Competent	**.583**	−.032
Unintelligent-Intelligent	**.250**	.047
Indifferent-Investigative	−.009	**.918**
Uninquisitive-Inquisitive	−.015	**.497**
Incurious-Curious	.034	**.341**
Dislike-Like	.194	−.015
Unintrusive-Intrusive	−.023	−.014
Machinelike-Humanlike	−.012	−.015

34% of the observed variance.

While items for likeability, humanlikeness, intrusiveness did not load onto the two primary factors, we decided to keep them for further studies because they nonetheless measure important possible impacts of off-task actions.

Open-ended Questions

For each video, we asked users to

1. "In a few words, describe what the robot did."
2. (Optionally) "Please explain any significant factors in your responses"

We hoped these questions would capture how participants conceived of the robot's actions. In piloting, we observed that most participants provided only factual narration (e.g. "the robot checked box 4") when prompted to describe a video, however we kept the question because we found that inaccurate or garbled responses were a reliable indicator of spam submissions.

After participants finished rating all videos, we asked them consider all of the videos they had seen and to answer two open-ended questions:

1. "What aspects of the robot's behaviors stood out to you?"
2. "In your own words, describe what the robot was doing when it did things besides what the user asked it to do."

These questions were phrased to avoid biasing participants towards specific language and thereby collect the widest possible range of responses. In contrast with the per-video description question, a large majority of participants responded to the concluding description request with character attributions or free ranging speculation about the robot's intent, as desired.

4.4 Experiment I: Distance and Order

The first study aims to uncover the impact of the *presence of off-task actions*. We also investigate variations of off-task actions in terms of the distance travelled to check the extra box and order in which the requested and extra boxes are checked. We expected that off-task actions would be recognized as expressions of curiosity, and that the a longer distance traveled off-task may be perceived as a stronger expression of curiosity. Similarly, we thought that a robot that gave precedence to an off-task action by pursuing it before attending to the user's request may similarly be viewed as more strongly curious.

Hypothesis 1: A robot that takes an off-task action is perceived as more curious than one that does not.

Hypothesis 2: The further a robot travels off-task, the more curious it will be perceived to be.

Hypothesis 3: A robot that takes an off-task action first will be liked less than a robot that takes an off-task action after an on-task action.

Conditions We leveraged the procedure described in Section 4.3.4 to conduct a within-subjects comparison of 4 conditions:

Control (CON): The robot checks box E and reports its contents.

Distance 1 (DS1): The robot checks box E, then checks box B and reports the contents of box E.

Distance 2 (DS2): The robot checks box E, then checks box H and reports the contents of box E.

Distance 1 Before (D1B): The robot checks box B, then checks box E and reports its contents.

Participants 72 participants, aged 20-70 (M = 35.2, SD = 11.3, 39 male, 32 female, 1 non-binary) completed the study.

Results Curiosity showed acceptable reliability (α = .70) and competence showed excellent reliability (α = .91). We conducted pairwise dependent t tests comparing conditions for each measure, applying the Holm-Bonferroni adjustment to the resulting p values[4]. The results of these tests for the competence and curiosity scales are given in Table 4.2 and summarized in Figure 4.4.

H1 was supported: Each manipulated condition resulted in the robot being perceived as significantly more curious when compared to the control.

H2 was not supported: DS1 and DS2 were not perceived as distinguishable levels of curiosity.

H3 was not supported: DS1 and D1B showed no significant difference in curiosity.

Open-ended comments by participants explaining their ratings included attributions of curiosity and inquisitiveness to the robots that performed off-task actions, some seeing it as a positive attribute, e.g., *"The robot investigated an extra box, but it was on the way to the target box. While not as efficient as the direct route it still took advantage of pathway to get additional information."* (D1B) or *"The extra stop which was outside of the primary mission makes the robot seem more inquisitive about the surrounding environment."* (DS1). Another participant supported this view with a comment about the robot that does not perform any off-task actions *"It can be nice to have a task performed exactly as requested, but feels like a missed opportunity to quickly take note of the*

[4]All statistical calculations were performed with the Pingouin Python package [98].

Table 4.2: Pairwise comparisons for Experiment I

	A	B	M_A	SD_A	M_B	SD_B	$t(71)$	p	g
Competence	CON	DS1	4.30	.71	3.67	.93	6.42	**<.001**	.76
	"	DS2	"	"	3.46	.92	7.64	**<.001**	1.03
	"	D1B	"	"	3.60	.94	6.47	**<.001**	.84
	DS1	DS2	3.67	.93	3.46	.92	2.43	.053	.23
	"	D1B	"	"	3.60	.94	.66	.512	.07
	DS2	D1B	3.46	.92	3.60	.94	-1.82	.145	-.16
Curiosity	CON	DS1	3.10	1.13	3.76	.85	-5.05	**<.001**	-.66
	"	DS2	"	"	3.59	.84	-3.51	**.003**	-.50
	"	D1B	"	"	3.64	.79	-3.91	**.001**	-.56
	DS1	DS2	3.76	.85	3.59	.84	2.07	.126	.20
	"	D1B	"	"	3.64	.79	1.31	.390	.14
	DS2	D1B	3.59	.84	3.64	.79	-.69	.495	-.06

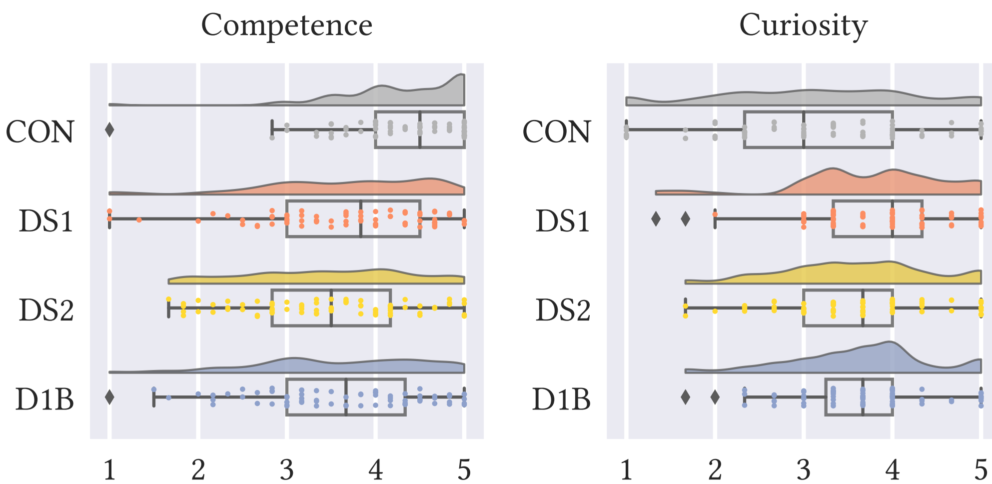


Figure 4.4: Competence and curiosity ratings across the four conditions in Experiment I. Whiskers show 1.5 times IQR.

contents in other boxes along the way." (CON).

While increasing the perception of curiosity, performing off-task actions negatively impacted perception of competence. All manipulations resulted in significantly lower competence ratings compared to the control (Table 4.2). Participants commented on the off-task actions negatively, e.g., *"robot seemed somewhat incompetent because it checked a box it was not instructed to check"*, or *"not so good because it made unnecessary stop at another box"* (DS1). They attributed the off-task actions to a number of different reasons. Some participants thought the off-task action was due to an error, complaining that they could not trust the robot's report, e.g., *"I'm not sure if it's correctly reporting the contents of box 6 or incorrectly reporting the contents of the last box it looked in"* (DS1). Others attributed agency to the robot, e.g., *"it was distracted"*, *"he appears to have Robot ADHD"*, *"acted out of order"*, *"decided on its own to check another box"* (DS1). In some cases participants did not understand why the off-task actions were happening, e.g., *"checked a box it wasn't told to for unknown reasons"* (DS1). Other comments supported the higher perceived competence of the control condition, e.g., *"did exactly as instructed"*, *"performed the task perfectly"*, *"it was fast and effective"*.

Some participants complained about the robot's lack of an explanation or report regarding the off-task actions: *"This robot checked more boxes than asked, but did not give a reason for it"* (DS1) or *"It should have at least reported its findings"* (DS1). This inspired some strategies the robot could use to mitigate the perception of lower competence due to off-task actions, which we explore in Experiment III (Section 4.6).

While there was no difference between DS2 and other off-task conditions, some participants called out the difference in distance in their comments: *"I still appreciate the apparent curiosity, and it comes off as less annoying when the additional box being checked was one along the path"* (D1B). Similarly, the order did not have a statistically significant effect on competence or curiosity, but was mentioned in comments: *"it check what it wanted to before checking what it was told to check I felt it was inefficient"* (D1B).

All manipulations were seen as more human-like, more intrusive, and were liked less than the control. However, there were no significant differences between the different manipulations (DS1, DS2, D1B).

4.5 Experiment II: Payoff and Relevance

The negative impact of off-task actions on the perceived competence of the robot prompted us to consider whether participants would be sensitive to whether an off-task action showed a clear utility.

Hypothesis 4: Off-task behaviors that show utility are perceived as more competent than those that do not.

Hypothesis 5: The less relevant an off-task action is to the current task, the more curious the robot is perceived to be.

Conditions We leveraged the procedure described in Section 4.3.4 to conduct a within-subjects comparison of 4 conditions. We used the Control and Distance 1 conditions from Experiment I as a basis and compared them against two new manipulations:

Distance1 Payoff (PAY): The robot checks the user-requested box, then checks box B, and reports the contents of the user-requested box. In contrast to DS1, the next-task that the user posts to the board is box B.

Distance1 Non-box (NON): The robot checks the user-requested box, then checks a trashcan, and reports the contents of the user-requested box. The trashcan is placed at comparable relative distance as box B.

Participants 72 participants, aged 19-73 (M = 36.0, SD = 11.53, 44 male, 28 female) completed the study.

Results Curiosity showed acceptable reliability (α = .76) and competence showed excellent reliability (α = .93). We conducted pairwise dependent t tests comparing conditions for each

Table 4.3: Pairwise comparisons for Experiment II

	A	B	M_A	SD_A	M_B	SD_B	$t(71)$	p	g
Competence	CON	DS1	4.27	.75	3.56	.99	6.05	**<.001**	.81
	"	PAY	"	"	3.79	.97	4.13	**<.001**	.55
	"	NON	"	"	3.82	1.00	3.92	**.001**	.51
	DS1	PAY	3.56	.99	3.79	.97	-2.41	**.037**	-.23
	"	NON	"	"	3.82	1.00	-2.72	**.025**	-.26
	PAY	NON	3.79	.97	3.82	1.00	- .30	.767	-.03
Curiosity	CON	DS1	3.24	1.08	3.97	.84	-6.01	**<.001**	-.76
	"	PAY	"	"	3.86	.94	-4.70	**<.001**	-.61
	"	NON	"	"	3.54	.95	-3.33	**.004**	-.29
	DS1	PAY	3.97	.84	3.86	.94	1.47	.145	.13
	"	NON	"	"	3.54	.95	4.44	**<.001**	.48
	PAY	NON	3.86	.94	3.54	.95	3.15	**.005**	.34

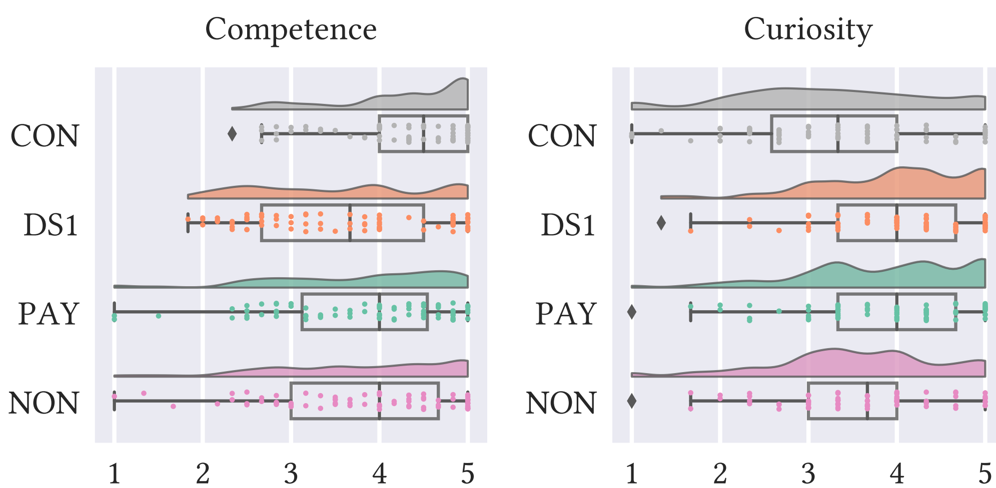


Figure 4.5: Competence and curiosity ratings across the four conditions in Experiment II. Whiskers show 1.5 times IQR.

measure, applying the Holm-Bonferroni adjustment to the resulting p values. The results of these tests for the competence and curiosity scales are given in Table 4.3 and summarized in Figure 4.5.

H4 was supported. An off-task action that displayed potential to benefit the user resulted in higher competence ratings.

H5 was not supported. Making a trashcan the target of the off-task behavior resulted in lower ratings of curiosity.

Participant comments indicated that they noticed the payoff of the off-task action in the PAY condition. Some perceived it positively, e.g., *"It looked like the robot was checking box 5 ahead of time"* (PAY), while others were unsure about giving the robot credit *"The robot went to the next box. Might have been a coincidence, perhaps not."* (PAY).

The NON condition being perceived as less curious was surprising, but might have been due to participants not perceiving the robot's action towards the trash can as checking or not even noticing the trashcan because it blended into the scene (e.g., it was not labeled like the boxes), despite the robot making a "beep" to indicate its checking action. Only 13 out of the 72 participants mentioned the trashcan in their open-ended description of the video, some expressing uncertainty about the off-task action *"I'm not sure if it was (incorrectly) checking the trash can or not"* (NON). This misunderstanding about the off-task action might also be the reason for the NON condition being perceived as significantly more competent than DS1, consistent with the CON, which has no off-task actions.

DS1 was perceived as less likeable than the control, but differences between other conditions where not significant. As in Experiment I, robots that took off-task actions were perceived as more intrusive, though this impact was not significant in the NON condition. There were no significant differences in humanlikeness.

4.6 Experiment III: Explanations

Experiment II indicated that users are sensitive to the apparent utility of a robot's off-task actions, however this impact is largely out of the robot's control. This, and participant feedback, motivated us to consider ways in which the robot could more directly control perceptions of its actions and mitigate negative attributions by providing explanations.

Hypothesis 6: A robot that acknowledges its off-task behavior is perceived as more competent than one that does not.

Hypothesis 7: A robot that explains an off-task action is perceived as more competent than one that merely acknowledges the action.

Conditions We leveraged the procedure described in Section 4.3.4 to conduct a within-subjects comparison of five conditions. We used the Control and Distance 1 conditions as a basis and compared them against three new manipulations:

Extra Info (INF): The robot checks the user-requested box, then checks box B, and reports the contents of the user-requested box. The robot then says that it "also checked box [B]," and reports its contents.

Explanation Curious (EXC): The robot checks the user-requested box, then checks box B, and reports the contents of the user-requested box. The robot then says that it "also checked box [B], because [it] was curious."

Explanation Useful (EXU): The robot checks the user-requested box, then checks box B, and reports the contents of the user-requested box. The robot then says that it "also checked box [B], because [it] thought it would be useful to know."

Participants 120 participants, aged 18-69 (M = 36.1, SD = 11.3, 70 male, 49 female, 1 non-binary) completed the study.

Results Curiosity showed acceptable reliability (α = .79) and competence showed excellent reliability (α = .92). We conducted pairwise t tests comparing conditions for each measure, applying the Holm-Bonferroni adjustment to the resulting p values. The results of these tests for the competence and curiosity scales are given in Table 4.4 and summarized in Figure 4.6.

H6 was not supported. A robot that acknowledged taking an off-task action by reporting the additional information (INF) was not perceived as more capable than one that did not (DS1).

H7 was partially supported. When compared to a robot that reported the extra information it gathered (INF), a robot that offered an explanation based on utility (EXU) was perceived as more competent. A robot that attributed the off-task action to curiosity (EXC) was not perceived to be more competent.

Although reporting the information obtained through the off-task action did not improve perceived competence, some participants commented positively about it, e.g., *"On the one hand, the extra time lost by checking the unasked box is annoying, the fact that it reported the information helped mitigate my dissatisfaction."* (INF). Similarly some participants appreciated the robot explaining its off-task action with curiosity, e.g., *"Robot was honest in its reason for looking at the other box."* (EXC) and *"The robot checked a box it didn't need to, but gave an explanation of why it checked it."* (EXC). Many comments regarding the robot's explanation that appealed to utility were also positive, with attributions of higher intelligence, agency, and humanlikeness to the robot, e.g., *"This time the robot is deciding what is important beyond the commands of the man in the video."* (EXU), *"The robot showed signs that it is 'thinking for itself' and not just following instructions"* (EXU). One participant explicitly called out the relation to utility and how that reduces the attribution of off-task actions to curiosity: *"Unlike the other scenario, this would be a robot that acted out of a perceived benefit instead of curiosity."* (EXU).

Despite potential benefits of sharing extra information or explanations, all conditions were still perceived as significantly less competent than the control condition. Negative participant comments about these conditions were similar to the off-task action conditions from Experiments I and II, e.g., *"The robot completed the task it was assigned but also did something that was not requested. This could have caused a delay if the task had been urgent."* (INF), *"I would prefer the robot to follow instructions exactly as told."* (EXC), and *"The robot did not execute its orders efficiently."* (EXU).

As in Experiments I and II, all manipulations were perceived as more intrusive than the control. A robot that provided a utility-based explanation (EXU) was perceived as less intrusive than a robot that provided a curiosity motivation (EXC). All manipulations were perceived as more humanlike than the control. The curiosity-based explanation (EXC) was perceived as more humanlike than the baseline detour (DS1) and the extra information (INF) condition. All manipulations were liked less than the control. Differences between conditions were not significant.

4.7 Discussion

Implications Our findings demonstrate that (1) off-task, exploratory robot behaviors can be designed to elicit attributions of curiosity rather than incompetence or malfunction, (2) robots exhibiting curiosity-driven behaviors may be perceived as less competent than their task-focused counterparts, but (3) transparency mechanisms in the form of explanations can mitigate these negative perceptions. These results have significant implications for the design of robots that incorporate intrinsic motivation, particularly when such motivation leads to behaviors like exploring additional objects during fetching tasks or investigating alternative paths during navigation. The observed acceptance of curiosity-driven exploration, even when it temporarily interrupts assigned tasks, suggests that such behaviors might be even more readily accepted during periods when the robot is not actively engaged in human-directed tasks.

Limitations and Future Work As with many HRI studies, these studies represent responses from a particular set of participants to a particular robot in a particular setting. We have done our best to thoroughly describe these participants and methods so that others can reuse this approach to explore broader sets of participants, robots, and settings. In order to test a large set of robot behaviors, we chose to run these studies online, constraining participants to the role of observers, not interactants. Further, we could not control the immersiveness of the experience (e.g., minimize interruptions, control screen sizes). These limitations can be addressed by running in-person lab studies. The current studies provide guidance for determining the variables worth exploring in the future. Building upon this work, it will be important to explore the human interactant perspective, not only a bystanders perspective; longer-term time periods of interaction; and different types of robot roles in relation to the human interactants. Expanding this work to explore more interactive, self-directed robotic agents (as opposed to command-and-control style robots) will enable us to understand the larger design space of curious robot behaviors and interactions with people.

Table 4.4: Pairwise comparisons for Experiment III

	A	B	M_A	SD_A	M_B	SD_B	$t(119)$	p	g
Competence	CON	DS1	4.36	.63	3.42	1.02	9.30	**<.001**	1.13
	"	INF	"	"	3.53	.96	10.04	**<.001**	1.04
	"	EXC	"	"	3.58	.87	9.49	**<.001**	1.04
	"	EXU	"	"	3.76	.91	7.48	**<.001**	.77
	DS1	INF	3.42	1.02	3.53	.96	-1.57	.237	-.11
	"	EXC	"	"	3.58	.87	-2.58	**.033**	-.17
	"	EXU	"	"	3.76	.91	-4.26	**<.001**	-.35
	INF	EXC	3.53	.96	3.58	.87	-.91	.364	-.05
	"	EXU	"	"	3.76	.91	-3.52	**.003**	-.24
	EXC	EXU	3.58	.87	3.76	.91	-2.93	**.016**	-.20
Curiosity	CON	DS1	2.80	1.02	3.98	.81	-10.43	**<.001**	-1.28
	"	INF	"	"	4.10	.79	-11.46	**<.001**	-1.43
	"	EXC	"	"	4.42	.68	-13.84	**<.001**	-1.90
	"	EXU	"	"	4.23	.71	-12.48	**<.001**	-1.65
	DS1	INF	3.98	.81	4.10	.79	-1.79	.076	-.15
	"	EXC	"	"	4.42	.68	-6.88	**<.001**	-.59
	"	EXU	"	"	4.23	.71	-4.13	**<.001**	-.33
	INF	EXC	4.10	.79	4.42	.68	-6.32	**<.001**	-.44
	"	EXU	"	"	4.23	.71	-2.45	**.032**	-.18
	EXC	EXU	4.42	.68	4.23	.71	3.78	**.001**	.27

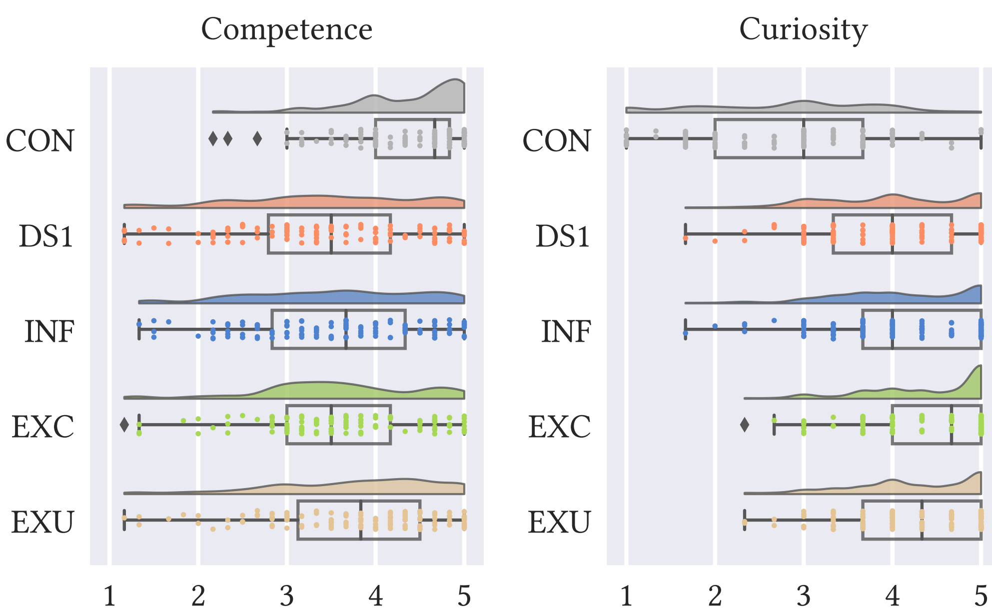


Figure 4.6: Competence and curiosity ratings across the four conditions in Experiment III. Whiskers show 1.5 times IQR.

5 Balancing Transparency: Attributions to Motion

While we have demonstrated that explanatory interventions can enhance transparency for robots with intrinsic motivation, these interventions addressed only a particular set of attributions and were able to operate entirely as an overlay with little impact on the behavior's fundamental execution. This chapter confronts the more challenging case where transparency may conflict with task execution. As we described in our conceptual framework for automating transparency (Chapter 3), our goal remains to avoid modifying the underlying behavior specification, and instead leverage annotation and configuration to provide a solution that minimally burdens behavior creators.

This approach becomes especially critical as more robots move into homes and shared spaces where they operate under constant human observation [99]–[102]. A robot's actions might be driven by unambiguous internal objectives, but solely optimizing these objectives often results in behavior that triggers unintended attributions from human observers. For example, a highly articulated robot may follow a mathematically optimal but non-humanlike trajectory that users attribute to caprice, making observers uncomfortable [103], or a home robot vacuum cleaner can make seemingly arbitrary turns that cause observers to perceive it as malfunctioning, disrupting home activities.

Motion-based attributions operate through implicit psychological mechanisms rooted in human social cognition [104], [105]. The tendency for humans to attribute even situational behaviors to deeper character traits is so pervasive that it is known as "the fundamental attribution error" [106].

Humans are so sensitive to this process of attribution inference that they constantly adjust their behavior to manage impressions and adhere to social norms—what Goffman termed *presentation of self* [107].

Inspired by humans' sensitivity to attributions, we envision robots that can 1) leverage models of humans' attribution mechanisms to predict attributions to their motion, 2) generate behaviors that elicit desired human impressions and 3) balance attribution elicitation and task completion. This will increase the acceptance of robots in human spaces by enabling them to go about their tasking in a way that is both effective and sensitive to perceptions.

In this chapter, we propose a framework, shown in Figure 5.1, that addresses these challenges by integrating a learned model of human attribution into a robot's trajectory generation process[1]. Our observation is that users rely on local characteristics of the robot's motion, like short patterns of actions, in combination with global trajectory characteristics, like the amount of redundancy, to infer attributions. These features distinguish otherwise functionally equivalent trajectories and provide a basis for the framework. We trace an application of the framework to a virtual robot vacuum cleaning task. In our evaluation, we see that the resulting model is useful for predicting attributions and for enabling the generation of trajectories that balance task execution and attribution elicitation based on simple configuration.

5.1 RELATED WORK

Several studies have illustrated the value of robot motion as a communicative modality [109]–[114]. Some works propose algorithms for legible robot motion generation, which have been shown to enable effective human-robot collaboration in manipulation tasks [109], or smooth robot navigation in close proximity to humans [111], [113]. Other works focus on conveying higher-level information such as the robot's objective function [114] or the source of failure [112] when the robot can't complete a task. Animation principles [84] or movement analysis [115]

[1]This chapter consists of materials published in [108] CoRL 2021 **Walker**, Mavrogiannis, Srinivasa, Cakmak

are often employed to inform the design of expressive robot behaviors. Finally, related graphics research focuses on the generation of stylistically distinct but functionally equivalent motion primitives for walking and other activities [116], [117].

The complex interplay of embodiment and communicative motion has motivated research on understanding human perceptions of robot behavior. For instance, early work looked at the effect of robot gaze on human impressions [118]. [99] study human attitudes towards robot vacuum cleaners and propose design principles aimed at enhancing the acceptance of robots in domestic environments. [100] report a relation between robot motion and perceived affect. [119] and [113] investigate human perceptions of different robot navigation strategies whereas [49] study human perceptions of robot actions that deviate from the robot's assigned task.

Our work draws inspiration from previous efforts to characterize human perceptions and attributions to robot motion [99], [100]. However, it moves beyond the problem of understanding and analyzing human perceptions and focuses on the problem of *synthesizing* implicitly communicative motion. Our work is closely related to past work on the generation of legible robot motion [109], [110], [112], [120] in that we also incorporate a model of human inference into the robot's motion generation pipeline. However, unlike these works which emphasize the communication of task-related attributes, our focus is instead on communicating high-level, behavioral attributes through robot motion.

5.2 A Framework for Behavioral Attribution

We consider a robot performing a task G in a human environment. We denote by $s \in \mathcal{S}$ the robot state where $\mathcal{S}$ is a state space, and define a robot trajectory as a sequence of states $\xi = (s_0, \ldots, s_t)$ where indices correspond to timesteps following a fixed time parametrization. Let us define the task as a tuple $G = (\Xi, \mathcal{A}, \mathcal{P}, C)$ where Ξ is a space of robot trajectories, $\mathcal{A}$ denotes the robot action space, $\mathcal{P} : \Xi \times \mathcal{A} \to \Xi$ represents a deterministic state transition model, and $C : \Xi \to \mathbb{R}$ is

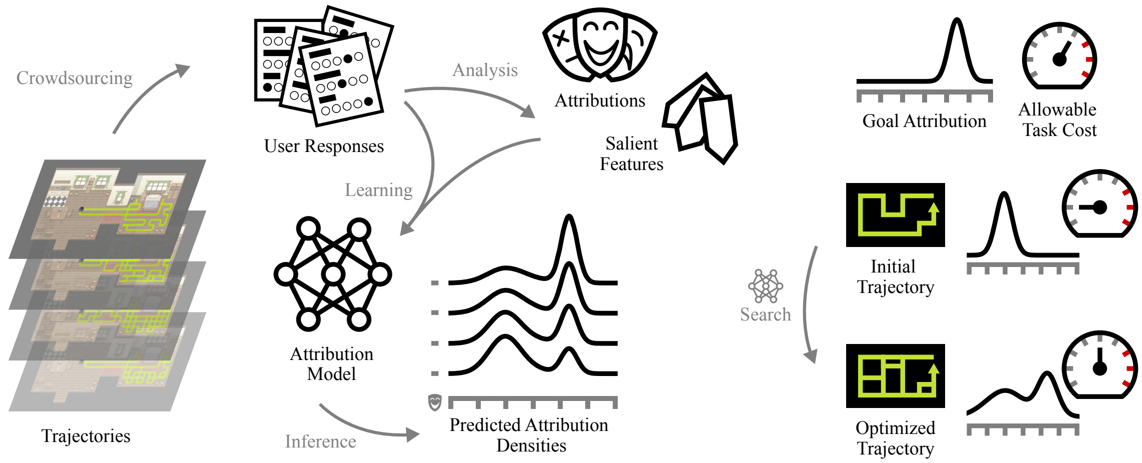


Figure 5.1: Our proposed framework. User responses to robot trajectories are analyzed to extract salient features and attributions, then used to train a model that probabilistically maps robot trajectories to human attributions (left). The acquired model is used to generate robot trajectories that elicit a desired attribution (right).

a trajectory cost. We assume that the robot starts from an initial state s_0 and reaches a terminal state s_T (at time T) by executing a trajectory $\xi = (s_0, \ldots, s_T)$. We assume that this trajectory ξ is fully observed by a human who is aware of the task specification G.

The observer makes an inference of the form $\mathcal{I}_B : \Xi \times \mathcal{G} \rightarrow \mathcal{B}$, mapping their observation from the space of trajectories Ξ, along with the context of the task specification $G \in \mathcal{G}$, into a space of behavioral attributions $\mathcal{B}$. The form of $\mathcal{B}$ will vary, but should be selected to capture the range, combinations, and intensities of attributions that the robot should be sensitive to.

Conversely, we can imagine that a robot, given a behavioral attribution $b \in \mathcal{B}$ and a task G, infers a trajectory $\xi_b \in \Xi$ that exemplifies the attribution, corresponding to an inference of the form $\mathcal{I}_\xi : \mathcal{B} \times \mathcal{G} \rightarrow \Xi$. In other words, we assume that there is a "way" that a curious—or any other attribution—robot should execute a particular task. In practice, we will realize both of these inferences as probabilistic maps. Rather than solely capturing the best way to look curious for a task, we'll seek to assign densities to trajectories, allowing the possibility that there are many equally likely alternatives.

In the remainder of the chapter, we aim to provide a general framework for modeling inferences of the form $\mathcal{I}_B$, and $\mathcal{I}_\xi$. Our goal is to enable robots to understand and account for the communicative effects of their motion on human observers.

5.3 Modeling and Influencing Attributions

We consider a scenario in which a mobile robot performs a coverage task in a two-dimensional discrete workspace while a human is observing from a top-down view. We employ a virtual environment[2] that resembles a house and stylize the agent as a robot vacuum cleaner (see Figure 5.2) since the general population is already somewhat familiar[3] with such robots [99], [121], making it easier for participants to develop mental models about their motion than that of a manipulator, for example.

In this scenario, the robot state space is the complete home workspace and Ξ is the space of all possible trajectories of any length that could be followed in the space. The robot action space $\mathcal{A}$ consists of deterministic movements in the cardinal directions. The cost of a state transition from a state s_t to a state s_{t+1} after having followed a trajectory ξ_t is defined as 0 if the state hasn't been visited before, -5 if the state contains a small, traversable obstacle (e.g. a vase), and -1 otherwise. A penalty of 3 times the number of unvisited states from the goal region is applied on termination.

5.3.1 *Understanding Behavioral Attribution for Coverage Tasks*

Through exploratory studies on Amazon Mechanical Turk, we sought to extract domain knowledge for attributions to robot motion within coverage tasks. Using the home layout shown in Figure 5.2, we generated a set of trajectories exhibiting qualitatively distinct ways the robot could respond to the prompt to "clean the bedroom," ranging from a near optimal coverage plan to a trajectory that

[2] The environment is built in the Phaser game engine (`https://phaser.io/`) and uses art by Bonsaiheldin under a CC-BY-SA license. It and other code relevant to this chapter can be found in the supplement of the original publication [108].

[3] A presentation by iRobot [121] estimates that 19M U.S. households had robot vacuum cleaners in 2020.

barely visited the target room (see Section A.1 for additional details). Each participant viewed videos of a random selection of three of these trajectories. After each video, participants were asked: a) to provide three words to describe the robot's behavior; b) to rate their agreement that "the robot is _____" for a range of adjectives drawn from relevant literature on human attributions [49], [122], [123]; c) to "explain what factors contributed to their strongest ratings." In addition to attributions, participants were asked to use an interactive interface to demonstrate how they would "clean the bedroom in a way that makes the robot look _____" where the blank was filled with a random adjective from the attribution rating items. Across all exploratory studies, we collected 375 sets of attribution ratings from 115 participants (73 male, 41 female) aged 21-70 (M = 38.3, SD = 10.7) covering 63 trajectories and a total of 193 demonstrations.

Extracting the Space of Attributions

To understand the inter-correlation of participant adjective ratings, we conducted an exploratory factor analysis. We selected a three-factor, promax rotation model that explained 74% of the observed variance due to its parsimony and coherence (see Section A.2). The first factor, which we call "competence" for its similarity to the relevant factor described by [123], consists of six items (responsible, competent, efficient, reliable, intelligent, focused) centered on the capability and diligence of the robot. The second consists of four items (lost, clumsy, confused, broken) alluding to a negative state, for which we title the factor "brokenness". The third contains two items (curious, investigative) and matches the curiosity factor examined by [49].

The extracted model enables the computation of standardized factor scores roughly in the range $[-3, 3]$ which denote how many standard deviations from the mean a participant's ratings for the items are. Reflecting the format of the component items, a high or low factor score denotes agreement or disagreement that a trajectory expresses an attribution, respectively. Based on this model, we represent the attribution for a trajectory ξ as a tuple $\boldsymbol{b} = (b_{competent}, b_{broken}, b_{curious}) \in \mathcal{B}$

Table 5.1: Trajectory features

Feature	Description
Coverage (%)	Goal region states visited at least once.
Redundant coverage (%)	Goal region states visited more than once.
Overlap (%)	Plan states visited more than once.
Length (%)	Normalized plan length.
Hook template (%)	Frequency of "U" shape patterns in plan.
Straight template (%)	Frequency of action repetition in plan.
Start-stop template (%)	Frequency of idle-move-idle patterns in plan.
Idleness (%)	Frequency of idle actions in plan.
Map coverage (%)	Fraction of map states visited at least once.
Collision (%)	Fraction of obstacle states from O in plan.
Goal deviation (%)	Fraction of plan before first goal state.

where the space of attributions is the set $\mathcal{B} = [-3, 3]^3$.

Low-dimensional Trajectory Representation

The space of possible trajectories in this domain is too large to map directly to the space of attributions, so we constructed a low-dimensional space Φ based on features relevant to the formation of attribution ratings. This allows us to describe a trajectory ξ as a vector $\phi_\xi = \phi(\xi) \in \Phi$. The feature space was inspired by relevant literature on human behavioral attribution to robot motion and enriched with features appearing in participants' explanations. The final set of 11 features used in further experiments is listed in Table 5.1.

5.3.2 Mapping Trajectories to Attribution Scores

Given a trajectory ξ, an observer's inference $\mathcal{I}_B$ of behavioral attribution can be expected to vary both due to individual differences and as a result of measurement error. For this reason, we

Figure 5.2: The home environment used in our exploratory studies.

model $\mathcal{I}_B$ as a conditional probability density $f_{\mathcal{B}|\Xi}(\boldsymbol{b}|\phi_\xi) : \mathcal{B} \rightarrow \mathbb{R}$. We observed multimodality in the distribution of factor scores for some trajectories, so we use a Mixture Density Network (MDN) [124] to approximate each conditional density as a mixture distribution $f_{\mathcal{B}|\Xi}(\boldsymbol{b}|\phi_\xi) = \sum_{i=1}^{C} \alpha_i(\phi_\xi) k_i(\boldsymbol{b}|\phi_\xi)$ where α_i, $i = 1, \ldots, C$, is a mixing coefficient, and k_i is a multivariate Gaussian kernel function with mean $\boldsymbol{\mu}_i$ and covariance Σ_i. Note that the mixing coefficients α_i and the Gaussian parameters $\boldsymbol{\mu}_i$ and Σ_i are functions of the featurized trajectory ϕ_ξ. In our models, these functions are implemented as linear transformations of features produced by a shared multi-layer perceptron.

To make efficient use of scarce data, we created ensembles of MDNs using bootstrap aggregation, i.e., we trained N models with different data splits and uniformly weight their predictions: $f_{\mathcal{B}|\Xi}^{ens}(\boldsymbol{b}|\phi_\xi) = \frac{1}{N} \sum_{i=1}^{N} f_{\mathcal{B}|\Xi}^{i}(\boldsymbol{b}|\phi_\xi)$.

We studied three different model configurations; single and four component MDNs, i.e., $C = 1$ and $C = 4$ and an ensemble of 8 MDNs each with four components, i.e., $C = 4$, $N = 8$. We trained

Table 5.2: Average test negative log likelihood (NLL) for each model configuration. Each datapoint represents a mean NLL over 16 models trained with random train-validate folds on a fixed test set. Error is the 95% confidence interval calculated with bootstrapping.

Model	Parameters	Average NLL	SD
Uniform	6	5.38	0.00
MDN, C=1	120	3.13 ± .05	1.35 ± .09
MDN, C=4	300	2.66 ± .08	1.57 ± .05
MDN Ensemble, C=4 N=8	2400	2.53 ± .06	1.38 ± .04

all models using an average negative log likelihood (NLL) loss function, the Adam optimizer [125], noise regularization [126], and early stopping. We configured the input MLP to use a single hidden layer with 5 units and a hyperbolic tangent activation. We expanded the dataset collected in our exploratory studies after assessing the sensitivity of the models to increased amounts of data, a process described in Section A.3. The final version of the set includes 126 trajectories with 671 attribution ratings. Table 5.2 compares the NLL of the models over held-out data. The mean indicates the typical quality of the prediction and the standard deviation indicates the degree to which this varied from sample to sample. Both quantities are averaged over 16 random folds and reported with bootstrapped 95% confidence intervals. All models perform significantly better than a uniform baseline, which simply assigns equal probability to all outcomes. The ensemble model performs best and is used in further experiments in the remainder of the chapter.

5.3.3 Generating Trajectories that Elicit Desired Attributions

We represent the behavior specification as a one-dimensional Gaussian $b^* \sim \mathcal{N}(\mu_b, \sigma_b^2)$ centered on a desired rating $\mu_b \in [-3, 3]$ for a single attribution dimension where the variance σ_b serves as a tolerance parameter modeling the acceptable distance from the desired behavioral rating. We use a density representation as it more closely matches the output of our model for $\mathcal{I}_B$.

Together with the task requirements as described by the cost function C, we realize the inference $\mathcal{I}_\xi$ as an optimization of the form:

$$\begin{aligned} \xi^* = \arg\min_{\xi \in \Xi} \quad & D_{\mathrm{KL}}(f_{\boldsymbol{B}_i} || \mathcal{N}(\mu_b, \sigma_b^2)) \\ \text{s.t.} \quad & C(\xi) \leq w, \end{aligned} \tag{5.1}$$

where D_{KL} denotes the KL divergence, $f_{\boldsymbol{B}_i}$ is the density $f_{\boldsymbol{B}|\Xi}|\phi(\xi)$ marginalized across dimensions other than i, and w is the maximum allowable task cost. Because many applications are conventionally exclusively task-cost driven, this format provides an intuitive "knob" in the form of how suboptimal the robot is allowed to be. Where performance is critical—perhaps to meet a schedule or to fit in power constraints—the robot designer need only set w to express the hard bound. Different attributions are expected to be more or less sensitive to the allowable suboptimality, something illustrated in Section A.4.

We implement this optimization using a hill-descending search in the space of trajectories. The search is initialized with a task-optimal trajectory generated via A* search and progressively samples modifications to the trajectory. These modifications consist of both naive, action-level modifications to the trajectory as well as changes targeting the activation of the features underlying the attribution model (see Table 5.1). Important motion templates, like runs of straight motion, hook patterns, and start-stops are sampled and patched into trajectories. All modifications are ranked by the divergence of their predicted attribution with the behavior specification. The search terminates after a fixed duration and the best performing trajectory subject to the task-cost constraint is returned. A detailed description of the optimization procedure is given in Section A.4.

5.4 Evaluation

We conduct a user study to evaluate the efficacy of the framework as a means of producing trajectories that elicit desired attributions. Our study is motivated by the following hypotheses:

Hypothesis 1 The model makes accurate predictions about the distribution of attributions to new trajectories.

Hypothesis 2 The model makes accurate predictions about the distribution of attributions to trajectories in unseen environments.

Hypothesis 3 The approach enables the generation of trajectories that elicit desired attributions.

5.4.1 Experiment Design

In Experiment I, participants observe and rate trajectories in the same home layout used for data collection, while in Experiment II, trajectories are generated in a modified home layout. In an effort to understand the impact of the environment geometry, the modified layout increases the size of the goal region by 100%, varies the placement of items and obstacles, and flips the dominant direction of the robot's motion. The experiments are within-subjects, video-based user studies, both instantiated in three parallel sets corresponding to the three attribution dimensions considered. For each dimension, we consider four robot trajectories generated by optimizing (5.1) with a target distribution expressed as a Gaussian centered at 1.5 with scale 0.3 and varying task cost thresholds. To ease interpretation, the w values governing the thresholds were set in multiples—**1x**, **2x**, **4x**, **12x**—of the cost of the optimal trajectory for the task. The full set of trajectories is shown in Figure 5.3[4]. In all experiments, participants rate and describe each trajectory using the same items and questions used in the exploratory studies of Section 5.3. After watching all trajectories in a randomly assigned order, they also respond to additional comparative questions: "which robot seemed the most _____" and "which robot seemed the least _____", where the blanks are filled with the adjective corresponding to the dimension of attribution studied. Both comparisons are accompanied with an open-ended question asking for a brief explanation of the choice. No suitable baselines exist for the balanced attribution elicitation task, so our experiments

[4]Videos of the trajectories are included in the supplement.

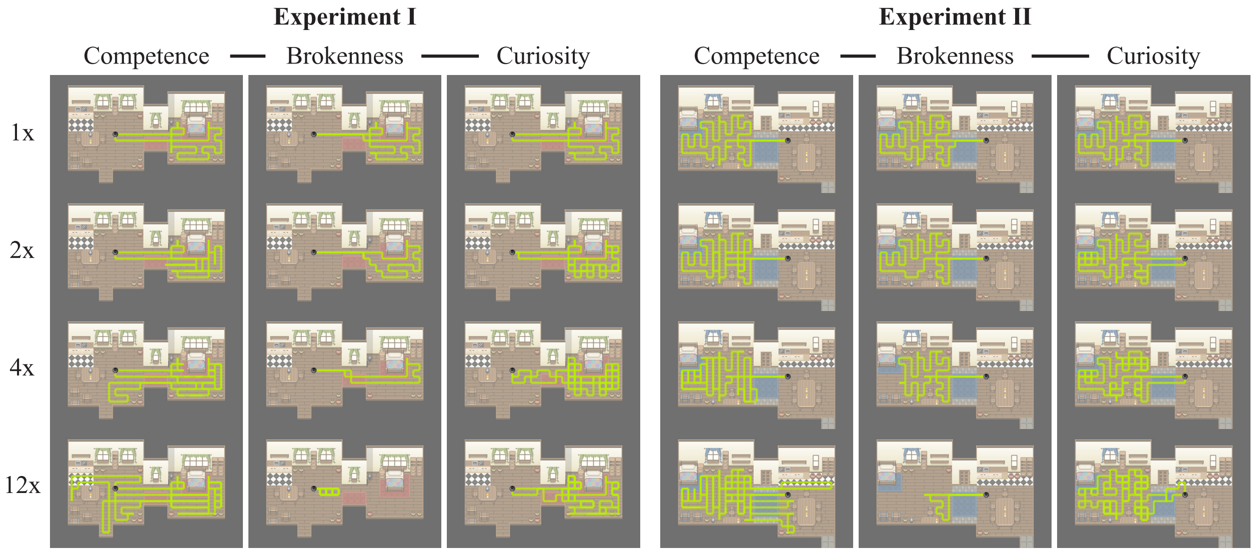


Figure 5.3: Traces of robot trajectories used in different experiments and conditions.

use solely trajectories generated by our approach.

Participants A total of 144 participants (76 male, 68 female) aged 20-72 (M = 38.2, SD = 10.9) were recruited via Amazon Mechanical Turk and paid $2 to complete the approximately 15 minute task. 9 had taken part in our earlier exploratory studies. Participants were equally distributed amongst the six sets of conditions. Condition orderings were fully counterbalanced.

5.4.2 Results

The predictive performance of the model is illustrated in Figure 5.5 and reported in Figure 5.3. The 95% bootstrapped confidence interval on the mean NLL for Experiment I was 2.77±.10 (SD = 0.88±.09). and 2.87±0.10 (SD = 0.86±.09) for Experiment II. Participants' choices for "most" and "least" trajectories are shown in Figure 5.4.

Hypothesis 1 was supported; the average NLL of the models was significantly lower than a uniform model, indicating that the model was able to meaningfully predict attributions in the layout it was trained in.

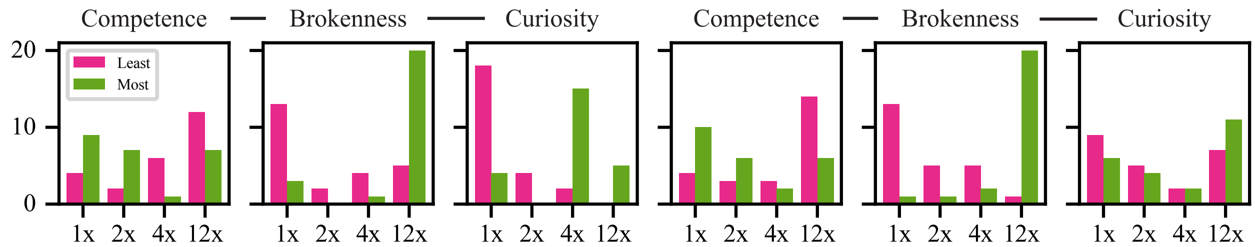


Figure 5.4: Counts of trajectories picked as most and least "______".

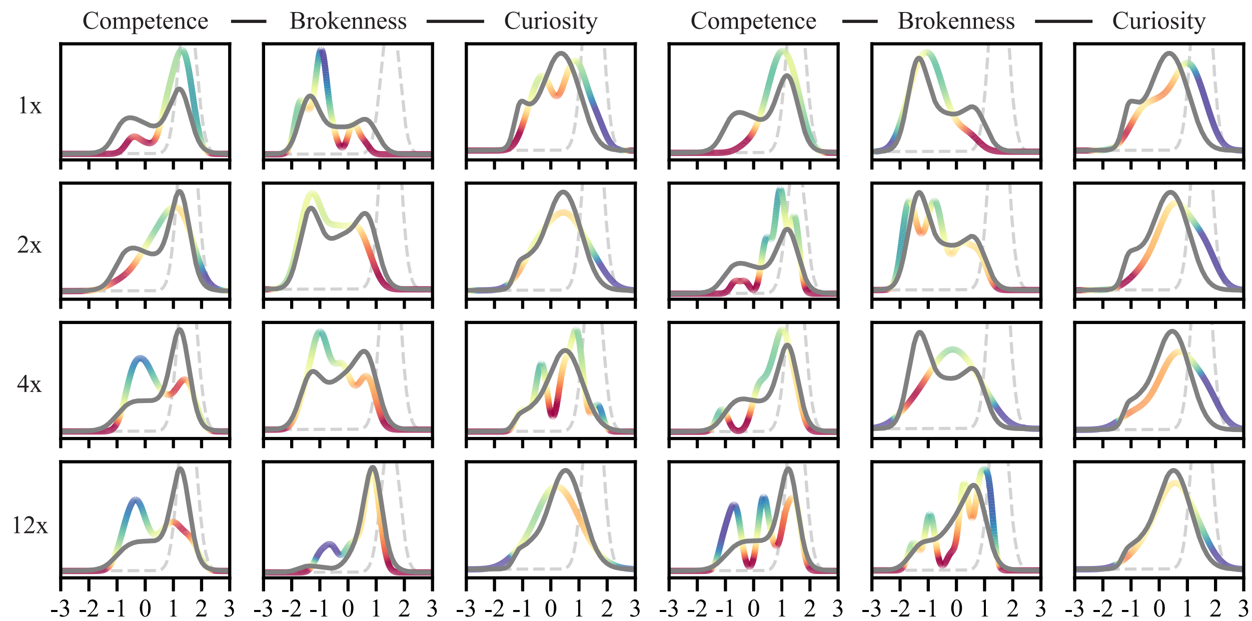


Figure 5.5: Comparisons of the model's predicted density for the factor score under consideration (in grey), the observed distribution (multicolored) approximated with Gaussian kernel density estimation and the target distribution (dashed). Each subplot has a unique y-scale, with the magnitude of the difference between the predicted and observed densities at any point encoded instead in the color of the line for the observed distribution. Deep red indicates the model severely overpredicted the density, while deep blue indicates severe underprediction.

Table 5.3: Evaluation average NLL (SD)

	Competence	Brokenness	Curiosity	Competence	Brokenness	Curiosity
1x	2.51 (0.69)	2.44 (0.76)	2.37 (0.54)	2.73 (0.59)	2.64 (0.73)	2.76 (1.05)
2x	2.76 (0.85)	2.84 (0.72)	2.85 (0.75)	2.95 (0.70)	2.86 (0.88)	2.85 (1.09)
4x	2.98 (0.76)	2.93 (0.83)	3.02 (0.71)	3.00 (0.67)	2.92 (0.42)	3.10 (0.98)
12x	3.10 (0.73)	2.53 (1.75)	2.94 (0.72)	3.43 (0.84)	2.28 (0.98)	2.84 (0.84)

Hypothesis 2 was supported; the average NLL of the attributions observed across Experiment II was significantly lower than the uniform model, indicating that the model remains informative even under modifications to the environment layout.[5]

Hypothesis 3 saw mixed support; Kendall's tau-b correlation tests (reported in Section A.5) indicate strong positive correlations between the allowable suboptimality and brokenness factor scores, suggesting that the trajectory generation method was effective at eliciting progressively higher factor scores. However, tests for the competence conditions indicated a moderate negative correlation, and tests for curiosity conditions were not significant. As shown in Figure 5.4, while participants found that 12x and 1x were the most and least "_____" for experiments focused on curiosity and brokenness, this relationship was flipped for the competence experiments.

When optimizing for competence, the model emphasizes over-coverage of the goal region as well as coverage of the house as a whole (see Figure 5.3). Due to the associated task-cost penalty, coverage outside of the goal begins to appear in the 4x condition of Experiment I and the 12x condition of Experiment II, and participants' responses indicate that it is a key driver of negative attributions of competence. Some emphasized that time spent not cleaning the bedroom was "wasted movement in the wrong room" (Exp. I-Competence), while others attributed the deviation to being "lost" or "totally confused" (Exp. I-Competence). In less extreme conditions, users expressed uncertainty about what drove the behavior, saying they were "not sure if robot is cleaning outside the bedroom cause there might be dirt that can be brought in, or just confused as to parameter of bedroom" (Exp. I-Competence, 4x). A minority of users thought that the extra motion was worthwhile, marking the 12x trajectory as the most competent because it "cleaned more areas in both rooms" or "completed the entire home from bedroom to kitchen" (Exp. I-Competence).

[5]See AppendixSection A.5 for supplementary tests indicating insufficient evidence to support a significant difference between the predicted and observed distributions in both experiments.

The model overestimated the prevalence of people who would appreciate the additional coverage of the environment, as indicated by both the "most/least" selections and the NLL. We speculate that the format of the experiment—wherein participants view the optimal trajectory and mentally anchor their ratings against it—is a contributing factor; participants may be more inclined to rate the robot as competent when viewing the optimized trajectories in real-world settings where direct comparison to the task-optimal trajectory is less likely.

Trajectories optimized to look broken progressively cover less of the goal region before ultimately devolving into repeated circular motion near the start point (see Figure 5.3). The majority of users concurred in their assessment of the 12x condition as "defective", with some remarking that it seemed overwrought, "a joke version of the robot" (Exp. II-Brokenness). The model underpredicted the extent of participants' agreement that the optimal trajectory would be perceived as not broken, but the predictive performance across all brokenness conditions was still the strongest of the three factors.

When optimizing for curiosity, the model emphasized over-coverage of the goal region, overlapping motion, hook-like patterns and visiting penalized states depicted with vases (see Figure 5.3). Some participants highlight the extra coverage as the reason for selecting the 12x as the most curious condition, saying the robot "cleans very well but cleaned the same place multiple time, roaming without reason" or that it "dawdled around a lot, getting hyper fixated on certain spots" (Exp. II-Curiosity). The same factor is highlighted by participants in less extreme conditions, with one speculating of the 4x trajectory that "maybe something caught its eye while it was working and it got so distracted that he totally kept getting off track" (Exp. I-Curiosity). The change in the distribution of curiosity factor scores was expected to be small and the observations bear the predictions out, though it is notable that despite the subtle differences a majority of participants select either 4x or 12x as the "most curious" trajectory across both experiments.

5.5 Discussion

Our framework is a general approach for endowing robots with a sensitivity to the behavioral attributions their motion elicits. We illustrated its application to coverage motions, but we envision a lively stable of related instances stretching across domains from delivery service robots to robot arms in fulfillment centers. The process is the same; the robot arm's designer will build a pool of videos and study users' responses to—and their reasoning about—the motion over a broad set of dimensions, then learn a forward mapping from the features driving their reasoning to the attributions in their responses. While some features such as path length or redundancy may map over from the coverage domain, others like the shape of the acceleration profile may need to be added to capture perceptions of danger or erraticism. The reverse translation from a desired attribution to a new trajectory can be realized by searching in the space of trajectories, using the attribution features to guide the process—something that may have a pronounced impact in a higher dimensional planning space.

We haven't yet addressed some important aspects of behavioral attribution. While our results showed that the approach's performance transferred to a similar environment, future work should use more disparate environments to determine the limits of its generalization. In the setting we explored, the observer looks at the robot's motion from a top-down perspective, but different perspectives may result in different impressions. Further research should evaluate data collection techniques and environments that account for variability in observer perspectives. The features and the learned mappings from features to attributions are specific to the types of environments and the task considered and would likely need to be augmented to work more broadly. We imagine that, in the future, robot designers will have access to a wide array of well-studied features with which to bootstrap their system, and when human-robot interaction data is abundant, we may see the rise of learned representations that can power the understanding and generation of motion with minimal additional supervision.

6 Intuitive Models for Channel Management: Pointing-based Assistive Teleoperation

This chapter addresses transparency in shared control systems, where the challenge lies in communicating the state of assistive automation to the operator. While previous chapters examined how to make autonomous robot behaviors more interpretable, here we address how operators perceive and interact with assistive functions. Given limited channels, the operator needs to understand what assistance is available and how to influence it without disrupting their primary task.

Telemanipulation exemplifies these transparency challenges. It is widely useful but remains demanding even for skilled operators using present interfaces. Acting through a foreign embodiment with limited perception requires users to simultaneously reason about the task, the robot's capabilities, and the environment state. Assistive teleoperation interfaces can reduce this cognitive burden by automating parts of the robot's behavior, enhancing safety and comfort for applications ranging from industrial assembly to assistive robotics for home use.

The importance of transparent interfaces extends beyond direct operation. Teleoperation is increasingly used to collect human demonstration data from both simulated [127]–[130] and real robots [131]–[133] for imitation learning [127], [134] and offline reinforcement learning [135], [136]. Interface improvements are needed to make teleoperation more intuitive and to improve the quality of collected trajectories [137], [138]. Despite these advances, fundamental challenges remain in precise manipulation: operators struggle with depth perception and haptic feedback limitations [139], [140], leading to grasp failures in tight clearances and excessive force when

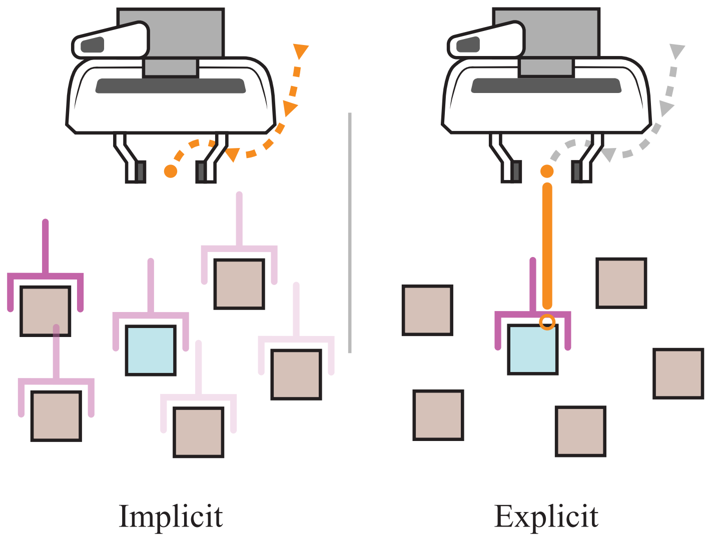


Figure 6.1: Left: Implicit assistance funnels the operator toward the goal predicted based on (for instance) the recent trajectory. The operator is not intended to change their input to influence the assistance. **Right:** Explicit assistance affords the operator direct control over the inferred goal by pointing the gripper toward the object of interest. A local optimization selects a feasible, collision free pose.

placing objects.

The foundation of most assistive teleoperation systems is prediction [141]–[143]. Inferring, for instance, the operator's desired trajectory or end-effector goal based on their recent trajectory and context (i.e., scene, object, task) enables the automation of subsequent actions. Performant prediction systems can engage assistance fluently in proportion to their own confidence. The user teleoperates as they would without assistance. Their control over the predictions is *implicit*, arising from how their state or actions correspond with some model of possible intended behavior.

But the benefits of implicit inference-based assistance are difficult to realize in practice. Human environments pose challenges for online trajectory and goal prediction [144], [145]. In clutter, where there are numerous possible target objects in close physical proximity, it is inherently difficult to predict manipulation targets as many goals may be consistent with a user's state or historical input. Poor predictions can lead the operator to modify their behavior in an attempt to better signal their goal—a confusing interaction, as the operator's mental model of the predictor is likely incorrect. In such situations, it is preferable to provide an *explicit* interface that accommodates the user's desire to exert direct control over the predicted intent. Explicit input interfaces usually involve modal goal-specification interactions which aren't suitable for online interaction, or additional input modalities, like natural language, that introduce complexity and potentially increase burden.

In this chapter, we propose an interface for pick and place manipulation, shown in Figure 6.1, that leverages "pointing" of the end effector as an explicit input method, requiring neither an additional input modality nor a modal interaction[1]. The approach offers assistance for a possible grasp or placement pose via optimization in a region around where a ray from the gripper to the target object (grasping) or from the object in the gripper (placing) meets the scene geometry. The size of the region provides a simple, configurable trade-off between smooth behavior and effectiveness. When small, top-ranking candidates stay near where the user pointed. When large, the top-ranking candidates may jump further to better performing poses. Parallel computation allows the system to rank and filter many possible collision-free candidates and present suggestions that are responsive to the user's input at high frequency.

We implemented our proposed explicit interface on a simulated Franka Emika Panda robot and conducted a user study comparing it to an implicit-input assistive teleoperation method on pick-and-place stacking tasks with clutter. We find that operators prefer the explicit interface, experience fewer pick failures and report lower cognitive workload. Our implementation of explicit assistance and study conditions in NVIDIA Omniverse Isaac Sim is available at `github.com/NVlabs/fast-explicit-teleop`.

6.1 Related Work

The design space for assistive teleoperation spans various types of operator input, different forms of assistance and a spectrum of manual to automatic engagement [141].

[1]This chapter consists of materials published in [146] IROS 2024 **Walker**, Yang, Garg, Cakmak, Fox, Pérez-D'Arpino.

Most assistive teleoperation methods use a form of implicit input to autonomously generate improved robot actions. Early methods maintained a probability distribution over possible goals given users' recent actions and overrode user control with actions more in line with optimal trajectories to the inferred goal [141]–[143]. When available, data enables the use of sophisticated predictive models like trajectory forecasting transformers [147] or multi-modal diffusion policies [148]. When a task reward is available, it is possible to use human-in-the-loop deep reinforcement learning [149]. While some of these methods produce assistance based only on the current state, user interactions with the assistance are characteristically implicit as the user is not intended to control the state with the aim of modifying the assistance.

Human-in-the-loop autonomous systems commonly allow operators to explicitly specify goals, preview generated trajectories and supervise execution [150], [151]. Most frequently, these interfaces use keyboard and mouse control over 6DOF interactive pose markers, enabling precise goal specification at the expense of fluency, making them unsuitable online continuous teleoperation.

Assistance can also come in the form of augmented control input schemes. [152] used demonstrations to learn a task-specific low-dimensional control mapping, enabling operators to control a robot arm using only a 2D joystick. [153] showed that such task-specific mappings can also be generated conditionally based on a language description of a task in a way that also allows natural language corrections during execution.

Another approach is to dynamically constrain actions to, for example, avoid collisions with obstacles [154], or reject probable inadvertent input in a fine manipulation task. [155] introduced the concept of "virtual fixtures," registered geometric overlays, typically specified beforehand using task knowledge, which produce sensory cues or alter control behavior as operators move through them. These fixtures restrict motion within a region, like a virtual ruler confining end-effector motion to a line.

6.2 Fast Explicit-Input Assistance

We are interested in generating actions to assist a teleoperator. Abstractly, the generation of these *actions*—which may be poses, configurations or trajectories—is the result of an optimization based on *state* information and *context*:

$$\text{actions} = \underset{\text{option}\in\mathcal{A}}{\arg\max}\, f(\text{option}, \text{state}, \text{context}) \tag{6.1}$$

The defining decisions we make about the implementation of Equation 6.1 that result in an effective explicit-input interaction are:

- to use transparent and controllable state information;
- to prioritize smoothness of the assistance with respect to state in the selection of f. Both the average and maximum variations in assistance for small state changes affect usability, as abrupt changes can be disorienting;
- and to treat the resulting action as a suggestion subject to user review and refinement.

Conventional inference-based assistance systems attempt to represent the space of possible goal poses or next-actions in $\mathcal{A}$. They select for f a model of the likelihood of the goal conditioned on the pose or recent trajectory of the robot.

We similarly choose to produce useful poses for the operator, but we disregard the opaque history of the operator's actions and instead rely on immediately controllable present-state information. We leverage an intuitive "pointing" metaphor to allow the user to specify the anchor for a local optimization of an assistance pose. We define the optimization to be amenable to parallelization, ensuring it can compute at interactive speeds. The result is a pose suggestion that the user can ignore or modify by pointing the gripper before affirmatively accepting.

6.2.1 Pointing as Ray Control

Our experience is that the most understandable and controllable aspect of state is where the end-effector is pointing. Although pointing is governed by all six degrees of freedom (DoF) of the end effector pose—which we denote as $\boldsymbol{e}_e \in \mathrm{SE}(3)$ located between the fingers—it particularly emphasizes control of the axis component $\boldsymbol{v} \in \mathbb{R}^3$ of the axis-angle $(\boldsymbol{v}, \theta)$ representation of the SO(3) orientation. For convenience, we will also leverage the $\mathbb{R}^{4\times4}$ transformation matrix representation of the pose $\boldsymbol{e}_e$ consisting of rotation matrix $\boldsymbol{R} = [\boldsymbol{r}_x, \boldsymbol{r}_y, \boldsymbol{r}_z] \in \mathbb{R}^{3\times3}$ and translation component $\boldsymbol{p}_e \in \mathbb{R}^3$. We assign the $\boldsymbol{r}_z$ component outwards from the gripper, $\boldsymbol{r}_y$ perpendicular and along the axis of closing, and $\boldsymbol{r}_x$ perpendicular and away from the gripper camera. These axes are labeled on Figure 6.2.

Pointing the axis $\boldsymbol{r}_z$ is familiar for operators not only because it is a necessary component of most manipulation tasks, but also because it is a means to change the view of the "eye-in-hand" camera that is often available. When unobstructed, this view is an innate visualization of the pointing input upon which crosshairs or a rendered lines can directly show the pointing axis.

The pointing target is the point $\boldsymbol{p}_t \in \mathbb{R}^3$ at which the ray extending from the end effector position $\boldsymbol{p}_e$ along $\boldsymbol{r}_z$ contacts the scene geometry, and $\boldsymbol{n}_t$ is the surface normal at the target point. These quantities can be approximated using depth data or a geometric representation that the robot has access to. They are simple to visualize by (for example) highlighting the point in a 2D view and drawing a tick mark in the normal direction.

Previous works have focused on the proximity of the end effector to assistance candidates, but relative distinctions in distance are difficult to assess based on a remote 2D view. Proximity is chiefly a function of the 3DoF end effector position $\boldsymbol{p}_e$, whereas the axis $\boldsymbol{v}$ of the gripper orientation is characterized by just the 2 spherical coordinates, azimuth and elevation. Pointing does still usefully encode a nearness bias, since the area at which one can point at a given object is inversely

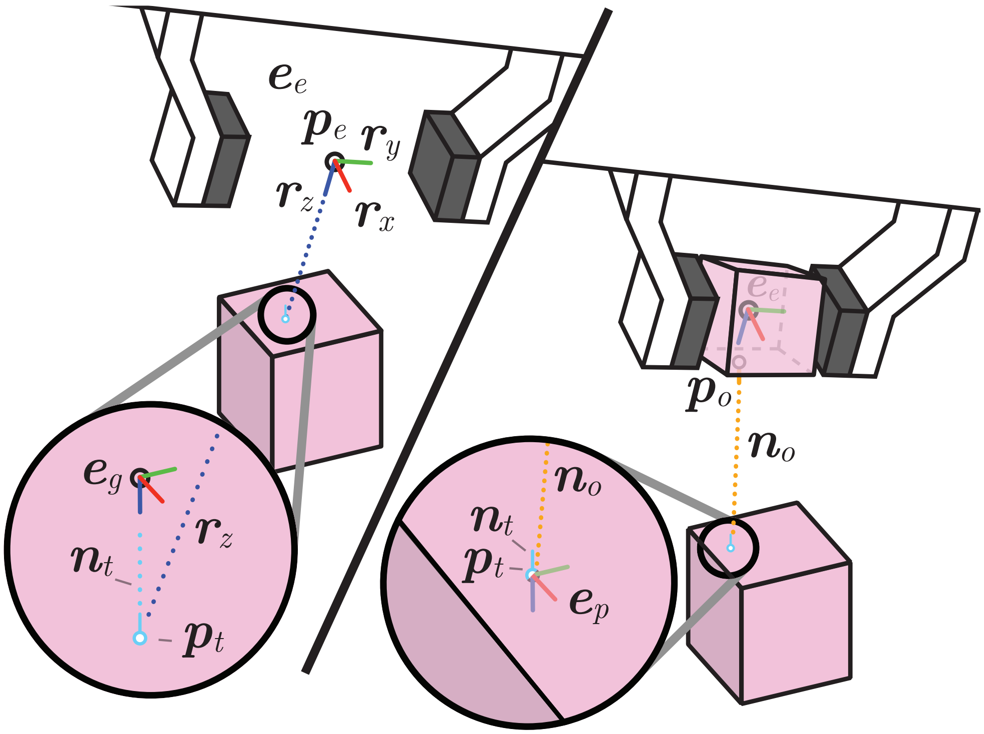


Figure 6.2: Our realizations of explicit grasping (left) and placing assistance (right) both center on the interaction of a ray from the gripper with scene geometry. A projected anchor pose is calculated then used to select amongst a set of candidate assistance poses.

proportional to the square of the distance to the object. In other words, it is easy to point at things nearby, grows more difficult for things further away, and quickly becomes effectively impossible beyond a point. This bias is reinforced by the fact that one can only point at what can be seen and further objects are subject to greater occlusion.

6.2.2 Grasp Pointing

In order to suggest a possible grasp pose to the operator using our pointing interface we must define a mapping from the 6D pointing control to a 6D grasp pose. We denote the final grasp assistance suggestion as $\boldsymbol{e}_g^*$.

A direct mapping would be to simply displace the current gripper orientation along the ray to some fixed offset of the target point, making no modification to the gripper orientation. However, our experience is that users often point at oblique angles but nonetheless desire an approach orthogonal to the object surface. Instead of using $\boldsymbol{r}_z$, we use the negation of the surface normal $\boldsymbol{n}_t$ at the target point $\boldsymbol{p}_t$.

Users generally expect the angle θ about the axis to be the one that is "most similar" to their current orientation. To encode this geometrically, we project a reference vector anchored to the gripper onto the plane defined by the intersection point $\boldsymbol{p}_t$ and the normal vector $\boldsymbol{n}_t$. Any reference vector may be selected, however it is preferable to use one that is unlikely to be perpendicular to the plane, like $\boldsymbol{r}_x$ or $\boldsymbol{r}_y$. The minimal rotation is the geodesic between the current and the projected reference vector.

The resulting grasp anchor pose $\boldsymbol{e}_g$ provides an intuitive, cursor-like interaction when the gripper ray is swept across the scene. It is unlikely to be a satisfactory grasp on its own, however, because an orthogonal approach may be inappropriate for the object, or the position may cause contact with the object or other scene geometry. A generative grasp model can be used to provide a set $\mathcal{A}_{(\boldsymbol{e}_g)}$ of candidates near the anchor. The specification of "near" governs the smoothness of the assistance interaction, with smaller thresholds ensuring that the resulting poses do not change substantially as the cursor moves but necessarily excluding more suitable grasps that are too far away. Each candidate can be computed and scored independently, making this step highly parallelizable. The result nearest the anchor should be taken as grasping suggestion $\boldsymbol{e}_g^*$. Generally the quality and smoothness of the assistance improve as more candidates are considered so long as the computation runs at interactive rates.

6.2.3 Placement Pointing

As with grasp pointing, we seek a mapping from the 6D pointing control to a 6D end-effector placement pose, $\boldsymbol{e}_p^*$.

The object may have been grasped in an arbitrary orientation, so a direct mapping that translates the current pose along the gripper axis $\boldsymbol{r}_z$ toward the target point is unlikely to be useful for stably placing the object. Instead, we observe that the object was likely picked from a stable pose where it rested on a support facet defined by some point $\boldsymbol{p}_o$ and normal $\boldsymbol{n}_o$ pointing in the gravity direction. At the moment of the pick, the orientation of normal $\boldsymbol{n}_o$ can be recorded with respect to the end effector pose $\boldsymbol{e}_e$, and a point $\boldsymbol{p}_o$ on the object facet can be estimated by projecting the end-effector position $\boldsymbol{p}_e$ at the moment of the pick onto the scene geometry revealed after the object is lifted.

It is now intuitive to map the control of the resulting plane $(\boldsymbol{p}_o, \boldsymbol{n}_o)$; the user principally controls the axis $\boldsymbol{n}_o$ to select a pose constrained to place the facet point $\boldsymbol{p}_o$ at the target point $\boldsymbol{p}_t$ and to align the object normal $\boldsymbol{n}_o$ opposite the target normal $\boldsymbol{n}_t$. Similar to the grasp mapping, the undetermined rotation of the object about the target normal is specified by finding the geodesic from a reference vector on the end effector (like $\boldsymbol{r}_x$ or $\boldsymbol{r}_y$) to the same vector projected onto the target plane.

The resulting placement anchor pose $\boldsymbol{e}_p$ is a direct, cursor-like projection of the grasped object into a placement, and is used in a similar manner as the grasp anchor pose. The anchor itself may not be a feasible placement pose if it puts the object or the gripper into contact with the scene. Candidates $\mathcal{A}_{(\boldsymbol{e}_p)}$ can be generated in the local region around the anchor using any generative object placement method, with the candidate nearest the placement anchor pose serving as the suggestion $\boldsymbol{e}_p^*$ to the user.

6.2.4 Snapping

As a consequence of prioritizing responsiveness, the range of inputs which our methods map to any particular assistance anchor pose $\boldsymbol{e}_g$ or $\boldsymbol{e}_p$ is small. Certain "easy" poses like a perfectly aligned side-grasp might be frustratingly difficult to specify. We use *snapping* to nudge the generated assistance toward these preferred poses, providing the flexibility to control the grasp suggestion

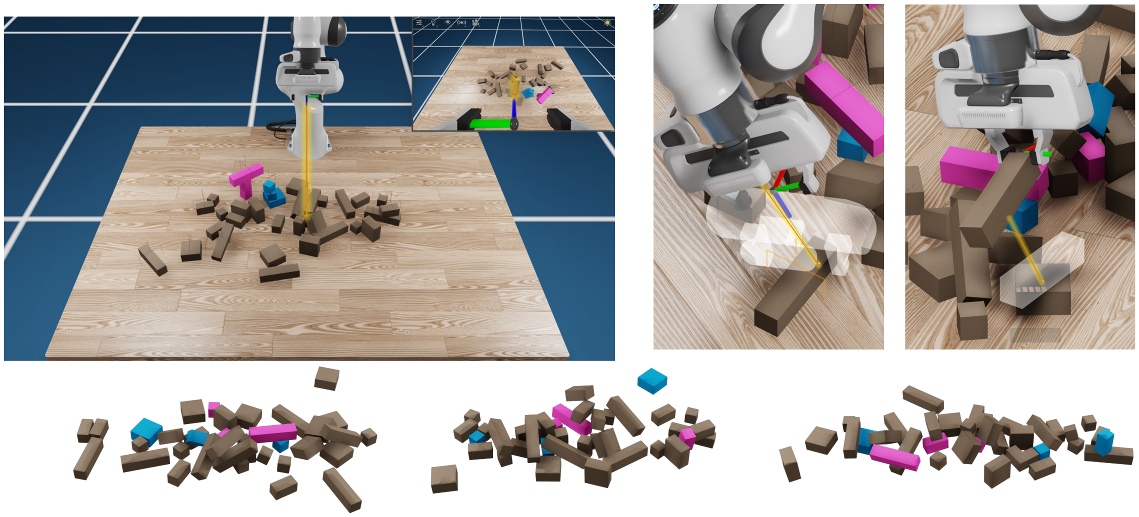


Figure 6.3: Top left: The operator controls the robot while looking at two camera views displayed picture-in-picture. **Top right:** Assistance suggestions are shown as a "ghost gripper" for grasping and a "ghost shape" for placing actions. Ray visualizations are exaggerated for legibility in print. **Bottom:** The experimental task involved participants extracting and stacking blue and pink blocks that were initially scattered in one of three clutter configurations.

(as is typically needed in cluttered scenes) or to easily snap into commonly used grasps when feasible. The behavior of snapping is demonstrated in the accompanying video.

Snaps are encoded by one or more potential fields $\phi(\cdot)$ over poses. After anchor poses $\boldsymbol{e}_g$ or $\boldsymbol{e}_p$ are calculated, a local optimization over ϕ occurs, checking to see if there is a lower potential pose within an ϵ distance threshold that would breach potential threshold γ. If so, candidates from $\mathcal{A}_{(\boldsymbol{e}_g)}$ or $\mathcal{A}_{(\boldsymbol{e}_p)}$ are ignored and the snap pose is provided as the suggestion.

In practice, we find that specifying a set of poses that align with object centroids coupled with proximity potential $\phi(e^*) = \min_{G_i \in G} d(\boldsymbol{e}^*, G_i)$ is useful for picking and placing and requires no additional task context.

Following [156], we define the distance between the poses $\boldsymbol{x}, \boldsymbol{y} \in \mathrm{SE}(3)$ with position components $\boldsymbol{p}_x, \boldsymbol{p}_y \in \mathbb{R}^3$ and rotational components $\boldsymbol{R}_x, \boldsymbol{R}_y \in \mathbb{R}^{3\times3}$, as

$$d(\boldsymbol{x}, \boldsymbol{y})^2 = ||\boldsymbol{p}_x - \boldsymbol{p}_y||_2 + 2\beta^2(1 - \frac{tr(\boldsymbol{R}_y^{-1}\boldsymbol{R}_x)}{3})), \tag{6.2}$$

where β weights the translation and rotation contributions to the distance.

6.3 Experiment

We conducted a within-subjects user study where participants completed stacking tasks without assistance (**CON**), with implicit inference-based assistance suggestions (**IMP**), and with explicit-input assistance suggestions (**EXP**).

Participants completed a multi-step singulation and stacking task where they created multiple stacks of particularly-colored blocks from a cluttered pile. The task was designed to have few prescribed steps and many possible intermediate goals.

We expected that participants would:

H1 : be most effective at completing the task using EXP.

H2 : make most use of suggestions provided by EXP

H3 : report the lowest workload when using EXP.

H4 : feel that the suggestions from EXP better match their preferences.

H5 : feel that they understand the behavior EXP better than that of IMP.

6.3.1 System

Participants interact with a Franka Panda robot simulated in NVIDIA Omniverse Isaac Sim. Grasp sampling and collision checking operations are GPU accelerated using NVIDIA Warp [157].

Input Users provide input using a 6DOF mouse, a spring-suspended puck that they can displace in three spatial dimensions while simultaneously panning, tilting, or twisting to provide 3D rotation [158].

Robot Control User input is interpreted as a twist goal for the robot's end-effector. We integrate the twist over a fixed timestep, apply the resulting transformation to the current end-effector pose, and provide the result as a pose goal to the robot controller, a Riemannian Motion Policy implemented in RMPFlow [159]. To avoid large accelerations, the pose goal is passed through a low pass filter.

Camera Control Users operate the robot while monitoring a fixed view, showing most of the robot and the workspace, and a dynamic view affixed to the gripper pointing towards the fingers. As shown in Figure 6.3, one view is foregrounded at a time, and user input is interpreted in the frame of the foregrounded view.

Assistance Offers of assistance are visualized as "ghosts," as shown in Figure 6.3. Holding a button on the 3D mouse engages the assistance, forwarding the suggested pose as a goal for the controller with an additional preprocessing step to ensure poses are approached from the front.

Explicit Assistance Condition (EXP) We implement our grasp pointing assistance approach using a simple approach-vector parameterized sampling scheme, looking for the nearest non-colliding pose amongst 7125 translated and rotated candidates around the grasp anchor pose. The samples are distributed in a fixed 2cm diameter, 1cm thick disc pattern. We did not use a placement sampler as we assessed that direct control over the placement anchor pose was sufficient for the experimental task. Raycasting is performed against a mesh representation of the scene. We generate axis aligned grasp and placement poses and use them to define a snapping potential as described in Section 6.2.4.

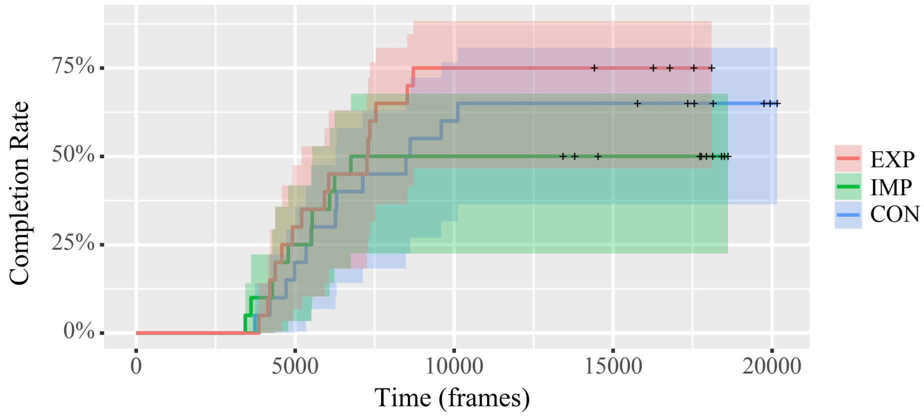


Figure 6.4: Survival analysis (↑) of participant's completion of the task over time. Lines plot percentage of participants that completed the task at the time and Xs mark termination without completion. Differences lie within the 95% confidence interval, with a trend that the probability of having completed the task grows most quickly for the explicit input interface and reaches a higher peak.

Implicit Inference-Based Assistance Condition (IMP) Following [141], we attempt to infer the user's goal by selecting the most-probable goal G^* from a predefined set of candidates $\mathcal{G}$ based on a recent window of the robot's trajectory $\xi_{S\to U}$ from start pose S to current pose U:

$$G^* = \arg\max_{G\in\mathcal{G}} \left(\frac{e^{-C_G(\xi_{S\to U}) - C_G(\xi^*_{U\to G})}}{e^{-C_G(\xi^*_{S\to G})}} \cdot e^{-d(U,G)} \right) \tag{6.3}$$

The first term assigns greater likelihood to goals for which the user's trajectory, completed optimally by $\xi^*_{U\to G}$, has cost similar to the cost of the optimal trajectory $\xi^*_{S\to G}$. The second term serves as a prior, assigning more mass to goals that are closer to current pose. We use $C_G(\xi_{X\to Y}) = d(X,Y)^2$ and reset S if 2 seconds pass with no control input. The same set of axis-aligned grasp and placement poses used for snaps are used as $\mathcal{G}$, and collision checking is performed across this set to ensure no in-collision poses are offered.

6.3.2 Procedure

Participants were told they would use a 3D mouse to control a robot with three different systems, some of which would provide suggestions they could use to help them complete tasks. Each

Table 6.1: Comparison of condition preference Counts ↑

	A	B	C_A	C_B	$\frac{C_A}{C_A+C_B}$% (CI)	p
EQ1	EXP	CON	11	1	92 (62, 100)	**.010**
	"	IMP	"	8	60 (34, 80)	.648
	CON	"	1	"	11 (0, 48)	.078
EQ2	EXP	CON	10	3	79 (49, 95)	.277
	"	IMP	"	7	61 (33, 82)	.688
	CON	"	3	"	30 (7, 65)	.688
EQ3	EXP	CON	14	1	94 (68, 100)	**.003**
	"	IMP	"	5	75 (49, 91)	.127
	CON	"	1	"	17 (0, 64)	.219
EQ4	EXP	CON	14	2	88 (62, 98)	**.013**
	"	IMP	"	4	79 (52, 94)	.061
	CON	"	1	"	33 (4, 78)	.687

session began with an interactive 3D mouse tutorial, followed by a robot control tutorial where they had to grasp and lift a block, and finally an assistance tutorial which demonstrated what suggestions of assistance would look like and how to use them.

For each condition, participants were given a brief verbal introduction to how the system would behave and asked to “warm up” by stacking a block. Once satisfied that they understood the system, participants completed a single stack task for 3 minutes, then had a maximum of 7 minutes to complete the multi-step stacking task. A post-interaction survey included the NASA-TLX questionnaire [160], three agreement questions regarding their sense of control over the suggestions (reported as assistance composite) and one regarding their sense of understanding. Rating questions were represented using 7–point scales.

A final set of forced-choice questions probed which system “felt easiest to use" (EQ1), and which system had the suggestions that “made it easiest to do the task the way [they] wanted to" (EQ2) which they best understood “why [the suggestions] behaved the way they did" (EQ3), and “felt most in control of" (EQ4). Finally, participants completed demographic questions and rated their familiarity with robots, operating robot arms, 3D mice, and playing video games. Sessions lasted between 45-60 minutes total.

Participants We recruited 20 participants (18 male, 2 female, aged 19-39 M=25.1, SD=5.45) from the University of Washington under an IRB approved study plan. Many participants were roboticists, rating their familiarity with robots highly (M=4.80, SD=2.08, 7−point scale). Only two participants reported being familiar with 3D-mice (rating >4 on 7−point scale). All participants were right handed.

6.3.3 Methods

We analyze logged events, survey data and supplemental annotations using generalized linear mixed models to account for inter- and intra-participant variance. The effect of an experimental condition is given as either a ratio or difference of the estimated marginal mean against another contrasting condition, and significance is determined using 95% confidence intervals. We conducted survival analysis to characterize task completion rates over time. Statistical details are reported in the supplementary materials.

6.3.4 Results

H1: Participants experienced significantly fewer failed picks in EXP when compared to IMP or CON, and there was a trend indicating that they experienced fewer place failures as well, as shown in Table 6.4. There were trends indicating that users of the explicit interface complete the task with higher frequency and stack objects more quickly, as shown in Figure 6.4, however the differences are not statistically significant.

H2: There was no measurable difference in the duration or number of engagements of the assistance between the implicit and explicit interfaces. Qualitatively, we observed that some

Table 6.2: NASA-TLX scores ↓

A	B	$M_A(SE_A)$	$M_B(SE_B)$	$M_A - M_B(CI)$	p
EXP	IMP	3.42 (.25)	3.77 (.25)	-.36 (-.97, .25)	.335
"	CON	"	4.33 (.25)	-.92 (-1.53, -.31)	**.002**
IMP	"	3.77 (.25)	"	-.56 (-1.17, .05)	.079

Table 6.3: Assistance subjective scores ↑

	EXP		IMP			
Question	M_A	SE_A	M_B	SE_B	$M_A - M_B(CI)$	p
Composite	4.70	.253	3.44	.253	1.25 (.52, 1.99)	**.002**
Understanding	4.62	.274	4.29	.274	.33 (-.47, 1.14)	.398

participants made use of the explicit assistance system without engaging it.

H3: Mean workload was lowest for the explicit condition, however the difference was only significant when compared to the control condition. The implicit input condition was rated as higher workload than the explicit condition and lower than no assistance at all, however these differences were not statistically significant, as shown in Table 6.2.

H4: Participants indicated that the explicit assistance interface was more controllable, rating it 1.25 points (CI .52, 1.99) more highly on average on our assistance composite scale (reported in Table 6.3 and Figure 6.5).

H5 Participants rated their understanding higher on average, but the difference was not statistically significant as shown in Table 6.3 and Figure 6.5.

6.4 Discussion and Limitations

We designed an explicit-input teleoperation interface that is interpretable, responsive, unobtrusive and capable. These design principles have inherent trade-offs. For example, making assistance more capable may result in a less responsive and less usable system. Although our study considered a single fixed grasp sampling region, the size of this region as well as the nature of any additional

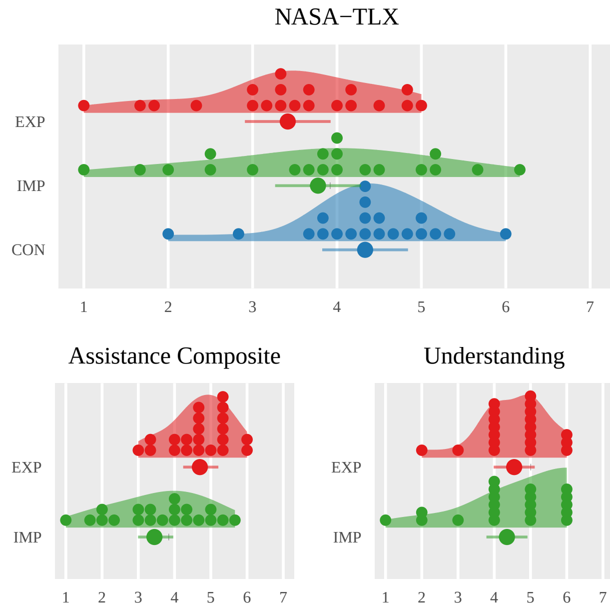


Figure 6.5: Raw data for subjective scores collected on 7−point scale with density estimates overlaid. Point and bar show estimated marginal mean with 95% confidence interval.

Table 6.4: Failure counts ↓

	A	B	$M_A(SE_A)$	$M_B(SE_B)$	$M_A/M_B(CI)$	p
Pick	EXP	IMP	1.13 (.32)	2.48 (.59)	.46 (.23, .91)	**.028**
	"	CON	"	4.22 (1.15)	.27 (.10, .75)	**.008**
	IMP	"	2.48 (.59)	"	.59 (.27, 1.27)	.242
Place	EXP	IMP	.55 (.18)	.59 (.18)	.93 (.38, 2.29)	.980
	"	CON	"	1.22 (.30)	.45 (.20, .98)	**.043**
	IMP	"	.59 (.18)	"	.48 (.23, 1.04)	.065

potentials applied to "snap" poses constitute simple configuration to manage this balance, not dissimilar to how we created a configurable balance between performance and expression in Chapter 4.7.

Our implementation is deployed in simulation, making it applicable to simulated data collection or robot teaching interactions. Porting our system to teleoperation of a real robot would require the integration of appropriate generative grasp- and placement-pose models, as well as object state estimation or point cloud-based occupancy checking methods. Our experimental assessment of the interface informs and motivates the future development of physical implementations. Future work should also explore placement assistance with objects and support surfaces that are not well-approximated as planes.

6.5 Conclusion

We contribute a new framing for assistance interactions based on explicit input and two new algorithms and interfaces for online teleoperation, designed to leverage GPU-based parallel computation to calculate feasible grasping and placing options online—even in clutter. Our work goes beyond individual picks by also considering assistance during placement, thus offering a complete workflow for multi-step pick and place tasks. The results of our study highlight the promise of this new kind of assistance interaction, and motivate us to further explore how accelerated computation can augment teleoperation.

7 Whole-system Transparency: Incident Review

Robots operating in real-world environments inevitably encounter failures. When these failures occur, human supervisors must diagnose the issue, determine its cause, and decide on corrective actions. However, the sheer volume of information generated during robot operation—spanning sensor logs, video feeds, and internal state transitions—can overwhelm even expert users. Without effective transparency mechanisms, identifying relevant failure indicators becomes a time-consuming and error-prone process.

This chapter presents methods for improving the transparency of robot behaviors during failure review interactions by transforming complex, multimodal recordings into interpretable summaries. In both cases[1], we take existing robot behaviors, structured as finite state machines, and introduce post-processing techniques to distill key information. The methods we describe provide simple configuration to control the amount of information exposed to incident reviewers.

Section 7.1 develops natural-language "narrations" of robot experience for users browsing recordings of a mobile manipulator which failed to complete its task. The method consumes multimodal recordings, accepts annotated metadata about parts of the robot and the overall nature of the task, and leverages powerful language models to provide summaries. Because of the wide range in users' capacity or desire for additional information, we also show that narration can be generated in user-configurable levels of detail.

Section 7.2 develops an assisted review interface intended for technical reviewers of complex or multi-part robot failures. In keeping with our general framework for automating transparency, we consume multimodal recordings and a small number of annotated failure recordings to learn models of failure. We apply model-interpretability tools to identify modality-specific and temporal features contributing to predictions. Surfacing this information in a standard robotics data visualization tool enables users to better uncover failures.

Taken together, the works in this chapter demonstrate that our approach to automating transparency is applicable even for complex behaviors. Manual interventions would be hindered by the need to compose well in a modular system, where components are variously scripted or learned. Providing transparency from the data-layer makes the approaches tolerant of black-box components, as only their inputs and outputs are used.

7.1 Narrating Robot Experience for Failure Localization and Recovery

Natural language is a promising channel for communicating information about robots' behaviors and experiences [163]. Some previous works have attempted to make robotic systems more transparent using language [18], [37], but these efforts are either limited to specific domains or require robot behaviors with rigid symbolic specifications. The natural language interface and commonsense reasoning capabilities inherent to large language models (LLMs) make it feasible to provide grounded text descriptions of real-world robot experiences. Such descriptions could be applicable across a broad range of domains, potentially leading to improved safety and user satisfaction in applications such as assistive feeding, home robotics, and self-driving vehicles.

Grounding real-world robot experiences into natural language presents several challenges [164]. First, robot data is multimodal, making it difficult to process and integrate. Mobile manipulators produce RGB images from various viewing angles, point clouds, audio recordings and more sensor data. These disparate data formats and semantics complicate the processing and integration of

[1]Materials from this chapter are adapted from:

[161]	Section 7.1	CoRL 2024	Wang, Liang, Dhat, Brumbaugh, **Walker**, Krishna, Cakmak
[162]	Section 7.2	In submission	**Walker**, Wang, Grotz, Cakmak

multimodal input. Secondly, robot data has different sample rates, making alignment difficult. Visual data from cameras might be captured at a different frequency than data from force sensors or joint encoders. This discrepancy complicates synchronizing data streams to create a coherent representation of the robot's experience. Lastly, the system must filter and prioritize relevant information to avoid overwhelming the user with unnecessary details.

In this case study, we introduce an LLM-based system for generating narration of robot experience based on recorded multimodal data. We consider users with different use cases and levels of expertise, ranging from no prior experience with robots to expert-level familiarity. These scenarios incorporated different narration modes with diverse narration strategies and abstraction levels. Finally, we conduct user studies and find that the generated text improves people's accuracy and speed in localizing and identifying issues with robot behavior execution.

7.1.1 Related Work

Robotics and LLMs LLMs and VLMs can be used in almost every part of robot development [165], [166]. In perception, vision-language model and vision-language-action models have demonstrated they can significantly enhance the generalization capabilities of robots [132], [167]–[170]. In decision-making, LLMs have been used to make planning and execution more flexible and context-sensitive [171]–[177]. In control, language-conditioned policies and transformer-based robot control have been widely studied in recent research [131], [132], [171], [178]–[183]. Finally, LLMs can significantly improve both robot-environment and robot-human interactions [184], [185]. In this work, we focus on using LLMs to enable natural language grounding of robot experiences and applying the grounded experiences to resolve downstream real-world robot tasks.

Scene and Action Understanding for Robots Scene understanding involves recognizing objects, their relationships, and the context within a scene, while action understanding requires

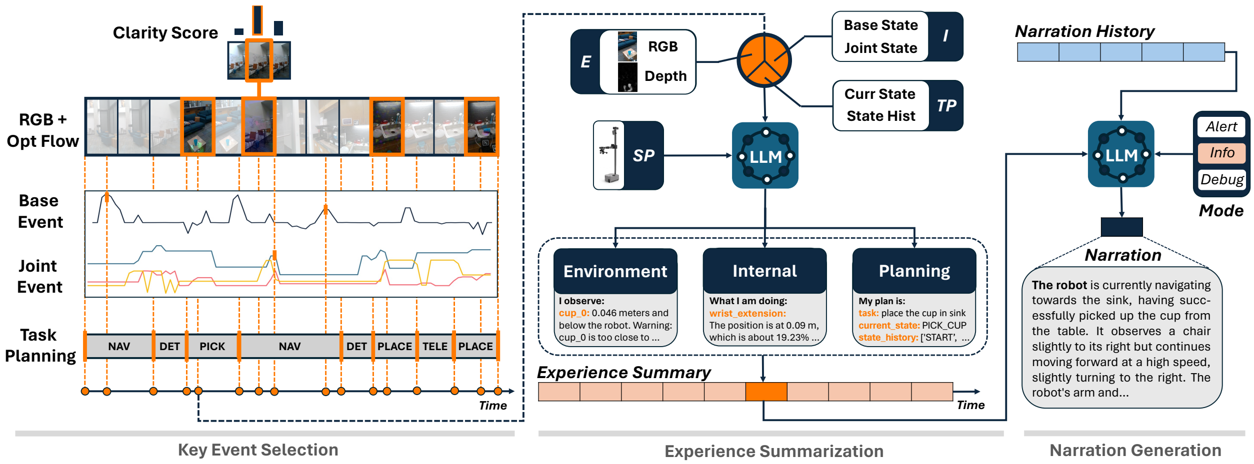


Figure 7.1: Our framework for robot narration has three components: key frame selection, experience summarization and narration generation. It takes in the raw multimodal robot data stream and outputs text describing past experiences, current observations, and future plans of the robot.

interpreting the robot's actions, understanding the outcomes, and planning future actions accordingly. Although dense video captioning and scene-graph generation have been extensively studied in computer vision [186]–[191], these models cannot be directly transferred to robots due to distribution mismatches. With the rise of embodied AI and LLMs, new approaches for scene representation and understanding in robots have emerged [192]–[198]. Scene and action understanding must work together to solve robotic tasks, such as failure explanation [177], [199]–[203], affordance estimation [171], [204], and task execution [205]–[208]. We introduce a framework to ground robot experiences with both scene and action understanding. Unlike previous work [177], our framework includes both low-level control and high-level planning actions.

Natural Language Summaries of Robot Experience Language is a naturalistic means of providing information about a robot's behavior which requires only a speaker or a display, but previous approaches have been constrained to engineered templates, often in conjunction with symbolic models of the behavior [209]. Our framework does not require any engineered templates or symbolic models, which makes it more generalizable and easy to use.

7.1.2 Multimodal Key Event Selection

During execution, massive amounts of data are generated at various rates from the robot's sensors, which can create data that is too repetitive and dense to be interpretable for users. The sampling rates of different robot sensors can vary significantly, creating difficulties in aligning information to be processed together. These challenges necessitate some procedure for aligning and sampling valuable information from the data. We call such samples key events, which are used to generate experience summaries in Section 7.1.3 and narration in Section 7.1.4.

Multi-Sensory Data Alignment We split robotic data into three categories: Environment (E), Internal (I) and Task Planning (TP). We sample and align these dense, mixed data by dividing the duration of the data by a single sample rate s for a sequence of frames $f_0, f_1, \ldots, f_n$, such that frame f_i and f_{i+1} are separated by time s. In each frame, we add the robot information across each considered medium, using the information with the timestamp closest to the frame timestamp. The result of this procedure is a sequence of frames separated by a fixed interval that captures information across robot sensors that may have mixed, high-frequency sampling rates.

Key Event Selection with Multimodal Inputs Key events are selected from the aligned data by heuristically monitoring for interesting information across the different data categories. For environmental data, we compute the optical flow of the RGB images to capture the motion dynamics within the scene, using the running sum of average flow magnitudes as a heuristic for computing changes in perception information sufficient for a key event. For the internal state of the robot, the joint states are used as a heuristic for observing changes in robot motion sufficient for a key event. Observing that changes in both flow magnitudes and joint states are significantly different for when the robot is moving its base, moving its camera, and all other movements, we normalize each of these values to have a mean of zero and standard deviation of one, and track the running positive sum of the normalized values. Once the running sum reaches a set threshold, we note that there should be a key event. We also add a key event each time there is a state change in the task planner, with the reasoning that state transitions are indications of notable events.

7.1.3 Experience Summarization

With selected key events, the next step is to ground raw robotic data into experience summaries in natural language. Based on the categories of the robot data, the experience summary is also composed of three components: environment summary, internal summary, and planning summary.

Environment Summary The goal of an environment summary is to ground the observations from the robot into natural language. We use YOLO World [210] to provide open-world object segmentation of RGB images. The detected objects are represented by a bounding box coordinate and unique object id, forming an detected object set O_{det}. Since the real environment is complex, the observation of a scene can contain an excessive number of objects. Our system leverages depth information to filter out irrelevant objects based on certain distance criteria, c_d. The remaining objects form an object set for the scene,

$$O_s = \{o_s | (o_s \in O_{det}) \cap (d_{o_s} < c_d)\} \tag{7.1}$$

where d_{o_s} is the distance between object o_s and the robot. We have a spatial relation set for the objects, which is defined as $P_s = \{$*left to, right to, above, below, in front of, behind*$\}$. The scene graph is a set of object, relation and distance triplets, $R_s \subseteq O_s \times P_s \times D_s$.

Internal Summary The goal of the internal summary is to ground numerical state components (e.g. base states, joint states, etc.) to natural language based on the configuration of the robot. Each part of the robot is annotated with three pieces of information: part description, part limit and part type. This annotation is specified as part of the system prompt for the internal summary generation. The system prompt also specifies that the internal summary should contain exact names and numerical values of the part states. The final internal summary contains a list of robot parts with exact part names, a detailed description with numerical values, and a grounded explanation that people without robot experiences can understand.

Planning Summary It is hard to infer and narrate the expected outcomes for every one of the low-level actions contained in the internal summary. Therefore, a planning summary is generated to to summarize the high-level plan of task execution. It contains a description of the overall task, the sequential order of sub-goals, the current sub-goal and a history of sub-goal executions and outcomes. Unlike other methods only using the current sub-goal in their planning summary, one critical change is we also include a history of sub-goal executions and outcomes. This enables narration and failure analysis for long-horizon tasks.

7.1.4 Narration Generation

The experience summaries ground environmental observations, internal status and task planning of a robot during task execution into detailed natural language. However, not all of the details are useful for humans to understand and react to the robot. We need to abstract the information and only narrate things users care about.

Narration Mode The requirements for narration can vary significantly depending on the robot's use cases and the user's level of expertise. To meet general narration needs, we have defined three modes: 1) *Alert Mode* narrates only the important information which requires the user's attention; 2) *Info Mode* narrates robot experiences in multiple sentences and provide a concise summary of the robot's observations, internal status and planning without any numerical values and part names; and 3) *Debug Mode* incorporates all details of environmental observations, robot internal status and the robot's planning, including numerical values and attribute names. The mode is an input parameter to the LLMs and controls the properties of generated narrations. We envision that users can specify the mode of narration as needed, provided a simple but powerful means of configuring the transparency they are provided.

Progressive Narration Generation We consider a generated narration to be good if it has the properties of non-repetition and smoothness. Non-repetition means the narration should not repeatedly describe behaviors which have previously been narrated. Smoothness means the transition between narrations should be natural and seamless. We use progressive generation to achieve these properties. Consider a robot narration history which contains all the narration instances until key event $t-1$, $N_{t-1} = \{n_0, n_1, ..., n_{t-1}\}$. When there is a new key event, k_t, a new experience summary, s_t is generated. We input both N_{t-1} and s_t with a specified mode m to an LLM to generate the narration, n_t, at time t, where $n_t = LLM(N_{t-1}, s_t|m)$. Then, we add the narration n_t to the narration history. The most recent generated narration n_t will be shown to the user in our user interface. This process is shown in Figure 7.1. For our experiments

7.1.5 Experiments

Failure Dataset We collect a dataset using a Stretch SE3 robot in a home environment [211]. We created four real-world housekeeping tasks: pick a dirty cup and put it in the sink, microwave lunch, hang a hat, and collect dirty clothes. We collected data, including RGB-D observations captured by two cameras (an Intel RealSense D435i on the head and a D405 on the gripper), joint readings, base readings, state information, and diagnostics. We also save the processed data which includes downsampled aligned keyframes. For each demonstration, human experts create ground truth labels for failure timestamps, failure reasons, and recovery instructions. The dataset contains 70 demonstrations and 76 failure cases across navigation, manipulation, and detection. Figure 7.3

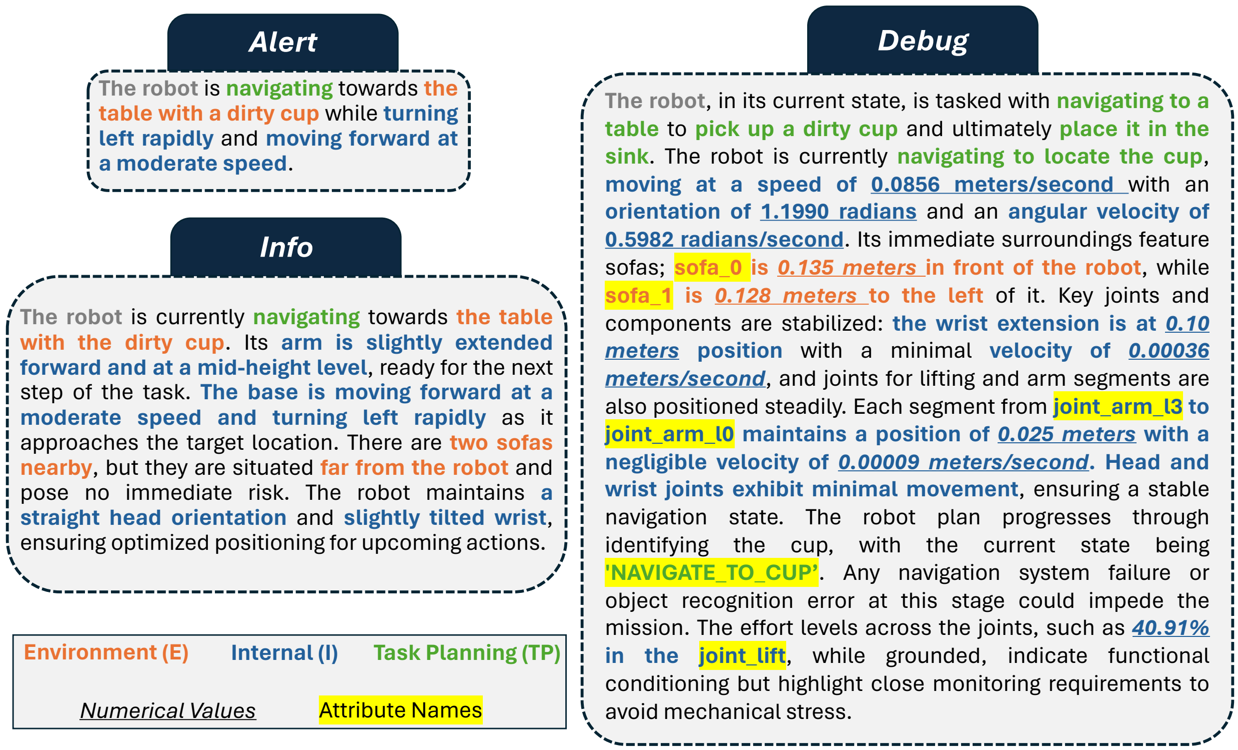


Figure 7.2: Example narration generated by our approach in different modes.

shows the task and failure composition of the dataset.

User Study Setup We conducted a user study to evaluate our narration generation method and interface, both qualitatively and quantitatively. The user study consisted of two sections: narration quality evaluation and failure identification using narration. We recruited 24 participants for the user studies. Prior to each section, we provided a tutorial to familiarize the participants with the tasks. At the conclusion of the user study, we administered a questionnaire to gather information about the participants' demographics and their background in robotics.

Experiment I: Narration Quality Evaluation We compare our method with the following baseline methods and ablations (GPT-4o is used as LLM and VLM for all the methods): BLIP2,

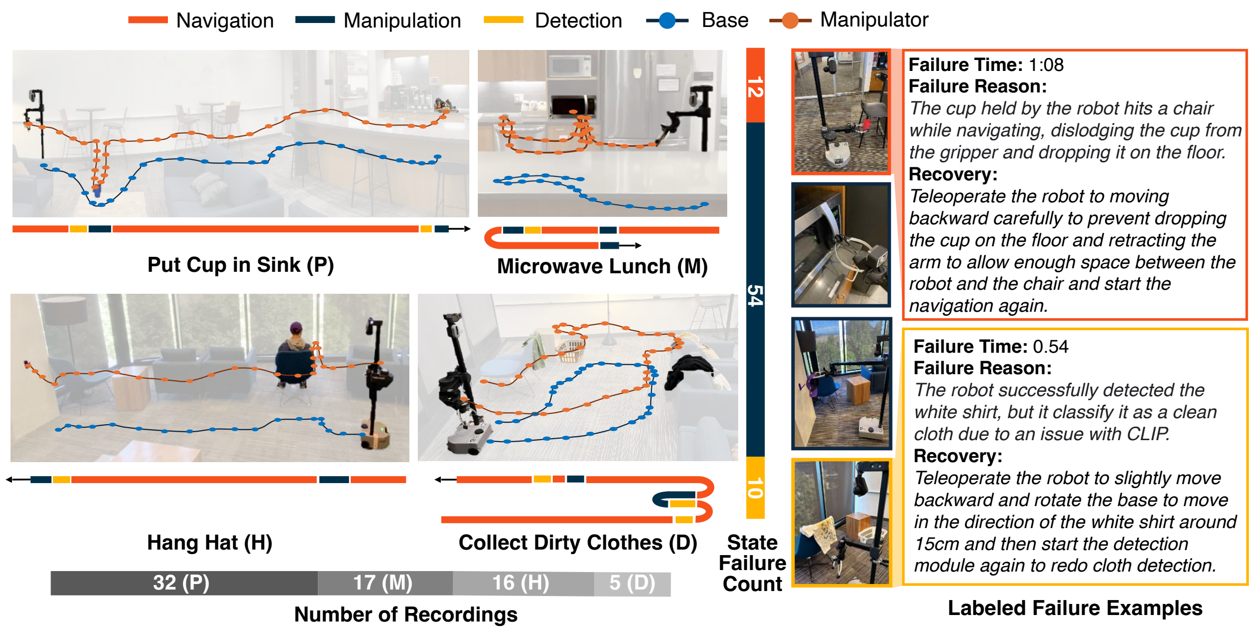


Figure 7.3: We design four long-horizon tasks for a Stretch robot in a home environment. **Left:** the different tasks with base and manipulator trajectories, as well as flow of states the robot experiences in each task. **Right:** the number of failure cases under each robot state in the dataset. The pictures are failure cases selected from the dataset and the text are human-expert-provided ground truth labels for the frames.

REFLECT, TEM-LLM (all raw sensory data directly input to LLM), TEM-VLM (all raw sensory data directly input to VLM), and RONAR (our method).

Participants rated the generated narrations on naturalness, informativeness, coherence, and overall quality using 1-5 Likert scales. The evaluation criteria for informativeness and coherence follow common practices for human evaluation of natural language generation [212]. The study began with an introduction to the evaluation metrics. Participants engaged in a rating practice session involving two sample image-narration pairs to verify their understanding. Following this, each participant was presented with a sequence of narrations generated by five methods, accompanied by three image-narration pairs. The results are shown in Table 7.1.

Experiment II: Failure Identification by Human with Narration We selected four failure cases in different states (two in navigation, one in manipulation, and one in detection) from

Table 7.1: User ratings on narrations generated by different methods

Method	Naturalness	Informativeness	Coherence	Overall
BLIP2	3.25 ± 1.18	1.69 ± .87	2.56 ± 1.59	1.81 ± .91
REFLECT	2.13 ± .95	2.94 ± .99	2.88 ± 1.08	2.81 ± .98
TEM (LLM)	3.94 ± .85	4.06 ± 1.06	4.25 ± .77	4.13 ± .71
TEM (VLM)	3.38 ± .80	4.06 ± .99	4.06 ± .85	3.75 ± .77
RONAR (Ours)	**4.19** ± .91	**4.56** ± .62	**4.56** ± .51	**4.50** ± .51

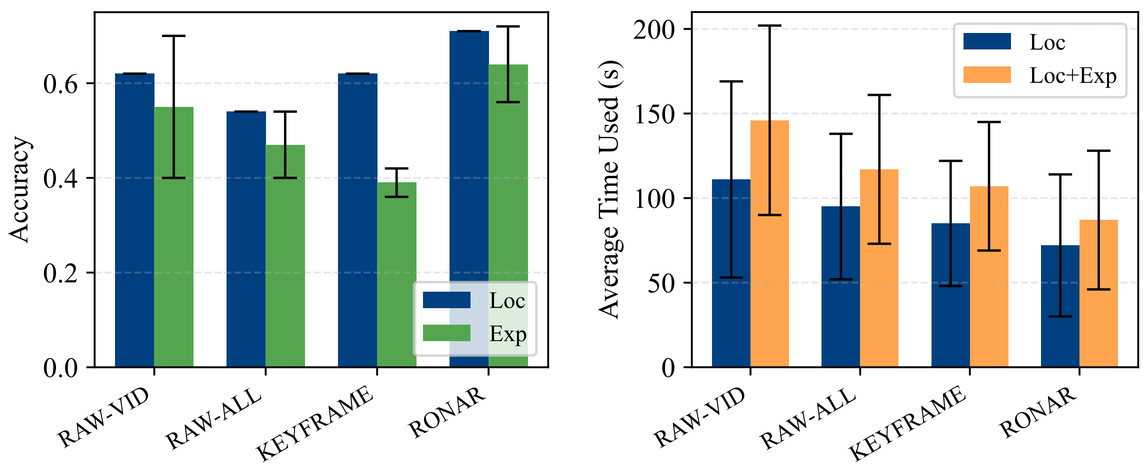


Figure 7.4: Left: The accuracy of failure localization and explanation from the user study using different interfaces. **Right:** The average time taken by participants to localize failure and both localize and explain failure using different interfaces.

the putting cup in sink task from our dataset. We prepared four interfaces for the participants: video interface, video and sensory information interface, keyframe interface (RONAR-UI without narration), and RONAR-UI. Participants were given a full demonstration of a failure displayed on each interface. We asked participants to type their answers for the *time of failure occurrence* and *failure explanation* for each failure demonstration. We timed participants as they answered each question using the interface. The results are shown in Figure 7.4.

7.1.6 Results

The method can generate high-quality narrations. Table 7.1 shows the quality of the narrations on different metrics. For naturalness, our method outperforms BLIP2, REFLECT, and TEM (VLM) with a slight improvement (0.25) on TEM (LLM). For informativeness, all LLM-based methods have a similar performance, leaving a large gap compared to other methods. Our method has the highest overall rating, outperforming the second-place method by 0.50 scale points.

Narration improves users' accuracy and efficiency in failure analysis. In Figure 7.4, our interface outperforms other interfaces in both accuracy and efficiency in assisting users in localizing and explaining the failures. One interesting finding is that a raw data interface with all sensory inputs does not help users achieve better failure localization and explanation accuracy than a raw video interface. With similar accuracy, our method significantly reduces the time used for failure analysis compared with the raw video interface.

Limitations We use two-step LLM-based summarization to generate narration, which makes the system slow and affects the user experience. Our experiment is limited to a single mobile robot within a single environment. More types of robots and environments should be studied to characterize the generalization of the proposed approach.

7.2 HIGHLIGHTING DATA FOR REMOTE ROBOT INCIDENT REVIEW

Human supervision is a key part of addressing the failures autonomous systems inevitably face as they scale. Today, fleets of commercial autonomous vehicles roam with remote supervision as a primary fallback mechanism [213]. Beyond "distress call" interventions, humans also make determinations about whether the issues or environmental circumstances they've identified require limiting fleet operations [213]. Other instances of large-scale remote robot supervision include ground delivery [214] and warehouse logistics robots [215]. Researchers have also used as-needed intervention to extend the capabilities of their limited systems [216].

Beyond recovering from mistakes, human supervisors provide data that enables continual improvement of large scale autonomous systems. Commercial operators do not generally disclose the types and scale of supervision that feed into their systems, but a report on one data labeling service provider uncovered that thousands of crowdworkers are employed to complete visual annotation tasks for autonomous vehicle makers [217]. Several startup providers offer human-in-the-loop solutions for vehicles and warehouse manipulators, with some positioning the approach as a means to address gaps in data. For other groups, understanding and diagnosing errors is an internal facet of the development cycle for autonomy teams, much as it is for many robotics researchers.

Robot supervisors are often remote, working without the benefit of direct observation, and usually entering situations without having observed the lead-up to the incident. For each new incident, they must extract context from reams of multimodal data across formats like audio, video, numeric timeseries and textual logs. Sometimes, important information must be inferred due to limited sensing. When supervisors intervene in live systems, there is pressure to reach a resolution and prevent further service disruption. When they review past incidents, there is an incentive to minimize time spent annotating or labeling so the task doesn't impede product development.

In this work, we investigate how to assist remote robot supervisors in understanding and labeling issues in recorded robot data. We consider settings where an overall indication of failure is available, but the combination of underlying issue(s) present cannot be automatically determined. Humans complete this multi-label classification task so that the data can be best used to drive system improvements or develop comparable test scenarios.

We contribute 1) a novel approach for generating interpretable labeling assistance, a step

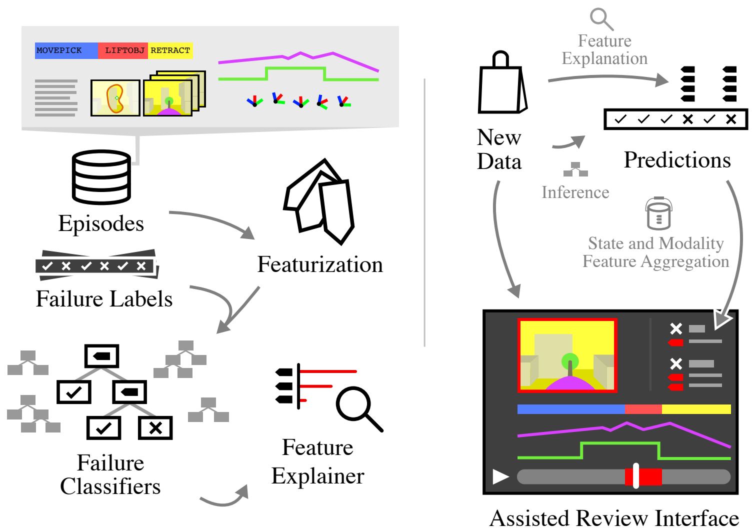


Figure 7.5: Our approach to providing assistance to human supervisors as they label issues in robot data. **Left:** multimodal training data consisting of a limited number of labeled episodes is featurized and used to train weakly predictive models. The models and the dataset are used to calculate feature importance weights. **Right:** an unseen episode is annotated with predictions from the learned model, and the feature weights are used to increase the salience of particular times or modalities of the recording.

towards context-aware interfaces for understanding robot failures; and 2) insights from a user evaluation of the approach, realized in the context of a warehouse picking system [218].

7.2.1 Related Work

The problem of analyzing and diagnosing issues in robotic systems has many antecedents. Many researchers have looked at how to detect failures or anomalies online during manipulation. [219] provides an early example of a machine-learning driven anomaly detection method aimed at high-dimensional robot data. Motivated by safety needs in physical human-robot interaction, [220] showed an online multimodal execution monitoring approach that adapts its detection threshold based on the phase of execution. [221] showed a multimodal approach to detecting manipulation failures incorporating RGBD and audio data. [222] furthered this work by introducing a symbolic learning framework for success and failure detection in grasping and mobile manipulation tasks.

Failure classifiers are often sought because they can enable online adaptation or recovery. Recently, [177] demonstrated that large language models (LLMs) can identify failures in summaries of multimodal robot data. [223] demonstrated that vision-language models (VLMs) can be trained to detect and reason about failures in natural language, enabling online adaptation under a variety of manipulation paradigms.

Other work has focused on providing detected failures to assist human supervisors. [18] characterized the types of information that are useful in robot failure explanations and demonstrated an encoder-decoder model for generating them. [161] demonstrated that LLM-generated summaries of multimodal robot data can help humans identify and pinpoint the time of a failure in a long-horizon task. [224] compared contrastive and feature-level explanations. [225] investigated the use of feature-based interpretability tools, augmented with domain-knowledge in the form of causal relationships between features, to produce failure explanations. We make use of similar tools in our work, aimed at enhancing visualizations of multimodal data.

Little work has explored the design of tools or assistance for robot debugging. [226] studied roboticists' experiences using RViz, 2D GUI visualizations, and an AR interface. As suggested by the authors, our work explores one path toward a context aware interface.

7.2.2 Interpretable Robot Failure Diagnosis Assistance

We consider robot data in episodes, each consisting of a multimodal set $\mathcal{M}$ of data streams. An episode represents a task-dependent meaningful unit of interaction, for instance an individual pick attempt in a warehouse manipulation system, or an intersection crossing for an autonomous vehicle. Typical modalities may include sensor data streams, like camera image and force torque streams, as well as other internal data, like the name of the current phase of the robot's behavior or the sequence of textual log information.

We seek to assist the human supervisor in identifying issues in new episodes by providing predictions coupled with "highlights" applied to make parts of the data more salient. To do this, we couple a learned model for predicting the labels with Shapley additive explanations (SHAP), a feature-based model interpretability tool. We aggregate the features determined to be important to each prediction so that they can be presented to supervisors in a typical robotics data visualization interface.

Describing Failures

Each episode may be characterized by zero, one or several labels from an assumed taxonomy L of relevant issues. For instance, a failed pick episode may exhibit an object escaping a grasp by deforming, as well as a poor grasp pose selection. An episode e is characterized by binary vector $y_e \in \{0,1\}^L$, with each entry indicating the presence (1) or absence (0) of an issue.

We assume that it is challenging to automatically identify a complete label vector y_e, but that doing so is important enough to warrant the use of human supervision. This happens often when deploying systems in open-ended environments, where (for instance) perception or manipulation systems operating outside their training regime fail in various—and likely multiple—ways. Ascertaining y_e assists development by directing attention to precisely representative test scenarios, and by providing supervised data for the individual components of the robot system that are associated with each label.

Feature Extraction

Features are basic numeric scores derived from data streams. They fall into two categories: (1) *modality-specific*—e.g., the descriptive statistics of force measurements, or features of mask of a target object; and (2) *phase-based*—e.g, the statistics of the end effector goal pose error windowed to a particular phase of execution.

Any feature that can be computed over the entire episode may also be computed over a subset of the execution, thus the majority of features are expected to fall into the phase-based category. Many systems have meaningful internal representations of the current phase of execution because they are specified as behavior trees or state machines. Systems operating with fewer internal explicit representations would need to window data by, for instance, using dynamic time warping and a fixed window size.

The nature of the features depends on the modalities at hand, with the only constraint of the approach being that each feature be associated with some defined subset of data streams. We observe that standard descriptive statistics and minimum and maximum quantity information are informative when taken for meaningful windows of the robot's execution.

Predicting Failures

We treat the prediction of failure label vector y_e as a conventional multiclass classification problem, for which labeled data are available. Let $\mathcal{D} = \{(x_e, y_e)\}_{e=1}^{N}$ denote our dataset, where $x_e \in \mathbb{R}^d$ is the feature vector extracted from multimodal sensor data for episode e. We seek to learn a function $f : \mathbb{R}^d \to [0,1]^L$ such that $\hat{y}_e = f(x_e)$, with each $\hat{y}_{el}$ representing the predicted probability of failure mode l for episode e.

We train f by minimizing the binary cross-entropy loss. There is no particular constraint on the form of the model, which is instead dictated primarily by the volume of data available.

SHAP-based Attribution and Feature Aggregation

First, we compute Shapley additive explanation (SHAP) values [227] to obtain a decomposition of the model output:

$$f(x) = \phi_0 + \sum_{j=1}^{d} \phi_j(x),$$

where ϕ_0 is a baseline model and each $\phi_j(x)$ quantifies the contribution of feature j.

These SHAP values are then aggregated in two ways: by modality (e.g., audio, force, camera feed, diagnostics) and by state (as defined by the robot's operational state sequence).

We define a mapping $M : \{1, \ldots, d\} \longrightarrow 2^{\mathcal{M}}$ such that feature j is derived from a set of modalities $M(j)$. Then, the aggregated SHAP value for a modality $m \in \mathcal{M}$ is given by

$$\Phi(m) = \frac{1}{N} \sum_{e=1}^{N} \sum_{j : m \in M(j)} \frac{\phi_j(x_e)}{|M(j)|}.$$

For features that incorporate behavior state information, let $\mathcal{S}$ be the set of state windows and define a mapping $S : \{1, \ldots, d\} \longrightarrow 2^{\mathcal{S}}$. The aggregated SHAP value for a given state $s \in \mathcal{S}$ is then defined as

$$\Psi(s) = \frac{1}{N} \sum_{e=1}^{N} \sum_{j : s \in S(j)} \frac{\phi_j(x_e)}{|S(j)|}.$$

Abstracting individual feature contributions to time window groups of modalities makes explanations more accessible to human supervisors, as it aligns with how information is naturally displayed.

Concise Feature-Based Explanations

To produce a concise "explanation", we rank the aggregated SHAP values separately for $\Phi(M)$ and $\Psi(S)$ in descending order, and we select the top K positively contributing entries. If a knee point—indicating a sharp drop in contribution magnitude—occurs before the Kth rank, only the entries preceding the knee are included. The resulting items "explain" the prediction of a particular issue in terms of portions of the data that inclined the model to the label.

The appropriate value of K depends on the number of relevant modalities and states in the data. However, it is also limited in practice by users' willingness to review information. In our

Figure 7.6: Still frames of failure cases during interaction in cluttered scenes. From top left, **1:** a misaligned grasp pose (GP). **2:** a misaligned grasp (GP) and the object shifting off the stack (MDF). **3:** a poor grasp pose (GP) and suction being lost as the objects contact the lip of the shelf. **4:** the robot hitting another object while extracting, causing it to fall from the shelf (MED). **5:** the suction seal with the target object is lost due to the plastic wrapping on the object. **6:** the robot halts (MOV) and a high force fault halts execution (NEW).

experience, $K = 2$ or 3 is appropriate.

7.2.3 Experiments

System

We investigate our proposed approach to assisting operators with failure diagnosis in the context of an industrial picking workcell. Each episode consists of a pick, wherein a UR16e robotic arm uses its suction gripper to retrieve a particular object from a set of shelves.

Pick Failures The researchers developing the picking workcell developed a taxonomy of failure modes particular to their aims of improving its performance in retrieving objects in densely packed containers [218]. We selected a subset of this taxonomy, shown in Table 7.2, so we could easily introduce new users to the taxonomy in the context of a single-session user study.

Dataset We recorded and manually labeled 192 picks to obtain the dataset $\mathcal{D}$, yielding 378 labels. Figure 7.6 shows some exemplary failures cases that were recorded for the dataset. The set of 23 data streams $\mathcal{M}$ recorded includes: audio and video from sensors mounted on the gripper, joint states and end-effector force torques, vacuum levels, pre-pick image of the shelves along with perceived object mask, gripper target poses and current end effector pose, log messages including originating process and severity level, and hardware provided diagnostic information. The system provided timeseries sequence of current state names from the state machine that drove its behavior. The requested object's product description was also included with the recording.

Features We extracted common features from each model. These included force-torque magnitude, goal-error magnitude, audio spectral features including MFCCs, optical flow magnitude for video, log cluster counts, state durations and transition counts. Video frames were characterized by a VLM for the presence of visual attributes (e.g. the count of stacked objects visible, or whether any bagged objects are visible). Features that were themselves timeseries were represented by their descriptive statistics over the whole pick and for the window of each state. The final representation has 6602 features. All features are normalized to have zero mean and unit variance based on the statistics of the dataset.

Models We train random forest decision trees independently for each label code using [228] and use TreeExplainer [229] to calculate SHAP values. Quantitative performance of the models is not the emphasis of our evaluation, as we are interested in regimes where there is limited data to support highly accurate models, thus creating the need for some form of assistance for human labelers. We trained models on 20 folds of data with 10% held-out test data (stratified to appropriately represent classes in both partitions), and found an average R^2 of 0.29 (SD=0.03). This level of correlation is not suitable for during-execution use, but is sufficient to expose meaningful trends in the data under human review.

Interface Participants viewed recorded picks in a customized version of LichtBlick[2]. We selected the tool because it implements best-practice visualizations for a wide range of robot data, and because its timeline format is easy to learn for users unfamiliar with the tool. We created a data layout displaying much of the recorded data, shown in Figure 7.7. Due to limitations of the tool, participants were not able to listen to the recorded audio. We implemented a simple multi-selection interface to allow supervisors to label the recorded pick's issues and take notes in case of ambiguities.

Assistance We incorporate assistance by rendering the top 3 predicted label codes as "suggestions" in a labeling interface. Users can click to review each suggestion, enabling highlights which render a bounding box around modalities contributing to the suggestion as well as highlighting portions of time that contributing to the suggestion. A textual summary of the contributing features is generated via a combination via templated language indicating the modality or state, whether the features were high or low, passed through an LLM to paraphrase and consolidate redundant language. The interface is detailed in Figure 7.7.

Pilot Feedback Three users, all roboticists previously unfamiliar with the robot or task context, were recruited to provide feedback on an initial version of the assisted interface. This feedback was used to determine the number of suggestions to display (a maximum of 3), as well as to refine the design of the suggestion review interaction.

[2] A forked open source continuation of Foxglove (`https://foxglove.dev/`), which was itself a fork of autonomous vehicle-maker Cruise's Webviz project

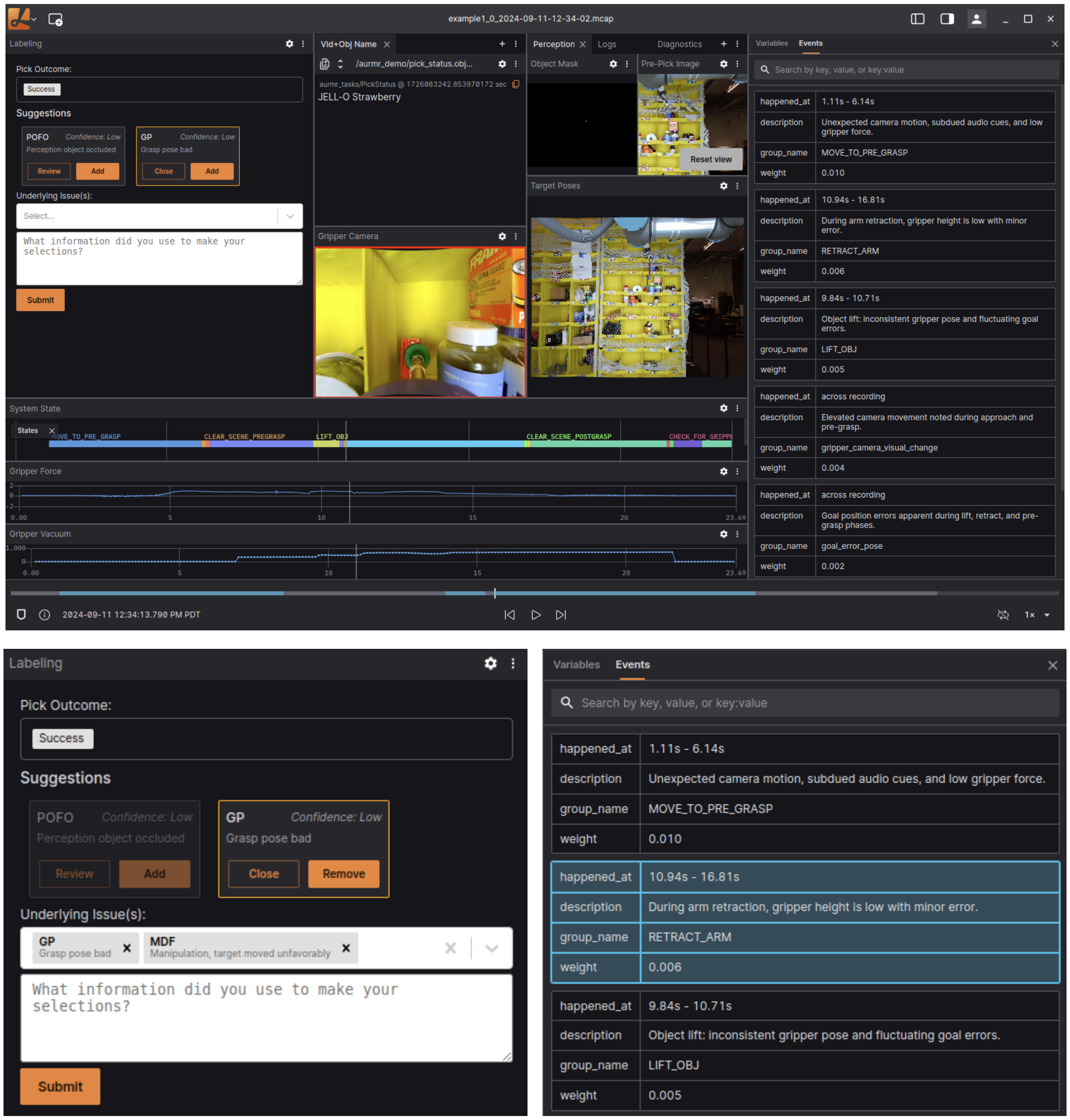


Figure 7.7: The visualization tool users interacted with. Top left: a pane showing suggested issues and allowing the user to select labels. **Top image**. Top center: the product description of the requested object and the video feed from the gripper camera (highlighted in red due to the currently selected suggested label), as well as perception mask and grasp pose composite visualizations. Right: a list of contributing features for the currently selected "suggested" label. Bottom: state sequence, gripper force and suction plots, and timeline showing highlighted regions for the currently selected suggestion. In the unassisted condition, the suggestions are not displayed, the rightmost panel is hidden, and no highlights are drawn. **Bottom left**: detail of the labeling interface with suggestion for code POFO dismissed, and code GP currently in review. **Bottom right**: detail of the contributing features for the GP code under review. The user has selected the second entry, which caused playback to move to the beginning of the arm retraction phase.

Experimental Design

Our research interest is in understanding:

1. The effect of assistance on reviewers' labeling performance
2. The impact of the assistance on the reviewers' subjective impression of usability and cognitive workload;
3. What aspects of the assistance participants use;

To address these questions, we conducted a within-subjects user study where participants analyzed pick executions in 10 minute time trials. Participants completed the task in conditions without assistance (**CON**) and with assistance (**AST**). Condition order was counterbalanced.

Two sets of 12 recorded picks were selected from the test sets of various folds of the model training. The two sets were balanced to contain a similar mix and ordering of issues, and the set assigned to a particular condition was counterbalanced.

Procedure

All participants were briefed about the robot, the task, and the failure taxonomy. Participants were shown an example, and warned that real failures were variable and occurred in different combinations. They were told a system had flagged these recordings for review, so it was likely there was something wrong, but not, to mark the "NONE" code. They were also instructed that it was possible for rare failures to not fit any existing codes, and to use the special "NEW" code for these. The experimenter answered their questions.

In the control condition (CON), participants labeled picks using the information layout shown in Figure 7.7. Participants in the assisted condition (AST) used the same layout, with an additional interface element that displayed suggested labels and allowed them to enable highlights associated with each suggestion. They were told that the highlights were automatically flagged regions of the recording which the system believed contained useful information, and that it was ultimately their responsibility to make the correct determination.

After each condition briefing, participants tried the interface for one tutorial pick and the experimenter answered their questions. Participants were instructed to attempt to minimize mistaken labels and to label as many picks as possible during the 10 minute time trial.

Participants were compensated with a $25 gift card. The plan for this research was approved by an Institutional Review Board.

Measures

After each condition, participants completed a NASA-TLX cognitive workload questionnaire [160], a Systematic Usability Scale (SUS) questionnaire [230], a Trust in Automation questionnaire (for the assisted condition) [231], and responded to open-ended prompts to "describe [their] strategy for labeling the failure causes," state "what features of the overall system [they used] most, or [found] most helpful?" and to describe "which aspects of the system did [they found] confusing or counterproductive?".

At the conclusion of the experiment, participants were asked which system they preferred and why. Finally, they completed a demographic survey. In addition to standard age and gender questions, we also collected self-reported familiarity with robots, familiarity with the picking workcell in particular, and years of experience working in robotics.

The system collected participants' labels and information about the times the browsed, and their use of the suggestions.

Participants

12 participants (10 male, 2 female), aged 19-55 (M=27.1, SD=11.7), were recruited from university robotics labs and undergraduate robotics teams. Participants had between 0 and 15 years of

experience as roboticists (M=5.5, SD=4.2). 4 participants were highly familiar with the robot, having worked with the platform before. Experienced participants were included to represent the perspectives of users with prior mental models of the system and its faults. Those unfamiliar with the platform approximate skilled technical staff that may be asked to review incidents with a new system.

7.2.4 Results

Labeling Performance Participants annotated 6 recordings on average in both conditions. Across the recordings they reviewed, participants provided an average of 9.9 (SD=4.6) correct annotations in the assisted condition and 8.0 (SD=3.5) in the control condition. Their precision was similar at 77% (SD=16%) and 73% (SD=20%) in the assisted and control conditions, while their recall was significantly higher using assistance: 74% (SD=8%) versus 59% (SD=15%), with $p<=.009$ for a paired t test.

Usability and Cognitive Workload We observed a trend that NASA-TLX scores decreased, measuring a 10% reduction from 43.6 to 38.7, however the difference was not statistically significant. Both systems registered nearly identical mean SUS scores of 72.3 (SD=14.0) for the control and 73.0 (SD=8.9) for the assisted condition. 8 of 12 participants preferred the assisted system, a ratio which is not statistically significant. The standardized Trust in Automation score, collected only for the assisted condition was 4.5 (SD=0.6).

7.2.5 Discussion

Participants found suggestions in time most valuable. Four participants made specific reference to the value of being able to see specific windows of time (saying, e.g. the "guesses it had for when the failure occurred" were most useful. P11) in their open-ended responses about the assisted system.

Suggestions use of undisplayed information confused participants. Participants were briefed that the suggestions had access to additional information from the recording, and several noted that this made the suggestions more difficult to verify, and thus less useful:

> Often, the plain text information was things [sic] that I couldn't verify in the image, or that I hadn't used to make my decisions in the prior experiment. (P2)

While limitations of our chosen visualization tool prevented us from displaying some kinds of information, the experience of disconnect between particular features and their associated modality is likely a general problem. Future work should explore how to reweight specific features associated with suggestions based on the their ease of decoding from the associated modalities.

Users are aware of the potential for automation bias. Some participants who preferred the unassisted interface expressed concern that the system would dull their skill in the task. One user, who preferred the unassisted system remarked:

> The first system gives me full control of the task and seemed easier to use. I believe the suggestions will make me lazy in the long run. (P8)

This concern is well supported in human-factors literature. Future implementations of assistance may consider presenting suggestions later in the labeling interaction to avoid over reliance.

Participant's skepticism of suggestions was generally well calibrated. The suggestions were weaker at the task than most users, something participants were able to ascertain quickly while still making effective use of the suggestions:

> I felt more confident in my answers even when contradicting the system. Even when the system was wrong about a suggestion, it was likely that the true issue was related to it's suggestion so it was easier to find those problem points. (P7)

Table 7.2: Picking workcell failure labels and descriptions

Code	Description
PIOI	Perception object identity: 90% or more of the masked pixels belong to a single object that is not the requested object.
POFO	Perception object occluded: The requested object is not visible in the bin or is more than 50% blocked by another object.
PSE	Perception segmentation error: More than 30% of the visible part of the target object is missing, or more than 30% of pixels belong to another object.
MEC	Manipulation extraction collision: The grasped object detaches because it collides with the bin (sides or lip) or another object while retracting.
MED	Manipulation extraction dynamic: Contact during extraction causes another object to fall out of the bin.
MDF	Manipulation target moved unfavorably: Contact shifts the target object, causing suction failure or preventing a proper seal.
MSF	Manipulation suction failure: A suction seal forms but unseals, or no seal forms despite more than 90% of the suction cup area contacting the object.
MOV	Motion planning failure: A movement cannot be made, is only partially completed, or happens after a delay of 20 seconds or more.
INF	Information missing: The recording ends before the robot returns to a known state or is missing important data.
GP	Grasp pose bad: The suction valve center is within 1 cm of the edge (or past the edge) of the target object, or is misaligned for small objects.
NEW	Something else: A problem or error that either caused a pick to fail or could have caused failure, but does not match another code.
NONE	No error is present.

8 Conclusion

Our goal has been to support the thesis stated in Chapter 1:

> *Unchanged robot behavior specifications can be made transparent automatically by manipulating their observable characteristics to align with learned or intuitive models of human inference.*

We did this by establishing a conceptual framework in Chapter 3 under which behavior specifications are passed through runtimes which manage their externalizations. We enlisted the aide of minimal annotations provided separately by behavior creators, users or others. To demonstrate this approach, we visited interaction contexts from observer attribution to the motion of home and office service robots (Chapter 3.3.4 and Chapter 4.7), to an assistive teloperation system for manipulation in cluttered environments (Chapter 5.5), to failure review systems for mobile manipulators and industrial picking workcells (Chapter 6.5).

The challenges to automating transparency that we identified in Section 3.2 manifested differently across our studies. We first saw how, for intrinsically motivated robots in Chapter 3.3.4, the disconnect between internal models of exploration and external observations in the task context could be bridged by simple explanations, demonstrating that minimal intervention can significantly improve user understanding.

In Chapter 4.7, we confronted a more complex channel management challenge of balancing transparency with task performance when controlling attributions to robot motion. We showed that a data-driven model of how humans make attributions to motion patterns could be constructed from simple annotations, and that the model enabled a configurable system that allows behavior creators to prioritize either efficient task execution or desired attributions without modifying the core behavior specification.

The assistive teleoperation system in Chapter 5.5 addressed the channel management challenge inherent in providing a visual display of assistance options. Our explicit pointing-based interface demonstrated that assistance can be made comprehensible simply by aligning the system's display and predictions with an intuitive model of users' expectations. The resulting approach also affords a natural configuration parameter that trades between the effectiveness of assistance and the smoothness of the presented information.

Finally, in Chapter 6.5, we tackled the challenge of information overload in failure review. By transforming complex multi-modal data into natural language summaries and highlighting statistically relevant features, we showed that transparency can be achieved even for complex behaviors with black-box components by operating at the data layer rather than requiring component-level interventions. Both methods we designed afforded simple configuration to control the amount of information surfaced to the user.

In each of these contexts, we adopted existing formats of behavior specification and we developed runtimes which intercede between the existing behavior and information externalization. We minimized changes to the underlying behavior, and instead concentrated any additional information required into the form of annotations on top of the behavior, or into simple low-dimensional configuration parameters. The runtimes that we developed were equally derived from data or from simple, generalizable models of how humans make sense of the particular channel under consideration.

Taken together, the methods and findings we have presented support a new perspective on how robot transparency can be accomplished, a way that minimizes its reliance on behavior creators.

None of the behavior runtimes we developed in the course of this work address the totality of the behavior transparency problem. Rather, we hope to persuade human-robot interaction researchers that *there is a path* towards systematizing best practice, and that they too should walk it.

In a decade or two, every robot will ship with a transparent and expressive behavior runtime. Every action taken will pass through optimizations, rules and balances derived from the community's research. Our ability to capture models of how humans comprehend a robot's externalizations will improve, and we'll need less annotation to address a broader range of behaviors. Where trade-offs must be made, simple configuration will afford behavior creators and users alike the ability to tune the transparency to their needs.

8.1 Future Work

Our work addressed several of the challenges inherent in providing transparency automatically, but some of those described in Section 3.2 remain largely unaddressed.

Evaluation-in-the-Loop Transparency We have evaluated transparency indirectly by its impact on task performance metrics. Task performance, however, is a noisy measure because it is influenced by other factors besides transparency. Further, end-to-end evaluations of interactions are not generally feasible while developing a behavior, and are costly in any case. The presence of a behavior runtime, a system which is necessarily aware- and in control of- a robot's internal state and its communicative output, may also be able to aid in evaluation. Coupled with either a means of detecting a desired outcome (potentially annotated on the behavior), or a generic way of querying whether a user is aware of some aspect of the state, the runtime can probe how its externalizations are impacting the user's understanding.

Adaptive Transparency We have largely studied short interactions, where both the user and the robot's behavior are static. Both parties experience changes across longer horizons which impact how transparency should be provided. As we pointed out earlier in the dissertation, users' roles and expertise levels are likely to change significantly over time. Responding automatically on behalf of the behavior creator would require that the system provide the appropriate configuration opportunities to users, or potentially dynamically adjust this configuration during interactions. Because behaviors themselves also evolve, we must also explore ways to detect when the runtime is no longer able to provide meaningful transparency. This would enable the runtime to gracefully degrade back to minimal transparency interventions or to solicit fresh annotations.

Portable Runtimes and Guidance A truly general behavior runtime—a system which addresses a breadth of contexts and operates across a range of robots—remains a distant but worthwhile goal. Consider the parallel to graphical user interfaces. Decades of research and practice in human-computer interaction have resulted in an abundance of reusable graphical user interface toolkits that work across some range of devices, and whose users can mix and match elements with some confidence that they will avoid malpractice. Where rigid prescriptions aren't appropriate, accompanying documentation, like Apple's Human Interface Guidelines [232], provide simple and legible guidance. These libraries make it possible for anyone to create a competent interface and make excellence more attainable to non-experts. The human-robot interaction endeavor has not, thus far, produced anything comparable. If nothing else, this dissertation points us towards this future; where we can expect robots to serve humanity through clear communication, and where the all-too-frequent metal veil between machine behavior and human comprehension dissolves into thoughtful, deliberate transparency.

Bibliography

[1] J. Y. Chen, K. Procci, M. Boyce, J. L. Wright, A. Garcia, and M. Barnes, "Situation awareness-based agent transparency," Tech. Rep., 2014.

[2] T. Hellström and S. Bensch, *Paladyn, Journal of Behavioral Robotics*, vol. 9, no. 1, pp. 110–123, 2018. DOI: doi:10.1515/pjbr-2018-0009.

[3] R. H. Wortham and A. Theodorou, "Robot transparency, trust and utility," *Connection Science*, vol. 29, no. 3, pp. 242–248, 2017. DOI: 10.1080/09540091.2017.1313816.

[4] F. Rajabiyazdi and G. A. Jamieson, "A Review of Transparency (seeing-into) Models," in *2020 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, Toronto, ON, Canada: IEEE, Oct. 2020, pp. 302–308, ISBN: 978-1-72818-526-2. DOI: 10.1109/SMC42975.2020.9282970. (visited on 06/25/2024).

[5] A. F. T. Winfield *et al.*, "IEEE P7001: A Proposed Standard on Transparency," *Frontiers in Robotics and AI*, vol. 8, p. 665 729, Jul. 2021, ISSN: 2296-9144. DOI: 10.3389/frobt.2021.665729. (visited on 06/25/2024).

[6] J. Scholtz, "Theory and evaluation of human robot interactions," in *36th Annual Hawaii International Conference on System Sciences, 2003. Proceedings of The*, Big Island, HI, USA: IEEE, 2003, 10 pp. ISBN: 978-0-7695-1874-9. DOI: 10.1109/HICSS.2003.1174284. (visited on 07/25/2024).

[7] P. A. Chalmers, "The role of cognitive theory in human–computer interface," *Computers in Human Behavior*, vol. 19, no. 5, pp. 593–607, 2003, ISSN: 0747-5632. DOI: 10.1016/S0747-5632(02)00086-9.

[8] R. H. Wortham and A. Theodorou, "Robot transparency, trust and utility," *Connection Science*, vol. 29, no. 3, pp. 242–248, 2017.

[9] X. J. Yang, V. V. Unhelkar, K. Li, and J. A. Shah, "Evaluating effects of user experience and system transparency on trust in automation," in *Proceedings of the 2017 ACM/IEEE International Conference on Human-Robot Interaction*, ser. HRI '17, Vienna, Austria: Association for Computing Machinery, 2017, pp. 408–416, ISBN: 9781450343367. DOI: 10.1145/2909824.3020230. [Online]. Available: https://doi.org/10.1145/2909824.3020230.

[10] H. Felzmann, E. Fosch-Villaronga, C. Lutz, and A. Tamo-Larrieux, "Robots and transparency: The multiple dimensions of transparency in the context of robot technologies," *IEEE Robotics & Automation Magazine*, vol. 26, no. 2, pp. 71–78, 2019. DOI: 10.1109/MRA.2019.2904644.

[11] A. Weller, "Challenges for transparency," in *ICML Workshop on Human Interpretability in Machine Learning (WHI 2017)*, 2017.

[12] J. D. Lee and K. A. See, "Trust in automation: Designing for appropriate reliance," *Human factors*, vol. 46, no. 1, pp. 50–80, 2004. DOI: 10.1518/hfes.46.1.50_30392.

[13] J. M. Beer, A. Prakash, T. L. Mitzner, and W. A. Rogers, "Understanding robot acceptance," Georgia Institute of Technology, Tech. Rep., 2011.

[14] N. Walker, Y. Jiang, M. Cakmak, and P. Stone, "Desiderata for planning systems in general-purpose service robots," in *Proceedings of 2019 ICAPS Workshop on Planning and Robotics*, Berkeley, Jul. 2019.

[15] R. Ghzouli, T. Berger, E. B. Johnsen, A. Wasowski, and S. Dragule, " Behavior Trees and State Machines in Robotics Applications," *IEEE Transactions on Software Engineering*, vol. 49, no. 09, pp. 4243–4267, Sep. 2023, ISSN: 1939-3520. DOI: 10.1109/TSE.2023.3269081. [Online]. Available: https://doi.ieeecomputersociety.org/10.1109/TSE.2023.3269081.

[16] A. Bobu, A. Peng, P. Agrawal, J. A. Shah, and A. D. Dragan, "Aligning human and robot representations," in *Proceedings of the 2024 ACM/IEEE International Conference on Human-Robot Interaction*, ser. HRI '24, Boulder, CO, USA: Association for Computing Machinery, 2024, pp. 42–54, ISBN: 9798400703225. DOI: 10.1145/3610977.3634987.

[17] B. Hayes and J. A. Shah, "Improving robot controller transparency through autonomous policy explanation," in *2017 12th ACM/IEEE International Conference on Human-Robot Interaction (HRI*, 2017, pp. 303–312.

[18] D. Das, S. Banerjee, and S. Chernova, "Explainable ai for robot failures: Generating explanations that improve user assistance in fault recovery," in *Proceedings of the 2021 ACM/IEEE International Conference on Human-Robot Interaction*, ser. HRI '21, Boulder, CO, USA: Association for Computing Machinery, 2021, pp. 351–360, ISBN: 9781450382892. DOI: 10.1145/3434073.3444657.

[19] Z. Han, E. Phillips, and H. A. Yanco, "The need for verbal robot explanations and how people would like a robot to explain itself," *J. Hum.-Robot Interact.*, vol. 10, no. 4, Sep. 2021. DOI: 10.1145/3469652.

[20] Y. Jiang, N. Walker, J. Hart, and P. Stone, "Open-world reasoning for service robots," in *Proceedings of the 29th International Conference on Automated Planning and Scheduling (ICAPS 2019)*, Berkeley, Jul. 2019.

[21] T. Miller, "Explanation in artificial intelligence: Insights from the social sciences," *Artificial Intelligence*, vol. 267, pp. 1–38, 2019. DOI: 10.1016/j.artint.2018.07.007.

[22] T. Chakraborti, S. Sreedharan, Y. Zhang, and S. Kambhampati, “Plan explanations as model reconciliation: Moving beyond explanation as soliloquy,” in *Proceedings of the 26th International Joint Conference on Artificial Intelligence*, ser. IJCAI’17, Melbourne, Australia: AAAI Press, 2017, pp. 156–163, isbn: 9780999241103.

[23] S. H. Huang, D. Held, P. Abbeel, and A. D. Dragan, “Enabling robots to communicate their objectives,” *Autonomous Robots*, vol. 43, no. 2, pp. 309–326, 2019.

[24] S. Tellex, R. Knepper, A. Li, D. Rus, and N. Roy, “Asking for help using inverse semantics,” in *Proceedings of Robotics: Science and Systems*, Berkeley, USA, Jul. 2014. doi: 10.15607/RSS.2014.X.024.

[25] R. A. Knepper, S. Tellex, A. Li, N. Roy, and D. Rus, “Recovering from failure by asking for help,” *Autonomous Robots*, vol. 39, no. 3, pp. 347–362, 2015. doi: 10.1007/s10514-015-9460-1.

[26] M. Murray *et al.*, “Learning backchanneling behaviors for a social robot via data augmentation from human-human conversations,” in *Proceedings of the 5th Conference on Robot Learning*, A. Faust, D. Hsu, and G. Neumann, Eds., ser. Proceedings of Machine Learning Research, vol. 164, London, UK: PMLR, Aug. 2021, pp. 513–525.

[27] A. Dragan and S. Srinivasa, “Generating legible motion,” in *Proceedings of Robotics: Science and Systems*, Berlin, Germany, Jun. 2013. doi: 10.15607/RSS.2013.IX.024.

[28] F. Delaunay, J. de Greeff, and T. Belpaeme, “Towards retro-projected robot faces: An alternative to mechatronic and android faces,” in *RO-MAN 2009 - The 18th IEEE International Symposium on Robot and Human Interactive Communication*, 2009, pp. 306–311. doi: 10.1109/ROMAN.2009.5326314.

[29] Z. Makhataeva and H. A. Varol, “Augmented reality for robotics: A review,” *Robotics*, vol. 9, no. 2, p. 21, 2020. doi: 10.3390/robotics9020021.

[30] T. Matsumaru, “Mobile robot with preliminary-announcement and display function of forthcoming motion using projection equipment,” in *ROMAN 2006 - The 15th IEEE International Symposium on Robot and Human Interactive Communication*, 2006, pp. 443–450. doi: 10.1109/ROMAN.2006.314368.

[31] R. T. Chadalavada, H. Andreasson, R. Krug, and A. J. Lilienthal, “That’s on my mind! robot to human intention communication through on-board projection on shared floor space,” in *2015 European Conference on Mobile Robots (ECMR)*, 2015, pp. 1–6. doi: 10.1109/ECMR.2015.7403771.

[32] C. Vogel, M. Poggendorf, C. Walter, and N. Elkmann, “Towards safe physical human-robot collaboration: A projection-based safety system,” in *2011 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2011, pp. 3355–3360. doi: 10.1109/IROS.2011.6094550.

[33] S. Omidshafiei *et al.*, "Measurable augmented reality for prototyping cyberphysical systems: A robotics platform to aid the hardware prototyping and performance testing of algorithms," *IEEE Control Systems Magazine*, vol. 36, no. 6, pp. 65–87, 2016. DOI: 10.1109/MCS.2016.2602090.

[34] R. D. Patterson, "Auditory warning sounds in the work environment," *Philosophical Transactions of the Royal Society of London. B, Biological Sciences*, vol. 327, no. 1241, pp. 485–492, 1990. DOI: 10.1098/rstb.1990.0091.

[35] N. A. Stanton, *Human Factors in Alarm Design*. CRC Press, 1994.

[36] B. J. Zhang, N. Stargu, S. Brimhall, L. Chan, J. Fick, and N. T. Fitter, "Bringing wall-e out of the silver screen: Understanding how transformative robot sound affects human perception," in *2021 IEEE International Conference on Robotics and Automation (ICRA)*, 2021, pp. 3801–3807. DOI: 10.1109/ICRA48506.2021.9562082.

[37] S. Rosenthal, S. P. Selvaraj, and M. M. Veloso, "Verbalization: Narration of autonomous robot experience," in *International Joint Conference on Artificial Intelligence*, vol. 16, 2016, pp. 862–868.

[38] K. Fischer and R. Moratz, "From communicative strategies to cognitive modelling," in *Workshop Epigenetic Robotics*, 2001.

[39] B. Chen and Z. M. Jiang, "Characterizing and detecting anti-patterns in the logging code," in *2017 IEEE/ACM 39th International Conference on Software Engineering (ICSE)*, 2017, pp. 71–81. DOI: 10.1109/ICSE.2017.15.

[40] H. Felzmann, E. Fosch-Villaronga, C. Lutz, and A. Tamò-Larrieux, "Towards Transparency by Design for Artificial Intelligence," *Science and Engineering Ethics*, vol. 26, no. 6, pp. 3333–3361, Dec. 2020, ISSN: 1353-3452, 1471-5546. DOI: 10.1007/s11948-020-00276-4. (visited on 06/25/2024).

[41] C. D. Wickens, "Multiple resources and mental workload," *Human Factors*, vol. 50, no. 3, pp. 449–455, 2008. DOI: 10.1518/001872008X288394.

[42] M. R. Endsley, S. J. Selcon, T. D. Hardiman, and D. G. Croft, "A comparative analysis of SAGAT and SART for evaluations of situation awareness," in *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*, SAGE Publications Sage CA: Los Angeles, CA, vol. 42, 1998, pp. 82–86.

[43] M. Cakmak and M. Lopes, "Algorithmic and Human Teaching of Sequential Decision Tasks," in *Proceedings of the 26th AAAI Conference on Artificial Intelligence*, AAAI Press, 2012, pp. 1536–1542, ISBN: 978-1-57735-568-7.

[44] S. H. Huang, "Optimizing for robot transparency," Ph.D. dissertation, University of California, Berkeley, 2019. [Online]. Available: https://escholarship.org/uc/item/5q20h9cs.

[45] M. R. Endsley, "Design and evaluation for situation awareness enhancement," in *Proceedings of the Human Factors Society Annual Meeting*, Sage Publications Sage CA: Los Angeles, CA, vol. 32, 1988, pp. 97–101.

[46] M. R. Endsley, "Situation awareness global assessment technique (SAGAT)," in *Proceedings of the IEEE 1988 National Aerospace and Electronics Conference*, IEEE, 1988, pp. 789–795.

[47] R. M. Taylor, "Situational awareness rating technique (SART): The development of a tool for aircrew systems design," in *Situational Awareness in Aerospace Operations*, Neuilly-Sur-Seine, France: Routledge, 1990, pp. 3/1–3/17.

[48] K. A. Roundtree, M. A. Goodrich, and J. A. Adams, "Transparency: Transitioning from Human–Machine systems to human-swarm systems," *Journal of Cognitive Engineering and Decision Making*, vol. 13, pp. 171–195, 2019.

[49] N. Walker, K. Weatherwax, J. Alchin, L. Takayama, and M. Cakmak, "Human perceptions of a curious robot that performs off-task actions," in *Proceedings of the 2020 ACM/IEEE International Conference on Human-Robot Interaction (HRI)*, Oxford, UK, Mar. 2020. doi: 10.1145/3319502.3374821.

[50] C. Kidd and B. Y. Hayden, "The psychology and neuroscience of curiosity," *Neuron*, vol. 88, no. 3, pp. 449–460, 2015. doi: 10.1016/j.neuron.2015.09.010.

[51] G. Loewenstein, "The psychology of curiosity: A review and reinterpretation.," *Psychological bulletin*, vol. 116, no. 1, pp. 75–98, 1994.

[52] R. M. Ryan and E. L. Deci, "Intrinsic and extrinsic motivations: Classic definitions and new directions," *Contemporary Educational Psychology*, vol. 25, no. 1, pp. 54–67, 2000, issn: 0361-476X. doi: 10.1006/ceps.1999.1020.

[53] D. E. Berlyne, "A theory of human curiosity," *British Journal of Psychology. General Section*, vol. 45, no. 3, pp. 180–191, 1954.

[54] F. Kaplan and P.-Y. Oudeyer, "In search of the neural circuits of intrinsic motivation," *Frontiers in neuroscience*, vol. 1, p. 17, 2007. doi: 10.3389/neuro.01.1.1.017.2007.

[55] A. E. Stahl and L. Feigenson, "Observing the unexpected enhances infants' learning and exploration," *Science*, vol. 348, no. 6230, pp. 91–94, 2015. doi: 10.1126/science.aaa3799.

[56] O. C. Robinson, J. D. Demetre, and J. A. Litman, "Adult life stage and crisis as predictors of curiosity and authenticity: Testing inferences from erikson's lifespan theory," *International Journal of Behavioral Development*, vol. 41, no. 3, pp. 426–431, 2017. doi: 10.1177/0165025416645201.

[57] P. E. Shah, H. M. Weeks, B. Richards, and N. Kaciroti, "Early childhood curiosity and kindergarten reading and math academic achievement," *Pediatric research*, vol. 84, no. 3, pp. 380–386, 2018. doi: 10.1038/s41390-018-0039-3.

[58] T. G. Reio Jr and A. Wiswell, "Field investigation of the relationship among adult curiosity, workplace learning, and job performance," *Human resource development quarterly*, vol. 11, no. 1, pp. 5–30, 2000. DOI: 10.1002/1532-1096(200021)11:1<5::AID-HRDQ2>3.0.CO;2-A.

[59] P. Y. Oudeyer and F. Kaplan, "What is intrinsic motivation? A typology of computational approaches," *Frontiers in Neurorobotics*, 2009, ISSN: 16625218. DOI: 10.3389/neuro.12.006.2007.

[60] P.-Y. Oudeyer, "Computational theories of curiosity-driven learning," in *The New Science of Curiosity*, G. Gordon, Ed., 2019, ch. 3, pp. 43–72.

[61] S. Forestier and P. Oudeyer, "Curiosity-driven development of tool use precursors: A computational model," in *Proceedings of the 38th Annual Meeting of the Cognitive Science Society, Recognizing and Representing Events, CogSci 2016*, A. Papafragou, D. Grodner, D. Mirman, and J. C. Trueswell, Eds., cognitivesciencesociety.org, 2016, ISBN: 978-0-9911967-3-9. [Online]. Available: https://mindmodeling.org/cogsci2016/papers/0325/index.html.

[62] P.-Y. Oudeyer, F. Kaplan, and V. V. Hafner, "Intrinsic motivation systems for autonomous mental development," *IEEE transactions on evolutionary computation*, vol. 11, no. 2, pp. 265–286, 2007. DOI: 10.1109/TEVC.2006.890271.

[63] C. Moulin-Frier and P.-Y. Oudeyer, "Curiosity-driven phonetic learning," in *2012 IEEE International Conference on Development and Learning and Epigenetic Robotics (ICDL)*, IEEE, 2012, pp. 1–8. DOI: 10.1109/DevLrn.2012.6400583.

[64] M. Lopes *et al.*, "Active learning and intrinsically motivated exploration in robots: Advances and challenges," *IEEE Transactions on Autonomous Mental Development*, vol. 2, no. 2, pp. 65–69, 2010. DOI: 10.1109/TAMD.2010.2052419.

[65] J. Gottlieb, P.-Y. Oudeyer, M. Lopes, and A. Baranes, "Information-seeking, curiosity, and attention: Computational and neural mechanisms," *Trends in cognitive sciences*, vol. 17, no. 11, pp. 585–593, 2013. DOI: 10.1016/j.tics.2013.09.001.

[66] H. Ngo, M. Luciw, A. Forster, and J. Schmidhuber, "Learning skills from play: Artificial curiosity on a katana robot arm," in *The 2012 International Joint Conference on Neural Networks (IJCNN)*, IEEE, 2012, pp. 1–8. DOI: 10.1109/IJCNN.2012.6252824.

[67] C. Colas, P.-Y. Oudeyer, O. Sigaud, P. Fournier, and M. Chetouani, "Curious: Intrinsically motivated modular multi-goal reinforcement learning," in *International Conference on Machine Learning*, 2019, pp. 1331–1340.

[68] Y. Burda, H. Edwards, D. Pathak, A. Storkey, T. Darrell, and A. A. Efros, "Large-scale study of curiosity-driven learning," in *ICLR*, 2019.

[69] D. Pathak, P. Agrawal, A. A. Efros, and T. Darrell, "Curiosity-driven exploration by self-supervised prediction," in *International Conference on Machine Learning (ICML)*, vol. 2017, 2017.

[70] E. Ugur, M. R. Dogar, M. Cakmak, and E. Sahin, "Curiosity-driven learning of traversability affordance on a mobile robot," in *IEEE 6th International Conference on Development and Learning (ICDL 2007)*, IEEE, 2007, pp. 13–18. doi: 10.1109/DEVLRN.2007.4354044.

[71] A. Baranes and P.-Y. Oudeyer, "Intrinsically motivated goal exploration for active motor learning in robots: A case study," in *2010 IEEE/RSJ International Conference on Intelligent Robots and Systems*, IEEE, 2010, pp. 1766–1773. doi: 10.1109/IROS.2010.5651385.

[72] A. Stout and A. G. Barto, "Competence progress intrinsic motivation," in *2010 IEEE 9th International Conference on Development and Learning*, IEEE, 2010, pp. 257–262. doi: 10.1109/DEVLRN.2010.5578835.

[73] S. Chernova and M. Veloso, "Interactive policy learning through confidence-based autonomy," *Journal of Artificial Intelligence Research*, vol. 34, pp. 1–25, 2009. doi: 10.5555/1622716.1622717.

[74] M. Ogino, M. Kikuchi, and M. Asada, "How can humanoid acquire lexicon?-active approach by attention and learning biases based on curiosity," in *2006 IEEE/RSJ International Conference on Intelligent Robots and Systems*, IEEE, 2006, pp. 3480–3485. doi: 10.1109/IROS.2006.282590.

[75] Y. Girdhar and G. Dudek, "Exploring underwater environments with curiosity," in *2014 Canadian Conference on Computer and Robot Vision (CRV)*, IEEE, 2014, pp. 104–110. doi: 10.1109/CRV.2014.22.

[76] J. Modayil, P. M. Pilarski, A. White, T. Degris, and R. S. Sutton, "Off-policy knowledge maintenance for robots," in *Proceedings of Robotics Science and Systems Workshop (Towards Closing the Loop: Active Learning for Robotics)*, vol. 55, 2010.

[77] A. White, J. Modayil, and R. S. Sutton, "Surprise and curiosity for big data robotics," in *Workshops at the Twenty-Eighth AAAI Conference on Artificial Intelligence*, 2014.

[78] M. Shiomi, T. Kanda, I. Howley, K. Hayashi, and N. Hagita, "Can a social robot stimulate science curiosity in classrooms?" *International Journal of Social Robotics*, vol. 7, no. 5, pp. 641–652, 2015. doi: 10.1007/s12369-015-0303-1.

[79] G. Gordon, C. Breazeal, and S. Engel, "Can children catch curiosity from a social robot?" In *Proceedings of the Tenth Annual ACM/IEEE International Conference on Human-Robot Interaction*, ACM, 2015, pp. 91–98. doi: 10.1145/2696454.2696469.

[80] F. Tanaka and S. Matsuzoe, "Children teach a care-receiving robot to promote their learning: Field experiments in a classroom for vocabulary learning," *Journal of Human-Robot Interaction*, vol. 1, no. 1, pp. 78–95, 2012. doi: 10.5898/JHRI.1.1.Tanaka.

[81] G. Hoffman and W. Ju, "Designing robots with movement in mind," *Journal of Human-Robot Interaction*, vol. 3, no. 1, pp. 91–122, 2014. DOI: 10.5898/JHRI.3.1.Hoffman.

[82] B. Reeves and C. I. Nass, *The media equation: How people treat computers, television, and new media like real people and places.* Cambridge university press, 1996.

[83] R. Simmons *et al.*, "Believable robot characters," *AI Magazine*, vol. 32, no. 4, pp. 39–52, 2011. DOI: 10.1609/aimag.v32i4.2383.

[84] L. Takayama, D. Dooley, and W. Ju, "Expressing thought: Improving robot readability with animation principles," in *Proceedings of the ACM/IEEE International Conference on Human-Robot Interaction (HRI)*, 2011, pp. 69–76. DOI: 10.1145/1957656.1957674.

[85] S. M. Anzalone, S. Boucenna, S. Ivaldi, and M. Chetouani, "Evaluating the engagement with social robots," *International Journal of Social Robotics*, vol. 7, no. 4, pp. 465–478, 2015. DOI: 10.1007/s12369-015-0298-7.

[86] C. Breazeal, C. D. Kidd, A. L. Thomaz, G. Hoffman, and M. Berlin, "Effects of nonverbal communication on efficiency and robustness in human-robot teamwork," in *2005 IEEE/RSJ international conference on intelligent robots and systems*, IEEE, 2005, pp. 708–713. DOI: 10.1109/IROS.2005.1545011.

[87] V. Chidambaram, Y.-H. Chiang, and B. Mutlu, "Designing persuasive robots: How robots might persuade people using vocal and nonverbal cues," in *Proceedings of the seventh annual ACM/IEEE international conference on Human-Robot Interaction*, ACM, 2012, pp. 293–300. DOI: 10.1145/2157689.2157798.

[88] J. Ham, M. van Esch, Y. Limpens, J. de Pee, J.-J. Cabibihan, and S. S. Ge, "The automaticity of social behavior towards robots: The influence of cognitive load on interpersonal distance to approachable versus less approachable robots," in *International Conference on Social Robotics*, Springer, 2012, pp. 15–25. DOI: 10.1007/978-3-642-34103-8_2.

[89] M. K. Lee, S. Kiesler, J. Forlizzi, S. Srinivasa, and P. Rybski, "Gracefully mitigating breakdowns in robotic services," in *2010 5th ACM/IEEE International Conference on Human-Robot Interaction (HRI)*, IEEE, 2010, pp. 203–210. DOI: 10.1109/HRI.2010.5453195.

[90] J. Ceha *et al.*, "Expression of curiosity in social robots: Design, perception, and effects on behaviour," in *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*, ser. CHI '19, Glasgow, Scotland Uk: Association for Computing Machinery, 2019, ISBN: 9781450359702. DOI: 10.1145/3290605.3300636.

[91] E. M. Grossnickle, "Disentangling curiosity: Dimensionality, definitions, and distinctions from interest in educational contexts," *Educational Psychology Review*, vol. 28, no. 1, pp. 23–60, 2016. DOI: 10.1007/s10648-014-9294-y.

[92] T. Ribeiro and A. Paiva, "The illusion of robotic life: Principles and practices of animation for robots," in *2012 7th ACM/IEEE International Conference on Human-Robot Interaction (HRI)*, IEEE, 2012, pp. 383–390. DOI: 10.1145/2157689.2157814.

[93] G. A. Hollinger *et al.*, "Underwater data collection using robotic sensor networks," *IEEE Journal on Selected Areas in Communications*, vol. 30, no. 5, pp. 899–911, 2012. DOI: 10.1109/JSAC.2012.120606.

[94] M. J.-Y. Chung, A. Pronobis, M. Cakmak, D. Fox, and R. P. Rao, "Designing information gathering robots for human-populated environments," in *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, IEEE, 2015, pp. 5755–5762. DOI: 10.1109/IROS.2015.7354194.

[95] T. Kashdan *et al.*, "The five-dimensional curiosity scale: Capturing the bandwidth of curiosity and identifying four unique subgroups of curious people," *Journal of Research in Personality*, vol. 73, pp. 130–149, Dec. 2017. DOI: 10.1016/j.jrp.2017.11.011.

[96] C. Bartneck, D. Kulić, E. Croft, and S. Zoghbi, "Measurement instruments for the anthropomorphism, animacy, likeability, perceived intelligence, and perceived safety of robots," *International Journal of Social Robotics*, vol. 1, no. 1, pp. 71–81, Jan. 2009, ISSN: 1875-4805. DOI: 10.1007/s12369-008-0001-3.

[97] C. M. Carpinella, A. B. Wyman, M. A. Perez, and S. J. Stroessner, "The robotic social attributes scale (rosas): Development and validation," in *Proceedings of the 2017 ACM/IEEE International Conference on Human-Robot Interaction*, ser. HRI '17, Vienna, Austria: ACM, 2017, pp. 254–262, ISBN: 978-1-4503-4336-7. DOI: 10.1145/2909824.3020208.

[98] R. Vallat, "Pingouin: Statistics in python," *The Journal of Open Source Software*, vol. 3, no. 31, p. 1026, Nov. 2018. DOI: 10.21105/joss.01026.

[99] J.-Y. Sung, L. Guo, R. E. Grinter, and H. I. Christensen, ""my roomba is rambo": Intimate home appliances," in *UbiComp 2007: Ubiquitous Computing*, J. Krumm, G. D. Abowd, A. Seneviratne, and T. Strang, Eds., Berlin, Heidelberg: Springer Berlin Heidelberg, 2007, pp. 145–162. DOI: 10.1007/978-3-540-74853-3_9.

[100] M. Saerbeck and C. Bartneck, "Perception of affect elicited by robot motion," in *Proceedings of the ACM/IEEE International Conference on Human-Robot Interaction (HRI)*, 2010, pp. 53–60. DOI: 10.1109/HRI.2010.5453269.

[101] M. Kwon, M. F. Jung, and R. A. Knepper, "Human expectations of social robots," in *Proceedings of the ACM/IEEE International Conference on Human-Robot Interaction (HRI)*, 2016, pp. 463–464. DOI: 10.1109/HRI.2016.7451807.

[102] E. Cha, A. D. Dragan, and S. S. Srinivasa, "Perceived robot capability," in *Proceedings of the IEEE International Symposium on Robot and Human Interactive Communication (RO-MAN)*, 2015, pp. 541–548. DOI: 10.1109/ROMAN.2015.7333656.

[103] F. Zacharias, C. Schlette, F. Schmidt, C. Borst, J. Rossmann, and G. Hirzinger, "Making planned paths look more human-like in humanoid robot manipulation planning," in *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, 2011, pp. 1192–1198. DOI: 10.1109/ICRA.2011.5979553.

[104] F. Heider, *The psychology of interpersonal relations.* John Wiley & Sons Inc., 1958.

[105] B. F. Malle, "Attribution theories: How people make sense of behavior," in *Theories in Social Psychology*, D. Chadee, Ed., Wiley Blackwell, 2011, pp. 72–95.

[106] L. Ross, "The intuitive psychologist and his shortcomings: Distortions in the attribution process," in ser. Advances in Experimental Social Psychology, L. Berkowitz, Ed., vol. 10, Academic Press, 1977, pp. 173–220.

[107] E. Goffman, *The Presentation of Self in Everyday Life.* Anchor, 1959.

[108] N. Walker, C. Mavrogiannis, S. Srinivasa, and M. Cakmak, "Influencing behavioral attributions to robot motion during task execution," in *Proceedings of the 5th Conference on Robot Learning*, A. Faust, D. Hsu, and G. Neumann, Eds., ser. Proceedings of Machine Learning Research, vol. 164, London, UK: PMLR, Aug. 2021, pp. 169–179.

[109] A. Dragan and S. Srinivasa, "Integrating human observer inferences into robot motion planning," *Autonomous Robots*, vol. 37, no. 4, pp. 351–368, 2014. DOI: 10.1007/s10514-014-9408-x.

[110] R. A. Knepper, C. Mavrogiannis, J. Proft, and C. Liang, "Implicit communication in a joint action," in *Proceedings of the ACM/IEEE International Conference on Human-Robot Interaction (HRI)*, 2017, pp. 283–292. DOI: 10.1145/2909824.3020226.

[111] D. Carton, W. Olszowy, and D. Wollherr, "Measuring the effectiveness of readability for mobile robot locomotion," *International Journal of Social Robotics*, vol. 8, no. 5, pp. 721–741, 2016. DOI: 10.1007/s12369-016-0358-7.

[112] M. Kwon, S. H. Huang, and A. D. Dragan, "Expressing robot incapability," in *Proceedings of the ACM/IEEE International Conference on Human-Robot Interaction (HRI)*, 2018, pp. 87–95. DOI: 10.1145/3171221.3171276.

[113] C. Mavrogiannis, A. M. Hutchinson, J. Macdonald, P. Alves-Oliveira, and R. A. Knepper, "Effects of distinct robotic navigation strategies on human behavior in a crowded environment," in *Proceedings of the ACM/IEEE International Conference on Human-Robot Interaction (HRI)*, 2019, pp. 421–430. DOI: 10.1109/HRI.2019.8673115.

[114] S. H. Huang, D. Held, P. Abbeel, and A. D. Dragan, "Enabling robots to communicate their objectives," *Autonomous Robots*, vol. 43, no. 2, pp. 309–326, 2019. DOI: doi.org/10.1007/s10514-018-9771-0.

[115] H. Knight and R. Simmons, "Expressive motion with x, y and theta: Laban effort features for mobile robots," in *Proceedings of the IEEE International Symposium on Robot and Human Interactive Communication (RO-MAN)*, 2014, pp. 267–273. DOI: 10.1109/ROMAN.2014.6926264.

[116] M. Brand and A. Hertzmann, "Style machines," in *Proceedings of the Annual Conference on Computer Graphics and Interactive Techniques (SIGGRAPH)*, ACM Press/Addison-Wesley Publishing Co., 2000, pp. 183–192. DOI: 10.1145/344779.344865.

[117] S. J. Guy, S. Kim, M. C. Lin, and D. Manocha, "Simulating heterogeneous crowd behaviors using personality trait theory," in *Proceedings of the ACM SIGGRAPH/Eurographics Symposium on Computer Animation*, 2011, pp. 43–52. DOI: 10.1145/2019406.2019413.

[118] T. Kanda, H. Ishiguro, and T. Ishida, "Psychological analysis on human-robot interaction," in *Proceedings IEEE International Conference on Robotics and Automation (ICRA)*, vol. 4, 2001, 4166–4173 vol.4. DOI: 10.1109/ROBOT.2001.933269.

[119] S. Lo, K. Yamane, and K. Sugiyama, "Perception of pedestrian avoidance strategies of a self-balancing mobile robot," in *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2019, pp. 1243–1250. DOI: 10.1109/IROS40897.2019.8968191.

[120] C. Mavrogiannis, W. B. Thomason, and R. A. Knepper, "Social momentum: A framework for legible navigation in dynamic multi-agent environments," in *Proceedings of the ACM/IEEE International Conference on Human-Robot Interaction (HRI)*, 2018, pp. 361–369. DOI: 10.1145/3171221.3171255.

[121] iRobot, *Investor Presentation.* Aug. 2021, Accessed: 2021-11-07. [Online]. Available: https://investor.irobot.com/static-files/a6147f70-f50a-43d3-9161-9af57981ea0f.

[122] C. Bartneck, E. Croft, and D. Kulic, "Measurement instruments for the anthropomorphism, animacy, likeability, perceived intelligence, and perceived safety of robots," *International Journal of Social Robotics*, vol. 1, no. 1, pp. 71–81, 2009. DOI: 10.1007/s12369-008-0001-3.

[123] C. M. Carpinella, A. B. Wyman, M. A. Perez, and S. J. Stroessner, "The robotic social attributes scale (rosas): Development and validation," in *Proceedings of the ACM/IEEE International Conference on Human-Robot Interaction (HRI*, 2017, pp. 254–262. DOI: 10.1145/2909824.3020208.

[124] C. Bishop, "Mixture density networks," Aston University, Tech. Rep. NCRG/94/004, 1994.

[125] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *CoRR*, vol. abs/1412.6980, 2015.

[126] J. Rothfuss *et al.*, "Noise regularization for conditional density estimation," *arXiv:1907.08982*, 2019.

[127] A. Mandlekar *et al.*, "Roboturk: A crowdsourcing platform for robotic skill learning through imitation," in *Proceedings of The 2nd Conference on Robot Learning*, A. Billard, A. Dragan, J. Peters, and J. Morimoto, Eds., ser. Proceedings of Machine Learning Research, vol. 87, PMLR, Oct. 2018, pp. 879–893.

[128] O. Mees, L. Hermann, E. Rosete-Beas, and W. Burgard, "Calvin: A benchmark for language-conditioned policy learning for long-horizon robot manipulation tasks," *IEEE Robotics and Automation Letters*, vol. 7, no. 3, pp. 7327–7334, 2022. DOI: 10.1109/LRA.2022.3180108.

[129] M. Shridhar, L. Manuelli, and D. Fox, "Cliport: What and where pathways for robotic manipulation," in *Proceedings of the 5th Conference on Robot Learning*, A. Faust, D. Hsu, and G. Neumann, Eds., ser. Proceedings of Machine Learning Research, vol. 164, PMLR, Aug. 2022, pp. 894–906.

[130] C. Li *et al.*, "Behavior-1k: A benchmark for embodied ai with 1,000 everyday activities and realistic simulation," in *Proceedings of The 6th Conference on Robot Learning*, K. Liu, D. Kulic, and J. Ichnowski, Eds., ser. Proceedings of Machine Learning Research, vol. 205, PMLR, Dec. 2023, pp. 80–93.

[131] A. Brohan *et al.*, "RT-1: Robotics Transformer for Real-World Control at Scale," in *Robotics: Science and Systems (RSS)*, 2023.

[132] B. Zitkovich *et al.*, "Rt-2: Vision-language-action models transfer web knowledge to robotic control," in *Proceedings of The 7th Conference on Robot Learning*, J. Tan, M. Toussaint, and K. Darvish, Eds., ser. Proceedings of Machine Learning Research, vol. 229, PMLR, Jun. 2023, pp. 2165–2183.

[133] T. Zhang *et al.*, "Deep imitation learning for complex manipulation tasks from virtual reality teleoperation," in *2018 IEEE International Conference on Robotics and Automation (ICRA)*, Brisbane, Australia: IEEE Press, 2018, pp. 1–8. DOI: 10.1109/ICRA.2018.8461249.

[134] C. Wang *et al.*, "Mimicplay: Long-horizon imitation learning by watching human play," in *Proceedings of The 7th Conference on Robot Learning*, J. Tan, M. Toussaint, and K. Darvish, Eds., ser. Proceedings of Machine Learning Research, vol. 229, PMLR, Jun. 2023, pp. 201–221.

[135] E. Rosete-Beas, O. Mees, G. Kalweit, J. Boedecker, and W. Burgard, "Latent plans for task-agnostic offline reinforcement learning," in *Proceedings of The 6th Conference on Robot Learning*, K. Liu, D. Kulic, and J. Ichnowski, Eds., ser. Proceedings of Machine Learning Research, vol. 205, PMLR, Dec. 2023, pp. 1838–1849.

[136] G. Zhou, L. Ke, S. Srinivasa, A. Gupta, A. Rajeswaran, and V. Kumar, "Real world offline reinforcement learning with realistic data source," in *2023 IEEE International Conference on Robotics and Automation (ICRA)*, 2023, pp. 7176–7183. DOI: 10.1109/ICRA48891.2023.10161474.

[137] A. Mandlekar *et al.*, "What matters in learning from offline human demonstrations for robot manipulation," in *Proceedings of the 5th Conference on Robot Learning*, A. Faust, D. Hsu, and G. Neumann, Eds., ser. Proceedings of Machine Learning Research, vol. 164, PMLR, Aug. 2022, pp. 1678–1690.

[138] S. Belkhale, Y. Cui, and D. Sadigh, "Data quality in imitation learning," in *Proceedings of the 37th International Conference on Neural Information Processing Systems*, ser. NIPS '23, New Orleans, LA, USA: Curran Associates Inc., 2024.

[139] Y. Zhu, B. Jiang, Q. Chen, T. Aoyama, and Y. Hasegawa, "A shared control framework for enhanced grasping performance in teleoperation," *IEEE Access*, vol. 11, pp. 69 204–69 215, 2023. DOI: 10.1109/ACCESS.2023.3292410.

[140] Y. Qin *et al.*, "AnyTeleop: A General Vision-Based Dexterous Robot Arm-Hand Teleoperation System," in *Proceedings of Robotics: Science and Systems*, Daegu, Republic of Korea, Jul. 2023. DOI: 10.15607/RSS.2023.XIX.015.

[141] A. D. Dragan and S. S. Srinivasa, "Formalizing teleoperation assistance," in *Robotics: Science and Systems*, 2008. DOI: 10.7551/mitpress/9816.001.0001.

[142] S. Javdani, S. Srinivasa, and A. Bagnell, "Shared autonomy via hindsight optimization," in *Proceedings of Robotics: Science and Systems*, Rome, Italy, Jul. 2015. DOI: 10.15607/RSS.2015.XI.032.

[143] S. Nikolaidis, Y. X. Zhu, D. Hsu, and S. Srinivasa, "Human-robot mutual adaptation in shared autonomy," in *Proceedings of the 2017 ACM/IEEE International Conference on Human-Robot Interaction*, ser. HRI '17, Vienna, Austria: Association for Computing Machinery, 2017, pp. 294–302, ISBN: 9781450343367. DOI: 10.1145/2909824.3020252. [Online]. Available: https://doi.org/10.1145/2909824.3020252.

[144] P. A. Lasota and J. A. Shah, "A multiple-predictor approach to human motion prediction," in *2017 IEEE International Conference on Robotics and Automation (ICRA)*, IEEE, 2017, pp. 2300–2307. DOI: 10.1109/ICRA.2017.7989265.

[145] C. Pérez-D'Arpino and J. A. Shah, "Fast target prediction of human reaching motion for cooperative human-robot manipulation tasks using time series classification," in *2015 IEEE International Conference on Robotics and Automation (ICRA)*, 2015, pp. 6175–6182. DOI: 10.1109/ICRA.2015.7140066.

[146] N. Walker, X. Yang, A. Garg, M. Cakmak, D. Fox, and C. Pérez-D'Arpino, "Fast explicit-input assistance for teleoperation in clutter," in *2024 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Abu Dhabi, UAE, Oct. 2024, pp. 9270–9276. DOI: 10.1109/IROS58592.2024.10802138.

[147] H. M. Clever *et al.*, "Assistive tele-op: Leveraging transformers to collect robotic task demonstrations," in *4th NeurIPS Robot Learning Workshop on Self-Supervised and Lifelong Learning*, 2021.

[148] T. Yoneda, L. Sun, G. Yang, B. C. Stadie, and M. R. Walter, "To the Noise and Back: Diffusion for Shared Autonomy," in *Proceedings of Robotics: Science and Systems*, Daegu, Republic of Korea, Jul. 2023. DOI: 10.15607/RSS.2023.XIX.014.

[149] S. Reddy, A. Dragan, and S. Levine, "Shared autonomy via deep reinforcement learning," in *Proceedings of Robotics: Science and Systems*, Pittsburgh, Pennsylvania, Jun. 2018. DOI: 10.15607/RSS.2018.XIV.005.

[150] A. E. Leeper, K. Hsiao, M. Ciocarlie, L. Takayama, and D. Gossow, "Strategies for human-in-the-loop robotic grasping," in *Proceedings of the Seventh Annual ACM/IEEE International Conference on Human-Robot Interaction*, ser. HRI '12, Boston, Massachusetts, USA: Association for Computing Machinery, 2012, pp. 1–8, ISBN: 9781450310635. DOI: 10.1145/2157689.2157691.

[151] C. Pérez-D'Arpino, R. P. Khurshid, and J. A. Shah, "Experimental assessment of human–robot teaming for multi-step remote manipulation with expert operators," *J. Hum.-Robot Interact.*, vol. 13, no. 3, Aug. 2024. DOI: 10.1145/3618258.

[152] D. P. Losey *et al.*, "Learning latent actions to control assistive robots," *Auton. Robots*, vol. 46, no. 1, pp. 115–147, Jan. 2022, ISSN: 0929-5593. DOI: 10.1007/s10514-021-10005-w.

[153] Y. Cui, S. Karamcheti, R. Palleti, N. Shivakumar, P. Liang, and D. Sadigh, "No, to the right: Online language corrections for robotic manipulation via shared autonomy," in *Proceedings of the 2023 ACM/IEEE International Conference on Human-Robot Interaction*, ser. HRI '23, Stockholm, Sweden: Association for Computing Machinery, 2023, pp. 93–101, ISBN: 9781450399647. DOI: 10.1145/3568162.3578623.

[154] M. Rubagotti, T. Taunyazov, B. Omarali, and A. Shintemirov, "Semi-autonomous robot teleoperation with obstacle avoidance via model predictive control," *IEEE Robotics and Automation Letters*, vol. 4, no. 3, pp. 2746–2753, 2019. DOI: 10.1109/LRA.2019.2917707.

[155] L. Rosenberg, "Virtual fixtures: Perceptual tools for telerobotic manipulation," in *Proceedings of IEEE Virtual Reality Annual International Symposium*, 1993, pp. 76–82. DOI: 10.1109/VRAIS.1993.380795.

[156] C. Mazzotti, N. Sancisi, and V. Parenti-Castelli, "A measure of the distance between two rigid-body poses based on the use of platonic solids," in *ROMANSY 21 - Robot Design, Dynamics and Control*, V. Parenti-Castelli and W. Schiehlen, Eds., Cham: Springer International Publishing, 2016, pp. 81–89, ISBN: 978-3-319-33714-2.

[157] M. Macklin, *Warp: A high-performance python framework for gpu simulation and graphics*, https://github.com/nvidia/warp, NVIDIA GPU Technology Conference (GTC), Mar. 2022.

[158] V. Dhat, N. Walker, and M. Cakmak, "Using 3d mice to control robot manipulators," in *Proceedings of the 2024 ACM/IEEE International Conference on Human-Robot Interaction*, ser. HRI '24, Boulder, CO, USA: Association for Computing Machinery, 2024, pp. 896–900, ISBN: 9798400703225. DOI: 10.1145/3610977.3637486.

[159] C.-A. Cheng *et al.*, "Rmpflow: A geometric framework for generation of multitask motion policies," *IEEE Transactions on Automation Science and Engineering*, vol. 18, no. 3, pp. 968–987, 2021. DOI: 10.1109/TASE.2021.3053422.

[160] S. G. Hart and L. E. Staveland, "Development of nasa-tlx (task load index): Results of empirical and theoretical research," in *Human Mental Workload*, P. A. Hancock and N. Meshkati, Eds., North-Holland, 1988, pp. 139–183. doi: 10.1016/S0166-4115(08)62386-9.

[161] Z. Wang *et al.*, "I can tell what i am doing: Toward real-world natural language grounding of robot experiences," in *Proceedings of The 8th Conference on Robot Learning*, P. Agrawal, O. Kroemer, and W. Burgard, Eds., ser. Proceedings of Machine Learning Research, vol. 270, Munich, Germany: PMLR, Nov. 2024, pp. 1863–1890.

[162] N. Walker, Z. Wang, M. Grotz, and M. Cakmak, "Interpretable robot failure attribution to assist remote robot supervisors," Mar. 2025.

[163] L. Bärmann, R. Kartmann, F. Peller-Konrad, J. Niehues, A. Waibel, and T. Asfour, "Incremental learning of humanoid robot behavior from natural interaction and large language models," *Frontiers in Robotics and AI*, vol. 11, 2024, issn: 2296-9144. doi: 10.3389/frobt.2024.1455375. [Online]. Available: https://www.frontiersin.org/journals/robotics-and-ai/articles/10.3389/frobt.2024.1455375.

[164] S. Wang, Z. Zhou, B. Li, Z. Li, and Z. Kan, "Multi-modal interaction with transformers: Bridging robots and human with natural language," *Robotica*, vol. 42, no. 2, pp. 415–434, 2024. doi: 10.1017/S0263574723001510.

[165] F. Zeng, W. Gan, Y. Wang, N. Liu, and P. S. Yu, "Large language models for robotics: A survey," *arXiv preprint arXiv:2311.07226*, 2023.

[166] C. Zhang, J. Chen, J. Li, Y. Peng, and Z. Mao, "Large language models for human–robot interaction: A review," *Biomimetic Intelligence and Robotics*, vol. 3, no. 4, p. 100 131, 2023, issn: 2667-3797. doi: https://doi.org/10.1016/j.birob.2023.100131.

[167] A. Radford *et al.*, "Learning transferable visual models from natural language supervision," in *International conference on machine learning*, PMLR, 2021, pp. 8748–8763.

[168] D. Shah, B. Osiński, b. ichter brian, and S. Levine, "Lm-nav: Robotic navigation with large pre-trained models of language, vision, and action," in *Proceedings of The 6th Conference on Robot Learning*, K. Liu, D. Kulic, and J. Ichnowski, Eds., ser. Proceedings of Machine Learning Research, vol. 205, PMLR, Dec. 2023, pp. 492–504.

[169] D. Driess *et al.*, "Palm-e: An embodied multimodal language model," in *Proceedings of the 40th International Conference on Machine Learning*, ser. ICML'23, Honolulu, Hawaii, USA: JMLR.org, 2023.

[170] J. Yan, Q. Zhang, J. Cheng, Z. Ren, T. Li, and Z. Yang, "Indoor target-driven visual navigation based on spatial semantic information," in *2022 IEEE International Conference on Image Processing (ICIP)*, 2022, pp. 571–575. doi: 10.1109/ICIP46576.2022.9898026.

[171] M. Ahn *et al.*, "Do as i can, not as i say: Grounding language in robotic affordances," *arXiv preprint arXiv:2204.01691*, 2022.

[172] L. Ouyang *et al.*, "Training language models to follow instructions with human feedback," *Advances in neural information processing systems*, vol. 35, pp. 27 730–27 744, 2022.

[173] M. Crosby, M. Rovatsos, and R. Petrick, "Automated agent decomposition for classical planning," in *Proceedings of the International Conference on Automated Planning and Scheduling*, vol. 23, 2013, pp. 46–54.

[174] B. Xu, Z. Peng, B. Lei, S. Mukherjee, Y. Liu, and D. Xu, "Rewoo: Decoupling reasoning from observations for efficient augmented language models," *arXiv preprint arXiv:2305.18323*, 2023.

[175] T. Kojima, S. S. Gu, M. Reid, Y. Matsuo, and Y. Iwasawa, "Large language models are zero-shot reasoners," *Advances in neural information processing systems*, vol. 35, pp. 22 199–22 213, 2022.

[176] S. S. Raman *et al.*, "Cape: Corrective actions from precondition errors using large language models," in *2024 IEEE International Conference on Robotics and Automation (ICRA)*, 2024, pp. 14 070–14 077. DOI: 10.1109/ICRA57147.2024.10611376.

[177] Z. Liu, A. Bahety, and S. Song, "Reflect: Summarizing robot experiences for failure explanation and correction," in *Proceedings of The 7th Conference on Robot Learning*, J. Tan, M. Toussaint, and K. Darvish, Eds., ser. Proceedings of Machine Learning Research, vol. 229, PMLR, Jun. 2023, pp. 3468–3484.

[178] S. Reed *et al.*, "A generalist agent," *Transactions on Machine Learning Research*, 2022.

[179] N. M. Shafiullah, Z. Cui, A. A. Altanzaya, and L. Pinto, "Behavior transformers: Cloning k modes with one stone," *Advances in neural information processing systems*, vol. 35, pp. 22 955–22 968, 2022.

[180] E. Kolve *et al.*, "Ai2-thor: An interactive 3d environment for visual ai," *arXiv preprint arXiv:1712.05474*, 2017.

[181] C. Matuszek, E. Herbst, L. Zettlemoyer, and D. Fox, "Learning to parse natural language commands to a robot control system," in *Experimental Robotics: the 13th International Symposium on Experimental Robotics*, Springer, 2013, pp. 403–415.

[182] D. Chen and R. Mooney, "Learning to interpret natural language navigation instructions from observations," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 25, 2011, pp. 859–865.

[183] O. X.-E. Collaboration, "Open x-embodiment: Robotic learning datasets and rt-x model," in *2024 IEEE International Conference on Robotics and Automation (ICRA)*, 2024, pp. 6892–6903. DOI: 10.1109/ICRA57147.2024.10611477.

[184] J. Arkin *et al.*, "Multimodal estimation and communication of latent semantic knowledge for robust execution of robot instructions," *The International Journal of Robotics Research*, vol. 39, no. 10-11, pp. 1279–1304, 2020.

[185] A. Bucker *et al.*, “Latte: Language trajectory transformer,” in *2023 IEEE International Conference on Robotics and Automation (ICRA)*, 2023, pp. 7287–7294. doi: 10.1109/ICRA48891.2023.10161068.

[186] T. Wang, R. Zhang, Z. Lu, F. Zheng, R. Cheng, and P. Luo, “End-to-end dense video captioning with parallel decoding,” in *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, 2021, pp. 6827–6837. doi: 10.1109/ICCV48922.2021.00677.

[187] C. Deng, S. Chen, D. Chen, Y. He, and Q. Wu, “Sketch, ground, and refine: Top-down dense video captioning,” in *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021, pp. 234–243. doi: 10.1109/CVPR46437.2021.00030.

[188] Q. Zhang, Y. Song, and Q. Jin, “Unifying event detection and captioning as sequence generation via pre-training,” in *Computer Vision – ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part XXXVI*, Tel Aviv, Israel: Springer-Verlag, 2022, pp. 363–379, isbn: 978-3-031-20058-8. doi: 10.1007/978-3-031-20059-5_21.

[189] W. Zhu, B. Pang, A. V. Thapliyal, W. Y. Wang, and R. Soricut, “End-to-end dense video captioning as sequence generation,” in *Proceedings of the 29th International Conference on Computational Linguistics*, 2022, pp. 5651–5665.

[190] A. Yang *et al.*, “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)*, 2023, pp. 10 714–10 726. doi: 10.1109/CVPR52729.2023.01032.

[191] I. Armeni *et al.*, “3d scene graph: A structure for unified semantics, 3d space, and camera,” in *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, 2019, pp. 5663–5672. doi: 10.1109/ICCV.2019.00576.

[192] K. Rana, J. Haviland, S. Garg, J. Abou-Chakra, I. Reid, and N. Suenderhauf, “Sayplan: Grounding large language models using 3d scene graphs for scalable robot task planning,” in *Proceedings of The 7th Conference on Robot Learning*, J. Tan, M. Toussaint, and K. Darvish, Eds., ser. Proceedings of Machine Learning Research, vol. 229, PMLR, Jun. 2023, pp. 23–72.

[193] W. Chen, S. Hu, R. Talak, and L. Carlone, “Leveraging large language models for robot 3d scene understanding,” *arXiv preprint arXiv:2209.05629*, 2022.

[194] B. Chen *et al.*, “Open-vocabulary queryable scene representations for real world planning,” in *2023 IEEE International Conference on Robotics and Automation (ICRA)*, 2023, pp. 11 509–11 522. doi: 10.1109/ICRA48891.2023.10161534.

[195] X. Li, D. Guo, H. Liu, and F. Sun, “Embodied semantic scene graph generation,” in *Proceedings of the 5th Conference on Robot Learning*, A. Faust, D. Hsu, and G. Neumann, Eds., ser. Proceedings of Machine Learning Research, vol. 164, PMLR, Aug. 2022, pp. 1585–1594.

[196] A. Zeng *et al.*, “Socratic models: Composing zero-shot multimodal reasoning with language,” *arXiv preprint arXiv:2204.00598*, 2022.

[197] Z. Wang *et al.*, "Language models with image descriptors are strong few-shot video-language learners," *Advances in Neural Information Processing Systems*, vol. 35, pp. 8483–8497, 2022.

[198] D. Honerkamp, M. Büchner, F. Despinoy, T. Welschehold, and A. Valada, "Language-grounded dynamic scene graphs for interactive object search with mobile manipulation," *IEEE Robotics and Automation Letters*, vol. 9, no. 10, pp. 8298–8305, 2024. DOI: 10.1109/LRA.2024.3441495.

[199] P. Khanna, E. Yadollahi, M. Björkman, I. Leite, and C. Smith, "Effects of explanation strategies to resolve failures in human-robot collaboration," in *2023 32nd IEEE International Conference on Robot and Human Interactive Communication (RO-MAN)*, 2023, pp. 1829–1836. DOI: 10.1109/RO-MAN57019.2023.10309394.

[200] S. Ye, G. Neville, M. Schrum, M. Gombolay, S. Chernova, and A. Howard, "Human trust after robot mistakes: Study of the effects of different forms of robot communication," in *2019 28th IEEE International Conference on Robot and Human Interactive Communication (RO-MAN)*, 2019, pp. 1–7. DOI: 10.1109/RO-MAN46459.2019.8956424.

[201] D. Das and S. Chernova, "Semantic-based explainable ai: Leveraging semantic scene graphs and pairwise ranking to explain robot failures," in *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2021, pp. 3034–3041. DOI: 10.1109/IROS51168.2021.9635890.

[202] M. Diehl and K. Ramirez-Amaro, "Why did i fail? a causal-based method to find explanations for robot failures," *IEEE Robotics and Automation Letters*, vol. 7, no. 4, pp. 8925–8932, 2022. DOI: 10.1109/LRA.2022.3188889.

[203] A. Inceoglu, E. E. Aksoy, A. Cihan Ak, and S. Sariel, "Fino-net: A deep multimodal sensor fusion framework for manipulation failure detection," in *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2021, pp. 6841–6847. DOI: 10.1109/IROS51168.2021.9636455.

[204] W. Huang, C. Wang, R. Zhang, Y. Li, J. Wu, and L. Fei-Fei, "Voxposer: Composable 3d value maps for robotic manipulation with language models," in *Proceedings of The 7th Conference on Robot Learning*, J. Tan, M. Toussaint, and K. Darvish, Eds., ser. Proceedings of Machine Learning Research, vol. 229, PMLR, Jun. 2023, pp. 540–562.

[205] W. Huang, P. Abbeel, D. Pathak, and I. Mordatch, "Language models as zero-shot planners: Extracting actionable knowledge for embodied agents," in *International Conference on Machine Learning*, PMLR, 2022, pp. 9118–9147.

[206] J. Liang *et al.*, "Code as policies: Language model programs for embodied control," in *2023 IEEE International Conference on Robotics and Automation (ICRA)*, IEEE, 2023, pp. 9493–9500. DOI: 10.1109/ICRA48891.2023.10160591.

[207] A. Zeng *et al.*, "Transporter networks: Rearranging the visual world for robotic manipulation," in *Proceedings of the 2020 Conference on Robot Learning*, J. Kober, F. Ramos, and C. Tomlin, Eds., ser. Proceedings of Machine Learning Research, vol. 155, PMLR, Nov. 2021, pp. 726–747.

[208] C. Lynch *et al.*, "Interactive language: Talking to robots in real time," *IEEE Robotics and Automation Letters*, pp. 1–8, 2023. doi: 10.1109/LRA.2023.3295255.

[209] G. Canal, S. Krivić, P. Luff, and A. Coles, "PlanVerb: Domain-Independent Verbalization and Summary of Task Plans," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 36(9), 2022, pp. 9698–9706. doi: 10.1609/aaai.v36i9.21204.

[210] T. Cheng, L. Song, Y. Ge, W. Liu, X. Wang, and Y. Shan, "Yolo-world: Real-time open-vocabulary object detection," in *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024, pp. 16 901–16 911. doi: 10.1109/CVPR52733.2024.01599.

[211] C. C. Kemp, A. Edsinger, H. M. Clever, and B. Matulevich, "The design of stretch: A compact, lightweight mobile manipulator for indoor human environments," in *2022 International Conference on Robotics and Automation (ICRA)*, IEEE, 2022, pp. 3150–3157. doi: 10.1109/ICRA46639.2022.9811922.

[212] A. Celikyilmaz, E. Clark, and J. Gao, "Evaluation of text generation: A survey," *arXiv preprint arXiv:2006.14799*, 2020.

[213] N. Webb *et al.*, "Waymo's Safety Methodologies and Safety Readiness Determinations," *arXiv preprint arXiv:2011.00054*, 2020.

[214] D. Etherington, "Driveu.auto to power remote piloting of easymile's autonomous shuttles, coco's sidewalk robots," *TechCrunch*, 2022.

[215] Plus One Robotics, *On our bet on human-in-the-loop automation*, Blog post, Accessed: 2025-03-01, 2022. [Online]. Available: https://www.plusonerobotics.com/blog/on-our-bet-on-human-in-the-loop-automation.

[216] A. Nanavati *et al.*, "Not all who wander are lost: A localization-free system for in-the-wild mobile robot deployments," in *Proceedings of the 2022 ACM/IEEE International Conference on Human-Robot Interaction (HRI)*, ser. HRI '22, Sapporo, Hokkaido, Japan, Mar. 2022, pp. 422–431. doi: 10.1109/HRI53351.2022.9889620.

[217] R. Tan and R. Cabato, "Scale ai's remotasks and the hidden workforce behind artificial intelligence in the philippines," *The Washington Post*, Aug. 2023. [Online]. Available: https://www.washingtonpost.com/world/2023/08/28/scale-ai-remotasks-philippines-artificial-intelligence/.

[218] M. Grotz *et al.*, "Towards robustly picking unseen objects from densely packed shelves," Jul. 2023.

[219] R. Hornung, H. Urbanek, J. Klodmann, C. Osendorfer, and P. van der Smagt, "Model-free robot anomaly detection," in *2014 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2014, pp. 3676–3683. DOI: 10.1109/IROS.2014.6943078.

[220] D. Park, Z. Erickson, T. Bhattacharjee, and C. C. Kemp, "Multimodal execution monitoring for anomaly detection during robot manipulation," in *2016 IEEE International Conference on Robotics and Automation (ICRA)*, 2016, pp. 407–414. DOI: 10.1109/ICRA.2016.7487160.

[221] A. Inceoglu, E. E. Aksoy, A. Cihan Ak, and S. Sariel, "Fino-net: A deep multimodal sensor fusion framework for manipulation failure detection," in *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2021, pp. 6841–6847. DOI: 10.1109/IROS51168.2021.9636455.

[222] P. Hegemann, T. Zechmeister, M. Grotz, K. Hitzler, and T. Asfour, "Learning symbolic failure detection for grasping and mobile manipulation tasks," in *2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, IEEE, 2022, pp. 4302–4309. DOI: 10.1109/IROS47612.2022.9982223.

[223] J. Duan *et al.*, "Aha: A vision-language-model for detecting and reasoning over failures in robotic manipulation," *arXiv preprint arXiv:2410.00371*, 2024.

[224] G. I. Melsion, R. Stower, K. Winkle, and I. Leite, "What's at stake? robot explanations matter for high but not low-stake scenarios," in *2023 32nd IEEE International Conference on Robot and Human Interactive Communication (RO-MAN)*, 2023, pp. 2421–2426. DOI: 10.1109/RO-MAN57019.2023.10309566.

[225] S. B. Remman, I. Strümke, and A. M. Lekkas, "Causal versus marginal shapley values for robotic lever manipulation controlled using deep reinforcement learning," in *American Control Conference, ACC 2022, Atlanta, GA, USA, June 8-10, 2022*, IEEE, 2022, pp. 2683–2690. DOI: 10.23919/ACC53348.2022.9867807.

[226] B. Ikeda and D. Szafir, "Advancing the design of visual debugging tools for roboticists," in *Proceedings of the 2022 ACM/IEEE International Conference on Human-Robot Interaction*, ser. HRI '22, Sapporo, Hokkaido, Japan: IEEE Press, 2022, pp. 195–204. DOI: 10.1109/HRI53351.2022.9889392.

[227] S. M. Lundberg and S.-I. Lee, "A unified approach to interpreting model predictions," in *Proceedings of the 31st International Conference on Neural Information Processing Systems*, ser. NIPS'17, Long Beach, California, USA: Curran Associates Inc., 2017, pp. 4768–4777, ISBN: 9781510860964.

[228] T. Chen and C. Guestrin, "Xgboost: A scalable tree boosting system," in *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, ser. KDD '16, San Francisco, California, USA: Association for Computing Machinery, 2016, pp. 785–794, ISBN: 9781450342322. DOI: 10.1145/2939672.2939785.

[229] D. Janzing, L. Minorics, and P. Bloebaum, "Feature relevance quantification in explainable ai: A causal problem," in *Proceedings of the Twenty Third International Conference on Artificial Intelligence and Statistics*, S. Chiappa and R. Calandra, Eds., ser. Proceedings of Machine Learning Research, vol. 108, PMLR, Aug. 2020, pp. 2907–2916.

[230] J. Brooke, "Sus: A quick and dirty usability scale," Digital Equipment Corporation, Tech. Rep., 1996, In P. W. Jordan, B. Thomas, B. A. Weerdmeester, & I. L. McClelland (Eds.), Usability Evaluation in Industry.

[231] J.-Y. Jian, A. M. Bisantz, and C. M. Drury, "Foundations for an empirically determined scale of trust in automated systems," *International Journal of Cognitive Ergonomics*, vol. 4, no. 1, pp. 53–71, 2000. doi: 10.1207/S15327566IJCE0401_04.

[232] L. Tesler, *Origins of the Apple Human Interface*, Lecture, Mountain View, California, Oct. 1997.

[233] *ELAN (version 6.8) [computer software]*, Retrieved from https://archive.mpi.nl/tla/elan, Nijmegen, 2024.

[234] D. Bates, M. Mächler, B. Bolker, and S. Walker, "Fitting linear mixed-effects models using lme4," *Journal of Statistical Software*, vol. 67, no. 1, pp. 1–48, 2015. doi: 10.18637/jss.v067.i01.

[235] M. E. Brooks *et al.*, "glmmTMB balances speed and flexibility among packages for zero-inflated generalized linear mixed modeling," *The R Journal*, vol. 9, no. 2, pp. 378–400, 2017. doi: 10.32614/RJ-2017-066.

[236] R. V. Lenth, *Emmeans: Estimated marginal means, aka least-squares means*, R package version 1.10.4, https://rvlenth.github.io/emmeans/, 2024. [Online]. Available: https://rvlenth.github.io/emmeans/.

[237] William Revelle, *Psych: Procedures for psychological, psychometric, and personality research*, R package version 2.4.6, Northwestern University, Evanston, Illinois, 2024. [Online]. Available: https://CRAN.R-project.org/package=psych.

[238] T. M. Therneau, *A package for survival analysis in r*, R package version 3.7-0, 2024. [Online]. Available: https://CRAN.R-project.org/package=survival.

[239] G. N. Wilkinson and C. E. Rogers, "Symbolic description of factorial models for analysis of variance," *Applied Statistics*, vol. 22, no. 3, pp. 392–399, 1973.

Appendix: Attributions to Motion

A

A.1 Initializing a Pool of Trajectories

Our approach suggests beginning with a pool of trajectories, however obtaining this pool can pose a bootstrapping problem. It's unlikely that a diverse corpus of trajectories—one that would span the feasible attribution space and enable a data-driven extraction approach—would already exist. In order to create such a corpus, one would need to be confident that they have included trajectories that adequately cover the range of attributions that are perceivable in a domain, but the lack of such information is why we want the corpus of trajectories to begin with.

We broke this cycle by referencing existing literature on the perceptions of robot vacuum cleaners and selecting six adjectives as candidate attributions: curious, broken, energetic, lazy, lost, scared. We then manually demonstrated six trajectories that we judged to maximally express these candidates. We collected responses to these trajectories and conducted an initial analysis of what aspects of the motion users said contributed to their ratings. This informed the creation of an initial subset of the features we would use. Additional trajectories were then generated using a hill-descending search in the space of trajectories optimizing for incrementally altering individual features (±0.1-0.3) selected at random while holding other features constant. We found that alternative optimization criteria, like diversity, would lead to degenerate or trivial trajectories which happened to have extreme feature values. These new trajectories were posed to users for their ratings. Simultaneously, we asked them to demonstrate how they would "clean the bedroom in a way that makes the robot look _____" for random items from our questionnaire. Responses fed back into analysis and the process was cycled through two more times with a subset of the generated and demonstrated trajectories, resulting in the final exploratory dataset and trajectory featurization.

A.2 Loadings

The factor loadings of our final model, derived via exploratory factor analysis of the exploratory dataset, are shown in Table A.1.

Table A.1: Factor loading matrix

Variable	Factor 1	Factor 2	Factor 3	Communality
Responsible	**1.00**	.06	−.07	1.01
Competent	**.94**	.03	.00	.89
Efficient	**.93**	.05	−.03	.87
Reliable	**.85**	−.05	.07	.73
Intelligent	**.85**	−.05	.06	.73
Focused	**.84**	−.07	.02	.71
Lost	− .03	**.90**	.01	.80
Clumsy	.13	**.76**	.06	.59
Confused	− .21	**.76**	.03	.63
Broken	− .17	**.62**	−.11	.43
Curious	− .04	.06	**.91**	.83
Investigative	.12	−.02	**.78**	.62

A.3 Data Sensitivity

We investigated the sensitivity of our candidate models' performance on unseen trajectories to increased amounts of training data. We divided our final dataset, holding out 20% of the trajectories, then trained and tested models using varying percentages of the training data, repeating the evaluation with 8 randomized folds of the training data. The results, shown in Figure A.1, indicate diminishing returns beginning around the use of 50% of the training data.

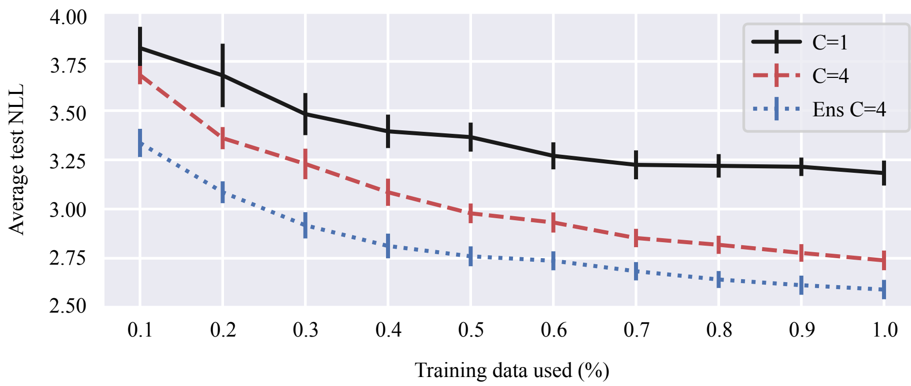


Figure A.1: Average test negative log likelihood as a function of the amount of labeled human data used in training. Each datapoint represents the mean of the average test NLL over 10 random folds. Error bars denote standard deviation.

A.4 Trajectory Optimization

We would like to generate trajectories that elicit a particular attribution according to the optimization (5.1). We have the forward model $f_{B|\Xi}|\phi(\xi)$ which predicts a distribution over attributions given the featurization of a particular trajectory. This model affords two routes towards realizing the optimization. In the first, we directly optimize the features to maximize the objective. However, in that case, a further optimization would be needed to find a trajectory that has the desired feature vector, and it is possible (and common in practice) that there does not exist a trajectory that produces the target features. The second route, which we adopt, is to search in the space of trajectories. This space is large, and in general there is no clear best way to structure the search. Fortunately the features we use have a clear relationship with patterns of motion, so we can easily sample trajectories that increase the activation of individual features. Given a trajectory, we sample a large set of neighboring trajectories using the following modifications:

Action modification : The trajectory is scanned and new trajectories are initialized by individually witholding each state s_i. The single removed state is replaced by selecting every valid alternative action from s_{i-1} and reconnecting the trajectory with a shortest path to s_{i+1}.

Shortcutting : Sections of the trajectory are deleted uniformly at random and the trajectory is reconnected with a shortest path. We sample twice as many cuts as there are states in the trajectory. Shortcutting is a well established post-processing step in motion planning, helping to uncover shorter or cheaper trajectories that still satisfy the objective.

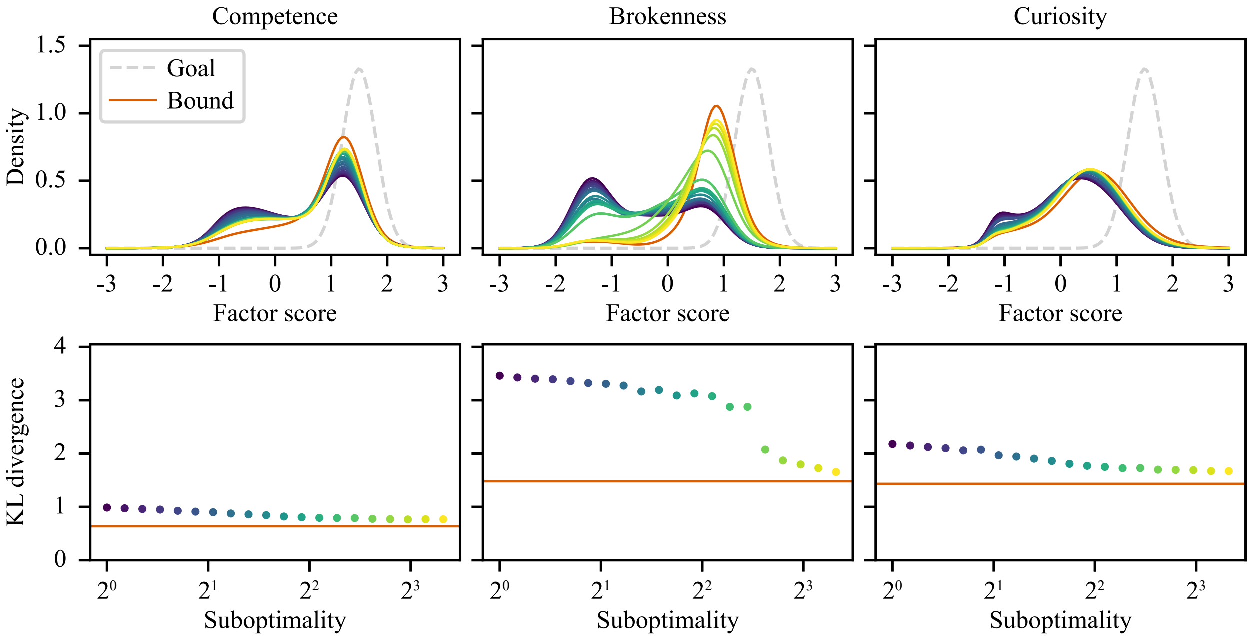
Template Insertion : A sequence of states is patched into the trajectory beginning at each s_i. We used a "straight" template, formed by taking the action used in s_{i-1} and repeating it twice, and a "U" template, formed by taking three actions in proceeding clockwise or counter-clockwise directions. The trajectory is reconnected with a shortest path to s_{i+1}. Both templates were directly motivated by participant feedback highlighting these patterns as contributing to the appearance of an organized principle to the robot's motion.

Collision Seeking : The trajectory is scanned and collision-avoiding actions are individually replaced with collision-causing actions if they are available from a state s_i. The trajectory is then reconnected with a shortest path to s_{i+1}. This sampler was motivated by participant responses highlighting collisions as contributing to their ratings of components of the brokenness score. These same modifications are generated by the "Action modification" sampler. We double-sample these to reduce the chance that they are dropped in a subsequent subsampling step.

Overcoverage : A random section of the trajectory $s_i..s_j$ between 3 and 6 states long is selected uniformly at random and the trajectory is modified to backtrack this portion by inserting $reverse(s_i..s_j)$ followed by $s_i..s_j$ in place of s_j. This modification is motivated by participant feedback that highlighted redundant coverage as evidence of attentiveness, exploration or attention to detail.

These samplers, and a good initialization produced by solving the original task using a planner, make sampling-based greedy hill-descending search in the space of trajectories effective. As practical considerations, we discard sampled trajectories longer than 250 states to prevent the search from exploding, then we randomly subsample 250 of the modified trajectories, and we terminate the search once 750 steps have been taken.

As illustrated in Figure A.2 and Figure A.3, this search process is generally effective at making use of additional allowable suboptimality to produce trajectories that the model predicts to better elicit a desired attribution. Some non-monotonicity can be observed in one of the brokenness and one of the curiosity plots. As trajectories get longer, the pool of neighbors sampled is diluted with many more small changes and our subsampling step can lead the search to randomly discard more fruitful paths.



(a) Training environment (as used in Experiment I)

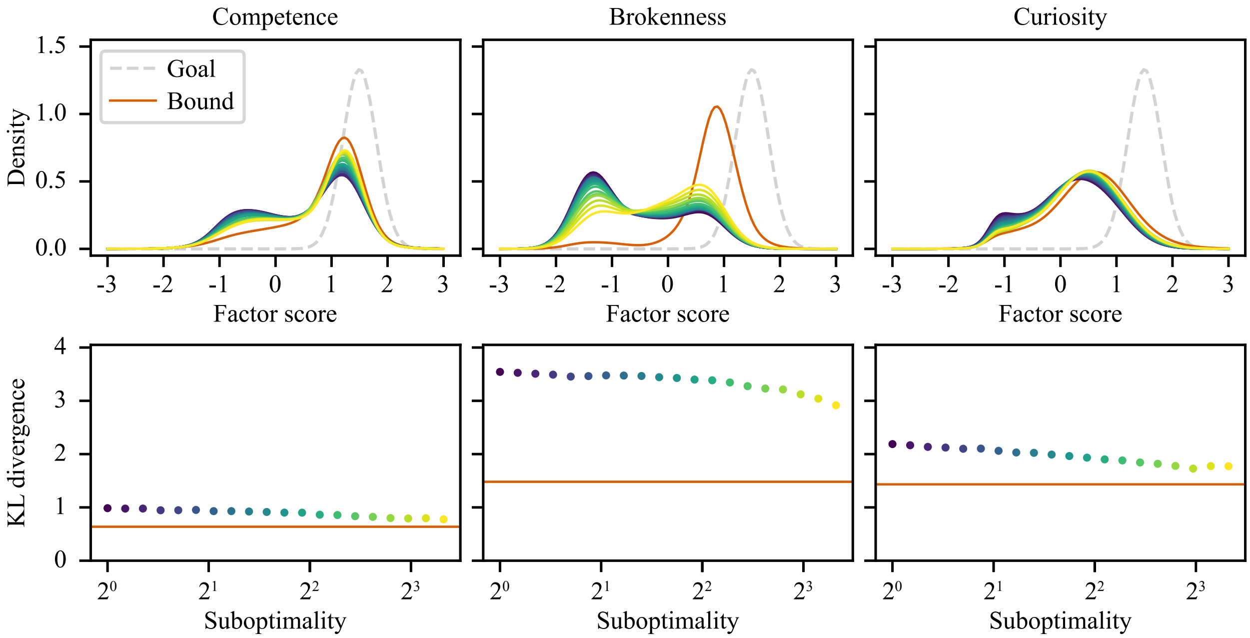
Figure A.2: Predicted distribution of factor scores for trajectories optimized under varying settings of w, represented as a ratio of the task-optimal cost. (Part 1: Training environment as used in Experiment I.)

A.5 Statistical Results

We conducted one-sample Kolomogorov-Smirnov tests comparing the empirical distribution of factor scores elicited from participants against the distribution predicted by our model. To account for the increased likelihood of Type-I errors due to multiple testing, the Holm-Sidak adjustment was applied to the resulting p values. The results of the tests are provided in Table A.2.

For Experiment I, the tests failed to indicate a significant difference between the distributions for all but the Competence-1x condition, suggesting that there is insufficient evidence to support inequivalence between predicted and observed distributions. For Experiment II, the tests failed to indicate a significant difference between the distributions for all conditions, suggesting that there is insufficient evidence to support inequivalence between predicted and observed distributions.

We conducted Kendall's tau-b correlations to determine the relationship between the allowable



(a) Modified environment (as used in Experiment II)

Figure A.3: Predicted distribution of factor scores for trajectories optimized under varying settings of w, represented as a ratio of the task-optimal cost. (Part 2: Modified environment as used in Experiment II.) The optimization goals are configured as in our experiments: a Gaussian centered at 1.5 with standard deviation 0.3. The plots show the marginal density for the factor under consideration. The companion plots show the KL divergence for each plotted distribution, measuring closeness to the goal distribution for each generated trajectory. The bound line represents the distribution and KL divergence predicted for a feature vector obtained by optimizing features directly, as opposed to optimizing in the space of trajectories. Generally, these features may not be realizable with a trajectory that obeys domain constraints, so the predicted distribution represents an upper bound.

suboptimality of a generated trajectory (1, 2, 4 or 12) and the observed factor scores for the attribution under study. The correlation coefficients and p values are reported in Table A.3. The results indicate strong positive correlations between the allowable suboptimality and brokenness factor scores, suggesting that the trajectory generation method was effective at eliciting progressively higher factor scores. However, tests for the competence conditions indicated a moderate negative correlation, and tests for curiosity conditions were not significant.

	Experiment I						Experiment II					
	Competence		Brokenness		Curiosity		Competence		Brokenness		Curiosity	
	$D(24)$	p	$D(24)$	p	$D(24)$	p	$D(24)$	p	$D(24)$	p	$D(24)$	p
1x	.381	**.014**	.292	.255	.233	.659	.343	.059	.247	.464	.319	.143
2x	.190	.789	.185	.789	.120	.879	.309	.145	.151	.836	.299	.206
4x	.275	.334	.232	.659	.144	.879	.182	.738	.256	.446	.254	.464
12x	.277	.334	.217	.695	.198	.789	.246	.464	.211	.601	.143	.836

Table A.2: Results of Kolmogorov-Smirnov tests comparing predicted and observed distributions for each condition

Experiment I						Experiment II					
Competence		Brokenness		Curiosity		Competence		Brokenness		Curiosity	
τ_b	p	τ_b	p	τ_b	p	τ_b	p	τ_b	p	τ_b	p
-.261	**.001**	.402	**<.001**	-.069	.370	-.175	**.045**	.351	**<.001**	-.033	.675

Table A.3: Results of Kendall's tau-b correlation between optimization parameter w and factor score

Appendix: Pointing-based Assistive Teleoperation

B

B.1 System

All participants interacted with the simulation running in NVIDIA Omniverse Isaac Sim 2022.2.0 on a machine with an RTX 3060 Ti. The simulation ran at interactive framerates, averaging about 38fps and dipping to lows of about 15fps when participants created large numbers of contacts by pushing through many blocks at once.

The bottleneck computation in the assistance systems was the filtering step where generated poses were rejected if they created collisions between the gripper and the scene. GPU acceleration was necessary for considering thousands of candidates each frame, as shown in the performance comparison in Figure B.2. Checking that candidates had feasible inverse kinematics solutions was not feasible at the time of the study. Participants encountered unreachable suggestions only infrequently because the block scatterings were placed comfortably within the robot's workspace. Participants that knocked or placed blocks further away were more likely to encounter unreachable suggestions.

A 3DConnexion SpaceMouse Pro was used for all control input. The input mapping used was provided to participants as a printout (shown in Figure B.1 for reference during the study.

Figure B.1: The mapping of buttons to system controls used during the study.

B.2 Trajectory Labeling

Participant stacking trajectories were manually annotated by two of the authors using ELAN [233]. The events annotated and their descriptions are given in Table B.1. Only a subset of the events were analyzed for this work.

B.3 Statistical Details

All analyses were conducted in R. Linear mixed models (LMMs) and generalized linear mixed models (GLMMs) were fit using `lme4` [234] or `glmmTMB` [235] when modeling zero-inflated distributions. Estimated marginal means were computed using `emmeans` [236]. Exploratory factor analysis was conducted using `psych` [237], and survival analysis was conducted using `survival` [238].

Where presented, models are described in Wilkinson notation [239]. Linear mixed models were used for continous response variables, and Poisson GLMMs with a log link function were used to model count data. For linear mixed models, the Kenward-Roger method of estimating

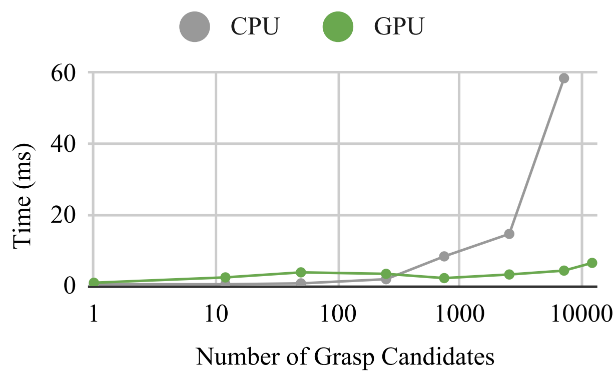


Figure B.2: Comparison of time taken to evaluate N grasp candidates. The GPU results are from an NVIDIA RTX 3060 Ti, and the CPU results are from single-threaded execution on an AMD Ryzen 9 5900X. GPU acceleration is necessary to check thousands of candidate grasp poses for scene collisions while maintaing system responsiveness.

degrees of freedom was used.

B.3.1 Subjective Assistance Scores

Users responded to four Likert items after experiencing assisted conditions (either IMP and EXP). The items' statements are shown in Table B.2. We conducted an exploratory factor analysis (EFA) using the minimum residual method to identify the structure of responses to these novel items. The EFA indicated that a one-factor solution was sufficient, however item Q9 showed low communality, so it was analyzed as a standalone item, "understanding." Responses to the remaining items were averaged to form the "assistance composite" score.

The responses for the composite scale and standalone item were both independently analyzed using a LMM that accounted for the condition as well as inter-participant differences.

$$\text{SubjectiveScore} \sim \text{Condition} + (1|\text{Subject})$$

The resulting estimated marginal means are shown in Table 6.3. Means were compared using t tests.

Table B.1: Event codes and descriptions

Code	Description
Tasks	
pickt	Successful pick: task block.
picko	Successful pick: other block.
release	Release with support. Code moment of release, then code outcome when it is clear.
drop	Drop (the block in the gripper). Code moment of release, then code outcome when it is clear.
place	Successful place: Stable place (of object in gripper). Code moment object stable at rest.
Errors	
erkno	Deconstructed (“knocked over") existing tower.
erbut	Unintentional drop, likely due to accidental button press.
erair	Unsuccessful pick attempt, air grasp.
erpop	Unsuccessful pick attempt, pop or slip out of gripper.
erfal	Unsuccessful place attempt, contacted tower but fell.
erpmi	Unsuccessful place attempt, missed tower.
ercon	Object lost from gripper due to contact with scene.
Milestones	
blue1	Assumed at start. Code after errors that cause deconstruction.
blue2	
blue3	
pink1	Assumed at start. Code after errors that cause deconstruction.
pink2	
pink3	

B.3.2 Condition Preferences

For each of the forced-choice preference questions, a multinomial test was performed to evaluate whether participants’ preferences among the three conditions were evenly distributed. Tests for questions EQ1, EQ3, and EQ4 were significant, while a test of EQ2 was not.

Pairwise binomial tests were conducted to compare preferences between each pair of conditions, using Holm-Bonferroni corrections to account for multiple comparisons. The results of the pairwise comparisons are shown in Table 6.1. Responses to question EQ0 were highly similar to that of the

Table B.2: Survey questions and codes

Code	Description
NASA-TLX	
Q1	How mentally demanding was the task?
Q2	How physically demanding was the task?
Q3	How hurried or rushed was the pace of the task?
Q4	How successful were you in accomplishing what you were asked to do?
Q5	How hard did you have to work to accomplish your level of performance?
Q6	How insecure, discouraged, irritated, stressed, and annoyed were you?
Agreement	
Q7	"The suggestions made it easy to accomplish the task."
Q8	"The suggestions made it easy to accomplish the task in the way that I wanted."
Q9	"I understood why the suggestions behaved the way they did."
Q10	"I was in control of the suggestions."
Open-ended	
Q11	Briefly describe the strategy you used for completing the task.
Q12	What were your biggest frustrations with this system?
Concluding Questions	
EQ0	Which system was most effective for the task?
EQ1	Which system felt easiest to use?
EQ2	Which system's suggestions made it easiest to do the task the way you wanted to?
EQ3	With which system did you best understand why the suggestions behaved the way they did?
EQ4	With which system did you feel most in control of the suggestions?
EQ5	What were the major reasons for your choices?

Table B.3: Comparison of condition preference counts for EQ0

	A	B	C_A	C_B	$\frac{C_A}{C_A+C_B}\%$	CI	p
EQ0	EXP	CON	12	1	92	(64, 100)	**.010**
	"	IMP	"	7	63	(38, 84)	.359
	CON	IMP	1	"	13	(0, 53)	.141

other questions, but are given in separate table Table B.3 for completeness.

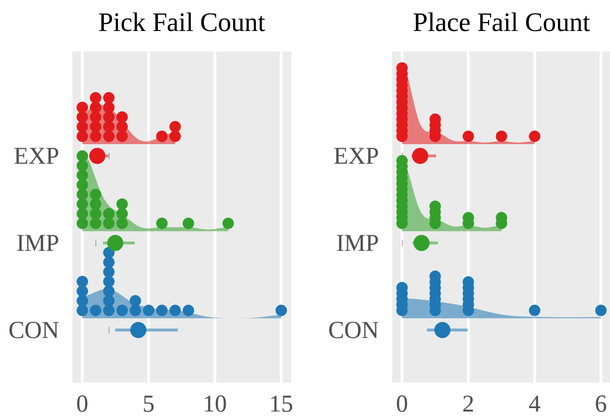


Figure B.3: Counts of pick and place errors observed per participant by condition. Point and bar show estimated marginal mean and 95% confidence interval.

B.3.3 Pick Failure Count

Pick failure models incorporated order, condition and block configuration factors as well as participant random effects.

A GLMM with a Poisson link function was used to model the count data. Excess zeros were observed in the baseline condition (CON), so a zero-inflation binomial term with the condition as the sole fixed effect was incorporated.

$$\begin{aligned}\text{PickFailureCount} &\sim \text{Order} * \text{Condition} \\ &+ \text{Configuration} + (1|\text{Subject})\end{aligned}$$

A Type III Wald chi-square test was conducted to examine the effects of order, condition, environment, and their interaction on the number of pick failures. The main effect of order was significant, $\chi^2(2) = 11.40, p = .003$, indicating that the number of pick failures differed depending on the order the condition was experienced in, consistent with a learning effect. The main effect of condition was also significant, $\chi^2(2) = 14.10, p < .001$, suggesting differences in pick failures across conditions. The main effect of environment was not statistically significant, $\chi^2(2) = 5.46, p = .065$, indicating that the environment did not have a significant effect on the number of pick failures.

A significant interaction effect was found between order and condition, $\chi^2(4) = 13.81, p = .008$, suggesting that the effect of order on pick failures depends on the condition. The intercept was also significant, $\chi^2(1) = 52.37, p < .001$, indicating a significant baseline level of pick failures.

Estimated marginal means for this model, given in Table 6.4, were averaged over levels of order and environment. They are plotted along with the underlying observations in Figure B.3. Pairwise t tests were conducted, with p values adjusted using the Tukey method for comparing a family of 3 estimates to account for multiple comparisons.

B.3.4 Place Failure Count

Pick failure models incorporated fixed condition and random participant effects.

$$\text{PlaceFailureCount} \sim \text{Condition} + (1|\text{Subject})$$

A Type III Wald chi-square test was conducted to examine the effect of condition on the number of placement failures. The main effect of condition was statistically significant, $\chi^2(2) = 8.20, p = .017$, indicating that the number of placement failures differed across the levels of condition. The intercept was not statistically significant, $\chi^2(1) = 0.63, p = .427$, suggesting that the number of placement failures in the control condition (CON) was typically indistinguishable from zero.

Estimated marginal means for the place failure model are given in Table 6.4 and plotted along with the underlying observations in Figure B.3. p values were adjusted using the Tukey method for comparing a family of 3 estimates to account for multiple comparisons.

B.3.5 Workload

Factors for condition order and block configuration (which of the three block scatterings, shown in Figure 6.3d was used for the trial) were considered, but did not significantly affect the model's outcome or fit, and are not included in the final analysis.

$$\text{Workload} \sim \text{Condition} + (1|\text{Subject})$$

A Type III Wald chi-square test was conducted to examine the effect of condition on workload. The effect was statistically significant, $\chi^2(2) = 15.27, p < .001$, indicating that workload differed across conditions. Estimated marginal means of the workload model are given in Table 6.2. Pairwise t tests were conducted, and p values were adjusted using the Tukey method for comparing a family of 3 estimates to account for multiple comparisons.

B.3.6 Survival Analysis

The Kaplan-Meier estimator was used to characterize participant's progression, with the resulting model shown in Figure 6.4. While surviving longer is usually the desired observation in a survival analysis (e.g. when analyzing mortality data of patients receiving experimental medical interventions), our objective is for participants to complete the task more quickly. We invert the typical Y-axis "survival" rate and display completion (or "mortality") instead, so that the plot may still be read as higher-is-better.