DISSERTATION

REAL-TIME AI ASSISTANCE FOR DISORIENTING CONTROL TASKS: PERFORMANCE, BEHAVIOR, AND TRUST

Submitted by

Sheikh A. Mannan

Department of Computer Science

In partial fulfillment of the requirements

For the Degree of Doctor of Philosophy

Colorado State University

Fort Collins, Colorado

Spring 2026

Doctoral Committee:

Advisor: Nikhil Krishnaswamy

Nathaniel Blanchard
Sarath Sreedharan
Matthew Rhodes

ABSTRACT

REAL-TIME AI ASSISTANCE FOR DISORIENTING CONTROL TASKS: PERFORMANCE, BEHAVIOR, AND TRUST

Spatial awareness is an important ability humans develop to use in everyday activities like walking and driving. It is an even more critical skill required in high-risk occupations, such as piloting an airplane or spacecraft. Erroneous inputs to the sensory system can lead to spatial disorientation, rendering a person unable to interpret their speed, position, and orientation with respect to other objects or the horizon. A report by the Federal Aviation Administration indicates that 367 fatal accidents in general aviation could be attributed to spatial disorientation among pilots between 2003 and 2021. An AI system situated in the problem space could monitor the vehicle's position and orientation, determine when loss of control is imminent, and provide corrective maneuvers to avoid accidents, all in real time. This work presents a first-of-a-kind AI system designed to assist humans with visual aids in disorienting continuous-control tasks, specifically the Virtual Inverted Pendulum and a navigation task in a flight simulator. An offline evaluation reveals that an AI system can select actions, per the task's specifications, objectively better than humans. The results align with the hypothesis that an AI system is not susceptible to the same sensory disruptions as humans and can therefore excel at the task and guide humans toward more accurate actions. This dissertation demonstrates, through multiple human-subject studies, that bi-directional learning can improve specific task performance metrics in both humans and AI while aligning the AI agent's resulting actions with human intuition, but sometimes at the cost of AI performance. Empirical evidence also indicates that humans prefer to trust AI systems that are more closely aligned with human behavior, and that they also trust intuitive and non-intrusive forms of assistance, even when there is no objective improvement in task performance.

ACKNOWLEDGEMENTS

While this dissertation bears my name, it is the culmination of 5 years of work that would not have been possible without the individuals who contributed to this research through their insight, encouragement, and friendship.

I would like to start by expressing my deepest gratitude to my advisor, Prof. Nikhil Krishnaswamy, for his continuous guidance, support, and immense patience throughout my PhD journey. I feel incredibly blessed to have an advisor who helped me bring my vision for research into reality, but also challenged me to reach my full potential as a researcher by providing essential structure and direction to my work. I was constantly surprised and humbled to send a message at 3 or 4 AM, only to receive a reply an hour later; it took me some time to realize that while it was late night for me, it was already early morning for him. I can only hope to find a similar work ethic in my own life moving forward, as it clearly hasn't quite taken root in me just yet.

I would also like to thank the members of my dissertation committee for their invaluable contributions. I am grateful to Prof. Nathaniel Blanchard for our engaging and eye-opening discussions that challenged my thinking. I also extend my sincere thanks to Prof. Sarath Sreedharan and Prof. Matthew Rhodes, who generously agreed to join the committee and provide their expertise during the final stages of this process. I thank you all for your valuable insights; your collective feedback was instrumental in transforming the final form of this dissertation.

I would also like to extend a special thanks to Prof. Paul DiZio and Dr. Vivekanand Vimal from the Ashton Graybiel Spatial Orientation Lab for their collaboration and perspectives they shared during our joint work. This research was inspired in part by their work and greatly benefited from their early guidance and feedback.

I am also deeply grateful to my colleagues and friends who made the lab feel like a second home. To my lab mates in the SIGNAL and Vision labs at CSU – Abhijnan Nath, Animesh Gurjar, Anju Gopinath, Carine Graff, Changsoo Jung, Ethan Seefried, Ibrahim Khebour, Jack Fitzgerald, Mariah Bradford, Paige Hansen, Sifatul Anindho, and Videep Venkatesha – and all those who

passed through the lab during my time there, thank you for the collaborative spirit, the technical troubleshooting, the willingness to be a sounding board for my ideas, and the much-needed humor during the most challenging phases of this work.

My close friends – Abdullah Shamail, Abdulhadi Qureshi, Abrar Tariq, Adeel Ibrahim, Ali Ahad, Dilawer Ahmed, Ghulam Ali, Huma Tariq, Muhammad Raffae, Syed Hamza Ahmad, and Umer Farooq – and the many others who supported me along the way, thank you for being my support system outside of the university. Whether it was through a listening ear or a necessary distraction, your friendship kept me grounded and reminded me of the world beyond my research.

This material is based in part upon work supported by Other Transaction award 1AY2AX000062 from the U.S. Advanced Research Projects Agency for Health (ARPA-H) Platform Accelerating Rural Access to Distributed Integrated Medical Care (PARADIGM) program. The views and conclusions contained in this document are those of the authors and should not be interpreted as representing the official policies, either expressed or implied, of the U.S. Government.

Additionally, I would also like to thank the volunteers who agreed to participate in the human subject studies; without their time and contribution, this work would not have been possible.

Lastly, to my family, thank you for empowering me to pursue my dreams. I am especially grateful to my mother, Unzela Shah, and my late grandmother, Rehana Shah, for the continuous and unwavering support throughout my life. This achievement is as much theirs as it is mine.

DEDICATION

For my mother

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

Chapter 1

Introduction

Spatial awareness and orientation are the abilities to perceive objects or landmarks in your surroundings and to maintain track of their relative position and direction relative to your own. Humans maintain awareness and orientation through a combination of the visual, vestibular, and proprioceptive systems working together [115, 11]. Maintaining spatial awareness is a critical faculty for humans when engaging in activities such as driving a car or in potentially hazardous occupations such as piloting an airplane or spacecraft. Any erroneous input to the sensory systems that leads to sensory mismatch can cause illusions that degrade this ability, resulting in spatial disorientation (SD). In aviation, SD has led to multiple fatal accidents and remains the leading cause of fatal aircraft accidents [26, 62]. Moreover, during extreme conditions, such as piloting a spacecraft, even expert humans are subject to gravitational transitions where they may not be able to rely on gravitational cues sensed by the vestibular system, leading to fatal accidents [158, 42].

Based on a report from the Federal Aviation Administration, from 2003 to 2021, a total of 26,535 accidents occurred in general aviation, and 367 fatal accidents were attributed to SD [17]. Additionally, the report also reveals a few more details about SD-related accidents: a) strong correlations between such accidents and pilots with less than 500 hours of flight experience, and b) even though there has been an overall decrease in accidents attributed to SD, there has been a recent upward trend in such accidents, suggesting an overall need for more education or countermeasure aids. But how would a system detect that the human is disoriented (or about to be) and intervene in a timely manner, and what is the best method for the system to provide corrective signals?

There have been multiple examples of systems providing a variety of support to pilots such as predicting hard landings for commercial flights [63], runway overruns during takeoff or landing [7], using large-language models to extract and synthesize information for aircraft type-specific manuals in case of emergencies [153] or general flight trajectory planning and harsh weather awareness [181]. Autonomous AI systems have also been tested by Airbus for routine flight procedures;

its Autonomous Taxi, Take-Off and Landing (ATTOL) can perform 3 tasks (taxiing, landing, and takeoff) using on-board image recognition technology [5]. Much research has been conducted on detecting and classifying SD occurrences in humans [56, 73] by monitoring pilots' physiological behavior and patterns, as pilots must maintain high concentration during flight. Daiker et al. [43] propose a proof of concept on how to alert pilots when they are spatially disoriented based on signals from the vestibular system.

However, no system has yet been developed that leverages AI to combat spatial disorientation during flight via real-time corrective maneuvers. Ideally, an AI system with access to quantitative information about position and orientation can potentially predict a loss of control by the pilot and provide corrective actions for the pilot to perform, or take control itself in the worst case. But for such a system to be deployed in the wild, there are certain challenges and requirements that would need to be addressed [20]: Are the decisions from the AI system explainable? Is the system robust to errors? Could it detect when an experienced pilot is suffering from spatial disorientation and is about to crash the plane and intervene by guiding the pilot, or in the worst case, override the pilot, to bring the aircraft back to normal operating conditions and avert the pending crash? Furthermore, how should the AI learn and embody the task space and collaborate with humans in a constructive and safe manner?

The Role of Embodiment in this Context

AI research has explored several questions regarding whether imitating a part of human intelligence is enough or whether intelligence is simply a consequence of lived experience, and no non-humanlike entity can ever achieve or "embody" human-like behavior and cognition [33]. Whether through imitation or actual learning, AI research has led to numerous new technologies, including virtual assistants, recommendation engines, autonomous systems, generative AI, and large-language models. However, gaps remain in high-risk domains, where a human touch is strongly preferred but could substantially benefit from additional guidance. In such areas, human performance is highly dependent on their *embodiment*, including the ability to maintain spatial orientation. If AIs perform the same tasks but do so in a substantially different way from that of

a human expert, this may have serious consequences for human trust and the perceived impact of AI systems. But what is embodiment, and what does it mean for an AI to be embodied in the task space?

The theory of Embodiment has been a key topic of research in cognitive science and psychology. Numerous studies have sought to identify the basic requirements for an object to be considered embodied. Ziemke [200] formalizes the theory introduced by Thompson and Varela [166], Varela et al. [170]. The most trivial form of embodiment an object can have is *structural coupling*, where the entity has an effect on the environment and the environment has an effect on the entity to some degree. As the most basic notion of embodiment, structural coupling can be applied not only to living but also to non-living entities as well, for example, a 4-legged wooden chair would exert a force on the floor it is placed on due to gravity, and the same floor would exert a similar but opposite force on the chair (Newton's third law of motion). Ziemke, Thompson, Varela, and others have built upon this definition to introduce notions of embodiment that could be used to understand more complex entities (e.g., "historical" and "physical" embodiment) and also more restrictive notions of embodiment that encompass only living organisms. I will go over these different notions of embodiment in Chapter 2 but I propose that for robust task guidance in any physically situated scenario and application, the underlying system should be "embodied" within the problem space such that the system should, at the bare minimum, be able to perceive the world similarly to how humans do, reason about the situation at hand and eventually actualize their intentions. I also propose another definition of embodiment that is more appropriate for different learning algorithms in AI and their effects on task behavior.

Current State of AI agents

The AI community has produced countless AI agents that could appear human-like in some manner, such as the Rogerian therapist Eliza from the 1960s [187], the emotional robot Kismet [27], or the current multi-modal large language models (LLMs) like GPT-X from OpenAI, or Gemini from Google [2, 165]. Along with agents, multiple simulation platforms, such as Habitat-Sim [152], VRKitchen [58], SAPIEN [192], etc., have been developed to study the effects of realism,

world physics, and interactivity on an embodied agent. One thing to highlight is that modern AI algorithms are vastly different from the symbolic AI of the 1960s (e.g., ELIZA), where rule-based algorithms, now known as good old-fashioned AI, led to agents whose "thought process"[1] allowed an outside observer some transparency to understand the outcome the agent/system delivered, unlike most current AI systems.

Recently, researchers have begun using LLMs as cognitive models for embodied agents such as PaLM-E [48], and current LLMs, such as GPT-4, Gemini, and Llama, are grounded in multiple modalities, enabling them to exhibit capabilities previously thought to require physical embodiment. However, LLMs face issues of explainability, hallucinations, awareness, and reasoning. Since 2010, explainability has been the primary criticism of deep learning models due to their lack of transparency in decision-making. This criticism was exacerbated by newer, larger models such as BERT, GPT -3, etc. There have been strides in enabling LLMs to provide explanations via Chain-of-Thought (CoT) prompting [186, 108, 97]. Some researchers would claim that these CoTs can be considered as the LLM's reasoning in response to a prompt.

Apart from a lack of explainability, another problem that almost all LLMs exhibit is hallucination: an LLM can generate outputs that are inaccurate or fictitious by recombining elements of its training data in ways inconsistent with reality. A good portion of research is focused on detecting and minimizing hallucinations using knowledge-based grounding, but therein lies the issue; Gekhman et al. [60] demonstrates that even though fine-tuning or grounding new knowledge into an LLM helps it use the knowledge more efficiently also increases the LLM's tendency to hallucinate [14, 3, 4, 84]. Lastly, LLMs fail to understand and even recognize the unspoken priors that humans carry, leading to poor social awareness and reasoning, which are important aspects of human social settings [101]. In Chapter 3, I present early evidence of language models' shortcomings and utility for guidance in mitigating spatial disorientation.

Whether or not based on LLMs, most modern embodied AI agents are trained and tested in simulated environments and have been shown to reach human-level performance on tasks such as

[1]Thought process here refers to the pattern recognition logic through pre-defined rules.

Visual Question Answering [50]. However, such simulated agents frequently fail in novel tasks and environments in 2 cases: 1) the physical embodiment of symbols and concepts is very limited, and 2) noise in the real world. For example, Diana, a mixed-reality, multi-modal, interactive, situated, and embodied agent, is trained to understand language and gestures, enabling her to interact with objects in her world [93]. If asked to build a house or tower given the same stack of blocks as shown in Figure 1.1, Diana may fail to perform the task as the symbols "house" and "tower" may not be grounded into its knowledge. If we consider an embodied robotic agent like PaLM-E, which displays *organismoidal* and physical embodiment (as I define in Chapter 2), it may fail to perform the same mobile manipulation task in a different real-world kitchen environment. Since PaLM-E is a multi-modal language model, it is also prone to hallucinations, which could be triggered by multiple modalities (vision and language). The ability to adapt to new environments or apply previously acquired knowledge to new tasks, a characteristic of humans, remains absent in LLM-based systems.

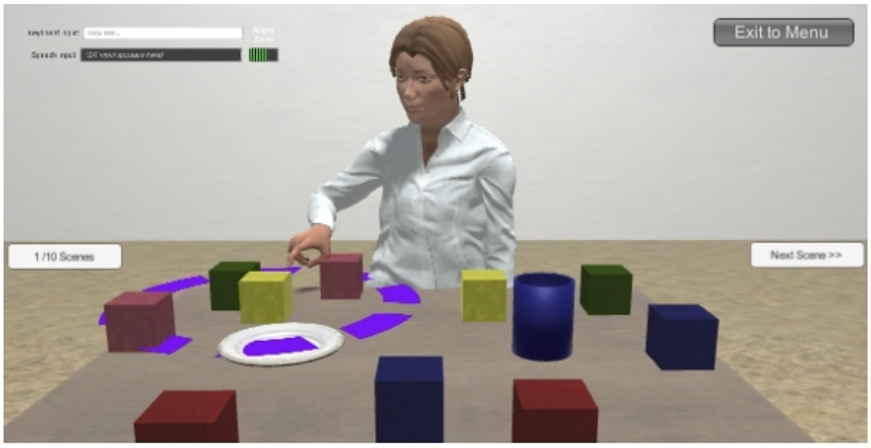


Figure 1.1: Diana is a mixed-reality, situated, multi-modal, and interactive agent trained to understand language and gesture that enables her to interact with objects in her world.

Techniques like (situated) grounding and human-in-the-loop (HITL) machine learning have been used widely to better ground knowledge[2] into AI agents and fills in the gaps where the cur-

[2]The term knowledge is used in a loaded form. Here, it combines the ability of perception and reasoning with human-like behavior that would be taught to an AI agent.

rent knowledge they possess fails them. While these techniques have shown promise in filling knowledge gaps and even adapting to new environments, current AI systems lack the functionality and nuance to provide **real-time** assistance for time-sensitive and life-critical tasks. This thesis uses spatial disorientation and piloting as the relevant domain, given the environmental awareness required for instantaneous decision-making. This creates a unique opportunity to test the capacity of AI and ML algorithms to provide decision support and guidance to humans in tasks that challenge human perception capabilities.

Human-AI Collaboration

The moon landing, the Human Genome Project, the Large Hadron Collider, and the internet are among the greatest achievements in human history. These were only made possible by individuals combining their diverse expertise in a constructive manner. With the ongoing evolution of technology, which has led to significant advancements in AI, there is increasing interest in new forms of collaboration between humans and AI systems. Early versions of these interactions used programs defined by a set of rules and/or heuristics that humans, while interacting, could understand and interpret the decisions and outcomes. AI systems are now seen to mimic or achieve near-human intelligence on a multitude of tasks. Despite these breakthroughs and the fact that the individual components, such as architectures and loss functions, are well known, the actual development of modern AI increasingly yields systems whose internal interpretations remain black boxes. Given the decisions and outcomes provided by these systems, they would need to be reliable, trustworthy, and explainable, and undergo rigorous scrutiny before being deployed in scenarios that must leverage Human-AI (HAI) interactions, especially in high-risk situations such as driving, flight, or medicine.

In the past, human programmers had to specify each instruction and action for their programs (and consequentially "agents" as they are more commonly known nowadays). Depending on the agent's responsibilities, situations that were not part of the original design can lead to either a system fault or, in the worst case, loss of life [191]. Earlier agents had limited generalizability to previously unseen situations with no fallback solutions, but in recent years, agents have evolved

to the point that, even in situations that were unknown to them during training, they can, in most cases, infer a plausible course of action. With deep neural networks, AI agents have high potential to dynamically learn from unseen scenarios and collaborate successfully with humans. This dissertation examines the requirement that such assistive agents be sufficiently embodied and situated to provide real-time task assistance for challenging action tasks.

Trust and Safety Prior research indicates that humans tend to over-rely on automated systems even though the system demonstrated a lack of reliability beforehand or an under-rely on a typically robust and reliable system after a single failure [167, 135, 146]. Formosa [54] also argues that, if we are not careful, robots that interact socially with humans can harm human autonomy by helping us achieve fewer goals and make less authentic and competent choices, thereby decreasing our trust and respect for our own autonomy. Lastly, since these embodied agents are to interact with humans, it is assumed that they should behave like humans in certain ways. It has been highlighted that the positive business impacts of conversational AI chatbots depend on customers engaging with these tools. A big factor for success is the conversational AI's likeness to the human beings it is intended to replace. Businesses and researchers need to understand what human-like characteristics and competencies should be embodied in customer-facing social AI agents to facilitate smoother user interaction [155].

I hypothesize that given relevant data (containing appropriate signals from the environment and task-space) with enough expert demonstrations in disorienting continuous control tasks, an AI agent actually develops a level of embodiment in the problem space, and can thus be trained to 1) replicate human behavior for the same task 2) guide novice-level humans in the completion of the task 3) offer corrective guidance in cases of (potential) errors. I also hypothesize that 4) the increase in similarity of human behavior also increases placed trust in the system. To design and develop such an AI system, I examine how the agent learns to perform the task while leveraging Human-AI collaborative frameworks for interaction, learning, and trust. The AI system is designed as a task-assistance system that would determine when a user needs help (either due to spatial

disorientation or other task-based goals) and provide instructions on how to recover to normal operating conditions. Ultimately, it would be up to the human user to accept or reject the how-to guidance.

Research Contributions

This dissertation will show that while AI systems can achieve high task performance, which is not surprising, this does not always translate into their ability to aid humans achieve better performance, especially in metrics representing safe behavior, but humans tend to trust an AI system's task behavior (and recommended actions) is more like that of a human's. The remainder of the document is framed as follows: (1) Chapter 2 explores related work on the definitions of embodiment from cognitive science and psychology, decision support systems, Human-AI system examples and how they learn to embody the task space as well as trust and safety requirements and finally introduce a challenging action task design to study spatial disorientation (2) Chapter 3 presents preliminary work and the insights gained on whether we can model and replicate human actions from a challenging action task, a disorienting balancing task influenced from space-flight conditions (3) Chapter 4 builds upon the preliminary work in Chapter 3 to provide real-time embodied task guidance to humans during the disorienting task and presents the findings of Human-AI study (4) Chapter 5 extends task guidance into a flight navigational task in a realistic flight simulator.

Chapter 2

Related Works

2.1 What is Embodiment?

Embodiment theory has been a prominent topic of discussion in psychology, anthropology, and cognitive science, with each providing multiple definitions. The closest definitions related to AI arise from cognitive science. Thompson and Varela [166], Varela et al. [170] in the late 1990s provide one of the earliest definitions of embodiment as a two-way process between brain and body/environment. Ziemke [200] formalizes this concept as *structural coupling* where a system embodied in an environment would take inputs from the environment, which may change the internal state of the system, and the system may also affect the environment's state, as depicted in Figure 2.1. Structural coupling implies that the system and environment are co-determined, i.e., the environment's constraints ultimately affect the system's design and capabilities [171]. In the learning sciences, embodiment entails learning approaches that draw on bodily experiences. In this field, it refers to the capacity of humans to leverage innate and developed intellectual capacities in interactive learning environments [1]. Researchers typically design activities to further develop a person's intellectual capabilities through interventions that engage sensorimotor capacities.

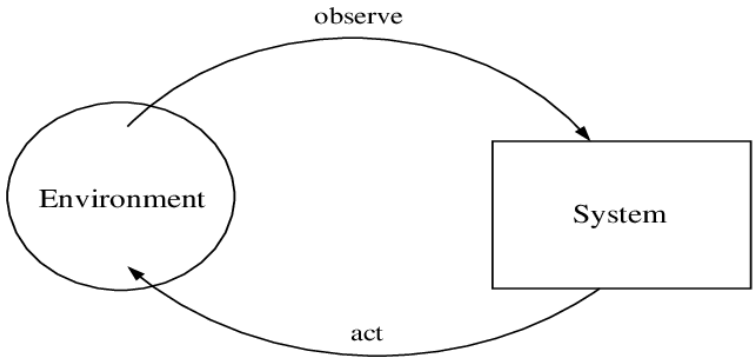


Figure 2.1: An example of a system structurally coupled with its environment. The system may observe the environment's states, which may affect its own states, and, in return, the system may perturb the environment's state [81].

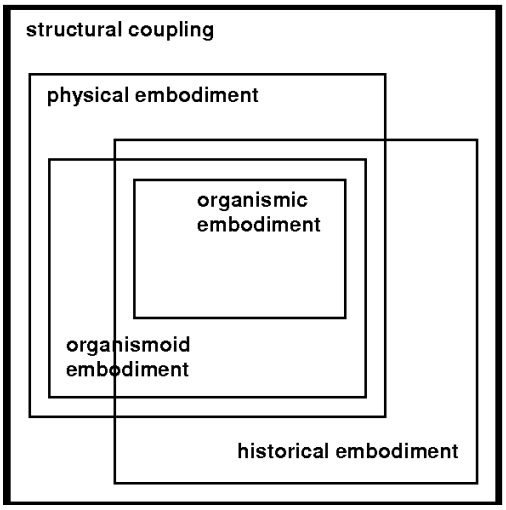


Figure 2.2: Ziemke's taxonomy for his notions of embodiment [200].

Recent approaches to embodiment focus on definitions of the form the agent or system takes, i.e., the actions an agent is capable of performing. For example: Can it move using wheels or legs, can it see using visual receptors (cameras), can it articulate speech, or interact with objects in the environment using limbs or grippers [13, 65]? Pfeifer and Scheier [137] argue that: 1) to properly define the problem of embodiment, some physical form is required for interacting with the real world; 2) for any form of intelligence, a body is required; and 3) the requirement for agents to interact with the environment is a consequence of their embodiment [29, 30]. Alternatively, in enactive cognition, the body is defined as adaptive and autonomous, rather than its physical form; i.e., regular interactions between the environment and the body adapt the agent's cognition.

Most of these definitions would fall into one of two perspectives, as noted by Ziemke [202]: 1) the physical form the agent would take (following an ***engineering*** perspective) or 2) the underlying modeling displayed by the intelligence of the agent (following a ***scientific*** perspective). He argues that each perspective tries to explain embodiment and what an embodied AI agent should be, but does so only partially. To address this issue, multiple researchers, e.g. [38, 127], have provided their own hierarchy or taxonomy of embodiment where the less-restrictive lower levels focus more on the scientific perspective of cognition, and as we move up to higher levels, the concept of a body emerges that contributes to a more restrictive, engineering-focused notion of embodiment.

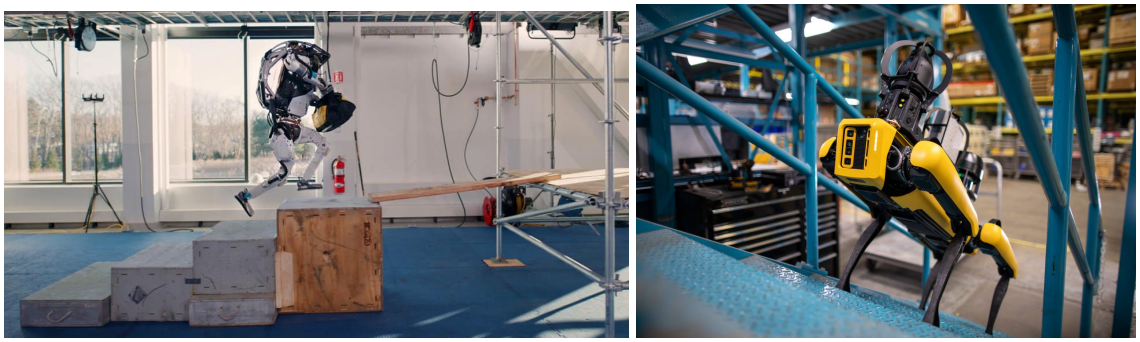


Figure 2.3: Boston Dynamics' Atlas (left) and Spot (right). An example of agents with organismoid embodiment inspired by limbs similar to those of humans and dogs, respectively.

Ziemke [201] provides 5 different notions of embodiment, neatly encapsulated into a Venn diagram, with each notion being a more restrictive set of the previous (Figure 2.2): 1. *Structural Coupling* is where an agent or system is said to be embodied if it is situated in the environment – it is the weakest and least restrictive notion of embodiment as the system may be cognitive or non-cognitive [201]. 2. *Historical Embodiment* is a reflection of repeated interactions of structural coupling between the system and the environment. For example, a humanoid robot learning to navigate a specific environment through trial and error until it learns to walk stably. 3. *Physical Embodiment* dictates that a system is physically embodied when it has representations of real-world concepts grounded into itself through sensors and actuators – aligning with cognitive systems in AI. 4. *Organismoid Embodiment* maintains that certain types of cognition are limited to systems that have bodily forms with sensorimotor capabilities similar to living organisms. Figure 2.3 depicts a few real-world examples of organismoidally embodied agents: Atlas, a humanoid robot, and Spot, a robot dog, both with limbs inspired by humans and dogs, respectively. 5. *Organismic Embodiment* is the most restrictive notion of embodiment where cognition is limited to actual organisms, i.e., living bodies such as humans who are "autonomous and autopoietic".

Ziemke later revises his notions of embodiment to include Barsalou et al. [15]'s notion of ***social embodiment*** where cues such as body posture, movement, and facial expressions that arise during social interactions play a big role in information processing, adding another layer of complexity to physical embodiment. This type of embodiment is more aligned with socially interactive robots

designed to work with humans to complete tasks and solve problems, and should therefore be considered more relevant to research on embodied agents in the human-AI collaboration domain.

Why do we need embodiment? Is embodiment of a particular kind necessary for AI and embodied agents? Does an embodied agent always require sensorimotor capacities or organismoidal forms? Or do we need AI systems to actually grow living tissue to ascertain and assert its embodiment? Chrisley and Ziemke [36] point out that for any notion of cognition or embodiment to be realized would require a detailed design of the actions performed by the system and the situated environment it lives in.

Tasks like inverted pendulum (IP) balancing are well-known use cases in reinforcement learning [16, 10, 52] that serve as demonstration benchmarks for newer continuous control algorithms like Soft Actor-Critic (SAC) [71] or Deep Deterministic Policy Gradient (DDPG) [102]. These have demonstrated proficiency in solving nonlinear control tasks, such as IP balancing, using reward signals extracted from observations of applied environmental physics. However, given the nature of environmental physics input versus sensorimotor input, I hypothesize that these AI algorithms learn to perform the task very differently from humans. In this dissertation, I explore the application of different RL and supervised learning algorithms to AI assistance in disoriented balancing.

For the AI assistants that I develop in Chapters 4 and 5, I adopt a definition of embodiment akin to Ziemke's *historical embodiment*, where a system embodied in an environment takes environmental states as input, and the system is able to learn to competently perform its tasks in its environment through repeated interactions between the itself and the environment [141, 201]. However, the types of information a system is exposed to, such as through different types of sensors, also condition the relations the system develops between itself and its environment [32], such that exposure to different types of data (or different ways of measuring the same underlying environmental state) may mean that different inputs and modeling strategies cause a model to learn different policies within the same action space, and thus may learn to perform the same task equally well through potentially radically different strategies. Therefore, I will speak of different

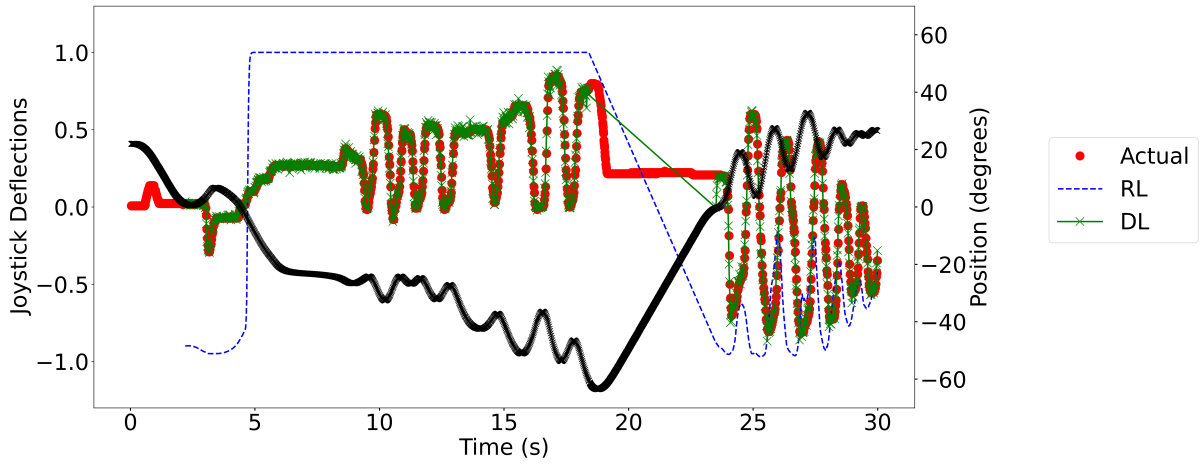


Figure 2.4: Joystick deflections predicted by AI models trained using a Deep Deterministic Policy Gradient (DDPG) through an objective reward function (blue) and a Long-Short Term Memory (LSTM) trained over human data (green) compared to an actual 30-second snippet of a participant trial from the Multi-Axis Rotation System balancing experiment (participant deflections in red and angular position in black). This instance of the LSTM displays a test root-mean-squared error of .013 while the DDPG gets .803.

"embodiments" of the task problem space as reflected through these different strategies. Here, the embodiment of AI refers to the difference in task understanding that arises from contrasting learning algorithms.

Figure 2.4 shows how this manifests in the MARS IP balancing task (see Section 2.4). "Actual" shows the deflections of a human subject—the subject balanced the MARS using small, intermittent deflections that are characteristic of proficient human performance [180]. Joystick deflections predicted by a DDPG model are shown in blue, and deflections predicted by an LSTM trained over actual human motions from other MARS trials are shown in green. The model trained on human data *embodies* the problem space similarly to a proficient human, making it a superior predictor of novel humans' small, intermittent actions, while the DDPG predicts long, continuous deflections that are more indicative of poor human performance, even though a DDPG successfully performs the task independently (see also [111]).

2.2 Decision Support Systems

Decision support systems have been a topic of interest since the 1970s with the emergence of personal computing [24, 120]. Earlier definitions of decision support focused more on (a) a sequential approach where a user would query the system and wait for a response (essentially a blocking system), and (b) targeted database management or organizational information processing for management systems [24, 88]. However, a few key (implied) notions in early decision support systems remain relevant: the system is defined by the task structure, it cognitively supports an individual's decision-making process, and it prioritizes responsive service and usable, humanly interpretable interfaces [88].

Previously, I discussed multiple aviation systems labeled as decision support systems. However, automated systems, including those explicitly advertised as "AI agents" are increasingly being used support human decision making across a multitude of domains such as healthcare, education, military and aviation applications [169, 134, 117]. Cohen et al. [39] show an examples of early models of the underlying "intelligence" of a decision aid called event trees (see Figure 2.5). An event tree, like a decision tree, enumerates the possible scenarios and outcomes for the task. The event tree is configured with both the likelihood of an event occurring (numbers on branches) and the user's trust in the aid at a given point in the tree (numbers in the circular nodes). While such frameworks are useful for modeling simpler tasks, they are difficult to scale. Thus, many researchers have explored using the flexibility of advanced artificial neural network algorithms, including transformers and LLMs, where decision support systems can benefit from real-time ingestion of large volumes of data and require complex reasoning [138, 69].

Past literature from Metzger and Parasuraman [117], Parasuraman et al. [134] and Cohen et al. [39] have shown how effective decision support systems are by using cognitive engineering constructs such as mental load - the cognitive effort required to perform a task, situational awareness - the ability to sense and understand the current state of the environment, reliance - calibrating the level of trust placed in the system for the task in order to achieve improved human performance

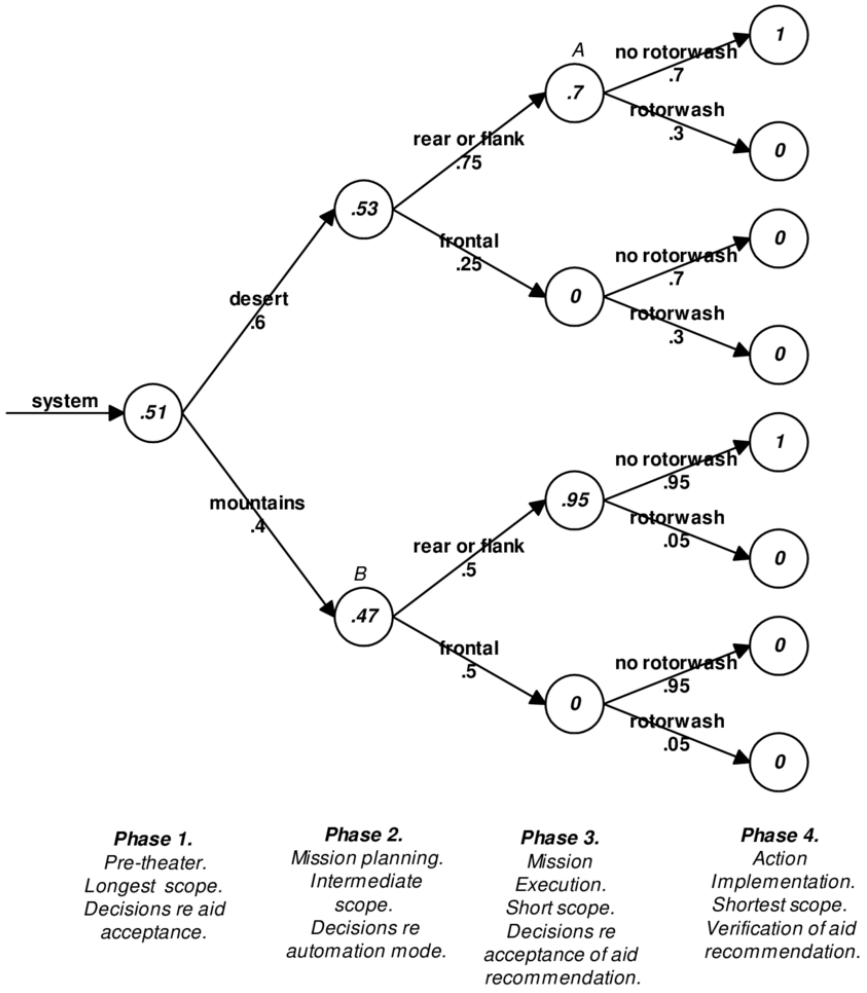


Figure 2.5: Decision aid called event trees, like a decision tree, maintains the possible scenarios and resulting outcomes that are possible for the task. The even tree is configured with both the likelihood of an event happening (numbers on the branch) and the trust a user has in the aid when arriving at a certain point in the tree (numbers in the circular nodes) [39]. In this example, the decision aid is intended to recommend a combat battle position given a set of conditions.

through empirical evidence. In any HAI system, these variables should be considered among the criteria for success, as they are key factors in system usability and adoption.

Decision Support vs. Task Guidance

Both the terms *decision support* and *task guidance* have been widely used by researchers. Zachary [195] defines a decision support system as "any interactive system that is specifically designed to improve the decision making of its user by extending the user's cognitive decision-making abilities". Ockerman and Pritchett [129] defines task guidance systems as "systems [that

are] better able to help workers take advantage of the benefits of procedures". Furthermore, the term "task guidance" has also gained traction due to DARPA's Perceptually-enabled Task Guidance (PTG) program. A task guidance system intends to provide not only decision support but also upskill novice/generalist humans, intervene when the optimal task trajectory is threatened, and provide corrections to recover from mistakes [44]. More recently, the Platform Accelerating Rural Access to Distributed and Integrated Medical Care (PARADIGM) program from the Advanced Research Project Agency for Health (ARPA-H) aims to upskill generalists with AI-guided task support to provide advanced medical care [53]. The rising interest in task guidance as both a term of art and a subclass of decision support can be attributed to its long-horizon outlook, where a series of decisions need to be made by an AI agent that is situated in the same environment as its user and interacts in real-time with the user in order to ensure that they complete the entire task successfully.

Parasuraman et al. [136] proposed 4 classes of functions where automation could be applied in interaction with a human: (a) information acquisition, (b) information analysis, (c) decision and action selection, and (d) action implementation. Most of the aviation support systems I introduced can be labeled as decision support systems, as they fall into the decision selection class of functions. The ATTOL system from Airbus, which supports autonomous taxi, take-off, and landing, can be classified as an action implementation system. Based on the definitions above and its use in the PTG and PARADIGM programs, a task guidance system provides both procedural support and includes a procedural decision support system that maintains high autonomy in decision and action selection. It would inform the user when the system believes they have made a mistake (or are about to), prompting for help. A task guidance system might also include an "action implementation support system" that provides how-to instructional guidance, i.e., the system determines the decision step and presents it to the user to accept or override. However, there are several examples of procedural decision support systems as well, such as nutrition recommendations in neonatal intensive care units [87], drug prescriptions and diagnostic support [139, 103], and forest ecosys-

tem management [128]. Moving forward, I will use the terms decision support and task guidance interchangeably to refer to AI assistance for disorienting action tasks.

2.3 Human-AI Systems

Wooldridge [191] defines an agent as an autonomous entity that satisfies its design objectives while being (a) reactive – perceiving changes in the environment and responding promptly (structural coupling), (b) proactive – pursuing the desired goal through initiative, and (c) socially interactive with other agents (humans or otherwise; social embodiment). Here I use the definition of intelligence defined by Dellermann et al. [45] "the ability to accomplish complex goals, learn, reason, and adaptively perform effective actions within an environment". These agents should not only learn from the environment but also from human preferences through interactions to adapt their actions to align more closely with human intuition and social standards [70]. Furthermore, the information and actions shared by agents should adapt to *a priori* knowledge and biases the partner human(s) may possess, where information may be abstracted away for knowledgeable/expert humans to provide succinct and prompt guidance in critical scenarios [198].

For an embodied agent to guide through interaction with humans, it must be situated in the same environment as the human, and the particulars of its interactions and information processing in that environment would affect how it embodies the problem space. Lallee and Verschure [95] introduces the H5W (How, Who, What, When, Where, Why?) framework that summarizes the key aspects an embodied agent would require to understand and reason about. The framework is illustrated in Figure 2.6, a collaborative problem-solving task in which humans work together to calculate the weight of blocks provided [89]. For an embodied AI agent to help humans complete the task, it would need to recognize and understand:

1. Who is there? The number of human subjects (subject).

2. What is there? The different blocks are the balancing scale (object).

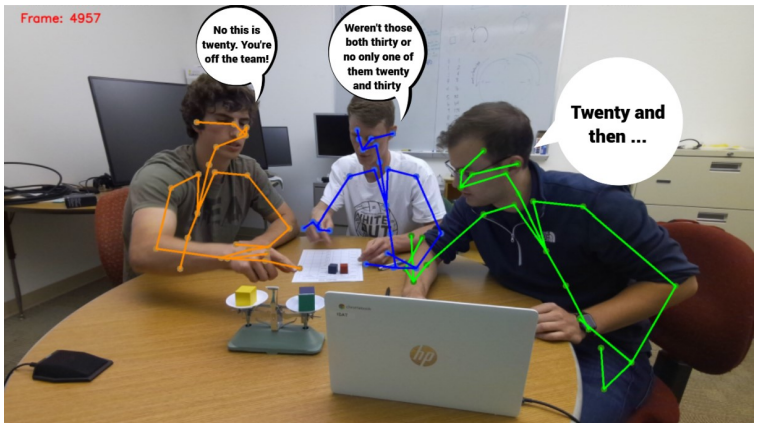


Figure 2.6: Collaborative problem-solving task where humans work together to calculate the weight of different blocks. The thought bubbles and pose estimates are highlighted as key data points that an embodied agent should capture before proceeding to help.

3. How do they behave? What are they pointing to? How do they feel? What is their posture? (action/verb).

4. When does it happen? When is a particular action taken or when is a decision reached (time)?

5. Where does it take place? The physical environment in which the conversation or task is held (place).

6. Why do they behave like this? The unspoken priors, any bias or friction between the collaborators (motivation/causality).

Currently, the hardest problem in developing situated AI for such tasks is the question, "Why do they behave like this?" Because AI agents cannot feel emotions or perceive human mental states in social settings, they face a significant handicap in this regard. To enable AI systems to behave more humanlike or better align with human preferences, researchers have developed approaches that involve humans in the learning process.

2.3.1 Human in the Loop Learning

Recent advancements in AI (NLP in particular, e.g., ChatGPT [194]) can be attributed to Reinforcement Learning with Human Feedback (RLHF; [37]), a form of human-in-the-loop (HITL)

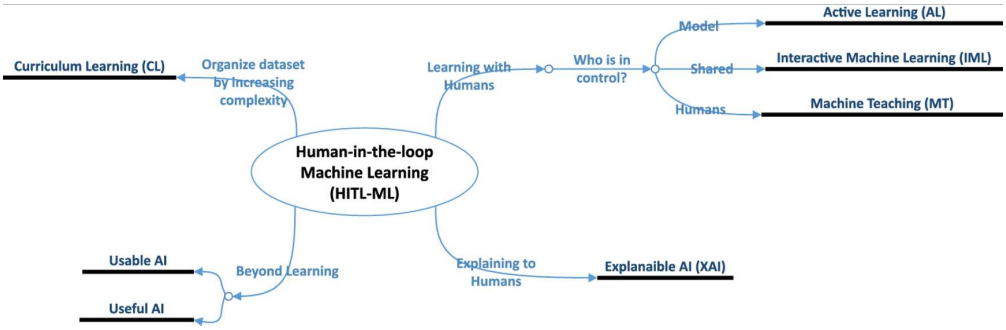


Figure 2.7: Mind map encompassing the various attributes of HITL learning [121].

ML. However, HITL is not limited to text-only language models and, in fact, has a longer history in training systems with greater levels of physicality, situatedness, and embodiment. The term HITL ML is broad and encompasses numerous concepts, ranging from learning "like" humans to learning "with" humans and beyond. Mosqueira-Rey et al. [121] provides a mindmap that neatly segments HITL-ML into 4 domains. 1) Curriculum learning, inspired by how humans learn, where we learn basic structures and then compose those basic structures to learn, incorporate, and create new knowledge and generalize better; 2) Learning with humans, where humans and AI learn and perform tasks together; 3) Explainable AI (XAI), where AI decisions should be explainable to humans when deployed in real life; and 4) Beyond learning, when AI systems and agents are analyzed for their usability and usefulness to humans. In this section, I will focus mainly on the "Learning with Humans" aspect, in particular on Machine Teaching, and on how this relates to improving the physical and in some cases social embodiment of an AI agent, where human preferences and nuances are learned directly from human-provided actions. However, the components of XAI and Useful/Usable AI should be considered important attributes of HAI systems. Not considering them would have major repercussions for AI trust and safety, as discussed in more detail in Section 2.3.2.

Machine Teaching is described as the case in which humans control the entire learning process, from the number of examples to learn to the task's difficulty. In machine Teaching, training examples are heavily curated from human demonstrations, making it particularly useful when labeled data are scarce. It helps to create the Minimum Viable Dataset required for an AI model or agent to

complete the bare minimum requirements of the task. Another advantage that may not have been deliberate is that it can make the learner (AI) more human-like in its behavior or approach to the task. A key factor in using machine teaching with social and collaborative robots is that the average human user has little to no technical background. Machine teaching should provide a simple method for novice humans to train robots to perform useful tasks from their demonstrations and to generalize to new circumstances [12, 22, 31]. Figure 2.8 depicts a task of disposal of unhealthy plants from a plant tray, and the colored ovals illustrate the target plants that have been determined to be unhealthy. Sena and Howard [154] conduct an experiment where the robot arm is able to learn from a small number of user-provided demonstrations (in cyan) using a task-parameterized Gaussian mixture model and replicate and generalize that behavior to new locations of unhealthy plants (in red).

Figure 2.8: Learning from human demonstrations on how to discard unhealthy potted plants. The targets are labeled with shaded circles, and the human demonstrations (in cyan) are provided to the agent for learning. After training, the robot's attempts at the task are shown in red.

Moreover, machine teaching has been used in developing autonomous vehicles that drive "like" humans, not only for safety and reliability but also for a comfortable user experience [47, 85, 94]. Kuderer et al. [94] conducted an experiment to learn different driving styles from real human demonstrations of autonomous driving. Behbahani et al. [18] illustrates how data from traffic

cameras can be calibrated to extract expert demonstrations and learn strategies using generative adversarial imitation learning (GAIL) to exhibit human-driver behavior in virtual simulations.

Machine teaching helps to improve the agent's physical embodiment by increasing task efficiency, especially in humanoid robots, as humans can teach the use of limbs to complete tasks (Figure 2.8). There is also the advantage of ingraining human-like behavior into the agent, as in the case of learning human driving styles from demonstrations and replicating them using reinforcement-based imitation learning. However, machine teaching has its own set of weaknesses. First, the performance of the learner (AI) depends on the teacher's knowledge and expertise; for example, if the teacher is not a reliable driver, the AI driver would not be very reliable either. Furthermore, if the MVD contains vague or ambiguous demonstrations, these will most likely negatively affect the learner AI.

2.3.2 Trust & Safety

As AI becomes increasingly used in real-world applications in multiple domains, questions have arisen about whether AI systems are designed with human users in mind. Are they fair and transparent? Are they accommodating to human control and autonomy while carrying out their own objectives autonomously? Table 2.1 illustrates the summary of Sheridan-Verplank's proposed levels of autonomy for decision selection class of functions with respect to human-machine interaction systems [136]. Their proposed automation framework has heavily influenced current human-agent interaction systems; for example, the Society of Automotive Engineers' (SAE) six levels of driving automation are also influenced by Sheridan-Verplank's taxonomy (Table 2.2) [130]. The main critique of such taxonomies is that the notion of human control disappears once the agent achieves sufficient autonomy to execute its decisions without human oversight and, in some cases, to ignore human suggestions and control. As it happens, such a scenario occurred when an autonomous system, the maneuvering characteristics augmentation system (MCAS), on Boeing's 737 MAX aircraft led to 2 fatal accidents [86]. The MCAS was designed to prevent and reduce the risk of aircraft stalling, but due to a faulty sensor, the MCAS continued to push the

nose of the airplane down, while the pilots of both airplanes counteracted the MCAS by pulling the airplane up. The lack of human control and override, combined with MCAS operating without any notification or transparency to the pilots, is concerning, to say the least.

Level		Description
Low	1	The computer offers no assistance; human must take all decisions and actions
	2	The computer offers a complete set of decision/action alternatives
	3	Narrows the selection down to a few alternatives
	4	Suggests one alternative
	5	Executes that suggestion if the human approves
	6	Allows the human a restricted veto time before automatic execution
	7	Executes automatically, then necessarily informs the human
	8	Informs the human only if asked
	9	Informs the human only if it, the computer, decides to
High	10	The computer decides everything, acts autonomously, ignores the human

Table 2.1: Levels of autonomy by Sheridan-Verplank for decision and action selection class of functions; level 1 systems would offer no assistance or automation, and level 10 systems would autonomously perform actions with no regard to human autonomy and control.

Level	Description
0	No Automation: Human driver controls all: steering, brakes, throttle, power.
1	Driver assistance: Most functions are still controlled by the driver, but a specific function (like steering or accelerating) can be done automatically by the car.
2	Partial automation: At least one driver assistance system is automated. Driver is disengaged from physically operating the vehicle (hands off the steering wheel AND foot off the pedal at the same time).
3	Conditional automation: Driver shifts "safety critical functions" to the vehicle under certain traffic or environmental conditions.
4	High automation: Fully autonomous vehicles perform all safety-critical driving functions in certain areas and under defined weather conditions.
5	Full autonomy: equal to that of a human driver, in every driving scenario.

Table 2.2: SAE levels of driving autonomy inspired by Sheridan-Verplank's autonomy hierarchy.

Shneiderman [159] critiques the influence that the Sheridan-Verplank framework has had on the design of autonomous systems as being one dimensional by design, where the system can only have human control or complete agent autonomy (Figure 2.9-top). Shneiderman argues for adding another dimension of human control to Human-Centered AI (HCAI) systems, incorporating both a high level of human control and a high level of machine automation; the resulting system could be considered reliable, safe, and trustworthy. Furthermore, other factors that contribute to trustworthy and reliable AI agents need to be recognized. Palmer et al. [131] defines dimensions of trust that arise from the use of automated systems, some of which include: (a) benevolence - the system's goals are always aligned with the user and their mission, (b) false alarm rate - that certain errors and failure states are known and acceptable beforehand, (c) perceived competence - the user believes the system handles given tasks, and (d) utility - the system adds value to the user's mission and goal.

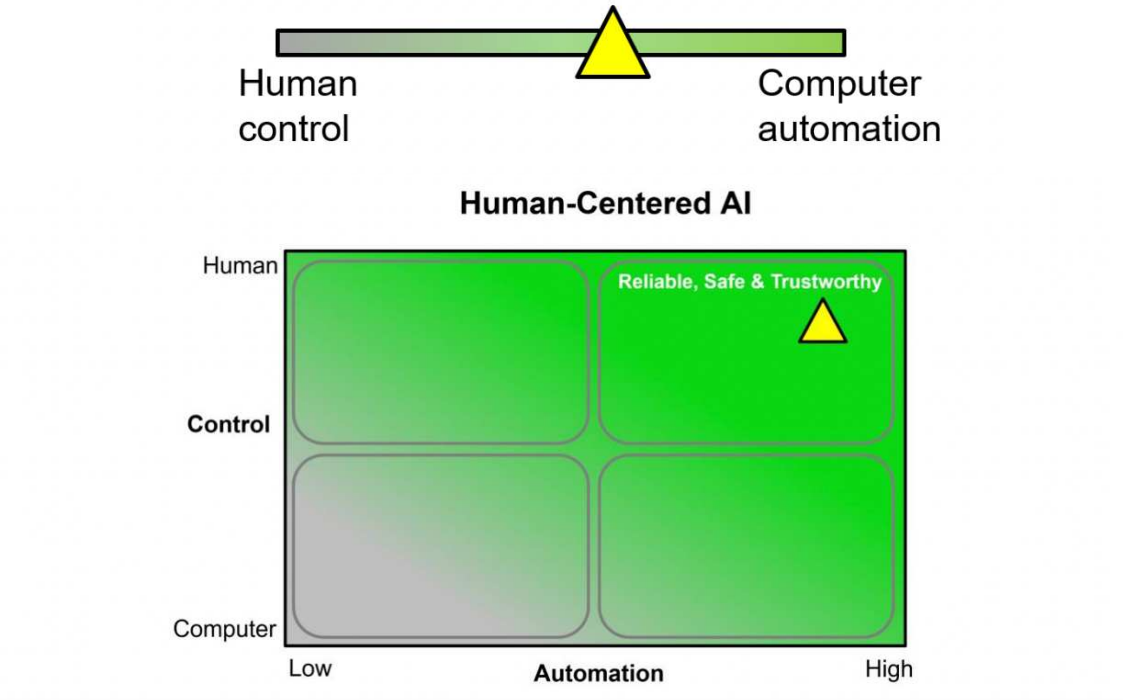


Figure 2.9: Dimensions of shared human-AI autonomy and control. The top image illustrates how most AI systems offer a 1-dimensional level of human control or machine automation. In the bottom image, Shneiderman proposes AI systems that should be designed so that human control should not be sacrificed for machine automation.

2.4 Spatial Disorientation and Balance

As we saw before, reinforcement learning has a well-known solution for the IP balancing task with reproducible benchmarks. But the way an RL agent would solve the task might be impossible for a human to follow or imitate. In the balancing tasks discussed in this section, evidence suggests that humans, and especially experts, prefer to make small, subtle, and intermittent actions [180]. Humans who display worse task performance tend to follow strategies similar to those of the RL agent, alternating between repeated and extreme actions, but never achieve comparable performance. Differences in task performance, due to differing task embodiments, suggest that to improve task performance in humans, an AI assistant should recommend actions that the human can actually adopt and execute in a timely manner. The design of the AI agent can also influence how it should be used; an RL agent that always predicts the next state given inputs might be a better aid when human control needs to be overridden. A supervised learning agent, such as an LSTM, can be trained to predict actions humans need to take X seconds into the future, allowing humans to react and incorporate the suggestion into their strategy. The alignment of the assistant's suggestions also plays a role in the trust humans place in the aid; repeated, alternating suggestions that the human cannot incorporate into their strategy might lead to task failure and thus damage trust in the assistant [116, 183].

Using decision support systems as a countermeasure against spatial disorientation is a new research direction. Multiple studies will need to be performed to investigate (a) The utility of the decision aid such that it aids and supports the task goals and does not counteract them; (b) When should the system intervene and provide suggestions, as untimely or unneeded suggestions might increase user frustration and harm skill and trust [161, 39]; (c) How much control should be shared between humans and the system? Ideally, a situation such as the MCAS fault should never occur. Humans should be in control of the vehicle and use the decision aid until sufficient trust has been established to enable increased automation of the decision support.

To study spatial disorientation, Panic et al. [133], Vimal et al. [177] explored the relevance of the balancing tasks to the perception of gravitational cues during unstable vehicle control to

investigate the causes of spatial disorientation and strategies for prevention. This work is inspired by the inverted pendulum (with the center of mass above the pivot point), a common model of human upright balance in the study of postural dynamics [144].

Human subjects were strapped into a **Multi-Axis Rotation System (MARS)** device programmed with inverted pendulum (IP) dynamics and instructed to stabilize themselves about the direction of balance (DOB) using a joystick. Subjects were blindfolded because the risk of spatial disorientation-related accidents is heightened when visual information is limited [107, 163]. Typically, humans rely on gravitational cues when balancing, which are detected by the vestibular and somatosensory systems as participants tilt away from the gravitational vertical; however, in spaceflight conditions, gravitational cues are not reliable. To create a disorienting spaceflight analog condition, Vimal et al. [174, 175], Panic et al. [132] placed participants in the Horizontal Roll Plane, where they were always perpendicular to the gravitational vertical and no longer tilting relative to it. 90% of participants reported spatial disorientation and in data 100% exhibited characteristic positional drifting [175]. Participants showed minimal learning and frequent "crashes" (reaching pre-programmed $\pm 60^{\circ}$ boundaries, after which the MARS automatically reset to the DOB).

In the **virtual inverted pendulum (VIP)** paradigm, an analog of the MARS, the same physics are programmed, and the subject balances a visually simulated circular array of dots (random dot kinematogram, RDK) that rolls in the plane of the display screen. This is visually rendered for humans and can be directly actuated by an algorithm. In the disorienting VIP condition (similar to the Horizontal Roll Plane in the MARS), the RDK is 50% coherent: alternating subsets of dots displace coherently across consecutive frames while the other half jump randomly. This eliminates configural displacement cues relative to the upright DOB while providing low-level retinal motion cues. Similar performance degradations occur between MARS upright vs. supine and VIP 100% vs. 50% coherence conditions, even with practice, as evidenced by an increased number of crashes and more frequent destabilizing actions (Figure 2.10).

2.4.1 MARS & VIP Tasks

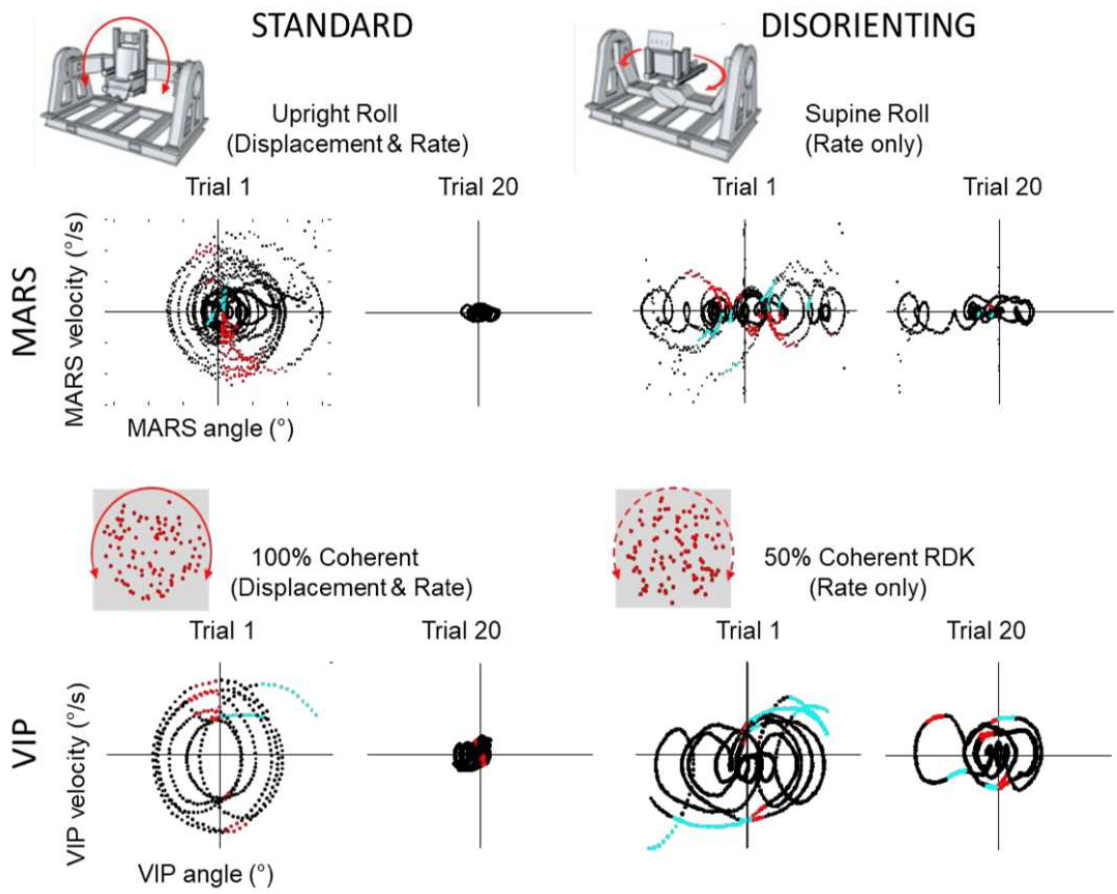


Figure 2.10: Typical performance in the Multi-Axis Rotation System and Virtual Inverted Pendulum tasks, before practice (Trial 1) and after practice (Trial 20). Phase plots show angular velocity versus angular displacement relative to the direction of balance (DOB). The "standard" conditions provide angular displacement and velocity cues, and subjects improve significantly between the first and last trials, as seen by clustering around the origin (balance point) by Trial 20. The "disorienting" conditions eliminate sensory signals of displacement from the DOB, increasing positional drift (as shown by phase loop oscillations around the X-axis) and destabilizing joystick commands that accelerate away from the DOB in the current direction of motion, with minimal learning and continued positional drift in Trial 20. Cyan dots indicate destabilizing deflections, where position, velocity, and joystick deflection all have the same sign. Red dots denote *anticipatory* deflections, where position and joystick deflection have the same sign but velocity has the opposite sign—usually done to slow the Inverted Pendulum down when velocity is perceived as being too high.

Figure 2.10 compares the MARS and the VIP paradigms. Both can be configured in challenging but non-disorienting modes, with standard sensory information, or difficult and disorienting

modes, with degraded information. In both disorienting conditions, subjects exhibit the same characteristic, drifting and a lack of learning, when compared to the coherent conditions.

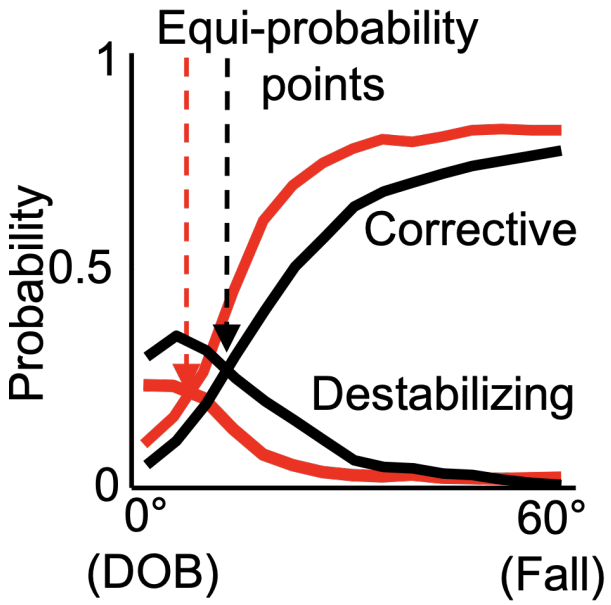


Figure 2.11: Complementary evolution of discrete destabilizing and corrective commands as a function of angular deviation away from the DOB and toward a fall boundary, seen in MARS (red) and VIP (black) tasks.

The same performance characteristics are exemplified in both MARS & VIP, as defined by subject matter experts in the neuroscience of balance dynamics [180]. In spaceflight, the pilot has cues about their motion but no orientation to gravity. The VIP 50% coherent reflects this by providing motion cues without configural orientation cues. The VIP & MARS tasks possess similar underlying physical models of instability and in both tasks, the probability of destabilizing commands decreases, and corrective commands increase as the pendulum position crosses the DOB and moves closer to a fall boundary (Figure 2.11). This shows that the VIP task is demonstrably analogous to the MARS task, which in turn is demonstrably analogous to spaceflight conditions. Importantly, these action distributions arise from human task performance and reflect a particular human action policy grounded in sensorimotor perception, an example of structural coupling.

Data Collection

MARS Wang et al. [184] released MARS human performance data from 34 healthy adults (18 female, 16 male). Each subject participated in two experimental sessions on consecutive days, each comprising 20 100-Section trials, during which they attempted to balance themselves with minimal oscillation while blindfolded. The data contain angular positions and velocities, as well as joystick deflections, sampled at 50 Hz.

VIP The VIP data consists of 31 healthy adults (22 female, 9 male). Subjects participated in 12 30-second trials in a single disorienting-condition session, with the same goal as in the MARS task. Angular pos./vel. and joystick deflections were sampled at 200 Hz.

Performance Evaluation

An ideal performer in both tasks would immediately rotate to the balance point and remain there with minimal motion. Calculable metrics from the collected data include number of crashes (excursions beyond $\pm 60°$), proportion of destabilizing deflections (*% destab.*—see Figure 2.10), mean/standard deviation of angular position θ, average magnitude of velocity ($\mu|Mag|_{vel}$), and root-mean-square (RMS) velocity. Lower metric values usually mean improvement, e.g., fewer crashes, more time spent near the DOB, less oscillation, slower motion, smaller deflections, etc.

MARS Vimal et al. [180] used a Bayesian Gaussian Mixture method and the aforementioned features to cluster subjects into 3 statistically distinct groups that represent Proficient, Somewhat-Proficient, and Not-Proficient performance (hereafter *Good*, *Medium*, and *Bad*). They use these clusters to partition the MARS data into training subsets and characterize digital twin performance on the task.

VIP Plotting the RMS velocity of VIP subjects vs. crash frequency revealed a positive linear relationship ($r = .73$). Based on the number of crashes in the 12^{th} trial and crash reduction between Trials 1 and 12 (i.e., final performance and overall improvement), they assigned participants rela-

tive rankings and divided them into tertiles to mirror the Good/Medium/Bad MARS classification. VIP participants typically exhibited more crashes, destabilizing actions, and higher RMS velocity than MARS participants of equivalent proficiency. These factors and differences in sample rate and environment enable us to assess the transferability of AI assistants to novel digital twins.

2.5 Summary

I discussed the various definitions and notions of embodiment by Ziemke and other prominent authors, and defined the ideal embodiment that an AI agent should possess. I also propose a notion of task embodiment that concerns how an AI learns to perform a task, depending on the learning algorithm. Differences between decision support and task guidance systems are explored; the former targets reducing the cognitive workload during decision making, while the latter goes a step further to enforce procedural compliance and correct errors based on decisions taken. Furthermore, I explore the basic requirements of embodied AI agents in HAI systems and methods to improve physical and social embodiment, where such agents could serve as support and guidance systems within an HAI framework. HAI systems need to be rigorously evaluated using mental load, situation awareness, trust, and safety as metrics, so that humans can reliably incorporate AI agents in general and, more importantly, for specialized tasks; AI decisions should be explainable while maintaining human autonomy over AI automation. Lastly, I introduce the MARS and VIP tasks, designed to study spatial disorientation and balance, which I will use to train foundational AI assistants to mitigate spatial disorientation and improve task performance.

Chapter 3

Preliminary AI Guidance with Natural Language

This chapter is based on the paper "Where am I and where should I go? Grounding positional and directional labels in a disoriented human balancing task" [109] published at the 2022 CLASP Conference on (Dis)embodiment.

Much of the recent success in AI can be attributed to the meteoric rise of large language models (LLMs), such as BERT [46] and the GPT family [142]. These language models facilitate coherent, grammatical text generation by using high-dimensional representations of words, sentences, and other entities that preserve similarity relations across dimensions. Although pretrained on an enormous amount of text, there are many ways in which they fail to demonstrate "understanding" as commonly defined. As argued by, e.g., Bender and Koller [21], these models lack knowledge of the current situational context, because that context comes from non-textual modalities. Certain multimodal language models, e.g., multimodal BART-large [100] appear to perform better according to certain benchmarks [119, 92], but there remain many important domains which, for the moment, appear to be out of reach for state-of-the-art AI.

Consider the problem of human spatial disorientation. During extreme conditions, such as piloting a spacecraft, even expert humans are subject to gravitational transitions where they may not be able to rely on gravitational cues sensed by the vestibular system, leading to fatal accidents [158, 42]. Even on Earth, the leading cause of fatal aircraft accidents in military pilots is spatial disorientation [62]. Numerical AI models, however, with direct access to quantitative information about position and movement, can potentially determine when a human appears to be losing control and intervene, for example, by instructing the human to right themselves. A successful AI partner that counteracts human disorientation to enhance task performance in real time would need to predict the intent of the human's motions, make decisions with incomplete information or under environmental uncertainty [185, 164], and, perhaps most importantly, foster trust in the human [76]. These are not requirements that even the impressive benchmark performance of

modern LLMs can meet. Successful guidance of a human through language requires that the AI "embody" relations between linguistic terms and the human's situation.

In this chapter, I combine disambiguated and contextualized linguistic embeddings [188] from BERT with embeddings extracted from numerical AI models trained to predict control movements and human performance in a spaceflight-analog disoriented-balancing task. Unlike the BERT embeddings, these latter embeddings are "situated," in that they come from models that are trained to *embody* a human participant's position in a phase space parameterized by angular position and velocity in the balancing task. This combined model is trained to predict the direction the human should move towards for better balance, given BERT embeddings that represent "thought vectors" about position relative to the balance point, and performance and motion control features extracted from the numerical models. I show that predictions made by the model "agree" on average with those made by a human with a moderate level of proficiency in the balancing task, and a deeper dive into misclassifications suggests that the model may actually be performing better in this task than the raw numerical results indicate.

3.1 Background

This chapter brings together research in two distinct and to date largely disjunct areas: multimodal language grounding through human-AI collaboration, and mitigating the effects of spatial disorientation. The Collaborative Research Center's Situated Artificial Communicator project was a significant early attempt to model the integration of language and sensorimotor skills in a situated context [145]. Recent work in multimodal conversational modeling has continued similar lines of research with multimodal Transformer architectures [34, 79]. Other relatively recent work attempts to integrate neurally-encoded robotic arm control with guidance and instruction through dialogue [157, 156].

Alomari et al. [8] use unsupervised learning for concepts such as colors, names and activities by an autonomous robot. Alomari et al. [9] combine PCFG trees and visual feature clustering to ground video depictions of actions to linguistic labels. Ilinykh and Dobnik [82] find that language

models in a multimodal task setting learn different semantic information about objects and relations cross-modally and unimodally (text-only).

Importantly, though, these lines of research subsume all grounding and multimodality under combinations of language and *vision*, to the exclusion of other channels, and where AI and humans interact, the interaction focuses on humans guiding AI, not AI assisting humans. This work brings in modal channels directly related to human motion in a situated environment to train an AI that ultimately assists humans in mitigating spatial disorientation.

While there is a wide and varied body of research from the neuroscience and biomechanics communities on other modal information channels, such as human spatial awareness, AI has largely not been applied here. Recent work in this line of research has begun to use machine learning and AI techniques, providing a path forward to integrate the two aforementioned broad areas Vimal et al. [180] group subjects performing the balancing task in the horizontal roll plane (HRP), without any gravitational cues, into performance proficiency categories using a Bayesian Gaussian Mixture model. Wang et al. [184] use the same data to train a stacked gated recurrent unit (GRU) model to predict the occurrence of crashes (where crash boundaries are set to ± 60 from the balance point) 800ms in advance. This work extends the line of research toward modeling human behavior in the balancing task, enabling AI to predict and counteract disorientation.

3.2 Dataset

I use data and performance proficiency labels from Vimal et al. [180] which are further explained below. Additionally, I further annotate the data with grounded positional annotations and directional labels for training an embodied AI classifier that predicts the optimal direction of movement.

3.2.1 MARS Data

The data is collected from 34 consenting healthy adult participants (18 females and 16 males, $\mu \approx 20.4$ years old, $\sigma \approx 2.0$ years) with no prior experience in the Multi-Axis Rotation System (MARS) (see Section 2.4.1).

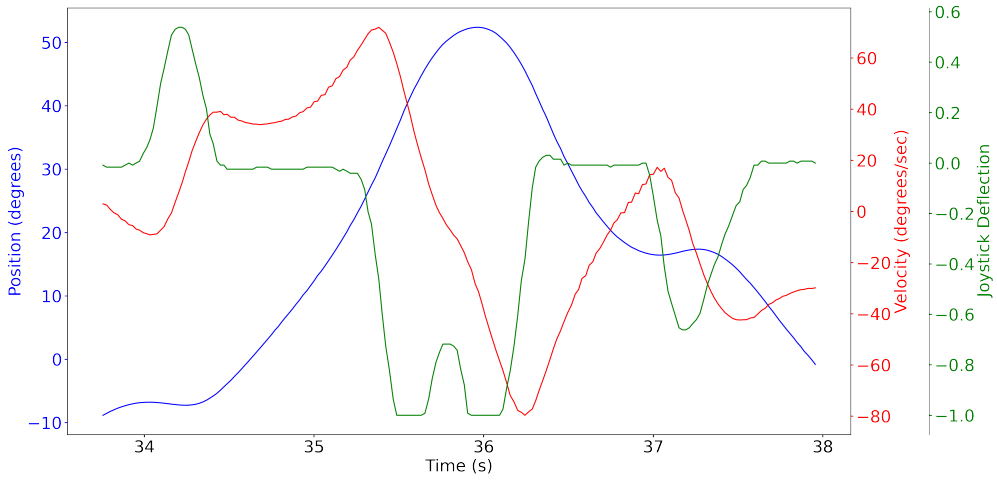


Figure 3.1: A segment of trial data from a medium proficiency participant showing angular position (blue), angular velocity (red) and joystick deflection (green). The participant barely prevents a crash as the MARS angular increases to +50° from DOB.

Figure 3.1 shows a segment of trial data from a representative participant showing changes in angular position (blue), angular velocity (red) and joystick deflection (green). I can see that this participant was able to just barely avert a crash as the MARS angular position reached +50°, or 10° from the crash boundary. Along with the numerical data, I also leveraged the proficiency labels they provide namely *Good, Medium & Bad* to train a proficiency classifier introduced later.

3.2.2 Positional & Direction Labels

To ground the situated numerical features from the MARS to a linguistic representation, I annotated the numerical features with sentences that represent position relative to the DOB, or simply put, with possible answers to the question “where am I?" given the numerical features. For example, if they are far off to the right of the DOB, a human may think “I have drifted more

towards the right" or if they think they are balanced near the DOB the equivalent thought may be "I think I am somewhere in the center". These sentence annotations were generated by third-party annotators for each of the three regions; ***left*** ($< -20^\circ$ from the DOB), ***right*** ($> +20^\circ$ from the DOB), and ***center*** (within $\pm 20^\circ$ of the DOB), within a total possible range of $\pm 60^\circ$.

For the direction labels, representing the direction towards which the human should move the MARS (or deflect the joystick) for better balance about the DOB or "where should I go?", I again divide it into three categories; ***left***: deflect the joystick with such amplitude that it prompts the MARS to the left, ***right***: deflect the joystick with such amplitude that it prompts the MARS to the right, and ***center***: deflect the joystick with as little amplitude as possible such that there is little to no change in the position of the MARS. These are discrete, one-hot vectors depicting the "where I should be going" grounded label, and are assigned based the joystick deflection made after the look-ahead time. The direction labels are defined as *left*: < -0.2, *right*: $> +0.2$, and *center*: between -0.2 and $+0.2$. +1 and -1 represent full deflection.

3.3 Methodology

The goal is to combine representations of motion and performance proficiency, which are learned from data directly capturing human embodiment during the MARS balancing task, with linguistic representations of the position and directional concepts involved. A successful model is one that can predict the label for the best direction of motion given the current circumstances by learning correlations between motion, proficiency, and linguistic representation.

The model architecture, shown in Figure 3.2, can be divided into five parts: (1) **data preprocessing**; (2) a **joystick-deflection predictor** of immediate future action; (3) a **performance proficiency classifier**, which provides a high-level view of the subject's task performance; (4) **BERT annotation embeddings**, which provide real-valued semantic representations that the outputs of previous two modules are correlated to, and (5) the combined model, or **embodied direction classifier** (EDC).

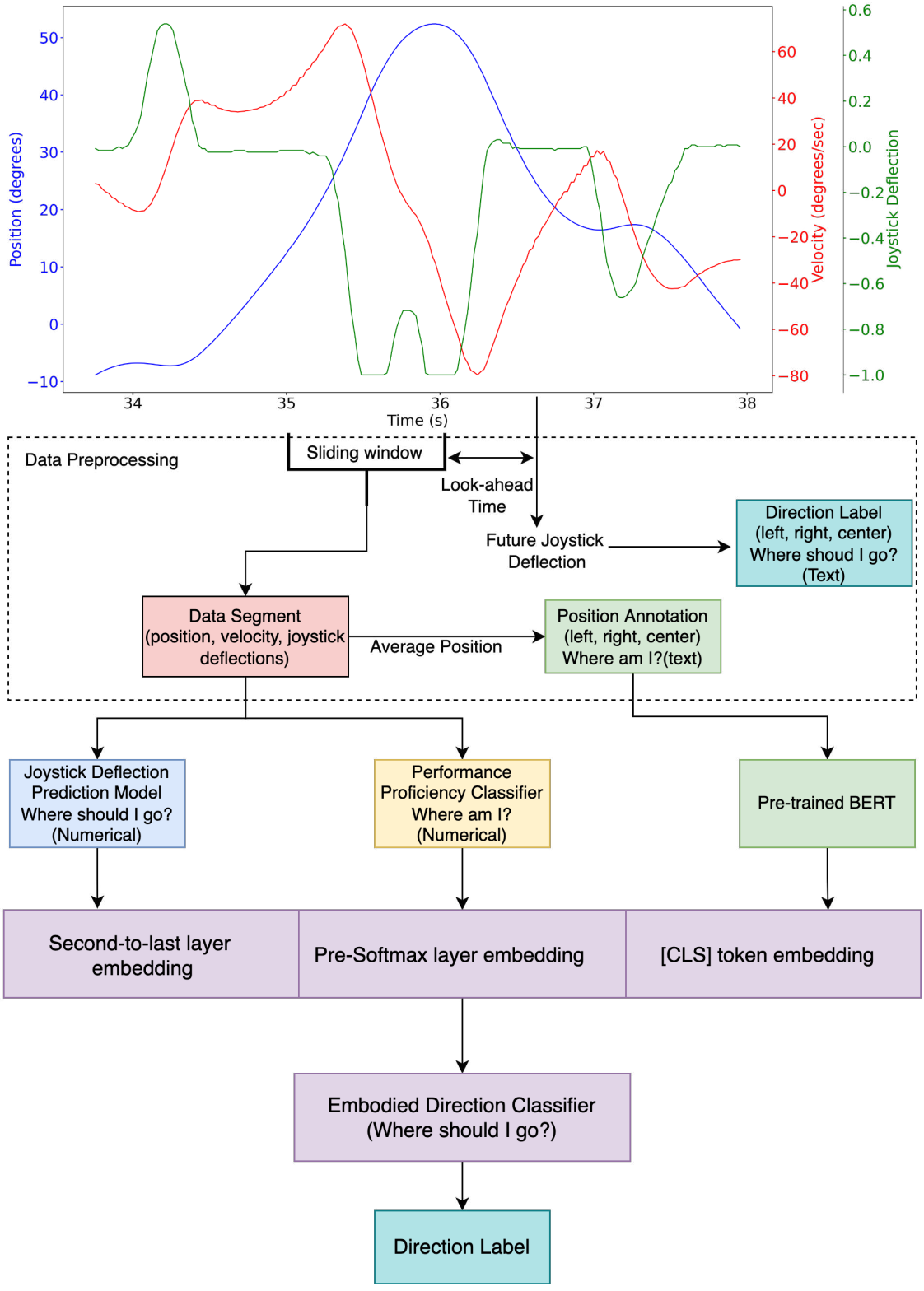


Figure 3.2: Overview of the embodied model architecture.

3.3.1 Data Preprocessing

For each trial in the data, I use a fixed sliding window technique to extract segments of joystick deflections, angular velocity and position where the user was in control and no crashes occurred for the given look-ahead time y seconds in the future.

For each viable window extracted, I assign a random sentence annotation for the region corresponding to the user's average position in the window, e.g., "I think I am somewhere in the center" or "I have drifted more towards the right."

The processed data has two parts for each sample: (1) the MARS machine features, i.e., joystick deflections, position, and velocity, and (2) the grounded position annotations.

3.3.2 Joystick-Deflection Prediction Model

Using the processed data on angular position, angular velocity and joystick deflections, I train a deep feedforward neural network model (see Section. 4.3 for hyperparameters) to predict how much the joystick should be deflected to keep the user balanced. Inputs are the 1000ms segments of joystick deflections, positions and velocities, and target values are the joystick deflections made 400ms in the future. Essentially, once operationalized, this model should tell how a user should deflect their joystick to balance themselves[3].

3.3.3 Performance Proficiency Classifier

To account for how well a user is performing the balancing task, I build a neural performance classifier that is able to tell us the user's ability to discern and gauge where they are in terms of position and where they should go. The proficiency labels are obtained from Vimal et al. [180] (described in Section. 2.4.1). I train a deep feedforward neural network model (see Section. 4.3 for hyperparameters) using the same inputs as those to the Joystick-Deflection Prediction Model (Section. 3.3.2). However, here the target labels are discrete proficiency labels of the participant for each sample in turn; ***Proficient***, ***Somewhat Proficient***, and ***Not Proficient***. This model should

[3]400ms is slightly below the reaction time of average humans [123] and well above the reaction time of trained pilots [23].

output a proficiency label for each segment, reflecting how proficient the participant is behaving at that time. The final pre-classification layer of this model outputs embeddings that are situated within the task phase space of the task by preserving high-dimensional similarity relations between actual direction and velocity values and task proficiency.

3.3.4 BERT Sentence Embeddings

I use pretrained BERT to produce the pooled sentence embedding (the embedding of the [CLS] token) for the the position annotations for each sample. This natural language representation serves as a rather literal "thought vector," representing the "where am I?" grounded positional label input to the embodied directional classifier.

3.3.5 Embodied Direction Classifier

The task now is to combine the numerical models derived from embodied human performance with the linguistic representations from BERT and train an embodied direction classifier that grounds the linguistic representation in circumstances described by the numerical data.

I combine the three aforementioned models to build a classification model that essentially embodies the operational physics of the disorienting balancing task through human performance data, and has grounding annotations in positional language ("where am I?"). This classifier takes these inputs to predict the grounded directional label, "where should I go?" for better balance.

Input to the EDC is threefold. **Joystick-Deflection Embeddings** are extracted for each sample from the penultimate layer of the Joystick-Deflection Prediction Model. These vector embeddings represent how much and in which direction the user should deflect their joystick to maintain balance. **Performance Embeddings** are also extracted from the pre-softmax layer of the Performance Proficiency Classifier to represent how well the user can gauge their position and direction. Finally, the **BERT Sentence Embeddings** for the positional thought vectors are extracted. For each sample, these three vector embeddings are concatenated and passed to the model.

The EDC is trained to predict the grounded directional labels, i.e., *left*, *right*, and *center*, which represent the "where should I go?" aspect in the balancing task. In operation, this would serve as a

cue to guide a human participant through linguistic instruction to either deflect the joystick to the left, deflect it to the right, or do nothing. Here, I assess the performance of the model and compare it to that of humans.

3.4 Evaluation

I randomly selected 12 participants from the dataset—4 participants of each proficiency. I use 38 of each participant's 40 trials for the train set and 2 for the test set. As described in Section 3.3.1, I use a sliding window of 1000ms and a look-ahead time of 400ms. After data processing, I end up with about 1.7 million training samples and 80,000 testing samples, for a $\sim$95:5 train-test split.

All neural networks have 3 layers (100 units each, $tanh$ activation), and are trained with Adam optimization for 50,000 epochs. The Joystick-Deflection Prediction Model was trained with MSE Loss and both the Performance Proficiency Classifier and EDC were trained with Cross Entropy Loss and a final softmax layer. To evaluate the performance/competence of the EDC I examine:

1. How well the model performs on average and for each proficiency group.

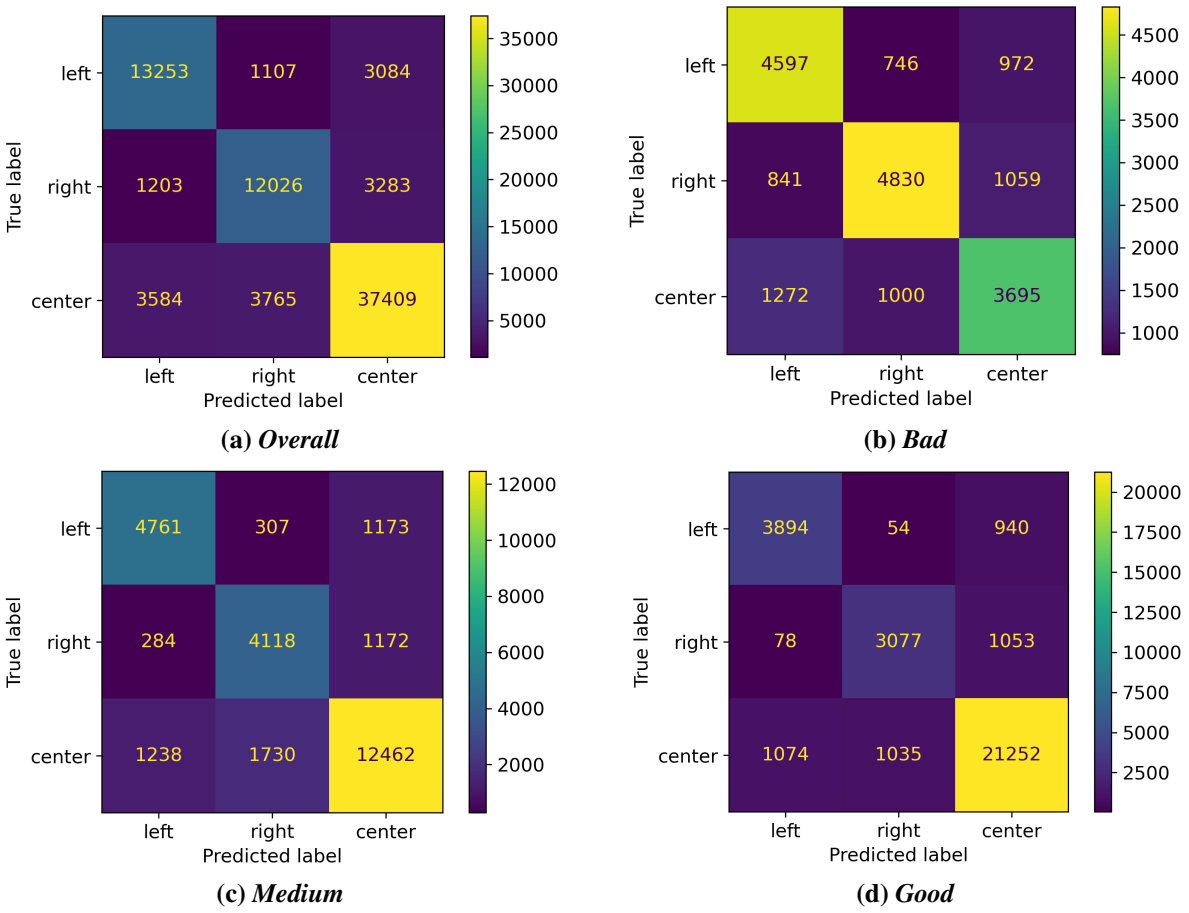
2. Misclassified samples where the model "disagrees" with the apparent ground truth, or the decision the human participant had made.

3.5 Results

Table 3.1 illustrates the performance of the EDC overall and for each of the three proficiency groups. I also show the EDC's precision, recall, and F1 score for the three target labels, i.e., *left*, *right*, and *center*. Here, a "correct" answer is one where the human participant made the correct movement choice with respect to their angular position and velocity, and the model predicted the same movement choice.

		Overall	*Bad*	*Medium*	*Good*
Prec.	LEFT	73	69	76	77
	RIGHT	71	73	67	74
	CENTER	85	65	84	91
Rec.	LEFT	76	73	76	80
	RIGHT	73	72	74	73
	CENTER	84	62	81	91
F1	LEFT	75	71	76	78
	RIGHT	72	73	70	73
	CENTER	85	63	82	91
Acc.		80	69	78	87

Table 3.1: EDC performance as %.



(a) ***Overall*** **(b)** ***Bad*** **(c)** ***Medium*** **(d)** ***Good***

Figure 3.3: (a) represents the confusion matrix for the full test set of the EDC. (b), (c), and (d) are broken down by proficiency group over the same test set.

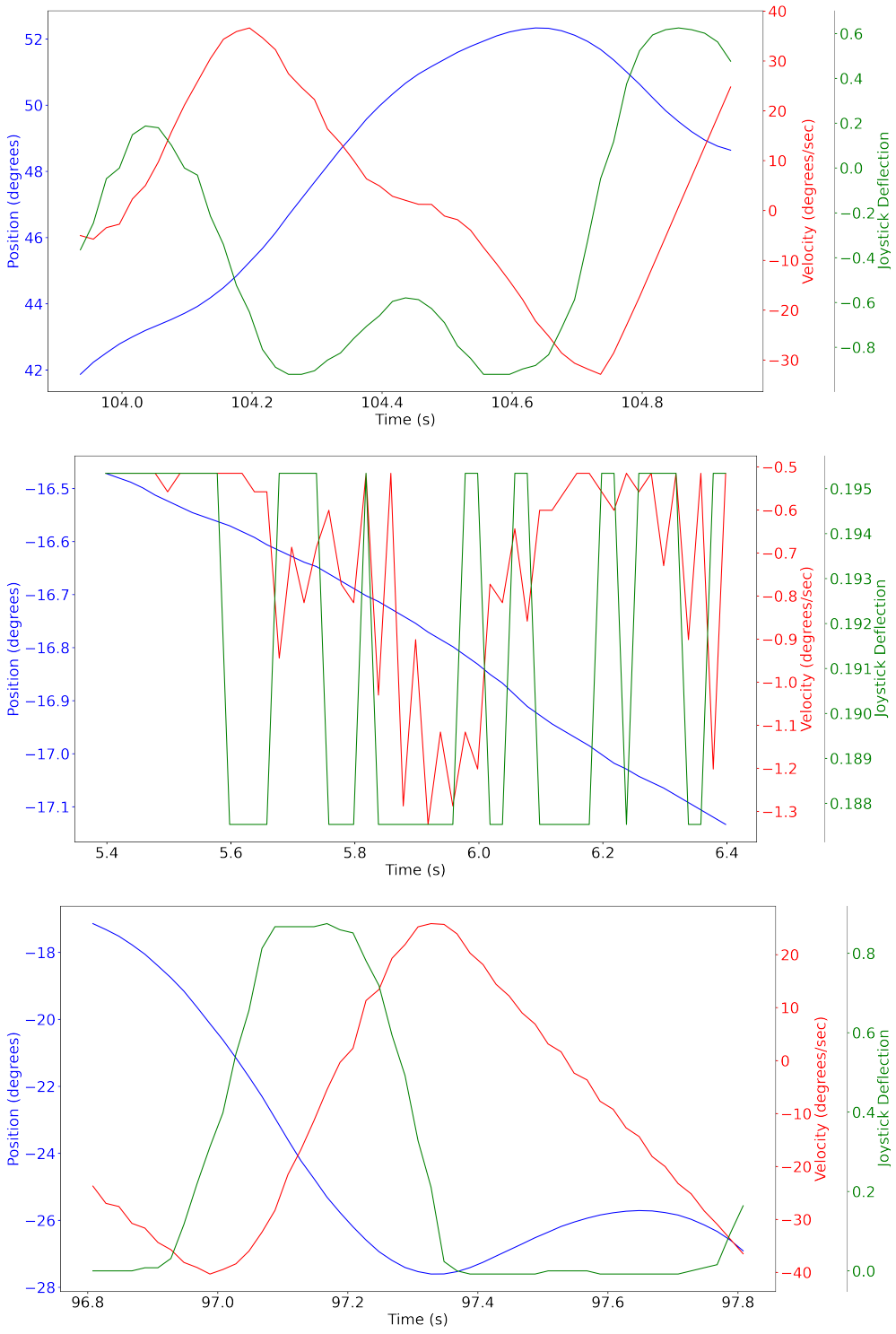


Figure 3.4: Misclassified test samples from each proficiency group (following conventions from Figure. 3.1). Top: Bad participant in the right region, truth label *center*, predicted label *left*. Middle: Medium participant drifting toward left region, truth label of *center*, predicted label *right*. Bottom: Good participant in the left region, truth label *center*, predicted label *right*.

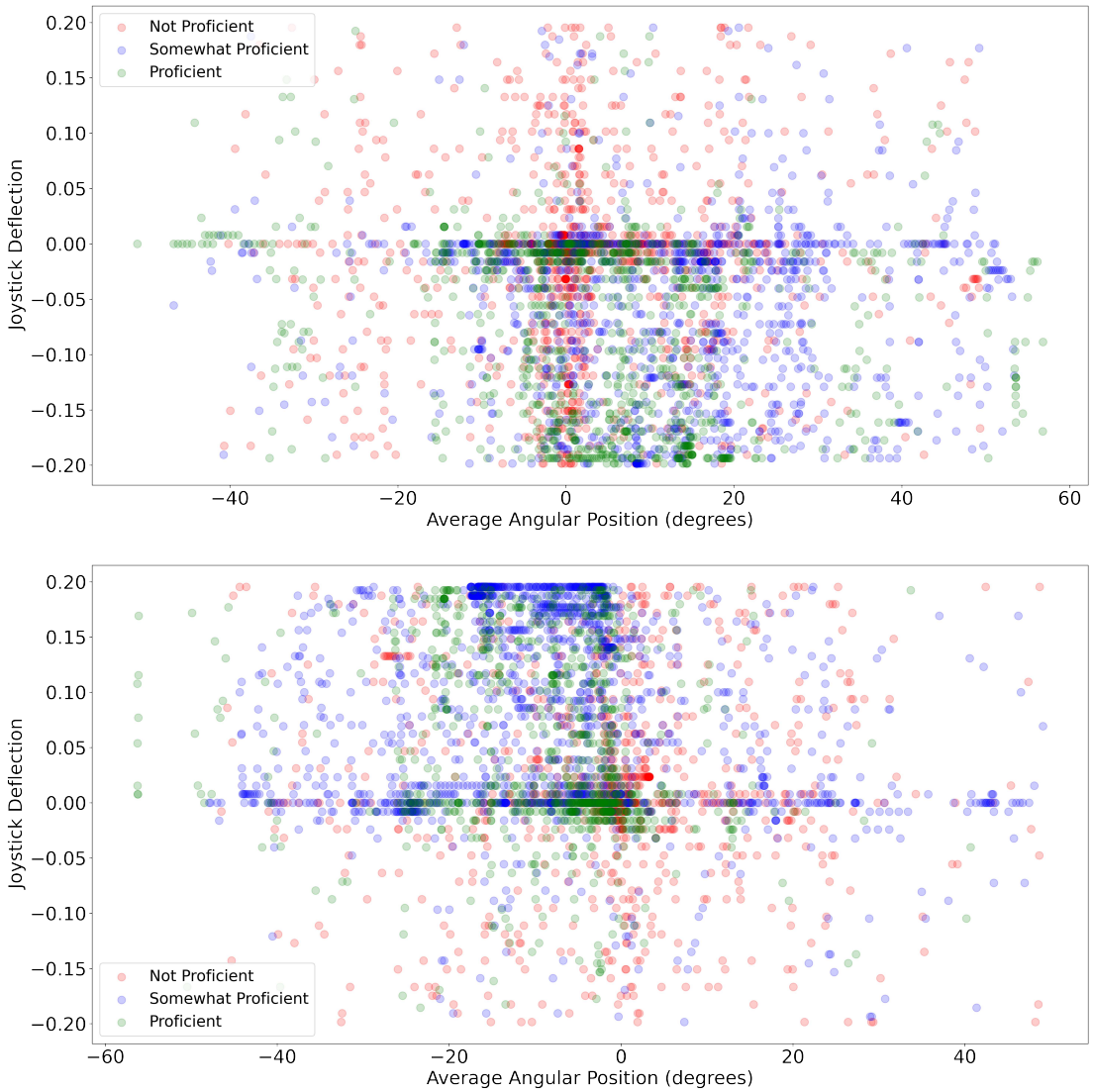


Figure 3.5: Misclassified test samples where the ground truth labels were center but predicted as *left* (top) and *right* (bottom), showing the spread of actual joystick deflection vs. sample average position when the EDC "disagrees" with the participant's movement.

3.6 Discussion

3.6.1 Proficiency Breakdown

In Table 3.1, we can see that the EDC's performance increases as the proficiency of the participant increases. The Bad proficiency group shows lower performance on correctly grounding the center label, i.e., these participants think they are in the center region, but the model thinks otherwise. They do appear to have a better understanding of whether they are in the left or right

region and balance themselves accordingly. The Medium and Good proficiency groups have a better understanding of where they are in the problem space than the Bad group, especially when the participants think they are in the center region. For the Good proficiency group, the EDC had an F1 score of 91% for the center label, which means that the model agrees with their decision to do nothing drastic when they are in the center region roughly 91% of the time. This is likely due in part to the fact that many Good (or proficient) participants are able to remain balanced within the center region for most of their trials.

Figure 3.3 provides a deeper insight into what kinds of samples are commonly confused with each other by the EDC. Regardless of proficiency group, the center labels are more often misclassified as left or right than the reverse. This is likely due in part to there being more center labels in the dataset overall (due to Medium and especially Good participants successfully keeping themselves balanced); however, the confusion matrices further validate the performance of the model for each of the three proficiency groups: the Bad group has the most confusions, and the Good group has the least. The EDC is able to combine the embodied numerical and language representation channels and determine that when a person is in the central region, they should not attempt to move out of it.

Bad participants, meanwhile, spend $\sim$72% of the time moving either left or right (for the correctly classified samples) whereas Medium and Good participants spend an average of 42% and 25% of their time, respectively, moving left or right. The rest of the time is spent making slight, intermittent movements to remain in the center. They do better at avoiding destabilizing deflections, which the EDC detects and outputs as directional labels that describe precisely that. The model, which is trained on data from all proficiency groups, makes decisions that align, in aggregate, with those of a Somewhat Proficient participant.

3.6.2 Analysis of Misclassified Labels

While the overall metrics for the EDC's performance are promising, and it performs particularly strongly on Good participants, those numbers do not tell the whole story. Figure 3.4 shows

one sample from each proficiency group that has a ground truth label of *center* but is predicted as *left* or *right* by the model. Figure 3.4 (top) shows a participant from the Bad proficiency group positioned in the right region, closer to the crash boundary, velocity increasing as they deflect the joystick to the right as well (a destabilizing joystick deflection). The truth label here is *center* because the participant does not move the joystick for 400ms after the end of this sample. However, the model predicts that the participant should deflect to the left, which appears to be objectively more correct than the "ground truth" label. Therefore, the training data itself may actually include noise introduced by subpar participants' suboptimal movements; however, the EDC can learn better, more intuitive representations from the combination of embodied data and language data from better participants. Figure 3.4 (middle and bottom) shows that participants from the Medium and Good proficiency groups, respectively, are also occasionally prone to the same situations faced by the participant in the top sample, and sometimes make mistakes. Here, the Medium and Good participants are both either in or moving closer to the left region, and the classifier predicts that the participant should deflect to the right, despite a ground truth label of *center*. This indicates that the EDC develops a more accurate model of both disoriented balancing task performance and in-the-moment guidance through language by learning from multiple participants. If the model were reevaluated against expert/common-sense judgments of optimal human actions, the metrics in Table 3.1 could rise substantially. Additionally, by accurately predicting subpar actions, the EDC can be used to guard against them.

Figure 3.5 shows samples labeled *center* where the human does not move the joystick but the classifier predicted an optimal movement to the left (top plot) or right (bottom plot). The graphs themselves show the joystick deflection on the Y-axis vs. the sample average position on the X-axis. In Figure 3.5 (top), many samples are clustered just right of center with joystick deflection to the left (bottom part of the plot). The opposite is true for the bottom plot, with deflections clustered right of center while the average position is just left of the DOB.

If I examine these plots by participant proficiency, the Proficient and Somewhat Proficient samples remain mostly in the center region, close to the DOB. These participants make slight

joystick deflections to remain within 20° of the DOB, but the model predicts that the best move is a stronger deflection in one direction. These may be cases where the participant is technically within the center region but perhaps close to a left/right boundary. The Not Proficient participants have a much wider spread of average positions where they make close to no deflection of the joystick. The EDC disagrees with them, demonstrating both the noise in the data when non-proficient participants' actions are taken as ground truth, and the ability of the EDC, despite this, to make objectively "good" decisions in the context of this task. The numerical performance of the model (Table 3.1) goes up as participant proficiency goes up, but in fact this reveals that the model is already able to make objectively good decisions, and as human performance improves and participants get better at balancing and become more likely to remain in the center region or recover from drifts, the human decisions are more likely to match these. This suggests that a combined embodied-linguistic method as demonstrated here may be suitable for guiding humans in such a task in real time. The EDC appears to actually display some understanding of the correlation between position and velocity in the problem space, and discrete directional labels.

3.7 Summary

The ultimate goal of this work is to train an AI model that provides real-time guidance cues to a human participant, thereby improving their performance in an embodied task, such as the MARS balancing task or a similar task. Successful guidance of a human through language requires that the AI "embody" the relationship between linguistic terms and the situation in which the human is situated. Here, I present evidence that an AI model can be trained to ground directional labels into embedding-level representations of angular position and velocity, and that this can be done in a way that is sensitive to a participant's proficiency level, provided that information is available as input. These grounded labels can serve as cues to a human participant, as the AI considers the situation and answers "where am I?" with an answer to "where should I go?" (e.g., "I am drifting to the left. I should deflect more to the right.").

The EDC model trained on data from participants of all proficiencies, displays apparent performance on par with a Somewhat Proficient participant, but a deeper dive into misclassifications reveals that even though the training data itself is noisy, as the ground truth is taken to be the actual actions of the participants, even non-proficient ones, the model's apparent mislabels may actually be better decisions than those of study participants.

Given the nature of the task and the need for an immediate human response, is a linguistic cue the best cue to use in this case? While disoriented, humans may not respond as quickly to language cues; perhaps visual or vibrotactile cues are more apt for prompting faster responses. Further experiments are needed to investigate real-time human-AI collaboration in this task (e.g., which AI cues help humans perform better?). Nonetheless, the language input seems to be important to the model for predicting directional guidance, regardless of how that guidance is ultimately expressed. Another feature that could improve the situated embodied model is speed of the MARS, i.e., adding thought vectors representing things like "too fast" or "in control" to positional thought vectors could bolster the combined model's effectiveness as a countermeasure to disorientation by factoring in gradations for things like speed or amount of deflection, which would be important for actually guiding humans in the MARS task where continuous joystick deflecton is being applied. Furthermore, ablation studies can be conducted to quantify the effect of each embedding type, particularly the precise role of language. By taking the existing sentence annotations and automatically transforming them into alternate phrasings (e.g., "I think I am somewhere in the center" $\rightarrow$ "I *am* somewhere in the center"), we can quantify the differences in sentence and contextualized word embeddings, and the resultant predictive power of the EDC.

In this chapter, I demonstrated that a machine learning model can predict directional signals similar to the Somewhat Proficient group, as shown in the offline analysis above. At times, the predictions were more accurate than the ground-truth labels extracted from the data. The disagreement between the EDC and the ground truth labels suggests that when the actual human is disoriented (as indicated by the ground truth data) during the actual task, the EDC (or similarly trained agents) can provide objective instructions to prevent the MARS from crashing and ultimately keep the

MARS balanced in the center. But how would the agent perform when paired with a human during the actual task in real-time, i.e., would it succeed at improving human performance; would the human rely and trust on the agent's suggestions; or would the human not trust the predictions at all? These questions are explored in multiple human-subject studies with real-time assistance in subsequent chapters.

Chapter 4

AI Guidance to Combat Spatial Disorientation

This chapter is based on the following published papers:

1. *"Embodying Human-Like Modes of Balance Control Through Human-In-the-Loop Dyadic Learning" [111] published in 2024 at the AAAI Spring Symposium on Human-Like Learning.*

2. *"Combating Spatial Disorientation in a Dynamic Self-Stabilization Task Using AI Assistants" [112] published in 2024 at the International Conference on Human-Agent Interaction. This paper received a Best Paper Award nomination at the conference.*

3. *"Bidirectional Human-AI Learning in Real-Time Disoriented Balancing" [110] published in 2025 at the AAAI Conference on Artificial Intelligence as an interactive demo.*

Maintaining spatial awareness and orientation is a critical human capacity in domains like piloting, spaceflight, and even driving. Spatial disorientation has been and continues to be a leading cause of fatal aircraft incidents [26, 62] and occurs when sensory information (e.g., from the visual, somatosensory, and vestibular systems) is erroneous, which can lead to unrecoverable crashes, immense injury, or loss of life [124, 62].

An AI agent in this situation could potentially use numerical signals to track the pilot and vehicle's positioning in the relevant orientational plane(s), detect if there is a risk of losing control [43, 197, 184], and even alert the pilot to make corrective maneuvers. However, there is evidence, particularly in high-risk domains such as aviation, of either over-trust or under-trust in highly automated systems. For instance, Sadler et al. [149] demonstrated that pilots' trust in the recommendations of an automated system correlated with the level of transparency (such as justification) in the recommendation. The *shared autonomy* literature indicates that even when an agent knows an optimal strategy, failing to comply with a suboptimal strategy its human partner insists on may negatively affect trust and lead to disuse of the system [72, 98, 151, 125].

Continuing from Chapter 3, where an embodied directional classifier (EDC) was able to make predictions similar to humans considered somewhat proficient in the task, and at times correct suggestions, than the actual ground truth data, suggesting an ability to help humans when they become disoriented. In this chapter, I aim to adapt the virtual inverted pendulum environment presented by Vimal et al. [180] to enable additional high-throughput studies and to facilitate human-agent paired experiments. I also extend the intermediate model from the previous chapter (joystick-deflection predictor) to incorporate LSTMs and GRUs to capture time-series patterns. I apply the HITL techniques explained in Chapter 2 to the MARS and VIP tasks while exploring the underlying embodiment of the learning algorithms. I also assess whether and how an AI model's ability to embody a task space, as a human would, translates into greater assistance to a human in that task or a stronger human preference for that assistant over others, and whether this preference aligns with trust measures.

4.1 Model Training

Our goals in model training were to train AI models capable of independently performing an IP balancing task parameterized by MARS physics and to create digital twins of humans that replicate different kinds of participant performance on the task. In some cases, these two categories overlapped, leading to a testable hypothesis: that a model that performs well on the task may also assist a "pilot" (real or simulated) in performing the task more effectively.

Figure 4.1 shows an I/O schematic of all models. At time t_x, models take in a window containing the past $winSize$ seconds of angular positions and velocities and predict the joystick deflection made at time $t_{x+future}$. Supervised learning (SL) models trained on human data additionally take into account the past $winSize$ joystick deflections made by the subject. If $winSize = 0.0$, the input consists of the values at t_x only. If $future = 0.0$, the next joystick deflection is predicted.

4.1.1 Reinforcement Learning Models

My collaborators and I trained reinforcement learning-based models that learn directly from exposure to environmental physics using a custom variation of Gymnasium's classic-control Pen-

dulum environment [168]. This included 1) a problem space bounded at $\pm 60^{\circ}$ from the DOB, like the MARS/VIP task; 2) a random starting point for the inverted pendulum within the newly defined problem space; 3) a custom reward function[4] given by Eq. 4.1, to encourage small continuous adjustments like those of Good MARS participants [175, 180].

$$r = \begin{cases} 0, & \text{if } -30^{\circ} \leq \theta \leq 30^{\circ} \\ -(\theta^2 + .1\omega^2 + .01d^2), & \text{if } \theta < 30^{\circ} \cup \theta > 30^{\circ} \end{cases} \tag{4.1}$$

The reinforcement learning (RL) algorithms were directly exposed to environmental physics, whereas the SL models received only implicit physics through human performance data. The default SAC and DDPG implementations routinely converged to an optimal strategy that manifests as an immediate rotation to the DOB and holding position there. We also trained and evaluated behavior cloning (BC) [148] and adversarial inverse RL (AIRL) [57] using Good MARS participant data, to teach the models strategies closer to what humans would execute, in terms of replicating behavior or uncovering implicit reward functions in the data.

RL models take the current angular position and velocity to predict the next joystick deflection ($winSize = 0.0, future = 0.0$, cf. Figure 4.1). We used Stable-Baselines3's SAC and DDPG implementations [143], and trained them with the default MLP policy, BC, or AIRL. Gaussian-distributed noise was added to the action space to encourage exploration, as the IP is considered an under-actuated task [78]. In total, we trained 5 RL models: 1) SAC & DDPG each with the standard policy; 2) SAC & DDPG each trained using BC; 3) AIRL implemented with a SAC-based generator model.

4.1.2 Supervised Learning Models

To replicate human-like real-time performance of the MARS task, I trained Multilayer Perceptron (MLP), Vanilla Recurrent (RNN), Long-Short Term Memory (LSTM) [77], and Gated Recurrent Unit (GRU) [35] network architectures over actions made by humans in the actual MARS

[4]Where θ is the angular position, ω is the angular velocity, and d is the joystick deflection.

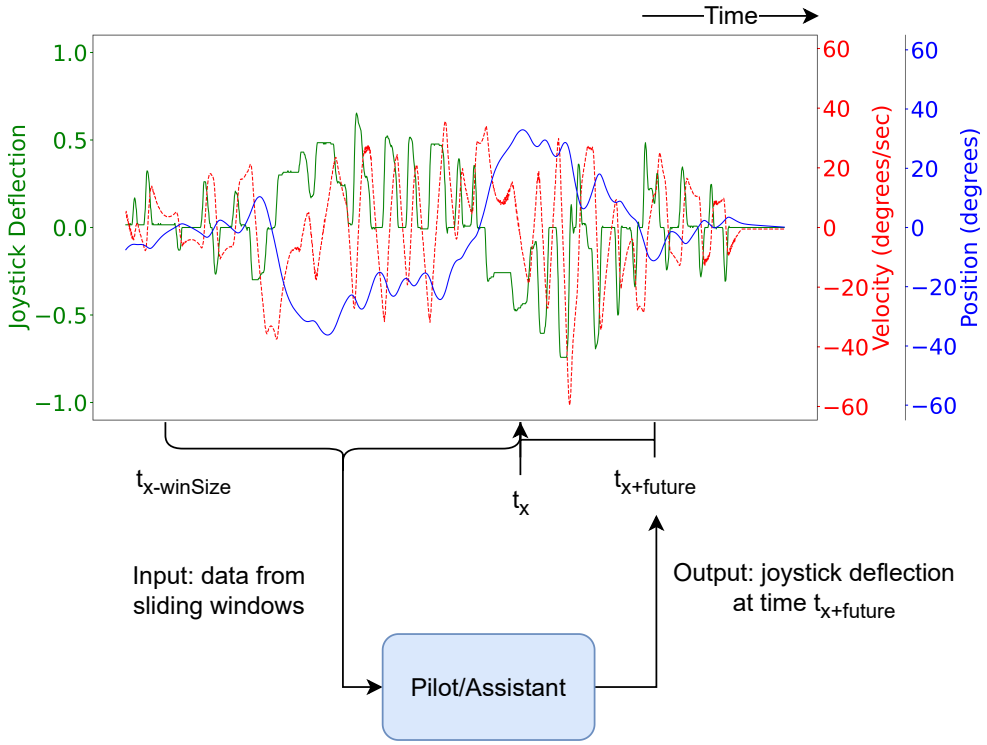


Figure 4.1: Model input and output structure. "Pilot/Assistant" stands in for any one of the trained prediction models.

data. Architectures were trained using different window sizes ($0.0s$, just the current timestep—MLP models only; and $0.2s$, $0.3s$, and $0.5s$). Training data was also split into Good, Medium, and Bad proficiencies, and individual models were trained on data of a specific proficiency. An additional set of models was trained using a combination of 1) Good & Medium and 2) Good, Medium & Bad proficiency data, to see if models could learn strategies employed by certain proficiency groups in scenarios that were not experienced by the others. In total, I trained 40 individual SL models, all of which, even when they successfully avoid crashes in the solo task, demonstrate suboptimal strategies with human-like oscillation and intermittent deflections. These behavioral differences show the differences in how the RL and SL models learn to situate themselves in (or "embody") the problem space.

Pilot	**Crashes**↓	**% destab.**↓	$\mu\lvert\theta\rvert$ (°)↓	$\sigma(\theta)$ (°)↓	$\mu\lvert Mag\rvert_{vel}$ (°/s)↓	vel **RMS**↓
Good	7 / 17	15.9 / 54.0	16.6 / 20.3	21.5 / 23.7	53.3 / 53.0	70.6 / 77.7
Med.	9 / 40	21.4 / 63.3	20.1 / 28.8	18.9 / 29.7	68.3 / 122.7	93.9 / 152.3
Bad	27 / 23	36.7 / 52.8	21.4 / 19.4	25.9 / 14.3	114.1 / 57.4	135.9 / 92.2

Table 4.1: Performance statistics of pilot exemplar models (values are averaged over 3×30 sec. trials except # crashes, which is summed). Slashes separate models trained over MARS and VIP data. Columns from L–R: # crashes, % destabilizing actions, mean and SD distance from DOB, mean and SD angular velocity magnitude, and RMS velocity. Lower values are better (Sec. 2.4.1).

Selecting Representative Pilots

As the same performance characteristics are exemplified in both MARS & VIP, we identified models that most closely approximate the performance categories from Vimal et al. [180], as shown in Table 4.1.

All SL models were trained to perform the VIP task ($3 \times 30s$ trials), with angular position, velocity, and joystick deflection recorded at each time step. We extracted performance features shown in Table 4.1. Since the data distribution could not be assumed to be spherical, these features were used in k-means clustering ($k = 3$) to approximate the split into Good, Medium, or Bad groups. Following Vimal et al. [175], the cluster of models that displayed higher oscillations and greater average magnitude of deflections was considered Bad, while the cluster that displayed smaller, more intermittent actions was considered Good (with the remainder considered Medium). We then took the models in each cluster that were trained over the equivalent data subset (e.g., models in the Good cluster trained over Good data, m.m.), and used the VIP performance characterization technique from Sec. 2.4.1 to identify which model best exemplified the characteristics of each proficiency group: **Good**—LSTM trained over Good data with a window size of 0.2s; **Medium**—GRU trained over Medium data with a window size of 0.3s predicting 0.1s into the future; **Bad**—MLP trained over Bad data with a window size of 0.5s. That each exemplar used a different architecture also suggests that the human subjects exhibited different strategies in performing the MARS task, with different effects.

Models trained on MARS data were also found to exhibit characteristics of different proficiencies compared to participants in the VIP task, suggesting a degree of generalizability between

the two environments. The selected architectures were then retrained using data from 3 VIP participants of each proficiency group to produce digital twins of VIP pilots. Table 4.1 shows the performance characteristics of each pilot exemplar model. All other models that were trained over Good MARS data were reserved to act as candidate assistants, for a total of 21[5].

4.2 System Functionality

An AI "assistant" can be instantiated as any of a variety of reinforcement learning (RL) or supervised learning (SL) models. Examples include SAC or DDPG instances trained in an environment programmed with the IP physics, or MLP, RNN, LSTM, or GRU models trained on data from humans performing disoriented IP balancing tasks using the VIP or an analogous physical apparatus [133, 174, 112, 111]. There are 26 available assistants, detailed in Mannan et al. [112]. All AI models predict the next joystick deflection given the current angular position and velocity, and thus, depending on the model type and training data, the AI model may exhibit different levels of native proficiency in performing the task.

In our system, after a tutorial to help the user acclimate to the joystick, controls, pendulum, and RDK movement, bidirectional human-AI learning proceeds in two phases:

1. Human training: First, the user performs the task alone to determine baseline performance. Then, the human is assisted by an AI that provides suggestions when deemed necessary, rendered as arrows on the screen.

2. AI training:

    a. An AI performs the task alone. During a solo performance, the AI receives only numerical input, but depending on the model and training data, it may exhibit varying levels of proficiency on the task.

[5]The selected Good pilot architecture was retrained with different weight initialization to create a distinct instance of the model to act as an assistant.

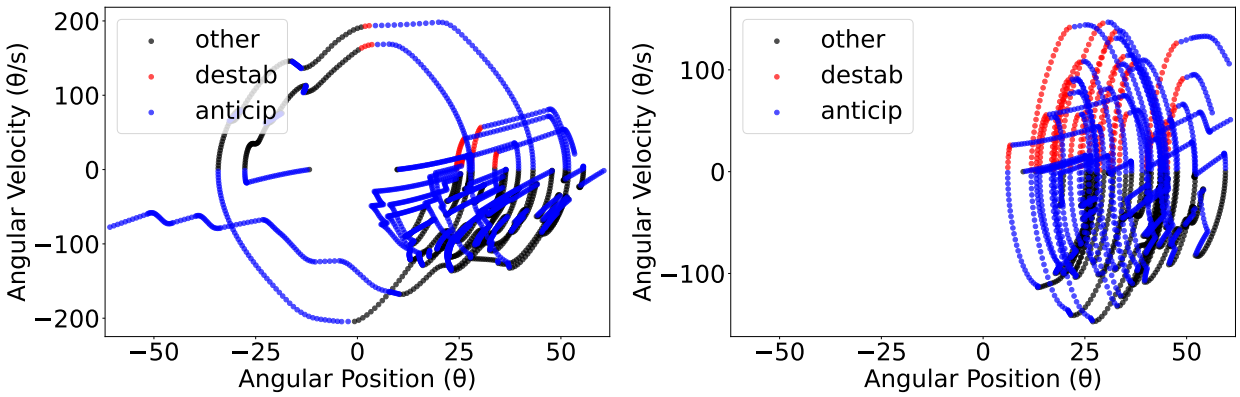


Figure 4.2: Phase portraits of sample human VIP performance without [L] and with [R] AI assistance. With AI assistance, this human subject decreased their oscillation and maintained stability even while offset from the DOB.

b. The human then assists the AI in a second run by deflecting the joystick to keep the VIP balanced. This is done under the 50% coherent VIP condition to ensure that the signals the AI receives are those of a human experiencing disorientation. All episodes in which the human and AI disagreed on the direction of movement are recorded. After the run, a brief finetuning is performed to update the assistant.

c. The updated AI performs the task again, where the human can determine whether the AI has improved or requires further corrections and updates. If further corrections are required, step (b) can be repeated until the AI achieves acceptable performance.

After each phase, the baseline and assisted performances of the human or AI are shown as phase portraits of angular velocity vs. angular position (e.g., Figure 4.2).

4.2.1 Technical Specifics

RL assistant models learned directly from exposure to environmental physics using a custom variation of Gymnasium's classic-control Pendulum environment, modified to reflect the dynamics of the VIP task. The supervised learning models have been pretrained on human-subject data gathered from trials performed on a physical Multi-Axis Rotation System (MARS) apparatus, programmed with identical dynamics, in which subjects use a joystick to balance the device while

deprived of orientational cues (details in Wang et al. [184]). Additional data was gathered from subject trials in the VIP setting (details in Mannan et al. [112]).

I also incorporate a *crash predictor*, a stacked GRU model as reported in Wang et al. [184], that predicts the likelihood of a crash. AI cueing is provided in cases of imminent danger (crash is $\geq$80% likely) where angular distance from the DOB exceeds 12°.

For fine-tuning models during the AI training phase, the actor networks of the SAC and DDPG are fine-tuned using behavior cloning on the trial data, the SAC-AIRL model is updated using AIRL on the trial data, and the supervised learning models are fine-tuned using standard methods. The RL models are fine-tuned for 100 epochs with a learning rate of $1e-5$ and a batch size of 64. The supervised learning models are fine-tuned for 20 epochs with a learning rate of $1e-7$, a 9:1 train-test ratio, and a batch size of 16. All model fine-tuning can be performed on a consumer laptop and takes approximately 30 seconds per training run.

Figure 4.3 illustrates the pipeline with VIP, crash predictor, assistant, and pilot. Upon crashes, PyVIP resets the VIP to a random angle given by:

$$(r+1) * sgn(r-0.5) * ipoff * 0.5$$

where r is a random number sampled from a uniform distribution over $[0, 1)$ and $ipoff$ is the maximum allowable offset of the pendulum reset position with respect to the DOB, specified as a fraction of the fall limit. I use a value of 0.25 for $ipoff$, corresponding to a maximum allowable offset of $\pm$15°. I use a random seed of 42. Raw data from PyVIP evaluation trials was saved in degrees.

The 4 major components of the evaluation pipeline include: 1) VIP 2) Crash predictor; 3) Assistant; 4) Pilot. The **VIP** component is *PyVIP*, a Python implementation that facilitates the easy integration of ML models. The **Crash predictor** is a trained instance of the best crash prediction architecture reported in Wang et al.: a stacked GRU trained over inputs like those in Section 4.1 that predicts the likelihood of a crash occurring. Due to the crash predictor's high false positive rate, I added a *crash probability threshold* of 0.8 where only highly imminent danger would permit

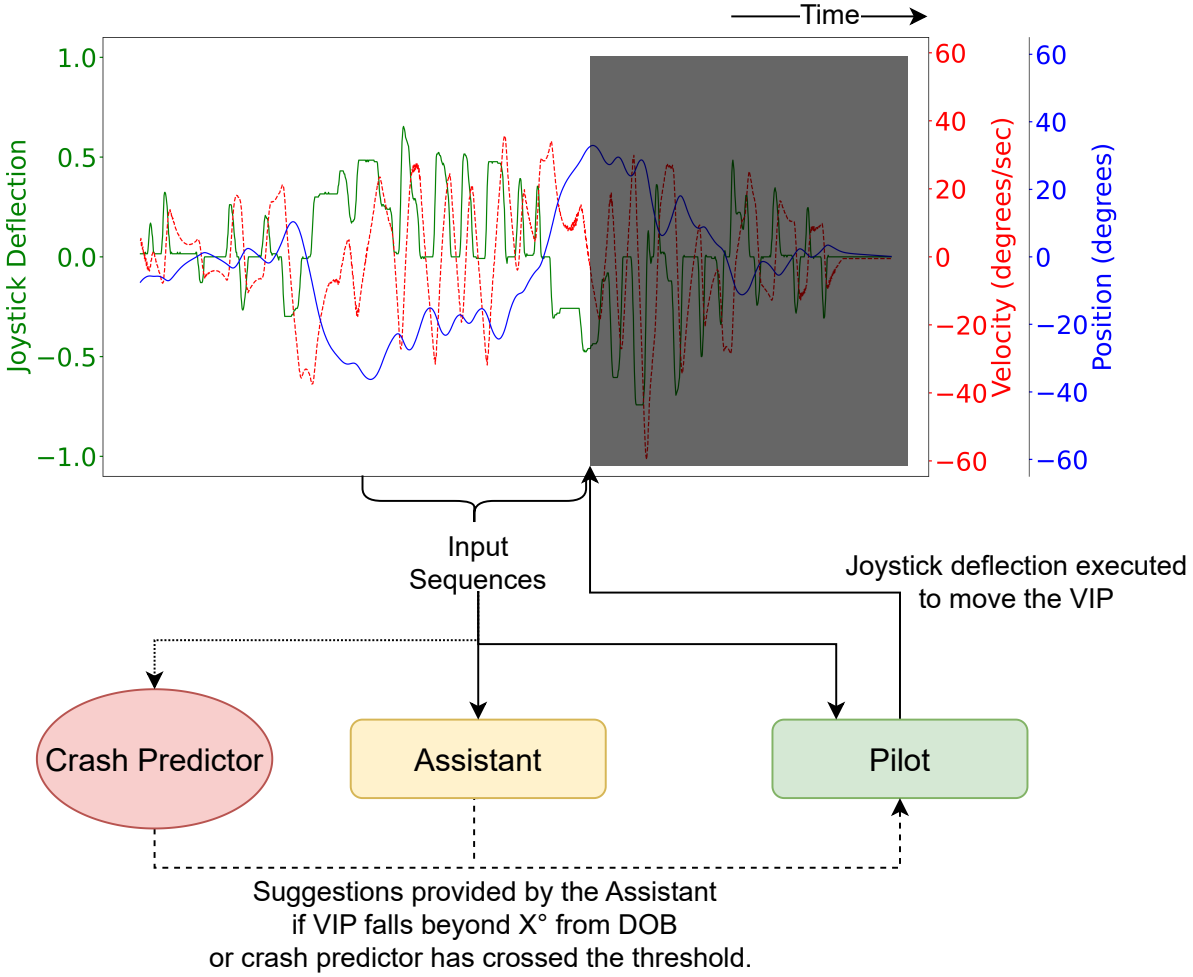


Figure 4.3: PyVIP evaluation pipeline.

assistant suggestions[6]. The assistant would provide suggestions when either 1) crash probability is greater than the threshold *and* angular distance from the DOB exceeds 12°, or 2) angular distance from the DOB exceeds 15°. The **Assistant** observes task performance and makes suggestions when certain conditions are met. The **Pilot** may be a digital twin or an actual human that controls the VIP. I used 6 digital twins, each trained on both MARS and VIP data, corresponding to the architectures mentioned in Section 4.1.2. Humans control the VIP with a joystick.

[6] Wang et al. [184]'s hypothesis proposed that too many false positives could cause a human pilot to lose trust in the assistant, but also need to not admit too many false negatives. Following this advice, 80% represents a balance in these constraints.

4.3 Evaluation

I performed two evaluations: 1) A high-throughput evaluation of pilot digital twins in co-performance of the VIP task with candidate assistants; 2) A human subject study of human co-performance of the VIP task with the best-performing assistants from the digital twin study. The source code for the VIP task (in python), both experiments, and scripts for training AI assistants can be found on GitHub[7].

4.3.1 Digital Twins Study

In these experiments, the pilot has an 80% probability of accepting and executing an assistant suggestion instead of its own next action. If accepted, the pilot makes suggested deflections with a noise of $\mathcal{U}(-.05, .05)$ added to simulate human imprecision, after a $.4 + \mathcal{U}(-.05, .05)s$ delay to simulate reaction time[8].

I ran each evaluation for 3 30-Section trials. Data was sampled at 200 Hz with each sample comprising of the angular position and velocity of the VIP, joystick deflection, crash probability, pilot and assistant's joystick deflections, which entity's deflection was performed, and whether the deflection made was destabilizing. 468 individual digital twin trials were collected, or 3.9 hours of data.

Results

Table 4.2 shows performance differences between the digital twins when unaided and when aided by different assistants. Following the performance evaluation from Section 2.4.1 (where lower metric values signal improvement), SAC-AIRL is the overall strongest assistant for digital twins, decreasing crashes, % destabilizing deflections, and RMS velocity to a statistically significant level (all $p < 0.0001$ according to a paired two-tailed t-test).

[7]https://github.com/csu-signal/HITL-VIP

[8]There is no prior work establishing the probability of human subjects following AI advice in *this* task, but work on other tasks report ∼80% correctness of/willingness to rely on AI advice [49, 182, 104]. A $0.4s$ reaction time is fast for an average human [123], but well slower than a trained pilot [23].

Assistant	Crashes↓	% destab.↓	$\mu\|\theta\|$ (°)↓	$\sigma(\theta)$ (°)↓	$\mu\|Mag\|_{vel}$ (°/s)↓	vel RMS↓
SAC	0 / -5 / -25 -8 / -25 / -12	2.2 / 4.9 / -18.3 -31.8 / -38.3 / -28.5	0.4 / 1.2 / -3.6 -2.0 / -7.6 -1.0	-0.9 / -6.6 / -7.9 -1.9 / -6.7 / 7.9	18.0 / -25.3 / -56.9 27.0 / -37.7 / 7.5	18.7 / -38.7 / -67.5 24.4 / -42.2 / -6.9
SAC-AIRL	-2 / -7 / -17 -15 / -33 / -12	1.9 / -3.8 / -14.4 -36.8 / -41.5 / -28.5	1.9 / -0.1 / 5.4 -0.5 / -6.1 / 0.9	0.9 / -7.7 / -0.3 -6.8 / -10.9 / 9.1	-9.8 / -38.7 / -62.8 -22.5 / -66.6 / -8.7	-11.2 / -53.5 / -71.2 -29.9 / -72.9 / -20.3
DDPG	1 / -2 / -21 -11 / -23 / -12	2.3 / 3.7 / -17.3 -36.4 / -36.4 / -25.1	2.4 / -2.4 / -2.2 2.6 / -7.5 / -1.0	2.1 / -3.1 / -3.4 3.5 / -3.6 / 9.2	25.3 / -11.0 / -43.1 47.4 / -34.1 / 5.5	24.6 / -21.2 / -51.0 43.5 / -38.6 / -12.5
MLP-GMB-0	-2 / 4 / -11 -1 / -18 / -4	-0.5 / 7.0 / -8.8 -21.6 / -29.7 / -21.5	2.4 / 3.1 / 2.2 2.1 / -6.8 / 4.0	2.2 / 2.8 / -1.7 0.8 / -2.5 / 12.6	18.8 / 7.0 / -43.2 31.4 / -15.5 / 12.2	24.5 / 1.3 / -48.0 38.7 / -18.6 / 3.8
LSTM-G-0.2	3 / 14 / -7 4 / -13 / 1	8.0 / 23.5 / -1.6 -11.0 / -20.5 / -8.8	3.6 / -0.2 / 0.9 1.7 / -7.3 / 3.2	2.8 / 0.6 / 1.5 2.7 / -4.0 / 12.6	25.4 / 15.6 / -33.7 32.9 / -13.2 / 27.4	34.6 / 15.2 / -33.2 41.5 / -14.2 / 24.8

Table 4.2: Differences in performance with and without assistance (e.g., *0* means no change in that metric, lower values are better—Sec. 2.4.1) In each cell, top line refers to MARS pilot models and bottom to VIP pilot models. Slashes separate Good/Medium/Bad pilot models. Under Assistant, G/M/B denotes the proficiency of the assistant training data, decimals denote window size. Assistants shown achieved a significant reduction in at least one metric value. See appendix for results for all 26 assistants.

RL models are generally better assistants than SL models over human data. Interestingly, *MLP-GMB-0* (MLP trained over all proficiencies with no window) decreased crashes as much as *SAC-AIRL*, but for the Good MARS digital twin only. The Medium exemplar architecture performs significantly worse when trained over VIP data than over MARS data, and the Bad VIP pilot performs much more like the Medium MARS pilot, while Good pilot models are roughly consistent with each other across tasks, suggesting that many strategies lead to poor task performance and relatively few do well. This also speaks to *SAC-AIRL*'s ability to reduce crashes in both tasks for all proficiency levels. Other high-performing models also reduce crashes (*DDPG*: $p = 0.0002$, *MLP-GMB-0*: $p = 0.0033$) and destabilizing deflections (*DDPG*: $p = 0.0006$, *MLP-GMB-0*: $p = 0.0080$) for digital twins trained over both MARS and VIP data, demonstrating transfer between digital twins trained over the different task data.

4.3.2 Human Subject Study

Results from the high-throughput digital twins setting indicated that the 5 assistant models shown in Table 4.2 had statistically significant effects on one or more metrics when co-performing with digital twins trained on both MARS and VIP data. These models — 3 RL-based models and 2 models trained on human data — were included as candidate assistants in the human subject study.

This provided a robust yet tractable sample of assistants to assess in the co-performance of the VIP task with real human subjects.

20 healthy adult subjects (6 female, 13 male, 1 non-binary) were recruited for this study. Each subject participated in 2 experimental sessions separated by approximately one week. In **Session 1**, subjects 1) attempted to balance a 50% coherent PyVIP RDK ($3 \times 30s$ trials); 2) controlled the RDK with assistance from an AI model, rendered as left/right arrows indicating the direction of suggested deflection ($3 \times 30s$); 3) watched the same AI control the RDK while providing directional suggestions via the joystick ($3 \times 30s$). Participants were randomly assigned one of the candidate assistant models during Session 1—subjects were grouped into fours and each group received assistance from a single type of architecture. Subjects were not informed of the type of model they were receiving assistance with.

Between sessions, each assistant model was fine-tuned using data from Task 3 in Session 1. Episodes (consisting of input window and predicted action) where the direction of agent-predicted deflection conflicted with the direction of human deflection were stored. These human-in-the-loop (HITL) disagreement samples were used to fine-tune the model: the actor networks of the SAC and DDPG were fine-tuned via behavior cloning on the new data, the SAC-AIRL model was updated via AIRL on the new data, and the supervised learning models were fine-tuned as usual.

In **Session 2**, subjects 1) undertook Task 1 as in Session 1 (solo RDK balancing—$3 \times 30s$); 2) undertook AI-assisted balancing as in Session 1 Task 2 but with a different assistant model ($3 \times 30s$); 3) undertook AI-assisted balancing with the version of their Session 2 Task 2 assistant fine-tuned with data from Session 1 subjects who interacted with that model type ($3 \times 30s$). In Session 2 Task 2, subjects were assigned a non-fine-tuned model of a different architecture and in Session 2 Task 3 they were given an instance of that same model fine-tuned with HITL data from Session 1. Participants were not informed that the Session 2 Task 3 model was fine-tuned on human data from Session 1.

Finally, subjects took a survey, based on Muir [122], about their solo performance, how AI assistance changed their performance, and the level of trust they had in the assistant for each task.

Results

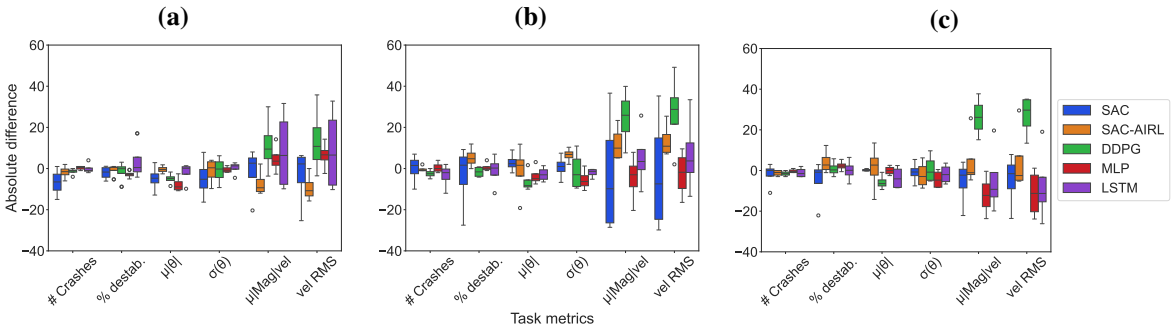


Figure 4.4: Absolute differences between baseline human performance metrics compared to AI-assistance in (a) Session 1 Task 2, (b) Session 2 Task 2 (different assistant model), and (c) Session 2 Task 3 (fine-tuned Session 2 Task 2 assistant).

I first assessed whether subjects displayed any adaptation to the balancing task within or across sessions that might confound apparent performance improvements due to AI assistance. Following Vimal et al. [174], in which participants in the disorienting MARS task showed minimal learning across consecutive days, I take performance in Session 1 Task 1 vs. Session 2 Task 1 (which were separated by approximately 1 week) as a baseline "no learning" condition in which participants lost familiarity with the task. I then compared performance differences between Session 1 Task 1 vs. Session 2 Task 1 and between Session 2 Task 1 vs. Session 2 Task 2. If the performance differences between Session 2 Task 1 and Session 2 Task 2 are similarly non-significant compared to the performance differences between Session 1 Task 1 and Session 2 Task 1, this indicates that no significant adaptation to the task occurred between tasks within Session 2, and likewise is unlikely to have occurred between Session 2 Task 2 and Session 2 Task 3; therefore apparent differences in Session 2 Task 2 and Session 2 Task 3 performance are likely to attributable to the nature of the AI assistance received.

I computed a score for each participant in each task of interest, given by Equation. 5.1,

$$s = \left(\frac{60 - \mu|\theta|}{60}\right) + \left(1 - \frac{C}{90}\right) + \left(1 - \frac{pD}{100}\right) + \frac{pA}{100} + \left(\frac{R}{\max_R} - \frac{C}{\max_C}\right), \tag{4.2}$$

where C is the count of crashes over the task ($3 \times 30s$ trials), pD is the percentage of deflections that were destabilizing, pA is the percentage of deflections that were anticipatory (see Figure 2.10), R is the task-level count of recoveries from beyond 20° away from the DOB to within 20° of the DOB, and $\max_R$ and $\max_C$ are the maximum number of recoveries and crashes in the data, respectively.

Because these scores are not normally distributed, a Wilcoxon Signed-Rank test was performed between Session 1 Task 1 and Session 2 Task 1 scores, and between Session 2 Task 1 and Session 2 Task 2 scores. No statistically significant differences were found between either pairing, with similar p-values (.2627 between Session 1 Task 1 and Session 2 Task 1, and .3681 between Session 2 Task 1 and Session 2 Task 2). While these non-significant results suggest that task performance remained stable, it should be acknowledged that with a larger sample size, subtle learning effects might have been detected. However, given that these p-values are relatively high, any such adaptation appears minimal and is unlikely to account for the much larger performance shifts observed when AI assistance was introduced.

Figure 4.4 shows the absolute difference in performance metrics between human solo VIP performance and 3 versions of AI-assisted performance for each model type: using the original model weights in Session 1 (4.4a), the non-fine-tuned assistant from Session 2 (4.4b), and the Session 2 assistant fine-tuned on data from humans in Session 1 who interacted with the assistant of the same architecture (4.4c). We observe that, as in the digital twin studies, the RL assistants in Session 1 were better at reducing the absolute number of crashes than the SL assistants, but this distinction often disappeared or reversed after the assistant models were fine-tuned on participant data and then re-evaluated in Session 2. In Session 2 Task 3, the SL models fine-tuned on Session 1 data were now, on average, better at reducing metrics associated with velocity and oscillation, such as RMS velocity, velocity magnitude, and the standard deviation of position. This effect is absent in Session 2 Task 2, where subjects were assisted by a different type of model.

	SAC	SAC-AIRL	DDPG	MLP	LSTM
μ	69.17	58.83	132.17	28.33	28.41
σ	50.47	74.55	164.25	22.27	12.87

Table 4.3: Mean & SD of number of disagreement episodes logged during HITL study, by assistant model type.

Perceived Performance Impact

Assistant	Task 2			Task 3		
	$+$	$\sim$	$-$	$+$	$\sim$	$-$
Overall	50	15	35	55	30	15
SAC	25	25	50	50	25	25
SAC-AIRL	50	0	50	50	25	25
DDPG	75	0	25	50	25	25
MLP-GMB-0	75	25	0	75	25	0
LSTM-G-0.2	25	25	50	50	50	0

(a) $+$: improved (incl. slightly/significantly), $\sim$: no change, $-$: decreased (incl. slightly/significantly).

Reported Trust

Assistant	Task 2			Task 3		
	$+$	$\sim$	$-$	$+$	$\sim$	$-$
Overall	25	35	40	30	45	25
SAC	25	0	75	25	50	25
SAC-AIRL	0	50	50	25	50	25
DDPG	0	75	25	25	25	50
MLP-GMB-0	50	25	25	25	50	25
LSTM-G-0.2	50	25	25	50	50	0

(b) $+$: high to complete, $\sim$: moderate, $-$: low to none.

Table 4.4: Perceived performance impact of (4.4a), and reported level of trust in (4.4b) Session 2 Task 2 and Task 3 assistants (as %).

4.4 Discussion

Like in the digital twins study, the SAC-AIRL assistant often helped the human subjects reduce crashes and oscillations. This is more pronounced in versions fine-tuned on HITL data, suggesting that models with a more human-like strategy contribute to this effect.

Assistance from the DDPG shows a strong tendency to increase RMS velocity and velocity magnitude values, and this is actually *more so* after HITL fine-tuning. This discrepancy is reflected in the number of disagreement episodes logged for each model type (Table 4.3). Human subjects registered a much higher mean number of disagreements with the DDPG model—and RL models more generally—than with the SL models. This further indicates that the DDPG and RL models behave in ways that may contradict human intuition and/or physical instinct. Through their data and training, SL models are embodying the problem space and performing the task in a more human-like way, including transitioning from destabilizing to corrective and anticipatory deflections at distances from the DOB that align with human behaviors (cf. Figure 2.11).

Due to different human reaction times, it is not possible to know exactly when human subjects followed assistant suggestions, but we can calculate a heuristic estimate based on instances where a subject deflects in the direction suggested by the AI within a threshold of the AI making a suggestion. Using a threshold of $450ms$, subjects followed AI suggestions approximately 44% of the time ($\sigma = 14\%$). This number is significantly lower than the $\sim$80% that can be seen in other domains (see Section 4.3.1), suggesting particularities of the task need to be accounted for. Broken down by assistant type, the DDPG was the most followed assistant at 53%, followed closely by SAC-AIRL (51%). The MLP was the least followed (32%). Interestingly, fine-tuned assistants were actually followed 7% *less* than non-fine-tuned assistants (4% less when comparing Session 2 Task 3 to only Session 2 Task 2), even though subjects rated these assistants as more trusted and preferable (see below).

Trust survey Participants were asked to assess how the AI's suggestions changed their performance, and to rate their trust in the AI. I report survey results after Session 2, where participants also expressed their preference for one of the two assistant models used: an assistant with no specific fine-tuning (*Task 2* assistant), or one fine-tuned using Session 1 human data (*Task 3* assistant).

Table 4.4a shows subjects' perception of their assistant's impact on their performance, overall and broken down by assistant type. Table 4.4b similarly shows the reported level of trust in each assistant. I observed an overall trend toward better perceived impact on performance and more reported trust in the fine-tuned model when compared to the original, although interestingly some models to which participants ascribed a positive effect on performance (e.g., the MLP), were rated as less trusted after fine-tuning. At the assistant type level, these numbers should be taken in the context of small sample sizes ($N = 4$). When asked to pick a preference between the Task 2 and Task 3 assistants, 15% chose Task 2 and 70% chose Task 3 (15% no preference).

Behavior Changes in AI Figure 4.5 shows velocity-position scatter plots for different AI agents, particularly supervised learning agents, where red dots represent destabilizing deflections while blue dots represent "anticipatory" deflections before, during, and after HITL trials with representa-

tive participants. The data used for retraining these models come from a different set of participants (see Mannan et al. [111] for details) but the principle is seen in the current experiment as well. We do see that the Bad and Good pilots undergo nuanced changes in their oscillatory patterns. Before HITL, the bad pilot displays destabilizing actions when the VIP's position is near the crash boundary but after HITL training the model learns to make better actions in those same risky positions. For the good pilot, it learns to oscillate around a focal pivot similar to behavior humans show in the VIP and MARS task (see Figure 4.2). The alignment in behavior and vis-a-vis recommended action is considered to be a major factor in increasing the trust humans place in AI assistance. In the MLP assistant, we see that it is already a good performer of the task but after HITL it becomes even better at maintaining balance around a focal point but with 10x less velocity than before (50 deg/sec before vs 5 deg/sec after).

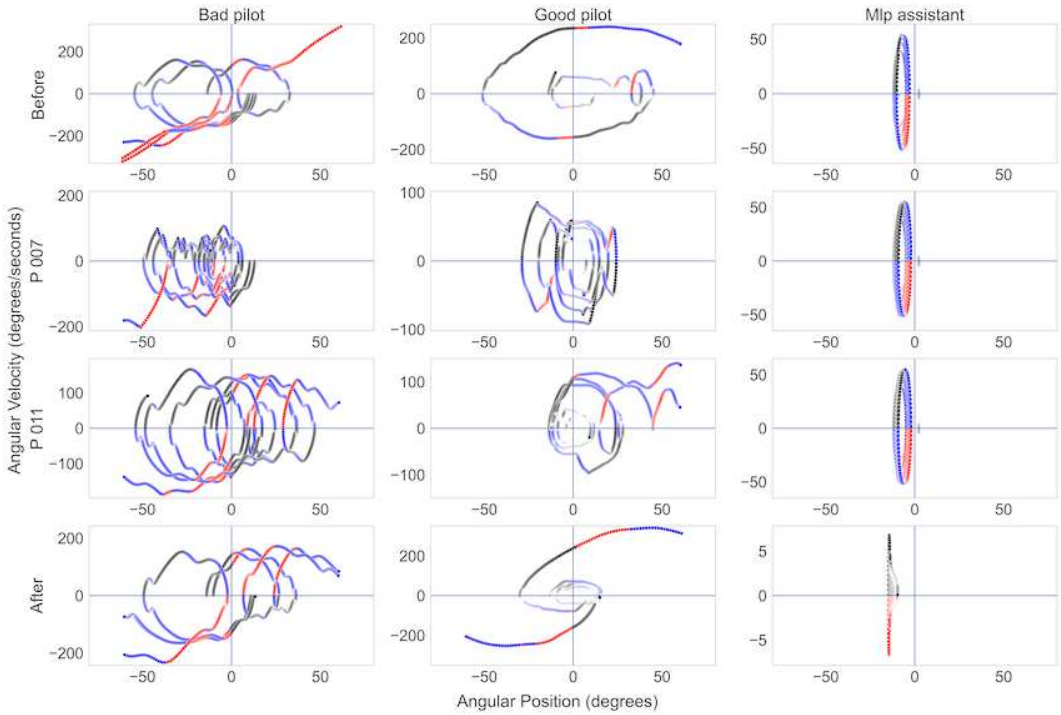


Figure 4.5: Velocity-position scatter plots. Red dots represent destabilizing deflections while blue dots represent "anticipatory" deflections. Scatter plots for each model before, during, and after HITL trials with representative participants.

4.5 Summary

In this chapter, I introduce a novel AI-assisted task that helps humans maintain balance under disorienting conditions. I first explored the space of possible AI assistance models using a high-throughput digital twins setting. The top-performing models from this experiment were then used in a human-subject study to assess both the impact on performance and participants' attitudes toward different assistants.

Given specific data and training methods, AIs capable of performing IP balancing alone also assisted real humans in reducing crashes and oscillations. SAC-AIRL, which learns rewards implicit in human performance data, appeared to be an effective disorientation countermeasure in both digital-twin and human-subject studies by embodying the problem space in a way that incorporates both physics and human signals. Although RL models, on average, perform better as assistants than SL models trained over human data, they do so by suggesting actions that often diverge significantly from the apparent model of the task captured in human actions. This is reflected in the human subject studies: human subjects empirically perceived the RL models as performing the task incorrectly, and models that learned and embodied a human-like strategy through pretraining over human data, then were fine-tuned over more human data in the subject study, were able to significantly reduce factors related to VIP oscillation and velocity.

Palmer et al. [131] illustrate trust dimensions in the use of autonomous or automated systems, such as *robustness* (handling perturbations/deviations appropriately), *benevolence* (supporting mission and operator), and *dynamism* (negotiating changes in environment). While the assistants' suggested actions may be appropriately corrective (benevolent), respond to pilot-induced perturbations such as ignoring cues (robust), and transfer between the MARS and VIP tasks (dynamic), they also need to be understandable in terms of the pilot's internal model of the situation to avoid corrections directly opposed to what the pilot expects (cf. the DDPG vs. MLP and LSTM in Figure 4.4c). The

findings indicate that humans are more receptive to assistance from an AI that demonstrates a more human-like, albeit objectively suboptimal, balancing strategy.

Future study may investigate fine-tuning on data from a specific participant rather than an aggregate sample, to uncover person-specific patterns in task performance, or an investigation of modeling techniques that can account for the fact that human behavior is likely to change over time, to account for assistance received from an AI agent, including one that is trained in real-time using live human feedback. Subsequent research may also involve transfer to more complex conditions, such as orientation in multiple roll planes or flight simulators, as well as investigating the transfer of AI-assisted high-throughput VIP to the physical MARS.

There also remains the question of *how* to deliver an AI assistant's cues to a human pilot. In this experiment, I rendered visual indicators on the screen, but other modalities may include aurally rendered tones or vibrotactile cues (as in [179]) to indicate the direction and magnitude of the corrective action, or linguistic instructions. In the previous chapter, I presented evidence on the utility of language understanding in task performance, and a multi-variable examination of intervention methods and timing is another avenue for future study.

The VIP task provides a valid and realistic, but deliberately simplified, simulation of balance and disorientation with 1 degree of freedom. The VIP task is demonstrated analogue to the MARS task (see Figure 2.11), which is a high-fidelity simulation for spatial disorientation, but piloting an aircraft requires a more complicated environmental design with at least 4 degrees of freedom (roll, pitch, yaw, speed). In the next chapter, I explore whether the lessons learned here also apply to the use of real-time AI assistance in a more complex and realistic task setting.

Chapter 5

Extending AI Guidance to a Navigational Flight Task

In Chapter 2, I introduced the definition of embodied AI and its significance in enhancing human-agent interactions, and in Chapter 4, I presented my findings on a spatial disorientation task, designed as a spaceflight analog, where an embodied AI agent is capable of not only assisting a human in navigating a simple environment but also displaying increased trust placed in it by the human user. In this chapter, I extend this work to a 3D navigational flight task in a realistic flight simulator. The complexity of the task increases from a 1D action space in a 2D plane (VIP) to a 2D action space in a 3D region (flight simulators). Furthermore, another issue that arises in the 3D space is how to provide guidance during the task; should it be visual or auditory? If it is auditory, then should natural language be leveraged, and if so, what level of verbosity should be used? There have been multiple studies [61, 68, 106] showing that responses to audio stimuli have faster reaction times, but depending on human skill levels, auditory instruction is not always understood promptly, especially in high-risk scenarios. Using auditory guidance would add further variables and parameters to the experiment I propose, and thus, be out of scope for this dissertation. Following the theme of visual guidance established in Chapter 4, I evaluate 2 methods of visual guidance in the navigational task, assuming that the proposed visual guidance should be understandable and unambiguous to humans of varying skill levels.

5.1 Background

Flight simulators have played a crucial role in aviation training since the early 20th century. The first widely recognized flight simulator, the Link Trainer [189], was developed in the 1920s and used extensively during World War II to train military pilots in instrument flying and emergency procedures [147]. Over the decades, flight simulators have evolved from simple mechanical devices to sophisticated computer-based systems capable of replicating a wide range of flight conditions, including weather, and aircraft systems [114, 96, 190]. Figure 5.1 illustrates the evolution

of flight simulators over the past century, highlighting key milestones and technological advancements.

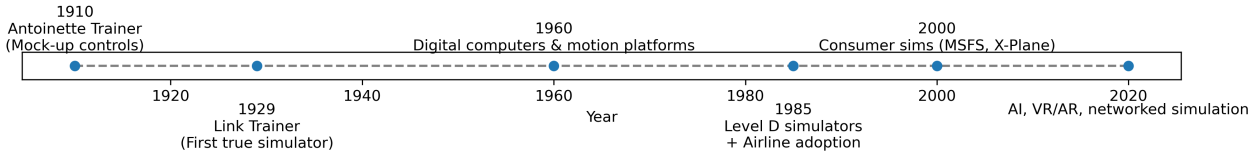


Figure 5.1: Graphical overview of the evolution of flight simulators over the past century, highlighting key milestones and technological advancements.

Modern flight simulators are used for pilot training, skill assessment, and research. They allow pilots to practice normal and emergency procedures in a safe, controlled environment, reducing training costs and risks associated with real flight, and also used for developing and testing new cockpit technologies, human factors research, and evaluating pilot performance [160, 19, 75]. Flight simulators serve several critical functions in aviation, including:

- **Pilot Training:** Simulators provide a risk-free environment for learning basic and advanced flight maneuvers, instrument procedures, and emergency responses.
- **Skill Assessment:** Regulatory agencies require pilots to demonstrate proficiency in simulators for certification and recurrent training.
- **Research and Development:** Simulators are used to study human factors, test new avionics, and develop training curricula.

Additionally, the aviation industry is increasingly recognizing the importance of personalized training experiences that cater to the unique needs of individual pilots [193, 55]. This shift towards more tailored training approaches is driven by technological advancements, including the use of artificial intelligence (AI) and machine learning (ML) algorithms. Furthermore, since COVID-19, the aviation industry has faced a shortage of qualified instructors, and consequently, a few systems have begun to introduce (or at least recognize the potential of) AI agents as part of the training process in flight simulators [193]. Fly-with-AI [51], is one example of a commercial product

that uses AI to provide feedback to student pilots during flight simulation sessions. It provides a virtual flight instructor that can analyze student performance and offer real-time verbal feedback on various aspects of flying, such as maintaining altitude, heading, and airspeed. However, the system primarily focuses on high-level feedback and does not provide fine-grained in-the-moment maneuvering assistance like the AI assistants in Chapter 4.

Additionally, most flight AI assistants are not open source. This lack of transparency can hinder research and development efforts, as other researchers cannot readily build on existing work or adapt the technology to their specific needs. This also reduces the trust users may place in the AI assistant, since its operational processes or the data it uses may not be transparent.

5.2 Research Questions & Hypothesis

Building on the lessons learned in Chapter 4 as the foundation, I extend this work to use flight simulators to study the impact and utility of AI assistance. More specifically, I examine (a) how the AI assistance can change human performance in the task and (b) how the trust placed by humans in the AI differs before and after the AI learns from human demonstrations (HITL). In this navigational flight task, I put the following hypotheses to the test:

1. **H1 (Performance):** AI assistance, irrespective of HITL, will improve task performance, especially in safety and behavioral metrics (see Section 5.3.3 for definitions).

2. **H2 (Behavioral Alignment):** After retraining of the AI agent using human demonstrations, AI performance will decline in performance metrics, but the updated behavior will more closely align with that of humans.

3. **H3 (Trust Calibration):** As a result of the updated aligned behavior, humans will express more trust in the updated AI assistance than in the original AI assistance.

4. **H4 (Assistance Mode):** Guidance using a simple visual aid will be more intuitive to follow with higher trust, i.e., following an arrow vs. aligning with a 3D rendering of a plane;

however, the higher fidelity aid (3D plane) will improve safety and behavior metrics more than the simple aid (arrow).

5. **H5 (Temporal Intervention):** A just-in-time assistance provided "only-when-needed" will result in fewer crashes compared to continuous assistance, where continuous assistance might reduce user attention to the primary task.

5.3 Methodology

For the experiments, I use an open-source framework called PyFlyt [162], a UAV flight simulator environment for training reinforcement learning agents to complete tasks involving flying drones or planes in specific scenarios. The base simulator supports the following scenarios: pole balancing with quad-drones, navigation waypoint collection with fixed-wing planes and quad-drones, rocket landing, and dogfighting with fixed-wing planes. The experiment is designed in a manner to test (a) how merely situated versus more embodied forms of assistance would affect human performance; and (b) how an AI agent, already competent in the task, is affected through retraining with human actions..

The selected scenario for the experiment is the navigational waypoint collection task, in which a pilot flies the plane through an arena, collecting waypoints as fast as possible without crashing (see Figure 5.2). The waypoints are placed at random throughout the arena, and the pilot would have to fly as close as possible to collect them. The waypoints are rendered as spheres that need to be collected in a particular order: the pilot must first find and collect the active green waypoint, after which one of the red waypoints will turn active and change color to green. The task completes when the pilot collects all the waypoints (success) or crashes the plane into the ground (failure).

5.3.1 Environment Design

To facilitate human subject experiments in the realm of HAI collaboration, I extend the base software to include support for human input via a joystick controller, visual aids via an altitude meter and an artificial horizon gyroscope, and the ability to visualize some form of assistance from

an AI. Figure 5.2 shows a screenshot of the software during a test with a human pilot. The fixed-wing plane is flown from a third-person perspective.[9] In the right corner of the screen, the human pilot can see their altitude, and just below that, the artificial horizon that indicates their alignment with the ground plane.

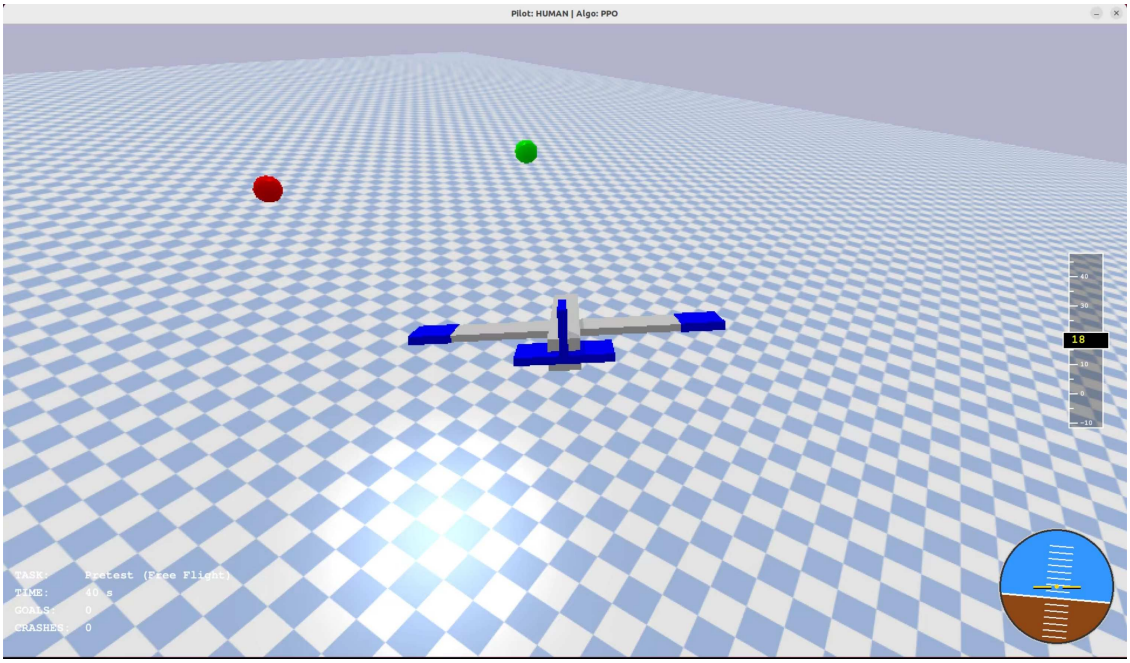


Figure 5.2: Screenshot of the PyFlyt software during a test with a human pilot.

A few constraints were built into the software to reduce mental load and increase the usability of the software:

1. The speed of the plane is fixed at 70% of the maximum. This constraint has 2 functions: (a) to ensure that the human pilot does not have to worry about changing the plane's speed during the task and (b) to ensure the task remains challenging enough.

[9]This is the preferred perspective in video games involving navigation. Additionally, at the cost of verisimilitude, it provides greater spatial awareness to the user and thus affords a better sense of direction for tackling challenges in the environment [99, 150, 67].

2. Removing yaw from the control mechanism since maintaining the roll and pitch orientations of the plane was sufficiently challenging, and removing the yaw mode would further reduce mental load during the task.

3. The PyFlyt software adds an invisible boundary around the arena, and by default, if the plane were to hit the boundary, it would result in a crash. To avoid frustrating human pilots, the boundary walls were reprogrammed such that a collision would automatically invert the plane's direction around and allow the user to continue the task.

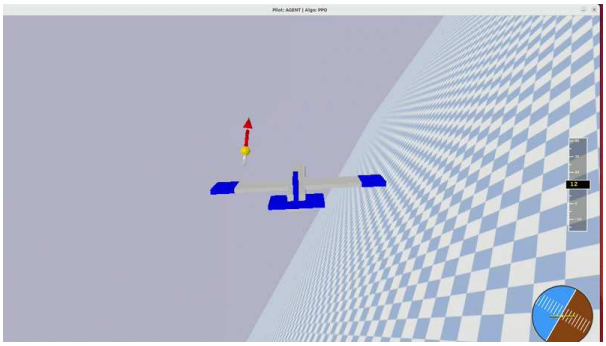
While I considered modifying the environment to allow waypoints to be collected in any order to increase task ease, this modification was ultimately abandoned. The first issue that arises is that the task might become too easy for the human participants, and the effect of AI assistance would be too weak. Second, the more important reason is that it inherently changes the nature of the task for the AI agent. As designed, the AI is aware only of the immediate target(s) in a particular order, and the goal is to optimize the flight path to complete the task quickly. However, with this change, the AI would need to be provided with information about all waypoints for which the order of collection should be determined by the agent itself, thereby reducing this to a version of the infamous NP-hard Traveling Salesman Problem. In the implementation, the agent chose its target based on the shortest-distance rule but kept switching targets when a closer waypoint was encountered, causing it to fly in circles without ever collecting any waypoints.

To verify that the given constraints did not affect an AI's ability to complete the task, an AI agent was successfully trained using the Proximal Policy Optimization (PPO) algorithm and was able to complete the task with no issue (see Section 5.3.4).

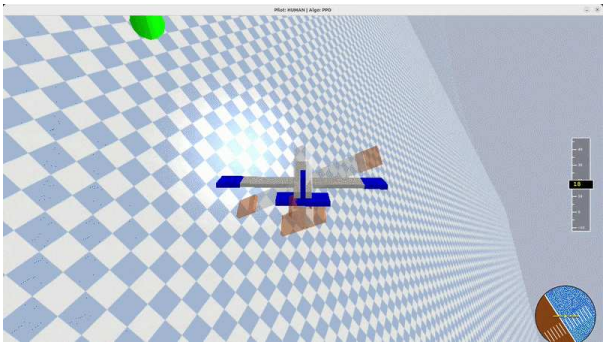
5.3.2 Assistance Modes

Two different assistance modes were designed for this experiment. The first is a form of situated (or a very weak embodiment) assistance rendered as a 3D arrow. Figure 5.3a shows the situated arrow in action, guiding the human participant to the active waypoint they need to collect. The assistance is defined as situated or weakly embodied, as it essentially knows the general direction

and guides the human to "go there" but cannot guide the human "how to get there". The second mode of assistance is a "ghost plane" mode, depicted in Figure 5.3b, which is an embodied form of assistance; it displays the maneuvers of the AI that has learned to complete the task using the training environment itself from PyFlyt. The ghost plane guides the human on "how to get" to the waypoint by instructing when to pitch the nose of the plane up or down, or when to roll left or right.

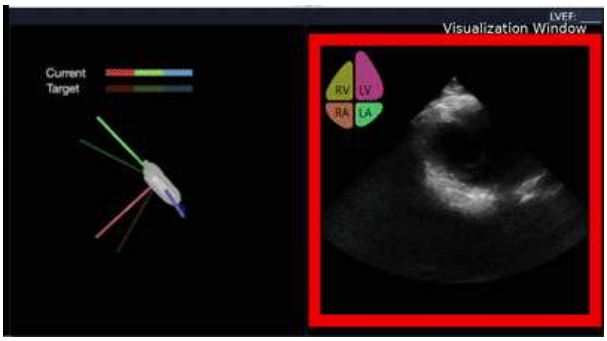


(a) Arrow Assist Visualization



(b) Ghost Plane Assist Visualization

(c) Lower-Limb Ultrasound



(d) Cardiac Ultrasound

Figure 5.3: Comparative presentation of flight simulation assistance visualization modes and the ultrasound feedback guidance modes from the VIGIL project, which inspired the flight simulation guidance.

The arrow and ghost-plane assistance visualizations are inspired by techniques from the *Vectors of Intelligence Guidance in Long-Reach Rural Healthcare* (VIGIL) project within the PARADIGM program (see Chapter 2). The VIGIL system currently guides generalist healthcare practitioners to perform 2 medical diagnosis tasks: a deep-vein thrombosis (DVT) scan of the lower limb (LL) [40] and a cardiac ultrasound to determine left ventricle ejection fraction (LVEF) [41]. In the LL

task (Figure 5.3c), the green arrow points the medical practitioner to "where" the femoral vein is in the ultrasound image, in order to enable them to center it and capture images of relevant portions of the vascular structure to enable diagnosis of whether a blood clot is present or not. The AI here is only aware of the locations of the landmarks and nothing else, and thus provides only a direction as guidance. The arrow assist in the navigational task is inspired by the LL task; here, the arrow is also only aware of the position of the waypoint and can only guide the user to where they should go. In the cardiac task (Figure 5.3d), the goal is to attain a good quality image of the 4 chambers of the heart, especially the left ventricle, by which LVEF can be measured. The "ghost probe" visualizes the orientation that the medical practitioner needs to adjust the probe to obtain the optimal image. This assistance mode can be thought of a fine-grained assistance where it informs the minor adjustments that need to made for optimal task performance. In the navigation task, the ghost plane mode, inspired by the cardiac task, informs the pilot of the specific attitude adjustments required to reach the destination safely and efficiently.

Just-in-time Assistance

Furthermore, a just-in-time mode was tested that only appeared when it was determined that help might be needed. The heuristic used for determining when to intervene and provide guidance to humans follows the same logic as the PyVIP evaluation pipeline (see Section 4.2.1).

$$A_t = C_w \vee W_g$$

Where: C_w is the Crash Warning trigger, defined by the predictive model $P(\text{crash} \mid s, a)$ exceeding the optimized threshold:

$$C_w = \begin{cases} 1 & \text{if } P(\text{crash}) \geq 0.3 \\ 0 & \text{otherwise} \end{cases}$$

W_g is the Waypoint Guidance trigger, defined by the temporal absence of the target T within the pilot's Field of View (FOV):

$$W_g = \begin{cases} 1 & \text{if } t_{last\_seen} > 30s \\ 0 & \text{otherwise} \end{cases}$$

The crash predictor is trained using a stack GRU network that predicts whether a crash is imminent in the next 0.5 seconds. Inputs to the model include linear and angular velocities, a quaternion, global position, and an action; sequences of 0.5 seconds are used for prediction. The 0.3 threshold for the crash predictor was selected after a probability threshold analysis where the crash-predictor achieved better recall than precision; the default decision boundary of 0.5 would increase the likelihood of false negatives, as the training data had an imbalanced amount of data points where crashes did not occur.

5.3.3 Metrics

For analysis of the flight data collected during the task, I designed and employed 2 types of metrics: *performance metrics*, which relate to the objective completion of task goals; and *behavior metrics*, which assess flight safety and actions. All metrics stated are calculated over a single flight trajectory. Due to the difference in task settings, the same metrics from Chapter 4 cannot be reused here; for example mean position of the VIP would indicate how close the pilot maintains the VIP to the DOB but in the navigational flight task would be meaningless since the pilot here needs to fly around to collect waypoints (which are randomly located). Thus, analogous metrics are used here with analogies described when possible.

Performance Metrics

1. # of waypoints collected.

2. # of successes: did the pilot collect all waypoints?

3. # of failures: did the pilot crash before completion? Similar to # of crashes in the VIP study.

Behavior Metrics

1. Cross-Track Error (σ_{CTE}) [25, 126] – mathematically, CTE measures the lateral deviation from the ideal line between the start (or current) position of the plane to the target position (the active waypoint in this task). While the average CTE quantifies navigational accuracy (a useful metric as well), the standard deviation of CTE can serve as a proxy for "search struggle". A high σ_{CTE} would indicate that the pilot spends more time searching for waypoints – a likely indication of low spatial awareness of surroundings. An analogue in the VIP study is the $\mu|\theta|$ (°), where a high value indicates that the pilot has low spatial awareness as they move farther from the DOB.

2. Inversion Ratio ($\mathcal{R}_{inv}$) [113] – is defined as the percentage of total flight time where the plane was inverted, or at a bank angle of $\phi > 90°$. In the navigational task, inversion is not a planned maneuver; if pilots spend a high proportion of their time inverted, this indicates that they are vertically disoriented, and in normal circumstances, actions here may lead to a crash.

3. Maximum G-Force (G_{max}) [64] – is the maximum instantaneous acceleration experienced along the plane's vertical axis normalized by gravity. In real life, there are physical limits to the G-Force a plane and a human body can withstand. While there are no direct adverse effects of high G-forces in the flight simulator, a pilot who can complete the task without executing aggressive, high-risk maneuvers is generally safer.

4. Control Entropy ($\mathcal{C}_{ent}$) [140] - measures the subtlety and complexity in the pilot's control inputs. Sample entropy is used because it is frequently employed to quantify unpredictability in time-series data. In this task, low control entropy would indicate that the pilot is in control, applying small, subtle actions, whereas high control entropy would indicate erratic actions and a struggle to maintain control. This metric measures whether expert pilots exhibit control inputs similar to those observed in the MARS/VIP tasks: small, intermittent, and regular joystick deflections.

5. Control Extremity ($\mathcal{C}_{ext}$) [91] – quantifies the percentage of flight time where the pilot's inputs displayed extreme actions (defined as $> 90\%$ of total stick deflection). High levels of $\mathcal{C}_{ext}$ are indicative of "bang-bang" control behavior. This indicates that the pilot has forgone proportional and rhythmic control in favor of extreme corrective actions. In principle, $\mathcal{C}_{ext}$ measures the extent to which extreme and *destabilizing* actions hinder the task trajectory, inducing oscillations and/or eventually crashing; similar to % destab. in the VIP study.

6. Pilot-Induced Oscillation (PIO) [118] – measures the number of control reversals within a specified input channel (Roll or Pitch). It represents a sustained or uncontrollable oscillation resulting from the pilot's over-corrective efforts to control the plane. PIO is also an analogue metric in the VIP study; $\sigma(\theta)$, which signals whether the pilot is suffering from large and often uncontrollable oscillations of their own design.

5.3.4 Training AI Assistants

The candidate AI assistants in this study were trained using the Soft Actor-Critic (SAC) or Proximal Policy Optimization (PPO) policies from `stable-baselines3`. At the first attempt, both agents were trained in the updated environment for $50M$ timesteps. The PPO agent used a batch size of 4096 and an Entropy Coefficient of 0.01 to prevent early convergence to a low-reward optimum. Furthermore, the PPO policy network was upgraded from the default to a size of $[256, 256]$ for both the actor policy and value functions. Similarly, for the SAC agent, a batch size of 2048 and a gradient step size of 100 were used to ensure uniform learning. Furthermore, the SAC policy network was upgraded to a size of $[400, 300]$ for both the actor and critic policies. These hyperparameters were empirically derived. The larger network sizes (default: $[64, 64]$) were chosen, given the complexity of the flight task, with the assumption that the policies would benefit from increased nonlinearity in the networks.

Note that the agent is provided with a context length of 2 during training. This means that at any given time, the agent is aware of the current waypoint to be collected and the next waypoint in

the collection order. By receiving information about the next waypoint, the agent should optimize its flight path to collect all waypoints as quickly as possible.

Initial attempts to train an agent to complete the task were unsuccessful due to throttle constraints, using both SAC and PPO. The throttle constraint of keeping the plane's speed fixed was added to reduce task complexity and cognitive workload for human pilots; they only needed to navigate the plane along the roll and pitch axes, with no other controls. However, the AI agents relied heavily on the throttle to complete the task as quickly as possible, and adjusting the speed enabled them to perform a variety of maneuvers. To mitigate the effects of that constraint, curriculum learning was used to first train the agents without constraints, and the environment was modified to allow the RL agents to collect waypoints from a distance of 8 meters. After this first step, the agents were then further trained for another $20M$ timesteps with the required constraints. Table 5.1 shows the mean reward and metric statistics for each trained pilot after 2 steps of curriculum learning. The SAC agent was still unable to complete the task and did not progress to step 2 of the curriculum. The PPO agent successfully performed the task to the desired level and was selected as the agent for embodied ghost plane assistance.

5.3.5 Experimental Setup

The human subject experiment was reviewed and approved by CSU's Institutional Review Board (Protocol $\#4388$). In the experiment, human participants were asked to attend 2 sessions.

Session 1

In the first session, each participant completed 4 exercises;

1. 60-second training task to adjust to the controls, plane movements, and task scenario;
2. (**Alone-S1**) 5-minute task to ascertain their baseline performance;
3. (**Arrow**) 5-minute task with the situated arrow assistance mode;
4. (**Ghost-S1**) 5-minute task with the embodied ghost plane assistance mode.

Metric	SAC		PPO	
	No-C	**C**	**No-C**	**C**
Episodes	2	2	5	9
Mean Reward	-49.58 ± 15.17	23.09 ± 49.74	230.53 ± 176.19	281.69 ± 151.56
$\#Waypoints$	0	0	17	30
$\#Successes$	0	0	4	7
$\#Failures$	2	2	1	2
$\mathcal{R}_{inv}$	10.75	44.19	21.18	16.09
$\mathcal{C}_{ext}$	45.34	52.18	82.09	75.34
σ_{CTE}	37.85	12.13	9.54	9.80
$\mathcal{C}_{ent}$	0.27	0.43	0.28	0.52
G_{max}	17.76	5.63	8.23	9.46
PIO_{pitch}	699.50	530.50	77.60	116.44
PIO_{roll}	1693.50	1240.00	251.60	197.56

Table 5.1: Results of training the SAC and PPO policies as the underlying AI agent for the ghost plane mode. No-C indicates the first training step without any of added constraints and C indicates the second training step with the constraints added for the experiment.

During each task, if the human crashed the plane or collected all required waypoints, the task would reset to collect as many episodes of the navigational task as possible before time ran out. After each task, except the training task, participants completed a survey that assessed the NASA-TLX mental load index [74], the utility of the provided assistance, and their subjective trust in the AI (based on Muir [122]). At the end, participants were asked, in a long-form answer, which mode of assistance they preferred and why, using Arrow-S1 vs. Ghost-S1 for session 1 and Ghost-S2 vs. Ghost-S2-jit for session 2 (see below). They were also asked whether they had experienced spatial disorientation, dizziness, or nausea. The survey questionnaire can be found in the appendix (see Section C.1). Furthermore, to ensure a fair assessment of the effects of assistance modes and to avoid confounding the results with learning effects, the assistance conditions were counterbalanced.

(Re)training with Imitation Learning

Following the first session, the underlying AI driving the ghost plane's action policy was re-trained using imitation learning since one of the motivations for this work is studying the effect

of human action data on the embodiment of an agent, or how the learned policy of the RL agent within its environment is affected when it is exposed to data from humans who performed the embodied task in the same environment. To this end, human-collected data are used to prepare "expert" trajectories that are used to update the assistive agents via imitation learning. Based on results from the VIP human subject study (Chapter 4), I repeat the use of the behavior cloning and adversarial inverse RL (AIRL) algorithms, given their success in improving velocity-based metrics and the increased trust placed by humans.

To prepare expert trajectories, a few preprocessing steps were performed, depending on the imitation algorithm being used. For behavior cloning (BC), since it is a supervised learning algorithm designed to mimic exact human behavior, only trajectories in which human participants collected all waypoints during the task were used. These imports on the agent behavior that leads to successful completion, bad behavior that leads to crashes, or continuously flying around and not collecting waypoints were not added to the expert training trajectories.

The AIRL algorithm is designed to train a policy network against a discriminator that aims to diminish the learned policy from the expert and, at the same time, recover a reward function from the expert trajectories, thereby making it more generalizable to environmental changes. For this algorithm, it is beneficial to provide all trajectories, regardless of completion, i.e., whether the human completed the task, crashed the plane, or continued flying until time ran out. To compare the usefulness of the different kinds of trajectories, a data ablation test is carried out where the AI agent is trained with AIRL using the following trajectories: (a) only successful trajectories, (b) all trajectories including crashes. Additionally, data from different conditions were varied to determine whether this would affect the resulting agent. The results of retraining are displayed in Table 5.4 and further discussed in Section 5.5.2.

Session 2

Participants were then called back for a second session approximately 7-10 days after their first session. This was done to account for the learning effect due to continuous use and to allow enough time for participants to forget the task, following Chapter 4 and prior work [172–180].

In the second session, human subjects were asked to perform the following tasks:

1. (**Alone-S2**) 5-minute task to ascertain baseline performance;
2. (**Ghost-S2**) 5-minute task with the updated ghost plane assistance mode;
3. (**Ghost-S2-jit**) 5-minute task with the updated ghost plane assistance and the heuristic trigger for just-in-time assistance.

A similar survey was provided in Session 2 as well (see Section C.2). Additionally, the Ghost-S2 and Ghost-S2-jit conditions were counterbalanced.

Experimental Apparatus

Figure 5.4 shows the environment in which the experiment was conducted, using a 27" screen, a mouse and keyboard for the survey, and a Logitech FlightStick (same as in Chapter 4) for the tasks. The experiment was run on a System76 laptop with 16 GB of memory and an Nvidia GTX 1650 graphics card. The exercises were run at 60 FPS with data collected at the same rate. During each exercise, the collected data included the plane's orientation and position, waypoint positions, human actions, and AI actions (when ghost-plane assistance was provided).

Power Analysis

As a pre-experimental step, a power analysis was performed to determine the required sample size to study the effects of AI assistance. For this experiment, a power of 0.8 or 80%, a large effect size of 0.8 (Cohen's d), and α of 0.05 (probability of a Type I error) was set, using a two-tailed t-test, 26 samples per group was required. The experiment uses a within-subjects design which helps reduces noise from individual human performance, i.e., if a human participant's baseline task performance is poor, they remain a poor performer throughout the experiment, thereby making the effect of AI assistance more visible.

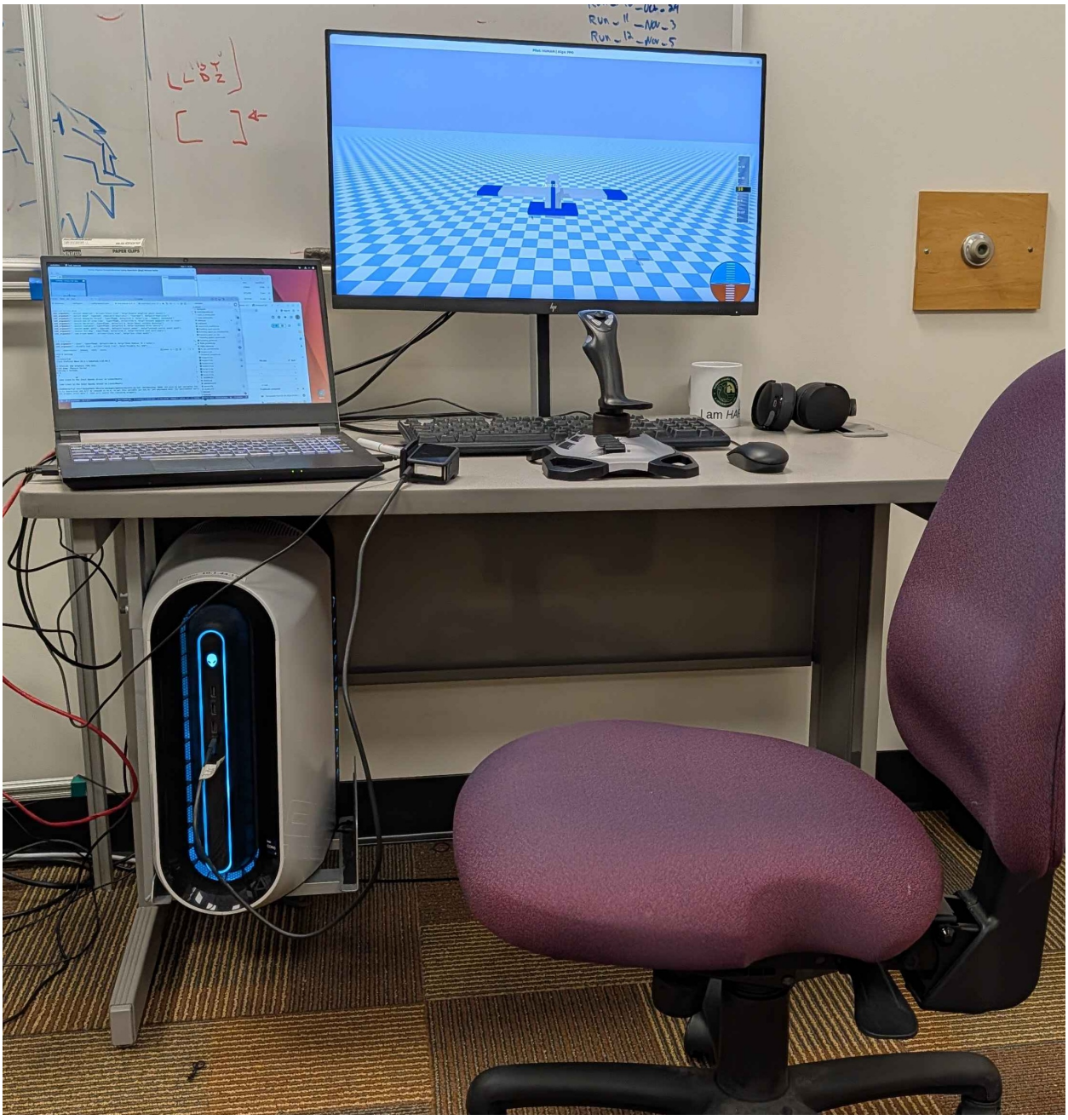


Figure 5.4: Experimental apparatus for the navigational flight study; participants used the Logitech flight stick to control the plane with the 27” screen as the primary display. The display is connected to the laptop, located to the left, which runs the program and stores the data.

Code Availability

The source code for running the human subject study and AI agent training developed for this chapter is publicly available on GitHub[10].

[10]https://github.com/sabdulm/HITL-flight-sim

5.4 Results

A total of 30 participants were recruited for this study; 3 were used in the pilot study, and of the remaining, only 1 participant could not complete the full 2-session experiment. The results that follow are based on data from the remaining 26 human participants (18 male and 8 female, average age: 27.71 ± 6.53) who successfully completed the entire experiment.

Following the protocol in the VIP study (Chapter 4), I first assess whether subjects had adapted or significantly learned the navigational task through adaptation across or within different sessions. I compare performance differences between Alone-S1 vs. Alone-S2. If the performance differences between Alone-S1 and Alone-S2 are non-significant, then no significant learning has occurred between sessions, and therefore, we can directly compare the performance gains achieved across conditions. I compute an objective score for each participant in each task, given by Equation 5.1,

$$S = \underbrace{\left(\frac{W}{\max_W} + \frac{Succ}{\max_{Succ}}\right)}_{\text{Achievement}} - \underbrace{\left(\frac{1}{n}\sum_{i=1}^{n}\frac{Neg_i}{\max_{Neg_i}}\right)}_{\text{Penalty Index}} \tag{5.1}$$

where: W and $Succ$ are total waypoints and successes, respectively, and Neg_i represents each of the 8 negative metrics (crashes, CTE, Inversion, Control Entropy, etc.). Each term is divided by the maximum value observed in the total sample space to normalize it to the $[0, 1]$ interval. Although the objective score was normally distributed according to the Shapiro-Wilk normality test, not all of the individual metrics that comprise the objective score were normally distributed. To keep comparisons fair and consistent, the Wilcoxon Signed-Rank test is used throughout, and, where relevant, the matched-pairs rank biserial correlation (r_{rb}) is also reported. Statistically significant adaptation to the task was not found between session baseline tasks (W-value 103, p-value 0.0669). It should be noted the p-value does indicate a trend toward task adaptation ($r_{rb} = .41$); these baseline performance shifts were notably smaller and less consistent than the primary effects observed under AI-assisted conditions.

Figure 5.5 displays the performance differences between the baseline solo task in each session with the respective AI conditions: Arrow-S1 and Ghost-S1 are compared against Alone-S1 in Session 1, and Ghost-S2 and Ghost-S2-jit are compared against Alone-S2 in Session 2.

The arrow condition helped subjects improve on the task objective (collecting the waypoints) quite significantly. All conditions reduced CTE SD in the subjects to a significant extent, but at the same time increased Control Extremity, indicating that more aggressive max-input maneuvers were performed when the subjects received AI assistance. Lastly, the Ghost-S1 condition increased the number of inversions performed by subjects and decreased the maximum G-Force and PIO pitch, but these effects were either reduced or insignificant in the S2 conditions.

Table 5.2 shows the distribution of scores for the 6 facets of NASA-TLX. Overall, the arrow condition improves the workload index by decreasing mental effort and frustration facets and significantly increasing self-reported performance rating. Although the Ghost-S1 condition also improved the self-performance rating, it increased the physical demand required to perform the task. It should be noted that subjects rated their performance more positively when they returned in Session 2 than in Session 1 (3.23 ± 1.58 vs. 2.19 ± 1.23), indicating a significant increase likely due to familiarity with the task. This shift is further supported by a large effect size in self-rated performance ($r_{rb} = .72$), which aligns with the near-significant performance trend ($p = .0669, r_{rb} = .41$) previously observed between sessions. Additionally, subjects were asked to rate the perceived impact of the AI condition on their performance and their trust in it, as detailed in Table 5.3. The most interesting result is that the Ghost-S1 condition received a significantly low trust rating, whereas the Ghost-S2 condition, retrained on human data with AIRL, was trusted even less. However, trust was recovered by the just-in-time Ghost-S2 condition, which might suggest that timely intervention, rather than continuous guidance, is considered helpful.

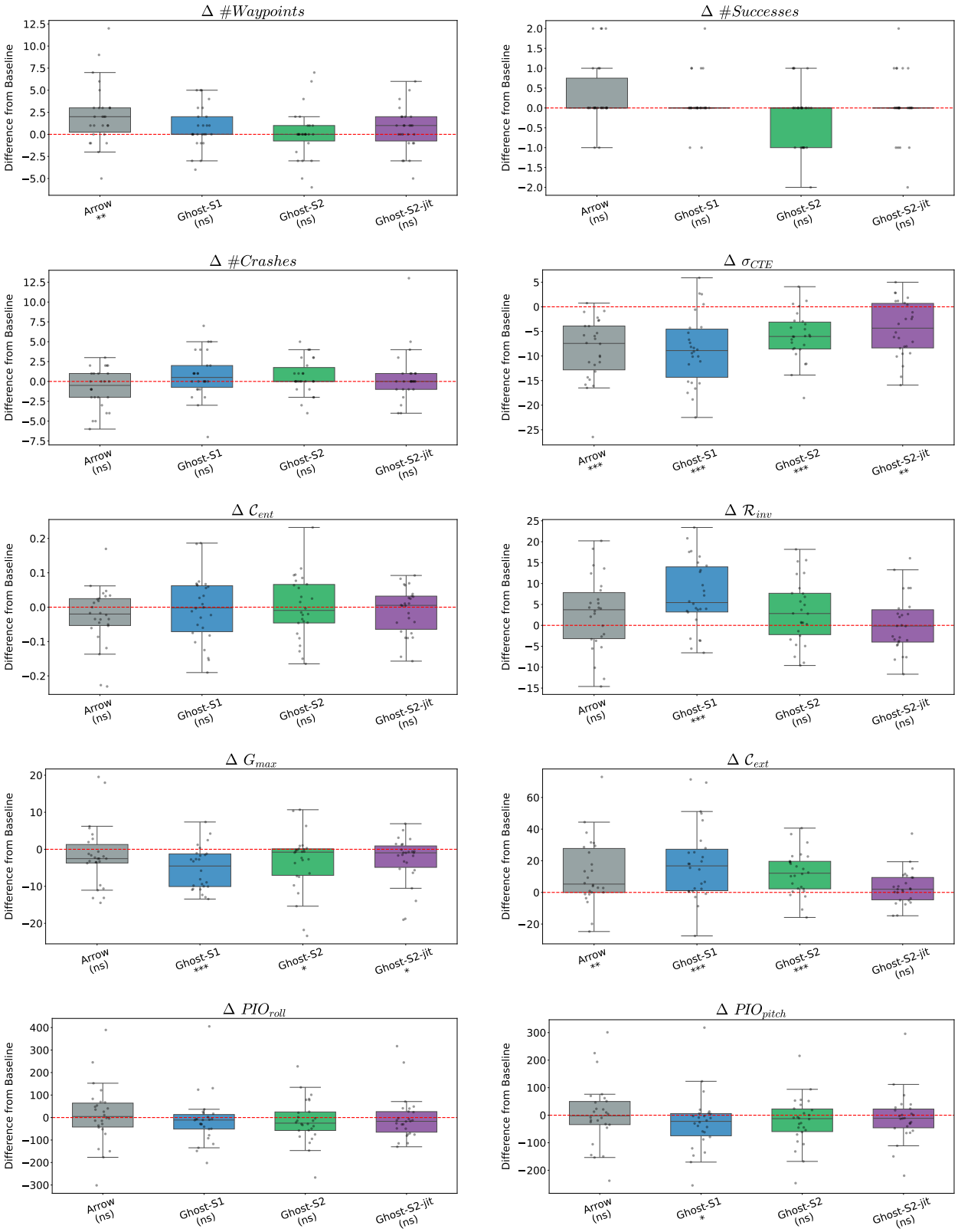


Figure 5.5: Performance differences between Session baselines and treatment conditions (Alone-S1 vs. Arrow and Ghost-S1; Alone-S2 vs. Ghost-S2 and Ghost-S2-jit). Significance levels $p < 0.05$, $p < 0.01$, $p < 0.001$: *, **, ***; ns = not significant.

Metric	Alone-S1	Arrow	Ghost-S1	Alone-S2	Ghost-S2	Ghost-S2-jit
Mental (↓)	5.04 ± 1.40	4.42 ± 1.50	4.73 ± 1.48	4.38 ± 1.24	4.69 ± 1.35	4.27 ± 1.25
Δ / *p-value*	–	-0.62 / **0.027**	-0.31 / 0.335	-0.66 / 0.068	+0.31 / 0.412	-0.11 / 0.731
Physical (↓)	3.23 ± 1.42	3.54 ± 1.39	3.92 ± 1.60	3.58 ± 1.42	4.04 ± 1.59	3.85 ± 1.59
Δ / *p-value*	–	+0.31 / 0.219	+0.69 / **0.005**	+0.35 / 0.282	+0.46 / **0.022**	+0.27 / 0.203
Temporal (↓)	4.54 ± 1.17	4.19 ± 1.20	4.42 ± 1.42	4.12 ± 1.51	4.50 ± 1.24	4.12 ± 1.42
Δ / *p-value*	–	-0.35 / 0.166	-0.12 / 0.719	-0.42 / 0.098	+0.38 / 0.147	0.00 / 0.954
Performance (↑)	2.19 ± 1.23	3.58 ± 1.50	2.77 ± 1.31	3.23 ± 1.58	2.77 ± 1.45	3.27 ± 1.19
Δ / *p-value*	–	+1.39 / **0.001**	+0.58 / **0.009**	+1.04 / **0.002**	-0.46 / 0.167	+0.04 / 0.922
Effort (↓)	5.46 ± 1.10	4.65 ± 1.23	5.00 ± 1.30	5.42 ± 1.10	5.31 ± 1.05	4.96 ± 1.18
Δ / *p-value*	–	-0.81 / **0.020**	-0.46 / 0.126	-0.04 / 0.852	-0.11 / 0.572	-0.46 / **0.049**
Frustration (↓)	4.77 ± 1.24	4.04 ± 1.64	4.73 ± 1.48	4.23 ± 1.34	4.69 ± 1.52	3.88 ± 1.37
Δ / *p-value*	–	-0.73 / **0.008**	-0.04 / 0.694	-0.54 / 0.076	+0.46 / 0.215	-0.35 / 0.206

Table 5.2: NASA TLX Metrics by Condition. (↑) indicates higher values are better; (↓) indicates lower values are better. Δ represents the mean difference from the respective session baseline (Alone-S1 vs Arrow/Ghost-S1; Alone-S2 vs Ghost-S2/jit; Alone-S1 vs Alone-S2). P-values represent paired Wilcoxon tests.

Condition	AI Impact (↑)		AI Trust (↑)	
	Mean ± SD	Δ / p-value	Mean ± SD	Δ / p-value
Arrow (S1)	5.69 ± 0.79	–	5.50 ± 1.30	–
Ghost-S1 (S1)	4.04 ± 1.48	-1.7 / **0.000**	3.73 ± 1.12	-1.8 / **0.000**
Ghost-S2 (S2)	4.00 ± 1.70	-0.0 / 0.849	3.38 ± 1.47	-0.3 / 0.195
Just-in-time (S2)	4.65 ± 1.20	+0.7 / 0.074	4.00 ± 1.10	+0.6 / **0.005**

Table 5.3: Evolution of Perceived AI Impact and Trust. (↑) indicates higher values are better. Δ represents the mean difference from the immediately preceding condition in the sequence. P-values represent paired Wilcoxon tests.

5.5 Discussion

5.5.1 Skill Group Analysis

Looking at the results in Figure 5.5 it seemed that the ghost plane assistance did not help that much at improving performance except for CTE SD and max G force. Furthermore, the ghost plane assistance significantly increased "bang-bang" control (Control Extremity) and aggressive inversions in human subjects. Before the performance hypothesis (H1) can be rejected, we should examine whether there is a particular group of subjects for whom AI assistance provided more benefit than others. For this analysis, human subjects are grouped by skill level; k-means clustering

is used to partition human performance data from the Alone-S1 task according to the defined task metrics. $K = 3$ is chosen, as evidence from the VIP and MARS tasks suggests that human performance can be divided into 3 groups (Good/Expert, Medium/Intermediate, Bad/Novice), with 4, 12, and 10 subjects assigned to these groups, respectively. Figure 5.6 shows a radar plot that defines each skill group in terms of each negative metric; larger areas for a metric show worse performance in that metric. The novice group typically crashes more often, spends more time inverted, and exhibits bang-bang behavior in their action inputs. Intermediates exhibit higher PIO, indicating that they may be spatially disoriented, and make more aggressive turns (higher max g-force). Experts in the task exhibit moderate PIO, along with high levels of control entropy, i.e., there are few regular patterns in their control inputs. This is surprising, since both experts and intermediates exhibit the same irregularities in their inputs, yet the less-skilled group exhibits more induced oscillations and aggressive maneuvers. It is possible that there is a nuance in how experts approach the task that these metrics do not fully capture.

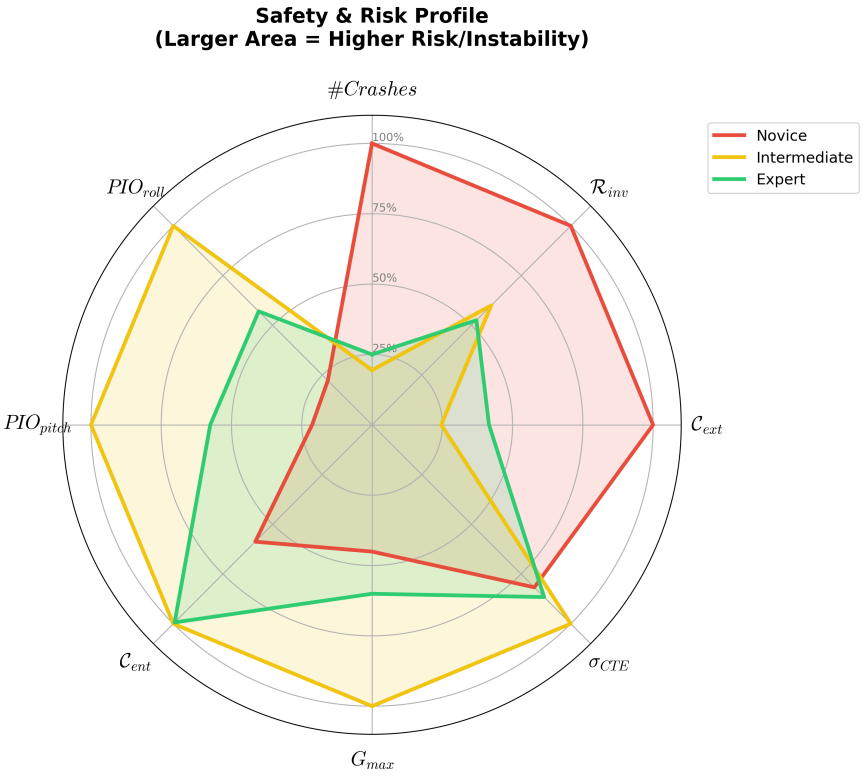


Figure 5.6: Performance clusters based on skill groups

Figure 5.7 displays the performance difference in each condition split by skill level. It is clear that the intermediate skill group benefits more from assistance, especially in the ghost plane mode; intermediates collect more waypoints and increase the number of successful trajectories. Additionally, with AI assistance, intermediates decrease aggressive maneuvers (max G-force), self-induced oscillations in both roll and pitch axes (PIO), and reduced searching in the arena (CTE SD). It should be noted that while the rank birserial correlation does indicate a medium to large effect of AI assistance, the overall power for the results is reduced for the smaller subset of human subjects, and thus the performance hypothesis (H1) is supported through findings, but weakly and partially for the intermediate skill group. However, the presence of a large effect size (rank-biserial correlation) across multiple metrics suggests that H1 hypothesis would likely reach full statistical significance with a larger sample population. Lastly, while AI assistance does reduce the struggle to locate waypoints by pointing subjects towards the active waypoint (CTE SD), other skill groups are not able to benefit as much; the expert skill group had reduced positive performance (in terms of number of waypoints collected and overall successful trajectories) and also displayed significantly higher inversions and bang-bang actions.

Furthermore, the arrow assistance increases the number of successful episodes and the number of waypoints for each skill group, but it fails to improve any other safety metric, except for CTE SD. The assistance mode hypothesis is supported by the observation that human subjects placed significantly more trust in the arrow (5.69 vs. 4.04 for Ghost-S1), as its intentions were unambiguous: the waypoint is in this direction. However, the ghost plane assistance mode improves safety behavior more than the arrow, at least in the intermediate skill group.

5.5.2 Agent selection after retraining

In the VIP study, the updated RL agent was selected based on the mean reward after the policy was evaluated for 5 episodes. Initially, the retrained agent in this flight study was going to be selected using the same method. Table 5.4 shows the mean rewards for a number of variants of the updated agent compared to the PPO baseline. One variant, AG-All (arrow+ghost-all), trained on

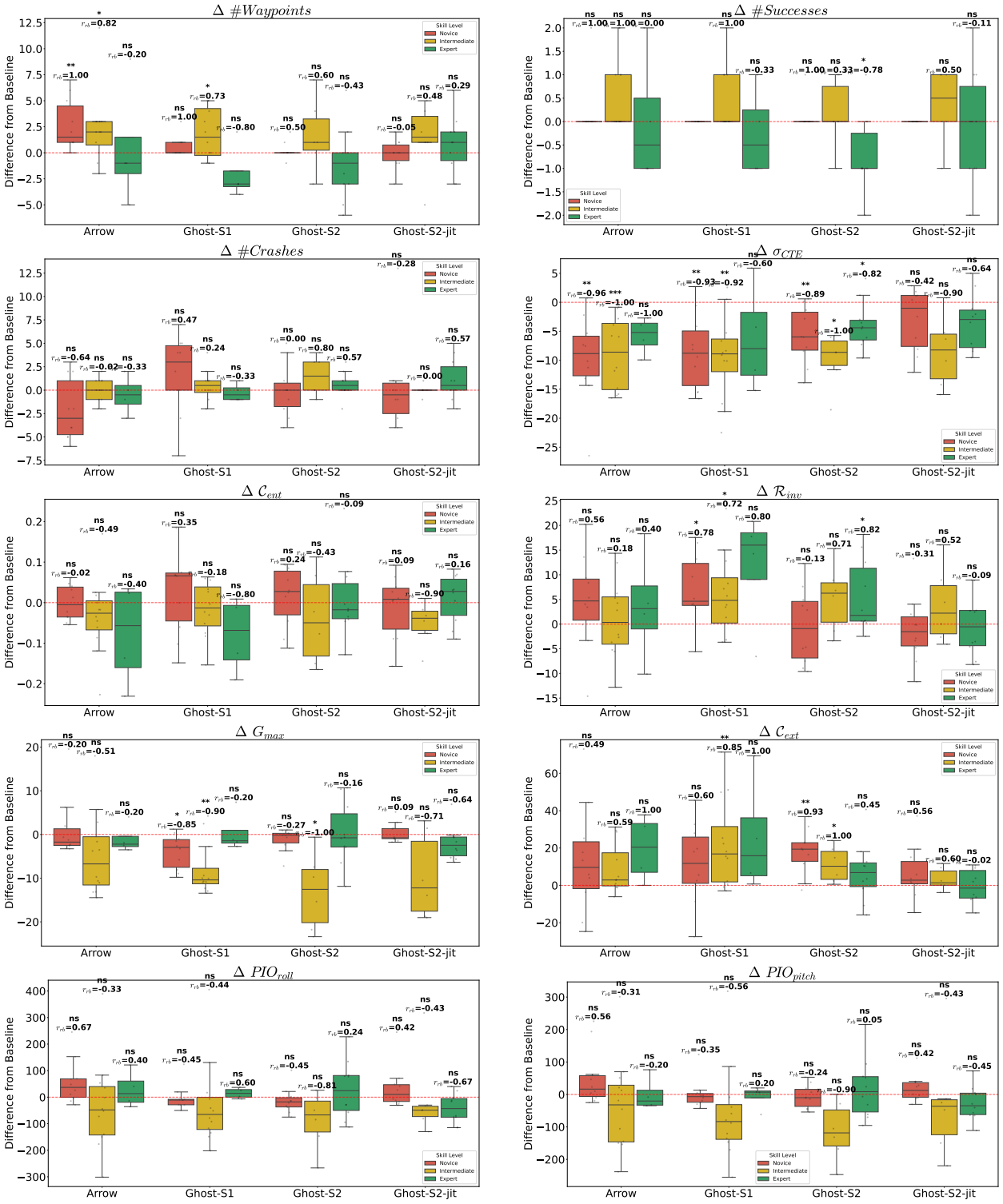


Figure 5.7: Performance differences between Session baselines and treatment conditions split by skill groups (Expert/Green, Intermediate/Yellow, Novice/Red). Individual plots show specific metrics, with Rank Biserial Correlation indicating the effect of AI assistance.

all episodes from the arrow and ghost task conditions, achieved a mean reward of 309.83, higher than the PPO baseline. Task metrics tell a different story, though; the AG-All agent was unable to complete the task even once during a 5-minute evaluation run. This suggests that the AIRL algorithm does sift through the noise in the human data but may have focused more on the wrong nuances, i.e., increasing flight distance. AIRL was able to fool the discriminator by selecting policies that resembled expert strategies but did not promote task objectives, as explained below

PyFlyt encourages waypoint navigation through a default dense reward function R_t that was calculated at each timestep t:

$$R_t = \max(k_p \cdot P_{progress}, 0) + \frac{k_d}{D_{target}} \tag{5.2}$$

where $P_{progress}$ denotes the agent's progress toward the subsequent waypoint, D_{target} is the Euclidean distance to the target, and k_p and k_d are scaling constants set to 3.0 and 1.0, respectively. Equation 5.2 indicates a part of the reward function programmed in the waypoints task in PyFlyt; this portion of the reward function promotes strategies that bring the plane closer to the waypoint. An agent can exhibit reward hacking in this task by using a strategy that increases the reward by proximity to the waypoints without ever collecting them.

The selected agent for session 2, ArrGh (arrow+ghost), used only the successful trajectories from both guidance conditions for training and completed 7 episodes in 5 minutes, slightly less than the PPO baseline. A finding from the study was that humans trusted the new ghost plane assistance even less than the original, rejecting the trust hypothesis (H3), suggesting that the ArrGh agent did not align with human intuition.

To further examine this, I used the Wasserstein distance[11] measure to compute a "H-likeness" score, which denotes how similar an AI agent's behavior is to humans in the same task. Using task metrics as a proxy for behavior, Table 5.4 shows the H-likeness scores for the PPO agent

[11] Wasserstein distance was chosen because (a) it is symmetric, i.e., difference between $A|B$ is the same as $B|A$; (b) it works even if samples are disjoint, unlike Kullback-Leibler (KL) divergence, which fails if there is no overlap; and (c) it can precisely quantify the effort required to convert one distribution into another, serving as a measure of distance.

	Baseline	All	Success-Only					
Task Data	**PPO**	**AG-All**	**Alon**	**Al-Gh**	**Al-Arr**	**Arr**	**ArrGh**	**Ghost**
Alone	—		✓	✓	✓			
Arrow	—	✓			✓	✓	✓	
Ghost	—	✓		✓			✓	✓
Metric								
Episodes	9	1	6	6	7	9	8	5
Mean Reward	281.69	309.83	268.45	268.37	268.60	278.46	274.48	274.74
Std Dev ($\pm$)	151.56	160.44	177.00	177.02	177.10	172.72	173.90	173.92
$\#Waypoints$	30	0	17	12	23	25	28	19
$\#Successes$	7	0	4	2	5	5	7	4
$\#Failures$	2	1	2	4	2	4	1	1
$\mathcal{R}_{inv}$	16.09	0.00	17.54	12.94	20.56	31.13	15.06	23.79
σ_{CTE}	9.80	31.53	16.09	21.55	16.71	15.19	14.06	17.97
G_{max}	9.46	39.01	9.24	9.56	9.60	8.79	7.67	10.54
$\mathcal{C}_{ext}$	75.34	0.0	81.98	80.14	78.70	78.23	75.71	84.02
PIO_{pitch}	116.44	0.00	100.00	79.00	130.00	86.11	113.13	123.80
PIO_{roll}	197.56	0.00	259.83	209.67	233.71	176.89	203.00	328.40
H-Likeness	10.62	18.79	11.23	10.24	11.69	11.48	11.56	14.46

Table 5.4: Comparative offline evaluation for variants of HITL-trained agents using AIRL. Variations include using different task data and trajectory status for training. H-Likeness is computed using Wasserstein distance to determine the similarity of behavior between Expert humans and Models. Lower values indicate greater proximity to human solo performance characteristics. Abbreviated agent names can be interpreted based on the data from which task they are trained on, and only successful trajectories are used unless mentioned otherwise. Column headers reflect data used to train the assistant as follows: AG-All (arrow+ghost tasks with all trajectories); Alon (alone); Al-Gh (alone+ghost); Al-Arr (alone+arrow); Arr (arrow); ArrGh (arrow+ghost); Ghost (ghost).

baseline (10.62) and the potential assistants. The chosen agent for the Ghost-S2 and Ghost-S2-jit conditions, ArrGh, was further from humans (11.56) than the PPO baseline. The decrease in human likeness provides some insight into why human subjects trusted the agent less in session 2, thereby supporting the *inverse* of the trust hypothesis (H3). The behavior hypothesis (H2) can be accepted as well, retraining of the PPO agent between sessions does increase its behavior likeness to humans, but does so at the severe cost of performance; the Al-Gh (alone+ghost) variant achieved the best H-likeness score (10.24) but can only complete 2 successful episodes in a 5 minute evaluation which is a $\sim$71% decrease in performance from the baseline (7 successful trajectories). Now, while an increase in human likeness is preferred, it should not come at the cost of such severe performance

degradation. It may be possible to retrain the agent's policy using HITL so that it can regain its performance while maintaining similar levels of human-like action behavior.

5.5.3 Temporal Intervention

The temporal intervention hypothesis (H5) examined whether a just-in-time intervention, which appears only when needed, would result in fewer crashes than continuous assistance. Furthermore, the hypothesis posited that continuous assistance would reduce user attention to the primary task of collecting waypoints. Results in Figure 5.5 show that there were no significant differences between the Ghost-S2 and Ghost-S2-jit conditions in both the number of crashes reduced and the number of waypoints collected. Additionally, human subjects exhibited only a slight reduction in task workload between the 2 conditions (Table 5.2). Human subjects were also asked to indicate their preferred assistive mode (continuous vs. just-in-time); approximately 84% preferred the just-in-time mode. Subjects reported that continuous assistance was a "distraction", often "hard to follow, disorienting, and mentally exhausting". Subjects also claimed that the just-in-time assistance gave them "more control" and "could leverage the information from they knew they were struggling". These findings support the H5 hypothesis: continuous assistance added more noise to the cognitive workload and would likely be ignored by human subjects. However, it also suggests that humans report preferences for assistive modes that do not necessarily objectively improve their performance.

5.5.4 Reported User Trust and Performance Impact

Table 5.3 shows that overall, human subjects trusted the arrow assistance the most (avg. trust score 5.69). User self-reports indicated that this form of assistance was easier to understand and cited difficulty in continuously orienting themselves to the ghost plane. Additionally, trust drops for the ghost plane assistance after retraining using HITL, but recovers significantly when using the just-in-time heuristic to provide assistance when needed. Figure 5.8 reveals an interesting finding when we break down subjective trust and perceived performance impact from AI assistance by skill levels. Experts in this flight simulation study trusted AI assistance more than other skill

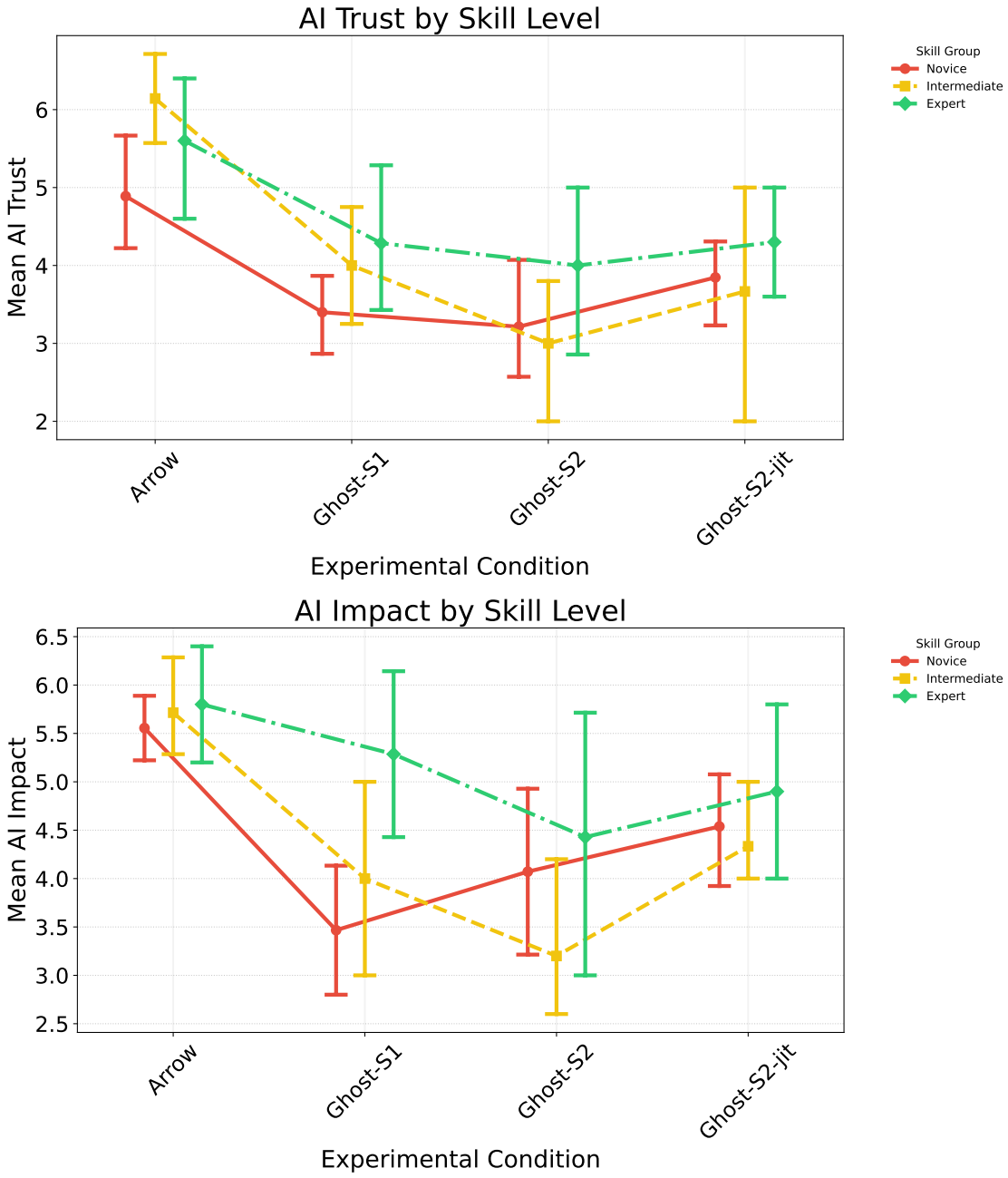


Figure 5.8: Average placed trust and perceived performance impact for AI assistance calculated from the subjective survey. Subjects answered questions "How did the AI's suggestions change your performance?" and "Overall, how much do you trust the AI?" on a 7-point Likert scale. Trust: 1 – no trust at all, 7 – complete trust. Impact: 1 – decreased performance significantly, 4 – no impact on performance, 7 – significant increase in performance.

levels. This is usually not the case in other domains such as medicine [59, 83]. Normally, experts are more discerning and skeptical of AI advice, whereas novices tend to over-rely on assistance [59, 83]. In this study, novices and intermediates are more critical judges of AI assistance and rate it as less trustworthy. Furthermore, novices are better at determining when AI is not helpful to their overall performance, whereas experts are overly optimistic when claiming that AI has improved their performance. Expert performers were, in fact, often hurt on metrics such as # waypoints, PIO_{roll}, $\mathcal{R}_{inv}$, and # crashes, whereas novices reported little impact on performance, which is supported by the objective metrics. It should be noted that while there is a decrease in trust in the ghost plane mode of assistance after HITL, the intermediate skill group places even lower trust in the Ghost-S2 assistant (mean ~ 3.2 – low trust) than in Ghost-S1 (mean ~ 4.0 – moderate trust). This suggests a correlation between trust and human-behavior similarity, as the Ghost-S2 condition has a greater H-likeness distance, indicating greater dissimilarity to human behavior than Ghost-S1.

5.6 Summary

In this chapter, I designed an experiment to evaluate AI-guided corrective maneuver guidance in a navigational flight task. Unsurprisingly, an AI agent can be trained to achieve high levels of task performance using reinforcement learning. Using that level of performance to enable humans to perform the same task is more challenging. I found that different modes of assistance, e.g., a simple situated arrow vs. an embodied ghost plane, aid human subjects in improving different performance metrics. The situated (non-embodied) arrow helped subjects improve positive task metrics such as the number of waypoints collected. The embodied ghost-plane assistance was more effective at improving safety-related metrics (max G-force, PIO, control entropy), especially for subjects with intermediate skill in the task. However, expert subjects were often harmed by the ghost plane assistance, as evidenced by reductions in their positive metrics. The results suggest that AI assistance is a potentially powerful decision aid for this task, but it should be used cautiously and judiciously. The ghost plane assistance shows potential as a decision aid for intermediate-

skilled pilots, but human subjects also report that continuous assistance can be distracting and disorienting, increasing cognitive load during the task. Furthermore, we observe that the subject skill group that benefits most from assistance does not place high trust in AI assistance; low trust would reduce the perceived impact and utility of the aid, and consequently decrease adoption of the system in the wild. These results suggest that while AI assistance is useful for the navigational flight task, it should be designed and implemented to preserve performance gains while fostering greater trust and perceived utility.

A limitation of using an open-source framework such as PyFlyt is limited control over the design of the cost function. A retrained embodied AI variant exhibited reward-hacking behavior. However, what was more surprising was that the AIRL-trained agent used in session 2 exhibited less human-like behavior than the original PPO baseline, leading to lower trust in the retrained agent. More importantly, increasing human likeness in AI often comes at the cost of severely reducing task performance, which is undesirable. The sample size ($N = 26$) contained more noise than expected in the human data, in which AI assistance was shown to be helpful for a smaller subset of subjects (12), thereby reducing the power of the analysis. In this study, as in the VIP study, I primarily focus on visual cues (arrows, ghost planes). In high-stress flight, the visual channel is might be overloaded by sensory illusions due to spatial disorientation, potentially reducing the efficacy of visual aids relative to auditory or haptic feedback.

Chapter 6

Conclusion

This dissertation explores the design and development of a first-of-a-kind AI system that provides real-time suggestions to humans in disorienting continuous-control tasks, namely the Visual Inverted Pendulum and a navigational flight task in PyFlyt. Using visual decision aids in disorienting action tasks and conducting a controlled human-subject evaluation, we identify several interesting findings regarding how performance, behavior, and trust interact. While it is not sufficient to say that AI globally improves or degrades performance, and while results depend on task setting and complexity, as well as baseline user expertise, some broad themes emerge.

We learned in Chapter 3 that an AI often makes objectively better decisions than humans in the MARS task given the same situation; likely due to the fact humans were disoriented while making their decisions (joystick deflections) while an AI is not subject to the same sensory illusions. When we put this intuition to the test in Chapter 4 in a real-time setting where humans receive guidance from AI, where in fact we see mixed results. Firstly, RL models, on average, perform better as solo performers and as better assistants than SL models, but humans perceive those same RL models as performing the task unintuitively and consequently rate them lower than SL models. It seems that humans place higher trust in assistance that may be suboptimal but demonstrates a more human-like balancing strategy. Similarly, in the navigational flight task (Chapter 5), we observe a similar trend, supported by empirical evidence, in which human subjects place greater trust in an assistant that is more similar in behavior.

From the perspective of how learning methodology affects AI "embodiment" within the task space, we observe markedly different strategies among AI agents trained with different learning algorithms, specifically reinforcement learning vs. supervised learning models. A reinforcement learning model can be an objectively superior task performer through trial and error, guided by a reward function that shapes its learned policy. Compare this to a supervised learning model

that learns through supervised pattern recognition, which may not always be a high performer, but performs actions that embody the problem space in a manner similar to that of humans.

It should be noted that an AI model can learn to perform each task to a high level of proficiency; this is unsurprising and not the main point of this study. These results demonstrate that high AI proficiency on the task does not always translate into the ability to serve as helpful assistants. While RL and supervised learning methods can train AIs to perform tasks well on their own, learning from human data does not automatically create an adept assistant. In the VIP digital twin study (Chapter 4), a MLP model, trained using data from Good and Medium level participants in the MARS task, was arguably the highest-performing AI in the VIP task when operating solo but was not the most effective assistant for any of the digital twin pilots (see Appendix B.7). An AI incapable of effectively aiding digital twins in a disorienting task could not be expected to help an actual human, where the risk of loss of life is even more critical.

HITL has been used as a foundational methodology for retraining AI agents to align AI behavior with that of humans. In the VIP task, retraining often led AI behavior to align more closely with that of humans, such as displaying similar oscillatory patterns around a focal pivot. In the case of the bad digital twin pilot, human retraining yields a model that takes better actions in extreme positions near the crash boundaries. In the flight study, however, HITL does lead to increased human likeness in the AI agents, but at the severe cost of performance. There are 2 reasons that I hypothesize why this happens; (a) the AIRL algorithm does sift through the noise in the task data and picks up suboptimal behavior from human actions; and (b) the suboptimal nuances are exaggerated in flight trajectories that do not result in success. What is surprising is that suboptimal behavior can trick the discriminator in AIRL into believing that the agent is an expert, and, when evaluated, the agent receives a higher reward than the baseline agent on the task. The learning of suboptimal actions in the task suggests that to support training AI assistance, the reward function in environments such as PyFlyt could be redesigned to focus on actions that promote the specific task objectives. In this case, using trajectories with disagreements between humans and AI, as in the VIP study, might have resulted in better performance and behavior.

Lastly, in the flight study, AI assistance appears to be a double-edged sword at times. AI assistance can improve the performance of those with some skill in the task (intermediates), but has limited utility for true novices and for those already skilled (experts). This finding mirrors trends observed in education and specialized training: humans who have a mental model of the problem and a sense of what the "right" answer is, either through educated guesses or past knowledge, benefit from the use of AI [6, 83]. Experts who do not need an AI would find it distracting, often hurting task performance. For true novices who lack knowledge of the task, display blind trust, and may be undermined by AI hallucinations, unexplainable assistance is often useless. Overall, evidence from both disorienting tasks shows that humans tend to trust assistive modes that are intuitive to follow and non-intrusive, but may not always objectively improve performance.

6.1 Future Directions

To extend this work, I propose 3 areas for investigation to develop AI assistance to combat spatial disorientation, with implications for AI assistance in various real-time tasks.

6.1.1 Modality of assistance

The primary modality for providing guidance to humans tested in this work has been visual: 1D arrows in the VIP study and 3D arrows or a ghost plane in the flight study. Visual modality is typically a reasonable method for providing assistance, but under high stress, it may not always be effective. When a pilot is spatially disoriented, sensory illusions can affect their decision-making processes, and these illusions may also affect their visual perception. In real settings, pilots may not be able to achieve the same benefits from visual assistance as observed in controlled experiments. Experiments with the use of visual, haptic, and audible assists need to be conducted in both controlled (VIP/PyFlyt) and real settings to determine the efficacy of each modality (as in [179, 51]). For such experiments, the level of assistance across modalities should be comparable in terms of the information provided to enable a reasonable comparison.

6.1.2 Control vs. Autonomy

It is natural to ask: should AI always recommend actions to humans, or should it have the ability to take over control if deemed necessary and return control only when danger is mitigated? In the case of a pilot suffering from spatial disorientation, if the system determines the pilot is not following corrective guidance appropriately, then taking control over from the pilot makes sense to ensure safety. But following Shneiderman's critique of current automation system designs, any assistive AI system must be developed with a human-centered philosophy, i.e., with high levels of automation, humans should still be able to override AI actions if they deem its utility has decreased, or it threatens task objectives. Multiple user studies will need to be conducted to determine the optimal balance of shared control between humans and AI and to assess its impact on trust across the multitude of conditions that can occur during flight.

6.1.3 Skill-based Adaptive Assistance

Lastly, as observed in the flight study, AI assistance can be detrimental to task performance for humans with specific skills. It would be highly beneficial if an AI could monitor and assess task proficiency and, consequently, adjust the level of assistance provided. For example, if an expert requires help, the AI system would provide just enough assistance to avoid being a nuisance and distracting the user. But if a novice requires help, the AI might provide additional information to increase context and, if using an audible or natural-language medium, more verbose, less-technical language, thereby providing better assistance. Furthermore, an AI could monitor the pilot's current mental and physiological state to further adjust the level and type of assistance. For example, in the flight study, it could choose between a more intuitive form of assistance, such as an arrow, or higher-fidelity ghost-plane assistance. Using skill-based adaptation, an AI system could even determine if it needs to take over control completely from the pilot. This adds another level of complexity and ties into the control-versus-autonomy research area proposed above.

Bibliography

[1] Dor Abrahamson and Robb Lindgren. 2014. Embodiment and embodied design. (2014).

[2] Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. 2023. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774* (2023).

[3] Garima Agrawal, Tharindu Kumarage, Zeyad Alghamdi, and Huan Liu. 2023. Can knowledge graphs reduce hallucinations in llms?: A survey. *arXiv preprint arXiv:2311.07914* (2023).

[4] Muhammad Aurangzeb Ahmad, Ilker Yaramis, and Taposh Dutta Roy. 2023. Creating trustworthy llms: Dealing with hallucinations in healthcare ai. *arXiv preprint arXiv:2311.01463* (2023).

[5] Airbus. 2020. Airbus concludes ATTOL with fully autonomous flight tests. (2020). https://www.airbus.com/en/newsroom/press-releases/2020-06-airbus-concludes-attol-with-fully-autonomous-flight-tests

[6] Ryan T. Allen and Prithwiraj (Raj) Choudhury. 2021. Algorithm-Augmented Work and Domain Experience: The Countervailing Forces of Ability and Aversion. *Organization Science* (2021). https://api.semanticscholar.org/CorpusID:249336258

[7] Georgios Alogdianakis, Ioannis Katsidimas, Athanasios Kotzakolios, Anastasios Plioutsias, and Vassilis Kostopoulos. 2024. An Embedded Decision Support System for Runway Safety and Excursion Avoidance. *arXiv preprint arXiv:2407.02504* (2024).

[8] Muhannad Alomari, Paul Duckworth, Nils Bore, Majd Hawasly, David C Hogg, and Anthony G Cohn. 2017. Grounding of human environments and activities for autonomous robots. In *IJCAI-17 Proceedings*. Lawrence Erlbaum Associates, Inc., 1395–1402.

[9] Muhannad Alomari, Paul Duckworth, David Hogg, and Anthony Cohn. 2017. Natural language acquisition and grounding for embodied robotic systems. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 31.

[10] Charles W Anderson. 1989. Learning to control an inverted pendulum using neural networks. *IEEE Control Systems Magazine* 9, 3 (1989), 31–37.

[11] Melchor J. Antunano. 2005. *Spatial Disorientation*. Technical Report AM-400-03/1. Federal Aviation Administration. https://rosap.ntl.bts.gov/view/dot/39656 Civil Aerospace Medical Institute.

[12] Brenna D Argall, Sonia Chernova, Manuela Veloso, and Brett Browning. 2009. A survey of robot learning from demonstration. *Robotics and autonomous systems* 57, 5 (2009), 469–483.

[13] Annalisa Baicchi, Rémi Digonnet, Jodi L Sandford, et al. 2018. *Sensory perceptions in language, embodiment and epistemology*. Vol. 42. Springer.

[14] Sourav Banerjee, Ayushi Agarwal, and Saloni Singla. 2024. Llms will always hallucinate, and we need to live with this. *arXiv preprint arXiv:2409.05746* (2024).

[15] Lawrence W Barsalou, Paula M Niedenthal, Aron K Barbey, and Jennifer A Ruppert. 2003. Social embodiment. *Psychology of learning and motivation* 43 (2003), 43–92.

[16] Andrew G Barto, Richard S Sutton, and Charles W Anderson. 1983. Neuronlike adaptive elements that can solve difficult learning control problems. *IEEE transactions on systems, man, and cybernetics* 5 (1983), 834–846.

[17] Hannah M Baumgartner, Jason Sigmon, Austin Ciesielski, and Russell J Lewis. 2025. Spatial Disorientation in Fatal General Aviation Accidents (2003–2021). (2025).

[18] Feryal Behbahani, Kyriacos Shiarlis, Xi Chen, Vitaly Kurin, Sudhanshu Kasewa, Ciprian Stirbu, Joao Gomes, Supratik Paul, Frans A Oliehoek, Joao Messias, et al. 2019. Learning

from demonstration in the wild. In *2019 International Conference on Robotics and Automation (ICRA)*. IEEE, 775–781.

[19] Herbert H Bell and Wayne L Waag. 1998. Evaluating the effectiveness of flight simulators for training combat skills: A review. *The international journal of aviation psychology* 8, 3 (1998), 223–242.

[20] Hymalai Bello, Daniel Geißler, Lala Ray, Stefan Müller-Divéky, Peter Müller, Shannon Kittrell, Mengxi Liu, Bo Zhou, and Paul Lukowicz. 2024. Towards certifiable AI in aviation: landscape, challenges, and opportunities. *arXiv preprint arXiv:2409.08666* (2024).

[21] Emily M Bender and Alexander Koller. 2020. Climbing towards NLU: On meaning, form, and understanding in the age of data. In *Proceedings of the 58th annual meeting of the association for computational linguistics*. 5185–5198.

[22] Aude Billard, Sylvain Calinon, Ruediger Dillmann, and Stefan Schaal. 2008. Survey: Robot programming by demonstration. *Springer handbook of robotics* (2008), 1371–1394.

[23] Bartosz Binias, Dariusz Myszor, Henryk Palus, and Krzysztof A Cyran. 2020. Prediction of pilot's reaction time based on EEG signals. *Frontiers in neuroinformatics* 14 (2020), 6.

[24] Robert H Bonczek, Clyde W Holsapple, and Andrew B Whinston. 2014. *Foundations of decision support systems*. Academic Press.

[25] Jan Boril, Vladimir Smrz, Erik Blasch, and Mudassir Lone. 2020. Spatial disorientation impact on the precise approach in simulated flight. *Aerospace Medicine and Human Performance* 91, 10 (2020), 767–775.

[26] Malcolm G Braithwaite, Simon J Durnford, John S Crowley, Norberto R Rosado, and John P Albano. 1998. Spatial disorientation in US Army rotary-wing operations. *Aviation, space, and environmental medicine* 69, 11 (1998), 1031–1037.

[27] Cynthia Breazeal. 2003. Emotion and sociable humanoid robots. *Int. J. Hum.-Comput. Stud.* 59, 1–2 (July 2003), 119–155. https://doi.org/10.1016/S1071-5819(03)00018-1

[28] Greg Brockman, Vicki Cheung, Ludwig Pettersson, Jonas Schneider, John Schulman, Jie Tang, and Wojciech Zaremba. 2016. Openai gym. *arXiv preprint arXiv:1606.01540* (2016).

[29] Rodney A. Brooks. 1991. Intelligence without reason. In *Proceedings of the 12th International Joint Conference on Artificial Intelligence - Volume 1* (Sydney, New South Wales, Australia) *(IJCAI'91)*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 569–595.

[30] Rodney A. Brooks. 1991. Intelligence without representation. *Artif. Intell.* 47, 1–3 (Feb. 1991), 139–159. https://doi.org/10.1016/0004-3702(91)90053-M

[31] Sylvain Calinon and Dongheui Lee. 2017. Learning control. In *Humanoid robotics: A reference*. Springer Netherlands, 1–52.

[32] Peter Cariani. 1992. Some epistemological implications of devices which construct their own sensors and effectors. *Towards a practice of autonomous systems* (1992), 484–493.

[33] Cristiano Castelfranchi. 2013. Alan Turing's "Computing machinery and intelligence". *Topoi* 32, 2 (2013), 293–299.

[34] Yen-Chun Chen, Linjie Li, Licheng Yu, Ahmed El Kholy, Faisal Ahmed, Zhe Gan, Yu Cheng, and Jingjing Liu. 2020. Uniter: Universal image-text representation learning. In *European Conference on Computer Vision*. Springer, 104–120.

[35] Kyunghyun Cho, Bart Van Merriënboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. 2014. Learning phrase representations using RNN encoder-decoder for statistical machine translation. *arXiv preprint arXiv:1406.1078* (2014).

[36] Ronald Chrisley and Tom Ziemke. 2006. Embodiment. *Encyclopedia of cognitive science* (2006).

[37] Paul F Christiano, Jan Leike, Tom Brown, Miljan Martic, Shane Legg, and Dario Amodei. 2017. Deep reinforcement learning from human preferences. *Advances in neural information processing systems* 30 (2017).

[38] Andy Clark. 1999. An embodied cognitive science? *Trends in cognitive sciences* 3, 9 (1999), 345–351.

[39] Marvin S Cohen, RAJA Parasuraman, DANIEL Serfaty, and R Andes. 1997. Trust in decision aids: A model and a training strategy. *Arlington, VA: Cognitive Technologies, Inc* (1997).

[40] Jason Corso. 2025. VIGIL AI in Rural Healthcare Outreach Project — Nov 2025 Demo of Lower Limb DVT Use Case. https://www.youtube.com/watch?v=qL46iG-To-I YouTube video.

[41] Jason Corso. 2025. VIGIL AI in Rural Healthcare Outreach Project — Nov 2025 Intelligent Task Guidance. https://www.youtube.com/watch?v=7-FjCJlS3gU YouTube video.

[42] Patricia S Cowings, William B Toscano, Millard F Reschke, and Addis Tsehay. 2018. Psychophysiological assessment and correction of spatial disorientation during simulated Orion spacecraft re-entry. *International Journal of Psychophysiology* 131 (2018), 102–112.

[43] Ronald Daiker, Kyle Ellis, Santosh Mathan, and W Redmond. 2018. Use of real-time, predictive human modeling for spatial disorientation detection and mitigation. In *Proceedings of the Modsim World 2018 Conference*.

[44] Defense Advanced Research Projects Agency (DARPA). [n. d.]. Perceptually-enabled Task Guidance. https://www.darpa.mil/research/programs/perceptually-enabled-task-guidance. Accessed: 2026-02-08.

[45] Dominik Dellermann, Philipp Ebel, Matthias Söllner, and Jan Marco Leimeister. 2019. Hybrid intelligence. *Business & Information Systems Engineering* 61, 5 (2019), 637–643.

[46] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*. Association for Computational Linguistics, Minneapolis, Minnesota, 4171–4186. https://doi.org/10.18653/v1/N19-1423

[47] Weishan Dong, Jian Li, Renjie Yao, Changsheng Li, Ting Yuan, and Lanjun Wang. 2016. Characterizing driving styles with deep learning. *arXiv preprint arXiv:1607.03611* (2016).

[48] Danny Driess, Fei Xia, Mehdi SM Sajjadi, Corey Lynch, Aakanksha Chowdhery, Brian Ichter, Ayzaan Wahid, Jonathan Tompson, Quan Vuong, Tianhe Yu, et al. 2023. Palm-e: An embodied multimodal language model. *arXiv preprint arXiv:2303.03378* (2023).

[49] Huiying Du, Ge Zhu, and Jiali Zheng. 2021. Why travelers trust and accept self-driving cars: An empirical study. *Travel behaviour and society* 22 (2021), 1–9.

[50] Jiafei Duan, Samson Yu, Hui Li Tan, Hongyuan Zhu, and Cheston Tan. 2022. A survey of embodied ai: From simulators to research tasks. *IEEE Transactions on Emerging Topics in Computational Intelligence* 6, 2 (2022), 230–244.

[51] Predictive Dynamics. [n. d.]. Fly with ai. https://www.predictivedynamics.ai/homepage

[52] Razvan V Florian. 2007. Correct equations for the dynamics of the cart-pole system. *Center for Cognitive and Neural Studies (Coneural), Romania* (2007), 63.

[53] Advanced Research Projects Agency for Health (ARPA-H). [n. d.]. Platform Accelerating Rural Access to Distributed and Integrated Medical Care. https://arpa-h.gov/explore-funding/programs/paradigm. Accessed: 2026-02-08.

[54] Paul Formosa. 2021. Robot autonomy vs. human autonomy: social robots, artificial intelligence (AI), and the nature of autonomy. *Minds and Machines* 31, 4 (2021), 595–616.

[55] Nicholas C Forrest, Raymond R Hill, and Phillip R Jenkins. 2022. An air force pilot training recommendation system using advanced analytical methods. *INFORMS Journal on Applied Analytics* 52, 2 (2022), 198–209.

[56] Jamilah Foucher, Anne-Claire Collet, Kevin Le Goff, Thomas Rakotomamonjy, Valérie Juppet, Thomas Descatoire, Jerémie Landrieu, Marielle Plat-Robain, François Denquin, Arthur J Grunwald, et al. 2022. Simulation and classification of spatial disorientation in a flight use-case using vestibular stimulation. *IEEE Access* 10 (2022), 104242–104269.

[57] Justin Fu, Katie Luo, and Sergey Levine. 2018. Learning Robust Rewards with Adverserial Inverse Reinforcement Learning. In *International Conference on Learning Representations*.

[58] Xiaofeng Gao, Ran Gong, Tianmin Shu, Xu Xie, Shu Wang, and Song-Chun Zhu. 2019. Vrkitchen: an interactive 3d virtual environment for task-oriented learning. *arXiv preprint arXiv:1903.05757* (2019).

[59] Susanne Gaube, Harini Suresh, Martin Raue, Alexander Merritt, Seth J. Berkowitz, Eva Lermer, Joseph F. Coughlin, John V. Guttag, Errol Colak, and Marzyeh Ghassemi. 2021. Do as AI say: susceptibility in deployment of clinical decision-aids. *NPJ Digital Medicine* 4 (2021). https://api.semanticscholar.org/CorpusID:231957153

[60] Zorik Gekhman, Gal Yona, Roee Aharoni, Matan Eyal, Amir Feder, Roi Reichart, and Jonathan Herzig. 2024. Does Fine-Tuning LLMs on New Knowledge Encourage Hallucinations? *arXiv preprint arXiv:2405.05904* (2024).

[61] Tejas P Ghuntla, Hemant B Mehta, Pradnya A Gokhale, and Chinmay J Shah. 2014. A comparison and importance of auditory and visual reaction time in basketball players. *Saudi Journal of Sports Medicine* 14, 1 (2014), 35–38.

[62] Randy Gibb, Bill Ercoline, and Lauren Scharff. 2011. Spatial disorientation: decades of pilot fatalities. *Aviation, space, and environmental medicine* 82, 7 (2011), 717–724.

[63] Debora Gil, Aura Hernández-Sabaté, Julien Enconniere, Saryani Asmayawati, Pau Folch, Juan Borrego-Carazo, and Miquel Angel Piera. 2021. E-Pilots: A system to predict hard landing during the approach phase of commercial flights. *IEEE Access* PP (2021), 1–1. https://api.semanticscholar.org/CorpusID:245469710

[64] Jaime Gil-Cabrera, José Francisco Tornero Aguilera, Miguel Angel Sanchez-Tena, Cristina Alvarez-Peregrina, Carolina Valbuena-Iglesias, and Vicente Javier Clemente-Suárez. 2021. Aviation-associated spatial disorientation and incidence of visual illusions survey in military pilots. *The International Journal of Aerospace Psychology* 31, 1 (2021), 17–24.

[65] Melita J Giummarra, Stephen J Gibson, Nellie Georgiou-Karistianis, and John L Bradshaw. 2008. Mechanisms underlying embodiment, disembodiment and loss of embodiment. *Neuroscience & Biobehavioral Reviews* 32, 1 (2008), 143–160.

[66] Adam Gleave, Mohammad Taufeeque, Juan Rocamonde, Erik Jenner, Steven H Wang, Sam Toyer, Maximilian Ernestus, Nora Belrose, Scott Emmons, and Stuart Russell. 2022. imitation: Clean imitation learning implementations. *arXiv preprint arXiv:2211.11972* (2022).

[67] Geoffrey Gorisse, Olivier Christmann, Etienne Armand Amato, and Simon Richir. 2017. First-and third-person perspectives in immersive virtual environments: presence and performance analysis of embodied users. *Frontiers in Robotics and AI* 4 (2017), 33.

[68] David M Green and Susanne M Von Gierke. 1984. Visual and auditory choice reaction times. *Acta psychologica* 55, 3 (1984), 231–247.

[69] Shivam Gupta, Sachin Modgil, Samadrita Bhattacharyya, and Indranil Bose. 2022. Artificial intelligence for decision support systems in the field of operations research: review and future scope of research. *Annals of Operations Research* 308, 1 (2022), 215–274.

[70] Hyowon Gweon, Judith Fan, and Been Kim. 2023. Socially intelligent machines that learn from humans and help humans learn. *Philosophical Transactions of the Royal Society A* 381, 2251 (2023), 20220048.

[71] Tuomas Haarnoja, Aurick Zhou, Pieter Abbeel, and Sergey Levine. 2018. Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor. In *International conference on machine learning*. PMLR, 1861–1870.

[72] Peter A Hancock, Deborah R Billings, Kristin E Schaefer, Jessie YC Chen, Ewart J De Visser, and Raja Parasuraman. 2011. A meta-analysis of factors affecting trust in human-robot interaction. *Human factors* 53, 5 (2011), 517–527.

[73] Chenru Hao, Xiaoya Fan, Chunnan Dong, Lihua Qiao, Xinwei Li, Xiuyuan Li, Li Cheng, Lisha Guo, and Ruibin Zhao. 2020. A classification method for unrecognized spatial disorientation based on perceptual process. *Ieee Access* 8 (2020), 140654–140660.

[74] Sandra G Hart. 2006. NASA-task load index (NASA-TLX); 20 years later. In *Proceedings of the human factors and ergonomics society annual meeting*, Vol. 50. Sage publications Sage CA: Los Angeles, CA, 904–908.

[75] Robert T Hays, John W Jacobs, Carolyn Prince, and Eduardo Salas. 1992. Flight simulator training effectiveness: A meta-analysis. *Military psychology* 4, 2 (1992), 63–74.

[76] Monika Hengstler, Ellen Enkel, and Selina Duelli. 2016. Applied artificial intelligence and trust–The case of autonomous vehicles and medical assistance devices. *Technological Forecasting and Social Change* 105 (2016), 105–120.

[77] Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural computation* 9, 8 (1997), 1735–1780.

[78] Jakob Hollenstein, Sayantan Auddy, Matteo Saveriano, Erwan Renaudo, and Justus Piater. 2022. Action Noise in Off-Policy Deep Reinforcement Learning: Impact on Exploration and Performance. *Transactions on Machine Learning Research* (2022).

[79] Ronghang Hu, Amanpreet Singh, Trevor Darrell, and Marcus Rohrbach. 2020. Iterative answer prediction with pointer-augmented multimodal transformers for TextVQA. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 9992–10002.

[80] Peter J Huber. 1992. Robust estimation of a location parameter. In *Breakthroughs in statistics: Methodology and distribution*. Springer, 492–518.

[81] Miranda Iersel, L.J.H.M. Kester, J. Bergmans, P. Hiemstra, Koen Benoist, and Bas Van den Broek. 2008. Creating shared situation awareness in a multi-platform sensor network. 1 – 6. https://doi.org/10.1109/ICIF.2008.4632245

[82] Nikolai Ilinykh and Simon Dobnik. 2022. Attention as Grounding: Exploring Textual and Cross-Modal Attention on Entities and Relations in Language-and-Vision Transformer. In *Findings of the Association for Computational Linguistics: ACL 2022*. 4062–4073.

[83] Kori Inkpen, Shreya Chappidi, Keri Mallari, Besmira Nushi, Divya Ramesh, Pietro Michelucci, Vani Mandava, Libuvse Hannah Vepvrek, and Gabrielle Quinn. 2022. Advancing Human-AI Complementarity: The Impact of User Expertise and Algorithmic Tuning on Joint Decision Making. *ACM Transactions on Computer-Human Interaction* 30 (2022), 1 – 29. https://api.semanticscholar.org/CorpusID:251622535

[84] Ziwei Ji, Delong Chen, Etsuko Ishii, Samuel Cahyawijaya, Yejin Bang, Bryan Wilie, and Pascale Fung. 2024. LLM Internal States Reveal Hallucination Risk Faced With a Query. *arXiv preprint arXiv:2407.03282* (2024).

[85] Yuande Jiang, Weiwen Deng, Jinsong Wang, and Bing Zhu. 2018. *Studies on drivers' driving styles based on inverse reinforcement learning*. Technical Report. SAE Technical Paper.

[86] Phillip Johnston and Rozi Harris. 2019. The Boeing 737 MAX saga: lessons for software organizations. *Software Quality Professional* 21, 3 (2019), 4–12.

[87] Ravneet Kaur, Monika Jain, Ryan M McAdams, Yao Sun, Shubham Gupta, Raghava Mutharaju, Su Jin Cho, Satish Saluja, Jonathan P Palma, Avneet Kaur, et al. 2023. An ontology and rule-based clinical decision support system for personalized nutrition recommendations in the neonatal intensive care unit. *IEEE Access* 11 (2023), 142433–142446.

[88] Peter GW Keen. 1980. Decision support systems: a research perspective. In *Decision support systems: Issues and challenges: Proceedings of an international task force meeting*. 23–44.

[89] Ibrahim Khebour, Richard Brutti, Indrani Dey, Rachel Dickler, Kelsey Sikes, Kenneth Lai, Mariah Bradford, Brittany Cates, Paige Hansen, Changsoo Jung, et al. 2024. When Text and Speech are Not Enough: A Multimodal Dataset of Collaboration in a Situated Task. *Journal of Open Humanities Data* 10, 1 (2024).

[90] Diederik P Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. In *International Conference on Learning Representations*.

[91] David H Klyde, Philip C Schulze, and Peter M Thompson. 2015. Exposing unique pilot behaviors from flight test data. In *AIAA Atmospheric Flight Mechanics Conference*. 0239.

[92] Satwik Kottur, Seungwhan Moon, Alborz Geramifard, and Babak Damavandi. 2021. SIMMC 2.0: A Task-oriented Dialog Dataset for Immersive Multimodal Conversations. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*. 4903–4912.

[93] Nikhil Krishnaswamy, Pradyumna Narayana, Rahul Bangar, Kyeongmin Rim, Dhruva Patil, David McNeely-White, Jaime Ruiz, Bruce Draper, Ross Beveridge, and James Pustejovsky. 2020. Diana's World: A Situated Multimodal Interactive Agent. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 34. 13618–13619.

[94] Markus Kuderer, Shilpa Gulati, and Wolfram Burgard. 2015. Learning driving styles for autonomous vehicles from demonstration. In *2015 IEEE international conference on robotics and automation (ICRA)*. IEEE, 2641–2646.

[95] Stephane Lallee and Paul F.M.J. Verschure. 2015. How? Why? What? Where? When? Who? Grounding Ontology in the Actions of a Situated Social Agent. *Robotics* 4, 2 (2015), 169–193. https://doi.org/10.3390/robotics4020169

[96] Alfred T Lee. 2017. *Flight simulation: virtual environments in aviation*. Routledge.

[97] Gyeong-Geon Lee, Ehsan Latif, Xuansheng Wu, Ninghao Liu, and Xiaoming Zhai. 2024. Applying large language models and chain-of-thought for automatic scoring. *Computers and Education: Artificial Intelligence* 6 (2024), 100213.

[98] Jin Joo Lee, Brad Knox, Jolie Baumann, Cynthia Breazeal, and David DeSteno. 2013. Computationally modeling interpersonal trust. *Frontiers in psychology* 4 (2013), 56004.

[99] Skye Lee Pazuchanics. 2006. The effects of camera perspective and field of view on performance in teleoperated navigation. In *Proceedings of the human factors and ergonomics society annual meeting*, Vol. 50. SAGE Publications Sage CA: Los Angeles, CA, 1528–1532.

[100] Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. 2020. BART: Denoising Sequence-to-Sequence Pre-training for Natural Language Generation, Translation, and Comprehension. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*. 7871–7880.

[101] Yuan Li, Yue Huang, Yuli Lin, Siyuan Wu, Yao Wan, and Lichao Sun. 2024. I Think, Therefore I am: Awareness in Large Language Models. *arXiv preprint arXiv:2401.17882* (2024).

[102] Timothy P Lillicrap, Jonathan J Hunt, Alexander Pritzel, Nicolas Heess, Tom Erez, Yuval Tassa, David Silver, and Daan Wierstra. 2015. Continuous control with deep reinforcement learning. *arXiv preprint arXiv:1509.02971* (2015).

[103] R Linnarsson. 1993. Decision support for drug prescription integrated with computer-based patient records in primary care. *Medical Informatics* 18, 2 (1993), 131–142.

[104] Siru Liu, Aileen P Wright, Barron L Patterson, Jonathan P Wanderer, Robert W Turer, Scott D Nelson, Allison B McCoy, Dean F Sittig, and Adam Wright. 2023. Using AI-generated suggestions from ChatGPT to optimize clinical decision support. *Journal of the American Medical Informatics Association* 30, 7 (2023), 1237–1245.

[105] Ilya Loshchilov and Frank Hutter. 2018. Decoupled Weight Decay Regularization. In *International Conference on Learning Representations*.

[106] Russell Lowell, David Saucier, Harish Chander, Reuben Burch, and Zachary Gillen. 2024. Effects of an Auditory Versus Visual Stimulus on Reaction and Response Time During Countermovement Jumps. *Perceptual and motor skills* 131, 4 (2024), 1080–1096.

[107] Terence J Lyons, William Ercoline, Kevin O'Toole, and Kevin Grayson. 2006. Aircraft and related factors in crashes involving spatial disorientation: 15 years of US Air Force data. *Aviation, space, and environmental medicine* 77, 7 (2006), 720–723.

[108] Qing Lyu, Shreya Havaldar, Adam Stein, Li Zhang, Delip Rao, Eric Wong, Marianna Apidianaki, and Chris Callison-Burch. 2023. Faithful chain-of-thought reasoning. *arXiv preprint arXiv:2301.13379* (2023).

[109] Sheikh Mannan and Nikhil Krishnaswamy. 2022. Where Am I and Where Should I Go? Grounding Positional and Directional Labels in a Disoriented Human Balancing Task. In *Proceedings of the 2022 CLASP Conference on (Dis)embodiment*, Simon Dobnik, Julian Grove, and Asad Sayeed (Eds.). Association for Computational Linguistics, Gothenburg, Sweden, 70–79. https://aclanthology.org/2022.clasp-1.8

[110] Sheikh Mannan and Nikhil Krishnaswamy. 2025. Bidirectional human-AI learning in real-time disoriented balancing. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 39. 29667–29669.

[111] Sheikh Mannan, Vivekanand Pandey Vimal, Paul DiZio, and Nikhil Krishnaswamy. 2024. Embodying Human-Like Modes of Balance Control Through Human-In-the-Loop Dyadic Learning. In *Proceedings of the AAAI Symposium Series*, Vol. 3. 565–569.

[112] Sheikh Abdul Mannan, Paige Hansen, Vivekanand Pandey Vimal, Hannah N Davies, Paul DiZio, and Nikhil Krishnaswamy. 2024. Combating spatial disorientation in a dynamic self-stabilization task using AI assistants. In *Proceedings of the 12th International Conference on Human-Agent Interaction*. 113–122.

[113] Geoffrey W Mccarthy and Roberta L Rollin Stott. 1994. In flight verification of the inversion illusion. *Aviation, space, and environmental medicine* 65, 4 (1994), 341–344.

[114] Donald McLean. 2003. Automatic flight control systems. *Measurement and Control* 36, 6 (2003), 172–175.

[115] Robert K Meeks, John Anderson, and Patrick M Bell. 2023. *Physiology of Spatial Orientation*. StatPearls Publishing, Treasure Island (FL). https://www.ncbi.nlm.nih.gov/books/NBK518976/ Updated 2023 Aug 14. In: StatPearls [Internet]..

[116] Siddharth Mehrotra, Ujwal Gadiraju, Eva Bittner, Folkert Van Delden, Catholijn M. Jonker, and Myrthe L. Tielman. 2025. "Even explanations will not help in trusting [this] fundamentally biased system": A Predictive Policing Case-Study. In *Proceedings of the 33rd ACM Conference on User Modeling, Adaptation and Personalization*. 51–62.

[117] Ulla Metzger and Raja Parasuraman. 2005. Automation in future air traffic management: Effects of decision aid reliability on controller performance and mental workload. *Human factors* 47, 1 (2005), 35–49.

[118] David Mitchell, Alfredo Arencibia, and Susana Munoz. 2004. Real-time detection of pilot-induced oscillations. In *AIAA Atmospheric Flight Mechanics Conference and Exhibit*. 4700.

[119] Seungwhan Moon, Satwik Kottur, Paul A Crook, Ankita De, Shivani Poddar, Theodore Levin, David Whitney, Daniel Difranco, Ahmad Beirami, Eunjoon Cho, et al. 2020. Situated and Interactive Multimodal Conversations. In *Proceedings of the 28th International Conference on Computational Linguistics*. 1103–1121.

[120] Michael S Scott Morton. 1971. *Management decision systems: computer-based support for decision making*. Division of Research Graduate School of Business Administrat.

[121] Eduardo Mosqueira-Rey, Elena Hernández-Pereira, David Alonso-Ríos, José Bobes-Bascarán, and Ángel Fernández-Leal. 2023. Human-in-the-loop machine learning: a state of the art. *Artificial Intelligence Review* 56, 4 (2023), 3005–3054.

[122] Bonnie M Muir. 1994. Trust in automation: Part I. Theoretical issues in the study of trust and human intervention in automated systems. *Ergonomics* 37, 11 (1994), 1905–1922.

[123] Charles Arthur Nagler and William Merle Nagler. 1973. Reaction time measurements. *Forensic science* 2 (1973), 261–274.

[124] David G Newman. 2007. *An overview of spatial disorientation as a factor in aviation accidents and incidents*. Number B2007/0063. Australian Transport Safety Bureau Canberra City.

[125] Stefanos Nikolaidis, Yu Xiang Zhu, David Hsu, and Siddhartha Srinivasa. 2017. Human-robot mutual adaptation in shared autonomy. In *Proceedings of the 2017 ACM/IEEE International Conference on Human-Robot Interaction*. 294–302.

[126] Gesang Nugroho, Andi Dharmawan, Danang Lelono, and Ariesta Martiningtyas Handayani. 2021. Waypoint Tracking of a Fixed-Wing UAV Using the L1 Cross Track Error Control. *ICIC Express Letters* 13 (2021), 115–22.

[127] Rafael E Nunez. 1999. Could the future taste purple? Reclaiming mind, body and cognition. *Journal of consciousness studies* 6, 11-12 (1999), 41–60.

[128] Donald Nute, Walter D Potter, Mayukh Dass, Astrid Glende, Frederick Maier, Hajime Uchiyama, Jin Wang, Mark Twery, Peter Knopp, Scott Thomasma, et al. 2003. An agent architecture for an integrated forest ecosystem management decision support system. *Decision Support for Multiple Purpose Forestry, April 23-25, Vienna, Austria, p. 1-11* (2003).

[129] Jennifer Ockerman and Amy Pritchett. 2000. A review and reappraisal of task guidance: Aiding workers in procedure following. *International Journal of Cognitive Ergonomics* 4, 3 (2000), 191–212.

[130] On-Road Automated Driving (ORAD) Committee. 2021. *Taxonomy and Definitions for Terms Related to Driving Automation Systems for On-Road Motor Vehicles*. https://doi.org/10.4271/J3016_202104

[131] Gari Palmer, Anne Selwyn, and Dan Zwillinger. 2016. *The "Trust V": Building and Measuring Trust in Autonomous Systems*. Springer US, Boston, MA, 55–77.

[132] Alexander Sacha Panic, Heather Panic, Paul DiZio, and James R Lackner. 2017. Gravitational and somatosensory influences on control and perception of roll balance. *Aerospace medicine and human performance* 88, 11 (2017), 993–999.

[133] Heather Panic, Alexander Sacha Panic, Paul DiZio, and James R Lackner. 2015. Direction of balance and perception of the upright are perceptually dissociable. *Journal of neurophysiology* 113, 10 (2015), 3600–3609.

[134] Raja Parasuraman, Michael Barnes, Keryl Cosenzo, and Sandeep Mulgund. 2007. Adaptive automation for human-robot teaming in future command and control systems. (2007).

[135] Raja Parasuraman and Victor Riley. 1997. Humans and automation: Use, misuse, disuse, abuse. *Human factors* 39, 2 (1997), 230–253.

[136] Raja Parasuraman, Thomas B Sheridan, and Christopher D Wickens. 2000. A model for types and levels of human interaction with automation. *IEEE Transactions on systems, man, and cybernetics-Part A: Systems and Humans* 30, 3 (2000), 286–297.

[137] Rolf Pfeifer and Christian Scheier. 2001. *Understanding intelligence*. MIT Press.

[138] Gloria Phillips-Wren. 2013. Intelligent decision support systems. *Multicriteria decision aid and artificial intelligence: links, theory and applications* (2013), 25–44.

[139] G Porenta, B Pfahringer, M Hoberstorfer, and R Trappl. 2019. A decision support system for village health workers in developing countries. In *Expert Systems In Developing Countries*. CRC Press, 193–207.

[140] Jinyao Qu, Ming Lv, Yufei Yang, and Yihao Tang. 2021. Flight motion recognition method based on multivariate phase space reconstruction and approximate entropy. In *2021 40th Chinese control conference (CCC)*. IEEE, 7247–7253.

[141] Tom Quick and Kerstin Dautenhahn. 1999. Making embodiment measurable. *Proceedings of '4. Fachtagung der Gesellschaft für Kognitionswissenschaft'. Bielefeld, Germany. http://supergoodtech. com/tomquick/phd/kogwis/webtext. html* (1999).

[142] Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. 2019. Language models are unsupervised multitask learners. *OpenAI blog* 1, 8 (2019), 9.

[143] Antonin Raffin, Ashley Hill, Adam Gleave, Anssi Kanervisto, Maximilian Ernestus, and Noah Dormann. 2021. Stable-baselines3: Reliable reinforcement learning implementations. *The Journal of Machine Learning Research* 22, 1 (2021), 12348–12355.

[144] Gary E Riccio, Eric J Martin, and Thomas A Stoffregen. 1992. The role of balance dynamics in the active perception of orientation. *Journal of Experimental Psychology: Human Perception and Performance* 18, 3 (1992), 624.

[145] Gert Rickheit and Ipke Wachsmuth. 2006. *Situated communication*. Mouton de Gruyter.

[146] Paul Robinette, Wenchen Li, Robert Allen, Ayanna M. Howard, and Alan R. Wagner. 2016. Overtrust of robots in emergency evacuation scenarios. In *2016 11th ACM/IEEE International Conference on Human-Robot Interaction (HRI)*. 101–108. https://doi.org/10.1109/HRI.2016.7451740

[147] John M Rolfe and Ken J Staples. 1988. *Flight simulation*. Number 1. Cambridge University Press.

[148] Stéphane Ross and Drew Bagnell. 2010. Efficient reductions for imitation learning. In *Proceedings of the thirteenth international conference on artificial intelligence and statistics*. JMLR Workshop and Conference Proceedings, 661–668.

[149] Garrett Sadler, Henri Battiste, Nhut Ho, Lauren Hoffmann, Walter Johnson, Robert Shively, Joseph Lyons, and David Smith. 2016. Effects of transparency on pilot trust and agreement in the autonomous constrained flight planner. In *2016 IEEE/AIAA 35th Digital Avionics Systems Conference (DASC)*. IEEE, 1–9.

[150] Patrick Salamin, Daniel Thalmann, and Frédéric Vexo. 2006. The benefits of third-person perspective in virtual and augmented reality?. In *Proceedings of the ACM symposium on Virtual reality software and technology*. 27–30.

[151] Maha Salem, Gabriella Lakatos, Farshid Amirabdollahian, and Kerstin Dautenhahn. 2015. Would you trust a (faulty) robot? Effects of error, task type and personality on human-robot cooperation and trust. In *Proceedings of the tenth annual ACM/IEEE international conference on human-robot interaction*. 141–148.

[152] Manolis Savva, Abhishek Kadian, Oleksandr Maksymets, Yili Zhao, Erik Wijmans, Bhavana Jain, Julian Straub, Jia Liu, Vladlen Koltun, Jitendra Malik, et al. 2019. Habitat: A platform for embodied ai research. In *Proceedings of the IEEE/CVF international conference on computer vision*. 9339–9347.

[153] Marc R. Schlichting, Vale Rasmussen, Heba Alazzeh, Houjun Liu, Kiana Jafari, Amelia F. Hardy, Dylan M. Asmar, and Mykel J. Kochenderfer. 2025. LeRAAT: LLM-Enabled Real-Time Aviation Advisory Tool. *ArXiv* abs/2503.16477 (2025). https://api.semanticscholar.org/CorpusID:277244474

[154] Aran Sena and Matthew Howard. 2020. Quantifying teaching behavior in robot learning from demonstration. *The International Journal of Robotics Research* 39, 1 (2020), 54–72.

[155] Anuragini Shirish Shalini Chandra and Shirish C. Srivastava. 2022. To Be or Not to Be . . . Human? Theorizing the Role of Human-Like Competencies in Conversational Artificial Intelligence Agents. *Journal of Management Information Systems* 39, 4 (2022), 969–1005. https://doi.org/10.1080/07421222.2022.2127441 arXiv:https://doi.org/10.1080/07421222.2022.2127441

[156] Lanbo She and Joyce Chai. 2017. Interactive learning of grounded verb semantics towards human-robot communication. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. 1634–1644.

[157] Lanbo She, Shaohua Yang, Yu Cheng, Yunyi Jia, Joyce Chai, and Ning Xi. 2014. Back to the blocks world: Learning new actions through situated human-robot dialogue. In *Proceedings of the 15th annual meeting of the special interest group on discourse and dialogue (SIGDIAL)*. 89–97.

[158] Mark Shelhamer. 2015. Trends in sensorimotor research and countermeasures for exploration-class space flights. *Frontiers in Systems Neuroscience* 9 (2015), 115.

[159] Ben Shneiderman. 2020. Human-Centered Artificial Intelligence: Reliable, Safe & Trustworthy. *CoRR* abs/2002.04087 (2020). arXiv:2002.04087 https://arxiv.org/abs/2002.04087

[160] Vladimír Socha, Luboš Socha, Stanislav Szabo, Karel Hana, J Gazda, M Kimlickova, Iveta Vajdova, A Madoran, Lenka Hanakova, V Nemec, et al. 2016. Training of pilots using

flight simulator and its impact on piloting precision. *Transport Means. Juodkrante: Kansas University of Technology* (2016), 374–379.

[161] Megan Su and Yuwei Bao. 2024. User modeling challenges in interactive AI assistant systems. *arXiv preprint arXiv:2403.20134* (2024).

[162] Jun Jet Tai, Jim Wong, Mauro Innocente, Nadjim Horri, James Brusey, and Swee King Phang. 2023. PyFlyt–UAV Simulation Environments for Reinforcement Learning Research. *arXiv preprint arXiv:2304.01305* (2023).

[163] Yuko Takada, Tetsuya Hisada, Naruo Kuwada, Masao Sakai, and Tomomitsu Akamatsu. 2009. Survey of severe spatial disorientation episodes in Japan Air Self-Defense Force fighter pilots showing increased severity in night flight. *Military medicine* 174, 6 (2009), 626–630.

[164] Kartik Talamadupula, J Benton, Subbarao Kambhampati, Paul Schermerhorn, and Matthias Scheutz. 2010. Planning for human-robot teaming in open worlds. *ACM Transactions on Intelligent Systems and Technology (TIST)* 1, 2 (2010), 1–24.

[165] Gemini Team, Rohan Anil, Sebastian Borgeaud, Jean-Baptiste Alayrac, Jiahui Yu, Radu Soricut, Johan Schalkwyk, Andrew M Dai, Anja Hauth, Katie Millican, et al. 2023. Gemini: a family of highly capable multimodal models. *arXiv preprint arXiv:2312.11805* (2023).

[166] Evan Thompson and Francisco J Varela. 2001. Radical embodiment: neural dynamics and consciousness. *Trends in cognitive sciences* 5, 10 (2001), 418–425.

[167] Kathryn Tomzcak, Adam Pelter, Corey Gutierrez, Thomas Stretch, Daniel Hilf, Bianca Donadio, Nathan L. Tenhundfeld, Ewart J. de Visser, and Chad C. Tossell. 2019. Let Tesla Park Your Tesla: Driver Trust in a Semi-Automated Car. In *2019 Systems and Information Engineering Design Symposium (SIEDS)*. 1–6. https://doi.org/10.1109/SIEDS.2019.8735647

[168] Mark Towers, Jordan K. Terry, Ariel Kwiatkowski, John U. Balis, Gianluca de Cola, Tristan Deleu, Manuel Goulão, Andreas Kallinteris, Arjun KG, Markus Krimmel, Rodrigo Perez-Vicente, Andrea Pierré, Sander Schulhoff, Jun Jet Tai, Andrew Tan Jin Shen, and Omar G. Younis. 2023. Gymnasium. https://doi.org/10.5281/zenodo.8127026

[169] Salih Tutun, Marina E Johnson, Abdulaziz Ahmed, Abdullah Albizri, Sedat Irgil, Ilker Yesilkaya, Esma Nur Ucar, Tanalp Sengun, and Antoine Harfouche. 2023. An AI-based decision support system for predicting mental health disorders. *Information Systems Frontiers* 25, 3 (2023), 1261–1276.

[170] Francisco J Varela, Evan Thompson, and Eleanor Rosch. 2017. *The embodied mind, revised edition: Cognitive science and human experience*. MIT press.

[171] David Vernon, Robert Lowe, Serge Thill, and Tom Ziemke. 2015. Embodied cognition and circular causality: on the role of constitutive autonomy in the reciprocal coupling of perception and action. *Frontiers in Psychology* 6 (2015). https://doi.org/10.3389/fpsyg.2015.01660

[172] Vivekanand Pandey Vimal. 2017. *The Role of Gravitational Cues in the Learning of Balance Control*. Brandeis University.

[173] Vivekanand Pandey Vimal, Paul DiZio, and James R Lackner. [n. d.]. The role of spatial acuity in a dynamic balancing task without gravitational cues. *Experimental brain research* ([n. d.]), 1–11.

[174] Vivekanand Pandey Vimal, Paul DiZio, and James R Lackner. 2017. Learning dynamic balancing in the roll plane with and without gravitational cues. *Experimental brain research* 235, 11 (2017), 3495–3503.

[175] Vivekanand Pandey Vimal, Paul DiZio, and James R Lackner. 2019. Learning and long-term retention of dynamic self-stabilization skills. *Experimental brain research* 237, 11 (2019), 2775–2787.

[176] Vivekanand Pandey Vimal, Paul DiZio, and James R Lackner. 2022. The role of spatial acuity in a dynamic balancing task without gravitational cues. *Experimental brain research* 240, 1 (2022), 123–133.

[177] Vivekanand Pandey Vimal, James R Lackner, and Paul DiZio. 2016. Learning dynamic control of body roll orientation. *Experimental brain research* 234, 2 (2016), 483–492.

[178] Vivekanand Pandey Vimal, James R Lackner, and Paul DiZio. 2018. Learning dynamic control of body yaw orientation. *Experimental brain research* 236, 5 (2018), 1321–1330.

[179] Vivekanand Pandey Vimal, Alexander Sacha Panic, James R Lackner, and Paul DiZio. 2023. Vibrotactile feedback as a countermeasure for spatial disorientation. *Frontiers in physiology* 14 (2023), 1249962.

[180] Vivekanand Pandey Vimal, Han Zheng, Pengyu Hong, Lila N Fakharzadeh, James R Lackner, and Paul DiZio. 2020. Characterizing individual differences in a dynamic stabilization task using machine learning. *Aerospace medicine and human performance* 91, 6 (2020), 479–488.

[181] Vittorio Di Vito, Patryk Tadeusz Grzybowski, Tomasz Rogalski, and Piotr Masłowski. 2021. A concept for an Integrated Mission Management System for Small Air Transport vehicles in the COAST project. *IOP Conference Series: Materials Science and Engineering* 1024 (2021). https://api.semanticscholar.org/CorpusID:234160404

[182] Kailas Vodrahalli, Roxana Daneshjou, Tobias Gerstenberg, and James Zou. 2022. Do humans trust advice more if it comes from ai? an analysis of human-ai interactions. In *Proceedings of the 2022 AAAI/ACM Conference on AI, Ethics, and Society*. 763–777.

[183] Jiashuo Wang, Chunpu Xu, Chak Tou Leong, Wenjie Li, and Jing Li. 2024. Mitigating unhelpfulness in emotional support conversations with multifaceted AI feedback. *arXiv preprint arXiv:2401.05928* (2024).

[184] Yonglin Wang, Jie Tang, Vivekanand Pandey Vimal, James R Lackner, Paul DiZio, and Pengyu Hong. 2022. Crash prediction using deep learning in a disorienting spaceflight analog balancing task. *Frontiers in physiology* (2022), 51.

[185] Martin Weber. 1987. Decision making with incomplete information. *European journal of operational research* 28, 1 (1987), 44–57.

[186] Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, et al. 2022. Chain-of-thought prompting elicits reasoning in large language models. *Advances in neural information processing systems* 35 (2022), 24824–24837.

[187] Joseph Weizenbaum. 1966. ELIZA—a computer program for the study of natural language communication between man and machine. *Commun. ACM* 9, 1 (Jan. 1966), 36–45. https://doi.org/10.1145/365153.365168

[188] Gregor Wiedemann, Steffen Remus, Avi Chawla, and Chris Biemann. 2019. Does BERT Make Any Sense? Interpretable Word Sense Disambiguation with Contextualized Embeddings. https://doi.org/10.48550/ARXIV.1909.10430

[189] Wikipedia contributors. 2025. Link Trainer — Wikipedia, The Free Encyclopedia. https://en.wikipedia.org/w/index.php?title=Link_Trainer&oldid=1305882774. [Online; accessed 26-August-2025].

[190] Wikipedia contributors. 2025. Microsoft Flight Simulator — Wikipedia, The Free Encyclopedia. https://en.wikipedia.org/w/index.php?title=Microsoft_Flight_Simulator&oldid=1307599397. [Online; accessed 26-August-2025].

[191] Michael Wooldridge. 1999. Intelligent agents. *Multiagent systems: A modern approach to distributed artificial intelligence* 1 (1999), 27–73.

[192] Fanbo Xiang, Yuzhe Qin, Kaichun Mo, Yikuan Xia, Hao Zhu, Fangchen Liu, Minghua Liu, Hanxiao Jiang, Yifu Yuan, He Wang, et al. 2020. Sapien: A simulated part-based interactive

environment. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 11097–11107.

[193] Shuiqiao Yang, Kun Yu, Thorsten Lammers, and Fang Chen. 2021. Artificial intelligence in pilot training and education–towards a machine learning aided instructor assistant for flight simulators. In *International conference on human-computer interaction*. Springer, 581–587.

[194] Junjie Ye, Xuanting Chen, Nuo Xu, Can Zu, Zekai Shao, Shichun Liu, Yuhan Cui, Zeyang Zhou, Chao Gong, Yang Shen, Jie Zhou, Siming Chen, Tao Gui, Qi Zhang, and Xuanjing Huang. 2023. A Comprehensive Capability Analysis of GPT-3 and GPT-3.5 Series Models. arXiv:2303.10420 [cs.CL] https://arxiv.org/abs/2303.10420

[195] Wayne W Zachary. 1988. Decision support systems: Designing to extend the cognitive limits. In *Handbook of human-computer interaction*. Elsevier, 997–1030.

[196] Matthew D Zeiler. 2012. Adadelta: an adaptive learning rate method. *arXiv preprint arXiv:1212.5701* (2012).

[197] Irina Zgonnikova, Arkady Zgonnikov, and Shigeru Kanemoto. 2016. Stick must fall: Using machine learning to predict human error in virtual balancing task. In *2016 IEEE 16th International Conference on Data Mining Workshops (ICDMW)*. IEEE, 173–177.

[198] Michelle Zhao, Reid Simmons, and Henny Admoni. 2025. The role of adaptation in collective human–AI teaming. *Topics in cognitive science* 17, 2 (2025), 291–323.

[199] Haoyi Zhou, Shanghang Zhang, Jieqi Peng, Shuai Zhang, Jianxin Li, Hui Xiong, and Wancai Zhang. 2021. Informer: Beyond Efficient Transformer for Long Sequence Time-Series Forecasting. arXiv:2012.07436 [cs.LG]

[200] Tom Ziemke. 2001. Are robots embodied. Citeseer.

[201] Tom Ziemke. 2003. What's that Thing Called Embodiment? *Proceedings of the 25th Annual meeting of the Cognitive Science Society* 6 (01 2003).

[202] Tom Ziemke. 2004. *Embodied AI as Science: Models of Embodied Cognition, Embodied Models of Cognition, or Both?* Springer Berlin Heidelberg, Berlin, Heidelberg, 27–36. https://doi.org/10.1007/978-3-540-27833-7_2

Appendix A

List of Notations

Table A.1: List of Abbreviations and Symbols

Abbreviation	Full Name / Description
AIRL	Adversarial Inverse Reinforcement Learning
ARPA-H	Advanced Research Projects Agency for Health
ATTOL	Autonomous Taxi, Take-Off and Landing
BC	Behavior Cloning
BERT	Bidirectional Encoder Representations from Transformers
DARPA	Defense Advanced Research Projects Agency
DDPG	Deep Deterministic Policy Gradient
DOB	Direction of Balance
EDC	Embodied Direction Classifier
HAI	Human-AI Interaction
HCAI	Human-Centered Artificial Intelligence
HITL	Human-in-the-Loop
IP	Inverted Pendulum
MARS	Multi-Axis Rotation System
MCAS	Maneuvering Characteristics Augmentation System
PARADIGM	Platform Accelerating Rural Access to Distributed and Integrated Medical Care
PIO	Pilot Induced Oscillations
PPO	Proximal Policy Optimization
PTG	Perceptually-enabled Task Guidance

Continued on next page...

Abbreviation	**Full Name / Description** (Cont.)
RDK	Random Dot Kinematogram
RL	Reinforcement Learning
SAC	Soft Actor-Critic
SD	Spatial Disorientation
SL	Supervised Learning
VIP	Virtual Inverted Pendulum
VIGIL	Vectors of Intelligence Guidance in Long-Reach Rural Healthcare

Symbol	**Description**
$\mathcal{C}_{ent}$	Control Entropy
$\mathcal{C}_{ext}$	Control Extremity
G_{max}	Max G-Force
$\mu\|Mag\|_{vel}$	Mean Magnitude of Angular Velocity
$\mu\|\theta\|$	Mean Angular Position
$\mathcal{R}_{inv}$	Inversion Ratio
r_{rb}	Rank Biserial Correlation
σ_{CTE}	Cross-Track Error
$\sigma(\theta)$	Standard Deviation in Angular Position
vel RMS	Root-Mean-Square of Angular Velocity

Appendix B

Appendix for Chapter 4: AI Guidance to Combat Spatial Disorientation

B.1 Model Size Comparison

Table B.1 shows the inputs and outputs and number of parameters over each class of model trained. Specific training data (proficiency class, MARS vs. VIP data) of course makes no difference to the size of the final model.

Type	Input	Prediction	# params
SAC	Current step	Next step	268.0K*
DDPG	Current step	Next step	244.8K*
MLP	Current step	Next step	20.7.0K
	Past 0.2s	Next step	23.7.0K
	Past 0.3s	0.1s future	25.2.0K
	Past 0.5s	Next step	28.2.0K
RNN	Past 0.2s	Next step	101.0K
	Past 0.3s	0.1s future	101.0K
	Past 0.5s	Next step	101.0K
LSTM	Past 0.2s	Next step	314.0K
	Past 0.3s	0.1s future	314.0K
	Past 0.5s	Next step	314.0K
GRU	Past 0.2s	Next step	243.0K
	Past 0.3s	0.1s future	243.0K
	Past 0.5s	Next step	243.0K

Table B.1: Model statistics.

*Sum over all actor and critic networks.

B.2 SL Training Details

The MLP networks comprised 3 $tanh$-activated layers of 100 units. The RNN, LSTM, and GRU networks used 4 stacked recurrent layers of the respective types, each comprising 100 units, followed by 3 100-unit $tanh$-activated linear layers with a dropout probability of 0.1. All of these models were trained with a learning rate of $1e-3$, batch size of 10K, Huber loss [80] with the default δ of 1.0, Adam optimization [90], and max epochs of 1,000 with early stopping (patience of 50 epochs on the validation MAE criterion) on an NVIDIA RTX A6000 48GB. Position/velocity inputs to SL models were in radians or radians per second and converted to degrees for data collection (with the exception of Wang et al. [184]'s crash predictor, where inputs were already in degrees). SL model outputs were then plotted over known test samples to assess if they were sufficient predictors of human actions, also using MAE as the criterion.

B.3 RL Training Details

The vanilla (default) DDPG and SAC models using our custom reward function are trained for 10K timesteps, τ of $5e-3$, γ 0.99, an MSBE loss with stochastic gradient descent and a learning rate of $1e-3$ and $3e-4$ for DDPG and SAC respectively. For all other optional parameters, the default initialization values were used, including an MLP policy consisting of 2 hidden layers of 256 units each with ReLU activation (SAC) and 2 hidden layers of 400 and 300 units with ReLU activation (DDPG). Like the SL models, RL model inputs were in radians/radians per second and converted for data collection.

For the behavior cloning variants of the models, I use the same initialization of SAC and DDPG as above, with different weights. An expert dataset was created using the observations (angular position in cartesian coordinates & angular velocity) and actions (joystick deflections) of four participants who were categorized as Good in the MARS task. The expert dataset is passed to the behavior learning algorithm (based on code from Stable-Baselines3) which is trained for 100 epochs, batch size of 64, an Adadelta optimizer with a learning rate of 1.0 [196], a learning rate scheduler with a γ of 0.7 and a step size of 1.

The AIRL variant of SAC was initialized with an OpenAI Gym [28] environment instead of Gymnasium since the `imitation` library I used for this [66] currently does not support the Gymnasium package. The expert dataset uses the same four participants as were used for behavior cloning but includes a resulting state (the next observation) and a termination state (whether a crash occurred). SAC AIRL was trained for 300K timesteps and variable horizon trajectories were allowed. I train only SAC with AIRL as AIRL requires stochastic policies. Behavior Cloning manually seeds PyTorch with a random seed of 1. Any other random seed values are defaults used by the packages in use. All models were trained on an NVIDIA RTX A6000 48GB.

B.4 Informer Model Training and Performance

I also trained from scratch an instance of the Informer model, a time-series transformer architecture by Zhou et al. [199] on the Good data subset. Past angular positions and velocities are used as features to predict future joystick deflections. For the Informer model, I trained using a prediction length of 0.1s, a context length of 0.2s, 4 encoder and decoder layers, and a 32-dimensional transformer layer. It was trained with a learning rate of $6e-4$, AdamW optimization [105], and batch size of 64 for 500 epochs on an NVIDIA A100 80GB. The Informer outputs 100 0.1s forecasts of which I took the mean.

The failure of the Informer model at performing the IP task precluded it from being used as a pilot or assistant for the full set of experiments. To further investigate this, I paired the trained Informer as a pilot with an assistant to observe performance patterns. I observed a very strong recency bias in the Informer's output, such that when performing the task alone, it would start—and continue—to make very small deflections, near 0, even when falling away from the DOB. For instance, when paired with the plain DDPG assistant, after the Informer accepted a suggestion, subsequent unassisted deflections would remain close to the magnitude of the previous suggestion, eventually reverting to the mean (see yellow highlights in Fig. B.1). This behavior, coupled with the size of the trained Informer ($\sim$3.4 billion parameters), the required training time ($\sim$3.75 hrs.), and the inference time (resulting in $\sim$60% fewer samples per 30-sec. trial), indicate that Transformer-

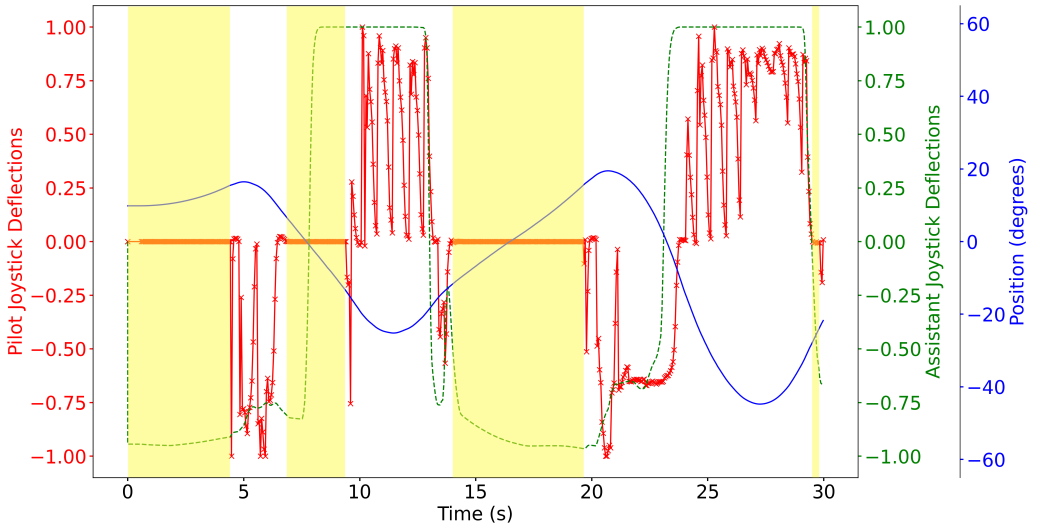


Figure B.1: Sample Informer (pilot) and DDPG (assistant) trial.

based models may not be particularly useful in this task, or require more sophisticated, time- and compute-intensive techniques, such as RLHF, to achieve parity with other models.

B.5 PyVIP Details

Upon crashes, PyVIP resets the VIP to a random angle given by $(r+1)*sgn(r-0.5)*ipoff*0.5$ where r is a random number sampled from a uniform distribution over $[0, 1)$ and $ipoff$ is the maximum allowable offset of the pendulum reset position with respect to the DOB, specified as a fraction of the fall limit. I use a value of 0.25 for $ipoff$, corresponding to a maximum allowable offset of $\pm 15°$. I use a random seed of 42. Raw data from PyVIP evaluation trials was saved in degrees. The PyVIP framework was developed under guidance from experts at the Ashton Graybeil Spatial Orientation Lab at Brandeis University.

B.6 Full Results Comparison

Table B.2 shows performance differences between the pilots when unaided and when aided by all 26 assistants (cf. Table 4.2 in the main body). Beyond the trends observed in the main paper,

Assistant	**Crashes**↓	**% destab.**↓	$\mu\|\theta\|$ (°)↓	$\sigma(\theta)$ (°)↓	$\mu\|Mag\|_{vel}$ (°/s)↓	$\sigma(\|Mag\|_{vel})$ (°/s)↓	vel **RMS**↓	$\mu\|d\|$↓
SAC	0 / -5 / -25 -8 / -25 / -12	2.2 / 4.9 / -18.3 -31.8 / -38.3 / -28.5	0.4 / 1.2 / -3.6 -2.0 / -7.6 -1.0	-0.9 / -6.6 / -7.9 -1.9 / -6.7 / 7.9	18.0 / -25.3 / -56.9 27.0 / -37.7 / 7.5	18.4 / -36.6 / -54.3 27.8 / -48.6 / 11.9	18.7 / -38.7 / -67.5 24.4 / -42.2 / -6.9	0.2 / 0.1 / 0.0 0.4 / -0.1 / 0.3
SAC-BC	1 / -1 / -22 0 / -20 / -3	0.8 / -2.0 / -22.6 -27.5 / -34.1 / -26.3	2.1 / 0.1 / 7.8 -2.3 / -7.6 / -1.3	1.0 / 1.1 / 6.1 -1.7 / -5.6 / 8.8	14.1 / -5.4 / -52.6 24.6 / -38.5 / 16.3	16.7 / -8.8 / -46.5 35.3 / -47.0 / 27.5	17.0 / -9.6 / -58.9 31.4 / -42.4 / 9.5	0.1 / -0.0 / -0.0 0.2 / -0.3 / 0.1
SAC-AIRL	-2 / -7 / -17 -15 / -33 / -12	1.9 / -3.8 / -14.4 -36.8 / -41.5 / -28.5	1.9 / -0.1 / 5.4 -0.5 / -6.1 / 0.9	0.9 / -7.7 / -0.3 -6.8 / -10.9 / 9.1	-9.8 / -38.7 / -62.8 -22.5 / -66.6 / -8.7	-11.0 / -49.8 / -61.0 -26.1 / -71.7 / -19	-11.2 / -53.5 / -71.2 -29.9 / -72.9 / -20.3	0.1 / -0.1 / -0.0 0.2 / -0.2 / 0.2
DDPG	1 / -2 / -21 -11 / -23 / -12	2.3 / 3.7 / -17.3 -36.4 / -36.4 / -25.1	2.4 / -2.4 / -2.2 2.6 / -7.5 / -1.0	2.1 / -3.1 / -3.4 3.5 / -3.6 / 9.2	25.3 / -11.0 / -43.1 47.4 / -34.1 / 5.5	22.4 / -19.7 / -38.2 47.3 / -36.6 / 7.24	24.6 / -21.2 / -51.0 43.5 / -38.6 / -12.5	0.2 / 0.1 / 0.1 0.6 / -0.1 / 0.3
DDPG-BC	1 / 2 / -19 -6 / -20 / -8	0.8 / -0.5 / -19.1 -31.3 / -30.4 / -27.9	4.9 / 4.2 / 1.5 -2.7 / -8.6 / 4.4	4.5 / 5.3 / -9.8 -2.0 / -7.5 / 13.6	18.1 / 7.4 / -56.5 4.5. / -34.4 / 11.6	17.3 / 1.3 / -51.5 3.4 / -37.0 / 16.3	18.1 / -0.8 / -61.4 2.9 / -37.1 / -1.7	0.1 / 0.0 / -0.1 0.1 / -0.2 / 0.1
MLP-G-0	-2 / 6 / -12 5 / -10 / -9	1.6 / 13.6 / -8.1 -14.6 / -20.8 / -25.0	4.3 / 2.2 / 0.5 1.1 / -3.1 / 4.5	2.8 / -0.4 / 0.1 1.9 / 0.1 / 13.9	12.3 / -4.3 / -43.5 35.8 / -3.6 / 11.2	11.9 / -19.0 / -36.7 50.3 / -9.9 / 22.2	10.4 / -8.0 / -48.0 46.5 / -5.0 / 2.6	0.2 / -0.0 / -0.0 0.3 / -0.1 / 0.2
MLP-G-0.5	6 / 8 / -8 8 / -20 / 8	11.2 / 14.2 / -1.6 -6.4 / -27.7 / -8.1	2.0 / 5.6 / 1.3 0.3 / -4.4 / -0.2	0.1 / 0.9 / -3.7 0.3 / -2.4 / 9.4	19.2 / -0.5 / -39.0 39.2 / -32.5 / 39.6	23.9 / -18.5 / -32.3 45.6 / -35.3 / 53.9	26.0 / -5.8 / -40.2 47.5 / -36.7 / 37.4	0.1 / -0.0 / -0.1 0.2 / -0.1 / 0.2
MLP-G-0.2	7 / 12 / -7 5 / -14 / 4	11.5 / 20.5 / -4.1 -10.7 / -22.3 / -6.1	2.9 / 1.9 / 0.4 0.8 / -6.5 / 1.1	2.2 / -1.1 / -0.1 1.6 / -3.8 / 9.3	26.3 / 3.0 / -30.0 34.5 / -16.8 / 22.7	33.9 / -17.5 / -22.6 45.2 / -29.1 / 35.3	32.5 / 2.6 / -33.3 46.3 / -15.6 / 18.3	0.1 / -0.0 / -0.0 0.3 / -0.1 / 0.2
MLP-G-0.3	3 / 14 / -7 3 / -14 / 3	3.8 / 25.2 / -7.0 -17.5 / -21.8 / -18.5	4.7 / 0.7 / 2.3 -0.7 / -6.7 / 2.0	4.1 / 0.8 / 1.6 0.5 / -3.2 / 12.0	27.9 / 14.7 / -41.4 33.5 / -16.0 / 33.2	31.1 / -18.1 / -39.2 41.7 / -24.9 / 45.2	32.9 / 12.1 / -48.6 40.1 / -17.9 / 25.8	0.2 / 0.0 / -0.1 0.3 / -0.1 / 0.2
MLP-GM-0	-1 / 3 / -12 1 / -18 / -1	2.0 / 8.3 / -10.4 -17.4 / -27.6 / -15.3	3.4 / 3.1 / -1.6 0.5 / -6.1 / 4.2	2.9 / 1.5 / -2.9 1.3 / -2.5 / 12.8	16.4 / 7.7 / -45.8 31.3 / -17.0 / 18.7	22.2 / -2.2 / -38.5 40.7 / -20.4 / 21.4	22.3 / 2.7 / -47.7 37.5 / -19.4 / 5.5	0.2 / 0.0 / -0.0 0.3 / -0.1 / 0.2
MLP-GMB-0	-2 / 4 / -11 -1 / -18 / -4	-0.5 / 7.0 / -8.8 -21.6 / -29.7 / -21.5	2.4 / 3.1 / 2.2 2.1 / -6.8 / 4.0	2.2 / 2.8 / -1.7 0.8 / -2.5 / 12.6	18.8 / 7.0 / -43.2 31.4 / -15.5 / 12.2	25.0 / -5.1 / -36.3 41.5 / -20.3 / 22.2	24.5 / 1.3 / -48.0 38.7 / -18.6 / 3.8	0.2 / 0.0 / -0.0 0.3 / -0.1 / 0.2
RNN-G-0.5	7 / 8 / -2 6 / -10 / 1	11.6 / 18.2 / -3.4 -6.4 / -18.4 / -9.6	3.3 / 4.2 / 1.7 -1.1 / -8.5 / -0.2	2.8 / 0.8 / 3.3 -0.8 / -4.3 / 8.7	24.5 / 2.8 / -22.0 30.1 / -2.5 / 26.0	30.5 / -13.6 / -11.5 45.9 / -5.9 / 37.6	29.6 / 1.1 / -23.6 46.6 / 0.8 / 25.0	0.1 / -0.0 / -0.0 0.2 / -0.1 / 0.2
RNN-G-0.2	5 / 11 / -4 6 / -15 / 0	7.5 / 18.6 / -3.4 -15.7 / -23.1 / -8.2	4.4 / 2.6 / -1.0 0.1 / -7.4 / 0.2	3.4 / 1.8 / -2.1 0.3 / -4.0 / 9.8	24.8 / 13.9 / -24.3 44.8 / -16.2 / 20.4	37.1 / -10.0 / -15.2 53.9 / -12.8 / 35.2	37.6 / 10.8 / -24.6 51.4 / -14.5 / 19.4	0.1 / 0.0 / -0.0 0.3 / -0.1 / 0.2
RNN-G-0.3	2 / 14 / -10 -2 / -18 / -5	4.0 / 19.6 / -10.7 -21.5 / -22.2 / -20.3	3.7 / 0.5 / 2.5 -0.1 / -6.5 / 0.5	.7 / -0.3 / 1.9 0.9 / -4.1 / 10.2	15.9 / 13.0 / -32.6 29.3 / -24.2 / 18.6	23.4 / -22.7 / -25.0 37.7 / -30.5 / 28.7	25.0 / 7.1 / -38.0 35.0 / -24.4 / 13.8	0.2 / -0.0 / -0.0 0.2 / -0.2 / 0.2
RNN-GM-0.5	10 / 14 / -8 6 / -19 / 0	13.2 / 22.9 / -6.7 -9.2 / -26.7 / -8.2	1.8 / 1.9 / 1.2 0.4 / -7.5 / 2.4	1.8 / -0.2 / 0.5 1.2 / -4.6 / 11.5	27.8 / 13.8 / -32.7 44.9 / -21.4 / 21.0	41.3 / -18.9 / -22.3 62.7 / -22.4 / 34.6	41.7 / 11.0 / -34.4 58.6 / -20.4 / 18.1	0.1 / 0.0 / -0.0 0.3 / -0.1 / 0.2
RNN-GMB-0.5	7 / 4 / -6 6 / -10 / 3	10.2 / 14.3 / -6.6 -7.3 / -13.9 / -9.5	4.7 / 3.0 / 1.7 0.6 / -7.0 / 3.6	3.8 / 3.4 / -1.3 0.1 / -4.4 / 13.4	29.3 / 3.6 / -26.8 33.6 / -4.2 / 34.8	41.5 / -4.8 / -23.0 48.7 / -12.8 / 50.6	41.4 / 3.6 / -28.6 47.5 / 0.6 / 31.5	0.2 / 0.0 / -0.0 0.3 / -0.1 / 0.2
LSTM-G-0.5	4 / 8 / -10 4 / -16 / 1	7.7 / 18.6 / -7.5 -7.9 / -19.7 / -10.8	2.3 / 2.1 / -1.5 -1.3 / -7.8 / 2.9	1.5 / 0.3 / -0.7 -0.7 / -5.3 / 13.2	30.9 / 10.0 / -26.6 33.2 / -13.9 / 30.9	37.0 / -19.3 / -20.3 49.5 / -23.7 / 46.3	38.2 / 4.3 / -29.8 46.7 / -16.0 / 28.1	0.2 / -0.0 / -0.0 0.2 / -0.1 / 0.2
LSTM-G-0.2	3 / 14 / -7 4 / -13 / 1	8.0 / 23.5 / -1.6 -11.0 / -20.5 / -8.8	3.6 / -0.2 / 0.9 1.7 / -7.3 / 3.2	2.8 / 0.6 / 1.5 2.7 / -4.0 / 12.6	25.4 / 15.6 / -33.7 32.9 / -13.2 / 27.4	33.1 / -15.2 / -22.5 43.8 / -24.4 / 43.6	34.6 / 15.2 / -33.2 41.5 / -14.2 / 24.8	0.2 / -0.0 / -0.0 0.2 / -0.1 / 0.2
LSTM-G-0.3	1 / 1 / -16 0 / -19 / -2	2.6 / 5.2 / -13.6 -21.7 / -24.7 / -18.1	5.7 / 1.2 / 2.8 0.7 / -5.7 / 1.9	4.5 / -0.1 / 1.5 0.4 / -2.8 / 11.2	12.2 / -8.4 / -40.2 30.5 / -30.1 / 23.5	11.2 / -18.5 / -36.6 30.5 / -45.3 / 32.0	12.2 / -15.1 / -48.1 34.6 / -30.4 / 15.5	0.2 / -0.0 / -0.0 0.3 / -0.2 / 0.2
LSTM-GM-0.5	5 / 5 / -15 5 / -21 / -3	7.0 / 10.7 / -10.0 -7.0 / -29.1 / -15.1	4.5 / 1.3 / 2.1 -2.0 / -7.5 / 0.9	4.5 / -1.1 / -1.7 -0.7 / -4.4 / 10.1	27.9 / -0.8 / -38.7 41.1 / -15.7 / 23.2	38.6 / -23.1 / -31.4 55.6 / -16.3 / 32.0	37.4 / -7.8 / -41.4 54.4 / -17.3 / 18.2	0.2 / -0.0 / -0.0 0.2 / -0.1 / 0.2
LSTM-GMB-0.5	9 / 13 / -11 6 / -15 / -3	12.7 / 25.4 / -9.2 -4.5 / -19.1 / -11.2	1.1 / 0.9 / 0.8 -0.4 / -4.8 / 4.4	1.5 / 0.1 / 1.1 0.9 / -1.4 / 10.6	24.8 / 13.8 / -29.2 40.4 / -11.6 / 19.0	27.6 / -15.7 / -20.7 54.1 / -16.6 / 36.3	31.4 / 12.3 / -31.9 51.8 / -12.3 / 18.8	0.1 / -0.0 / -0.0 0.2 / -0.1 / 0.2
GRU-G-0.5	5 / 4 / -10 3 / -15 / -4	8.0 / 11.9 / -3.0 -7.7 / -18.0 / -17.2	3.0 / 2.1 / -1.9 0.3 / -6.1 / 0.6	3.4 / 1.3 / -2.7 1.4 / -2.6 / 9.1	30.0 / 8.2 / -36.4 43.2 / -3.1 / 24.0	41.3 / -11.1 / -25.8 59.3 / -8.9 / 39.7	44.3 / 2.1 / -37.5 59.1 / -0.6 / 22.8	0.2 / 0.0 / -0.0 0.3 / -0.1 / 0.2
GRU-G-0.2	7 / 6 / -8 2 / -14 / 5	5.8 / 8.7 / -7.5 -15.3 / -24.9 / -11.9	3.1 / 4.0 / 0.1 -1.2 / -6.9 / 0.8	3.6 / 2.8 / 0.9 -0.5 / -3.1 / 10.6	25.9 / 5.5 / -32.2 40.8 / -7.0 / 36.4	33.5 / -11.3 / -19.9 62.1 / -18.3 / 42.5	33.5 / -1.0 / -32.5 58.3 / -7.8 / 30.8	0.2 / 0.0 / -0.0 0.3 / -0.1 / 0.2
GRU-G-0.3	3 / 3 / -18 -3 / -16 / -3	6.6 / 11.0 / -15.1 -26.0 / -21.3 / -16.2	5.6 / 3.5 / 6.2 2.4 / -5.9 / 1.2	3.6 / -1.0 / 5.9 3.5 / -2.4 / 9.9	13.8 / -13.2 / -44.5 33.3 / -24.7 / 17.7	12.8 / -26.6 / -35.7 42.4 / -30.1 / 29.3	14.2 / -18.6 / -48.8 38.3 / -24.9 / 13.0	0.2 / -0.0 / 0.0 0.3 / -0.2 / 0.2
GRU-GM-0.5	7 / 8 / -10 4 / -19 / -1	10.5 / 16.3 / -4.2 -10.9 / -27.7 / -12.6	5.3 / -0.3 / -0.6 0.4 / -6.6 / 1.9	4.2 / 1.4 / -0.8 1.0 / -2.9 / 11.7	30.7 / 13.4 / -31.7 40.7 / -17.9 / 24.9	37.8 / -8.8 / -19.6 54.7 / -29.0 / 41.0	39.2 / 12.0 / -30.8 51.3 / -22.4 / 21.4	0.2 / 0.0 / -0.0 0.3 / -0.1 / 0.2
GRU-GMB-0.5	8 / 13 / -11 2 / -16 / 1	10.3 / 25.5 / -7.3 -11.0 / -21.7 / -7.6	1.7 / 2.1 / 2.0 0.1 / -5.0 / 0.0	1.1 / 0.2 / 2.5 1.3 / -3.0 / 10.0	30.9 / 12.4 / -34.8 42.0 / -11.4 / 32.6	41.0 / -19.6 / -24.0 56.5 / -18.7 / 51.1	40.9 / 11.1 / -34.7 55.6 / -9.4 / 32.2	0.2 / -0.0 / -0.0 0.3 / -0.1 / 0.2

Table B.2: Differences in performance with and without assistance (e.g., *0* means no change in that metric). In each cell, top line refers to MARS pilot models and bottom to VIP pilot models, and slashes separate Good, Medium, and Bad pilot models. In the Assistant column, G/M/B denotes the proficiency of the assistant training data, decimals denote window size (for window size 0.3s, future prediction was always 0.1s; for all other window sizes, the next step [0.0s] was predicted).

we can also see from these extended results that almost any assistance was beneficial in reducing crashes for the pilot models that truly performed the task badly (e.g., both Bad models and the Medium VIP model).

B.7 Assistant Models Performance Statistics

Assistant	**Crashes**↓	**% destab.**↓	$\mu\lvert\theta\rvert$ (°)↓	$\sigma(\theta)$ (°)↓	$\mu\lvert Mag\rvert_{vel}$ (°/s)↓	$\sigma(\lvert Mag\rvert_{vel})$ (°/s)↓	vel **RMS**↓	$\mu\lvert d\rvert$↓
SAC	0	27.2	9.8	2.4	9.7	12.3	12.3	.52
SAC-BC	0	0.0	2.4	3.2	5.8	7.8	7.9	.03
SAC-AIRL	0	12.2	7.7	0.7	1.0	1.3	1.3	.08
DDPG	0	0.3	5.3	1.7	3.0	14.8	15.0	.07
DDPG-BC	0	0.7	2.5	3.2	6.3	8.0	8.1	.03
MLP-G-0	0	13.5	4.4	4.3	4.4	5.2	5.6	.05
MLP-G-0.5	23	76.0	20.6	23.0	57.9	77.5	81.8	.13
MLP-G-0.2	6	22.8	11.9	15.9	45.5	61.4	61.7	.24
MLP-G-0.3	0	20.4	22.6	25.6	58.5	68.6	68.8	.47
MLP-GM-0	0	0.0	9.7	0.2	0.3	0.7	0.7	.06
MLP-GMB-0	0	20.9	4.4	4.3	4.4	5.2	5.6	.05
RNN-G-0.5	7	9.6	17.2	21.7	71.3	97.8	99.0	.30
RNN-G-0.2	17	22.9	22.1	27.8	101.9	129.9	130.9	.49
RNN-G-0.3	0	10.0	13.3	15.8	55.5	65.6	65.8	.38
RNN-GM-0.5	14	18.7	20.0	25.1	94.5	127.5	127.7	.38
RNN-GMB-0.5	28	67.2	25.2	27.4	78.8	95.9	99.6	.23
LSTM-G-0.5	11	32.6	18.5	22.8	76.7	101.3	105.0	.29
LSTM-G-0.2	8	14.8	19.9	24.6	77.8	98.8	100.7	.31
LSTM-G-0.3	18	22.2	19.2	22.8	97.9	101.1	122.3	.57
LSTM-GM-0.5	12	31.7	18.1	22.0	78.8	100.7	104.1	.28
LSTM-GMB-0.5	8	23.0	19.3	24.8	78.9	104.4	107.7	.30
GRU-G-0.5	15	44.0	18.3	23.5	94.0	119.5	131.7	.35
GRU-G-0.2	6	14.8	17.9	22.4	60.9	82.3	83.5	.27
GRU-G-0.3	3	13.9	16.8	19.9	68.6	81.8	82.1	.46
GRU-GM-0.5	10	47.2	20.5	23.5	67.2	94.0	94.0	.21
GRU-GMB-0.5	10	21.0	18.3	22.9	76.6	103.9	105.2	.30

Table B.3: Performance statistics of all assistant models when performing task as a solo pilot (values are averaged over 3×30 sec. trials except # crashes, which is summed). Columns from L–R: # crashes, % destabilizing actions, mean and SD distance from DOB, mean and SD angular velocity magnitude, RMS velocity, and mean deflection magnitude. Assistant model evaluations were conducted without noise added to deflection time or magnitude.

Table B.3 shows performance statistics of all candidate assistant models when performing the task alone (as a pilot). Comparing this table to Table 4.2 and Table B.2, we can see that ability to perform the task well alone does not necessarily correspond to an ability to act as a good assistant. For instance, in terms of the magnitude of most metrics, MLP-GM-0 is arguably the best-performing assistant model when acting as a solo pilot but was not the most effective assistant for any of the pilot exemplars.

B.8 Heuristic Assessment of Human Acceptance of AI Suggestions

Due to the issue of varying human reaction time and no non-intrusive method for the human subjects to definitively indicate acceptance of a suggestion, we apply a heuristic to calculate the number of times an AI suggestion was provided and the human followed that direction in the duration of the trial. Table B.4 shows the proportion of the trial where the AI assistant provided a suggestion to the human pilot (P), and the proportion of the trial where the human pilot followed that suggestion (F).

Assistant	**S1 T2**		**S2 T2**		**S2 T3**	
	P	F	P	F	P	F
Overall	69.1	49.8	73.3	43.3	74.8	39.5
SAC	69.9	49.9	78.9	43.2	68.7	36.9
SAC-AIRL	84.8	52.0	79.3	44.4	86.1	56.4
DDPG	69.0	57.2	61.3	49.9	66.7	50.7
MLP-GMB-0	57.1	39.2	75.3	34.5	78.3	23.1
LSTM-G-0.2	64.7	50.5	71.7	44.4	74.4	30.4

Table B.4: Proportion of trial (as %) of Session 1 Task 2, Session 2 Task 2 and Session 2 Task 3 assistants. P: suggestion was **provided** by AI assistant, F: provided suggestion was **followed** by human pilot.

B.9 Retraining Assistants from HITL Data

Using the disagreement episodes from Session 1 Task 3, the AI assistants were retrained, or more appropriately fine-tuned. The best version of each assistant model was saved after hyperparameter tuning using the following learning rate-epoch pairs: $[(100, 1e{-}3), (100, 1e{-}4), (100, 1e{-}5), (200, 1e-3), (200, 1e-4), (200, 1e-5)]$. For the supervised learning models, the best version was selected using the MAE metric using a test trial that was held out for testing from the MARS dataset. The reinforcement learning models were evaluated by running the policy for 5 episodes and the mean reward was calculated; the model with the highest mean reward was saved for Session 2.

B.10 Details on Post-Trial Survey for PyVIP

Assistant	Task 2							Task 3						
	$+++$	$++$	$+$	$\sim$	$-$	$--$	$---$	$+++$	$++$	$+$	$\sim$	$-$	$--$	$---$
Overall	0	10	40	15	15	15	5	5	20	30	30	5	5	5
SAC	0	0	25	25	25	0	25	0	0	50	25	0	0	25
SAC-AIRL	0	0	50	0	0	50	0	0	50	0	25	0	25	0
DDPG	0	25	50	0	25	0	0	0	0	50	25	25	0	0
MLP-GMB-0	0	25	50	25	0	0	0	25	25	25	25	0	0	0
LSTM-G-0.2	0	0	25	25	25	25	0	0	25	25	50	0	0	0

Table B.5: Perceived performance impact (as %) of Session 2 Task 2 and Task 3 assistants at finer granularity. $+++$: Significantly improved, $++$: Improved, $+$: Slightly improved, $\sim$: No significant impact, $-$: Slightly decreased, $--$: Decreased, $---$: Significantly decreased.

Each human subject took a survey to assess their trust in the AI assistants they were provided with. The survey was based on Bonnie Muir's survey on trust in automated systems, which is generally regarded as standard [122]. Below we provide the questions asked in each survey taken by the human subjects, along with answer format or options (if multiple choice).

B.10.1 Session 1

1) Email: *text input*

Assistant	Task 2							Task 3						
	$+++$	$++$	$+$	$\sim$	$-$	$--$	$---$	$+++$	$++$	$+$	$\sim$	$-$	$--$	$---$
Overall	0	10	15	35	25	10	5	5	10	15	45	10	10	5
SAC	0	25	0	0	50	0	25	0	25	0	50	0	25	0
SAC-AIRL	0	0	0	50	25	25	0	0	0	25	50	0	25	0
DDPG	0	0	0	75	0	25	0	0	0	25	25	25	0	25
MLP-GMB-0	0	25	25	25	25	0	0	25	0	0	50	25	0	0
LSTM-G-0.2	0	0	50	25	25	0	0	0	25	25	50	0	0	0

Table B.6: Reported level of trust (as %) of Session 2 Task 2 and Task 3 assistants at finer granularity. $+++$: Complete, $++$: Very high, $+$: High, $\sim$: Moderate, $-$: Low, $--$: Very low, $---$: No trust at all.

2) Name: *text input*

3) Age: *18-24, 25-34, 35-44, 45-54, 55 & above*

4) Gender: *text input*

5) Department at University: *text input*

6) How would you rate your performance in task 1? *Extremely Poor, Poor, Below Average, Average, Above Average, Good, Excellent*

7) In task 2, how did the AI's suggestions change your performance? *Decreased performance significantly, Decreased performance, Slightly decreased performance, No significant impact on performance, Slightly improved performance, Improved performance, Significantly improved performance*

8) In task 3, to what extent can the AI's behavior be predicted from moment to moment? *Very unpredictable, Unpredictable, Somewhat unpredictable, Neither predictable nor unpredictable, Somewhat predictable, Predictable, Very predictable*

9) To what extent can you count on the AI to do its job? *Cannot count on it at all, Can hardly count on it, Can somewhat count on it, Can count on it moderately, Can count on it to a good extent, Can largely count on it, Can completely count on it*

10) What degree of faith do you have that the AI will be able to cope with similar situations in the future? *No faith at all, Very low faith, Low faith, Moderate faith, High faith, Very high faith, Complete faith*

11) Overall, how much do you trust the AI? *No trust at all, Very low trust, Low trust, Moderate trust, High trust, Very high trust, Complete trust*

Tables B.5 and B.6 show subject responses to the Session 2 survey questions regarding performance impact and trust in the assistant, broken down at the finer granularity from all possible answer options (these should be compared to Tables 4 and 5 in the main body, where the positive and negative valence options were conflated for space reasons). As in Tables 4 and 5, we see general trends toward more positive perceived impact on task performance, and greater levels of trust, in the Task 3 assistants, which were fine-tuned on human subject data.

B.10.2 Session 2

1) Email: *text input*

2) Name: *text input*

3) How would you rate your performance in task 1? *Extremely Poor, Poor, Below Average, Average, Above Average, Good, Excellent*

4) In task 2, how did the AI's suggestions change your performance? *Decreased performance significantly, Decreased performance, Slightly decreased performance, No significant impact on performance, Slightly improved performance, Improved performance, Significantly improved performance*

5) Overall, how much do you trust the AI? *No trust at all, Very low trust, Low trust, Moderate trust, High trust, Very high trust, Complete trust*

6) In task 3, how did the AI's suggestions change your performance? *Decreased performance significantly, Decreased performance, Slightly decreased performance, No significant im-*

pact on performance, Slightly improved performance, Improved performance, Significantly improved performance

7) Overall, how much do you trust the AI? *No trust at all, Very low trust, Low trust, Moderate trust, High trust, Very high trust, Complete trust*

8) Comparing the AI in task 2 vs task 3, which AI would you prefer? *Strongly prefer Task 2 AI, Prefer Task 2 AI, Slightly prefer Task 2 AI, No preference, Slightly prefer Task 3 AI, Prefer Task 3 AI, Strongly prefer Task 3 AI*

Appendix C

Appendix for Chapter 5: Extending AI Guidance to a Navigational Flight Task

C.1 PyFlyt Session 1 Survey

C.1.1 Demographics and Background

1. Subject ID (ask from experimenter) _____

2. Your full name _____

3. What is your department at CSU? _____

4. Age

   ☐ 18-24

   ☐ 25-34

   ☐ 35-44

   ☐ 45-54

   ☐ 55 & above

5. Gender _____

6. Do you or have you ever played video games?

   ☐ Yes (racing games, flight simulators)

   ☐ Yes (but none of the above)

   ☐ No

C.1.2 PyFlyt Session 1 - Task 1

Task 1 was when you were flying the plane alone

7. How would you rate your performance in the task?

   ☐ Extremely Poor

   ☐ Poor

   ☐ Below Average

   ☐ Average

   ☐ Above Average

   ☐ Good

   ☐ Excellent

8. How mentally demanding was the task?

   ☐ Very Low

   ☐ Low

   ☐ Somewhat Low

   ☐ Neither Low or High

   ☐ Somewhat High

   ☐ High

   ☐ Very High

9. How physically demanding was the task?

   ☐ Very Low

   ☐ Low

   ☐ Somewhat Low

☐ Neither Low or High

☐ Somewhat High

☐ High

☐ Very High

10. How hurried or rushed was the pace of the task?

☐ Very Low

☐ Low

☐ Somewhat Low

☐ Neither Low or High

☐ Somewhat High

☐ High

☐ Very High

11. How hard did you have to work to accomplish your level of performance?

☐ Very Low

☐ Low

☐ Somewhat Low

☐ Neither Low or High

☐ Somewhat High

☐ High

☐ Very High

12. How insecure, discouraged, irritated, stressed, or annoyed were you?

☐ Very Low

☐ Low

☐ Somewhat Low

☐ Neither Low or High

☐ Somewhat High

☐ High

☐ Very High

C.1.3 PyFlyt Session 1 - Task 2

Task 2 was when you were flying the plane with a form of AI task guidance

13. How would you rate your performance in the task?

    ☐ Extremely Poor

    ☐ Poor

    ☐ Average

    ☐ Above Average

    ☐ Good

    ☐ Excellent

14. How mentally demanding was the task?

    ☐ (Same scale as above: Very Low to Very High)

15. How physically demanding was the task?

16. How hurried or rushed was the pace of the task?

17. How hard did you have to work to accomplish your level of performance?

18. How insecure, discouraged, irritated, stressed, or annoyed were you?

19. How did the AI's suggestions change your performance?

    ☐ Decreased performance significantly

    ☐ Decreased performance

    ☐ Slightly decreased performance

    ☐ No significant impact on performance

    ☐ Slightly improved performance

    ☐ Improved performance

    ☐ Significantly improved performance

20. Overall, how much do you trust the AI?

    ☐ No trust at all

    ☐ Very low trust

    ☐ Low trust

    ☐ Moderate trust

    ☐ High trust

    ☐ Very high trust

    ☐ Complete trust

C.1.4 PyFlyt Session 1 - Task 3

Task 3 was when you were flying the plane with a new form of AI task guidance

21. How would you rate your performance in the task?

22. How mentally demanding was the task?

23. How physically demanding was the task?

24. How hurried or rushed was the pace of the task?

25. How hard did you have to work to accomplish your level of performance?

26. How insecure, discouraged, irritated, stressed, or annoyed were you?

27. How did the AI's suggestions change your performance?

28. Overall, how much do you trust the AI?

C.1.5 Final Questions

29. Comparing the assistance provided in task 2 vs task 3, which mode would you prefer?

    ☐ Strongly prefer task 2

    ☐ Prefer task 2

    ☐ Slightly prefer task 2

    ☐ No preference

    ☐ Slightly prefer task 3

    ☐ Prefer task 3

    ☐ Strongly prefer task 3

30. Why did you think the guidance method you chose was better than the other?

    _____

31. At any point during the task were you spatially disoriented, nauseated or dizzy?

    ☐ Yes

    ☐ No

    ☐ Maybe

32. If yes, kindly elaborate during which task you were in when it occurred, and what made you disoriented? _____

33. Did you feel the need to pause? If so, could you elaborate on when and why?

    _____

C.2 PyFlyt Session 2 Survey

C.2.1 Identification

1. Subject ID (ask from experimenter) _____

C.2.2 PyFlyt Session 2 - Task 1

Task 1 was when you were flying the plane alone

2. How would you rate your performance in the task?

    ☐ Extremely Poor

    ☐ Poor

    ☐ Average

    ☐ Above Average

    ☐ Good

    ☐ Excellent

3. How mentally demanding was the task?

    ☐ Very Low

    ☐ Low

    ☐ Somewhat Low

    ☐ Neither Low or High

    ☐ Somewhat High

    ☐ High

    ☐ Very High

4. How physically demanding was the task?

   ☐ Very Low

   ☐ Low

   ☐ Somewhat Low

   ☐ Neither Low or High

   ☐ Somewhat High

   ☐ High

   ☐ Very High

5. How hurried or rushed was the pace of the task?

   ☐ Very Low

   ☐ Low

   ☐ Somewhat Low

   ☐ Neither Low or High

   ☐ Somewhat High

   ☐ High

   ☐ Very High

6. How hard did you have to work to accomplish your level of performance?

   ☐ Very Low

   ☐ Low

   ☐ Somewhat Low

   ☐ Neither Low or High

   ☐ Somewhat High

☐ High

☐ Very High

7. How insecure, discouraged, irritated, stressed, or annoyed were you?

   ☐ Very Low

   ☐ Low

   ☐ Somewhat Low

   ☐ Neither Low or High

   ☐ Somewhat High

   ☐ High

   ☐ Very High

C.2.3 PyFlyt Session 2 - Task 2

Task 2 was when you were flying the plane with a form of AI task guidance

8. How would you rate your performance in the task?

   ☐ Extremely Poor

   ☐ Poor

   ☐ Average

   ☐ Above Average

   ☐ Good

   ☐ Excellent

9. How mentally demanding was the task?

   ☐ (Same scale as Task 1: Very Low to Very High)

10. How physically demanding was the task?

11. How hurried or rushed was the pace of the task?

12. How hard did you have to work to accomplish your level of performance?

13. How insecure, discouraged, irritated, stressed, or annoyed were you?

14. How did the AI's suggestions change your performance?

    ☐ Decreased performance significantly

    ☐ Decreased performance

    ☐ Slightly decreased performance

    ☐ No significant impact on performance

    ☐ Slightly improved performance

    ☐ Improved performance

    ☐ Significantly improved performance

15. Did the assistance appear at the moment you felt you needed help?

    ☐ (Choose this option if the assistance was always on)

    ☐ Much too early

    ☐ Slightly too early

    ☐ Just in time

    ☐ Neutral

    ☐ Slightly too late

    ☐ Much too late

    ☐ Did not appear when needed

16. Overall, how much do you trust the AI?

    ☐ No trust at all

☐ Very low trust

☐ Low trust

☐ Moderate trust

☐ High trust

☐ Very high trust

☐ Complete trust

C.2.4 PyFlyt Session 2 - Task 3

Task 3 was when you were flying the plane with a new form of AI task guidance

17. (Questions repeat the metrics used in Task 2)

18. How would you rate your performance in the task?

19. How mentally demanding was the task?

20. How physically demanding was the task?

21. How hurried or rushed was the pace of the task?

22. How hard did you have to work to accomplish your level of performance?

23. How insecure, discouraged, irritated, stressed, or annoyed were you?

24. How did the AI's suggestions change your performance?

25. Did the assistance appear at the moment you felt you needed help?

26. Overall, how much do you trust the AI?

C.2.5 Final Questions

27. Comparing the assistance provided in task 2 vs task 3, which mode would you prefer?

☐ Strongly prefer task 2

☐ Prefer task 2

☐ Slightly prefer task 2

☐ No preference

☐ Slightly prefer task 3

☐ Prefer task 3

☐ Strongly prefer task 3

28. Why did you think the guidance method you chose was better than the other?

    _____

29. At any point during the task were you spatially disoriented, nauseated or dizzy?

    ☐ Yes

    ☐ No

    ☐ Maybe

30. If yes, kindly elaborate during which task you were in when it occurred, and what made you disoriented? _____

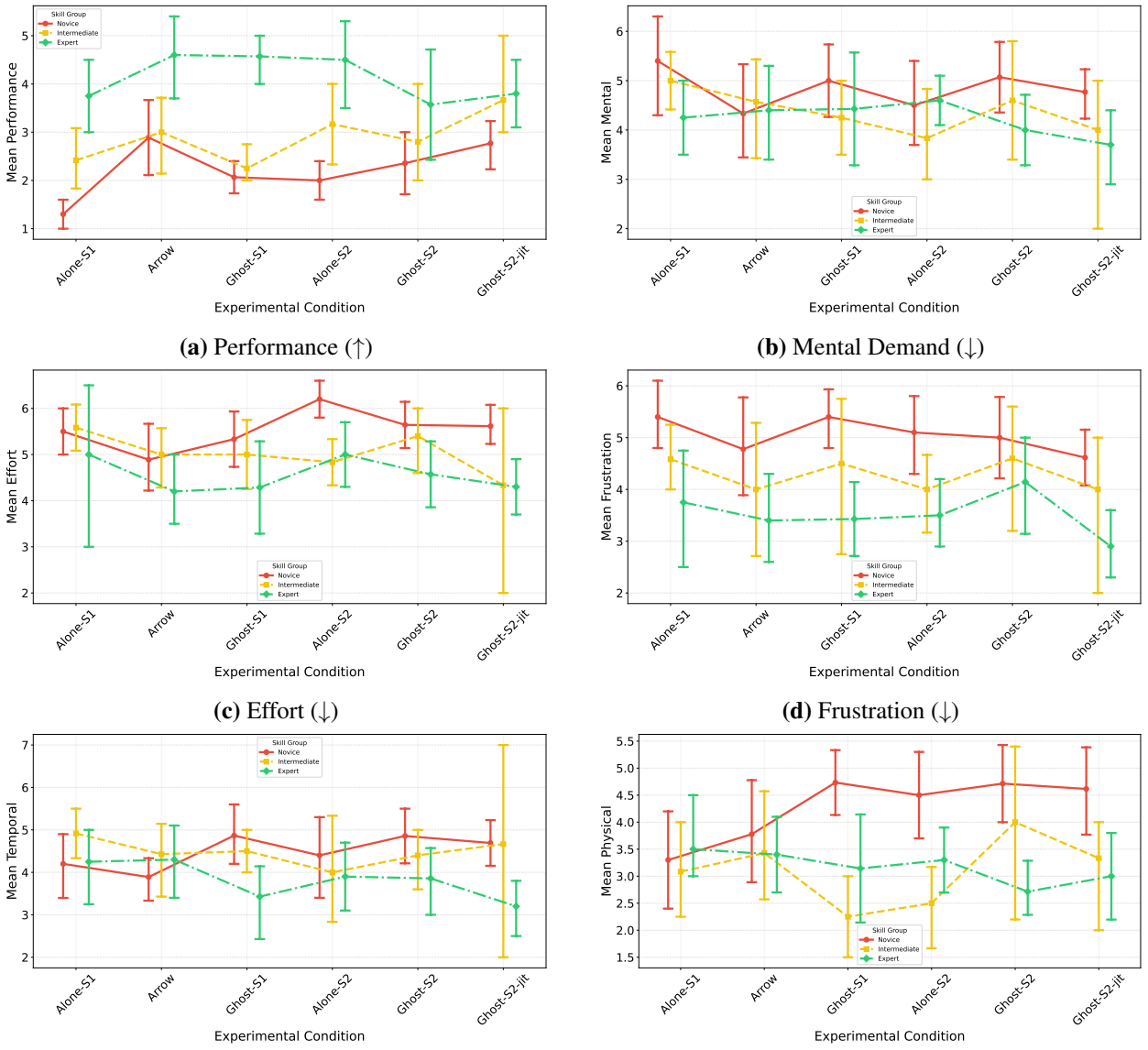
31. Did you feel the need to pause? If so, could you elaborate on when and why?

    _____

C.3 NASA-TLX Survey Results

The results from the NASA-TLX survey are further broken down by skill groups in Figure C.1. The breakdown is not surprising where experts display lower task workload than others apart from mental demand which is no clear skill group that comes out the winner. Surprisingly, experts which had better task performance overall but were hurt by AI assistance in certain metrics like $\mathcal{R}_{inv}$, $\#Waypoints$, and $\mathcal{C}_{ext}$ still display higher trust in AI and at the same time show low frustration

levels. I hypothesize if human subjects were informed of their objective performance after each task the subjective survey results would align more with objective results and what are typically observed in other studies related to trust and reliance as in [59, 83].



(a) Performance (↑)

(b) Mental Demand (↓)

(c) Effort (↓)

(d) Frustration (↓)

(e) Temporal Demand (↓)

(f) Physical Demand (↓)

Figure C.1: Distribution of NASA-TLX facets on 7-point Likert scale. Lower values correlate with lower task workload except for performance where higher values are preferred.