



Delft University of Technology

Document Version

Final published version

Citation (APA)

Zhang, R. (2026). *Information-Processing Alignment in Human-Robot Interaction*. [Dissertation (TU Delft), Delft University of Technology]. <https://doi.org/10.4233/uuid:1a4e58ba-0c9e-49a3-857a-745ab463bca7>

Important note

To cite this publication, please use the final published version (if applicable).
Please check the document version above.

Copyright

In case the licence states "Dutch Copyright Act (Article 25fa)", this publication was made available Green Open Access via the TU Delft Institutional Repository pursuant to Dutch Copyright Act (Article 25fa, the Taverne amendment). This provision does not affect copyright ownership.
Unless copyright is transferred by contract or statute, it remains with the copyright holder.

Sharing and reuse

Other than for strictly personal use, it is not permitted to download, forward or distribute the text or part of it, without the consent of the author(s) and/or copyright holder(s), unless the work is under an open content license such as Creative Commons.

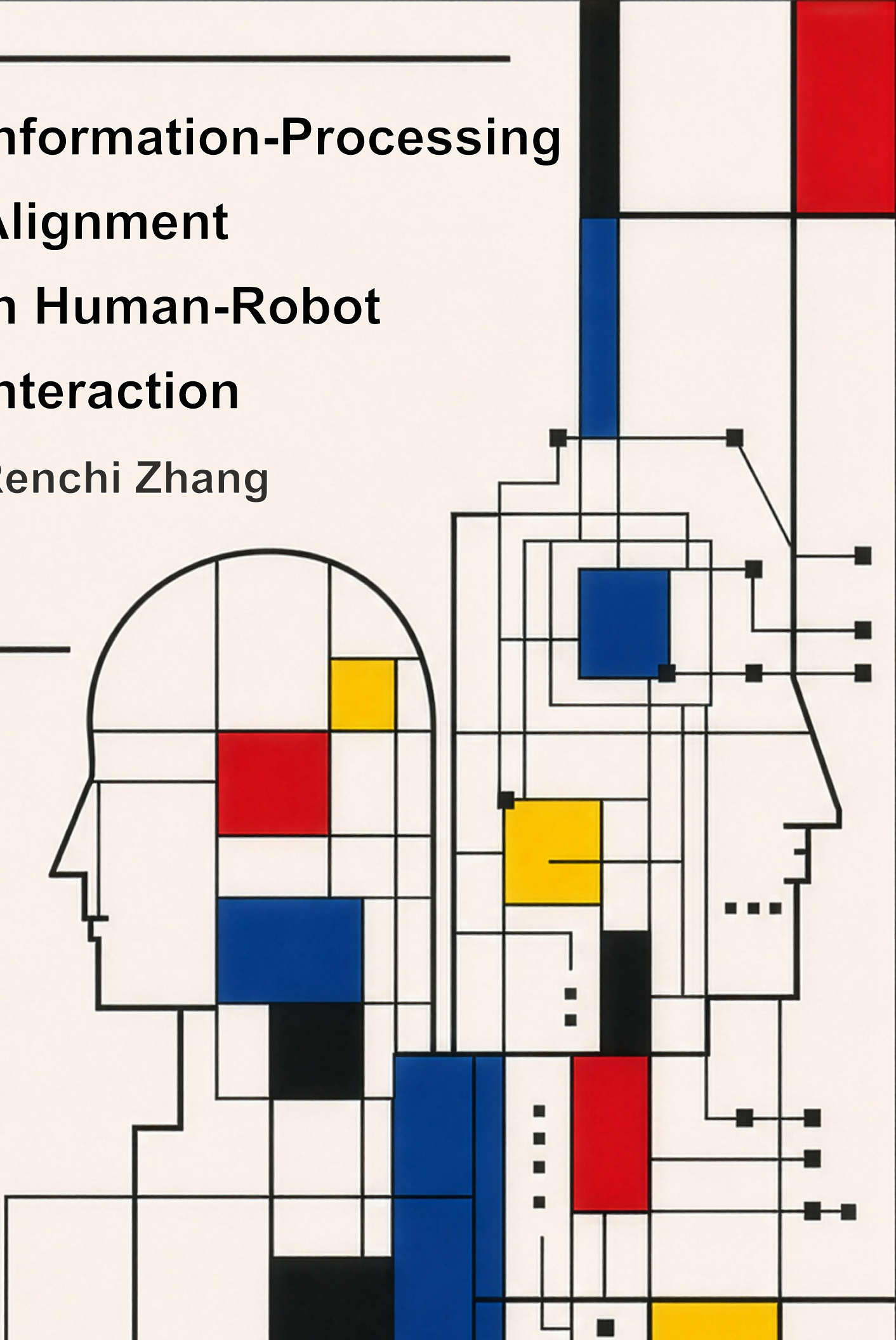
Takedown policy

Please contact us and provide details if you believe this document breaches copyrights.
We will remove access to the work immediately and investigate your claim.

This work is downloaded from Delft University of Technology.

Information-Processing Alignment in Human-Robot Interaction

Renchi Zhang



Information-Processing Alignment in Human-Robot Interaction

Dissertation

for the purpose of obtaining the degree of doctor

at Delft University of Technology

by the authority of the Rector Magnificus,

Prof.dr.ir. H. Bijl,

chair of the Board for Doctorates

to be defended publicly on

Monday, 14 September 2026 at 15:00

by

Renchi ZHANG

This dissertation has been approved by the (co)promoters.

Composition of the doctoral committee:

Rector Magnificus,	chairperson
Prof. dr. ir. J. C. F. de Winter,	Delft University of Technology, promotor
Dr. D. Dodou,	Delft University of Technology, promotor
Dr. ir. Y. B. Eisma,	Delft University of Technology, copromotor
Dr. H. C. Seyffert,	Delft University of Technology, copromotor

Independent Members:

Prof. dr. ir. J. Hellendoorn,	Delft University of Technology
Prof. dr. R. R. Negenborn,	Delft University of Technology
Prof. dr. ir. W. P. Brinkman,	Delft University of Technology
Prof. dr. S. van der Stigchel,	Utrecht University
Dr. ir. J. F. P. Kooij,	Delft University of Technology, <i>reserve member</i>



Key words: Human-robot interaction, information-processing alignment, multimodal large language models, human-AI collaboration, cognitive ergonomics

Printed by: Gildeprint

Cover by: designed by Renchi Zhang; image generated with ChatGPT (OpenAI)

Copyright: © 2026 by Renchi Zhang

ISBN: 978-94-6518-387-9

An electronic version of this dissertation is available at: <http://repository.tudelft.nl>

Contents

Summary	v
Samenvatting	ix
1. Introduction	1
1.1 Motivation.....	2
1.2 Human and robot information processing.....	3
1.3 Research questions and thesis content.....	3
1.4 Summary.....	7
2. Human-like attention? A psychophysical comparison of visual search in humans and MLLMs	11
2.1 Introduction.....	12
2.1.1 Visual processing and potential bottlenecks in MLLMs.....	13
2.1.2 The present study: a comparative psychophysical approach.....	15
2.2 Methods.....	16
2.2.1 Experimental conditions.....	16
2.2.2 MLLM evaluation.....	19
2.2.3 Human evaluation.....	19
2.2.4 Analysis.....	20
2.3 Results.....	21
2.3.1 Human performance.....	21
2.3.2 MLLM performance.....	23
2.3.3 MLLMs vs humans.....	25
2.4 Discussion.....	26
2.4.1 Parallels in human and MLLM performance.....	26
2.4.2 Differences between human and MLLM performance.....	27
2.4.3 Limitations.....	28
2.4.4 Conclusion and future directions.....	28
3. Walk along: An experiment on controlling the mobile robot ‘Spot’ with voice and gestures	39
3.1 Introduction.....	40
3.1.1 Voice and gestures as natural interfaces.....	41
3.1.2 Existing applications of voice and gestures for mobile robot control.....	41
3.1.3 A comparative study of voice and gesture control in standing and walking conditions.....	43
3.2 Methods.....	44
3.2.1 Participants.....	44
3.2.2 Hardware and software.....	44
3.2.3 Task.....	46
3.2.4 Independent variables.....	47
3.2.5 Speech recognition.....	48

3.2.6 Gesture recognition.....	48
3.2.7 Heads-up display information.....	49
3.2.8 Dependent measures.....	51
3.2.9 Experimental procedure.....	54
3.2.10 Statistical analyses.....	55
3.3 Results.....	56
3.3.1 Trial completion and missing data.....	56
3.3.2 Dependent measures.....	57
3.3.3 Post-experiment interviews.....	62
3.4 Discussion.....	65
3.4.1 Effects of voice control versus gesture control.....	65
3.4.2 Effects of participant mobility (walking vs. standing).....	66
3.4.3 Limitations.....	66
3.5 Conclusion.....	67
4. An open-source reproducible chess robot for human-robot interaction research	93
4.1 Introduction.....	94
4.2 Hardware and software architecture.....	96
4.3 Modules.....	98
4.3.1 Perception.....	99
4.3.2 Analysis & evaluation.....	102
4.3.3 Motion planning & execution.....	102
4.4 Interactive modules.....	104
4.4.1 Verbal interaction.....	105
4.4.2 Non-verbal interaction.....	106
4.5 Technical evaluation.....	107
4.5.1 Chess perception.....	107
4.5.2 Grasping of pieces.....	110
4.5.3 Performance in human-robot chess play.....	111
4.6 Evaluation of people's views on the chess robot.....	112
4.6.1 Participants.....	113
4.6.2 Experiment design.....	113
4.6.3 Data analysis.....	114
4.6.4 Results.....	114
4.7 Discussion and conclusion.....	117
5. Educating future engineers about LLMs: A scalable workshop	137
5.1 Introduction.....	138
5.2 Methods.....	140
5.3 Results.....	147
5.4 Discussion.....	153

6. Discussion and conclusions	163
6.1 Recap: Motivation, research questions, and approach.....	164
6.2 Main research findings.....	165
6.2.1 Chapter 2: Human-like attention? A psychophysical comparison of visual search in humans and MLLMs.....	165
6.2.2 Chapter 3: Walk Along: An Experiment on Controlling the Mobile Robot ‘Spot’ with Voice and Gestures.....	166
6.2.3 Chapter 4: An Open-Source Reproducible Chess Robot for Human-Robot Interaction Research.....	167
6.2.4 Chapter 5: Educating future engineers about LLMs: A scalable workshop.....	167
6.2.5 Synthesis and misalignment as a unifying construct.....	168
6.3 Practical contributions.....	169
6.4 Limitations.....	171
6.5 Recommendations for future research.....	172
6.6 Broader reflections and outlook.....	174
6.7 Conclusions.....	176
Acknowledgement	181
Curriculum vitae	183
List of publications	184
Propositions	186

Summary

Machines that work alongside humans have progressed from mechanical devices to AI-driven robots that can perceive, reason, and act in environments shared with people. Multimodal large language models (MLLMs) now enable robots to interpret open-ended instructions, plan multi-step actions, and generate natural language explanations. However, technical capability alone does not guarantee effective collaboration, because humans and AI-driven robots perceive, reason, and respond through different information-processing mechanisms, and when these processes are misaligned, interaction can fail regardless of the robot's capability.

This thesis investigated where alignment and misalignment occur between humans and AI-driven robots, and how human-robot interaction should be designed for complementary collaboration. This thesis adopts the vocabulary of Wickens' Multiple Resource Theory, which distinguishes processing stages (perception, cognition, responding), processing modalities (visual and auditory), and processing codes (spatial, verbal). Four empirical studies targeted different processing stages and progressed in platform complexity and realism.

Chapter 2 isolated the **perception stage** and focused on **visual-spatial** processing. Human visual search is predominantly focal, relying on serial foveal fixations, whereas MLLMs process the entire image in a single parallel pass, analogous to ambient processing. To empirically test the difference between human and AI perception, 1,250 online participants and three MLLMs completed identical visual search tasks that ranged from simple feature searches to complex conjunction searches with varying distractor heterogeneity in 2D and 3D scenes. The results showed that the outcome-level correlation between human error rates and MLLM error rates was strong. However, the underlying mechanisms differed: humans exhibited a speed-accuracy trade-off characteristic of foveated search strategies, whereas the MLLMs defaulted uniformly to 'present' or 'absent' responses in complex scenes.

Chapter 3 examined how the human operator's command modality affects performance and preference during teleoperation. A total of 218 participants navigated a Boston Dynamics Spot robot along an indoor trajectory using the Microsoft HoloLens 2 in a within-subjects 2×2 design crossing command modality (voice vs. gesture) with operator mobility (standing vs. walking). The experiment covered **all three processing stages**: the operator monitored the robot visually (**perception, visual-spatial**), maintained a spatial model of the robot's pose and selected the next command (**cognition, spatial**), and issued commands through either speech (**response, verbal**) or gesture (**response, spatial**). Results showed that voice control produced faster inter-command times, higher self-reported detection ratings, and was preferred by 71% of participants. Gesture control while standing in a fixed position produced significantly more left-right errors than while walking, pointing to stimulus-response incompatibility arising from mental rotation of gestures into the robot's reference frame. Because gestures share the spatial code already used for visual-spatial monitoring, they competed for the same resources, whereas verbal-coded voice commands did not. When allowed to walk, participants physically

aligned themselves with the robot more in the gesture condition, reducing spatial misalignment through repositioning.

Chapter 4 addressed the need for a controlled yet sufficiently complex testbed to study alignment across the perception and response stages simultaneously. An open-source chess-playing robot, the OpenChessRobot, integrated **visual-spatial perception** of the chessboard with LLM-generated spoken commentary (**response, verbal**) and physical piece manipulation (**response, spatial**). The platform combined a Franka Emika Panda robot arm, a ZED2 stereo camera, the Stockfish chess engine, and ChatGPT. An international survey of 597 online participants found highest acceptance for robotics education and lowest for home entertainment. An expert session with a FIDE Master revealed that the LLM-based commentary projected verbal confidence that exceeded the actual strategic understanding of the game, illustrating a form of response misalignment in which fluency masks shallow reasoning.

Chapter 5 focused on the **cognition stage**, engaging both **spatial and verbal processing** codes: students formulated verbal prompts that encoded spatial navigation goals and evaluated the spatial plans that LLMs produced. The study asked whether a scalable, gamified workshop could teach engineering students to recognize LLM reasoning limitations and redistribute cognitive effort when prompting the LLM. In ten sessions, 226 MSc engineering students prompted a non-reasoning and a reasoning LLM to solve navigation tasks of increasing complexity, with plans executed in simulation and on a Spot robot. The reasoning model achieved more error-free solutions than the non-reasoning model, but its performance declined for the harder tasks. Students adapted their prompting strategies accordingly: step-by-step instructions for the non-reasoning model, but higher-level guidance such as waypoints and heuristics for the reasoning model on harder tasks. The chapter demonstrated that misalignment could serve as a learning opportunity: by experiencing where LLM reasoning broke down, students adapted by redistributing cognitive work and better calibrating their collaboration with the model, addressing the “complementary” aim of the overarching research question.

Across the four studies, a combined 2,292 participants generated evidence that misalignment can occur at every information-processing stage. Three overarching findings are extracted. First, outcome-level results alone are insufficient for detecting misalignment: high correlations in Chapter 2 masked different search mechanisms, and verbal fluency in Chapter 4 masked shallow reasoning. Second, as the AI system becomes more automated, the human role changes from direct operator to supervisor, from commanding every movement (Chapter 3), through evaluating automated perception and strategy (Chapter 4), to diagnosing complete LLM-generated plans (Chapter 5). Third, the quality of collaboration depends on how cognition is distributed, not only on the absolute capability of either partner: voice control freed spatial resources while gestures overloaded them (Chapter 3), and students learned to redistribute cognitive work based on model capability and task difficulty (Chapter 5).

This thesis also has limitations. The studies focused heavily on the visual modality; haptic and auditory modalities were not explicitly investigated. Each study concerned a specific domain and

a relatively homogeneous participant population. Whether the documented forms of misalignment transfer to other human-robot-interaction domains, more complex environments, and more diverse populations requires further investigation. Nonetheless, the findings support a set of design principles: use modality switching to offload overloaded cognitive resources, expose AI uncertainty rather than only AI outcomes, provide calibration training before deployment, and design function allocation as dynamic instead of static.

Samenvatting

Machines die naast mensen werken zijn geëvolueerd van mechanische apparaten tot AI-gestuurde robots die kunnen waarnemen, redeneren en handelen in omgevingen die zij met mensen delen. Multimodale grote taalmodellen (MLLMs) stellen robots tegenwoordig in staat om open instructies te interpreteren, meerstaps-acties te plannen en uitleg te geven in natuurlijke taal. Echter, technische capaciteit alleen garandeert geen effectieve samenwerking, omdat mensen en AI-gestuurde robots waarnemen, redeneren en reageren via verschillende informatieverwerkingsmechanismen. Wanneer deze processen niet op elkaar zijn afgestemd, kan de interactie mislukken, ongeacht de capaciteit van de robot.

Dit proefschrift onderzoekt waar afstemming en misafstemming optreden tussen mensen en AI-gestuurde robots, en hoe mens-robotinteractie ontworpen zou moeten worden voor complementaire samenwerking. Dit proefschrift hanteert de terminologie van Wickens' Multiple Resource Theory, welke onderscheid maakt tussen verwerkingsstadia (perceptie, cognitie, respons), verwerkingsmodaliteiten (visueel en auditief) en verwerkingscodes (ruimtelijk, verbaal). Vier empirische studies richtten zich op verschillende verwerkingsstadia en namen toe in platform-complexiteit en realisme.

Hoofdstuk 2 isoleerde het **perceptiestadium** en richtte zich op **visueel-ruimtelijke** verwerking. Menselijk visueel zoeken is overwegend focaal en berust op seriële foveale fixaties, terwijl MLLMs het volledige beeld in één parallelle stap verwerken, vergelijkbaar met ambient processing. Om het verschil tussen menselijke en AI-perceptie empirisch te toetsen, voltooiden 1.250 online deelnemers en drie MLLMs identieke visuele zoektaken die varieerden van eenvoudige feature searches tot complexe conjunction searches met wisselende heterogeniteit van afleiders in 2D- en 3D-scènes. De resultaten lieten zien dat de correlatie op uitkomstniveau tussen menselijke foutpercentages en MLLM-foutpercentages sterk was. De onderliggende mechanismen verschilden echter: mensen vertoonden een snelheid-nauwkeurigheds trade-off die kenmerkend is voor foveale zoekstrategieën, terwijl de MLLMs in complexe scènes terugvielen op eenzijdige 'present' of 'absent' antwoorden.

Hoofdstuk 3 onderzocht hoe de commando-modaliteit van de menselijke operator de prestaties en voorkeur tijdens teleoperatie beïnvloedt. In totaal navigeerden 218 deelnemers een Boston Dynamics Spot-robot langs een traject met behulp van de Microsoft HoloLens 2 in een within-subjects 2x2-ontwerp dat commando-modaliteit (stem versus gebaar) kruiste met mobiliteit van de operator (stilstaand versus lopend). Het experiment besloeg **alle drie de verwerkingsstadia**: de operator volgde de robot visueel (**perceptie, visueel-ruimtelijk**), onderhield een ruimtelijk model van de pose van de robot en koos het volgende commando (**cognitie, ruimtelijk**), en gaf commando's via spraak (**respons, verbaal**) of gebaar (**respons, ruimtelijk**). De resultaten lieten zien dat stembesturing snellere intervallen tussen commando's opleverde, hogere zelfgerapporteerde detectie-beoordelingen ontving en de voorkeur had van

71% van de deelnemers. Gebarenbesturing tijdens het staan op een vaste positie leverde significant meer links-rechts-fouten op dan tijdens lopen, wat wijst op stimulus-respons incompatibiliteit als gevolg van mentale rotatie van gebaren naar het referentiekader van de robot. Omdat gebaren dezelfde ruimtelijke code delen die al voor visueel-ruimtelijke monitoring werd gebruikt, concurreerden ze om dezelfde hulpbronnen, terwijl verbaal gecodeerde stemcommando's dat niet deden. Wanneer deelnemers mochten lopen, brachten ze zich in de gebarenconditie fysiek meer in lijn met de robot en verminderden zo de ruimtelijke misafstemming via herpositionering.

Hoofdstuk 4 behandelde de behoefte aan een gecontroleerd doch voldoende complex testbed om afstemming over de perceptie- en respons-stadia gelijktijdig te onderzoeken. Een open-source schakende robot, de OpenChessRobot, integreerde **visueel-ruimtelijke perceptie** van het schaakbord met door een LLM gegenereerd gesproken commentaar (**respons, verbaal**) en fysieke manipulatie van de stukken (**respons, ruimtelijk**). Het platform combineerde een Franka Emika Panda-robotarm, een ZED2-stereocamera, de Stockfish-schaakengine en ChatGPT. Een internationale enquête onder 597 online deelnemers liet zien dat de acceptatie het hoogst was voor robotica-onderwijs en het laagst voor thuisentertainment. Een expertsessie met een FIDE-Meester onthulde dat het op LLM gebaseerde commentaar verbaal vertrouwen uitstraalde dat het werkelijke strategische begrip van het spel overtrof, een vorm van misafstemming in de respons, waarin vloeiendheid oppervlakkig redeneren maskeert.

Hoofdstuk 5 richtte zich op het **cognitiestadium** en betrok zowel **ruimtelijke als verbale verwerkingscodes**: studenten formuleerden verbale prompts die ruimtelijke navigatie-doelen codeerden en evalueerden de ruimtelijke plannen die LLMs produceerden. De studie onderzocht of een schaalbare, gegamificeerde workshop technische studenten kon leren om de redeneerlimieten van LLMs te herkennen en cognitieve inspanning te herverdelen bij het prompten van de LLM. In tien sessies lieten 226 MSc-studenten technische wetenschappen een niet-redenerend en een redenerend LLM navigatietaken oplossen van toenemende complexiteit, waarbij de plannen werden uitgevoerd in simulatie en op een Spot-robot. Het redenerende model behaalde meer foutloze oplossingen dan het niet-redenerende model, maar de prestaties namen af bij de moeilijkere taken. Studenten pasten hun promptingstrategieën dienovereenkomstig aan: stapsgewijze instructies voor het niet-redenerende model, en hogere-niveau begeleiding zoals waypoints en heuristieken voor het redenerende model bij moeilijkere taken. Het hoofdstuk toonde aan dat misafstemming als leermogelijkheid kan dienen: door te ervaren waar het redeneren van de LLM faalde, herverdeelde studenten het cognitieve werk en kalibreerden zij hun samenwerking met het model opnieuw, en gaven zo invulling aan het “complementariteits”-doel van de overkoepelende onderzoeksvraag.

Over de vier studies heen, genereerden in totaal 2.292 deelnemers bewijs dat misafstemming in elk informatieverwerkingsstadium kan optreden. Drie overkoepelende bevindingen worden uitgelicht. Ten eerste zijn resultaten op uitkomstniveau alleen onvoldoende om misafstemming te detecteren: hoge correlaties in Hoofdstuk 2 maskeerden verschillende zoekmechanismen, en verbale vloeiendheid in Hoofdstuk 4 maskeerde oppervlakkig redeneren. Ten tweede verandert

de rol van de mens, naarmate het AI-systeem meer geautomatiseerd wordt, van directe operator naar supervisor: van het commanderen van iedere beweging (Hoofdstuk 3), via het evalueren van geautomatiseerde perceptie en strategie (Hoofdstuk 4), tot het diagnosticeren van volledige door een LLM gegenereerde plannen (Hoofdstuk 5). Ten derde hangt de kwaliteit van de samenwerking af van de manier waarop cognitie wordt verdeeld, en niet alleen van de absolute capaciteit van een van beide partners: stembesturing maakte ruimtelijke hulpbronnen vrij, terwijl gebaren deze overbelastten (Hoofdstuk 3), en studenten leerden cognitief werk te herverdelen op basis van de capaciteit van het model en de moeilijkheid van de taak (Hoofdstuk 5).

Dit proefschrift kent ook beperkingen. De studies richtten zich grotendeels op de visuele modaliteit; haptische en auditieve modaliteiten zijn niet expliciet onderzocht. Elke studie betrof een specifiek domein en een relatief homogene deelnemerspopulatie. Of de gedocumenteerde vormen van misafstemming overdraagbaar zijn naar andere mens-robot-interactie-domeinen, complexere omgevingen en meer diverse populaties vergt nader onderzoek. Niettemin ondersteunen de bevindingen een aantal ontwerpprincipes: gebruik het wisselen van modaliteiten om overbelaste cognitieve hulpbronnen te ontlasten, maak AI-onzekerheid zichtbaar in plaats van alleen uitkomsten, bied kalibratietraining aan voorafgaand aan inzet, en ontwerp functietoewijzing als dynamisch in plaats van statisch.

1

Introduction

1.1 Motivation

Rapid progress in computational power, transformer architectures, and large-scale foundation models has transformed what robots can perceive, reason about, and do (Firoozi et al., 2025). Large language models (LLMs), and their multimodal counterparts (MLLMs) as well as vision-language-action models (VLAs), now enable robots to interpret open-ended instructions, plan multi-step actions (e.g., Ahn et al., 2023; Driess et al., 2023), and generate motor commands from visual and linguistic input (Black et al., 2025). As a result, robots have moved from isolated industrial cells into spaces shared with humans: surgical systems assist in operating rooms (Morrell et al., 2021), automated vehicles navigate urban traffic (Baek et al., 2024), socially assistive companions support elder care (e.g., Abdi et al., 2018; Yen et al., 2024), and LLM-driven planners have begun to direct Mars rover traversals (Anthropic, 2026; NASA JPL, 2026). This trend is driven by the ability of robots to increase labor productivity (Graetz & Michaels, 2018), reduce work-related physical injuries (Gihleb et al., 2022), and mitigate labor shortages caused by aging populations (Acemoglu & Restrepo, 2022). Yet, technical capability alone does not guarantee effective collaboration with humans who must supervise, instruct, or work together with these systems.

Fitts (1951) documented that humans and machines excel at fundamentally different aspects of information processing, and that observation remains important: humans bring flexible, context-sensitive perception and judgment, while artificial systems offer speed, consistency, and tireless computation. Modern LLMs have partially narrowed this divide by demonstrating general in-context information-processing, yet their errors can sound fluent and convincing, leading users to over-rely on incorrect responses (Kim et al., 2025). When a robot takes over more decision-making, the human role changes from active operator to monitor, supervisor or collaborator (Sheridan & Verplank, 1978), often degrading situation awareness and the ability to intervene when failures occur (Endsley & Kiris, 1995).

At the same time, new communication modalities open up: humans may direct robots through speech, gesture, or high-level prompting (Goodrich & Schultz, 2007; Vemprala et al., 2024; Zhang et al., 2023), each of which may recruit different cognitive resources and create different opportunities for misunderstanding (Goodrich & Schultz, 2007). The result is that interaction failures increasingly arise not from what a robot cannot do, but from mismatches in how humans and robots perceive the environment, interpret cues, select actions, and respond, i.e., mismatches that degrade interaction quality. This thesis refers to such mismatches as ‘misalignment’.

As the four studies in this thesis will show, misalignment is not a monolithic phenomenon: it can, for example, take the form of differing visual encoding between human and AI (visual-spatial perceptual misalignment), reference-frame conflicts between operator and robot (spatial cognition-response misalignment), output that superficially appears coherent, but which masks shallow reasoning (verbal response misalignment), or divergent reasoning processes between human and AI that lead to ineffective distribution of cognitive resources (cognitive misalignment). When any of these forms of misalignment is present, the human may need to

invest more effort to sustain the interaction, leading to frustration, errors, and eventual disuse of the technology (Breazeal, 2003; Coronado et al., 2022; Parasuraman & Riley, 1997). The central question of this thesis is therefore: *where do the processes of humans and AI-driven robots align, where do they diverge, and how can interaction be designed so that those differences become complementary?*

1.2 Human and robot information processing

Human information-processing is traditionally modeled through stage-based frameworks, progressing from (1) sensation and perception of the environment, through (2) cognitive decision selection, to (3) response execution (e.g., Wickens, 1992). Artificial agents follow an analogous sequence: namely (1) sensory detection and machine perception, (2) planning and decision selection, and (3) action implementation (e.g., Parasuraman et al., 2000), whether using rule-based symbolic architectures or deep learning paradigms (Laird, 2012; Iosifidis & Tefas, 2022). The underlying assumption is that human and robot partners perceive the same environment through different sensors, select actions through different reasoning processes, and execute responses through different effectors.

To make this three-stage view more precise about where resources overlap between human and robot, this thesis adopts the vocabulary of Wickens' Multiple Resource Theory (MRT) (Wickens, 2002, 2008). The MRT characterizes human information processing not only along the aforementioned processing stages (perception, cognition, and responding) but also across modalities (visual vs. auditory for perception), and processing codes (spatial vs. verbal). Originally, the MRT was developed to predict human performance decrements when two tasks compete for the same resource pool, and to explain efficiency in multitasking (Wickens, 2002). In this thesis, the MRT terminology is used to describe how studies in the thesis are distributed across the information-processing stages and modalities.

Despite decades of relevant work on function allocation (Sheridan & Verplank, 1978; Parasuraman et al., 2000), mutual understanding in communication (Clark & Brennan, 1991), situation awareness (Endsley, 1995), trust calibration (Parasuraman & Riley, 1997), and more recently robot self-explanation (Wallkötter et al., 2021) and robot learning with multimodal feedback (Habibian et al., 2025), the field still lacks an integrated account of how human and robot align or misalign across the full information-processing chain (Xia et al., 2025). This thesis addresses that gap by examining where alignment breaks down and how interaction roles can be distributed for better alignment.

1.3 Research questions and thesis content

To address the central question posed in Section 1.1, this thesis investigates alignment across four studies that span different interaction modalities, robot platforms, and methods. The four studies progress from controlled benchmarking of non-embodied AI on isolated tasks, through laboratory teleoperation and multimodal dialogue with mobile and manipulator robots, to real-world educational deployment of human-AI cognitive collaboration.

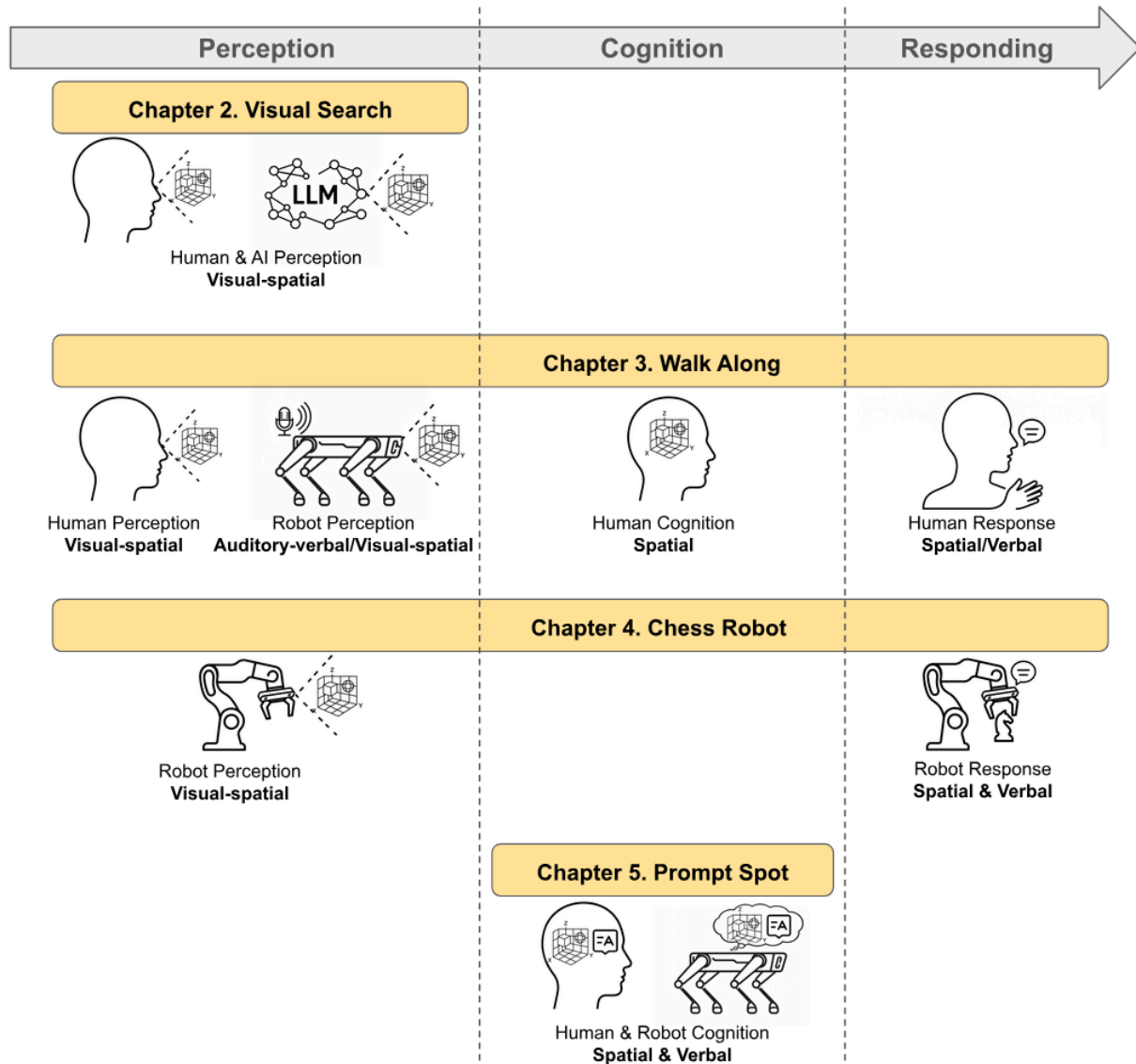


Figure 1.1. Mapping of the four thesis chapters onto the dimensions of Wickens' multiple resource theory (MRT) terminology. The yellow bars indicate which stages of information processing each chapter covers, while the icons highlight the specific components that are explicitly studied or developed in that chapter. The information-processing stages are: (1) perception, (2) cognition, and (3) responding. Furthermore, perception in this thesis can involve the visual or auditory modalities, and processing code can be spatial or verbal.

Research question 1 (Chapter 2): *Do MLLMs exhibit human-like attentional bottlenecks in visual search, and where does their processing diverge from human scanning strategy?*

Human-AI collaboration presupposes at least a partial correspondence between what each agent perceives. However, similar perceptual outcomes do not guarantee similar underlying processes. Chapter 2 isolates the perception stage of visual-spatial processing. Human visual processing operates using two channels: focal and ambient (Wickens, 2002). Like human ambient processing, MLLMs encode the full scene in a single parallel pass. The critical difference is that humans then deploy strategic focal fixations and continue searching serially until a quitting threshold is reached.

The chapter compares human and MLLM performance on identical visual search tasks that vary in difficulty, set size, and dimensionality (2D and 3D). Human and MLLM error patterns turn out to be strongly correlated overall, but diverge in complex conditions: humans trade speed for accuracy and continue searching via focal fixations until a quitting threshold is reached, whereas MLLMs, processing in a single ambient-like pass, tend to default to a binary present-or-absent judgment in difficult conjunction searches. This dissociation between outcome-level alignment and mechanism-level divergence is relevant to the overarching research question, because it defines the most elementary level at which misalignment can arise: two agents that appear to perceive the same scene may in fact fail under different conditions.

Research question 2 (Chapter 3): *How do control modality (voice vs. gestures) and user mobility (standing vs. walking) jointly affect task performance and perceived intuitiveness when directing a mobile robot?*

This chapter moves to the interaction itself: how the human communicates with a robot determines which cognitive resources are recruited, and thereby where misalignment may occur. The study spans all three processing stages: the operator monitors the robot visually (perception, visual-spatial), maintains a spatial model of the robot's pose and selects the next command (cognition, spatial), and issues commands through either speech (response, verbal) or gesture (response, spatial). On the robot side, a custom gesture recognition module was developed to receive these commands through vision (visual-spatial), while speech commands were captured through the HoloLens built-in speech recognition (auditory-verbal). Task performance therefore depends on two factors: whether the robot correctly detects the command (a technical recognition issue) and whether the command-to-movement mapping is intuitive for the operator (a human factors issue). For the human operator, gestures share the spatial processing already used for visual-spatial monitoring, whereas verbal-coded voice commands do not compete for the same resources (Wickens, 2002, 2008).

In a controlled experiment, 218 participants teleoperate the quadruped robot Spot using voice or gesture commands while either standing in a fixed position or walking alongside the robot. Voice

control combined with walking emerges as the most favored and intuitive condition, while gestures issued from a stationary position cause systematic left-right confusion due to spatial misalignment between operator and robot. Chapter 3 contributes to the overarching research question by demonstrating that misalignment is not just a property of the agents themselves but also of the interface: changing the command modality redistributes cognitive load across the operator's resources, with measurable consequences for spatial reasoning and task performance.

Research question 3 (Chapter 4): *Chapter 4 presents a design study with a research component: an open-source, reproducible robotic platform integrating visual perception, AI-generated commentary, and multimodal (verbal and gestural) interaction as a testbed for HRI. The research question it addressed is: how do the robot's physical task execution and verbal commentary jointly shape the robot's perceived usefulness among general and expert users?*

Chapters 2 and 3 each isolate specific information-processing stages, yet realistic collaboration engages perception, reasoning, and communication simultaneously. Investigating alignment across these dimensions requires a platform that is controlled enough for precise measurement yet sufficiently complex to reveal where coordination succeeds and where it fails. Chess satisfies both criteria: it is rule-bounded with objectively verifiable outcomes, and human and AI chess reasoning differ in well-documented ways (McCarthy, 1990), with humans relying on pattern recognition and positional intuition (Chase & Simon, 1973) while AI employs tree search with either hand-crafted (Shannon, 1950) or learned (Silver et al., 2018) evaluation.

Chapter 4 is primarily a design study with a research layer. As shown in Figure 1.1, the platform covers the perception and response stages on the robot side: the robot perceives the board visually and responds through both physical piece manipulation and LLM-generated spoken commentary. An international survey ($N = 597$) and an expert gameplay session reveal that the platform is valued in educational contexts for its capacity to make the robot's reasoning accessible through verbal explanations, while also exposing the limitations of current LLMs in providing the kind of strategic depth that expert players expect. The chapter contributes to the overarching research question by providing the research infrastructure needed to observe alignment and misalignment when multiple information-processing stages are engaged concurrently, and by revealing a concrete instance of verbal response misalignment: LLM-generated commentary that is fluent yet lacks strategic depth, illustrating how coherent output can mask shallow reasoning.

Research question 4 (Chapter 5): *Can a scalable, gamified educational workshop teach engineering students to recognize cognitive limitations in different LLM types and to redistribute reasoning effectively between themselves and the model across tasks of increasing complexity?*

Chapters 2 through 4 characterize where human and AI cognition align and misalign across perception, communication, and multimodal interaction. A recurring implication is that effective collaboration requires the human not only to interact with the AI but to recognize where its reasoning is limited and to compensate accordingly. This chapter addresses that cognitive dimension. As shown in Figure 1.1, the study is located at the cognition stage, engaging both spatial and verbal processing codes: students formulate verbal prompts that encode spatial navigation goals and evaluate the spatial plans that LLMs produce. In a series of workshops ($N = 226$), students prompt a non-reasoning and a reasoning LLM to solve grid-based navigation tasks of increasing complexity, with plans executed in simulation and demonstrated on a Boston Dynamics Spot robot. As tasks grow harder, students transition from providing step-by-step instructions to offering higher-level guidance such as waypoints and strategic heuristics, effectively learning to redistribute cognitive effort between themselves and the model based on each model's planning capacity. The chapter addresses the “complementary” component of the overarching research question: understanding the cognitive profile of an AI partner enables the human to structure the collaboration so that differences in reasoning become productive.

1.4 Summary

As robots powered by LLMs enter shared human spaces, misalignment becomes a risk: failures arising not from insufficient robot capability but from mismatches in how human and robot perceive, reason, and respond. This thesis includes four studies that span the space of processing stages, modalities, and codes. Chapter 2 isolates the perceptual level and reveals that strongly correlated error patterns between humans and MLLMs coexist with fundamentally different processing mechanisms, with error patterns diverging particularly in complex conditions, defining the most elementary form of misalignment. Chapter 3 spans all three information-processing stages and demonstrates that the operator's choice of command modality redistributes cognitive load across resources, identifying the interface as a locus of misalignment distinct from the agents themselves. Chapter 4 provides a multimodal chess robot platform in which alignment and misalignment become observable across perception and response simultaneously, and reveals a verbal response misalignment in which fluent LLM commentary obscures its shallow strategic understanding. Chapter 5 addresses the cognitive stage: engineering students learn, through prompted navigation tasks, where LLM reasoning diverges from their own and how to redistribute effort accordingly, turning cognitive differences into a collaborative asset. Together, these studies progress from isolated perception through embodied interaction to real-world educational deployment. Chapter 6 synthesizes findings across all four studies, revisits the research questions, and outlines directions for future work.

References

- Abdi, J., Al-Hindawi, A., Ng, T., & Vizcaychipi, M. P. (2018). Scoping review on the use of socially assistive robot technology in elderly care. *BMJ Open*, 8(2), Article e018815. <https://doi.org/10.1136/bmjopen-2017-018815>
- Acemoglu, D., & Restrepo, P. (2022). Demographics and automation. *The Review of Economic Studies*, 89(1), 1–44. <https://doi.org/10.1093/restud/rdab031>
- Ahn, M., Brohan, A., Brown, N., Chebotar, Y., Cortes, O., David, B., Finn, C., Fu, C., Gopalakrishnan, K., Hausman, K., Herzog, A., Ho, D., Hsu, J., Ibarz, J., Ichter, B., Irpan, A., Jang, E., Jauregui Ruano, R.,

- Jeffrey, K., ... Zeng, A. (2023). Do as I can, not as I say: Grounding language in robotic affordances. In K. Liu, D. Kulic, & J. Ichnowski (Eds.), *Proceedings of the 6th Conference on Robot Learning* (Vol. 205, pp. 287–318). PMLR. <https://proceedings.mlr.press/v205/ichter23a/ichter23a.pdf>
- Anthropic. (2026). *Claude AI powers NASA's first AI-planned Mars rover drive*. <https://www.anthropic.com/mars>
- Baek, D., Kim, S., Seo, S. W., & Lee, S. H. (2024). *Results and Lessons Learned from Autonomous Driving Transportation Services in Airfield, Crowded Indoor, and Urban Environments* [Preprint]. arXiv. <https://doi.org/10.48550/arXiv.2403.01233>
- Black, K., Brown, N., Driess, D., Esmail, A., Equi, M., Finn, C., Fusai, N., Groom, L., Hausman, K., Ichter, B., Jakubczak, S., Jones, T., Ke, L., Levine, S., Li-Bell, A., Mothukuri, M., Nair, S., Pertsch, K., Shi, L., ... Zhilinsky, U. (2025). π_0 : A vision-language-action flow model for general robot control. *Robotics: Science and Systems XXI*, Los Angeles, CA. <https://doi.org/10.15607/rss.2025.xxi.010>
- Breazeal, C. (2003). Emotion and sociable humanoid robots. *International Journal of Human-Computer Studies*, 59(1–2), 119–155. [https://doi.org/10.1016/S1071-5819\(03\)00018-1](https://doi.org/10.1016/S1071-5819(03)00018-1)
- Chase, W. G., & Simon, H. A. (1973). Perception in chess. *Cognitive Psychology*, 4(1), 55–81. [https://doi.org/10.1016/0010-0285\(73\)90004-2](https://doi.org/10.1016/0010-0285(73)90004-2)
- Clark, H. H., & Brennan, S. E. (1991). Grounding in communication. In L. B. Resnick, J. M. Levine, & S. D. Teasley (Eds.), *Perspectives on socially shared cognition* (pp. 127–149). American Psychological Association.
- Coronado, E., Kiyokawa, T., Ricardez, G. A. G., Ramirez-Alpizar, I. G., Venture, G., & Yamanobe, N. (2022). Evaluating quality in human-robot interaction: A systematic search and classification of performance and human-centered factors, measures and metrics towards an industry 5.0. *Journal of Manufacturing Systems*, 63, 392–410. <https://doi.org/10.1016/j.jmsy.2022.04.007>
- Driess, D., Xia, F., Sajjadi, M. S. M., Lynch, C., Chowdhery, A., Ichter, B., Wahid, A., Tompson, J., Vuong, Q., Yu, T., Huang, W., Chebotar, Y., Sermanet, P., Duckworth, D., Levine, S., Vanhoucke, V., Hausman, K., Toussaint, M., Greff, K., Zeng, A., Mordatch, I., & Florence, P. (2023). PaLM-E: An embodied multimodal language model. *Proceedings of the 40th International Conference on Machine Learning*, 202, 8469–8488. PMLR. <https://proceedings.mlr.press/v202/driess23a/driess23a.pdf>
- Endsley, M. R. (1995). Toward a theory of situation awareness in dynamic systems. *Human Factors*, 37(1), 32–64. <https://doi.org/10.1518/001872095779049543>
- Endsley, M. R., & Kiris, E. O. (1995). The out-of-the-loop performance problem and level of control in automation. *Human Factors*, 37(2), 381–394. <https://doi.org/10.1518/001872095779064555>
- Firoozi, R., Tucker, J., Tian, S., Majumdar, A., Sun, J., Liu, W., ... Schwager, M. (2025). Foundation models in robotics: Applications, challenges, and the future. *The International Journal of Robotics Research*, 44(5), 701–739. <https://doi.org/10.1177/02783649241281508>
- Fitts, P. M. (Ed.). (1951). *Human engineering for an effective air-navigation and traffic-control system*. National Research Council, Division of Anthropology and Psychology. <https://apps.dtic.mil/sti/tr/pdf/ADB815893.pdf>
- Gihleb, R., Giuntella, O., Stella, L., & Wang, T. (2022). Industrial robots, workers' safety, and health. *Labour Economics*, 78, Article 102205. <https://doi.org/10.1016/j.labeco.2022.102205>
- Goodrich, M. A., & Schultz, A. C. (2007). Human–robot interaction: A survey. *Foundations and Trends® in Human–Computer Interaction*, 1(3), 203–275. <https://dl.acm.org/doi/10.1561/11000000005>
- Graetz, G., & Michaels, G. (2018). Robots at work. *The Review of Economics and Statistics*, 100(5), 753–768. https://doi.org/10.1162/rest_a_00754
- Habibian, S., Alvarez Valdivia, A., Blumenschein, L. H., & Losey, D. P. (2025). A survey of communicating robot learning during human-robot interaction. *The International Journal of Robotics Research*, 44(4), 665–698. <https://doi.org/10.1177/02783649241281369>
- Iosifidis, A., & Tefas, A. (Eds.). (2022). *Deep learning for robot perception and cognition*. Academic Press. <https://doi.org/10.1016/c2020-0-02902-6>
- Kim, S. S. Y., Wortman Vaughan, J., Liao, Q. V., Lombrozo, T., & Russakovsky, O. (2025). Fostering appropriate reliance on large language models: The role of explanations, sources, and inconsistencies. In *Proceedings of the 2025 CHI Conference on Human Factors in Computing Systems* (Article 420). Association for Computing Machinery. <https://doi.org/10.1145/3706598.3714020>
- Laird, J. E. (2012). *The Soar cognitive architecture*. MIT Press. <https://doi.org/10.7551/mitpress/7688.001.0001>

- McCarthy, J. (1990). Chess as the Drosophila of AI. In T. A. Marsland & J. Schaeffer (Eds.), *Computers, chess, and cognition* (pp. 227–237). Springer. https://doi.org/10.1007/978-1-4613-9080-0_14
- Morrell, A. L. G., Morrell-Junior, A. C., Morrell, A. G., Mendes, J. M. F., Tustumi, F., De-Oliveira-e-Silva, L. G., & Morrell, A. (2021). The history of robotic surgery and its evolution: When illusion becomes reality. *Revista do Colégio Brasileiro de Cirurgiões*, 48, Article e20202798. <https://doi.org/10.1590/0100-6991e-20202798>
- NASA JPL. (2026). *NASA's Perseverance Rover completes first AI-planned drive on Mars* <https://www.jpl.nasa.gov/news/nasas-perseverance-rover-completes-first-ai-planned-drive-on-mars>
- Parasuraman, R., & Riley, V. (1997). Humans and automation: Use, misuse, disuse, abuse. *Human Factors*, 39(2), 230–253. <https://doi.org/10.1518/001872097778543886>
- Parasuraman, R., Sheridan, T. B., & Wickens, C. D. (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), 286–297. <https://doi.org/10.1109/3468.844354>
- Shannon, C. E. (1950). XXII. Programming a computer for playing chess. *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science*, 41(314), 256–275. <https://doi.org/10.1080/14786445008521796>
- Sheridan, T. B., & Verplank, W. L. (1978). *Human and computer control of undersea teleoperators*. Massachusetts Institute of Technology Man-Machine Systems Lab. <https://doi.org/10.21236/ada057655>
- Silver, D., Hubert, T., Schrittwieser, J., Antonoglou, I., Lai, M., Guez, A., ... & Hassabis, D. (2018). A general reinforcement learning algorithm that masters chess, shogi, and Go through self-play. *Science*, 362(6419), 1140–1144. <https://doi.org/10.1126/science.aar6404>
- Vemprala, S. H., Bonatti, R., Bucker, A., & Kapoor, A. (2024). ChatGPT for robotics: Design principles and model abilities. *IEEE Access*, 12, 55682–55696. <https://doi.org/10.1109/ACCESS.2024.3387941>
- Wallkötter, S., Tulli, S., Castellano, G., Paiva, A., & Chetouani, M. (2021). Explainable embodied agents through social cues: A review. *ACM Transactions on Human-Robot Interaction*, 10(3), Article 27. <https://dl.acm.org/doi/10.1145/3457188>
- Wickens, C. D. (1992). *Engineering psychology and human performance* (2nd ed.). HarperCollins Publishers.
- Wickens, C. D. (2002). Multiple resources and performance prediction. *Theoretical Issues in Ergonomics Science*, 3(2), 159–177. <https://doi.org/10.1080/14639220210123806>
- Wickens, C. D. (2008). Multiple resources and mental workload. *Human Factors*, 50(3), 449–455. <https://doi.org/10.1518/001872008X288394>
- Xia, G., Ghairi, Z., Wuest, T., Hribernik, K., Heuermann, A., Liu, F., ... Thoben, K. D. (2025). Towards human modeling for human-robot collaboration and digital twins in industrial environments: research status, prospects, and challenges. *Robotics and Computer-Integrated Manufacturing*, 95, Article 103043. <https://doi.org/10.1016/j.rcim.2025.103043>
- Yen, H. Y., Huang, C. W., Chiu, H. L., & Jin, G. (2024). The effect of social robots on depression and loneliness for older residents in long-term care facilities: A meta-analysis of randomized controlled trials. *Journal of the American Medical Directors Association*, 25(6), Article 104979. <https://doi.org/10.1016/j.jamda.2024.02.017>
- Zhang, C., Chen, J., Li, J., Peng, Y., & Mao, Z. (2023). Large language models for human–robot interaction: A review. *Biomimetic Intelligence and Robotics*, 3(4), Article 100131. <https://doi.org/10.1016/j.birob.2023.100131>

2

Human-like attention? A psychophysical comparison of visual search in humans and MLLMs

Intermezzo

The introduction set out the thesis's central question: where human and robot processing align, diverge, and can be made complementary. This chapter begins at the perception stage, the most elementary level at which misalignment can arise. Chapter 2 compares human and MLLM performance on identical visual search tasks, asking whether MLLMs exhibit human-like attentional bottlenecks and where their processing diverges from human scanning strategy (Research question 1).

Abstract

Visual search is a fundamental cognitive ability. This study investigates whether Multimodal Large Language Models (MLLMs) exhibit human-like attentional bottlenecks. We compared the performance of humans and several MLLMs on an identical visual search paradigm using synthetic 2D and 3D stimuli. The task involved finding a target (a yellow square in 2D or a yellow cube in 3D) across nine conditions, ranging from simple feature “pop-out” to complex conjunction searches, with set sizes varying from 4 to 256 items. Human performance was measured by response time and error rates, while MLLM performance was assessed via error rates. Results revealed a clear parallel: both humans and MLLMs performed efficiently in feature searches but showed substantial performance degradation in conjunction searches as set size increased. Across 18 conditions, a strong correlation was found between overall human and MLLM error rates ($\rho = 0.82$) and between overall human response times and MLLM error rates ($\rho = 0.91$), which suggests the models have implicitly learned principles of stimulus similarity and heterogeneity. However, while humans traded speed for accuracy, MLLMs showed non-human-like behavior in conjunction search, tending to default uniformly to either a “present” or “absent” response in complex scenes. This suggests that while MLLMs replicate core human search patterns, they do not search in a human-like manner. Our finding highlights the risk of perceptual misalignment between humans and MLLMs.

2.1 Introduction

The ability to extract relevant information from visual scenes is a key characteristic of intelligent perceptual systems (Krizhevsky et al., 2012; Marr, 2010). In cognitive psychology, visual search paradigms have long served as a tool for investigating how humans allocate attention. Visual search paradigms have revealed constraints on how humans encode features, select targets, and bind them into objects (Duncan & Humphreys, 1989; Treisman & Gelade, 1980; Wolfe, 2021).

Classic theories of visual search distinguish between two modes of visual search: efficient feature search versus effortful conjunction search. In feature search, a target defined by a unique basic property, such as a red item among green items, “pops out” rapidly. This results in response times (RTs) that are largely independent of set size (i.e., the total number of items in the display) (Treisman & Gelade, 1980). In contrast, conjunction search requires binding multiple features, for example, identifying a red vertical line among red horizontal and green vertical lines. This typically results in RTs that increase with set size. Target-absent trials yield RTs approximately twice as high as those for target-present trials, a pattern historically interpreted as evidence for a self-terminating serial scan (Treisman & Gelade, 1980; Wolfe, 1998). However, this interpretation has been debated, with some proposing that capacity-limited parallel models can also account for these effects (McElree & Carrasco, 1999; Thornton & Gilden, 2007).

Not all stimulus properties guide attention with equal efficiency. A limited set of features, including color, orientation, size, and curvature, are known to provide strong guidance (Wolfe &

Horowitz, 2017; Wolfe et al., 1992), while other features such as texture or material are less efficient (Wolfe, 2021; Wolfe & Myers, 2010). Building on these insights, contemporary models have evolved to integrate multiple interacting factors that shape attentional deployment. For example, Guided Search 6.0 (Wolfe, 2021) proposes that attentional guidance is the product of five factors that combine to create a priority map that directs attention where to focus next. These factors include bottom-up salience (how much an item stands out), top-down feature guidance (how well an item matches the search target), scene structure, search history, and the relative value of items (Wolfe & Horowitz, 2017).

The efficiency of this entire process is modulated by several key properties of the visual display. For example, the effectiveness of both bottom-up and top-down guidance is influenced by target-distractor similarity and distractor heterogeneity, where the more distractors resemble the target, the harder the search becomes (Calder-Travis & Ma, 2020; Duncan & Humphreys, 1989; Lleras et al., 2019). Similarly, top-down guidance is more efficient when a linear boundary in feature space can separate targets from distractors (Bauer et al., 1996; D’Zmura, 1991), though this rule has been debated (Xu et al., 2021). The limits of parallel processing channels are reflected in the fact that searches for conjunctions within a single feature dimension (e.g., color $\times$ color, such as searching for an item that is both red and green among items that are red-blue or green-blue) are typically less efficient than those across different dimensions (e.g., color $\times$ orientation) (Wolfe et al., 1990). Finally, the scene structure factor is directly affected by depth cues and 3D geometry, which can either aid search by segregating items (Enns & Rensink, 1990; Plewan & Rinkenauer, 2021) or hinder it via occlusion (Enns & Rensink, 1990). Textural depth cues, converging lines, and stereoscopic depth have also been found to affect search efficiency (Aks & Enns, 1996; Finlayson & Grove, 2015; O’Toole & Walker, 1997; Previc & Blume, 1993).

2.1.1 Visual processing and potential bottlenecks in MLLMs

Recent developments in artificial intelligence have led to the development of Multimodal Large Language Models (MLLMs), which extend traditional large language models by incorporating visual inputs (Alayrac et al., 2022; Liu et al., 2023). This integration enables the execution of sophisticated tasks that require joint vision-language understanding (Bordes et al., 2024; Yin et al., 2024).

MLLMs are trained on vast datasets of human-generated images and text, making them increasingly capable of tasks that traditionally required human visual expertise (Alayrac et al., 2022; Yue et al., 2024). Similarities in performance, such as efficient feature pop-out or serial-like bottlenecks in conjunction search, would suggest that large-scale pretraining induces cognitive-like processing in foundation models. This could reveal a connection between statistical learning and biological vision, and provide insights that allow us to predict MLLM performance based on human perceptual patterns. Conversely, differences in outcomes could indicate architectural limitations in MLLMs, such as the absence of serial attention or Gestalt grouping. Understanding these differences can inform efforts to align MLLMs more closely with human cognition, which is vital for real-world applications such as robotics and assistive technology.

We expected partial similarities between human and MLLM performance, due to the large-scale pretraining on human-curated image-text data. Recent work reports human-like attentional bottlenecks in vision-language models. In particular, Campbell et al. (2024) found near set-size-invariant performance for feature pop-out, together with systematic set-size costs for conjunction-like conditions, and numerosity/binding capacity limits that scale with distractor heterogeneity, paralleling human results. Yiu et al. (2025) reported that model error patterns correlate with children’s errors and adult response times across domains (easier for color/size; harder for rotation/reflection/number) in visual analogy tasks. Tangtartharakul and Storrs (2025) reported widespread deficits on neuropsychological and psychophysical tests involving multi-object binding and occlusion. In the same vein, Luo et al. (2025) showed that MLLMs, like humans, have difficulty in resolving cognitive conflict and show performance decrements in tasks that are similar to conjunction search.

We also expected differences between human and machine performance, given the largely parallel token-based processing by MLLMs, which lacks explicit saccadic or salience-driven mechanisms (Dosovitskiy et al., 2021). Indeed, architectural analyses caution against equating Transformer self-attention with human attention (Mehrani & Tsotsos, 2023; Zhao et al., 2024). Most MLLMs employ a pretrained visual encoder (e.g., Vision Transformer, ViT), which encodes an entire image simultaneously by dividing it into a grid of patches and comparing the contextual relationship of every patch to all others. These resulting image tokens are then processed alongside text tokens through the LLM’s self-attention mechanisms (Liu et al., 2023). Unlike human saccadic exploration, this architecture ingests the entire scene in parallel, often after downsampling to the ViT’s working resolution, which can erode fine-grained details (Zhang et al., 2024) and bypass the perceptual stages underlying human visual cognition (Huang et al., 2025). For example, Gemma 3 (Gemma Team, 2025) processes images with a vision encoder working at a fixed 896×896 resolution, and adaptively uses “Pan & Scan” to segment a high-resolution image and resize the crops for the encoder during inference.

Evidence confirms various search-like weaknesses: MLLMs tend to struggle with multi-object scenes involving occlusion, counting, or relational binding, often failing to leverage feature relationships or salience as humans do (Huang et al., 2025; Li, 2023). Similarly, testing of various deep neural network architectures, including ViTs, revealed only mixed evidence for human-like Gestalt grouping (Biscione & Bowers, 2023). While some convolutional networks showed sensitivity to principles such as proximity and linearity, these effects were only present at the final output layer, in contrast to human vision where such grouping is understood to be an intermediate process that organizes scenes for object recognition (Biscione & Bowers, 2023).

Iterative agents, such as SEAL/V* (Wu & Xie, 2024), simulate serial search by prompting models to zoom into sub-regions sequentially. While effective for detail-oriented tasks, these augmentations obscure whether core MLLM architectures inherently replicate human attentional constraints. Other work has proposed single-prompt interventions, such as adding simple visual scaffolds (e.g., horizontal lines) to the image and pairing them with a textual prompt that encourages sequential processing. This approach has been shown to improve visual search performance (Izadi et al., 2025). Recent work explored chained visual-textual reasoning to

achieve more human-like, stepwise visual focus, such as mimicking human-like attention selection (Gao et al., 2025) on a single image or iteratively restricting visual focus to relevant subsets when viewing multiple images (Zhang et al., 2025). However, an image-to-text and text-to-image cross-modality evaluation suggested that contemporary MLLMs have difficulty aligning the same entity across modalities to retrieve a requested attribute, even with chain-of-thought prompting, when scenes grow more complex (Alonso et al., 2025).

In contrast to intervention-based approaches, our study directly probes the baseline capabilities of MLLMs without augmentation. This allows us to assess whether these models guide search via feature values, relationships, or salience-driven strategies akin to human mechanisms (Becker et al., 2014; D’Zmura, 1991; Sobel & Cave, 2002), or if they rely on superficial patterns that break under heterogeneity or real-scene complexity (Wolfe et al., 2011).

2.1.2 The present study: a comparative psychophysical approach

We developed a unified paradigm, inspired by benchmarks such as VISFACTOR, which use psychometric tests to assess fundamental visual cognition (Huang et al., 2025), to evaluate visual search in humans and MLLMs. Using synthetic 2D and 3D scenes, we posed a binary question: *“Is there a solid yellow square (2D) or cube (3D)?”* Scenes featured varying set sizes (4–256 items) and target presence (50% present, 50% absent), with images generated to incorporate realistic depth cues such as occlusion, shadows, and perspective in 3D conditions.

We tested nine conditions spanning feature and conjunction searches. While human search efficiency is classically measured using response time (RT) and error rates, our MLLM evaluation is based solely on error rate. We hypothesized that these MLLM accuracy profiles would be analogous to human patterns: the model would show near-perfect accuracy in feature searches regardless of set size, but declining accuracy in conjunction searches as set sizes increase.

We further hypothesized that 3D scenes would pose a greater challenge than 2D scenes for both humans and MLLMs. The introduction of depth cues, perspective, shadows, and occlusion, where some items partially obscure others, is expected to increase processing demands. For humans, this requires additional effort to segregate and identify items, while for MLLMs, it tests their ability to deconstruct layered visual information. We therefore predicted that search efficiency would be lower, reflected in higher RTs and error rates, in 3D conditions compared to their 2D counterparts.

This work extends prior studies in several ways. First, we provide a comparison of humans and MLLMs on identical stimuli within a classic visual-search framework. This enables a direct comparison between human and MLLM search-efficiency on the same stimuli (including target-absent costs), which extends recent work that has compared MLLMs to established cognitive science findings in visual search (Campbell et al., 2024) or surveyed a wider range of visual deficits using broad test batteries (Tangtartharakul & Storrs, 2025). Second, we orthogonally manipulate distractor heterogeneity and 3D scene structure (occlusion, shading, perspective) and tested a wide set-size range (4 to 256), to allow us to isolate

binding/interference from depth-related factors. Third, we build a new benchmark involving three fundamental attributes (color, shape and material), based on human visual search theories, and evaluate multiple modern MLLMs with repeated queries to reduce response stochasticity. Fourth, we avoid agentic zoom-and-crop pipelines or reasoning MLLMs to examine the models’ capacity limits. These choices let us assess whether MLLMs exhibit human-like patterns with attribute binding or with a specific attribute, and whether those patterns generalize from 2D to 3D scenes.

2.2 Methods

2.2.1 Experimental conditions

We designed nine conditions around a fixed target, a yellow square (2D) or yellow cube (3D), to span classic “pop-out” feature search and effortful conjunction search. Table 2.1 provides an overview of the nine conditions, while Figure 2.1 provides example screenshots.

In the feature conditions (C1–C6), the distractors were arranged so the target is a singleton on one dimension (e.g., the only yellow among blues or the only square among circles), which should pop out via parallel preattentive guidance to that single feature. When adding heterogeneity in C5–C6 (mixing distractor colors or shapes), pop-out weakens because more items partially match the target template.

In the conjunction conditions (C7–C9), no item is unique on color or shape alone, so attention must bind color and shape (yellow + square/cube) at each candidate, presumably yielding slower, more serial inspection. We progressively increased the heterogeneity of the distractors from C7 to C9, which increases partial matches and the number of locations that must be checked.

Each condition was implemented with seven set sizes (4, 8, 16, 32, 64, 128, 256) and target presence (present vs absent). For conditions with multiple distractor types, we targeted equal proportions for all the distractor types. For every condition per set size per target presence configuration, we generated 25 versions of images by randomizing the object positions, resulting in a total of 6300 images (9 conditions $\times$ 2 presence/absence $\times$ 7 set sizes $\times$ 2 dimensionalities $\times$ 25 versions).

Based on these conditions, we generated separate 2D and 3D image datasets, which are detailed further in Appendix A. The 3D dataset was rendered in Blender (Blender Foundation, 2017), building upon the framework of the CLEVR dataset (Johnson et al., 2017).

Table 2.1. Experimental conditions

Condition	Label	Distractor composition*	Search type	Predicted search efficiency
C1	F_ColorShape	127 solid blue circles (spheres)	Redundant feature search. Target is unique in both color and shape.	Easy
C2	F_Color	127 solid blue squares (cubes)	The base version of a color feature search.	Easy
C3	F_Shape	127 solid yellow circles (spheres)	The base version of a shape feature search.	Medium
C4	F_Material	127 textured yellow squares (cubes)	The base version of a material feature search.	Medium
C5	F_Color_Het	64 solid blue squares (cubes) & 63 solid blue circles (spheres)	Color feature search with heterogeneous distractor shapes.	Easy
C6	F_Shape_Het	64 solid yellow circles (spheres) & 63 solid blue circles (spheres)	Shape feature search with heterogeneous distractor colors.	Medium
C7	C_ColorShape	64 solid yellow circles (spheres) & 63 solid blue squares (cubes)	The base version of a two-feature conjunction search.	Hard
C8	C_ColorShape_Het	43 solid blue squares (cubes), 42 solid yellow circles (spheres) & 42 solid blue circles (spheres)	The same as C7, but with added heterogeneous distractors (blue circles).	Hard
C9	C_ColorShapeMaterial	43 solid blue squares (cubes), 42 solid yellow circles (spheres) & 42 textured yellow squares (cubes)	A three-feature (triple) conjunction search	Very hard

* example composition with target present and set size = 128

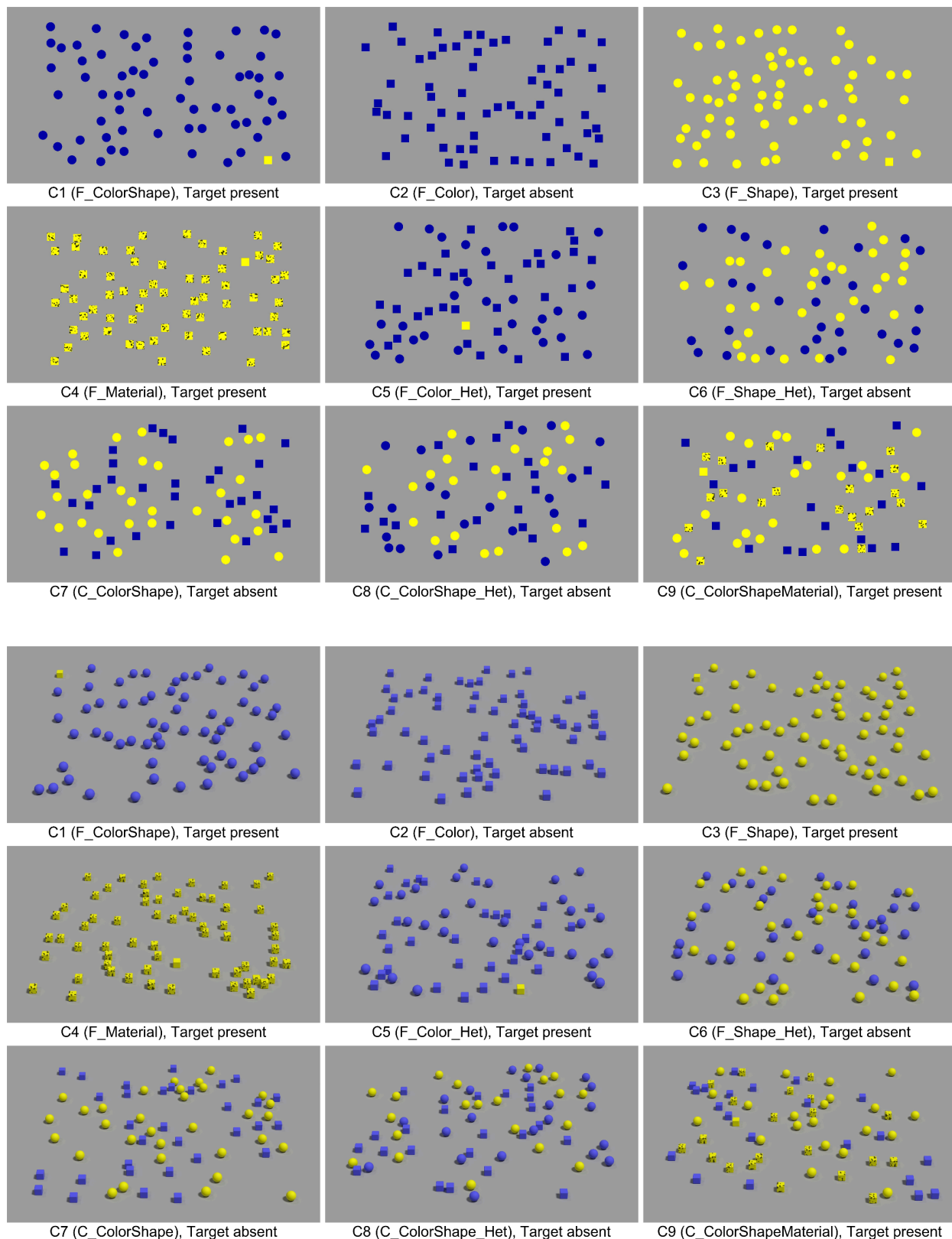


Figure 2.1. Images used in the practice block covering all 9 conditions. Top 9 figures; 2D conditions; Bottom 9 figures: 3D conditions.

2.2.2 MLLM evaluation

We used gemini-2.0-flash-lite (access date: 2025-08-07), gemini-2.5-flash-lite (access date: 2025-08-07), and gemma-3 (access date: 2025-08-09). Each image was queried 10 times at temperature = 0, using a fixed base prompt plus a minimal variation component, consisting of a number of period symbols, to produce distinct prompts and induce response diversity without altering the task instruction. The following is the full prompt given to each model:

*In this task, you will search for a solid yellow {block} without any texture on its face in an image.
{variations}*

Solid yellow {block}: Output 0 for absent or 1 for present.

The placeholder {block} was replaced by “square” or “cube” depending on the dimension of the image while the other placeholder {variations} was replaced by the variation component of dots. The correctness of responses from the model was recorded per image per repetition.

2.2.3 Human evaluation

A between-subjects design was implemented to assign each participant to one branch of a display dimension (2D or 3D). Within that branch, participants completed one practice block and nine experimental blocks (one per condition). Each block comprised 14 trials of the 14 configurations (7 set sizes × presence/absence). To ensure all image versions were viewed equally, each participant was randomly assigned a version number from 1 to 25. This number determined which of the 25 versions of each image the participant would see. Block order and trial order within blocks were randomized. Between each block, there was a short break.

The practice block consisted of nine trials at set size 64 (covering all conditions, with a mix of present/absent), which served as standardized familiarization and reduced learning effects before the experimental blocks. Figure 2.1 shows the nine images used in the practice block, for the 2D and 3D branches. Within each branch, all the conditions of the practice trial were displayed in an order that was the same for all participants.

In each trial, participants were first shown one of the following task instructions depending on the branch: “*Solid yellow square: press **A** for **absent** or **P** for **present***” or “*Solid yellow cube: press **A** for **absent** or **P** for **present***”.

Then a fixation cross was displayed with a duration of 1000 ms before the task image was presented. Participants indicated the target present/absent via keypress. Upon a valid keypress, the task image was removed and textual feedback of “Correct” in green or “Incorrect” in red appeared. Then the next trial started automatically by displaying the fixation cross. Accuracy and RT of each trial were recorded.

We recruited 1,250 participants via the Prolific academic research platform (Douglas et al., 2023; Peer et al., 2022). Participants were required to be from one of the following English-speaking countries to be eligible: Australia, Canada, Ireland, New Zealand, the United

Kingdom, or the United States. The experiment consisted of 50 questionnaire versions (2D/3D × 25 image versions), which meant each version was completed by 25 participants on average. The study launched on September 15, 2025, with the first participant starting at 19:17 CEST and the last participant starting at 20:51 CEST. Participants were compensated with £2.10 for their contribution. This study received ethical approval from the host institution (Reference Number: 4182), and informed consent was obtained from all respondents through a dedicated item at the beginning of the questionnaire.

The sample consisted of 50.6% female, 48.2% male, and 1.2% who identified as non-binary or other. The participants' mean age was 44.28 years (Median = 43, $SD = 13.41$), and they completed the experiment in an average of 14.10 minutes (Median = 12.22, $SD = 6.27$). Most participants (62.2%) completed the study on a laptop, while 30.3% used a desktop computer. A minority used a laptop connected to an external monitor or keyboard (7.2%) or a tablet (0.4%).

2.2.4 Analysis

All data from practice trials were excluded. A cell-wise procedure was applied to remove outlier RTs from the human experimental data. Trials were first grouped into cells based on a combination of three factors: stimulus dimension (2D or 3D), experimental condition, and set size. Within each of these cells, any trial with an RT less than 150 ms or greater than 20,000 ms was removed. Out of 157,500 experimental trials, 241 were unavailable due to an unspecified error. The outlier procedure excluded an additional 217 trials, resulting in a final dataset of 157,042 trials for analysis.

Several figures were generated to visualize and compare human and model performance:

- (1) Mean human RT as a function of set size, with error bars representing 95% confidence intervals.
- (2) The human error rate (in percent) as a function of set size.
- (3) Mean human RT for target-absent trials against the mean human RT for target-present trials, and the error rate for target-absent trials against the error rate for target-present trials. While the slope of the entire function measures search efficiency (the rate at which performance declines with added items), the present analysis focuses on the largest set size (256). This specific point was selected because it provided the clearest snapshot of performance limits and makes the absolute magnitude of the differences between conditions most apparent.

Next, two figures were generated to visualize the performance of the MLLMs, based on data averaged across three different MLLMs. This averaging approach was chosen to provide a more robust and generalizable measure of the capabilities of the MLLMs, minimizing the influence of model-specific quirks (Li et al., 2024; Lu et al., 2024).

- (4) A figure that mirrors the human error rate plot, showing the MLLM error rate.
- (5) A figure that is analogous to the human “Present vs. Absent” plot, showing the average MLLM error rate for absent trials versus present trials at the largest set size.

(6) Finally, a scatter plot figure is provided with a comparison of human and MLLM performance, averaged across all seven set sizes. We also calculated the Spearman rank correlation coefficient (ρ) for the 2D and 3D data points combined across the nine conditions to quantify the relationship between human and model error rates, and between human RT and model error rates.

2.3 Results

2.3.1 Human performance

Analysis of the search functions (Figure 2.2 for RT and Figure 2.3 for error rate) reveals clear differences in search efficiency across conditions. A parallel search pattern was observed for Conditions 1, 2, and 5, evidenced by nearly zero slopes. This outcome is consistent with a feature search, where the unique color allowed for efficient identification regardless of the number of distractors. Conversely, all other conditions exhibited serial search patterns, where RTs and error rates increased systematically with set size. Simple shape searches (C3, C6) were of relatively low difficulty. A material feature search (C4) and shape-color conjunction searches (C7, C8) were more difficult. Finally, the triple conjunction search (C9) produced the steepest slope.

Figure 2.4 reveals a speed-accuracy trade-off in human search performance. The left panel shows that for target-absent trials, participants were consistently slower, with all data points falling well above the diagonal unity line. However, the right panel shows that this additional time was used to ensure correctness; in these slow absent trials, participants were also more accurate. All data points in the error rate plot lie below the diagonal, indicating that the error rate was lower for target-absent than for target-present trials. The higher error rate in the target-present trials indicates that participants frequently terminated their search prematurely, causing them to miss the target and erroneously report that it was absent. Across both metrics, performance in 3D conditions (stars) was generally worse, involving longer RTs and higher error rates, than in their 2D counterparts (squares).

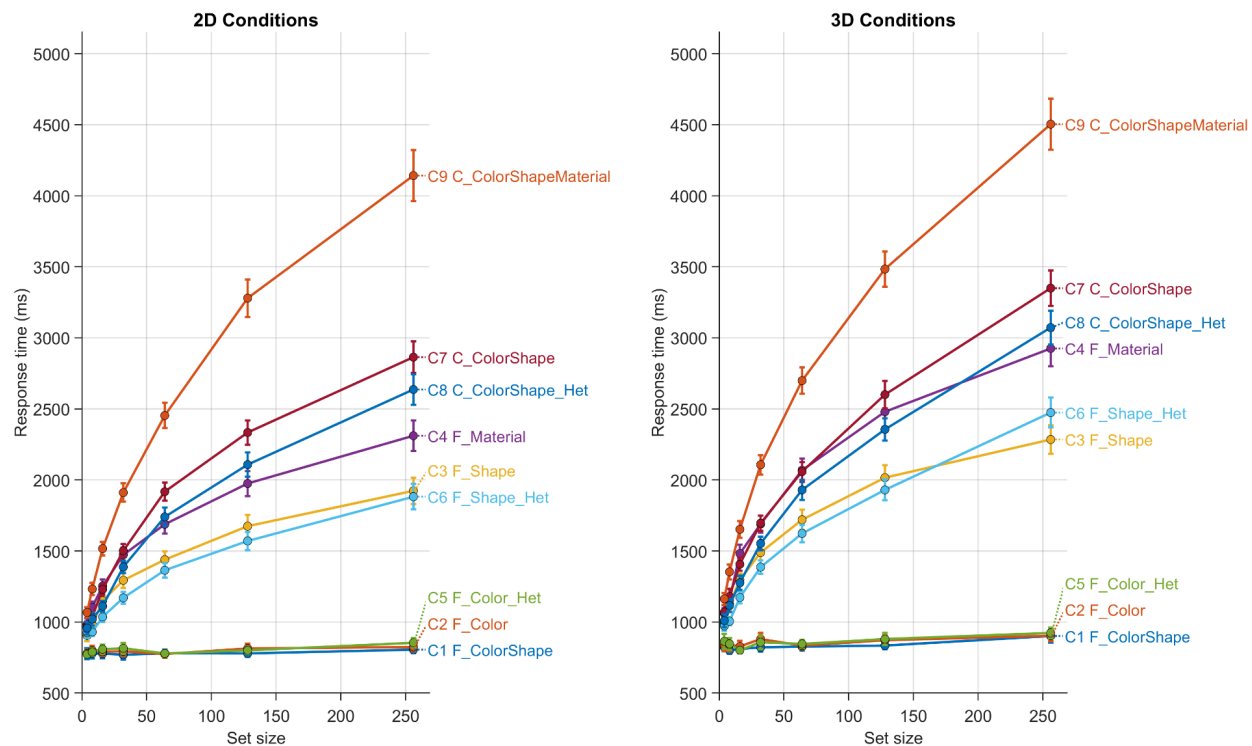


Figure 2.2. Mean human response time as a function of set size for the different experimental conditions. Left: 2D conditions, Right: 3D conditions.

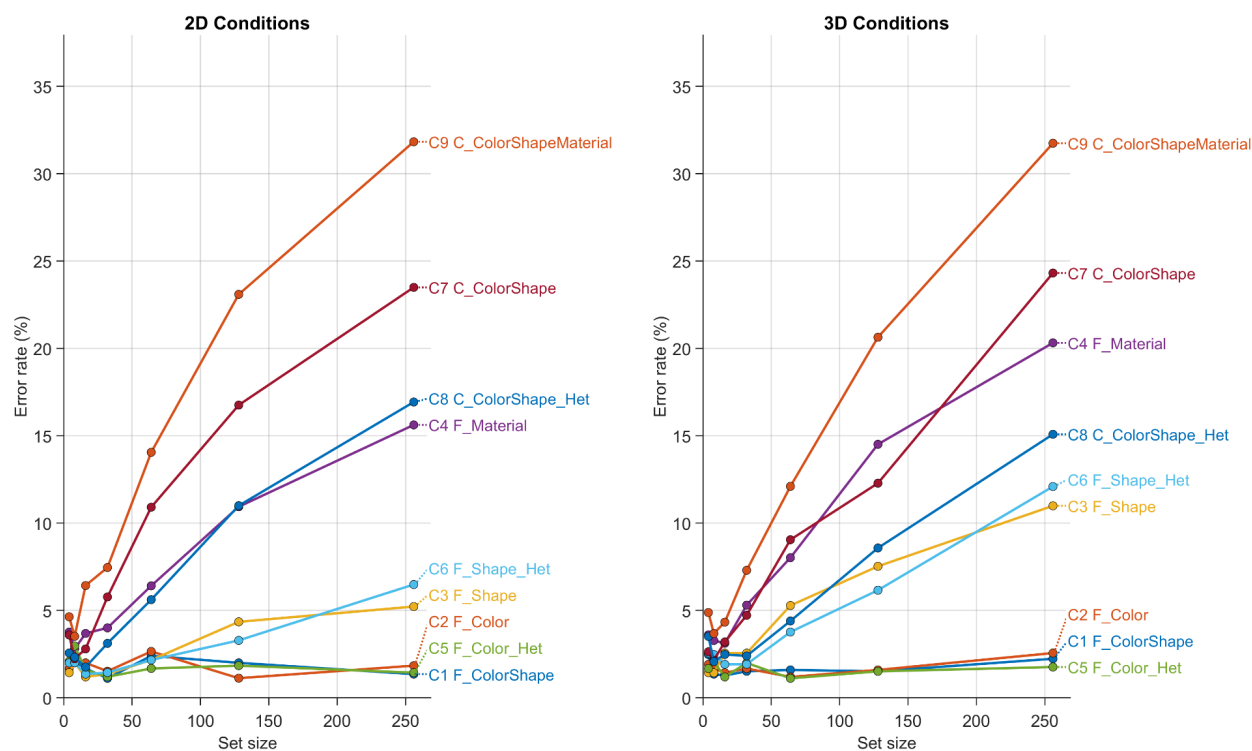


Figure 2.3. Mean human error rate as a function of set size for the different experimental conditions. Left: 2D conditions, Right: 3D conditions.

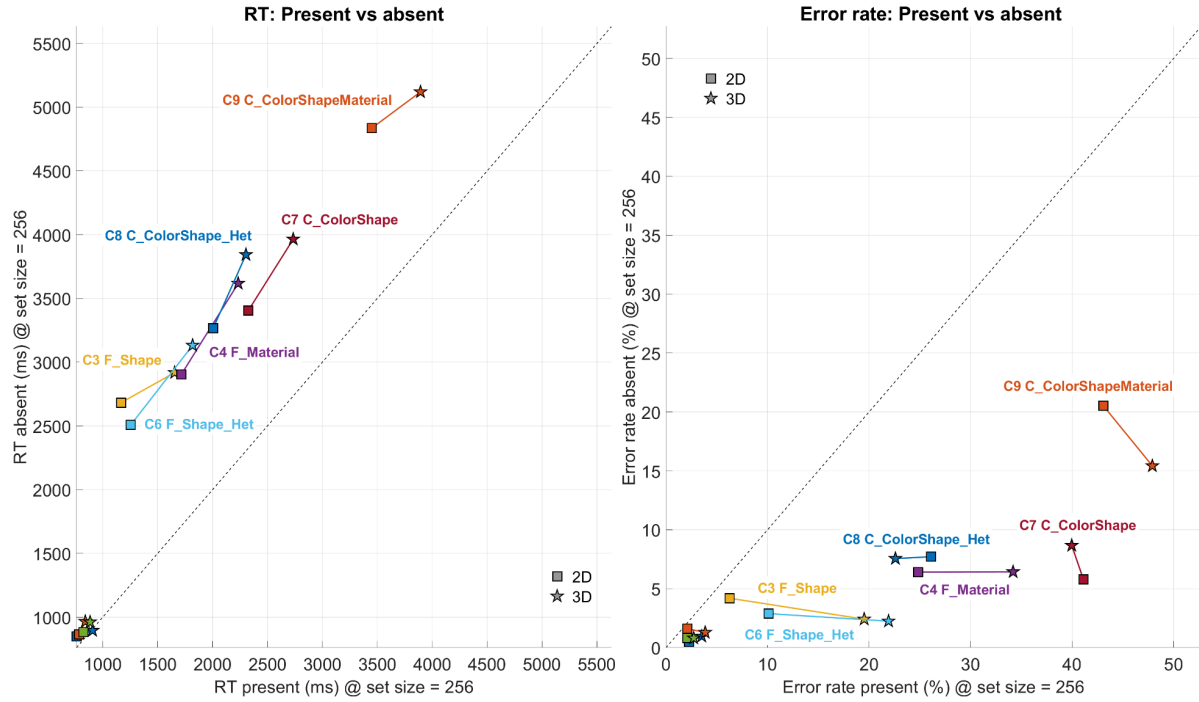


Figure 2.4. Mean human response time and human error rate for target-absent trials versus target-present trials, at the largest set size. A connecting line is shown between the 2D and 3D conditions. The diagonal line is the line of unity.

2.3.2 MLLM performance

To analyze the MLLMs' performance we treated error rate as analogous to human error rates, expecting it to increase with set size in difficult conditions. The results demonstrate a clear distinction between feature and conjunction search conditions that strongly parallels human performance.

For feature searches defined by a unique color (C1, C2, C5), all models maintained very low error rates (<5%) across all set sizes (Figure 2.5). This pattern mirrors the efficient pop-out search observed in humans. Feature searches based on shape (C3, C6) were also highly accurate, with a slight increase in errors at larger set sizes in the 3D condition. The material feature search (C4) was found to be more challenging than other feature conditions for all models, exhibiting a distinct positive slope where errors increased with set size. A clear performance degradation was observed for all conjunction conditions (C7, C8, C9) as set size increased, paralleling the effortful serial search in humans. The error rates for the most complex conditions with high distractor heterogeneity (C8 and C9) rose sharply, reaching 40–50% at the largest set sizes for all three models. This indicates a difficulty in binding multiple features (color, shape, and material) when processing complex scenes.

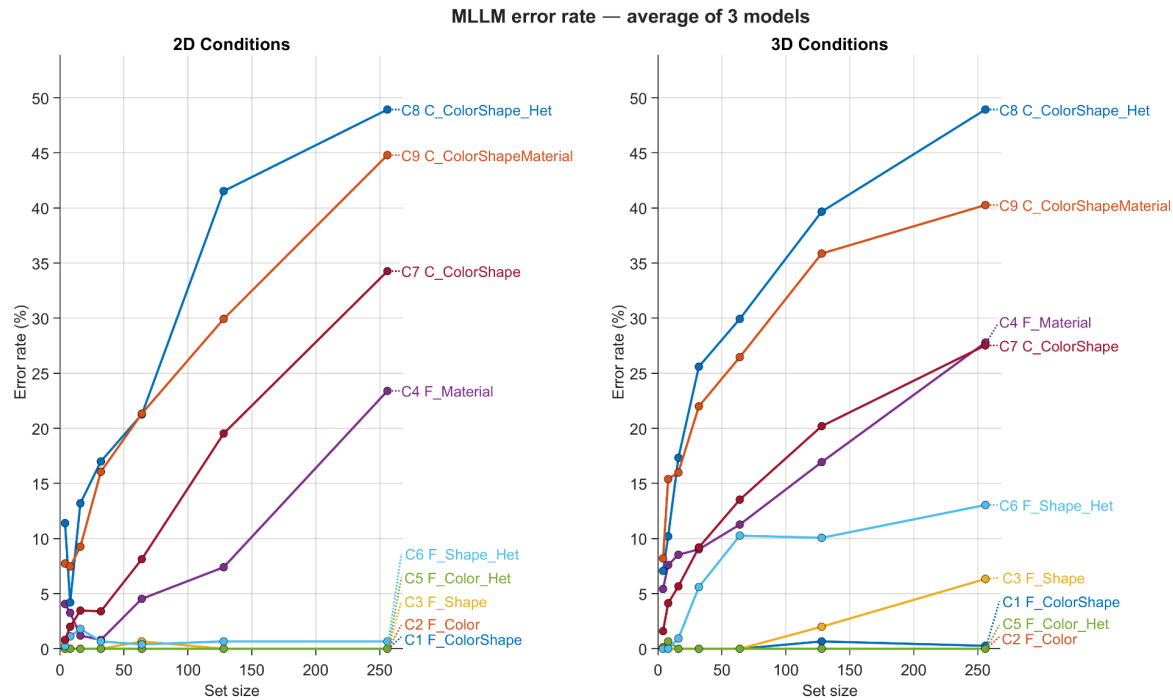


Figure 2.5. MLLM error rate as a function of set size for the different experimental conditions. Left: 2D conditions, Right: 3D conditions.

An analysis of error rates for target-absent versus target-present trials (Figure 2.6) revealed a non-human-like pattern of errors. For the simpler conjunction in C7, the MLLMs exhibited a classic target-absent cost: they were nearly perfect when the target was present but struggled to confirm its absence, with error rates above 50% at large set sizes. This mirrors the difficulty humans have with exhaustive searches. However, this pattern was reversed for the most complex triple-conjunction (C9) in 3D. Here, the MLLMs made frequent errors when the target was present (close to 80% error rate) but achieved near-perfect accuracy when the target was absent. This suggests that instead of performing an exhaustive search, the models may adopt a different strategy or bias, perhaps defaulting to an “absent” response when faced with overwhelming visual complexity in a 3D scene. The MLLMs showed the same defaulting tendency, but in the opposite direction in C8, the conjunction with three types of distractors, where the model almost always chose a “present” response.

Furthermore, 3D stimuli had a mixed effect on model performance. For some conditions, such as the material feature search (C4) and shape feature search (C3, C6), 3D scenes with depth and occlusion led to higher error rates than their 2D counterparts. However, this trend was not universal; for example, in the C7 conjunction search, the model made more errors in the 2D target-absent condition than in the 3D version at the largest set sizes, suggesting that the impact of dimensionality is context-dependent.

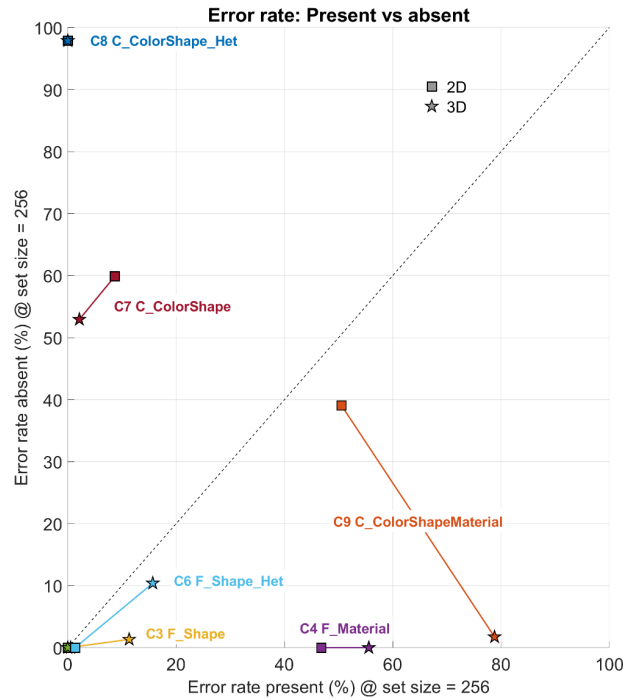


Figure 2.6. MLLM error rate for target-absent trials versus target-present trials, at the largest set size. A connecting line is shown between the 2D and 3D conditions. The diagonal line is the line of unity.

2.3.3 MLLMs vs humans

To quantify the relationship between human and MLLM search performance, we analyzed the correlation between MLLM and human performance. Figure 2.7 reveals a strong positive correlation: $p = 0.91$ for the human RT versus MLLM error rates (left subplot), and $p = 0.82$ for the human error rates versus MLLM error rates (right subplot) ($n = 18$ conditions). This indicates a remarkable alignment in what makes a visual search task difficult for both humans and AI. Conditions that were easy for humans (e.g., feature pop-outs C1, C2, C5), were also easy for the MLLMs. Conversely, conditions that demanded a serial-like search from humans (e.g., complex conjunctions C8 and C9), were the ones that caused MLLM error rates to increase most dramatically with set size.

Feature-wise patterns mirror these global trends: color-based feature search is efficient for both humans and models, with near-zero error rates and very low RTs in both 2D and 3D. Shape-based feature search is moderately harder: for humans, 2D shape search is noticeably slower and more error-prone than color search, and this difficulty increases further in 3D. By contrast, material is the most challenging feature for both humans and MLLMs in 2D and 3D alike, consistent with prior evidence that material search is inefficient (Wolfe & Myers, 2010).

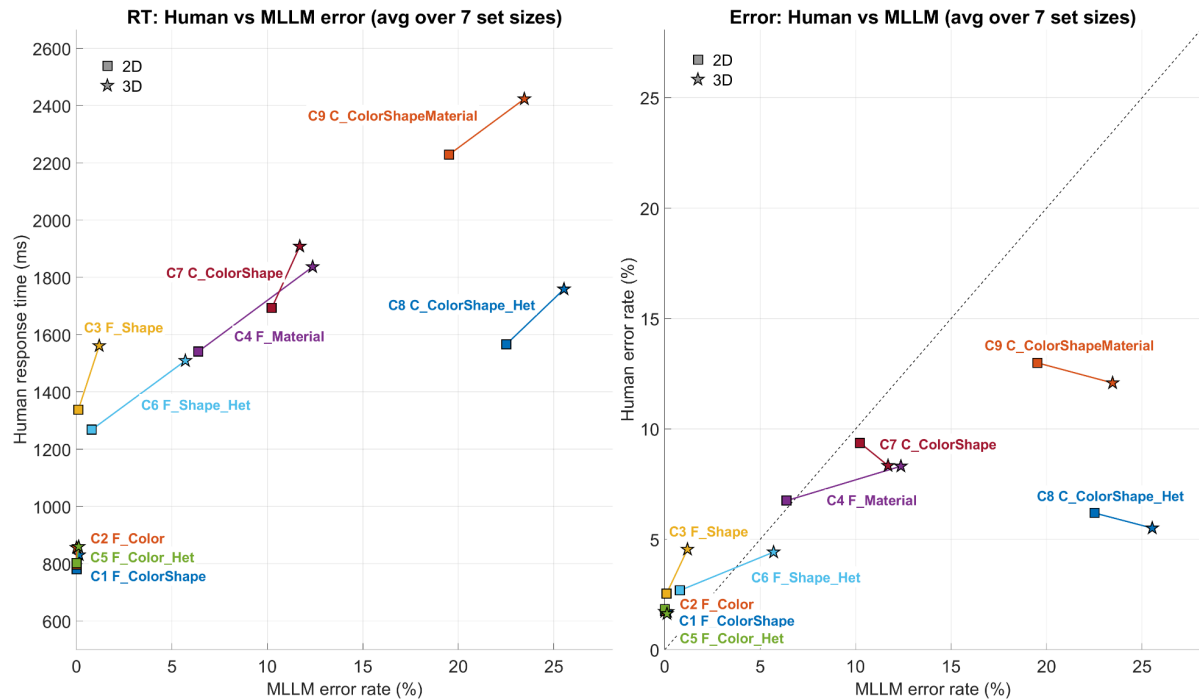


Figure 2.7. Left: Human response time versus MLLM error rate, averaged across all set sizes. Right: Human error rate versus MLLM error rate, averaged across all set sizes. A connecting line is shown between the 2D and 3D conditions. The diagonal line in the right figure is the line of unity.

2.4 Discussion

The present study was designed to compare the visual search performance of human observers and non-reasoning MLLMs. By employing an identical set of synthetic stimuli for both groups, we aimed to determine whether MLLMs exhibit cognitive bottlenecks analogous to those well-documented in human visual attention, particularly the distinction between efficient feature search and inefficient conjunction search. We hypothesized that MLLM error rates would serve as a functional analogue to human error rates, remaining low and flat for feature pop-out tasks but increasing with set size for conjunction searches, with these effects being modulated by distractor characteristics and 2D versus 3D scenes. Our central finding indicates a striking parallel: for both humans and MLLMs, performance remained stable in simple feature searches but degraded systematically as set size increased in complex conjunction searches.

2.4.1 Parallels in human and MLLM performance

For human participants, the data replicated decades of visual search research. Feature searches defined by a unique color (Conditions C1, C2, C5) were highly efficient, with nearly flat RT slopes, demonstrating the classic pop-out effect where the target is detected in parallel (Treisman & Gelade, 1980). In contrast, searches requiring the binding of features (C7–C9) or the discrimination of less salient features such as shape (C3, C6) and material (C4) produced steep RT slopes, indicative of an effortful capacity-limited process. The addition of 3D depth cues generally increased search times and steepened slopes, consistent with findings that cast shadows, when discounted by early vision, make shape differences harder to detect and slow

search (Rensink & Cavanagh, 2004). The 3D cues make search harder by introducing ‘crowding’ and limiting the size of the Functional Visual Field (FVF), the area where an item can be successfully identified in a single fixation (Wolfe, 2021).

Our findings also extend the understanding of search in three-dimensional space. The fact that MLLMs found 3D scenes more challenging suggests their internal representations capture some of the complexity introduced by occlusion, perspective, and shadows, though their failure mode in the C9-3D condition indicates these representations may be fragile. This aligns with findings that MLLMs have deficits in tasks involving occlusion and 3D reasoning (Tangtartharakul & Storrs, 2025).

The performance of the MLLMs paralleled these human patterns with remarkable fidelity. In feature pop-out conditions, MLLM accuracy remained high and stable regardless of set size. In conjunction conditions, accuracy declined as set size increased, with the steepest declines observed in the most heterogeneous conditions (C8, C9). This alignment was confirmed by strong positive correlations between human performance and the MLLM error rates. This suggests that, with the current foundation models’ architecture and pre-training on vast datasets of visual and textual information, MLLMs have implicitly learned the same principles of stimulus similarity and heterogeneity that govern human attentional guidance (Duncan & Humphreys, 1989; Wolfe, 2021). The models’ difficulty with conjunctions also echoes recent findings that MLLMs struggle with the binding problem (Alonso et al., 2025; Campbell et al., 2024; Tangtartharakul & Storrs, 2025; Yiu et al., 2025), a core challenge for human attention.

2.4.2 Differences between human and MLLM performance

However, our results also revealed a divergence between human and machine performance, particularly in how they handle the absence or presence of a target. Humans demonstrated a speed-accuracy trade-off, with search RTs for absent trials being roughly twice as high, yet error rates being much lower compared to present trials. The MLLMs showed an entirely opposite error-rate pattern in some of the conjunction searches (C7 and C8), with a high error rate for target-absent trials and a low error rate for target-present trials. Conversely, in the most complex condition (C9) in 3D, the models performed perfectly on target-absent trials but made numerous errors when the target was present. This suggests a breakdown in the models’ ability to conduct an exhaustive search and a lack of associated speed-accuracy trade-off. Instead of thoroughly examining the display, the models appear to adopt a non-human-like strategy, perhaps defaulting to a “present” or “absent” response when faced with visual complexity.

This pattern of human search can be explained by the Guided Search (GS) 6.0 model of an adaptive ‘quitting threshold’ (Wolfe, 2021). Humans search until an internal quitting signal reaches this threshold; they take longer on absent trials to be confident the target is not there. The MLLMs’ erratic error patterns reveal they lack such a mechanism. When the priority map offers no clear guidance, their process appears to break down, causing the MLLMs to default to a biased response rather than completing a strategically terminated exhaustive search. This failure to bind features aligns with recent work identifying the binding problem as a core

limitation of current vision-language architectures (Alonso et al., 2025; Campbell et al., 2024; Tangtartharakul & Storrs, 2025; Yiu et al., 2025).

The strategic mechanism in human search and its absence in the MLLMs is most clearly observed in the performance difference on the C8 condition (a yellow square among yellow circles, blue squares and blue circles). There is a reversal in difficulty: for humans C8 was easier than C7 (Figure 2.2 & 2.3), whereas for MLLMs, it had the highest error rate among all conjunction search conditions (Figure 2.5). It is likely that human participants have a two-template process as described in the GS 6.0 model (Wolfe, 2021). They first use a ‘Guiding Template’ based on the most efficient feature, color in this case, to filter out the blue distractors, reducing the practical set size to only one-third of its original size. Then a ‘Target Template’ is used to serially verify the shape of the remaining candidates. In contrast, the MLLMs’ parallel architecture applied a single fused template based on the tokenized image and prompt text in one pass.

The graded difficulty we observed across our conjunction conditions supports human perception models that view search efficiency as a continuum rather than a simple serial-parallel dichotomy. Our findings are well-explained by frameworks such as GS (Wolfe, 2021; Wolfe et al., 1989), where distractor heterogeneity reduces the effectiveness of guidance. Specifically, guidance is strongest when featural differences between the target and distractors are large and differences among distractors are small (Duncan & Humphreys, 1989). Our C8 and C9 conditions, which increased distractor heterogeneity, systematically weakened guidance by violating these principles. The results also align with the Distractor Rejection Cost model (Xu et al., 2021), which posits that the cost of rejecting distractors is modulated by both target-distractor similarity and distractor-distractor interactions.

2.4.3 Limitations

Several limitations should be considered. First, our human data were collected online, which introduces variability in viewing conditions and participant engagement compared to a controlled laboratory setting. However, the clear replication of classic visual search effects suggests the data are robust. Second, our evaluation of MLLMs relied on error rate as a proxy for cognitive effort, as RT is not a meaningful measure for these models. While the strong correlation with human RTs supports this approach, it is an indirect measure of the underlying process. Third, our stimuli, while more complex than simple letters, were still synthetic. Search in natural, cluttered scenes involves additional guiding factors, such as scene grammar and object co-occurrence, that were not tested here (Wolfe & Horowitz, 2017). Finally, our study examined the baseline capabilities of MLLMs. We did not employ iterative prompting, visual scaffolding techniques, or reasoning MLLMs that have been shown to improve MLLM performance on search-like tasks (Izadi et al., 2025; Wu & Xie, 2024). In addition, our tests were limited to three models, all produced by the same developer. The behavior of MLLMs may change over time and models developed by other authors may perform differently.

2.4.4 Conclusion and future directions

In conclusion, our comparative study demonstrates that MLLMs have implicitly learned many of the fundamental principles that govern human visual search. Their performance is systematically affected by target-distractor similarity, distractor heterogeneity, and the feature-conjunction distinction in a manner that closely mirrors human RT data. This suggests that the statistical patterns in large-scale visual-language datasets are sufficient to instill human-like sensitivities to the factors that guide attention. However, a difference appears in the most complex search scenarios, particularly in target-absent decisions, revealing a potential weakness in the models' ability to handle combined feature binding and high distractor load.

Future research should aim to bridge this gap. One promising avenue is to compare human fixation patterns directly with the internal attention maps of MLLMs using the same stimuli to see if the models' "attention" corresponds to human gaze. Further investigation is also needed to understand the MLLMs' failure in complex target-absent conditions. Is this a failure of binding, a premature quitting strategy, or an artifact of the models' training? Finally, extending this comparative paradigm to include other known modulators of human search, such as scene context, search history, and reward (Wolfe & Horowitz, 2017), will be essential for developing a more complete understanding of the similarities and differences between biological and artificial visual intelligence.

References

- Aks, D. J., & Enns, J. T. (1996). Visual search for size is influenced by a background texture gradient. *Journal of Experimental Psychology: Human Perception and Performance*, 22(6), 1467–1481. <https://doi.org/10.1037/0096-1523.22.6.1467>
- Alayrac, J. B., Donahue, J., Luc, P., Miech, A., Barr, I., Hasson, Y., Lenc, K., Mensch, A., Millican, K., Reynolds, M., Ring, R., Rutherford, E., Cabi, S., Han, T., Gong, Z., Samangooei, S., Monteiro, M., Menick, J. L., Borgeaud, S., ... Simonyan, K. (2022). Flamingo: A visual language model for few-shot learning. In S. Koyejo, S. Mohamed, A. Agarwal, D. Belgrave, K. Cho, & A. Oh (Eds.), *Advances in Neural Information Processing Systems* (Vol. 35, pp. 23716–23736). Curran Associates. https://proceedings.neurips.cc/paper_files/paper/2022/file/960a172bc7fbf0177ccccbb411a7d800-Paper-Conference.pdf
- Alonso, I., Azkune, G., Salaberria, A., Barnes, J., & Lopez de Lacalle, O. (2025). Vision-language models struggle to align entities across modalities. In W. Che, J. Nabende, E. Shutova, & M. T. Pilehvar (Eds.), *Findings of the Association for Computational Linguistics: ACL 2025* (pp. 18846–18862). Association for Computational Linguistics. <https://doi.org/10.18653/v1/2025.findings-acl.965>
- Bauer, B., Jolicoeur, P., & Cowan, W. B. (1996). Visual search for colour targets that are or are not linearly separable from distractors. *Vision Research*, 36(10), 1439–1466. [https://doi.org/10.1016/0042-6989\(95\)00207-3](https://doi.org/10.1016/0042-6989(95)00207-3)
- Becker, S. I., Harris, A. M., Venini, D., & Retell, J. D. (2014). Visual search for color and shape: When is the gaze guided by feature relationships, when by feature values? *Journal of Experimental Psychology: Human Perception and Performance*, 40(1), 264–291. <https://doi.org/10.1037/a0033489>
- Biscione, V., & Bowers, J. S. (2023). Mixed evidence for Gestalt grouping in deep neural networks. *Computational Brain & Behavior*, 6(3), 438–456. <https://doi.org/10.1007/s42113-023-00169-2>
- Blender Foundation. (2017). Blender [Computer software]. <https://www.blender.org/download/releases/2-78/>
- Bordes, F., Pang, R. Y., Ajay, A., Li, A. C., Bardes, A., Petryk, S., Mañas, O., Lin, Z., Mahmoud, A., Jayaraman, B., Ibrahim, M., Hall, M., Xiong, Y., Lebensold, J., Ross, C., Jayakumar, S., Guo, C., Bouchacourt, D., Al-Tahan, H., ... Chandra, V. (2024). *An introduction to vision-language modeling* [Preprint]. arXiv. <https://doi.org/10.48550/arXiv.2405.17247>

- Calder-Travis, J., & Ma, W. J. (2020). Explaining the effects of distractor statistics in visual search. *Journal of Vision*, 20(13), Article 11. <https://doi.org/10.1167/jov.20.13.11>
- Campbell, D., Rane, S., Giallanza, T., De Sabbata, C. N., Ghods, K., Joshi, A., Ku, A., Frankland, S. M., Griffiths, T. L., Cohen, J. D., & Webb, T. (2024). Understanding the limits of vision language models through the lens of the binding problem. In A. Globerson, L. Mackey, D. Belgrave, A. Fan, U. Paquet, J. Tomczak, & C. Zhang (Eds.), *Advances in Neural Information Processing Systems* (Vol. 37, pp. 113436–113460). Curran Associates.
https://proceedings.neurips.cc/paper_files/paper/2024/file/cdcc6d47c1627350014a3076112ab824-Paper-Conference.pdf
- Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., Uszkoreit, J., & Hounsby, N. (2021). An image is worth 16x16 words: Transformers for image recognition at scale. *Proceedings of the International Conference on Learning Representations*, Vienna, Austria. <https://openreview.net/pdf?id=YicbFdNTTy>
- Douglas, B. D., Ewell, P. J., & Brauer, M. (2023). Data quality in online human-subjects research: Comparisons between MTurk, Prolific, CloudResearch, Qualtrics, and SONA. *PLOS ONE*, 18(3), Article e0279720. <https://doi.org/10.1371/journal.pone.0279720>
- Duncan, J., & Humphreys, G. W. (1989). Visual search and stimulus similarity. *Psychological Review*, 96(3), 433–458. <https://doi.org/10.1037/0033-295X.96.3.433>
- D’Zmura, M. (1991). Color in visual search. *Vision Research*, 31(6), 951–966.
[https://doi.org/10.1016/0042-6989\(91\)90203-H](https://doi.org/10.1016/0042-6989(91)90203-H)
- Enns, J. T., & Rensink, R. A. (1990). Sensitivity to three-dimensional orientation in visual search. *Psychological Science*, 1(5), 323–326. <https://doi.org/10.1111/j.1467-9280.1990.tb00227.x>
- Finlayson, N. J., & Grove, P. M. (2015). Visual search is influenced by 3D spatial layout. *Attention, Perception, & Psychophysics*, 77(7), 2322–2330. <https://doi.org/10.3758/s13414-015-0924-3>
- Gao, J., Li, Y., Cao, Z., & Li, W. (2025). Interleaved-modal chain-of-thought. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, Nashville, TN, 19520–19529.
<https://doi.org/10.1109/CVPR52734.2025.01818>
- Gemma Team. (2025). *Gemma 3 technical report* [Preprint]. arXiv.
<https://doi.org/10.48550/arXiv.2503.19786>
- Huang, J.-T., Dai, D., Huang, J.-Y., Yuan, Y., Liu, X., Wang, W., Jiao, W., He, P., Tu, Z., & Duan, H. (2025). *Human cognitive benchmarks reveal foundational visual gaps in MLLMs* [Preprint]. arXiv.
<https://doi.org/10.48550/arXiv.2502.16435>
- Izadi, A., Banayeeanzade, M. A., Askari, F., Rahimiakbar, A., Vahedi, M. M., Hasani, H., & Baghshah, M. S. (2025). *Visual structures helps visual reasoning: Addressing the binding problem in VLMs* [Preprint]. arXiv. <https://doi.org/10.48550/arXiv.2506.22146>
- Johnson, J., Hariharan, B., van der Maaten, L., Fei-Fei, L., Zitnick, C. L., & Girshick, R. (2017). CLEVR: A diagnostic dataset for compositional language and elementary visual reasoning. *Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition*, Honolulu, HI, 1988–1997.
<https://doi.org/10.1109/CVPR.2017.215>
- Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012). ImageNet classification with deep convolutional neural networks. In F. Pereira, C. J. Burges, L. Bottou, & K. Q. Weinberger (Eds.), *Advances in Neural Information Processing Systems* (Vol. 25, pp. 1097–1105). Curran Associates.
<https://proceedings.neurips.cc/paper/2012/file/c399862d3b9d6b76c8436e924a68c45b-Paper.pdf>
- Li, J. (2023). Compositional reasoning in vision-language models [Undergraduate honors thesis, Brown University].
https://cs.brown.edu/media/filer_public/84/e0/84e05bd1-6d4a-4690-9fe3-af782aab013f/lijessica_honors_thesis.pdf
- Li, J., Zhang, Q., Yu, Y., Fu, Q., & Ye, D. (2024). More agents is all you need. *Transactions on Machine Learning Research*. <https://openreview.net/forum?id=bgzUSZ8aeg>
- Liu, H., Li, C., Wu, Q., & Lee, Y. J. (2023). Visual instruction tuning. In A. Oh, T. Naumann, A. Globerson, K. Saenko, M. Hardt, & S. Levine (Eds.), *Advances in Neural Information Processing Systems* (Vol. 36, pp. 34892–34916). Curran Associates.
https://proceedings.neurips.cc/paper_files/paper/2023/file/6dcf277ea32ce3288914faf369fe6de0-Paper-Conference.pdf

- Lleras, A., Wang, Z., Madison, A., & Buetti, S. (2019). Predicting search performance in heterogeneous scenes: Quantifying the impact of homogeneity effects in efficient search. *Collabra: Psychology*, 5(1), Article 2. <https://doi.org/10.1525/collabra.151>
- Lu, X., Liu, Z., Liusie, A., Raina, V., Mudupalli, V., Zhang, Y., & Beauchamp, W. (2024). *Blending is all you need: Cheaper, better alternative to trillion-parameters LLM* [Preprint]. arXiv. <https://doi.org/10.48550/arXiv.2401.02994>
- Luo, D., Wang, M., Wang, B., Zhao, T., Li, Y., & Deng, H. (2025). Machine psychophysics: Cognitive control in vision-language models. *Proceedings of the 8th Annual Conference on Cognitive Computational Neuroscience*, Amsterdam, The Netherlands. https://2025.ccneuro.org/abstract_pdf/Luo_2025_Measuring_Cognitive_Control_Vision-Language_Models.pdf
- Marr, D. (2010). *Vision: A computational investigation into the human representation and processing of visual information*. MIT Press.
- McElree, B., & Carrasco, M. (1999). The temporal dynamics of visual search: Evidence for parallel processing in feature and conjunction searches. *Journal of Experimental Psychology: Human Perception and Performance*, 25(6), 1517–1539. <https://doi.org/10.1037/0096-1523.25.6.1517>
- Mehrani, P., & Tsotsos, J. K. (2023). Self-attention in vision transformers performs perceptual grouping, not attention. *Frontiers in Computer Science*, 5, Article 1178450. <https://doi.org/10.3389/fcomp.2023.1178450>
- O'Toole, A. J., & Walker, C. L. (1997). On the preattentive accessibility of stereoscopic disparity: Evidence from visual search. *Perception & Psychophysics*, 59(2), 202–218. <https://doi.org/10.3758/BF03211889>
- Peer, E., Rothschild, D., Gordon, A., Evernden, Z., & Damer, E. (2022). Data quality of platforms and panels for online behavioral research. *Behavior Research Methods*, 54(4), 1643–1662. <https://doi.org/10.3758/s13428-021-01694-3>
- Plewan, T., & Rinkenauer, G. (2021). Visual search in virtual 3D space: The relation of multiple targets and distractors. *Psychological Research*, 85(6), 2151–2162. <https://doi.org/10.1007/s00426-020-01392-3>
- Previc, F. H., & Blume, J. L. (1993). Visual search asymmetries in three-dimensional space. *Vision Research*, 33(18), 2697–2704. [https://doi.org/10.1016/0042-6989\(93\)90229-P](https://doi.org/10.1016/0042-6989(93)90229-P)
- Rensink, R. A., & Cavanagh, P. (2004). The influence of cast shadows on visual search. *Perception*, 33(11), 1339–1358. <https://doi.org/10.1068/p5322>
- Sobel, K. V., & Cave, K. R. (2002). Roles of salience and strategy in conjunction search. *Journal of Experimental Psychology: Human Perception and Performance*, 28(5), 1055–1070. <https://doi.org/10.1037/0096-1523.28.5.1055>
- Tangtartharakul, G., & Storrs, K. R. (2025). *Visual language models show widespread visual deficits on neuropsychological tests* [Preprint]. arXiv. <https://doi.org/10.48550/arXiv.2504.10786>
- Thornton, T. L., & Gilden, D. L. (2007). Parallel and serial processes in visual search. *Psychological Review*, 114(1), 71–103. <https://doi.org/10.1037/0033-295X.114.1.71>
- Treisman, A. M., & Gelade, G. (1980). A feature-integration theory of attention. *Cognitive Psychology*, 12(1), 97–136. [https://doi.org/10.1016/0010-0285\(80\)90005-5](https://doi.org/10.1016/0010-0285(80)90005-5)
- Wolfe, J. M. (1998). What can 1 million trials tell us about visual search? *Psychological Science*, 9(1), 33–39. <https://doi.org/10.1111/1467-9280.00006>
- Wolfe, J. M. (2021). Guided Search 6.0: An updated model of visual search. *Psychonomic Bulletin & Review*, 28(4), 1060–1092. <https://doi.org/10.3758/s13423-020-01859-9>
- Wolfe, J. M., Cave, K. R., & Franzel, S. L. (1989). Guided search: An alternative to the feature integration model for visual search. *Journal of Experimental Psychology: Human Perception and Performance*, 15(3), 419–433. <https://doi.org/10.1037/0096-1523.15.3.419>
- Wolfe, J. M., & Horowitz, T. S. (2017). Five factors that guide attention in visual search. *Nature Human Behaviour*, 1(3), Article 0058. <https://doi.org/10.1038/s41562-017-0058>
- Wolfe, J. M., & Myers, L. (2010). Fur in the midst of the waters: Visual search for material type is inefficient. *Journal of Vision*, 10(9), Article 8. <https://doi.org/10.1167/10.9.8>
- Wolfe, J. M., Reijnen, E., Horowitz, T. S., Pedersini, R., Pinto, Y., & Hulleman, J. (2011). How does our search engine “see” the world? The case of amodal completion. *Attention, Perception, & Psychophysics*, 73(4), 1054–1064. <https://doi.org/10.3758/s13414-011-0103-0>
- Wolfe, J. M., Yee, A., & Friedman-Hill, S. R. (1992). Curvature is a basic feature for visual search tasks. *Perception*, 21(4), 465–480. <https://doi.org/10.1068/p210465>

- Wolfe, J. M., Yu, K. P., Stewart, M. I., Shorter, A. D., Friedman-Hill, S. R., & Cave, K. R. (1990). Limitations on the parallel guidance of visual search: Color × Color and Orientation × Orientation conjunctions. *Journal of Experimental Psychology: Human Perception and Performance*, 16(4), 879–892. <https://doi.org/10.1037/0096-1523.16.4.879>
- Wu, P., & Xie, S. (2024). V*: Guided visual search as a core mechanism in multimodal LLMs. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, Seattle, WA, 13084–13094. <https://doi.org/10.1109/CVPR52733.2024.01243>
- Xu, Z. (J.), Lleras, A., Shao, Y., & Buetti, S. (2021). Distractor–distractor interactions in visual search for oriented targets explain the increased difficulty observed in nonlinearly separable conditions. *Journal of Experimental Psychology: Human Perception and Performance*, 47(9), 1274–1297. <https://doi.org/10.1037/xhp0000941>
- Yin, S., Fu, C., Zhao, S., Li, K., Sun, X., Xu, T., & Chen, E. (2024). A survey on multimodal large language models. *National Science Review*, 11(12), Article nwae403. <https://doi.org/10.1093/nsr/nwae403>
- Yiu, E., Qraitem, M., Majhi, A. N., Wong, C., Bai, Y., Ginosar, S., Gopnik, A., & Saenko, K. (2025). KiVA: Kid-inspired visual analogies for testing large multimodal models. *Proceedings of the 13th International Conference on Learning Representations*, Singapore. <https://openreview.net/pdf?id=vNATZfmY6R>
- Yue, X., Ni, Y., Zhang, K., Zheng, T., Liu, R., Zhang, G., ... & Chen, W. (2024). MMMU: A massive multi-discipline multimodal understanding and reasoning benchmark for expert AGI. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, Seattle, WA, 9556–9567. <https://doi.org/10.1109/CVPR52733.2024.00913>
- Zhang, J., Cheng, C., Liu, Y., Liu, W., Luan, J., & Yan, R. (2025). Weaving context across images: Improving vision-language models through focus-centric visual chains. In W. Che, J. Nabende, E. Shutova, & M. T. Pilehvar (Eds.), *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics* (pp. 27782–27798). Association for Computational Linguistics. <https://doi.org/10.18653/v1/2025.acl-long.1347>
- Zhang, J., Hu, J., Khayatkhoei, M., Ilievski, F., & Sun, M. (2024). *Exploring perceptual limitation of multimodal large language models* [Preprint]. arXiv. <https://doi.org/10.48550/arXiv.2402.07384>
- Zhao, M., Xu, D., & Gao, T. (2024). *From cognition to computation: A comparative review of human attention and transformer architectures* [Preprint]. arXiv. <https://doi.org/10.48550/arXiv.2407.01548>

Appendices

A. Stimulus construction

To systematically investigate visual search, we developed a configurable stimulus generator with two parallel branches: one for 2D images and one for 3D rendered scenes.

2D stimulus generation

Stimuli were constructed from three basic visual features: shape, color, and material, which are combined according to conditions and set size to create the targets and distractors in each image.

- Shape & size: We used circles (radius: 28 pixels) and squares (side length: 49 pixels), with dimensions chosen to minimize the difference in their pixel area.
- Color: Shapes were either bright yellow (RGB: [255, 255, 0]) or dark blue (RGB: [0, 0, 180]).
- Material: Materials were either a uniform color fill or a Gaussian noise texture. The texture was applied to 1/6 of the shape's surface, creating the perception of a darker yellow (RGB: [38, 38, 0]).

Images were generated at a 1920×1080 pixel resolution. Shapes were randomly placed within a central display area, defined by a margin of 92 pixels from the top/bottom and 204 pixels from the left/right. Object placement was sequential: the target (if present) was placed first, followed by distractors. To prevent overlap, a new position was sampled if an object's bounding box was too close to any existing object. All shapes were placed straight up without rotation.

3D stimulus generation

The 3D stimulus generation followed a similar configurable framework, building upon the CLEVR dataset (Johnson et al., 2017) and using Blender 2.78c (Blender Foundation, 2017) for rendering. A standard Blender project file defined the global scene parameters. The camera was positioned at a 33.7° angle relative to the horizontal plane, providing a view that revealed the front, top and side faces of the objects, thus balancing object visibility with depth perception. The scene was illuminated by three lights to ensure that objects are clearly visible with distinct shadows.

As in the 2D branch, the configurable visual properties of the 3D scenes included shapes, colors, and materials.

- Shape & size: We used cubes and spheres in the 3D scenes. To match their perceived size, the cube's side length was scaled by a factor of 0.764 relative to the sphere's diameter. The absolute size was calibrated so that a 3D sphere in the center of the scene had the same projected pixel area as a 2D circle. Due to linear perspective, the perceived size of 3D objects varied with their distance from the camera.
- Color: We used the same bright yellow (RGB: [255, 255, 0]) as in 2D. However, a brighter blue (RGB: [30, 30, 255]) was used because the original dark blue appeared too dark in 3D renders due to reflection and shadow effects.

- **Material:** As in 2D, a noise texture defined as a custom Blender material was created, which contains the noise in darker yellow on 1/6 of the object's surface.

Rendered images were 1920×1080 pixels. Objects were positioned upright within a square area, whose projection was calibrated to approximately match the 2D display area. To prevent collisions, a minimum distance between object centers was enforced. However, partial occlusion between objects was permitted, serving as a natural depth cue. Consequently, the projected size of each object varied with its position due to perspective.

Trial generation per condition

For both 2D and 3D stimuli, the generator saved the target's location for each image/trial. For 3D scenes, the target's precise pixel coordinates and total number of pixels it occupied were also recorded.

Each experimental condition consisted of an equal number of target-present and target-absent trials. We also balanced the distribution of distractor types within each trial. To ensure an equitable distribution of distractor types when the total number of distractors was not perfectly divisible by the number of types, the remaining distractors were assigned sequentially to the distractor types in a fixed order until all distractor slots were filled. Consequently, the counts of some distractor types may have differed by one.

B. Prompt for MLLMs evaluation

To evaluate the MLLMs, we used a simple prompt designed to closely mirror the instructions given to human participants. However, initial pilots showed performance degradation when textured distractors were present (i.e., C4, C9), as the models struggled to differentiate the 'solid yellow' target from 'textured yellow' non-targets. To resolve this ambiguity, we added the explicit clarification 'without any texture on its face'. This addition stabilized model performance and aligned the task with the intended search parameters. The final prompt template was as follows:

```
In this task, you will search for a solid yellow {block} without any texture on its
face in an image.
{variations}
Solid yellow {block}: Output 0 for absent or 1 for present.
```

C. Spatial distribution of target locations by difficulty rank

To investigate whether the spatial location of the target systematically influenced search difficulty, we conducted a post-hoc analysis of target positions and their associated error rates with all target-present trials. For each specific experimental condition (e.g., C7, 2D, set size 256), the 25 unique image versions were ranked from 1 (easiest; lowest error rate) to 25 (hardest; highest error rate). The normalized spatial coordinates of the target were then plotted, with marker size representing this difficulty rank.

The analysis reveals a modest center-of-screen advantage only for humans (Figure 2.C1). Targets located closer to the center of the display tended to have a lower difficulty rank compared to those in the periphery. However, human performance also exhibited a left-side

bias: targets appearing on the left half of the display had a higher mean difficulty rank. This left-side bias is presumably a result of stimulus-response incompatibility related to the response mapping ('A' for Absent on the left, 'P' for Present on the right). The presence of a target on the left may have induced wrong presses of the spatially corresponding left key. It should be noted that the overall error rates in feature searches are low (Figure 2.3); this effect is revealed by our relative ranking method.

In contrast, the MLLM data showed a more uniform pattern, with difficulty ranked similarly toward both the left and right peripheries. This suggests that, compared with human observers, MLLM performance is more invariant to target position, suggesting different search mechanisms between humans and MLLMs.

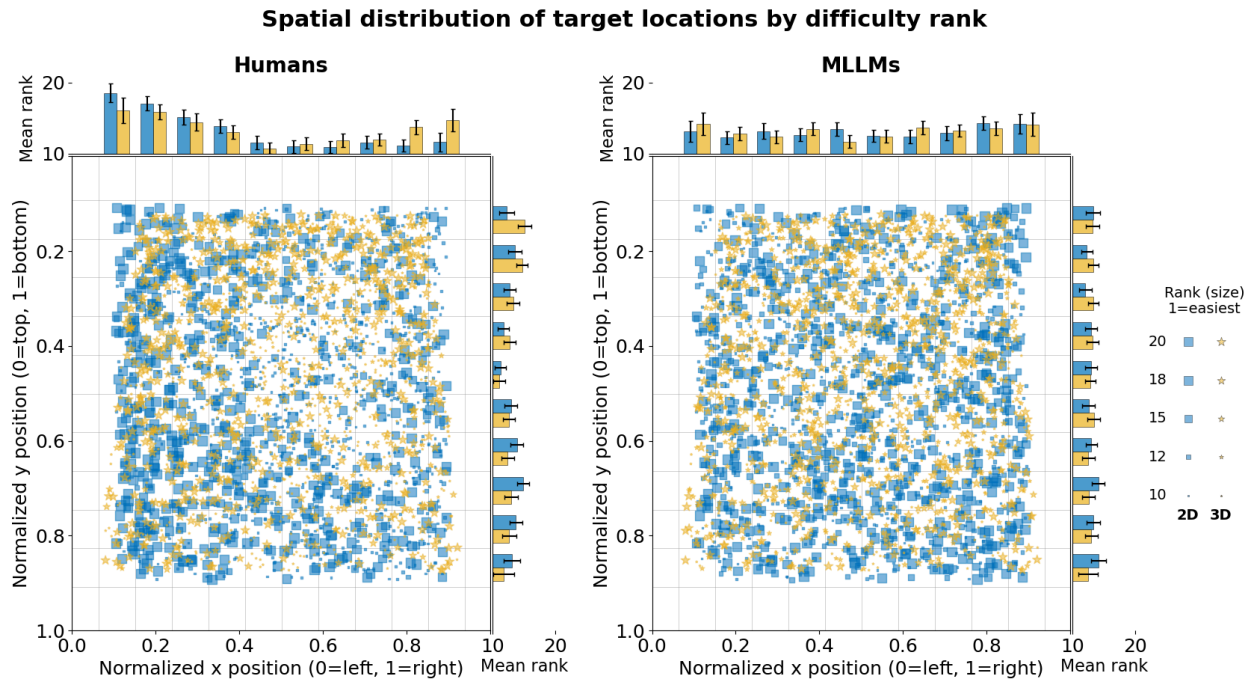


Figure 2.C1. Spatial distribution of target locations for humans (left) and MLLMs (right), with marker size indicating the mean difficulty rank (1 = easiest, 25 = hardest). Marker size is scaled by difficulty rank, with larger markers indicating higher error rates. The marginal plots show the mean difficulty rank binned by spatial location with error bars of 95% confidence intervals of the mean rank for each bin.

We repeated the rank-based analysis using human response time as the difficulty measure (Figure 2.C2). For each condition, the 25 image versions were ranked from 1, the lowest median/mean response time/easiest, to 25, the highest response time/hardest. The spatial pattern broadly mirrors the error-based rank analysis: humans show a modest center-of-screen advantage (faster responses near the center) and a small left-side cost consistently.

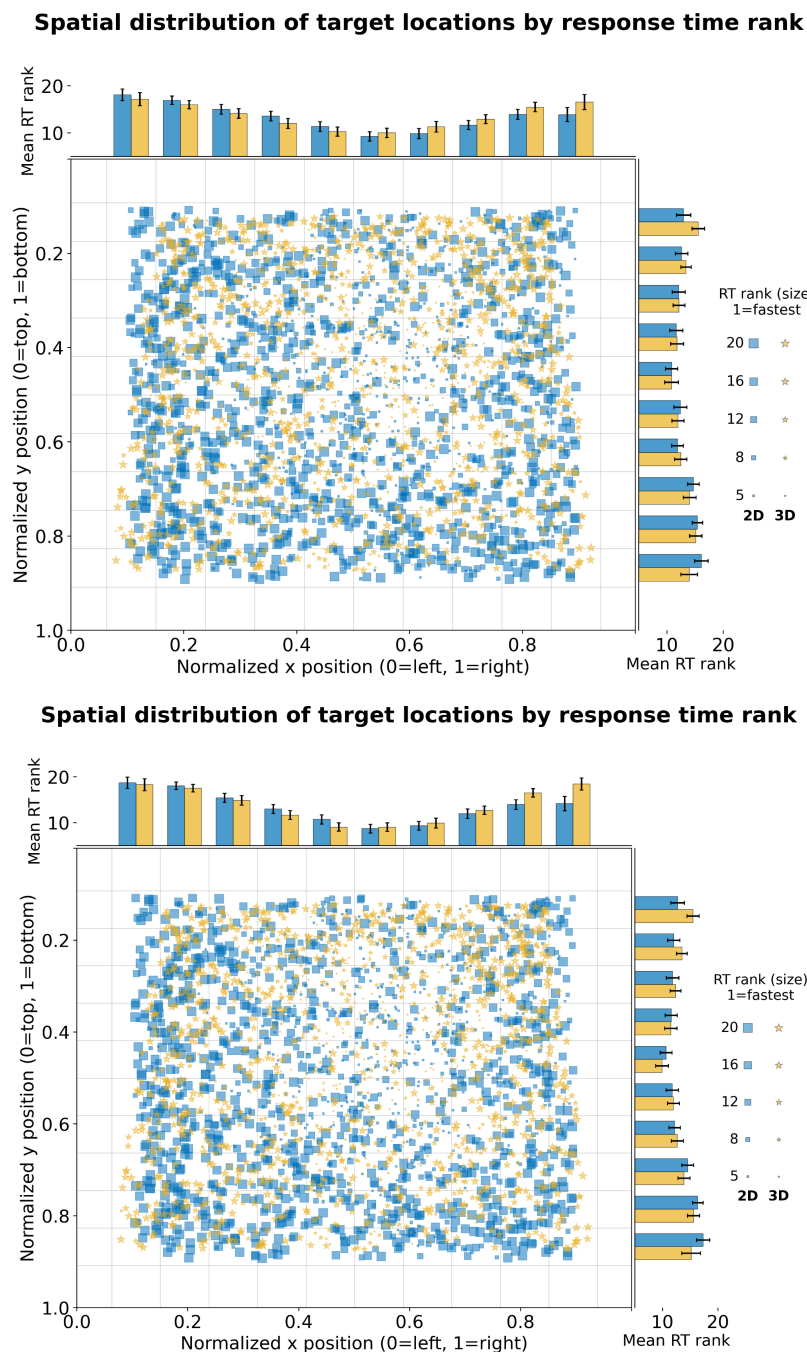


Figure 2.C2. Spatial distribution of target positions with response time-based difficulty rank. The marker size indicates the mean difficulty rank of median response times (1 = easiest, 25 = hardest), with larger markers indicating higher response times. The marginal plots summarize mean rank by spatial bin with 95% confidence intervals.

Figures C3 and C4 show mean-based visualizations of the same spatial layout: mean error rate (C3) and mean response time (C4). In both panels marker size encodes the metric (larger markers = higher error rate or longer response time), and marginal bar plots show the mean metric within spatial bins with 95% confidence intervals.

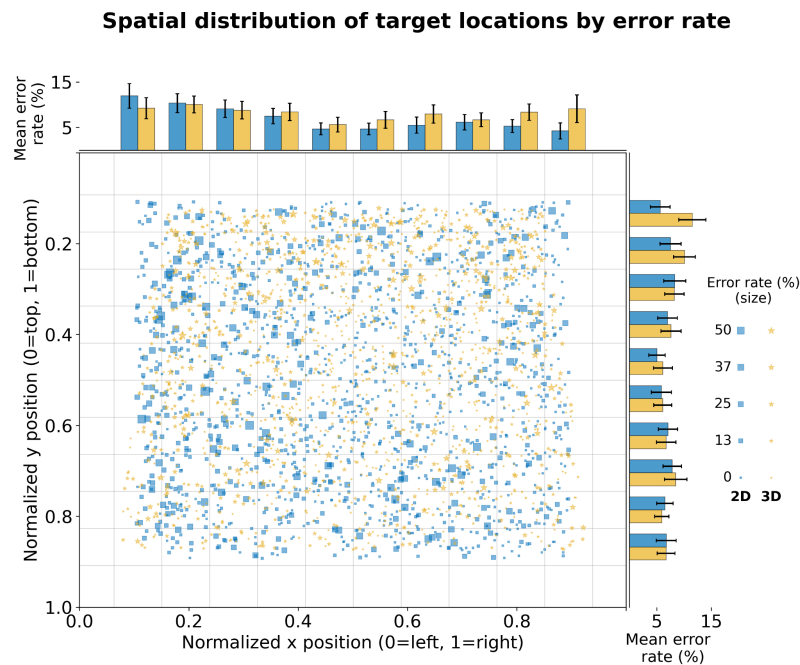


Figure 2.C3. Spatial distribution of target positions with human error rates. Marker size is scaled by error rate (larger markers indicating higher error rates). The marginal plots summarize mean error rate by spatial bin with 95% confidence intervals.

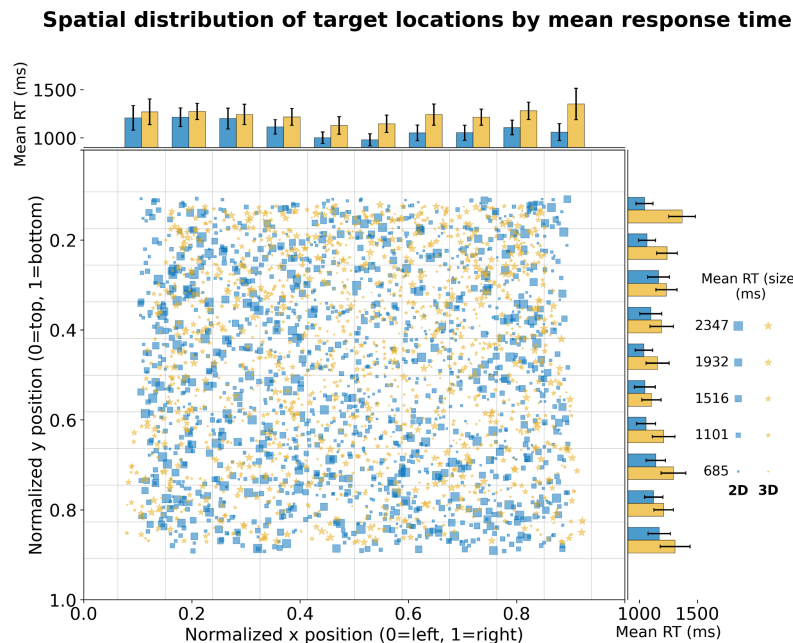


Figure 2.C4. Spatial distribution of target positions with mean human response times. Marker size is scaled by mean response time (larger markers indicating longer response times). The marginal plots summarize the mean response time by spatial bin with 95% confidence intervals.

3

Walk along: An experiment on controlling the mobile robot ‘Spot’ with voice and gestures

Intermezzo

Chapter 2 investigated an elementary form of misalignment in information processing: humans and MLLMs reach similar visual search outcomes through different mechanisms, with errors diverging in complex scenes (Research question 1). Perception alone, however, says little about interaction. Chapter 3 moves to an embodied setting in which operators command a mobile robot, covering all three processing stages and asking how command modality (voice or gesture), and thus the processing codes (verbal or spatial), influences task performance and perceived intuitiveness (Research question 2).

This chapter was published as:

Zhang, R.*, Van der Linden, J.*, Dodou, D., Seyffert, H., Eisma, Y. B.#, & De Winter, J.# (2025). Walk Along: An Experiment on Controlling the Mobile Robot ‘Spot’ with Voice and Gestures. *ACM Transactions on Human-Robot Interaction*, 14(4), Article 64. <https://doi.org/10.1145/3729540>

*Joint first authors

#Joint last authors

Abstract

Robots are becoming more capable and can autonomously perform tasks such as navigating between locations. However, human oversight remains crucial. This study compared two touchless methods for directing mobile robots: voice control and gesture control, to investigate the efficiency of these methods and the preference of users. We tested these methods in two conditions: one in which participants remained stationary and one in which they walked freely alongside the robot. We hypothesized that walking alongside the robot would result in higher intuitiveness ratings and improved task performance, based on the idea that walking promotes spatial alignment and reduces the effort required for mental rotation. In a 2×2 within-subject design, 218 participants guided the quadruped robot Spot along a circuitous route with multiple 90° turns using rotate left, rotate right, and walk forward commands. After each trial, participants rated the intuitiveness of the command mapping, while post-experiment interviews were used to gather the participants' preferences. Results showed that voice control combined with walking with Spot was the most favored and intuitive, whereas gesture control while standing caused confusion for left/right commands. Nevertheless, 29% of participants preferred gesture control, citing increased task engagement and visual congruence as reasons. An odometry-based analysis revealed that participants often followed behind Spot, particularly in the gesture control condition, when they were allowed to walk. In conclusion, voice control with walking produced the best outcomes. Improving physical ergonomics and adjusting gesture types could make gesture control more effective.

3.1 Introduction

Robots have traditionally been viewed as devices designed to perform repetitive tasks, mainly in industrial settings and logistical operations. However, with the growing capabilities of AI, robots are taking on new roles. Modern robots can understand and adapt to their surroundings, paving the way for mobile robotics. This field extends beyond the familiar household cleaning robots and now also encompasses drones (Halder & Afsari, 2023; Roldán-Gómez et al., 2021), surveillance robots (Chen et al., 2021; Hafezi et al., 2024), underwater robots (Brantner & Khatib, 2021; Nauert & Kampmann, 2023), warehouse robots (Fottner et al., 2021; Jacob et al., 2023), agricultural robots (Benos et al., 2023; Gonzalez-de-Santos et al., 2020), and assistant robots (Hong et al., 2022; Mišeikis et al., 2020), among others.

The human-machine interface (HMI) plays a vital role in the control of mobile robots, as these robots are not yet capable of fully autonomous operation in open-ended environments (e.g., Endsley, 2017; Ezenkwu & Starkey, 2019; Hatanaka et al., 2023; Pianca & Santucci, 2023). Although mobile robots can execute certain maneuvers or subtasks, human supervision and interaction are still needed for overall task completion. Traditional HMIs for controlling mobile robots include joysticks (Dinh et al., 2017; Moniruzzaman et al., 2022), gamepads (Solanes et al., 2022; Wan et al., 2023), keyboards (Di Vincenzo et al., 2022), and graphical user interfaces on tablets carried by human operators (Colceriu et al., 2023; Kaczmarek et al., 2021).

3.1.1 Voice and gestures as natural interfaces

While traditional HMIs have demonstrated their effectiveness over the years, they are not necessarily the most ideal solutions (as pointed out by Gorecky et al., 2014). Instead, an ideal UI for mobile robots might mimic human-to-human or human-to-animal interactions (Krueger, 1993). Given that the human brain has evolved specialized areas for processing speech and body language, using such natural UIs could require less mental effort compared to interactions via handheld devices or other physical media. Hancock (1993), ahead of his time, proposed the ‘sheepdog metaphor’ for human-machine collaboration. This metaphor describes the relationship between a human (the shepherd) and a robot (the sheepdog). Just as a shepherd gives basic commands to a sheepdog, allowing it to manage subtasks based on its instincts and learned experience, a human operator can set high-level targets for a mobile robot, which then performs these tasks independently. The sheepdog metaphor illustrates that a limited set of commands can facilitate effective human-machine collaboration. More recently, and in accordance with the sheepdog metaphor, the concept of maneuver-based control has also been explored in automated driving (Detjen et al., 2020; Fink et al., 2023).

Possible candidates for interaction between humans and mobile robots include voice control (Li et al., 2023; Naeem et al., 2024) and control by means of mid-air gestures (Carfi & Mastrogiovanni, 2021; Coronado et al., 2017). Beyond technical issues regarding the detectability of the commands, such as voice control being less effective in noisy environments (Brunete et al., 2021; D’Attanasio et al., 2024) or gesture control being impractical when someone is wearing gloves or when otherwise constrained (Hatscher & Hansen, 2018; Sadhu et al., 2023), there are human factors to consider when implementing such HMIs. Voice commands are transient, while gestures can be sustained by the user for longer. It has been found that gesture control can generate a higher sense of embodiment than a traditional keyboard interface, although with a lower task performance (Di Vincenzo et al., 2022). Research has shown that when individuals were tasked with solving challenging spatial-visualization tasks (such as mentally rotating 3D objects), they spontaneously produced hand gestures and showed improved performance when doing so (Chu & Kita, 2011). These findings indicate that gestures can serve as a bridge between physical movement and abstract spatial reasoning, and suggest that gestures might represent an intuitive interface for controlling a robot’s movements and rotations. With speech interfaces, on the other hand, an intended movement of the robot must first be mentally translated into a verbal command. At the same time, Norman (2010) argued that gestural interfaces are not inherently intuitive or easy to learn.

3.1.2 Existing applications of voice and gestures for mobile robot control

Previous research on voice versus gesture control for operating mobile robots (Abich & Barber, 2017; Chivarov et al., 2019), in car cockpits (Detjen et al., 2019), or for controlling visual interfaces (Flick et al., 2021) shows that users generally prefer voice control over gesture control (but see Korkiakoski et al., 2024, for mixed findings). Differences in the robustness of detection and the simplicity or trainability of gestures (Nogales & Benalcázar, 2021; Zhou et al., 2023) may partly explain the relatively low ratings attributed to gesture control in some of the prior research. Factors such as the social acceptability of commands (e.g., speaking or gesturing in a shared space may confuse others; Rico & Brewster, 2010), competition with other tasks (e.g.,

drivers needing to keep their attention on the road and their hands on the wheel; Bilius & Vatavu, 2021), and recognition time delays (e.g., Halder et al., 2024) also play important roles.

As for quadruped mobile robots, previous studies investigated giving commands via gestures and/or body pose (Rippy et al., 2024; Sandberg, 2023; Shin et al., 2024; Steinke et al., 2023) or via voice, eye gaze, and head pose (Zhang et al., 2023). Despite several technical limitations needing resolution, these innovative methods of control showed promise. Other studies demonstrated the potential of quadruped mobile robots for guiding people with visual impairments (Cai et al., 2024; Chen et al., 2023; Doore et al., 2024; Due, 2023; Hata et al., 2024; Kim et al., 2025) or examined people's perceived safety as a function of the robot's walking style (Hashimoto et al., 2024). Of particular interest is a study by Chacón Quesada and Demiris (2024), which evaluated the effectiveness of an Augmented Reality (AR) interface compared to a traditional handheld interface for controlling a quadruped robot. Participants completed navigation and manipulation tasks, including a condition that incorporated a cognitive offloading feature into the AR interface. In this context, cognitive offloading refers to allowing users to physically position themselves and use hand gestures, such as pointing to where the robot should go and using voice commands to direct its actions. The study found that with cognitive offloading, the AR-based interface yielded faster task completion times, reduced mental workload, and increased usability ratings.

For a complete comparison between voice and gesture control in mobile robot operation, user orientation relative to the robot has to be considered. Research on stimulus-response compatibility shows that a difference in orientation between stimulus and response (requiring mental rotation) increases errors and information processing time (Fitts & Seeger, 1953; Shepard & Metzler, 1971).

Wickens and Preveit (1995) identified two HMI display perspectives: egocentric and exocentric. An egocentric display matches the user's viewpoint, since control inputs correspond with the vehicle's/robot's direction, as in first-person-view (FPV) displays used by expert drone racing pilots (Pfeiffer & Scaramuzza, 2021; Tezza et al., 2021). In contrast, an exocentric display, such as a top-down view or bird's-eye view, provides a detached perspective, which may increase the user's situational awareness but complicates control input generation (Smolyanskiy & Gonzalez-Franco, 2017). In practice, the user's mobility and positioning relative to the robot will result in different perspectives: egocentric for users closely aligned with the robot's orientation and exocentric for those observing from a distant or static perspective, both of which are likely to influence performance in control tasks.

To the best of our knowledge, no comparative study has investigated voice and gesture interfaces for mobile robots while accounting for differences in users' mobility (e.g., walking vs. standing). Most existing research that used voice or gestures to control a mobile robot combined these modes or lacked a direct comparative analysis (Chacón Quesada & Demiris, 2024; Doore et al., 2024; Du et al., 2021; Fardana et al., 2013; Hata et al., 2024). In a previous study involving older adults and individuals with disabilities controlling a mobile robot, gesture control was found to be less preferred and more fatiguing than voice or joystick control

(Chivarov et al., 2019). Abich and Barber (2017) compared speech and sign-language-based gestures for robot navigation and investigation tasks. However, because the study replaced each spoken command with an artificial one-to-one gesture mapping based on sign language (unlike natural/spontaneous gestures), its comparison between voice and gesture control may be hard to interpret.

Existing gesture interfaces for mobile robot control have relied on wearable gloves (Abich & Barber, 2017) or an inertial measurement unit to track hand movements (Kandalaft et al., 2018; Meghana et al., 2020) and non-head-mounted cameras to track body posture (Fardana et al., 2013; Sandberg, 2023; Shin et al., 2024), hand position (Chivarov et al., 2019) or finger counts (Vanamala et al., 2022), which limited the users' mobility and might not be effective in mobile scenarios. We chose to use an AR headset that tracks the user's inputs well in both mobile and stationary conditions. The vision-based gesture recognition we implemented does not require the participants to wear sensors or stay in front of cameras or microphones, leading to natural hand and body movements. Furthermore, we have published our code as open-source, allowing others to reproduce our setup and build upon it.

3.1.3 A comparative study of voice and gesture control in standing and walking conditions

We conducted an experiment using the AR device Microsoft HoloLens 2 to compare voice and gesture control for operating Spot, a popular quadruped robot developed by Boston Dynamics. In the current study, participants navigated the robot along a trajectory with multiple 90° turns using each control method twice: once while standing still at the starting point and once while walking alongside the robot. When standing still, participants had a consistent view of the trajectory but experienced stimulus–response incompatibility, especially when the robot was oriented toward or approaching them. For instance, a *Rotate Right* command in that case would make the robot rotate left from the participants' perspective. When walking, participants could position themselves behind or next to the robot to reduce this incompatibility.

We examined the effects of these two independent variables (1. control method, 2. participant mobility) on task performance and self-reported experience. We differentiated between the detectability of the commands (i.e., a technical issue regarding speech recognition and computer vision) and the intuitiveness of command-to-robot-movement mapping (a human factors issue). We hypothesized that walking with the robot would result in better task performance and higher intuitiveness ratings than controlling the robot while standing still.

Additionally, we expected that gesture control while standing still would be particularly unintuitive due to a spatial mismatch between the participant's hand gestures and the robot's movement. In contrast, voice control, which involves verbal and auditory processing, conflicts less with visual tasks according to the multiple resource theory by Wickens (2002, 2008). This theory posits that humans have distinct cognitive resources (visual, auditory, spatial, and verbal), and tasks using the same resource type interfere more with each other. Therefore, voice control should cause less cognitive conflict during moments of incompatibility compared to gesture control, which relies heavily on visual-spatial resources.

3.2 Methods

3.2.1 Participants

A total of 218 participants took part in the experiment. Of these, 216 were students enrolled in an MSc course on Human-Robot Interaction offered by the Faculty of Mechanical Engineering at TU Delft. The remaining two participants were staff members from the same faculty. The course was part of the MSc Robotics program and the Biomechanical Design track within the MSc Mechanical Engineering program at TU Delft. Consequently, it can be assumed that the students had a strong interest in, or basic to advanced knowledge of, technical and robotic systems. However, the HoloLens 2 in combination with the Spot robot was introduced to the students for the first time, meaning that they were unfamiliar with the experimental facility and task. The experiment was conducted from 4 December 2023 to 19 January 2024.

In addition to these 218 participants, 3 participants could not complete the experiment because their control inputs were not recognized, presumably due to an issue with the wireless connection during the first 2 days of the experiment, and were excluded from consideration. Because our interest was in making a complete comparison between gesture control and voice control, we did not exclude participants whose hand gestures or voice commands were less well recognized during the experiment.

Participants' ages, collected via a post-experiment questionnaire, ranged from 21 to 30 years, with a mean age of 23.5 years ($SD = 1.64$). Among the participants, 194 were right-handed, 18 were left-handed, and 6 were mixed-handed. Gender distribution included 160 males, 55 females, and 3 individuals who preferred not to disclose their gender. Of the participants, 121 reported never wearing any vision aids, 49 wore glasses during the experiment, 36 wore contact lenses, and 12 typically wore glasses or contact lenses but did not wear them during the experiment. In response to a question: "*Before the experiment, did you ever wear a HoloLens or similar augmented-reality device?*", 155 participants answered *no*, and 63 participants answered *yes*.

The experiment was approved by the Human Research Ethics Committee of the TU Delft, approval number 3502, with each participant providing written informed consent before the experiment.

3.2.2 Hardware and software

The experiment setup included the quadruped robot Spot Explorer (Boston Dynamics, 2020), measuring 1100 mm in length, 500 mm in width, and 840 mm in standing height, as well as a HoloLens 2 (Microsoft, 2024a), a device used in combination with Spot in earlier studies (Chacón Quesada & Demiris, 2024; Chen et al., 2024; Rippey et al., 2024; Soler et al., 2024; Steinke et al., 2023; Vom Hofe et al., 2023; Zhang et al., 2023). Additionally, a Windows PC equipped with an Intel i7-8700K processor and an NVIDIA RTX 2080 graphics card was used. These devices were connected to a TP-Link AX3000 Gigabit Wi-Fi 6 Router (TP-Link, 2019), with the PC connected via a network cable and the HoloLens 2 and Spot connected via a 2.4 GHz Wi-Fi network.

The HoloLens 2 was used to capture the participant's control inputs. It ran an application developed in Unity v2021.3.28f1 (Unity, 2021). The application was created using the Mixed Reality Toolkit (MRTK) v2.8 (Microsoft, 2022a) for interaction design and UI components. The OpenXR Plugin was used to ensure compatibility with the HoloLens platform. Communication between the HoloLens 2 and the PC took place via the Robot Operating System (ROS) (Quigley et al., 2009) and ROS-TCP-Connector package (Unity, 2022). Spot was controlled from the PC using the Python interface of Spot-SDK 3.3.2 (Boston Dynamics, 2023). An overview of the hardware and software infrastructure is provided in Figure 3.1.

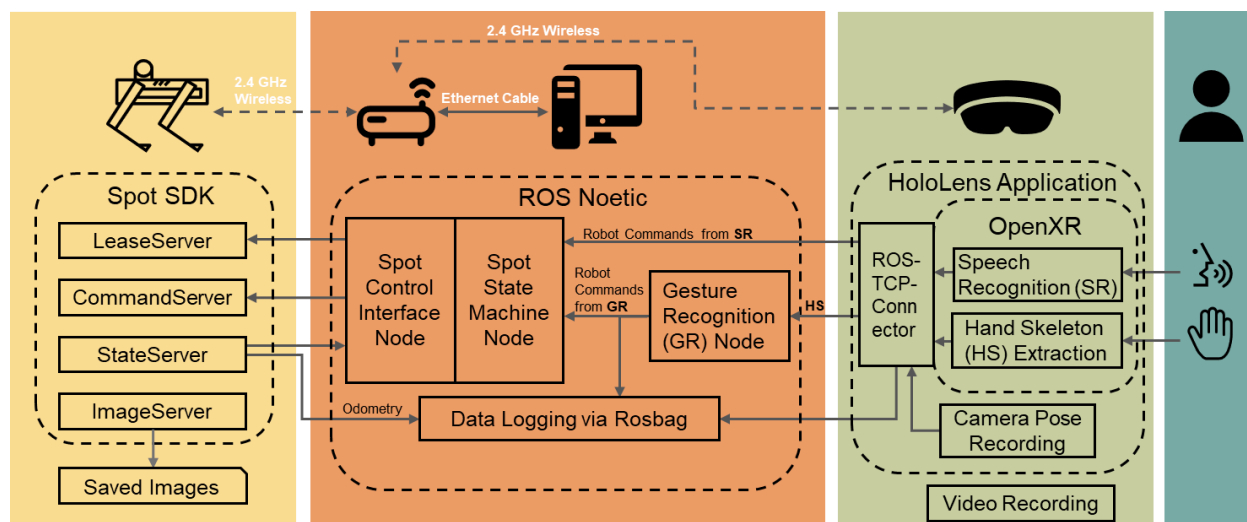


Figure 3.1. The hardware and software structure of the setup. The HoloLens 2 captures human user inputs (i.e., speech and hand skeleton data) and processes them via a HoloLens application, which was developed in Unity (Unity, 2021) with the OpenXR Plugin for Unity, managed by the Mixed Reality Toolkit 2 (MRTK) (Microsoft, 2022a). The application also records the 6D pose of the central front camera of the HoloLens 2. The processed user inputs from the plugin and the camera's 6D pose are recorded via Rosbag. Additionally, a corresponding video of the user's view is recorded by the HoloLens 2. Robot commands are recognized either from speech by the plugin or from gestures by a ROS node with the hand skeleton data from the application. A state machine manages the actions of the Spot robot and controls it via the Spot Software Development Kit (SDK).

3.2.3 Task

The participants were tasked with controlling the robot using either their voice or gestures to direct Spot along a trajectory on the floor as quickly as possible. The trajectory contained 14 virtual nodes, including its start and end nodes (Figures 3.2 & 3.3). Participants needed to use three maneuvers/commands: *Walk Forward*, *Rotate Left*, and *Rotate Right*, to complete the trajectory. The *Walk Forward* command moved Spot 1 m forward in its Crawl locomotion mode, in which three feet touch the ground at all times. The default velocity limit for crawling was used, resulting in an execution duration of 3 s for each command. Crawl mode was selected because Spot walks more slowly in this mode compared to the Walk (Trot) mode. It was reasoned that the faster Walk mode could appear intimidating and reduce the participants' willingness to walk alongside the robot. The *Rotate Left* and *Rotate Right* commands rotated Spot 90° to its left and right, respectively. The task was completed with a minimum number of 23 commands, including 13 *Walk Forward*, 6 *Rotate Left*, and 4 *Rotate Right* commands, resulting in 23 checkpoints of the robot pose.

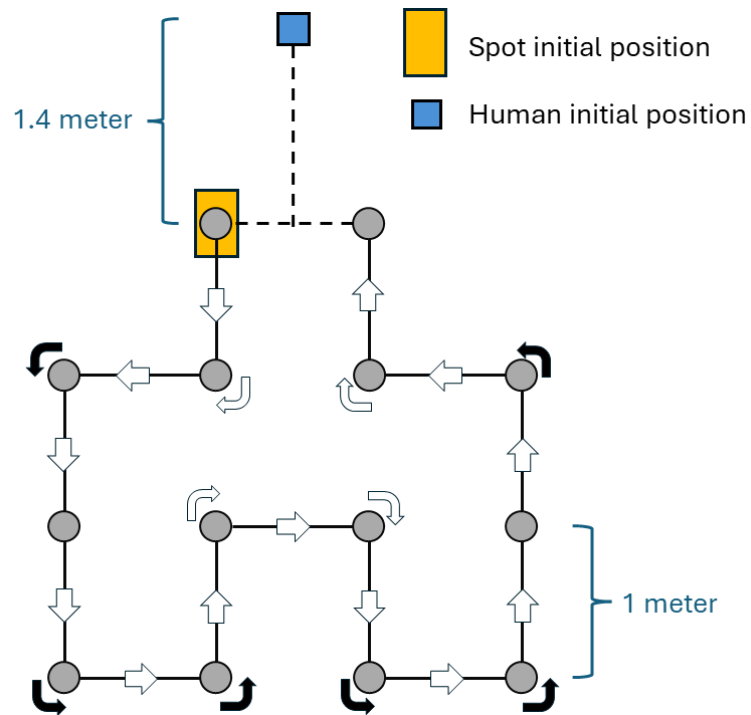


Figure 3.2. The target trajectory of the Spot robot. The 14 nodes are indicated by means of gray circles. The *Walk Forward*, *Rotate Left*, and *Rotate Right* commands are represented by straight arrows, black curved arrows, and white curved arrows, respectively.

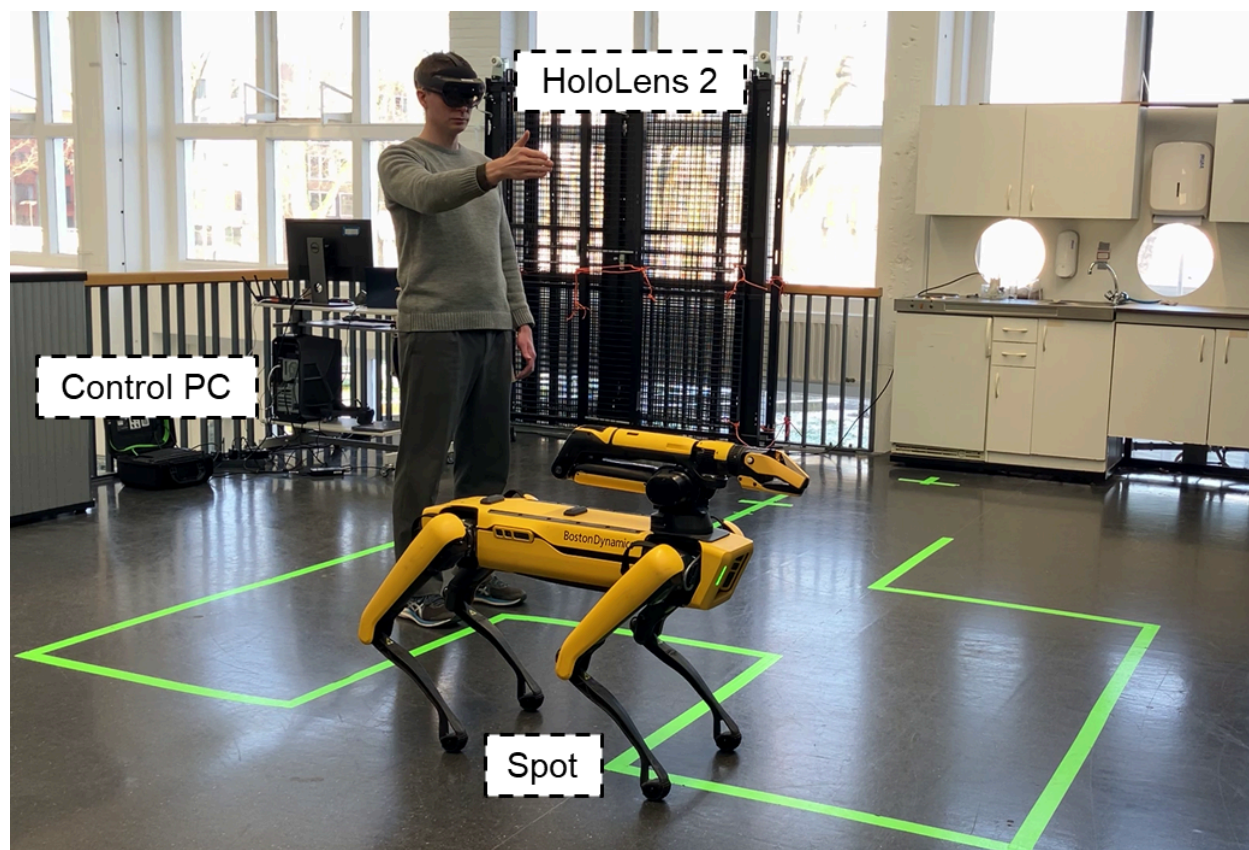


Figure 3.3. Overview of the experimental setting. A participant wearing the HoloLens 2 controlled Spot using the gesture interface while walking alongside it.

3.2.4 Independent variables

There were two independent variables, each consisting of two levels, and both were manipulated within subjects:

1. *Participant's mobility.* Participants were instructed to either remain in their initial position or walk with the robot.
2. *Control method.* Participants used either voice or gestures to control Spot. In the voice control conditions, participants vocalized one robot command at a time and then waited for the robot to execute it. Similarly, in the gesture control conditions, participants performed a gesture, held it until it was recognized, and then waited for the robot to execute the command.

These two independent variables resulted in the four experimental conditions shown in Table 3.1. The experimental conditions were fully counterbalanced, with the 24 possible sequences of the four experimental conditions repeating every 24 participants.

Table 3.1. The four conditions tested in the experiment. Participants were instructed to use either voice (V) or gesture (G) to control the robot while walking (W) with it or standing (S) at the initial position.

		Participant's mobility	
		<i>Walking</i>	<i>Standing</i>
Control method	<i>Voice</i>	Voice control–Walking (VW)	Voice control–Standing (VS)
	<i>Gesture</i>	Gesture control–Walking (GW)	Gesture control–Standing (GS)

3.2.5 Speech recognition

The HoloLens features speech recognition that can be customized within Unity using the MRTK (Microsoft, 2022b). In this experiment, three voice commands were programmed to be recognized in order to control the robot, namely *Walk Forward*, *Rotate Left*, and *Rotate Right*. A pilot study showed that participants took about 1 s to utter a command, with an additional 1.2 s required for the system to process it. Thus, the entire process from speaking to recognition took approximately 2.2 s.

3.2.6 Gesture recognition

A custom gesture recognition pipeline was implemented based on the MRTK of the HoloLens (Figure 3.4). The HoloLens provides the 3D coordinates of detected hand skeletons in 26 joints (25 joints from the hand skeleton and 1 joint of the palm) with respect to the HoloLens frame at a frequency of approximately 30 Hz (Microsoft, 2020, 2021). The data of the hand skeletons from the HoloLens were forwarded to the PC and classified into one of four gestures using a Support Vector Machine (SVM) trained by previously collected gesture data from the HoloLens (see Appendix A). Three gestures were used for robot commands (Figure 3.5F). One gesture (fist) acted as a neutral gesture, similar to silent periods between voice commands.

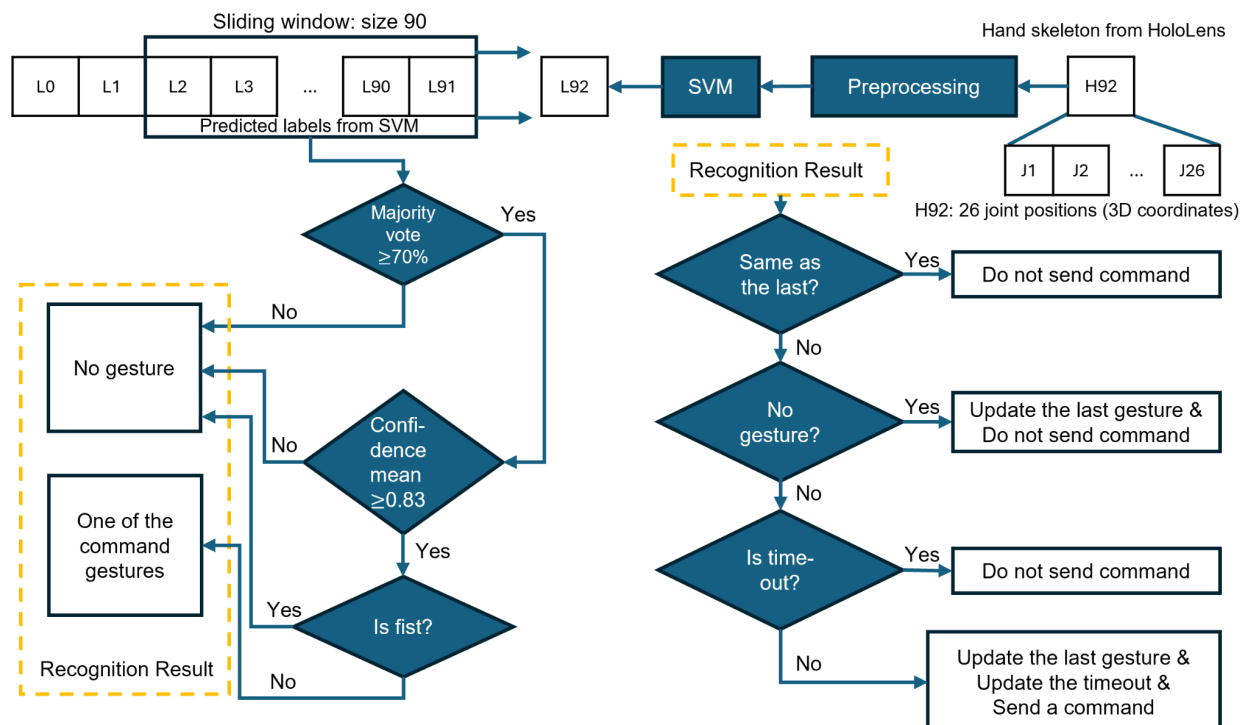


Figure 3.4. Gesture recognition process. Each hand skeleton frame detected by the HoloLens 2 was streamed to the PC as 3D coordinates of 26 joints and filled into a sliding window. A Support Vector Machine (SVM) gesture classifier predicted gestures in the latest 90 frames (frame rate of approximately 30 Hz). These predictions were processed by majority voting and a confidence check to determine the current hand gesture, which was sent as a valid gesture command only if it was a new (other than the fist/neutral) gesture and the timeout was not activated.

Hand skeleton data from the HoloLens were streamed to the SVM gesture classifier, and a sliding window system was applied to the predicted labels from SVM. The sliding window buffered the last 90 predictions of the incoming hand skeletons. This buffer remained unchanged if the HoloLens did not capture any hand skeleton data. A gesture was recognized when at least 70% (i.e., 63 or more samples) of the predicted labels within the current buffer were in the same class, and the mean classification confidence level of all 90 samples was above 0.83. Given that the recording frequency was approximately 30 Hz, the minimum time required for recognition was approximately 2.1 s. If the current gesture was the neutral gesture or the same as the last recognized gesture, no command was sent to the robot. After a gesture command was successfully sent to the robot, a 3-s timeout was activated during which no further recognized commands would be transmitted.

3.2.7 Heads-up display information

Once the participant had put on the HoloLens, a series of visual AR elements appeared to guide the participant through each phase of the experiment (Figure 3.5). After the participant completed the built-in HoloLens instructions and performed the eye-tracking calibration (Figure 3.5A), our custom application was initialized. Initially, the first guidance panel and two buttons to start the experiment or to exit the application were displayed (Figure 3.5B). Upon pressing the

start button (Figure 3.5C), the initial panel was replaced by the second panel and two switches, corresponding to gesture and voice control (Figure 3.5D). This guidance panel instructed the participant to follow the demonstration given by the experimenter (Figure 3.5E) and to toggle the appropriate switch to proceed with the upcoming condition (Figure 3.5F).

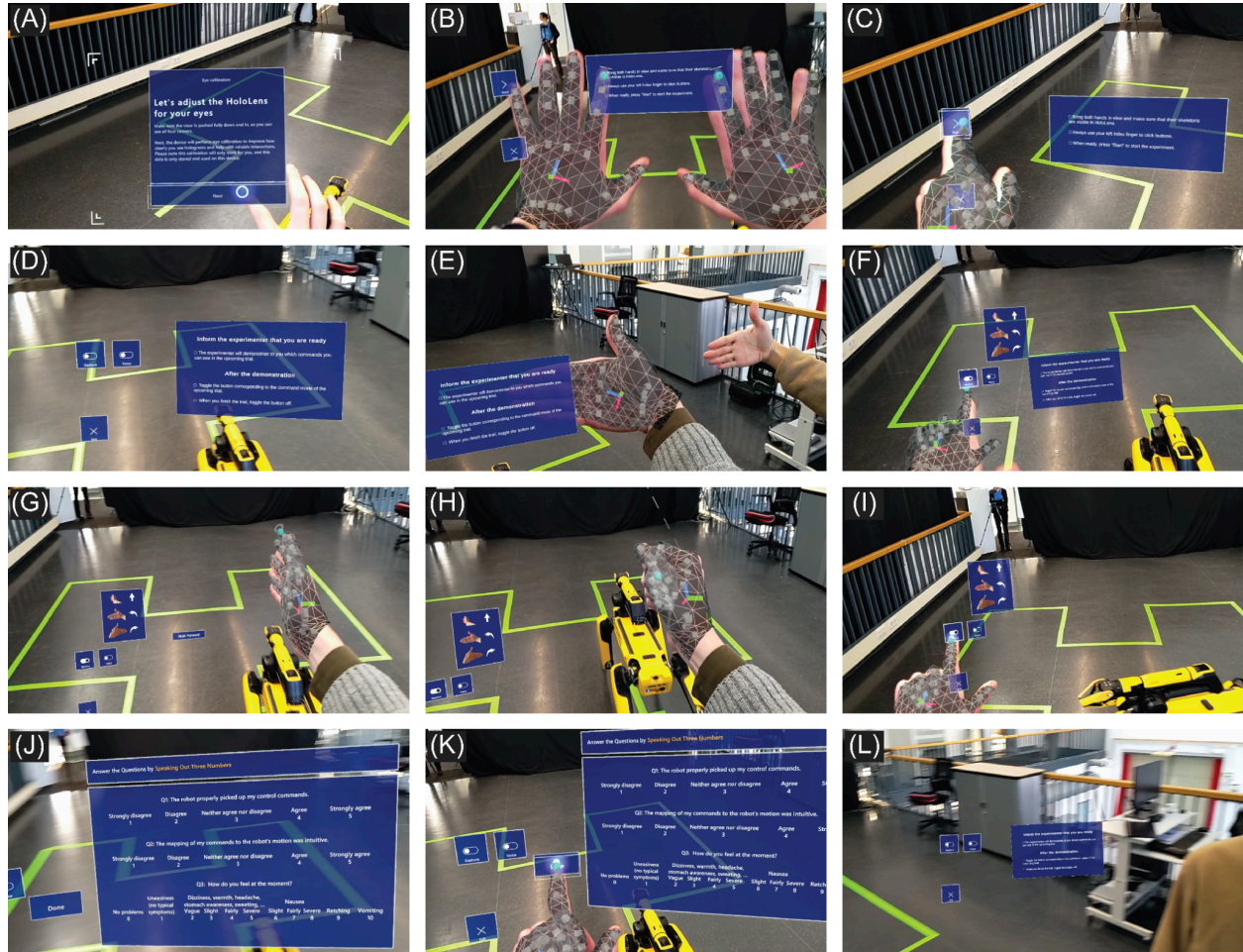


Figure 3.5. HoloLens interfaces during various phases of the experiment under the Gesture control–Standing (GS) condition. (A) The participant pressed the eye-tracking calibration button. (B) After the calibration, the participant saw the initial guidance panel with general instructions, along with “Start” and “Exit” buttons. (C) The participant pressed “Start”. (D) A second guidance panel, informing the participant about a demonstration from the experimenter and how to start a trial, was visible along with two switches for gesture and voice control. (E) The participant was mimicking the experimenter’s demonstration of commands. (F) Toggling gesture control made a command panel visible, while the second guidance panel was fading out. (G) A popup message confirmed a successfully recognized gesture command. (H) The participant continued the trial. (I) The participant toggled off the gesture control upon completion of the trial. (J) A post-trial questionnaire panel and a “Done” button were shown. (K) The participant pressed “Done” to finalize the questionnaire. (L) The questionnaire panel disappeared, and the second instruction panel was resumed.

Once a switch was toggled, the second guidance panel faded out (Figure 3.5G). A command panel with the three available commands remained available (gesture commands: Figure 3.5F;

voice commands: Figure 3.6). The command panel served as a memory aid for the participant, so that they could always refer to the three possible commands. Upon recognition of a voice or gesture command, a popup message (*Walk Forward*, *Rotate Right*, or *Rotate Left*) was displayed for 1.5 s (Figures 3.5G & 3.6). Additionally, the HoloLens emitted a beep tone approximately 3 s after the command was recognized to indicate that the participant could issue the next command. The execution of the robot's movement lasted 3 s, commencing with the appearance of the popup message and concluding approximately at the same time as the beep.

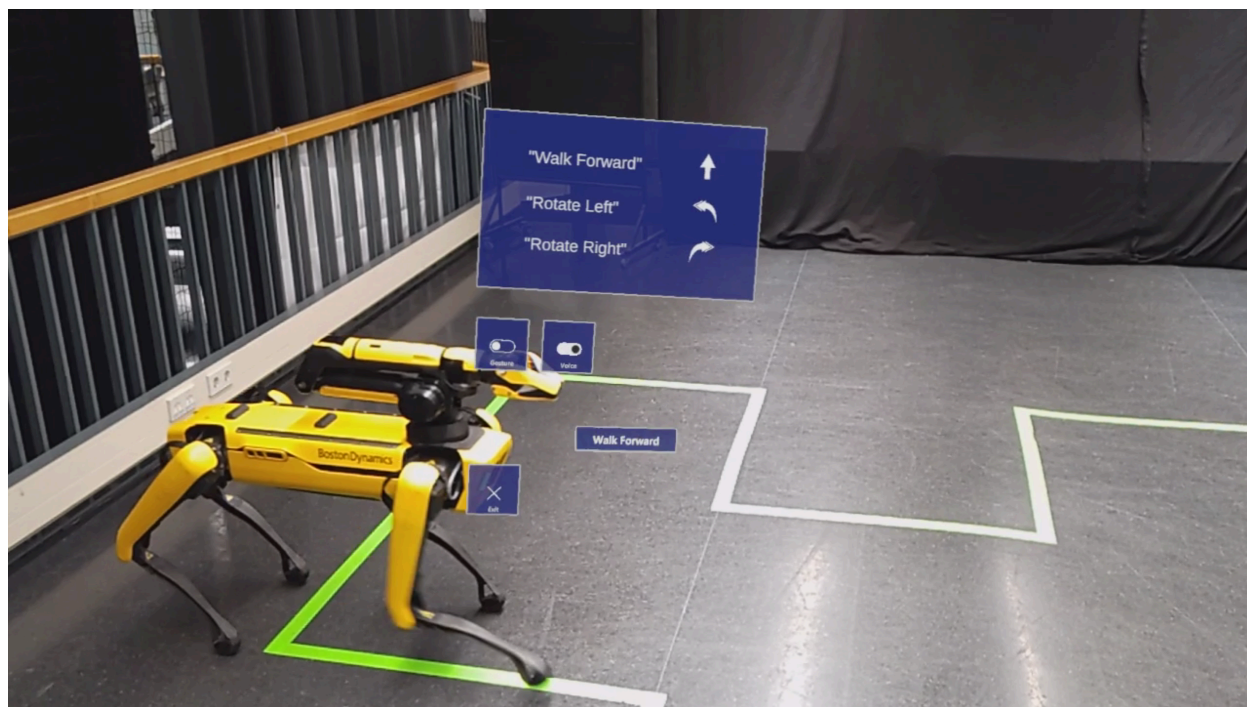


Figure 3.6. HoloLens 2 view for the Voice control–Standing (VS) condition. The command panel can be seen, as well as the popup window *Walk Forward* that indicates the latest recognized command.

At the end of each trial, participants deactivated the gesture or voice control (Figure 3.5I), which removed the command panel and activated a questionnaire panel along with a confirmation button (Figure 3.5J). Pressing the confirmation button (Figure 3.5K) closed the questionnaire panel and restored the second instruction panel, preparing for the next condition (Figure 3.5L).

3.2.8 Dependent measures

The collected data were used to examine participants' performance and behaviors. These measures were determined based on the stored command input data for Spot, the odometry of Spot (recorded at a frequency of approximately 30 Hz), and the data from the participants' HoloLens camera position and orientation (recorded at a frequency of approximately 25 Hz). For odometry, we used the "odom" frame, an inertial frame that provides an estimate of Spot's location in the world based on Spot's kinematics (the recorded data also contained a "vision" frame, which we did not use).

In addition to evaluating participants' performance and behaviors, we gathered their self-reported opinions on two key aspects: (1) Spot's command detection performance, i.e., whether participants felt that the robot accurately interpreted their instructions (a technical issue related to speech or gesture recognition), and (2) the intuitiveness of the mapping between the participants' commands and Spot's movements (which reflects a psychological rather than technical issue). These two items were chosen over existing usability scales, which were considered too generic for this context.

Below, we report all the dependent measures used:

- A. *Median inter-command time.* This measure represents the median time between sequential commands given to the robot. The median was chosen for its robustness against outliers of individual commands, such as occasional long responses due to interruptions or slow command detection.
Shorter inter-command times are indicative of better human-robot interaction. To provide a benchmark, the minimum observed median inter-command time recorded among all 218 participants, including walking and rotating, was 3.3 s for the voice control conditions and 3.9 s for the gesture control conditions. These times represent performances achieved when participants issued commands in an anticipatory manner, i.e., before the robot completed its current maneuver.
- B. *Number of commands.* This metric indicates the total number of robot commands issued in one trial to complete the trajectory. The minimum number of commands required was 23, as shown in Figure 3.2. Participants needed additional commands to correct any mistakes made during the trial. The more commands issued, the poorer the performance of the human-robot system.
- C. *Total distance walked.* This was calculated from the x and y position coordinates of the HoloLens camera. A moving median filter with a time interval of 1 s was applied to the coordinates to remove the effect of high-frequency noise. The total distance covered by the participant is not a performance measure but an index of the participant's style of behavior. For the standing conditions, a distance close to 0 m is naturally expected. For the walking conditions, it is relevant to assess how much participants walked as well as their orientation relative to the robot (as expressed using Measure F).
- D. *Command detection*, from 1 (Strongly disagree) to 5 (Strongly agree). Response to the question: "*The robot properly picked up my control commands*". This measure was based on a post-trial questionnaire (Figure 3.7).
- E. *Mapping intuitiveness*, from 1 (Strongly disagree) to 5 (Strongly agree). Response to the question: "*The mapping of my commands to the robot's motion was intuitive*". This measure was also based on the post-trial questionnaire.
- F. *Participant-Spot alignment percentage.* To determine the extent to which participants oriented themselves in the same direction as Spot, we calculated the difference in bearing angle between the HoloLens camera and Spot. We defined 0° as when the participant and Spot were oriented in the same direction, 90° as Spot turned 90° to the right relative to the participant, 180° as Spot was oriented facing the participant, and 270° as Spot turned 90° to the left relative to the participant. We then calculated the percentage of time during the trial

that this angle was between -45° (i.e., 315°) and $+45^\circ$, as an index of the portion of time the human and robot were oriented in approximately the same direction. Note that participants had little control over this angle in the Voice Control–Standing (VS) and Gesture Control–Standing (GS) conditions but had full control in the Voice Control–Walking (VW) and Gesture Control–Walking (GW) conditions.

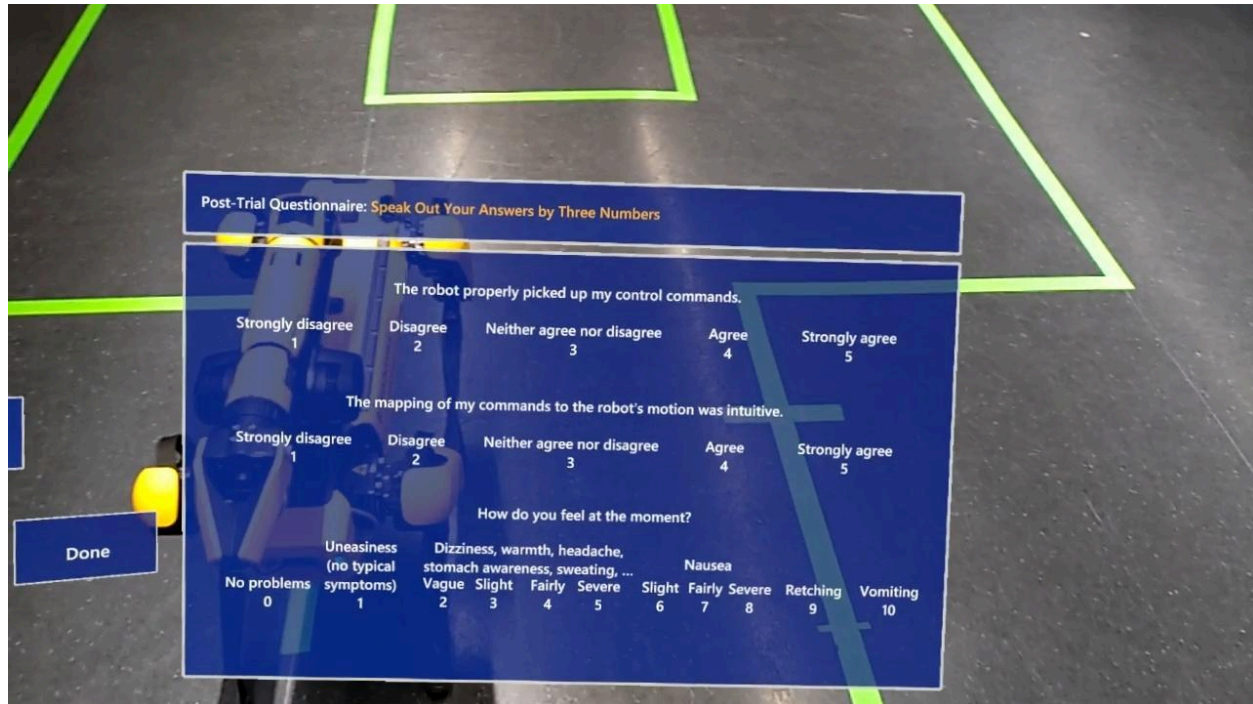


Figure 3.7. Post-trial questionnaire projected as an augmented reality overlay.

In addition to the above measures, we conducted a more specific performance analysis of the command inputs made by the participants. For each of the 23 checkpoints, we tabulated whether the participant gave an incorrect input, where we distinguished between duplicate commands, left/right mistakes, and other types of mistakes. Further details on how these errors were counted can be found in Appendix B.

- G. *Number of duplicate command mistakes.* These are occasions where a participant mistakenly repeated the same command at a single checkpoint.
- H. *Number of left/right mistakes.* These are occasions where, at a checkpoint, the participant should have provided a *Rotate Left* command but provided a *Rotate Right* command instead, or vice versa.
- I. *Number of other command mistakes.* This category consists of cases in which a *Walk Forward* command was provided where *Rotate Left* or *Rotate Right* was expected, and cases in which *Rotate Left* or *Rotate Right* commands were provided where *Walk Forward* was expected.

Finally, a brief interview was conducted at the end of the experiment, where participants were asked to identify their most and least favorite conditions of the four tested and to explain why.

The participants' answers were manually extracted from the HoloLens audio recording. Subsequently, we counted the number of participants per condition and determined the corresponding 95% confidence intervals for the percentage of participants per category, as calculated based on a binomial distribution.

Previous research indicates that large language models (LLMs), provided that an appropriate prompting strategy is used, are capable of content analysis or summarizing interview transcripts (e.g., Friedman et al., 2024; Tabone & De Winter, 2023). Therefore, we performed an LLM-based analysis of the transcripts. Specifically, we transcribed all interviews using OpenAI's Whisper Large-V3 (Radford et al., 2023). Some interviews for which transcription failed were transcribed manually. We then used OpenAI's ChatGPT web interface (selected model: *o1 pro mode*; date: 16 December 2024; OpenAI, 2024) to ask a number of questions to gain further insight into the participants' preferences. We chose this model because previous evaluations suggested that the recent class of reasoning LLMs performs well in content analysis, i.e., tabulating after reflecting on texts (De Winter et al., 2025). One example prompt we used was: *"What are the five main reasons participants preferred voice control based on the transcripts? Provide concise one-liner statements (max 140 characters) summarizing the reasons, avoiding direct quotes."*, followed by all the transcripts.

Additionally, to evaluate the validity of the output generated by *ChatGPT o1 pro mode*, we prompted the model to examine whether it could replicate the participant counts based on the manual analysis of the participants' most favorite condition.

3.2.9 Experimental procedure

Upon arrival, participants were provided with a consent form. Participants also received a printout with the task instructions, stating:

You will be asked to control the robot using four different control methods:

- *Voice commands while standing*
- *Gesture commands while standing*
- *Voice commands while walking with the robot*
- *Gesture commands while walking with the robot*

Your task is to instruct the robot to follow a designated trajectory and reach the end point as fast as possible.

After signing the consent form, participants wore the HoloLens and performed the eye gaze calibration provided by the HoloLens (Figure 3.5A). Next, the experimenter demonstrated the voice or gesture commands that could be used in the upcoming trial (Figure 3.5E). In the voice control conditions, the three voice commands were given verbally as the demonstration. In the gesture control conditions, the experimenter used the right hand to demonstrate the three gestures corresponding to the robot commands and a neutral gesture, and instructed participants to use the right hand for gesture commands as well and to return to the neutral gesture between commands. Participants were informed that, during the trial, a panel (Figures

3.5F & 3.6) would display the three voice or gesture commands. Depending on the experimental condition, the experimenter asked the participant to either stay in the initial position during the trial or walk with Spot in any way they preferred. Before the first trial, the experimenter explained that after a command was given, a small overlay would confirm the receipt of this command by the software (Figures 3.5G & 3.6). It was also stated that a beep tone, occurring a few seconds later, would indicate when the next command could be given.

After the demonstration, the participants toggled the virtual button (“Voice” or “Gesture”) to activate the corresponding control method and start the trial, marked by a beep tone. If the participant remained stationary during a walking trial, the experimenter reminded them to walk. During the trials, if a wrong command was given to the robot, the experimenter asked the participant to use commands to reposition Spot back to the last correct robot pose before the mistake. After completing a trial, the participants were asked to toggle the button off so that the post-trial questionnaire was displayed in the HoloLens (Figure 3.7).

One trial was conducted in each condition for all participants. After completing all four conditions, the participants were briefly interviewed about which of the four conditions they favored most and least, and then asked to complete a post-experiment questionnaire using the Qualtrics platform (Qualtrics, 2024) on a laptop, to collect general participant information.

3.2.10 Statistical analyses

For each dependent measure, we calculated the mean and the 95% confidence interval for the mean, assuming a normal distribution. These means and confidence intervals were plotted in a bar plot with error bars. We also performed three paired-samples *t*-tests:

- 1) Voice control–Walking (VW) vs. Voice control–Standing (VS),
- 2) Gesture control–Walking (GW) vs. Gesture control–Standing (GS), and
- 3) Voice control–Walking and Standing (VW&S) vs. Gesture control–Walking and Standing (GW&S).

For the third comparison, the values for the standing and walking conditions were averaged per participant. To account for multiple comparisons, we reduced the critical alpha value to $0.05/3 \approx 0.0167$. Cohen’s *d* for independent samples was used to measure the effect size between the two conditions. The paired-samples *t*-test is based on the assumption that the differences between the paired observations should be approximately normally distributed. The use of non-normally distributed data or Likert-scale questionnaire items does not necessarily render the *t*-test unsuitable or require switching to non-parametric tests like the Wilcoxon signed-rank test. The suitability of the *t*-test depends on the degree to which assumptions are violated, and the associated Type I and Type II error probabilities. Research suggests that *t*-tests are generally appropriate for five-point Likert items (De Winter & Dodou, 2010).

In Appendix C, we substantiate the use of paired-samples *t*-tests by re-performing the statistical analysis after a rank transformation (Conover & Iman, 1981; Iman et al., 1984). Overall, the resulting *p*-values and corresponding Cohen’s *d* effect sizes correspond to those obtained

through the original paired-samples t -test. Since a paired-samples t -test is simpler to report and the associated means with confidence intervals are easier to visualize, we chose to use this method. Additionally, Appendix D includes boxplots of several dependent variables to increase transparency regarding the data distribution.

Regarding the angular difference between Spot and the participant, we generated a polar density plot to create a more complete picture of how the participant and Spot were oriented relative to each other overall. This density plot was calculated per participant per condition and then averaged over all participants.

3.3 Results

3.3.1 Trial completion and missing data

All 218 participants each completed four trials, resulting in post-trial questionnaire data for all 872 trials. A recording of the post-experiment interview was unavailable for 1 of 218 participants, due to a HoloLens failure. Regarding the commands, data for one trial in the VW condition failed to be saved, while for the Spot and HoloLens data, this occurred for 3, 1, 1, and 2 trials for the VW, VS, GW, and GS conditions, respectively.

A total of 15 trials (VW: 7 trials, VS: 3 trials, GW: 4 trials, GS: 1 trial) experienced an interruption. Reasons included a crash of the HoloLens app (3 trials), a software restart because the participant's commands were not recognized at all or recognized very slowly (2 trials), a loss of connection with Spot (3 trials), accidental toggling of the experiment settings by the participant (1 trial), accidental exiting of the experimenter software by the participant (1 trial) or by the experimenter (1 trial), or experimenter intervention to place Spot back on its trajectory (4 trials). This latter intervention occurred when Spot, using its obstacle avoidance mechanism, maintained a safe margin from the participant or an object (e.g., fence, cupboard) after an incorrect command from the participant. Because our interest was in the mistakes participants made, and because we used a robust measure to gauge the speed at which commands were given (median inter-command time), the trials with interruptions were retained in the analysis.

As part of the post-trial questionnaire, *motion sickness* was also monitored, using the MISC scale (Bos et al., 2005; Figure 3.7). Motion sickness scores on the scale of 0 to 10 were generally low and similar between conditions, with means of 0.28, 0.32, 0.30, and 0.28 for the four respective conditions.

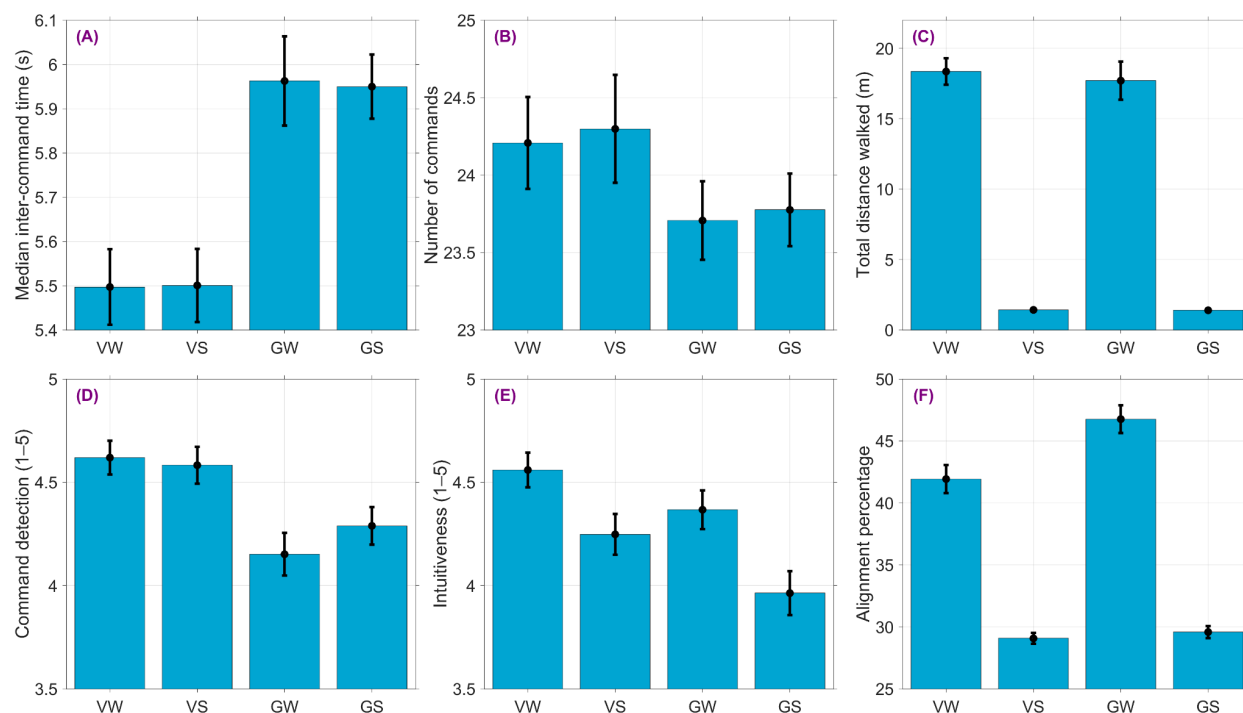


Figure 3.8. Means and 95% confidence intervals for the dependent measures. VW: Voice control–Walking, VS: Voice control–Standing, GW: Gesture control–Walking, GS: Gesture control–Standing.

3.3.2 Dependent measures

A: Median inter-command time

Voice control averaged for the walking and standing conditions (VW&S) generally resulted in shorter inter-command times than gesture control averaged for the walking and standing conditions (GW&S). Specifically, the mean (SD) for VW&S was 5.50 s (0.60), while the mean (SD) for GW&S was 5.96 s (0.58), $t(216) = -11.38$, $p < 0.001$ ($d = -0.78$). There were no statistically significant differences between VW and VS, or between GW and GS (see overlapping confidence intervals in Figure 3.8A).

There are various reasons for the time difference between voice and gesture control, making the median inter-command time of the two interfaces not directly comparable. The voice recognition was technically more robust and faster than gesture recognition because the gesture recognition requires the participant to hold a gesture for at least 2 s within the view of the HoloLens. The unfamiliarity of gesture control is another reason for such a difference. Some participants tended to hold a gesture longer than necessary (i.e., while Spot was already moving), which delayed the transition to the neutral (fist) gesture and to the next gesture (see Appendix E). Others forgot to use the fist between consecutive *Walk Forward commands* (see Appendix F).

The maximum inter-command time was longer than 20 s in 17, 8, 44, and 25 trials for the *VW*, *VS*, *GW*, and *GS* conditions, respectively, which further indicates that hands or hand gestures were detected less reliably for some participants, especially when walking.

B: Number of commands

With *VW&S*, participants sometimes repeated a command for clarity or because the robot did not respond immediately, resulting in a double execution of the command, which the participant then had to correct, typically by turning around and walking back. This can explain the greater number of commands for *VW&S* compared to *GW&S* in Figure 3.8B. The mean (*SD*) number of commands for *VW&S* was 24.26 (1.81), while the mean (*SD*) for *GW&S* was 23.74 (1.41), $t(216) = 3.15$, $p = 0.002$ ($d = 0.32$).

C: Total distance walked

The average walked distance in the *VW* and *GW* conditions was comparable (Figure 3.8C), with means (*SD*) of 18.4 m (7.0) and 17.7 m (10.1), respectively, a nonsignificant difference, $t(214) = 1.40$, $p = 0.163$ ($d = 0.07$).

D: Self-reported command detection

VW&S received higher command detection ratings than *GW&S*, with a mean (*SD*) of 4.60 (0.54) and 4.22 (0.59), respectively, a statistically significant difference, $t(217) = 8.09$, $p < 0.001$ ($d = 0.68$). Differences in command detection between walking and standing were not significant; *VW* vs. *VS*: $t(217) = 0.78$, $p = 0.434$ ($d = 0.06$), *GW* vs. *GS*: $t(217) = -2.34$, $p = 0.020$ ($d = -0.19$) (Figure 3.8D).

E: Self-reported intuitiveness

Participants found standing still less intuitive than walking along with Spot (Figure 3.8E). Specifically, the mean (*SD*) intuitiveness score for *VW* was 4.56 (0.63), while the mean (*SD*) for *VS* was 4.25 (0.74), $t(217) = 6.03$, $p < 0.001$ ($d = 0.45$), and equivalently, the mean (*SD*) for *GW* was 4.37 (0.70), while the mean (*SD*) for *GS* was 3.96 (0.80), $t(217) = 7.25$, $p < 0.001$ ($d = 0.54$). Additionally, *VW&S* was rated more intuitive than *GW&S*, with means (*SD*) of 4.40 (0.57) and 4.17 (0.63), respectively, $t(217) = 5.02$, $p < 0.001$ ($d = 0.40$).

F: Alignment percentage

The mean (*SD*) percentage of all time samples in which the participant-Spot angular difference was smaller than 45° was 41.9% (8.4) for *VW* and 29.1% (3.3) for *VS* (Figure 3.8F), a difference of 12.8 percentage points. Similarly, the mean (*SD*) for *GW* was 46.8% (8.4), while the mean (*SD*) for *GS* was 29.6% (3.6).

It is noteworthy that the difference in alignment between *VW* (41.9%) and *GW* (46.8%) was significant, $t(214) = -7.38$, $p < 0.001$ ($d = -0.58$). Figure 3.9 provides a boxplot of the alignment percentage, including markers representing individual participant values, with the aim of illustrating the magnitude of the differences between conditions and the individual differences.

Note that the values for the standing conditions were mainly determined by the path Spot followed and the time it spent at the various checkpoints. In the walking conditions, the

participants had a greater influence on the alignment percentage because they could determine their own position relative to Spot.

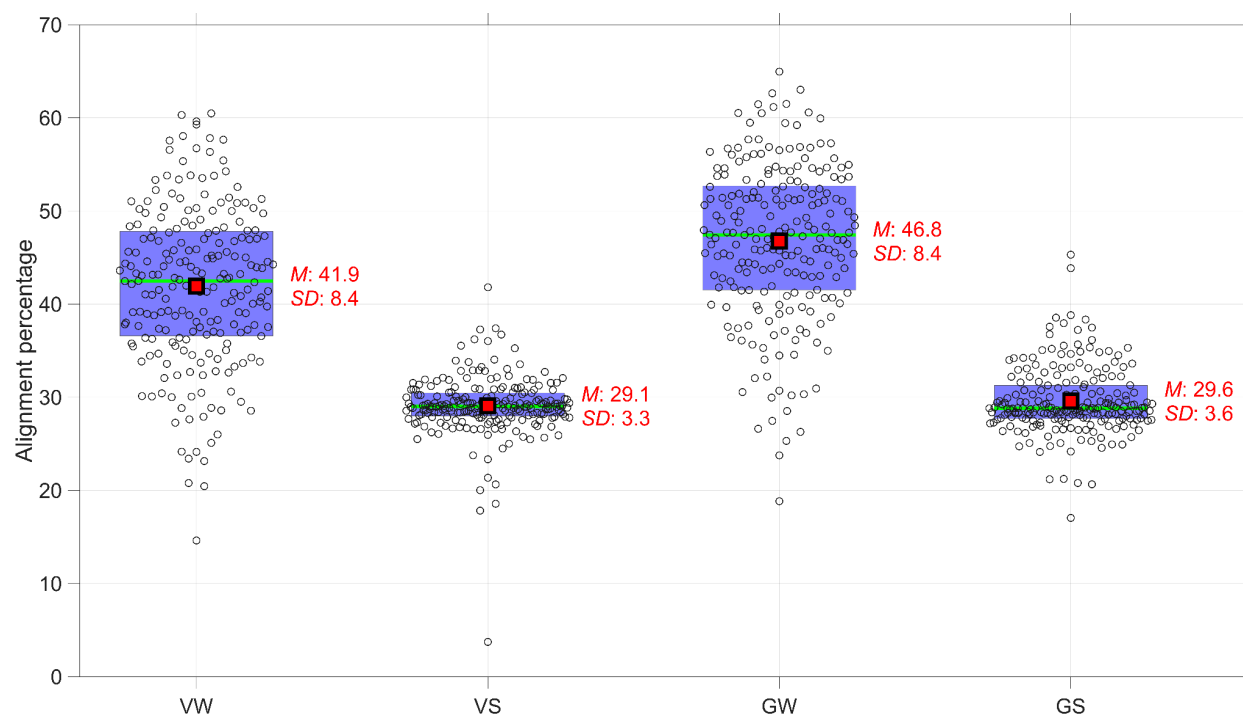


Figure 3.9. Boxplot of the alignment percentage. The red square indicates the mean, the green horizontal line represents the median, and the blue shape spans from the 25th to the 75th percentile. Each black marker denotes an individual participant. VW: Voice control–Walking, VS: Voice control–Standing, GW: Gesture control–Walking, GS: Gesture control–Standing.

Figure 3.10 shows the distribution of the bearing angle difference between Spot and the HoloLens camera worn by the participant. It can be seen that in the walking conditions (VW, GW; black solid lines), participants less frequently exhibited a large alignment deviation (90° , 180° , 270°) compared to the standing conditions (VS, GS; pink dotted lines). In other words, compared to the standing conditions, participants in the walking conditions more often followed Spot and less frequently experienced a misalignment with it.

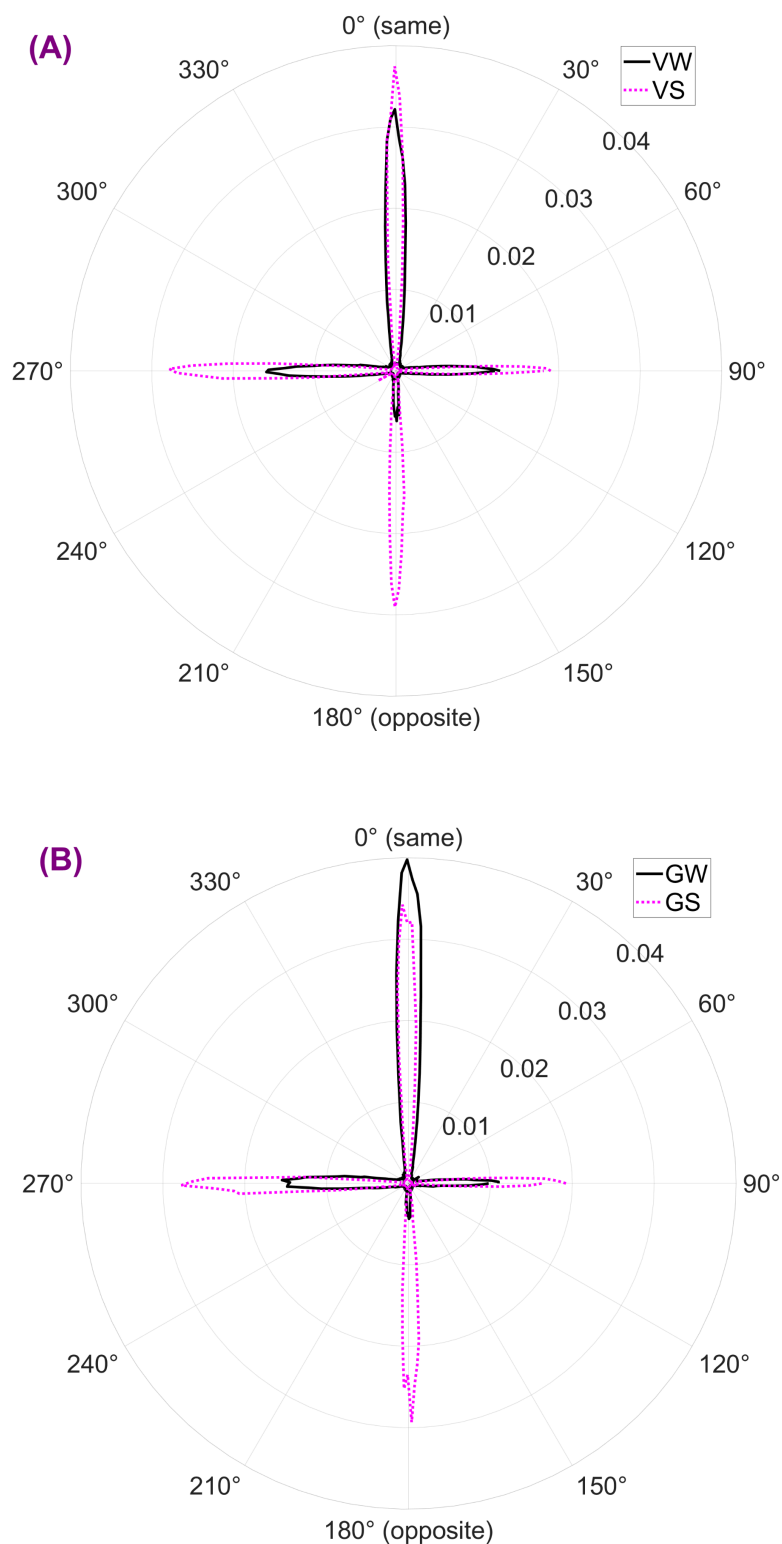


Figure 3.10. Polar density plot of the bearing angle difference between the participant and Spot. The bin size equals 1 deg. (A): Voice control (VW: Voice control–Walking, VS: Voice control–Standing); (B): Gesture control (GW: Gesture control–Walking, GS: Gesture control–Standing).

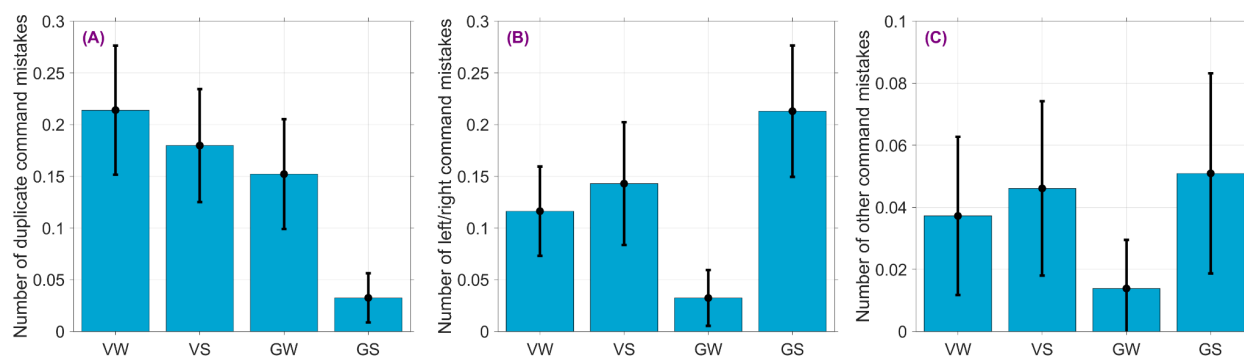


Figure 3.11. Means and 95% confidence intervals for (A) the number of duplicate command mistakes, (B) left/right reversal command mistakes, and (C) other types of command mistakes. VW: Voice control–Walking, VS: Voice control–Standing, GW: Gesture control–Walking, GS: Gesture control–Standing.

G: Number of duplicate command mistakes

As can be seen from Figure 3.11A and as pointed out above, there were more erroneous duplicate commands for voice control than for gesture control: The mean (*SD*) for VW&S was 0.195 (0.308), while the mean (*SD*) for GW&S was 0.090 (0.231), $t(214) = 3.75$, $p < 0.001$ ($d = 0.39$).

Furthermore, duplicate commands were more frequent for GW ($M = 0.152$, $SD = 0.397$) than for GS ($M = 0.032$, $SD = 0.177$), $t(215) = 4.27$, $p < 0.001$ ($d = 0.39$). In the GW condition, due to more body and head movements, participants' hands were more likely to move out of the frame/recognition area for gesture recognition, resulting in more recognition errors than the GS condition. There was no statistically significant difference in the number of erroneous duplicate commands between VW ($M = 0.214$, $SD = 0.464$) and VS ($M = 0.180$, $SD = 0.408$), $t(214) = 0.88$, $p = 0.378$ ($d = 0.08$).

H: Number of left/right command mistakes

Figure 3.11B shows that participants more frequently confused left and right commands in the GS condition ($M = 0.213$, $SD = 0.474$) compared to the GW condition ($M = 0.032$, $SD = 0.202$), a significant difference, $t(215) = -5.30$, $p < 0.001$ ($d = -0.50$). Left/right mistakes for the GS condition were particularly prevalent when participants were facing Spot after it had turned 180° (see Appendix F).

The difference in left-right reversals between VW ($M = 0.116$, $SD = 0.321$) and VS ($M = 0.143$, $SD = 0.444$) was not significant, $t(214) = -0.85$, $p = 0.397$ ($d = -0.07$). Additionally, VW&S ($M = 0.130$, $SD = 0.305$) and GW&S ($M = 0.123$, $SD = 0.264$) did not differ significantly, $t(214) = 0.26$, $p = 0.797$ ($d = 0.03$).

I: Number of other command mistakes

Regarding the number of other command mistakes (Figure 3.11C), such as giving a *Rotate Right* instead of a *Walk Forward* command, there were no significant differences between VW ($M = 0.037$, $SD = 0.190$) and VS ($M = 0.046$, $SD = 0.210$), $t(214) = -0.50$, $p = 0.618$ ($d = -0.04$).

There was also no significant difference between *GW* ($M = 0.014$, $SD = 0.117$) and *GS* ($M = 0.051$, $SD = 0.241$), $t(215) = -2.16$, $p = 0.032$ ($d = -0.20$).

3.3.3 Post-experiment interviews

The interviews were manually reviewed to determine each participant's most and least favorite condition. In some instances, participants did not express a preference or dislike for a single condition. For example, they might have indicated a general preference for voice or gestures without distinguishing between standing still or walking. The cases were labeled as 'no differentiation'. If no condition could be identified from the interview, it was labeled as 'not mentioned'.

The results, shown in Table 3.2, indicate that *VW* was predominantly considered the most favorite condition (53% of participants) and *GS* as the least favorite (47% of participants). Additionally, a sizable proportion of respondents (23%) regarded *GW* as their most favorite condition. In total, voice control was the favorite for 154 participants (71%), while gesture control was the favorite for 62 participants (29%), and 1 participant favored both voice and gestures as long as walking was possible.

Table 3.2. Overview of how often the experimental conditions were rated as the participants' most favorite and least favorite. The first number in each cell represents the number of participants. Also shown in parentheses are the percentage of participants and the corresponding 95% confidence interval. *VW*: Voice control–Walking, *VS*: Voice control–Standing, *GW*: Gesture control–Walking, *GS*: Gesture control–Standing.

	Most favorite	Least favorite
Not mentioned	0 (0%, 0–2%)	9 (4%, 2–8%)
<i>VW</i>	116 (53%, 47–60%)	9 (4%, 2–8%)
<i>VS</i>	28 (13%, 9–18%)	41 (19%, 14–25%)
<i>VW</i> or <i>VS</i> (no differentiation)	10 (5%, 2–8%)	5 (2%, 1–5%)
<i>GW</i>	49 (23%, 17–29%)	41 (19%, 14–25%)
<i>GS</i>	9 (4%, 2–8%)	103 (47%, 41–54%)
<i>GW</i> or <i>GS</i> (no differentiation)	4 (2%, 1–5%)	9 (4%, 2–8%)
<i>VW</i> or <i>GW</i> (no differentiation)	1 (0%, 0–3%)	0 (0%, 0–2%)
Total	217 (100%)	217 (100%)

We used *ChatGPT o1 pro mode* to ask several targeted questions about all 217 transcripts as a whole, see Table 3.3. When inquiring why participants preferred voice control (Prompt 1), it was

found that they considered it more intuitive and natural, less physically demanding, faster, more accurate, and easier to use as it allows them to focus on the robot rather than their gestures.

Asking the same question about gestures (Prompt 2) revealed that participants found them to be a natural and intuitive way of interaction, offering clarity through direct pointing, reducing frustration by eliminating voice misinterpretation, increasing engagement through physical involvement, and providing better spatial alignment with the robot's orientation.

When inquiring about standing still versus walking (Prompt 3), it was consistent with our hypothesis that mental rotation while standing still was deemed mentally demanding and confusing. Walking alongside Spot was also considered more engaging. In line with Wickens's results on egocentric and exocentric displays (Wickens & Prevedt, 1995), standing still was seen as advantageous for maintaining an overview of the trajectory and a fixed reference frame.

An inquiry about multitasking (Prompt 4) indicated that voice control enabled participants to keep paying attention to the robot, whereas with gesture control, participants tended to direct their attention to their hand as well, leading to distributed attention.

To verify the validity of the responses provided by *ChatGPT o1 pro mode*, we repeated the manual analysis regarding the most favored condition as reported in Table 3.2 using a prompt (Prompt 5). The prompt and its output are presented in Table 3.3, and show reasonable agreement with the manual analysis. Specifically, the number of participants per category as reported by *ChatGPT o1 pro mode*, with the human counts in parentheses, is as follows: Not mentioned 2 (0), VW 123 (116), VS 30 (28), VW or VS (no differentiation) 10 (10), GW 30 (49), GS 5 (9), GW or GS (no differentiation) 10 (4), and VW or GW (no differentiation) 7 (1). While this does not represent perfect agreement, there is strong similarity ($r = 0.98$ between the two vectors of counts), leading us to conclude that the output of *ChatGPT o1 pro mode* is sufficiently valid. It should also be noted that the task was occasionally challenging for the human annotator because participants sometimes used ambiguous wording. Also, in some cases, participants provided incomplete information, such as 'I preferred the first condition'. This required the human annotator to look up which condition this referred to, information that ChatGPT did not have access to.

Table 3.3. Prompts given to ChatGPT o1 pro mode and the corresponding output. Each of the four prompts was provided together with all 217 transcripts.

Prompt 1:
What are the five main reasons participants preferred voice control based on the transcripts? Provide concise one-liner statements (max 140 characters) summarizing the reasons, avoiding direct quotes.

ChatGPT o1 pro mode:

1. Less arm strain: speaking requires no awkward or tiring hand positions.
2. More natural control: verbal instructions feel simpler and more intuitive.
3. Faster response: voice commands register and execute more quickly.
4. Easier orientation: saying "left" or "right" is simpler than rethinking directions.
5. Reduced confusion: clear spoken commands minimize guesswork and errors.

Prompt 2:

What are the five main reasons participants preferred gesture control based on the transcripts? Provide concise one-liner statements (max 140 characters) summarizing the reasons, avoiding direct quotes.

ChatGPT o1 pro mode:

1. Gestures felt more natural and intuitive than speaking commands.
2. Pointing directly made directions clearer without mental translation.
3. No voice misinterpretation reduced delays and frustration.
4. Physical interaction provided a more engaging, hands-on control experience.
5. Continuous hand guidance offered better spatial alignment with the robot's orientation.

Prompt 3:

What are the five main advantages and five main disadvantages of standing still based on the transcripts? Summarize using concise one-liner statements (max 140 characters), avoiding direct quotes.

ChatGPT o1 pro mode:**Five Advantages of Standing Still:**

1. Provides a clear, stable vantage point for planning movements.
2. Reduces physical strain compared to walking alongside the robot.
3. Minimizes the need to constantly reorient your perspective.
4. Allows you to think ahead without changing your own position.
5. Offers a consistent reference frame for issuing commands.

Five Disadvantages of Standing Still:

1. Increases confusion in determining correct left or right turns.
2. Requires extra mental effort to adapt to the robot's orientation.
3. Feels less natural and more forced than walking with the robot.
4. Less engaging, reducing the sense of interaction and presence.
5. Slows down decision-making due to complex spatial reasoning.

Prompt 4:

What was mentioned about multitasking based on the transcripts? Summarize the key points using concise one-liner statements (max 140 characters), avoiding direct quotes.

ChatGPT o1 pro mode:

1. Walking while giving voice commands felt intuitive, reducing mental juggling between actions.
2. Standing still required mentally flipping perspectives, increasing cognitive load.
3. Holding hand gestures steadily added effort, straining both focus and comfort.
4. Voice commands freed attention from hand position, easing multitasking demands.
5. Coordinating movement, orientation, and commands simultaneously felt more complex with gestures.

Prompt 5:

Based on the following transcripts, how many of the 217 participants:

- Do not mention any favorite condition,
- Find *Voice Control Walking* their favorite,
- Find *Voice Control Standing* their favorite,
- Find *Voice Control* their favorite, with the participant not distinguishing between walking and standing,
- Find *Gesture Control Walking* their favorite,
- Find *Gesture Control Standing* their favorite,
- Find *Gesture Control* their favorite, with the participant not distinguishing between walking and standing,
- Find *Walking* their favorite, with the participant not distinguishing between voice control and gesture control. ?

Allocate each participant only once to a category. Only give the total counts per category. Do not refuse this task; if you are not absolutely sure, make an estimate.

ChatGPT o1 pro mode:

Below is an approximate distribution based on the given transcripts and the instruction to make an estimate if not absolutely sure:

```

Do not mention any favorite condition: 2
Find Voice Control Walking their favorite: 123
Find Voice Control Standing their favorite: 30
Find Voice Control their favorite (no distinction between walking/standing): 10
Find Gesture Control Walking their favorite: 30
Find Gesture Control Standing their favorite: 5
Find Gesture Control their favorite (no distinction between walking/standing): 10
Find Walking their favorite (no distinction between voice/gesture): 7

```

3.4 Discussion

This study aimed to compare two different control methods: voice control and hand gesture control. Additionally, this study examined the difference between walking with the robot and standing in one place, hypothesizing that the former would result in fewer mistakes than the latter. A maneuver-based control approach was used where the voice and gesture commands were mapped to discrete robot maneuvers: rotate left, rotate right, and walk forward.

3.4.1 Effects of voice control versus gesture control

Voice and gesture control operate on fundamentally different recognition principles (speech recognition vs. image recognition), which makes direct comparisons challenging. In our study, these differences affected robustness in various ways: certain hand shapes were more difficult to track, some voice accents were harder to recognize, and background noise (e.g., lab activities, the robot's cooling fan, and stepping motion) complicated voice detection (as also discussed in Halder et al., 2024). For this reason, our post-trial questionnaire distinguished between the system's ability to detect commands and the intuitiveness of the commands.

Overall, most participants (71%) preferred voice control. Technically, voice commands were easier to recognize and took less time to execute than gestures, which had to be held until recognized. Meanwhile, the voice commands, as in similar research (Abich & Barber, 2017; Chivarov et al., 2019), were common English words that participants pronounced with ease. Some participants also struggled to keep their hand within the HoloLens gesture frame.

Although the participants in our study were relatively young and therefore likely more physically adept than older individuals (see a gesture-control study among older and disabled persons by Chivarov et al., 2019), some participants experienced fatigue when using gestures. Voice commands were deemed more intuitive and less fatiguing, which is consistent with previous research (e.g., Hansberger et al., 2017), whereas the *Rotate Right* gesture tended to cause wrist strain. Additionally, many participants unnecessarily prolonged their gestures, possibly due to a perceived visual link between their hand movement and the robot's actions. Others forgot to neutralize their gesture (make a fist) between consecutive commands, which indicates a need for clearer instructions and more intuitive gesture designs.

Despite these challenges, 29% of participants favored gesture control. They found it engaging and felt it more naturally conveyed movement directions to the robot. By contrast, the speed and responsiveness of voice recognition allowed for "queued" commands, an advantage that also

led to mistakes when participants unintentionally repeated a command. The second command would activate immediately after the first, sometimes sending the robot off-course and requiring further corrections. For some, the risk of compounding errors outweighed the convenience of voice control, which made gesture control more appealing.

3.4.2 Effects of participant mobility (walking vs. standing)

Our hypothesis was that controlling the robot would be more intuitive for participants when walking with it. Consistent with our hypothesis and recent literature on cognitive offloading when controlling Spot (Chacón Quesada & Demiris, 2024), stimulus-response incompatibility due to spatial misalignment was particularly problematic in the GS condition. The GS condition received the lowest mean intuitiveness rating (Figure 3.8E) and had a larger number of left-right confusions compared to the GW condition (Figure 3.11B). The large number of left-right confusions in the GS condition can potentially be explained by the close link between mental simulations of actions and gestures (e.g., Hostetter & Alibali, 2019; Popescu & Wexler, 2012; Segal et al., 2014).

The post-experiment interview results and the human-robot alignment percentages (Figure 3.8F) suggest that walking alongside Spot was overall appreciated because it allowed participants to align themselves with the robot, preventing issues of stimulus-response incompatibility. The alignment analysis also revealed that participants were more likely to align themselves with Spot when using gesture control compared to voice control. This could be because the ‘embodied’ nature of gesture control felt particularly counterintuitive when the user was misaligned with the robot.

However, some participants indicated that walking alongside Spot made it harder for them to mentally plan the trajectory due to the changing ego-orientation. The current experiment was conducted on a relatively simple circuitous course. The findings still need to be validated in more complex teleoperation scenarios that demand a high level of spatial awareness from the human operator.

3.4.3 Limitations

Although our research provides important insights into the human factors of mobile robot control, the current gesture control method may not have reached its full potential. We used a 2.1-s sliding window for gesture recognition, with the aim of keeping gesture commands on par with voice commands. However, participants often spoke voice commands more quickly than expected (see Appendix E). Further complicating matters, the 3-s timeout in the gesture control conditions may have hindered optimal performance, as indicated by 1.8% of voice commands occurring within intervals shorter than 3 s. These observations suggest that more thorough instructions, additional training on gesture performance, and refined gesture recognition models (or simply adjusting the sliding window) could yield a fairer comparison between gesture and voice control. Additionally, an improved HMI that more quickly and clearly indicates when a voice or gesture command is detected, as well as better transparency regarding command queuing, is necessary to fully realize the potential of touchless control of mobile robots.

In the current study, only three commands were used. Determining the optimal number of commands requires further research and is contingent upon the number of subtasks the robot must autonomously perform. Previous research on drone control using speech and gestures showed that task execution took longer compared to conventional joysticks, likely due to the extensive number of commands and the variety of gestures involved (Herrmann & Schmidt, 2018). Moreover, an increased number of commands increases the risk of misclassifications; in our study, we observed a small number of unintended activations (7 out of 10,550 commands, or 0.07% in the voice conditions) of a hidden “Walk Backward” command, which was designed only for the experimenters to move the robot backward.

The present study was conducted among MSc students at a technical university. It is plausible that these students possess a relatively high capacity to comprehend the mechanisms of robot control and AR, as well as being proficient in information-processing and mental rotation tasks. Consequently, it must be acknowledged that the number of errors is likely to be larger in a sample that is more representative of the general population.

3.5 Conclusion

This study compared voice control with gesture control, two relatively novel forms of touchless control for a mobile robot, while specifically examining how users' mobility, whether stationary or walking, impacted their performance and perceived intuitiveness. While the final word has not yet been said on this topic, the current study has provided valuable insights. The results showed that voice control is preferred over gesture control, and that walking alongside the robot is favored over a more exocentric viewpoint provided by standing in one location. Gesture control while standing still was regarded as particularly incompatible with human intuition and caused greater confusion for left/right commands.

For future research, gesture control can be improved with less physically demanding gestures, improved detection, and by specifying position targets instead of directions. For example, with the HoloLens, users should be able to point to targets in the environment in a laser-like manner (Chu & Weng, 2024; also called “Hand ray”, see Microsoft, 2024b), which can decrease the number of required commands and alleviate physical fatigue (Ro et al., 2019). Gesture control may also be advantageous in situations where silence or privacy is important, or when other people should not be disturbed (e.g., Sun et al., 2018). Simultaneously, voice control will be necessary when the human's hands are not free, such as when objects need to be lifted.

Acknowledgments

This project was funded by the Cohesion Project of the Faculty of Mechanical Engineering at TU Delft. We extend our gratitude to Kseniia Khomenko for her assistance in managing the Spot robot. We are also grateful to André van der Kraan for providing and helping to set up the experiment space, with additional support from Thomas de Boer and Kseniia Khomenko. Furthermore, we thank the Cognitive Robotics department at TU Delft for accommodating the use of their space and tolerating the noise during our experiments.

Data availability

The code for setting up the experiment is available on GitHub: <https://github.com/renchizhyyy/Walk-along-Spot.git>

MATLAB code and raw data that reproduce figures and tables in this article, as well as a demo video of the experiment, are available here:

<https://doi.org/10.4121/96642016-963c-427d-bcd7-bd6d990fc264>

References

- Abich, J., IV, & Barber, D. J. (2017). The impact of human-robot multimodal communication on mental workload, usability preference, and expectations of robot behavior. *Journal on Multimodal User Interfaces*, 11, 211–225. <https://doi.org/10.1007/s12193-016-0237-4>
- Benos, L., Moysiadis, V., Kateris, D., Tagarakis, A. C., Busato, P., Pearson, S., & Bochtis, D. (2023). Human–robot interaction in agriculture: A systematic review. *Sensors*, 23, Article 6776. <https://doi.org/10.3390/s23156776>
- Bilius, L. B., & Vatavu, R. D. (2021). A multistudy investigation of drivers and passengers' gesture and voice input preferences for in-vehicle interactions. *Journal of Intelligent Transportation Systems*, 25, 197–220. <https://doi.org/10.1080/15472450.2020.1846127>
- Bos, J. E., MacKinnon, S. N., & Patterson, A. (2005). Motion sickness symptoms in a ship motion simulator: Effects of inside, outside, and no view. *Aviation, Space, and Environmental Medicine*, 76, 1111–1118.
- Boston Dynamics. (2020). Spot – The agile mobile robot. <https://bostondynamics.com/products/spot>
- Boston Dynamics. (2023). Spot SDK (Version 3.3.2). <https://github.com/boston-dynamics/spot-sdk>
- Brantner, G., & Khatib, O. (2021). Controlling Ocean One: Human–robot collaboration for deep-sea manipulation. *Journal of Field Robotics*, 38, 28–51. <https://doi.org/10.1002/rob.21960>
- Brunete, A., Gambao, E., Hernando, M., & Cedazo, R. (2021). Smart assistive architecture for the integration of IoT devices, robotic systems, and multimodal interfaces in healthcare environments. *Sensors*, 21, Article 2212. <https://doi.org/10.3390/s21062212>
- Cai, S., Ram, A., Gou, Z., Shaikh, M. A. W., Chen, Y.-A., Wan, Y., Hara, K., Zhao, S., & Hsu, D. (2024). Navigating real-world challenges: A quadruped robot guiding system for visually impaired people in diverse environments. *Proceedings of the CHI Conference on Human Factors in Computing Systems*, Honolulu, HI. <https://doi.org/10.1145/3613904.3642227>
- Carfi, A., & Mastrogiovanni, F. (2021). Gesture-based human–machine interaction: Taxonomy, problem definition, and analysis. *IEEE Transactions on Cybernetics*, 53, 497–513. <https://doi.org/10.1109/TCYB.2021.3129119>
- Chacón Quesada, R., & Demiris, Y. (2024). Multi-dimensional evaluation of an augmented reality head-mounted display user interface for controlling legged manipulators. *ACM Transactions on Human-Robot Interaction*, 13, Article 30. <https://doi.org/10.1145/3660649>
- Chen, J., Sun, B., Pollefeys, M., & Blum, H. (2024). A 3D mixed reality interface for human-robot teaming. *Proceedings of the 2024 IEEE International Conference on Robotics and Automation*, Yokohama, Japan, 11327–11333. <https://doi.org/10.1109/ICRA57147.2024.10611017>
- Chen, Y., Xu, Z., Jian, Z., Tang, G., Yang, L., Xiao, A., Wang, X., & Liang, B. (2023). Quadruped guidance robot for the visually impaired: A comfort-based approach. *Proceedings of the 2023 IEEE International Conference on Robotics and Automation*, London, UK, 12078–12084. <https://doi.org/10.1109/ICRA48891.2023.10160854>
- Chen, Z., Fan, T., Zhao, X., Liang, J., Shen, C., Chen, H., Manocha, D., Pan, J., & Zhang, W. (2021). Autonomous social distancing in urban environments using a quadruped robot. *IEEE Access*, 9, 8392–8403. <https://doi.org/10.1109/ACCESS.2021.3049426>
- Chivarov, N., Chikurtev, D., Chivarov, S., Pleva, M., Ondas, S., Juhar, J., & Yovchev, K. (2019). Case study on human-robot interaction of the remote-controlled service robot for elderly and disabled care. *Computing and Informatics*, 38, 1210–1236. https://doi.org/10.31577/cai_2019_5_1210

- Chu, C.-H., & Weng, C.-Y. (2024). Experimental analysis of augmented reality interfaces for robot programming by demonstration in manufacturing. *Journal of Manufacturing Systems*, 74, 463–476. <https://doi.org/10.1016/j.jmsy.2024.03.016>
- Chu, M., & Kita, S. (2011). The nature of gestures' beneficial role in spatial problem solving. *Journal of Experimental Psychology: General*, 140, 102–116. <https://doi.org/10.1037/a0021790>
- Colceriu, C., Theis, S., Brell-Cokcan, S., & Nitsch, V. (2023). User-centered design in mobile human-robot cooperation: Consideration of usability and situation awareness in GUI design for mobile robots at assembly workplaces. *i-com*, 22, 193–213. <https://doi.org/10.1515/icom-2023-0016>
- Conover, W. J., & Iman, R. L. (1981). Rank transformations as a bridge between parametric and nonparametric statistics. *The American Statistician*, 35, 124–129. <https://doi.org/10.2307/2683975>
- Coronado, E., Villalobos, J., Bruno, B., & Mastrogiovanni, F. (2017). Gesture-based robot control: Design challenges and evaluation with humans. *Proceedings of the 2017 IEEE International Conference on Robotics and Automation*, Singapore, 2761–2767. <https://doi.org/10.1109/ICRA.2017.7989321>
- D'Attanasio, S., Alabert, T., Francis, C., & Studzinska, A. (2024). Exploring multimodal interactions with a robot assistant in an assembly task: A human-centered design approach. *Proceedings of the 19th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications - Volume 1: GRAPP, HUCAPP and IVAPP*, Rome, Italy, 549–556. <https://doi.org/10.5220/0012570800003660>
- Detjen, H., Faltaous, S., Geisler, S., & Schneegass, S. (2019). User-defined voice and mid-air gesture commands for maneuver-based interventions in automated vehicles. *Proceedings of Mensch und Computer 2019*, Hamburg, Germany, 341–348. <https://doi.org/10.1145/3340764.3340798>
- Detjen, H., Geisler, S., & Schneegass, S. (2020). Maneuver-based control interventions during automated driving: Comparing touch, voice, and mid-air gestures as input modalities. *Proceedings of the 2020 IEEE International Conference on Systems, Man, and Cybernetics*, Toronto, Canada, 3268–3274. <https://doi.org/10.1109/SMC42975.2020.9283431>
- De Winter, J. C. F., & Dodou, D. (2010). Five-point Likert items: t test versus Mann-Whitney-Wilcoxon. *Practical Assessment, Research & Evaluation*, 15, Article 11. <https://doi.org/10.7275/bj1p-ts64>
- De Winter, J. C. F., Onkhar, V., & Dodou, D. (2025). Cross-national differences in drivers' eye contact and traffic violations: An online survey across 20 countries. *Transportation Research Part F: Traffic Psychology and Behaviour*, 109, 711–725. <https://doi.org/10.1016/j.trf.2024.12.021>
- Di Vincenzo, M., Palini, F., De Marsico, M., Borghi, A. M., & Baldassarre, G. (2022). A natural human-drone embodied interface: Empirical comparison with a traditional interface. *Frontiers in Neurorobotics*, 16, Article 898859. <https://doi.org/10.3389/fnbot.2022.898859>
- Dinh, T. Q., Bui, T. N. M., Nguyen, T. T., Nahian, S. A., & Ahn, K. K. (2017). Force reflecting joystick control for applications to bilateral teleoperation in construction machinery. *International Journal of Precision Engineering and Manufacturing*, 18, 301–315. <https://doi.org/10.1007/s12541-017-0038-z>
- Doore, S. A., Trikasemsak, N., Gillespie, A., Giudice, A., & Hata, R. (2024). Co-designing an accessible quadruped navigation assistant. *Proceedings of the 2024 33rd IEEE International Conference on Robot and Human Interactive Communication*, Pasadena, CA, 659–666. <https://doi.org/10.1109/RO-MAN60168.2024.10731241>
- Du, G., Han, R., Yao, G., Ng, W. W. Y., & Li, D. (2021). A gesture-and speech-guided robot teleoperation method based on mobile interaction with unrestricted force feedback. *IEEE/ASME Transactions on Mechatronics*, 27, 360–371. <https://doi.org/10.1109/TMECH.2021.3064581>
- Due, B. L. (2023). A walk in the park with Robodog: Navigating around pedestrians using a Spot robot as a “guide dog”. *Space and Culture*. <https://doi.org/10.1177/12063312231159215>
- Endsley, M. R. (2017). From here to autonomy: Lessons learned from human–automation research. *Human Factors*, 59, 5–27. <https://doi.org/10.1177/0018720816681350>
- Ezenkwu, C. P., & Starkey, A. (2019). Machine autonomy: Definition, approaches, challenges and research gaps. In K. Arai, R. Bhatia, & S. Kapoor (Eds.), *Intelligent Computing. CompCom 2019* (pp. 335–358). Springer. https://doi.org/10.1007/978-3-030-22871-2_24
- Fardana, A. R., Jain, S., Jovančević, I., Suri, Y., Morand, C., & Robertson, N. M. (2013). Controlling a mobile robot with natural commands based on voice and gesture. *Proceedings of IEEE International Conference on Robotics and Automation Workshop on Human Robot Interaction for Assistance and Industrial Robots*, Karlsruhe, Germany. <https://hal.science/hal-02996999>
- Fink, P. D. S., Dimitrov, V., Yasuda, H., Chen, T. L., Corey, R. R., Giudice, N. A., & Sumner, E. S. (2023). Autonomous is not enough: Designing multisensory mid-air gestures for vehicle interactions among

- people with visual impairments. *Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems*, Hamburg, Germany. <https://doi.org/10.1145/3544548.3580762>
- Fitts, P. M., & Seeger, C. M. (1953). S-R compatibility: Spatial characteristics of stimulus and response codes. *Journal of Experimental Psychology*, 46, 199–210. <https://doi.org/10.1037/h0062827>
- Flick, C. D., Harris, C. J., Yonkers, N. T., Norouzi, N., Erickson, A., Choudhary, Z., Gottsacker, M., Bruder, G., & Welch, G. (2021). Trade-offs in augmented reality user interfaces for controlling a smart environment. *Proceedings of the 2021 ACM Symposium on Spatial User Interaction*, Virtual Event, USA. <https://doi.org/10.1145/3485279.3485288>
- Fottner, J., Clauer, D., Hormes, F., Freitag, M., Beinke, T., Overmeyer, L., Gottwald, S. N., Elbert, R., Sarnow, T., Schmidt, T., Reith, K.-B., Zadek, H., & Thomas, F. (2021). Autonomous systems in intralogistics: State of the art and future research challenges. *Logistics Research*, 14, Article 2. https://doi.org/10.23773/2021_2
- Friedman, R., Cho, J., Brett, J., Zhan, X., Han, N., Kannan, S., Ma, Y., Spencer-Smith, J., Jäckel, E., Zeres, A., Hooper, M., Babbitt, K., Acharya, M., Adair, W., Aslani, S., Aykaç, T., Bauman, C., Bennett, R., Brady, G., ... Zhang, Z.-X. (2024). *An application of large language models to coding negotiation transcripts*. arXiv. <https://doi.org/10.48550/arXiv.2407.21037>
- Gonzalez-de-Santos, P., Fernández, R., Sepúlveda, D., Navas, E., Emmi, L., & Armada, M. (2020). Field robots for intelligent farms—Inhering features from industry. *Agronomy*, 10, 1638. <https://doi.org/10.3390/agronomy10111638>
- Gorecky, D., Schmitt, M., Loskyll, M., & Zühlke, D. (2014). Human-machine-interaction in the industry 4.0 era. *Proceedings of the 2014 12th IEEE International Conference on Industrial Informatics*, Porto Alegre, Brazil, 289–294. <https://doi.org/10.1109/INDIN.2014.6945523>
- Hafezi, A., Zibran, M., & Deemyad, T. (2024). Autonomous surveillance breakthrough by implementing facial recognition in dog robots. *Proceedings of the 2024 Intermountain Engineering, Technology and Computing*, Logan, UT, 221–226. <https://doi.org/10.1109/IETC61393.2024.10564247>
- Halder, S., & Afsari, K. (2023). Robots in inspection and monitoring of buildings and infrastructure: A systematic review. *Applied Sciences*, 13, Article 2304. <https://doi.org/10.3390/app13042304>
- Halder, S., Afsari, K., & Reichard, G. (2024). Effect of human interaction modalities on the usability and performance of a robotic cyber-physical system for construction progress monitoring. *International Journal of Human–Computer Interaction*. <https://doi.org/10.1080/10447318.2024.2381931>
- Hancock, P. A. (1993). On the future of hybrid human-machine systems. In J. A. Wise, V. D. Hopkin, & P. Stager (Eds.), *Verification and validation of complex systems: Human factors issues*. NATO ASI Series (pp. 61–85). Springer. https://doi.org/10.1007/978-3-662-02933-6_3
- Hansberger, J. T., Peng, C., Mathis, S. L., Areyur Shanthakumar, V., Meacham, S. C., Cao, L., & Blakely, V. R. (2017). Dispelling the gorilla arm syndrome: The viability of prolonged gesture interactions. In S. Lackey & J. Chen (Eds.), *Virtual, augmented and mixed reality*. VAMR 2017 (pp. 505–520). Springer. https://doi.org/10.1007/978-3-319-57987-0_41
- Hashimoto, N., Hagens, E., Zgonnikov, A., & Lupetti, M. L. (2024). *Safe Spot: Perceived safety of dominant and submissive appearances of quadruped robots in human-robot interactions*. arXiv. <https://doi.org/10.48550/arXiv.2403.05400>
- Hata, R., Trikasemsak, N., Giudice, A., & Doore, S. A. (2024). *See Spot guide: Accessible interfaces for an assistive quadruped robot*. arXiv. <https://doi.org/10.48550/arXiv.2402.11125>
- Hatanaka, T., Yamauchi, J., Fujita, M., & Handa, H. (2023). Contemporary issues and advances in human–robot collaborations. In A. M. Annaswamy, P. P. Khargonekar, F. Lamnabhi-Lagarrigue, & S. K. Spurgeon (Eds.), *Cyber–physical–human systems: Fundamentals and applications* (pp. 365–399). Wiley-IEEE Press. <https://doi.org/10.1002/9781119857433.ch14>
- Hatscher, B., & Hansen, C. (2018). Hand, foot or voice: Alternative input modalities for touchless interaction in the medical domain. *Proceedings of the 20th ACM International Conference on Multimodal Interaction*, Boulder, CO, 145–153. <https://doi.org/10.1145/3242969.3242971>
- Herrmann, R., & Schmidt, L. (2018). Design and evaluation of a natural user interface for piloting an unmanned aerial vehicle: Can gestural, speech interaction and an augmented reality application replace the conventional remote control for an unmanned aerial vehicle? *i-com*, 17, 15–24. <https://doi.org/10.1515/icom-2018-0001>
- Hong, B., Lin, Z., Chen, X., Hou, J., Lv, S., & Gao, Z. (2022). Development and application of key technologies for Guide Dog Robot: A systematic literature review. *Robotics and Autonomous Systems*, 154, Article 104104. <https://doi.org/10.1016/j.robot.2022.104104>

- Hostetter, A. B., & Alibali, M. W. (2019). Gesture as simulated action: Revisiting the framework. *Psychonomic Bulletin & Review*, 26, 721–752. <https://doi.org/10.3758/s13423-018-1548-0>
- Iman, R. L., Hora, S. C., & Conover, W. J. (1984). Comparison of asymptotically distribution-free procedures for the analysis of complete blocks. *Journal of the American Statistical Association*, 79, 674–685. <https://doi.org/10.2307/2288416>
- Jacob, F., Grosse, E. H., Morana, S., & König, C. J. (2023). Picking with a robot colleague: A systematic literature review and evaluation of technology acceptance in human–robot collaborative warehouses. *Computers & Industrial Engineering*, 180, Article 109262. <https://doi.org/10.1016/j.cie.2023.109262>
- Kaczmarek, W., Lotys, B., Borys, S., Laskowski, D., & Lubkowski, P. (2021). Controlling an industrial robot using a graphic tablet in offline and online mode. *Sensors*, 21, Article 2439. <https://doi.org/10.3390/s21072439>
- Kandalafi, N., Kalidindi, P. S., Narra, S., & Saha, H. N. (2018). Robotic arm using voice and gesture recognition. *Proceedings of the 2018 IEEE 9th Annual Information Technology, Electronics and Mobile Communication Conference*, Vancouver, Canada, 1060–1064. <https://doi.org/10.1109/IEMCON.2018.8615055>
- Kim, J. T., Byrd, M., Crandell, J. L., Walker, B. N., Turk, G., & Ha, S. (2025). *Understanding expectations for a robotic guide dog for visually impaired people*. arXiv. <https://doi.org/10.48550/arXiv.2501.04594>
- Korkiakoski, M., Alaves, P., & Kostakos, P. (2024). Preference in voice commands and gesture controls with hands-free augmented reality with novel users. *IEEE Pervasive Computing*, 23, 18–26. <https://doi.org/10.1109/mprv.2024.3364541>
- Krueger, M. W. (1993). An easy entry artificial reality. In A. Wexelblat (Ed.), *Virtual reality. Applications and explorations* (pp. 147–161). Academic Press. <https://doi.org/10.1016/B978-0-12-745045-2.50017-9>
- Li, S.-A., Liu, Y.-Y., Chen, Y.-C., Feng, H.-M., Shen, P.-K., & Wu, Y.-C. (2023). Voice interaction recognition design in real-life scenario mobile robot applications. *Applied Sciences*, 13, Article 3359. <https://doi.org/10.3390/app13053359>
- Meghana, M., Kumari, C. U., Priya, J. S., Mrinal, P., Sai, K. A. V., Reddy, S. P., Vikranth, K., Kumar, T. S., & Panigrahy, A. K. (2020). Hand gesture recognition and voice controlled robot. *Materials Today: Proceedings*, 33, 4121–4123. <https://doi.org/10.1016/j.matpr.2020.06.553>
- Microsoft. (2020). TrackedHandJoint Enum. <https://learn.microsoft.com/en-us/dotnet/api/microsoft.mixedreality.toolkit.utilities.trackedhandjoint?view=mixed-reality-toolkit-unity-2020-dotnet-2.7.0>
- Microsoft. (2021). Articulated hand tracking. <https://learn.microsoft.com/en-us/windows/mixed-reality/develop/native/hands-and-motion-controllers-in-directx#hand-skeleton>
- Microsoft. (2022a). Mixed Reality Toolkit 2. <https://learn.microsoft.com/en-us/windows/mixed-reality/mrtk-unity/mrtk2/?view=mrtkunity-2022-05>
- Microsoft. (2022b). Speech — MRTK2. <https://learn.microsoft.com/en-us/windows/mixed-reality/mrtk-unity/mrtk2/features/input/speech?view=mrtkunity-2022-05>
- Microsoft. (2024a). Microsoft HoloLens 2. <https://www.microsoft.com/en-us/hololens>
- Microsoft. (2024b). HoloLens 2 gestures for authoring and navigating in Dynamics 365 Guides. <https://learn.microsoft.com/en-us/dynamics365/mixed-reality/guides/authoring-gestures-hl2>
- Mišeikis, J., Caroni, P., Duchamp, P., Gasser, A., Marko, R., Mišeikienė, N., Zwilling, F., De Castelbajac, C., Eicher, L., Früh, M., & Früh, H. (2020). Lio-a personal robot assistant for human-robot interaction and care applications. *IEEE Robotics and Automation Letters*, 5, 5339–5346. <https://doi.org/10.1109/LRA.2020.3007462>
- Moniruzzaman, M., Rassau, A., Chai, D., & Islam, S. M. S. (2022). Teleoperation methods and enhancement techniques for mobile robots: A comprehensive survey. *Robotics and Autonomous Systems*, 150, Article 103973. <https://doi.org/10.1016/j.robot.2021.103973>
- Naeem, B., Kareem, W., Saeed-Ul-Hassan, Naeem, N., & Naeem, R. (2024). Voice controlled humanoid robot. *International Journal of Intelligent Robotics and Applications*, 8, 61–75. <https://doi.org/10.1007/s41315-023-00304-z>
- Nauert, F., & Kampmann, P. (2023). Inspection and maintenance of industrial infrastructure with autonomous underwater robots. *Frontiers in Robotics and AI*, 10, Article 1240276. <https://doi.org/10.3389/frobt.2023.1240276>

- Nogales, R. E., & Benalcázar, M. E. (2021). Hand gesture recognition using machine learning and infrared information: A systematic literature review. *International Journal of Machine Learning and Cybernetics*, 12, 2859–2886. <https://doi.org/10.1007/s13042-021-01372-y>
- Norman, D. A. (2010). Natural user interfaces are not natural. *Interactions*, 17, 6–10. <https://doi.org/10.1145/1744161.1744163>
- OpenAI. (2024, December 5). Introducing ChatGPT Pro. <https://openai.com/index/introducing-chatgpt-pro>
- Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., & Duchesnay, É. (2011). Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12, 2825–2830.
- Pfeiffer, C., & Scaramuzza, D. (2021). Human-piloted drone racing: Visual processing and control. *IEEE Robotics and Automation Letters*, 6, 3467–3474. <https://doi.org/10.1109/LRA.2021.3064282>
- Pianca, F., & Santucci, V. G. (2023). Interdependence as the key for an ethical artificial autonomy. *AI & Society*, 38, 2045–2059. <https://doi.org/10.1007/s00146-021-01313-x>
- Popescu, S. T., & Wexler, M. (2012). Spontaneous body movements in spatial cognition. *Frontiers in Psychology*, 3, Article 136. <https://doi.org/10.3389/fpsyg.2012.00136>
- QualtricsSM. (2024). Qualtrics. <https://www.qualtrics.com>
- Quigley, M., Conley, K., Gerkey, B., Faust, J., Foote, T., Leibs, J., Berger, E., Wheeler, R., & Ng, A. Y. (2009). ROS: An open-source Robot Operating System. *Proceedings of the ICRA Workshop on Open Source Software*, Kobe, Japan.
- Radford, A., Kim, J. W., Xu, T., Brockman, G., McLeavey, C., & Sutskever, I. (2023). Robust speech recognition via large-scale weak supervision. *Proceedings of the 40th International Conference on Machine Learning*, Honolulu, HI, 28492–28518. <https://proceedings.mlr.press/v202/radford23a.html>
- Rico, J., & Brewster, S. (2010). Usable gestures for mobile interfaces: Evaluating social acceptability. *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, Atlanta, GA, 887–896. <https://doi.org/10.1145/1753326.1753458>
- Rippy, K., Gangopadhyay, A., & Jayarajah, K. (2024). *GestRight*: Understanding the feasibility of gesture-driven tele-operation in human-robot teams. *Proceedings of the 2024 IEEE/RSJ International Conference on Intelligent Robots and Systems*, Abu Dhabi, United Arab Emirates, 13487–13494. <https://doi.org/10.1109/IROS58592.2024.10802649>
- Ro, H., Byun, J.-H., Park, Y. J., Lee, N. K., & Han, T.-D. (2019). AR pointer: Advanced ray-casting interface using laser pointer metaphor for object manipulation in 3D augmented reality environment. *Applied Sciences*, 9, Article 3078. <https://doi.org/10.3390/app9153078>
- Roldán-Gómez, J. J., González-Gironda, E., & Barrientos, A. (2021). A survey on robotic technologies for forest firefighting: Applying drone swarms to improve firefighters' efficiency and safety. *Applied Sciences*, 11, Article 363. <https://doi.org/10.3390/app11010363>
- Sadhu, A., Peplinski, J. E., Mohammadkhorasani, A., & Moreu, F. (2023). A review of data management and visualization techniques for structural health monitoring using BIM and virtual or augmented reality. *Journal of Structural Engineering*, 149, Article 03122006. [https://doi.org/10.1061/\(ASCE\)ST.1943-541X.0003498](https://doi.org/10.1061/(ASCE)ST.1943-541X.0003498)
- Sandberg, T. H. (2023). *Gesture control of quadruped robots. A study of technological and user acceptance barriers in real world situations* (Master's thesis). University of Oslo, Norway.
- Segal, A., Tversky, B., & Black, J. (2014). Conceptually congruent actions can promote thought. *Journal of Applied Research in Memory and Cognition*, 3, 124–130. <https://doi.org/10.1016/j.jarmac.2014.06.004>
- Shepard, R. N., & Metzler, J. (1971). Mental rotation of three-dimensional objects. *Science*, 171, 701–703. <https://doi.org/10.1126/science.171.3972.701>
- Shin, S., Evetts, T., Saylor, H., Kim, H., Woo, S., Rhee, W., & Kim, S. W. (2024). *Non-verbal interaction and interface with a quadruped robot using body and hand gestures: Design and user experience evaluation*. arXiv. <https://doi.org/10.48550/arXiv.2408.17066>
- Smolyanskiy, N., & Gonzalez-Franco, M. (2017). Stereoscopic first person view system for drone navigation. *Frontiers in Robotics and AI*, 4, Article 11. <https://doi.org/10.3389/frobt.2017.00011>
- Solanes, J. E., Muñoz, A., Gracia, L., & Tornero, J. (2022). Virtual reality-based interface for advanced assisted mobile robot teleoperation. *Applied Sciences*, 12, Article 6071. <https://doi.org/10.3390/app12126071>

- Soler, P., Lukovic, P., Reynaud, L., Sgobbi, A., Bruni, F., Brun, M., Zünd, M., Bollati, R., Pollefeys, M., Blum, H., & Bauer, Z. (2024). *HoloSpot: Intuitive object manipulation via mixed reality drag-and-drop*. arXiv. <https://doi.org/10.48550/arXiv.2410.11110>
- Steinke, J., Rischke, J., Sossalla, P., Hofer, J., Vielhaus, C. L., Vom Hofe, N., & Fitzek, H. P. F. (2023). The future of dog walking—four-legged robots and augmented reality. *Proceedings of the 2023 IEEE 24th International Symposium on a World of Wireless, Mobile and Multimedia Networks*, Boston, MA, 352–354. <https://doi.org/10.1109/WoWMoM57956.2023.00060>
- Sun, K., Yu, C., Shi, W., Liu, L., & Shi, Y. (2018). Lip-Interact: Improving mobile device interaction with silent speech commands. *Proceedings of the 31st Annual ACM Symposium on User Interface Software and Technology*, Berlin, Germany, 581–593. <https://doi.org/10.1145/3242587.3242599>
- Tabone, W., & De Winter, J. C. F. (2023). Using ChatGPT for human-computer interaction research: A primer. *Royal Society Open Science*, 10, Article 231053. <https://doi.org/10.1098/rsos.231053>
- Tezza, D., Laesker, D., & Andujar, M. (2021). The learning experience of becoming a FPV drone pilot. *Companion of the 2021 ACM/IEEE International Conference on Human-Robot Interaction*, Boulder, CO, 239–241. <https://doi.org/10.1145/3434074.3447167>
- TP-Link. (2019). Archer AX50 | AX3000 Dual-Band Wi-Fi 6 Router. <https://www.tp-link.com/en/home-networking/wifi-router/archer-ax50>
- Unity. (2021). Unity 2021.3.28. <https://unity.com/releases/editor/whats-new/2021.3.28>
- Unity. (2022). ROS-TCP-Connector. <https://github.com/Unity-Technologies/ROS-TCP-Connector>
- Vanamala, H. R., Akash, S. M., Vinay, A., Kumar, S., & Rathod, M. (2022). Gesture and voice controlled robot for industrial applications. *Proceedings of the 2022 International Conference for Advancement in Technology*, Goa, India. <https://doi.org/10.1109/ICONAT53423.2022.9725903>
- Vom Hofe, N., Sossalla, P., Hofer, J., Vielhaus, C. L., Rischke, J., Steinke, J., & Fitzek, F. H. (2023). Robotics meets augmented reality: Real-time mapping with Boston Dynamics Spot and Microsoft HoloLens 2. *Proceedings of the 2023 IEEE 24th International Symposium on a World of Wireless, Mobile and Multimedia Networks*, Boston, MA, 358–360. <https://doi.org/10.1109/WoWMoM57956.2023.00062>
- Wan, Y., Sun, J., Peers, C., Humphreys, J., Kanoulas, D., & Zhou, C. (2023). Performance and usability evaluation scheme for mobile manipulator teleoperation. *IEEE Transactions on Human-Machine Systems*, 53, 844–854. <https://doi.org/10.1109/THMS.2023.3289628>
- Wickens, C. D. (2002). Multiple resources and performance prediction. *Theoretical Issues in Ergonomics Science*, 3(2), 159–177. <https://doi.org/10.1080/14639220210123806>
- Wickens, C. D. (2008). Multiple resources and mental workload. *Human Factors*, 50(3), 449–455. <https://doi.org/10.1518/001872008X288394>
- Wickens, C. D., & Prevett, T. T. (1995). Exploring the dimensions of egocentricity in aircraft navigation displays. *Journal of Experimental Psychology: Applied*, 1, 110–135. <https://doi.org/10.1037/1076-898X.1.2.110>
- Zhang, G., Zhang, D., Duan, L., & Han, G. (2023). *Accessible robot control in mixed reality*. arXiv. <https://doi.org/10.48550/arXiv.2306.02393>
- Zhou, H., Wang, D., Yu, Y., & Zhang, Z. (2023). Research progress of human–computer interaction technology based on gesture recognition. *Electronics*, 12, Article 2805. <https://doi.org/10.3390/electronics12132805>

Appendices

A. Training the gesture classifier

To train the gesture classifier, a researcher wearing the HoloLens 2 demonstrated each of the four right-hand gestures. He kept his head stationary and held the gestures in front of him. To add variety in the dataset, the hand moved horizontally at different heights, with fingers continuously bending and extending within a specific range. The hand movement limits were approximately 25 cm left and right from the middle camera of the HoloLens, with the upper limit at the same height as the HoloLens and the lower limit extending down to about 40 cm. The hand reached as far as 45 cm away from and as close as 20 cm to the camera. These limits were inside the gesture frame of the HoloLens.

Although each gesture was demonstrated for 60 s at 60 Hz (expected 14,400 frames), occasional frame loss occurred, resulting in 14,021 samples in total for the four gestures. The gesture data, including demos of three gesture commands (*Walk Forward*, *Rotate Left*, *Rotate Right*) and one gesture (fist) for the neutral hand pose, made up the dataset for training and testing the gesture classifier. Each sample consisted of a set of joints on the extracted hand skeleton.

The collected dataset was preprocessed using `MinMaxScaler` and `StandardScaler` from the Python `scikit-learn` package (Pedregosa et al., 2011) to normalize the data. The data were then split into a training set and a test set in a 70:30 ratio. The training set was fed into an SVM with a linear kernel, chosen for its effectiveness with a small number of classes (in our case, four gestures). With our dataset and preprocessing, the SVM-based gesture classifier achieved an overall accuracy of 99% on the test set.

The trained gesture classifier could run at a maximum of 60 Hz on our PC with the offline hand skeleton data. However, during the experiment, due to computational and communication overhead between devices, its frequency dropped to approximately 30 Hz.

B. Detecting command mistakes

The logged data for each trial contained the commands participants gave to the robot, either through gestures or speech. Each entry included a command and its corresponding timestamp. The minimum 23 commands required to complete the task resulted in 23 unique corresponding robot poses in $(x, y, orientation)$. By using these 23 correct robot poses and their corresponding commands at each checkpoint, incorrect commands and extra commands given by participants to correct Spot's position after a mistake were detected. These were then analyzed to identify specific types of human mistakes.

For each trial, the initial robot pose was set to $(0, 0, 0)$, and the robot's poses after each command were computed sequentially. Then, the robot poses were compared with the reference poses one by one to determine if they were correct. If they did not match, the first incorrect command was marked as a human mistake, while the subsequent commands needed to return the robot to the correct position were not counted as additional mistakes. The comparison resumed at the next matching robot pose between the participant's command sequence and the reference sequence, skipping any intermediate corrective poses.

When a mistake happened, sometimes the participant did not reposition the robot to the pose before the mistake (e.g., steps 7–10 in Table 3.B, right). In these cases, correct robot poses in the reference command list were skipped in order to continue the comparison and mistake detection.

The detected mistakes were classified into one of three classes: (1) duplicated command mistakes, (2) left/right command mistakes, and (3) other command mistakes. A mistake was marked as a duplicate if the incorrect command was the same as the immediately preceding correct command (i.e., issuing the same command twice unnecessarily). Left/right mistakes occurred when the participant issued a rotation command opposite to the rotation required. The rest were marked as other mistakes.

Table 3.B. The reference command list (left table) and the command list of Participant 10 in the VW condition (right table). The underlined row in the right table indicates a duplicate command mistake (*Walk Forward* followed by *Walk Forward*). The boldfaced row in the right table indicates Spot's first aligned pose achieved through different commands than those in the correct command list (left table), due to the participant not correcting the robot's position after making a previous command mistake.

Step	Command	Pose (x, y, orientation)
1	Walk Forward	[1, 0, 0]
2	Rotate Right	[1, 0, 3]
3	Walk Forward	[1, -1, 3]
4	Rotate Left	[1, -1, 0]
5	Walk Forward	[2, -1, 0]
6	Walk Forward	[3, -1, 0]
7	Rotate Left	[3, -1, 1]
8	Walk Forward	[3, 0, 1]
9	Rotate Left	[3, 0, 2]
10	Walk Forward	[2, 0, 2]
11	Rotate Right	[2, 0, 1]
12	Walk Forward	[2, 1, 1]
13	Rotate Right	[2, 1, 0]
14	Walk Forward	[3, 1, 0]
15	Rotate Left	[3, 1, 1]
16	Walk Forward	[3, 2, 1]
17	Rotate Left	[3, 2, 2]
18	Walk Forward	[2, 2, 2]
19	Walk Forward	[1, 2, 2]
20	Rotate Left	[1, 2, 3]
21	Walk Forward	[1, 1, 3]
22	Rotate Right	[1, 1, 2]
23	Walk Forward	[0, 1, 2]

Step	Command	Pose (x, y, orientation)	Mistake
1	Walk Forward	[1, 0, 0]	no
2	Rotate Right	[1, 0, 3]	no
3	Walk Forward	[1, -1, 3]	no
4	Rotate Left	[1, -1, 0]	no
5	Walk Forward	[2, -1, 0]	no
6	Walk Forward	[3, -1, 0]	no
<u>7</u>	<u>Walk Forward</u>	<u>[4, -1, 0]</u>	<u>yes</u>
8	Rotate Left	[4, -1, 1]	no
9	Walk Forward	[4, 0, 1]	no
10	Rotate Left	[4, 0, 2]	no
11	Walk Forward	[3, 0, 2]	no
12	Walk Forward	[2, 0, 2]	no
13	Rotate Right	[2, 0, 1]	no
14	Walk Forward	[2, 1, 1]	no
15	Rotate Right	[2, 1, 0]	no
16	Walk Forward	[3, 1, 0]	no
17	Rotate Left	[3, 1, 1]	no
18	Walk Forward	[3, 2, 1]	no
19	Rotate Left	[3, 2, 2]	no
20	Walk Forward	[2, 2, 2]	no
21	Walk Forward	[1, 2, 2]	no
22	Rotate Left	[1, 2, 3]	no
23	Walk Forward	[1, 1, 3]	no
24	Rotate Right	[1, 1, 2]	no
25	Walk Forward	[0, 1, 2]	no

C. Full statistical results

This appendix provides the means and *SD* of all dependent measures. Additionally, the correlation matrix is presented for the scores of the four conditions, along with the results of paired-samples *t*-tests.

Additionally, the data were transformed into ranks using the RT-1 procedure as described by Conover and Iman (1981). Specifically, the 218×4 matrix of scores was transformed into a rank-order variable ranging from 1 to 872 (or a smaller number in case of missing data). In cases of tied scores, the mean of the ranks was used.

Below, the corresponding means and *SD* of the ranks, as well as the results of paired-samples *t*-tests on the ranks, are provided. According to Monte Carlo simulations by Iman et al. (1984), this rank-based *t*-test is a statistically powerful test in the case of skewed distributions. It can be observed that the *p*-values and Cohen's *d* effect sizes are roughly comparable to those of the regular paired-samples *t*-test. This supports the justification for using the paired-samples *t*-test in our article.

MEDIAN INTER-COMMAND TIME (s)

	VW	VS	GW	GS
Sample size:	217	218	218	218
Mean:	5.50	5.50	5.96	5.95
Standard deviation:	0.64	0.62	0.75	0.54
Mean rank:	325.60	319.79	539.19	558.90
Standard deviation of rank:	225.84	215.71	234.27	223.48

Pearson product-moment correlation matrix

	VW	VS	GW	GS
VW	1.00	0.79	0.42	0.29
VS	0.79	1.00	0.50	0.45
GW	0.42	0.50	1.00	0.59
GS	0.29	0.45	0.59	1.00

Spearman rank-order correlation matrix

	VW	VS	GW	GS
VW	1.00	0.79	0.39	0.37
VS	0.79	1.00	0.47	0.50
GW	0.39	0.47	1.00	0.72
GS	0.37	0.50	0.72	1.00

Normal: VW&S vs. GW&S, $t(216) = -11.38$, $p = 8.03e-24$ ($d = -0.78$).

Rank: VW&S vs. GW&S, $t(216) = -15.11$, $p = 1.08e-35$ ($d = -1.09$).

Normal: VW vs. VS, $t(216) = -0.10$, $p = 9.24e-01$ ($d = -0.01$).

Rank: VW vs. VS, $t(216) = 0.54$, $p = 5.89e-01$ ($d = 0.03$).

Normal: GW vs. GS, $t(217) = 0.31$, $p = 7.60e-01$ ($d = 0.02$).

Rank: GW vs. GS, $t(217) = -1.70$, $p = 9.00e-02$ ($d = -0.09$).

NUMBER OF COMMANDS

Chapter 3

	VW	VS	GW	GS
Sample size:	217	218	218	218
Mean:	24.21	24.30	23.71	23.78
Standard deviation:	2.22	2.61	1.90	1.76
Mean rank:	459.98	454.79	402.53	426.81
Standard deviation of rank:	205.50	205.25	168.79	180.19

Pearson product-moment correlation matrix

	VW	VS	GW	GS
VW	1.00	0.11	-0.11	0.01
VS	0.11	1.00	-0.07	-0.04
GW	-0.11	-0.07	1.00	0.18
GS	0.01	-0.04	0.18	1.00

Spearman rank-order correlation matrix

	VW	VS	GW	GS
VW	1.00	0.10	-0.11	0.06
VS	0.10	1.00	-0.02	0.04
GW	-0.11	-0.02	1.00	0.10
GS	0.06	0.04	0.10	1.00

Normal: VW&S vs. GW&S, $t(216) = 3.15$, $p = 1.88e-03$ ($d = 0.32$).

Rank: VW&S vs. GW&S, $t(216) = 3.12$, $p = 2.08e-03$ ($d = 0.30$).

Normal: VW vs. VS, $t(216) = -0.44$, $p = 6.60e-01$ ($d = -0.04$).

Rank: VW vs. VS, $t(216) = 0.25$, $p = 8.06e-01$ ($d = 0.03$).

Normal: GW vs. GS, $t(217) = -0.43$, $p = 6.65e-01$ ($d = -0.04$).

Rank: GW vs. GS, $t(217) = -1.54$, $p = 1.26e-01$ ($d = -0.14$).

TOTAL DISTANCE WALKED (m)

	VW	VS	GW	GS
Sample size:	215	217	217	216
Mean:	18.4	1.4	17.7	1.4
Standard deviation:	7.0	0.9	10.1	0.6
Mean rank:	662.0	211.7	637.1	222.4
Standard deviation of rank	124.2	132.7	124.6	117.2

Pearson product-moment correlation matrix

	VW	VS	GW	GS
VW	1.00	0.09	0.62	-0.03
VS	0.09	1.00	0.02	0.37
GW	0.62	0.02	1.00	-0.05
GS	-0.03	0.37	-0.05	1.00

Spearman rank-order correlation matrix

	VW	VS	GW	GS
VW	1.00	0.09	0.64	-0.02
VS	0.09	1.00	0.07	0.46
GW	0.64	0.07	1.00	0.00
GS	-0.02	0.46	0.00	1.00

Normal: VW vs. GW , $t(214) = 1.40$, $p = 1.63e-01$ ($d = 0.07$).

Rank: VW vs. GW , $t(214) = 3.66$, $p = 3.22e-04$ ($d = 0.20$).

Chapter 3

COMMAND DETECTION (1-5)

	VW	VS	GW	GS
Sample size:	218	218	218	218
Mean:	4.62	4.58	4.15	4.29
Standard deviation:	0.61	0.67	0.77	0.68
Mean rank:	507.07	498.67	353.19	387.07
Standard deviation of rank:	203.04	212.31	226.13	218.60

Pearson product-moment correlation matrix

	VW	VS	GW	GS
VW	1.00	0.42	0.16	0.07
VS	0.42	1.00	0.26	0.14
GW	0.16	0.26	1.00	0.29
GS	0.07	0.14	0.29	1.00

Spearman rank-order correlation matrix

	VW	VS	GW	GS
VW	1.00	0.43	0.19	0.11
VS	0.43	1.00	0.24	0.11
GW	0.19	0.24	1.00	0.32
GS	0.11	0.11	0.32	1.00

Normal: VW&S vs. GW&S, $t(217) = 8.09$, $p = 4.33e-14$ ($d = 0.68$).

Rank: VW&S vs. GW&S, $t(217) = 8.89$, $p = 2.41e-16$ ($d = 0.74$).

Normal: VW vs. VS, $t(217) = 0.78$, $p = 4.34e-01$ ($d = 0.06$).

Rank: VW vs. VS, $t(217) = 0.56$, $p = 5.74e-01$ ($d = 0.04$).

Normal: GW vs. GS, $t(217) = -2.34$, $p = 2.03e-02$ ($d = -0.19$).

Rank: GW vs. GS, $t(217) = -1.94$, $p = 5.33e-02$ ($d = -0.15$).

INTUITIVENESS (1-5)

	VW	VS	GW	GS
Sample size:	218	218	218	218
Mean:	4.56	4.25	4.37	3.96
Standard deviation:	0.63	0.74	0.70	0.80
Mean rank:	525.04	422.70	460.15	338.11
Standard deviation of rank:	208.21	224.93	224.37	223.27

Pearson product-moment correlation matrix

	VW	VS	GW	GS
VW	1.00	0.38	0.19	0.19
VS	0.38	1.00	0.08	0.40
GW	0.19	0.08	1.00	0.40
GS	0.19	0.40	0.40	1.00

Spearman rank-order correlation matrix

	VW	VS	GW	GS
VW	1.00	0.45	0.23	0.21
VS	0.45	1.00	0.14	0.39
GW	0.23	0.14	1.00	0.41
GS	0.21	0.39	0.41	1.00

Normal: VW&S vs. GW&S, $t(217) = 5.02$, $p = 1.05e-06$ ($d = 0.40$).

Rank: VW&S vs. GW&S, $t(217) = 5.17$, $p = 5.22e-07$ ($d = 0.40$).

Normal: VW vs. VS, $t(217) = 6.03$, $p = 7.16e-09$ ($d = 0.45$).

Rank: VW vs. VS, $t(217) = 6.58$, $p = 3.57e-10$ ($d = 0.47$).

Normal: GW vs. GS, $t(217) = 7.25$, $p = 7.05e-12$ ($d = 0.54$).

Rank: GW vs. GS, $t(217) = 7.41$, $p = 2.71e-12$ ($d = 0.55$).

Chapter 3

ALIGNMENT PERCENTAGE

	VW	VS	GW	GS
Sample size:	215	217	217	216
Mean:	41.9	29.1	46.8	29.6
Standard deviation:	8.4	3.3	8.4	3.6
Mean rank:	575.8	243.5	667.4	245.8
Standard deviation of rank:	187.9	130.3	166.8	153.1

Pearson product-moment correlation matrix

	VW	VS	GW	GS
VW	1.00	-0.07	0.37	-0.10
VS	-0.07	1.00	-0.02	-0.03
GW	0.37	-0.02	1.00	0.03
GS	-0.10	-0.03	0.03	1.00

Spearman rank-order correlation matrix

	VW	VS	GW	GS
VW	1.00	-0.05	0.35	-0.13
VS	-0.05	1.00	-0.02	0.00
GW	0.35	-0.02	1.00	0.02
GS	-0.13	0.00	0.02	1.00

Normal: VW vs. GW , $t(214) = -7.38$, $p = 3.36e-12$ ($d = -0.58$).

Rank: VW vs. GW , $t(214) = -6.47$, $p = 6.40e-10$ ($d = -0.52$).

SICKNESS (0-10)

	VW	VS	GW	GS
Sample size:	218	218	218	218
Mean:	0.28	0.32	0.30	0.28
Standard deviation:	0.73	0.82	0.77	0.60
Mean rank:	432.64	434.83	436.92	441.61
Standard deviation of rank:	169.35	173.95	173.15	174.88

Pearson product-moment correlation matrix

	VW	VS	GW	GS
VW	1.00	0.56	0.58	0.53
VS	0.56	1.00	0.67	0.66
GW	0.58	0.67	1.00	0.73
GS	0.53	0.66	0.73	1.00

Spearman rank-order correlation matrix

	VW	VS	GW	GS
VW	1.00	0.59	0.61	0.53
VS	0.59	1.00	0.59	0.68
GW	0.61	0.59	1.00	0.74
GS	0.53	0.68	0.74	1.00

Normal: VW&S vs. GW&S, $t(217) = 0.36$, $p = 7.21e-01$ ($d = 0.02$).

Rank: VW&S vs. GW&S, $t(217) = -0.70$, $p = 4.84e-01$ ($d = -0.04$).

Normal: VW vs. VS, $t(217) = -0.65$, $p = 5.19e-01$ ($d = -0.04$).

Rank: VW vs. VS, $t(217) = -0.21$, $p = 8.36e-01$ ($d = -0.01$).

Normal: GW vs. GS, $t(217) = 0.77$, $p = 4.40e-01$ ($d = 0.04$).

Rank: GW vs. GS, $t(217) = -0.55$, $p = 5.80e-01$ ($d = -0.03$).

Chapter 3

NUMBER OF DUPLICATE COMMAND MISTAKES

	VW	VS	GW	GS
Sample size:	215	217	217	216
Mean:	0.214	0.180	0.152	0.032
Standard deviation:	0.464	0.408	0.397	0.177
Mean rank:	458.360	448.922	435.396	389.354
Standard deviation of rank:	171.295	162.406	150.025	75.877

Pearson product-moment correlation matrix

	VW	VS	GW	GS
VW	1.00	-0.00	-0.15	-0.08
VS	-0.00	1.00	0.00	-0.01
GW	-0.15	0.00	1.00	0.20
GS	-0.08	-0.01	0.20	1.00

Spearman rank-order correlation matrix

	VW	VS	GW	GS
VW	1.00	-0.02	-0.16	-0.09
VS	-0.02	1.00	-0.00	-0.01
GW	-0.16	-0.00	1.00	0.23
GS	-0.09	-0.01	0.23	1.00

Normal: VW&S vs. GW&S, $t(214) = 3.75$, $p = 2.30e-04$ ($d = 0.39$).

Rank: VW&S vs. GW&S, $t(214) = 3.85$, $p = 1.58e-04$ ($d = 0.40$).

Normal: VW vs. VS, $t(214) = 0.88$, $p = 3.78e-01$ ($d = 0.08$).

Rank: VW vs. VS, $t(214) = 0.66$, $p = 5.09e-01$ ($d = 0.06$).

Normal: GW vs. GS, $t(215) = 4.27$, $p = 2.92e-05$ ($d = 0.39$).

Rank: GW vs. GS, $t(215) = 4.34$, $p = 2.20e-05$ ($d = 0.39$).

NUMBER OF LEFT/RIGHT COMMAND MISTAKES

	VW	VS	GW	GS
Sample size:	215	217	217	216
Mean:	0.116	0.143	0.032	0.213
Standard deviation:	0.321	0.444	0.202	0.474
Mean rank:	434.651	435.111	397.023	465.380
Standard deviation of rank:	137.198	139.330	71.523	169.152

Pearson product-moment correlation matrix

	VW	VS	GW	GS
VW	1.00	0.24	0.01	0.08
VS	0.24	1.00	-0.05	0.01
GW	0.01	-0.05	1.00	0.07
GS	0.08	0.01	0.07	1.00

Spearman rank-order correlation matrix

	VW	VS	GW	GS
VW	1.00	0.23	0.03	0.09
VS	0.23	1.00	-0.06	0.04
GW	0.03	-0.06	1.00	0.06
GS	0.09	0.04	0.06	1.00

Normal: VW&S vs. GW&S, $t(214) = 0.26$, $p = 7.97e-01$ ($d = 0.03$).

Rank: VW&S vs. GW&S, $t(214) = 0.39$, $p = 6.99e-01$ ($d = 0.04$).

Normal: VW vs. VS, $t(214) = -0.85$, $p = 3.97e-01$ ($d = -0.07$).

Rank: VW vs. VS, $t(214) = -0.08$, $p = 9.37e-01$ ($d = -0.00$).

Normal: GW vs. GS, $t(215) = -5.30$, $p = 2.91e-07$ ($d = -0.50$).

Rank: GW vs. GS, $t(215) = -5.60$, $p = 6.48e-08$ ($d = -0.53$).

Chapter 3

NUMBER OF OTHER COMMAND MISTAKES

	VW	VS	GW	GS
Sample size:	215	217	217	216
Mean:	0.037	0.046	0.014	0.051
Standard deviation:	0.190	0.210	0.117	0.241
Mean rank:	433.574	437.408	423.472	437.572
Standard deviation of rank:	81.957	90.785	50.559	91.317

Pearson product-moment correlation matrix

	VW	VS	GW	GS
VW	1.00	0.07	-0.02	-0.04
VS	0.07	1.00	-0.03	-0.05
GW	-0.02	-0.03	1.00	0.14
GS	-0.04	-0.05	0.14	1.00

Spearman rank-order correlation matrix

	VW	VS	GW	GS
VW	1.00	0.07	-0.02	-0.04
VS	0.07	1.00	-0.03	-0.05
GW	-0.02	-0.03	1.00	0.16
GS	-0.04	-0.05	0.16	1.00

Normal: VW&S vs. GW&S, $t(214) = 0.65$, $p = 5.18e-01$ ($d = 0.07$).

Rank: VW&S vs. GW&S, $t(214) = 0.84$, $p = 4.03e-01$ ($d = 0.08$).

Normal: VW vs. VS, $t(214) = -0.50$, $p = 6.18e-01$ ($d = -0.04$).

Rank: VW vs. VS, $t(214) = -0.50$, $p = 6.18e-01$ ($d = -0.04$).

Normal: GW vs. GS, $t(215) = -2.16$, $p = 3.22e-02$ ($d = -0.20$).

Rank: GW vs. GS, $t(215) = -2.13$, $p = 3.42e-02$ ($d = -0.19$).

D. Boxplots of selected dependent measures

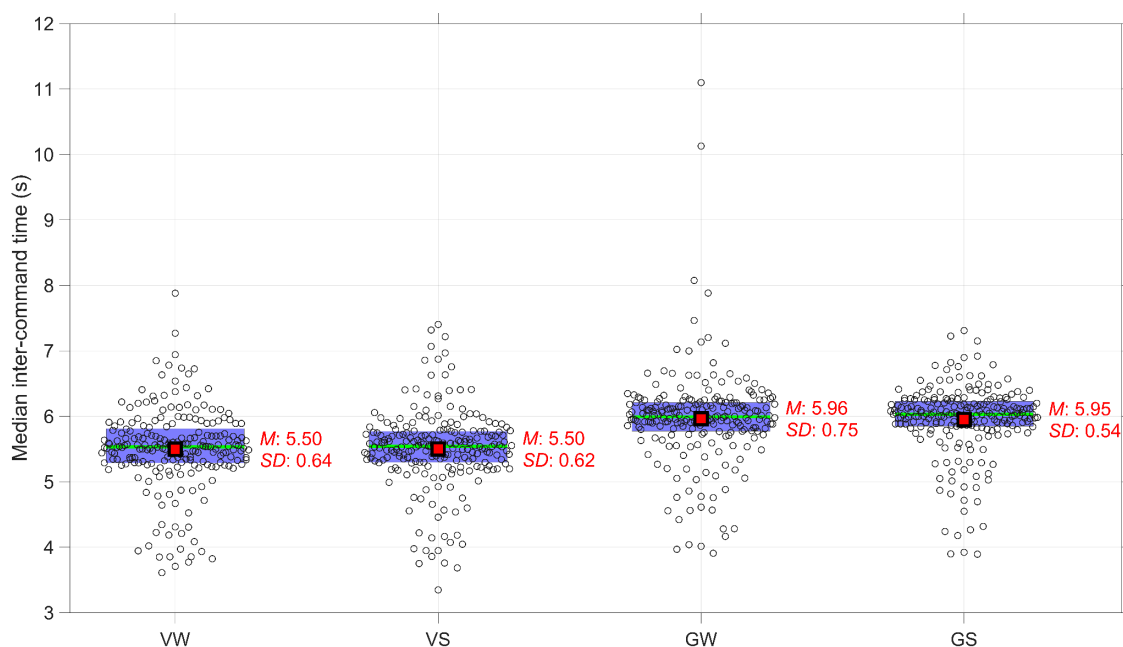


Figure 3.D1. Boxplot of the median inter-command time. The red square indicates the mean, the green horizontal line represents the median, and the blue shape spans from the 25th to the 75th percentile. Each black marker denotes an individual participant. VW: Voice control–Walking, VS: Voice control–Standing, GW: Gesture control–Walking, GS: Gesture control–Standing.

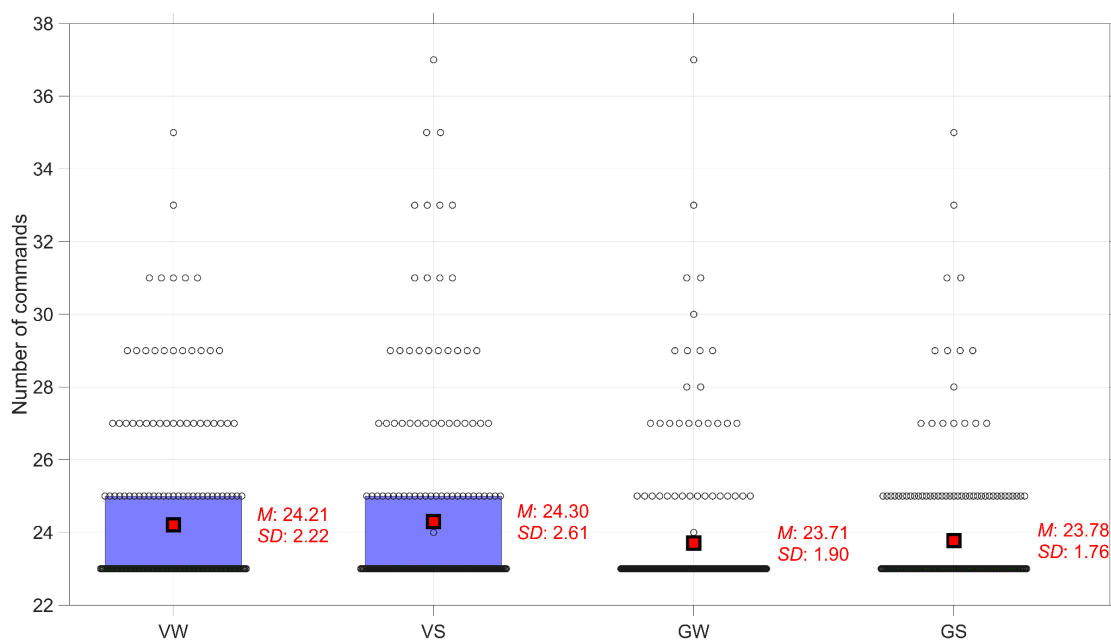


Figure 3.D2. Boxplot of the number of commands. The red square indicates the mean, the green horizontal line represents the median, and the blue shape spans from the 25th to the 75th percentile. Each black marker denotes an individual participant. VW: Voice control–Walking, VS: Voice control–Standing, GW: Gesture control–Walking, GS: Gesture control–Standing.

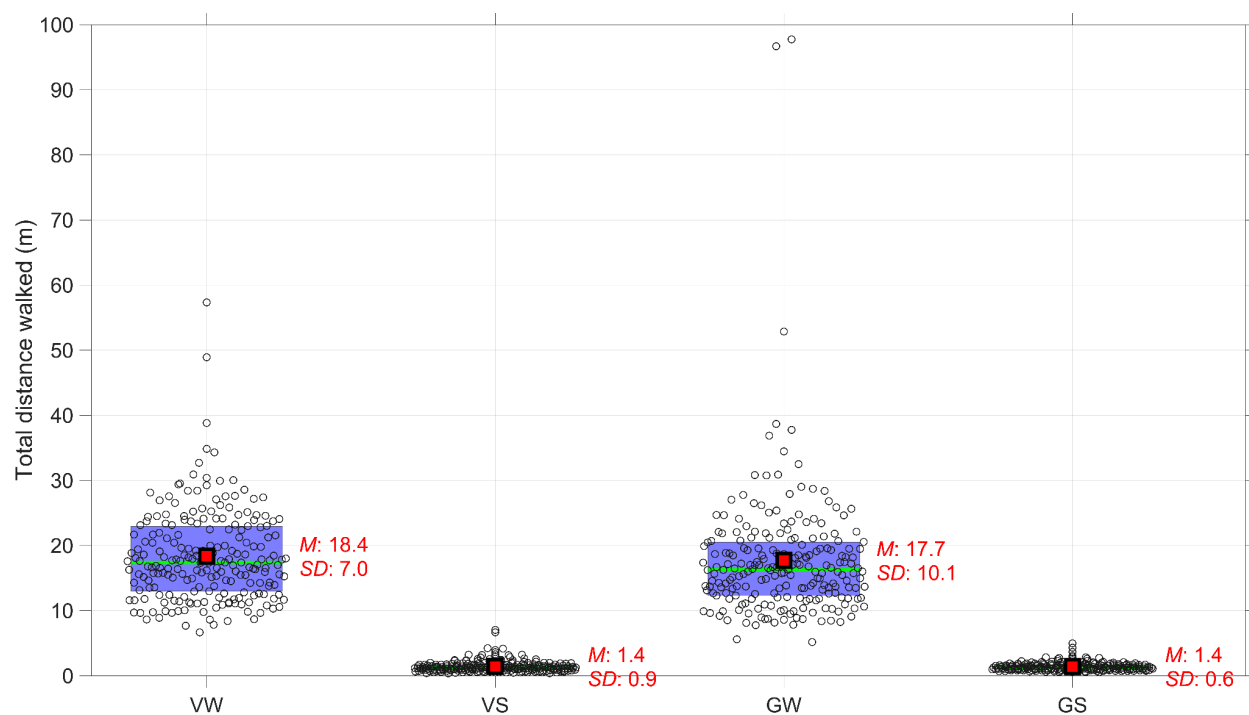


Figure 3.D3. Boxplot of the total distance walked. The red square indicates the mean, the green horizontal line represents the median, and the blue shape spans from the 25th to the 75th percentile. Each black marker denotes an individual participant. VW: Voice control–Walking, VS: Voice control–Standing, GW: Gesture control–Walking, GS: Gesture control–Standing.

E. Timeline of voice and gesture interface

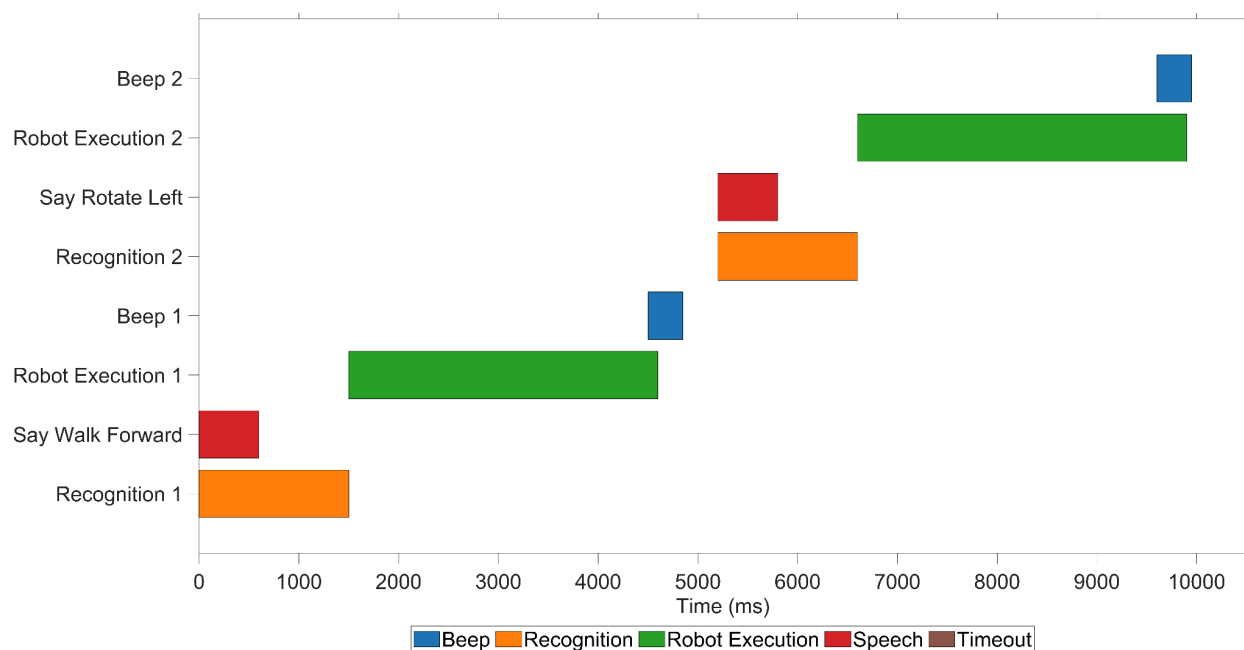


Figure 3.E1. Timeline analysis of a participant with typical performance in the voice conditions. The participant uttered a voice command (taking approximately 600 ms), which was recognized in approximately 900 ms, waited for the robot to finish the command execution, heard the beep, and then gave the next command.

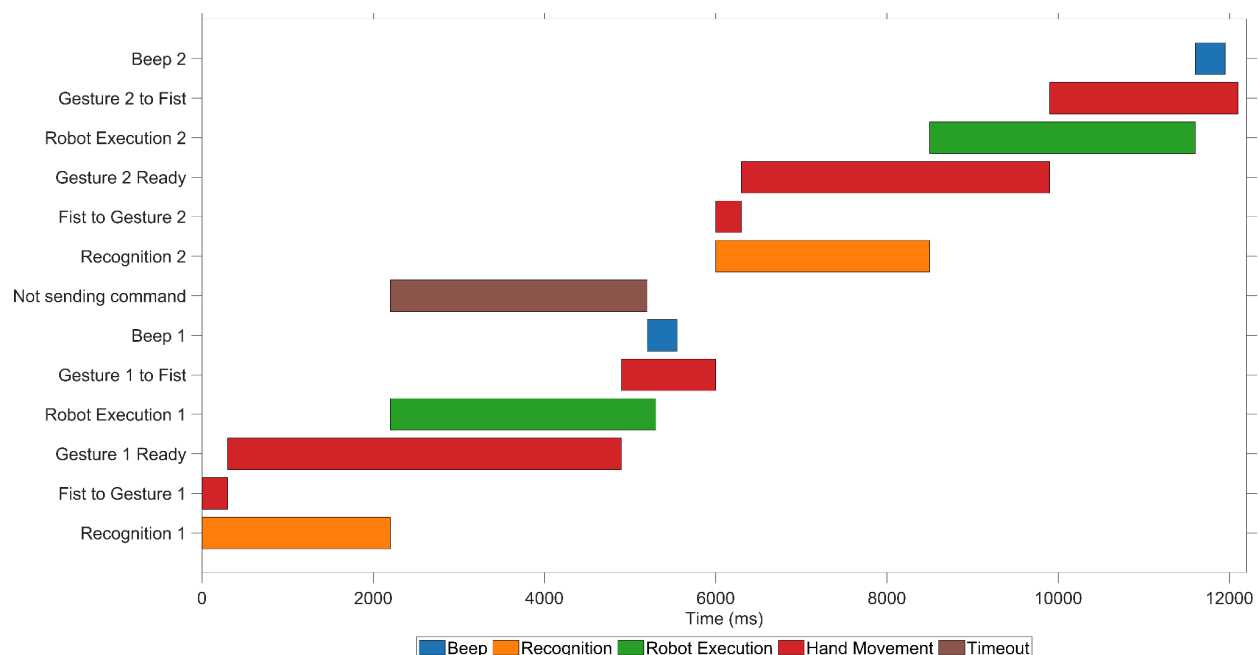


Figure 3.E2. Timeline analysis of a participant with typical performance in the gesture conditions. The participant first changed the hand from a fist to a gesture. After a valid gesture command was recognized (in around 2200 ms), the robot executed this command, and a 3-s timeout was activated, preventing the transmission of additional recognized gestures during this period. The participant held the gesture until the robot completed its execution, transitioned to the fist (neutral) gesture, and changed to the next gesture after the beep. It can be noticed that the participant held the gesture longer than necessary (i.e., while the robot was already moving, and in the first case, almost until the robot had completed its movement toward the next checkpoint).

F. Mistakes and command times per checkpoint

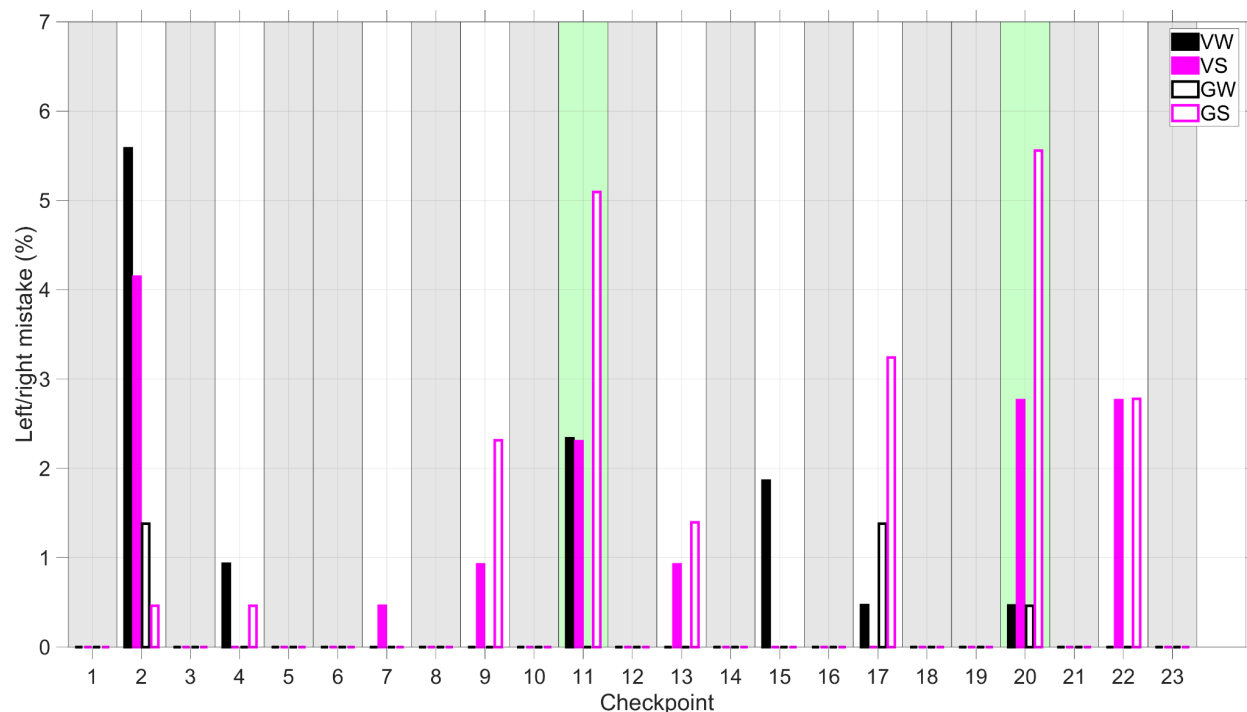


Figure 3.F1. Percentage of participants who made a left/right mistake, for each of the 23 checkpoints (see Appendix B). Light gray backgrounds represent checkpoints where the correct command was *Walk Forward*. Light green backgrounds (Checkpoints 11 and 20) indicate checkpoints where the correct command was *Rotate Left* or *Rotate Right*, and Spot was rotated 180° with respect to its initial orientation. It can be observed that the GS condition resulted in a large number of mistakes when Spot was facing the participants during Checkpoints 11 and 20. VW and VS conditions led to a large number of left/right mistakes at Checkpoint 2, possibly due to a misunderstanding of the task and reliance on the command panel (Figure 3.5).

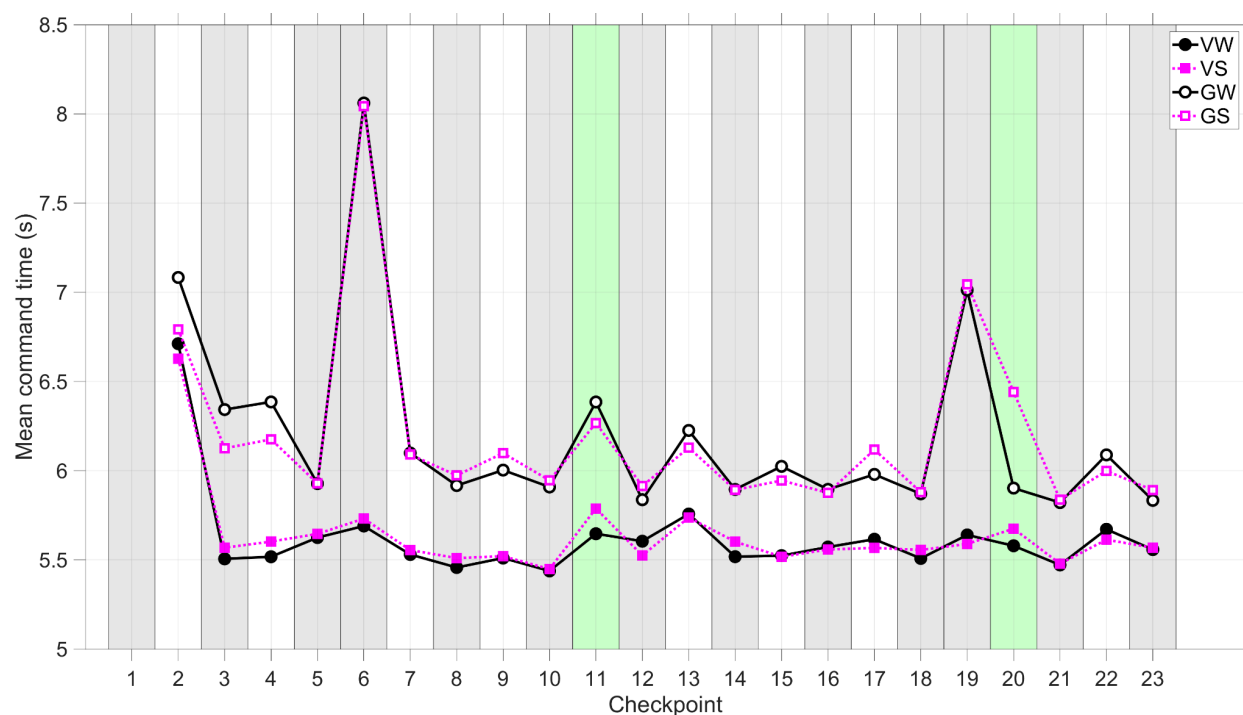


Figure 3.F2. Trimmed mean of the time since the previous command, for each of the 23 checkpoints. For each data point, only correct commands were considered, and the fastest 5% and slowest 5% of values have been removed to provide a more robust estimate. Light gray backgrounds represent checkpoints where the correct command was *Walk Forward*. Light green backgrounds (Checkpoints 11 and 20) indicate checkpoints where the correct command was *Rotate Left* or *Rotate Right*, and Spot was rotated 180° with respect to its initial orientation. It can be observed that participants in the GW and GS conditions experienced difficulty with double *Walk Forward* commands.

4

An open-source reproducible chess robot for human-robot interaction research

Intermezzo

Misalignment has now been located in the agents' perceptual mechanisms (Chapter 2) and in the interface that couples operator and robot (Chapter 3): the choice of command modality determined where cognitive conflicts arose. Studying alignment across several processing stages at once requires a controlled yet sufficiently rich platform. Chapter 4 will introduce the OpenChessRobot, combining visual perception, physical manipulation, and LLM-generated spoken commentary in one reproducible testbed (Research question 3).

This chapter was published as:

Zhang, R., De Winter, J., Dodou, D., Seyffert, H., & Eisma, Y. B. (2025). An open-source reproducible chess robot for human-robot interaction research. *Frontiers in Robotics and AI*, 12, Article 1436674. <https://doi.org/10.3389/frobt.2025.1436674>

Abstract

Recent advancements in AI have accelerated the evolution of versatile robot designs. Chess provides a standardized environment for evaluating the impact of robot behavior on human behavior. This article presents an open-source chess robot for human-robot interaction research, specifically focusing on verbal and non-verbal interactions. The OpenChessRobot recognizes chess pieces using computer vision, executes moves, and interacts with the human player through voice and robotic gestures. We detail the software design, provide quantitative evaluations of the efficacy of the robot, and offer a guide for its reproducibility. An online survey examining people's views of the robot in three possible scenarios was conducted with 597 participants. The robot received the highest ratings in the robotics education and the chess coach scenarios, while the home entertainment scenario received the lowest scores. The code is accessible on GitHub: <https://github.com/renchizhyyy/OpenChessRobot>

4.1 Introduction

Robots are becoming increasingly common across a variety of traditionally human-controlled domains. Examples range from automated mowers that maintain community lawns to robots on assembly lines and in agricultural settings. Recent advancements in AI have created new opportunities for intelligent sensing, reasoning, and acting by robots. In particular, the rapid development of large language models, such as ChatGPT and vision-language models, has lowered the barrier to human-robot communication by transforming text and images into interpretable actions or vice versa.

As technology advances, robots will likely attain greater capabilities and be able to tackle tasks previously within the exclusive realm of human expertise. This ongoing evolution may also lead to closer and more productive interactions between humans and robots. At the same time, integrating different AI-based robotic components remains a challenge, and the human-robot interaction (HRI) field lags in terms of endorsing reproducibility principles (Gunes et al., 2022). Encouraging transparent and reproducible research, therefore, remains an ongoing task.

The use of chess as a testbed for evaluating the effect of technology on human perception and behavior dates back to the 18th-century chess automaton Mechanical Turk (Standage, 2002). Furthermore, chess has played an important role in advancing the field of AI, from Claude Shannon's chess-playing algorithm (Shannon, 1950) to the success of IBM's Deep Blue (Campbell et al., 2002) and DeepMind's self-play learning algorithm (Silver et al., 2018). In this paper, we incorporate AI algorithms into the design of a chess-playing robot to be used for studying HRI. HRI research may benefit from a chess-based setup because the game of chess provides a controlled, rule-based environment in which the impact of robots on human players can be precisely measured.

HRI-oriented studies with chess robots have typically used them to imitate human behaviors and assess the resulting impact on the human opponent. Pereira et al. (2008) and Leite et al. (2013) used the social robot iCat to play chess with children. This robot relies on an electronic

chessboard as input and emits emotional responses and verbal utterances, guided by an emotion model. Sajó et al. (2011) developed Turk-2, a multimodal chess robot with human-like communication skills, while LC et al. (2021) explored human-robot and robot-robot-human interaction using artistic intervention, where expressive robot arms played chess and embodied distinct personalities.

When an electronic chessboard is not used, a camera is needed to determine if a move has been made and, if so, which move it was. A common solution is a monocular top-view camera, which has the advantage of not experiencing perspective-induced occlusion, making it relatively easy to identify any changes in the chess position (Golz & Biesenbach, 2015; Kołosowski et al., 2020; Larregay et al., 2018; Luqman & Zaffar, 2016; Mac et al., 2023; Srivatsan et al., 2020; Thanh Tra et al., 2018). An exception to this concept is the Gambit chess robot by Matuszek et al. (2011), which does not require a top-down view but uses a stereo camera instead. This robot is unique because it classifies the pieces by type and color, whereas other robots identify and track move-by-move changes from the starting position (e.g., Golz & Biesenbach, 2015; Kołosowski et al., 2020; Larregay et al., 2018; Siraj, 2017).

Other research focuses on the development of computer-vision methods for the detection of the chessboard, and the subsequent classification of the pieces on it, without always linking these computer-vision methods to a chess robot (Christie et al., 2017; Czyżewski et al., 2020; Ding, 2016; Koray & Sümer, 2016; Ranasinghe et al., 2023; Schwenk & Yuan, 2015; Xie et al., 2018). The detection of the chessboard typically involves line- or edge-detection techniques (Chen & Wang, 2019; Czyżewski et al., 2020; Srivatsan et al., 2020; Wölflein & Arandjelović, 2021; Xie et al., 2018), while the classification of chess pieces commonly uses convolutional neural networks (CNN) (Mallasén Quintana et al., 2020; Shin et al., 2023; Wölflein & Arandjelović, 2021). Challenges in both cases lie in testing for robustness under various conditions. A common technique to improve classification performance is to verify if the move/position is legal or plausible according to a chess engine. With such methods, it is possible to let the algorithm home in on the most probable classification outcome (e.g., Czyżewski et al., 2020; Mallasén Quintana et al., 2020; Ranasinghe et al., 2023). Currently, a state-of-the-art method is that of Wölflein and Arandjelović (2021), which used a diverse set of neural network models and achieved a per-square classification accuracy of 99.8%.

Our contribution centers on a software platform that others can reproduce using open-source robotics middleware. The chosen robot arm is widely available in research, while other hardware components can be substituted if required. The OpenChessRobot can detect pieces using computer vision and execute moves. For verbal interaction, we rely on large language models. Specifically, by connecting ChatGPT to the chess engine output, our robot interprets strategies and explains them to humans in natural language. Non-verbally, the robot is able to express its evaluation of the game by either nodding or shaking its end-effector. Appendix A provides a comparison of the OpenChessRobot with other chess robots designed for HRI studies.

This paper first presents the hardware and software design of the chess robot. It then provides an outline of its core modules, which include Perception, Analysis and Evaluation, Motion

Planning and Execution, and Interaction. Module-specific experiments for performance evaluation are presented as well. Next, the paper reports findings from two human evaluations: an interview with an expert chess player who played two games with the robot, and an online study of 597 participants, which investigated perceptions of a talking robot equipped with AI in three different scenarios. Finally, the paper concludes with a discussion of the robot's limitations and outlines directions for future research in HRI.

4.2 Hardware and software architecture

Our chess-playing robot (Figure 4.1A) consists of the following hardware components (Figure 4.1B):

- A Franka Emika Panda robot arm (Franka Emika, 2020) equipped with a Franka Hand and a customized 3D-printed robot gripper. The robot arm has 7 degrees of freedom and is a widely used collaborative robot, suited for safe human interaction. It operates on firmware version 5.4.0.
- A ZED2 StereoLabs camera (StereoLabs, 2020)
- A keyboard
- An external microphone and a speaker, connected to the PC
- An NVIDIA Jetson Nano (NVIDIA, 2019)
- A Linux PC with Ubuntu 20.04, running a real-time kernel. It is equipped with an Intel i7-8700K processor and an NVIDIA RTX 2080 graphics card

Our setup includes a number of accessories. These comprise a chessboard (48×48 cm), a set of chess pieces (model: Staunton No. 4; ranging in height from 4 to 7.8 cm and in width from 2 to 4 cm), and four printed ArUco markers. Considering the robot arm's range, the chessboard should be positioned so that the farthest rank remains within 65 cm of the robot. In Figure 4.1B, the distance between the closest edge of the board and the robot is 16 cm.

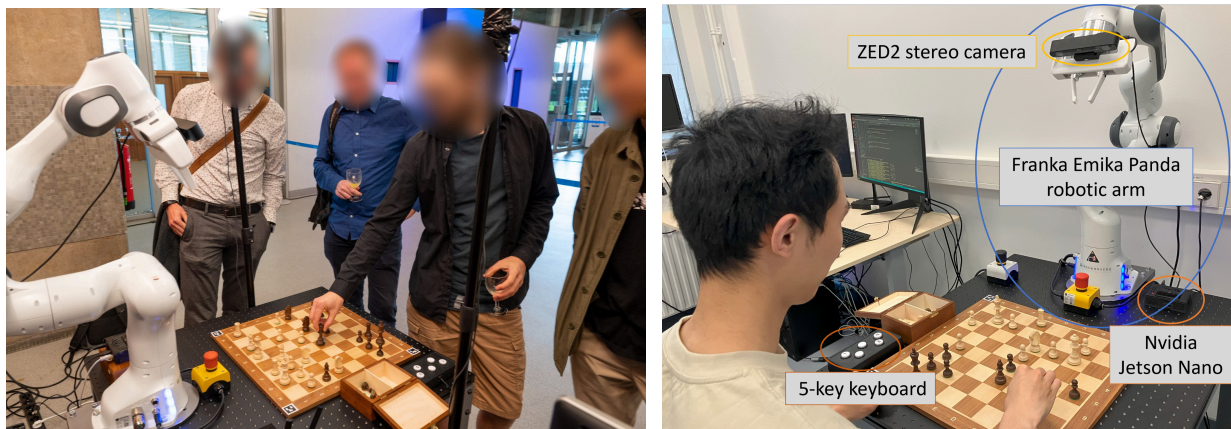


Figure 4.1. The system at a glance. (A) The robot at a demonstration event during which attendees could interact with it. (B) The complete system, comprising a Franka Emika Panda robotic arm, a ZED2 stereo camera, and an NVIDIA Jetson Nano computing board.

The robot arm, the Jetson Nano, and the Linux PC connect to a local network via a router. Communication between the arm and the control PC is realized through the Franka Control Interface (Franka Robotics GmbH, 2023) by integrating the Libfranka (0.9.2) library for low-level control and the franka_ros package for the Robot Operating System (ROS) (Quigley et al., 2009). The ZED2 camera is attached to the Franka Hand via a custom mount, and it interfaces with the NVIDIA Jetson Nano through a USB connection. The Jetson Nano streams the camera view to the PC through ZED SDK v3.5. Considering that our setup uses images from a single camera, the ZED2 camera and Jetson Nano can be substituted with more economical monocular alternatives.

The software architecture of the OpenChessRobot is built upon ROS Noetic (Open Robotics, 2020), which offers common robotics data formats and message-passing among software modules. Figure 4.2 provides an overview of the software architecture, divided into four modules: Perception, Analysis and Evaluation, Motion Planning and Execution, and Interaction.

The Perception module uses the ZED2 camera SDK to capture the chess game images and relies on a neural network-based chess identifier to translate the game images into text descriptions. The latter identifier requires PyTorch and CUDA runtime. The Analysis & Evaluation module feeds the chess game annotation to a chess engine in order to get predicted moves and their corresponding scores. In the Motion Planning & Execution module, the predicted move, accompanied by the 3D chessboard localization results from the Perception module, is used by the MoveIt! motion planner (Coleman et al., 2014) to plan and execute chess moves.

The Interaction module takes the human inputs and manages the OpenChessRobot's verbal and non-verbal feedback. For the verbal feedback, a prompt wrapper, which combines the user request and outputs from the Analysis & Evaluation module, serves as a client of OpenAI's ChatGPT API service and generates the responses to talk with the human.

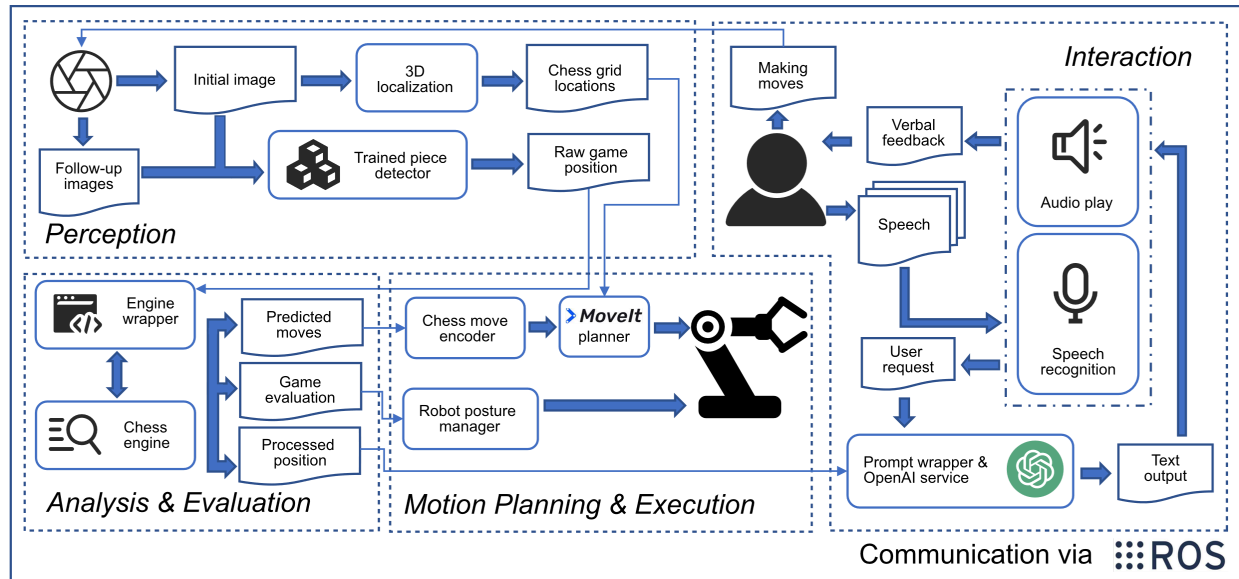


Figure 4.2. The software architecture of four modules and inter-module communication. Within the Perception module, the camera images are used to determine the 3D positions of the squares on the chessboard and identify individual pieces. The identified game position is analyzed through the chess engine wrapper, yielding the best next move and an evaluation of the current board position. The OpenChessRobot then enacts the chess move. Additionally, the robot is capable of responding through posture and speech.

4.3 Modules

The Perception module uses computer vision to identify the chessboard and the pieces; it can distinguish between occupied and empty squares and recognize pieces and their color. The perceived game position is sent to the Analysis & Evaluation module, which interfaces with a chess engine to determine the best move. The Perception module also calculates the chess squares' real-world locations, which are sent to the motion planner for planning the robot's end-effector trajectory to execute chess moves provided by the Analysis & Evaluation module.

The OpenChessRobot runs a data collection pipeline (orange box in Figure 4.3) and a human-robot gameplay pipeline (yellow box in Figure 4.3). The data collection pipeline is used to collect real-world data of chess pieces for retraining the Perception module and adapting to a new chessboard. The basic gameplay pipeline allows the robot to play the chess game with a human from any game position.

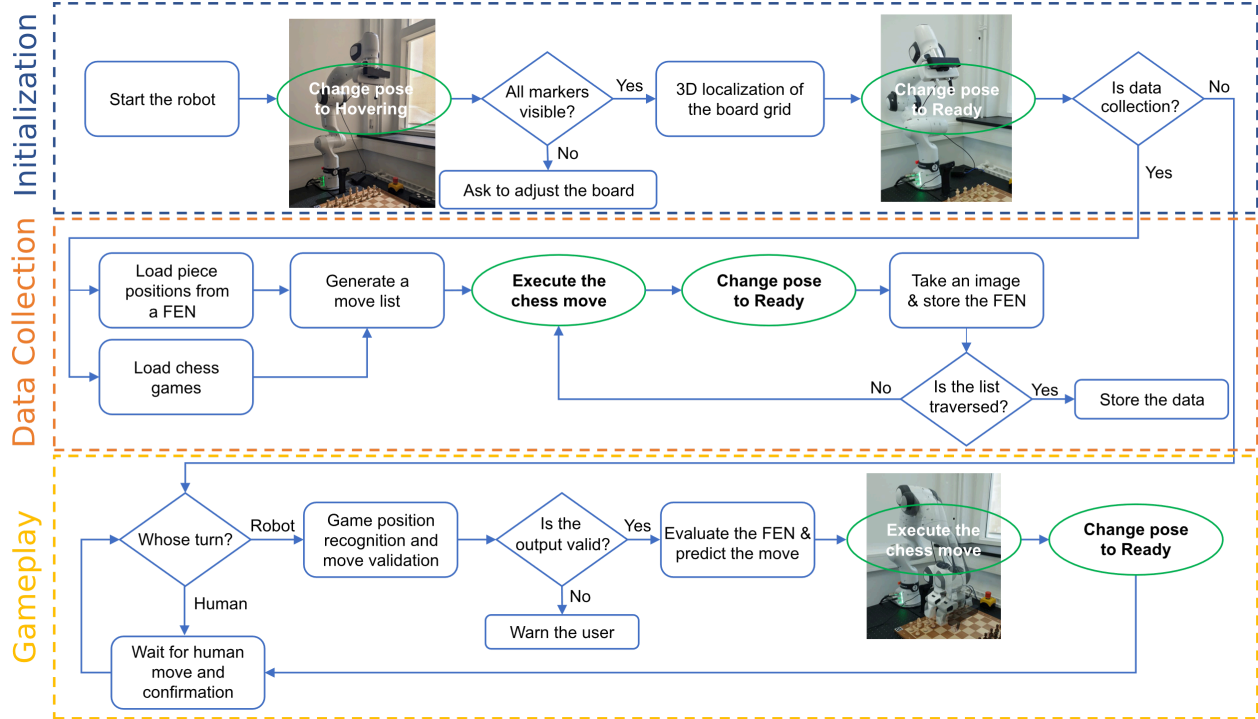


Figure 4.3. Flowchart of the OpenChessRobot execution using the core modules. The robot uses computer vision to recognize the chessboard and pieces, engages with a chess engine to determine moves, and executes chess moves for gameplay or data gathering. The chart splits into two key workflows: one for collecting data to improve the robot’s perception of different chessboards and chess pieces, and another for playing chess with a human.

4.3.1 Perception

The ability to perceive the chess pieces is a key function of the chess robot. While the commonly used top-down view resolves the issue of piece occlusion, its practicality is limited by its narrow perspective on the pieces. Adopting lower camera angles and dynamic camera positions to observe the game supports the robot’s natural interaction with the human player (see Figure 4.1). Additionally, our approach uses RGB images captured by a single camera instead of point clouds (Matuszek et al., 2011), improving the simplicity and reproducibility of our setup.

Figure 4.4 shows the Perception module of the OpenChessRobot. The module consists of two distinct classifiers, one for occupancy and the other for piece classification.

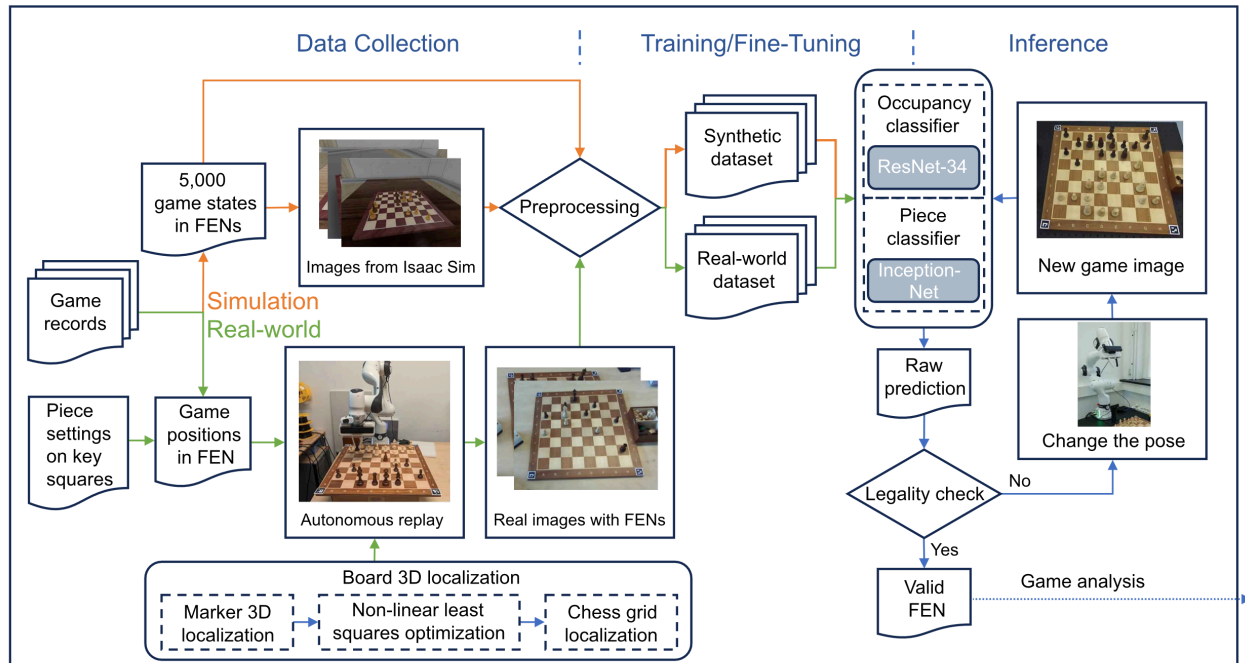


Figure 4.4. The Perception module of the OpenChessRobot relies on two distinct classifiers, one for occupancy and the other for piece classification. To train these classifiers, we synthesized chess positions in the NVIDIA Isaac Simulator (Makoviychuk et al., 2021) based on previous chess games and assigned ground truth labels, including the game positions represented in Forsyth–Edwards Notation (FEN) and pixel coordinates of board corners. In the real-world setup, the robot collects game images from the predefined camera pose (Figure 4.3) after autonomously placing pieces on key squares and replaying given games. These real-world images are then used to fine-tune the classifiers that were initially trained on synthetic datasets.

3D chessboard localization

To manipulate the chess pieces, their 3D coordinates in the real world need to be acquired. Four ArUco markers are used to indicate the 3D position of the chessboard. An image capturing these markers, along with the marker length and the camera’s intrinsic and extrinsic parameters and distortion coefficients, allows for the computation of the marker-to-camera translation and rotation using OpenCV.

The 3D board localization is activated when the OpenChessRobot is initialized. The robot moves to its Hovering pose to observe the chessboard from above (blue box in Figure 4.3). Although the robot typically operates with low camera positions during the game, a vertical camera angle is used for 3D chessboard localization to reduce errors caused by image distortion. Then, a grid corresponding to the chess squares is formed based on the actual square size, and a non-linear least squares algorithm is used to optimize the grid corner positions to closely match the marker positions.

Chess piece detection

In the preceding section, we explained how the 3D coordinates of the squares are estimated. The next task is to identify the chess pieces on each square.

The piece detection model is an extension of a CNN-based model from Wölflein and Arandjelović (2021). We complemented their method by using a different synthetic dataset as well as a real-world dataset to fine-tune the pretrained models in order to effectively handle our real-world chess set.

To synthesize images of chess pieces on a board, we varied backgrounds, lighting conditions, camera poses, and piece locations. A total of 5,000 game positions, randomly selected from games played by grandmaster Bobby Fischer, were used. We synthesized images of selected chess games using the NVIDIA Isaac Simulator (Makoviychuk et al., 2021). Ground truth labels were created, consisting of the game positions represented in Forsyth–Edwards Notation (FEN) and the positions of the board corners. Following a similar approach to previous works (Mallasén Quintana et al., 2020; Matuszek et al., 2011; Wölflein & Arandjelović, 2021), the images were cropped into sub-images, each corresponding to a square on the chessboard. Two categories, “empty” or “occupied” were assigned for occupancy classification, while 12 categories (representing the six types of chess pieces in both black and white) were designated for piece classification.

To train the occupancy and piece classifiers with the synthetic data, 80% of the cropped images were used as the training set, while the remaining 20% was evenly split between a test set and a validation set. A ResNet-34 (He et al., 2016) and an InceptionV3 (Szegedy et al., 2016) were trained independently on their respective training datasets. This is illustrated in the left portion of Figure 4.4.

To adapt to real-world chess, a piece-square dataset was created by manually iterating over all the chess pieces on the board. During the iteration, the game positions were recorded both in FEN notation and in images taken at 1080p resolution from the perspective of the robot in the Ready pose (Figure 4.3). A number of training datasets to fine-tune the piece classifier were extracted from the piece-square dataset using sub-images of chess pieces on key squares. The trained models were evaluated on the remaining portion of the piece-square dataset. Additionally, to evaluate the classifiers’ performance in games, a game dataset was collected by the robot autonomously replaying the provided chess games. Two games played by the grandmaster Michael Adams were used, with the game positions recorded both in FEN and in images, following the same procedure used when collecting the piece-square set. Dataset creation, training settings, and evaluation are explained in more detail in Section 5.1 and Appendix B.

The preceding chess detection model considers only the position of the chess game. It does not verify the legality of the observed game position and does not possess knowledge of the previous game position. Such information can potentially improve the accuracy of the Perception module. Our legality check consists of two steps: first, it verifies if the game position conflicts with the chess rules, such as detecting no king or too many pawns. If the game position is invalid, it requests a new image (using the same camera pose but with potentially better camera exposure) and reanalyzes it. The robot can also change the camera angle to improve perception. More specifically, in case the predicted game state is detected as illegal, the robot

slightly moves to its side in order to capture an image from a different perspective to improve the occupancy and piece classification.

In the second step of the legality check, the robot examines if the recognized chess move is among the legal moves from the previous game position. When the inferred game position and chess move prove to be legitimate, the robot adopts the inference and proceeds with gameplay. If not, the robot will halt its operation and wait for manual correction by the operator.

4.3.2 Analysis & evaluation

The processed chess FEN is forwarded to a chess engine wrapper, which uses the Universal Chess Interface (UCI) protocol for interaction with standard engines. Our chess engine wrapper builds upon an existing open-source Python wrapper for Stockfish (Zhelyabuzhsky, 2022). Our system integrates Stockfish 15.1 (Stockfish, 2022) as the default engine, using a total of 10 CPU threads. The chess engine assigns scores to candidate moves, which are monotonically related to the player's win rate, should the engine play against an equally strong opponent.

4.3.3 Motion planning & execution

For the planning of the OpenChessRobot's motions, we rely on the MoveIt! motion planner (Coleman et al., 2014). To determine the joint configurations, an inverse kinematics solver named IKFast is used (MoveIt, 2021). This solver calculates suitable robot joint angles based on the specified end-effector coordinates relative to the robot's base.

The robot arm is equipped with a customized gripper (Figure 4.5), designed to secure the underside of the piece, which is cylindrical in shape. The gripper is designed to be tolerant to minor deviations between the anticipated pickup positions (square centers) and the actual positions. Consequently, the human (or robotic) chess player is not bound to place a piece exactly at the center of a square.

There are two basic robot motions: a sliding motion (Figure 4.6A) and a jumping motion (Figure 4.6B). If the path between the start and end positions of a movement is unobstructed, the robot grasps the piece and pushes it to the destination square. When the path is obstructed, the robot raises the piece to leap over other pieces. Special movements like capture, castling, and en passant are manually programmed as combinations of the two basic motions.

A chess move for the robot to execute is provided by the Analysis & Evaluation module in an encoded text format, e.g., g1f310000. The first four characters indicate the chess move. Additional information for a move is stored in the remaining characters. Specifically, the fifth to ninth characters indicate whether the move is a jump, capture, castling, en passant, or promotion move.

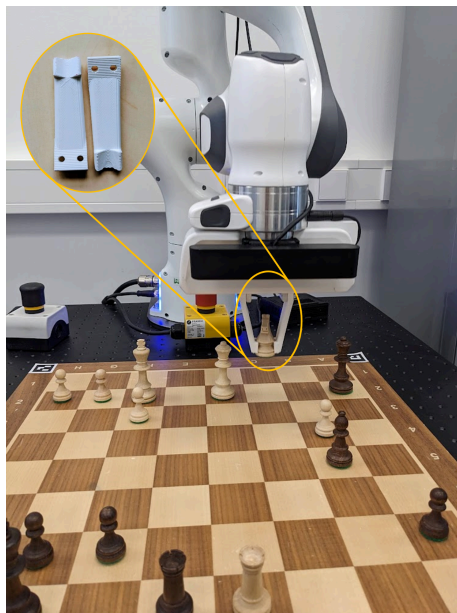


Figure 4.5. The customized 3D-printed gripper. This gripper has two fingers that are tolerant to deviations of chess pieces if they are not in the center of a square.



Figure 4.6. The robot is executing planned trajectories to move chess pieces. (A) The robot slides a pawn from the E2 square to the E4 square. (B) The robot lifts the knight on G1 to jump over other pieces on its way.

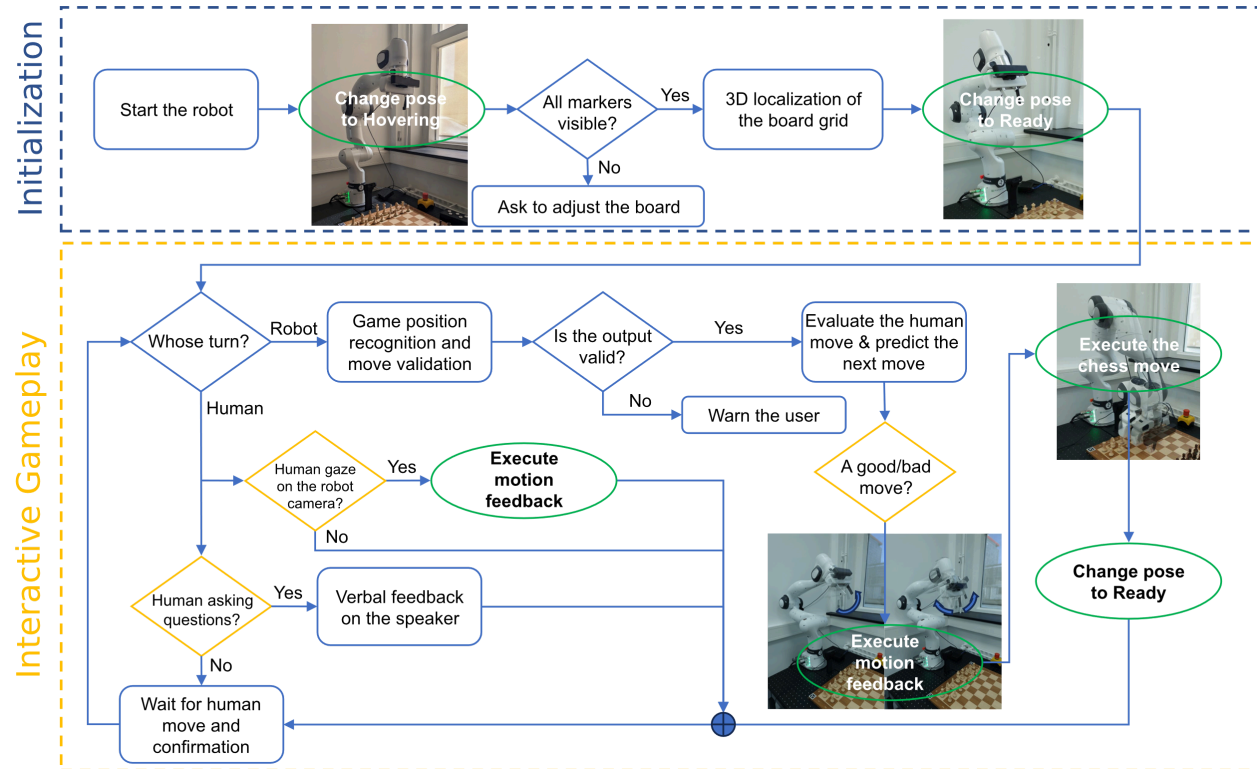


Figure 4.7. Flowchart of the OpenChessRobot execution using the core modules and Interactive module. As an extension of the core modules, the robot has an interactive gameplay pipeline. It evaluates human moves using predefined criteria and provides feedback by adjusting its posture. The system is also programmed to interpret human behaviors, such as gazing or asking questions, to provide a responsive and engaging experience.

With the encoded string, we create waypoints, using the 3D locations of the start square, the destination square, as well as intermediate states for special motions. The intermediate states are generated when the robot needs to lift pieces first rather than slide them on the board. The motion planner receives these waypoints and plans the trajectory of the end-effector using Rapidly-exploring Random Trees (RRTConnect; Kuffner & LaValle, 2000), with a 3D box surrounding the chessboard as a workspace constraint. After trajectory planning, velocity and acceleration limits are enforced on the planned trajectory (Kunz & Stilman, 2013).

4.4 Interactive modules

As the chess robot is designed for HRI research, providing various modalities for human-robot communication is important. We implemented an interactive gameplay pipeline based on the core modules (Figure 4.7). Specifically, depending on the human's chess performance, the OpenChessRobot can move its end-effector to express its evaluation, and it can provide verbal information regarding move qualities.

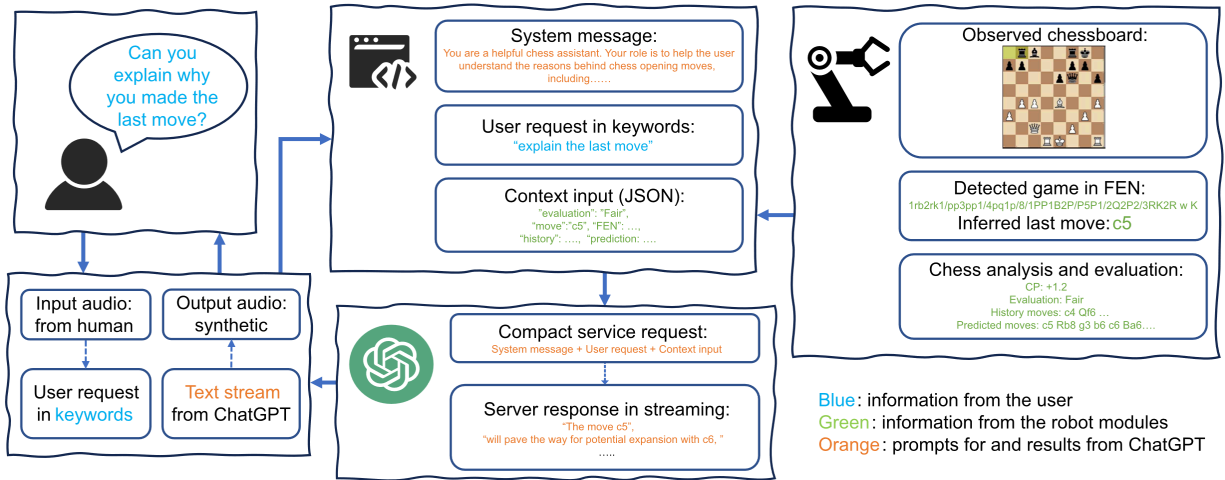


Figure 4.8. Verbal interaction. In this diagram, the user asks for an explanation of the last move. Next, a prompt consisting of a system message, the user request, and the processed game position is provided to the ChatGPT API to generate verbal feedback.

4.4.1 Verbal interaction

In typical chess games, verbal communication does not hold a central role, although some professional players have explained the rationale and inner thoughts of chess moves in “Banter Blitz”. Additionally, chess trainers commonly use verbal feedback to train their students. OpenAI’s ChatGPT has proven capable of generating human-like dialogues. However, ChatGPT is unable to properly play chess on its own (Kuo et al., 2023). Our proposed solution relies on combining the analysis from the chess engine with ChatGPT without model fine-tuning.

Depending on the human player’s request, the OpenChessRobot can explain the last move or the next move. The process is depicted in Figure 4.8. Capturing the user’s request is achieved through voice recognition. To trigger the voice interaction, the robot should be in the Ready pose, waiting for the user’s move. The player must vocalize specific keywords of the phrases, such as “explain the (last) move” or “analyze/predict the (next) move”.

When explaining the last move, the move history and the current game FEN are sent to the Analysis & Evaluation module. Alternatively, when the user seeks advice on subsequent moves, the system uses the chess engine to predict several top-ranked moves, along with their game continuations. A wrapper integrates global instructions and the chess game information into a prompt ready for submission to the ChatGPT API (using model GPT-4-0613). The global instructions, which remain consistent across requests, are embedded in a system message (Table 4.1).

In response to the prompt, the ChatGPT API is requested to provide an analysis within the specified chess context in a tutor-like tone. To reduce the latency in vocalizing text responses from the server while maintaining a natural flow in the synthetic voice, the feedback from the server, streaming in words/characters, is organized into short sentences, which are stored in a queue and played sequentially.

Table 4.1. Prompt and ChatGPT output to explain the last move (bishop c4)

Prompt	
System message	You are a helpful chess assistant. Your role is to help the user understand the reasons behind chess opening moves, including their benefits, weaknesses, and overall strategic concepts. There are some instructions on how to respond to the user. Instruction 1: You should always start your response with a short sentence or few words to acknowledge that you've received the user's message and are considering it. Instruction 2: The user prefers concise explanations, so please keep your responses brief. Instruction 3: Your response should be informative and fluid, avoiding bullet points or index numbers. Instruction 4: When analyzing a board, mention one key reason for a move. Try to use the given history moves and predicted future moves. The input will include a user question, a chess FEN for the current game, the move to make, predicted future moves and a move history in a dictionary format under the keys 'question', 'fen', 'move', 'evaluation', 'future' and 'history'. Input:
User message	<pre>{'question':'can you explain?', 'fen':'r1bqkbnr/pppp1ppp/2n5/4p3/4P3/5N2/PPPP1PPP/RNBQKB1R w KQkq - 2 3', 'move':'Bc4', 'evaluation':'Excellent', 'future':'Bc4 Bc5 d3 Nf6 O-O d6 c3 O-O h3 h6 ', 'history': '1. e4 e5 2. Nf3 Nc6'}</pre>
GPT-4-0613	
Sure, I can assist with that. The move Bc4 in this position is known as the Italian Game. The aim of this move is to control the center quickly with your pawn and knight and then put your bishop on its most dangerous square. After this, both white and black usually castle and then start a pawn storm against the opponent's king. In this case, future moves predict a traditional development of pieces with both sides castling kingside. The main goal is to have a solid position and try to exploit tiny advantages.	

4.4.2 Non-verbal interaction

In human-human communication, non-verbal cues play an important role. In professional chess games, verbal exchanges are often sparse. Apart from the moves played, participants may rely on behavioral cues to understand their opponents.

The posture feedback consists of basic nodding and shaking motions executed by the robot's hand (Figure 4.9). These gestures are triggered based on the chess engine's evaluation. Specifically, the robot enacts nodding or shaking gestures when the reduction in win probability due to the latest move exceeds a predefined threshold.

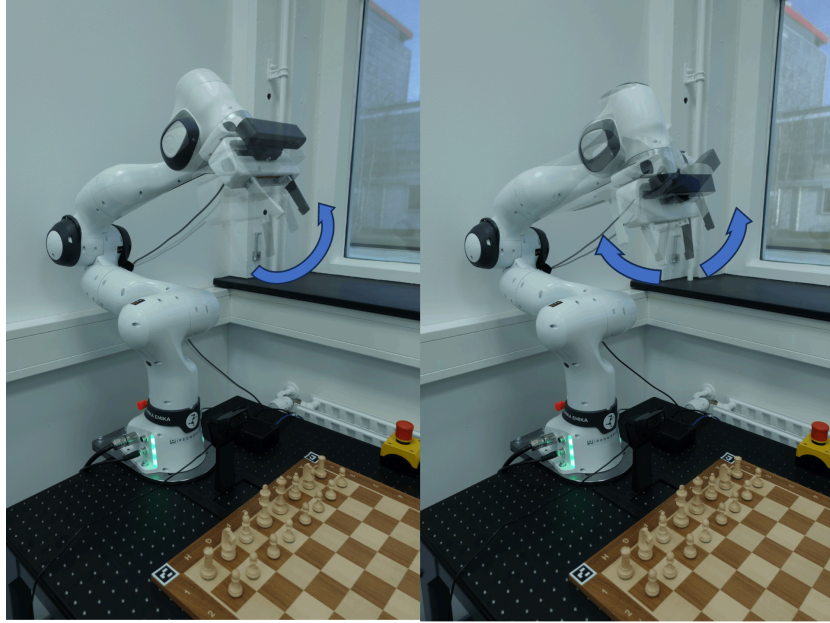


Figure 4.9. Posture feedback of the robot nodding and shaking its end-effector after a chess move was evaluated as good or bad, respectively.

4.5 Technical evaluation

We performed the following assessments of the chess robot system: (1) Perception module: accuracies of the piece classifiers; (2) Motion Planning & Execution module: success rates of grasping a piece from different sections of each square.

4.5.1 Chess perception

For the piece classifier of the chess perception module, several training settings were tested, using two sets of key squares, different components of the training dataset, and varying training lengths. The challenge was to identify a training setting that minimized the number of training samples while achieving good generalization performance.

The two sets of key squares were:

- 3×3 squares, which include intersections between the key files A, E, and H and the key ranks 1, 4, and 8.
- 4×4 squares, which contain intersections between the key files A, D, and E, H and the key ranks 1, 4, 5, and 8.

The piece-square dataset contains three subsets: default, rotated, and visually shifted:

- The default dataset (**D**) contains sub-images of 768 unique piece-square pairs ($64 \text{ squares} \times 12 \text{ pieces}$), with each pair occurring at least once.
- The rotated dataset (**R**) includes sub-images of four types of pieces: white king, white knight, black king, and black knight. Kings were rotated twice by 45° ($64 \text{ squares} \times 2 \text{ rotations}$), while knights were rotated four times ($64 \text{ squares} \times 4 \text{ rotations}$) to create different shapes

due to their rotational asymmetry (see Figure 4.10 right). Sixty-four samples of each rotated piece were randomly selected, forming the final rotated dataset.

- The visually shifted dataset (**S**) contains sub-images with cropping windows shifted in four directions (up, down, left, right). The displacement in each direction was $\frac{1}{4}$ square in length. For each shift direction, all 768 unique piece-square pairs were extracted. The **S** dataset serves as a test set with increased noise.

Additionally, the data recorded by the OpenChessRobot while replaying the games was used as another test set (**G**) to evaluate the piece classifiers trained with piece images on the key squares. Only unique piece-square pairs in each game were extracted, making up the **G** dataset.

Different compositions of training and test datasets were extracted from these three datasets:

- The training set (**d**) contained sub-images of each piece on the key squares. The remaining samples in the **D** dataset were divided into a validation set and a test set in a 20:80 ratio.
- The training set (**r**) was generated from the **R** dataset using pieces on the key squares rotated by 90° .
- Similarly, the training set (**rr**) contains all pieces on the key squares with all the rotations from the **R** dataset. The corresponding test sets and validation sets of **r** and **rr** were created in the same manner as the **D** dataset.

The fine-tuning was conducted in two phases: in the first phase, the weights of most layers of the model were frozen, and only the last layer (head) was trained over epochs with a learning rate of 0.001. In the second phase, all the model's weights were trained with a learning rate of 0.0001. The Adam optimizer was used in both phases.

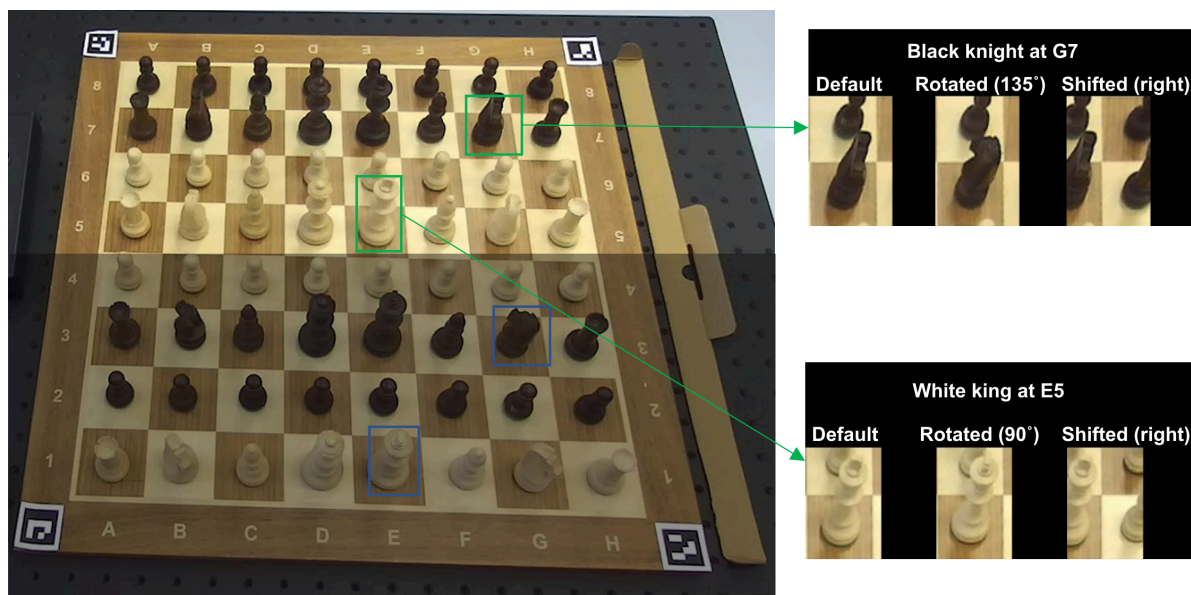


Figure 4.10. The raw images to create the piece-square dataset. The upper half image is taken without any rotations, and the lower half image is taken with 135-degree piece rotations.

Following the aforementioned training conditions, the piece classifier models were evaluated on their corresponding test sets, as well as on the **S** and **G** datasets. The evaluation results are detailed in Table 4.2. When comparing the two selections of key squares based on the model accuracies on the **G** dataset, it is evident that using 4×4 squares provides more samples for fine-tuning and results in good accuracies with better robustness. Additionally, including the rotated samples in the training set also helps improve performance, but it requires a longer training duration to be effective. Results on the **S** dataset show that using 4×4 squares provides better robustness against the noise caused by piece displacements compared to using 3×3 squares. While longer training durations and adding rotations to the training set yield some improvements, these improvements are not substantial.

Regarding the occupancy classifier, it was found to achieve 100% accuracy on the **G** dataset after retraining for 50 epochs using piece sub-images at the 4×4 squares and all the adjacent empty squares.

Table 4.2. Piece classifier accuracies in different training conditions. The best accuracies on the S and G datasets are underlined.

Key squares		3×3								
Epoch		500+500			1000+500			1000+1000		
Test set	Training set	d	d+r	d+rr	d	d+r	d+rr	d	d+r	d+rr
D-d		0.9924			0.9943			0.9943		
D+R-d-r			0.9794			0.9835			0.9863	
D+R-d-rr				0.9915			0.9887			0.9930
S		0.9762	0.9704	0.9684	0.9704	0.9691	0.9759	0.9652	0.9688	0.9740
G		0.9932	<u>1.0000</u>	0.9932	0.9865	0.9662	<u>1.0000</u>	0.9932	0.9865	<u>1.0000</u>
Key squares		4×4								
Epoch		500+500			1000+500			1000+1000		
Test set	Training set	d	d+r	d+rr	d	d+r	d+rr	d	d+r	d+rr
D-d		0.9978			1.0000			1.0000		
D+R-d-r			0.9939			0.9939			0.9954	
D+R-d-rr				0.9984			0.9922			1.0000
S		0.9837	0.9831	0.9844	0.9834	0.9883	<u>0.9886</u>	0.9814	<u>0.9886</u>	0.9844
G		0.9865	<u>1.0000</u>	<u>1.0000</u>	<u>1.0000</u>	<u>1.0000</u>	<u>1.0000</u>	<u>1.0000</u>	<u>1.0000</u>	<u>1.0000</u>

4.5.2 Grasping of pieces

An experiment was performed in which the OpenChessRobot was tasked with grasping various pieces positioned on different squares of the chessboard, repeated multiple times. To reduce positional discrepancies, a compensation of 1 cm along the y-axis and 0.55 cm along the x-axis (Figure 4.11) was applied as the system error.

Specifically, the evaluation focused on testing the grasp performance using the largest piece, the king (4 cm in width, 7.8 cm in height), and the smallest piece, the pawn (2 cm in width, 4 cm in height). These pieces were positioned at each of the four corner squares (A1, A8, H1, H8) and physically shifted from the edges of the squares to their centers (depicted in Figure 4.11). Two shifting distances were used: (1) placing the piece tangent to the square's edge or (2) 0.625 cm away from the edge, corresponding to $1/8$ the length of a square. Each piece in each square position was tested 10 times.

Figure 4.12 shows the success rates for the different independent variables of the grasping experiment. A distinction is made between three categories: (1) *accurate grasping*: a correct grasp in 10 out of 10 trials, (2) *remedied grasping*: a correct grasp in 8 or 9 out of 10 trials (usually correct grasping occurred here in such a way that the piece slid into the grasper's teeth), and (3) *missed grasping*: a correct grasp in 0–7 out of 10 trials. Table 4.3 shows the corresponding results in numerical form. A trend can be noticed, whereby the grasping was least successful when the piece was positioned toward the edge of the square or toward the 'up' direction, i.e., in the positive direction of the y-axis (see Figure 4.11), positioned away from the base of the robot arm. It is hypothesized that this effect stems from image projection and coordinate transformation errors in the 3D chessboard localization.

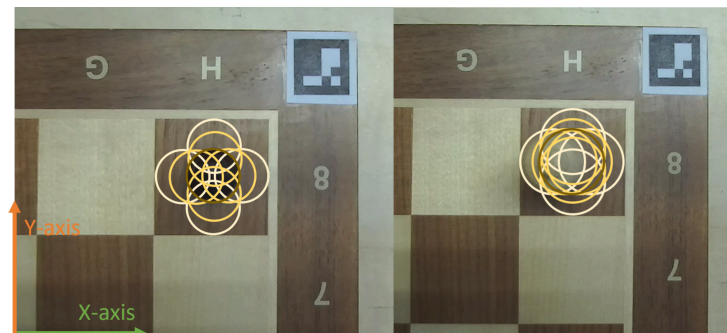


Figure 4.11. The positions by physically shifting the piece at square H8. In the left image, the black pawn is in the center without shifting. In the right image, the white king shifts to the left edge of the square. The brown circle shows the center position. The white circles indicate the first shifting distance (to the edge), and the yellow circles indicate the second shifting distance (intermediate).

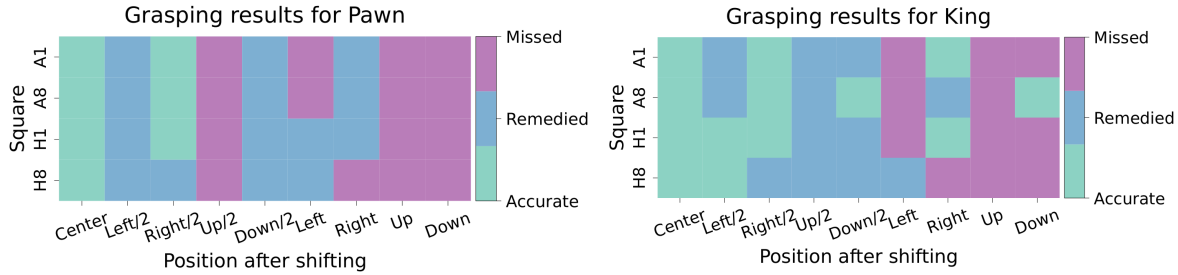


Figure 4.12. Results from the grasping experiment.

Table 4.3. Piece grasping results. The percentages represent the number of trials that were marked as ‘accurate grasping’ or ‘remedied grasping’.

Piece	Overall		Shift Distance		Shift Direction			
	Without Shifting (<i>n</i> = 4)	With Shifting (<i>n</i> = 32)	Halfway (<i>n</i> = 16)	Edge (<i>n</i> = 16)	Left (<i>n</i> = 8)	Right (<i>n</i> = 8)	Up (<i>n</i> = 8)	Down (<i>n</i> = 8)
Pawn	100%	65%	75%	31%	75%	88%	0%	50%
King	100%	78%	100%	31%	63%	88%	50%	63%

4.5.3 Performance in human-robot chess play

To assess the core components of the chess robot, a chess game between a human and the OpenChessRobot was conducted. The robot had control of the white pieces while the human played with the black pieces. To determine its next move, Stockfish 15.1, with a search depth of 20, was used.

The OpenChessRobot initiated the game from its Ready pose, executed a move, and subsequently returned to its Ready pose, awaiting the human player’s move. The human player was prompted to press the middle keyboard button (see Figure 4.1B) upon completing their move. The manual button press was maintained to mirror traditional chess tournament practices and ensure clear separation of moves. This design also maintains safety and user control safeguards by providing an explicit signal before the robot begins its movement. The robot then captured an image, identified the human’s move, and made its subsequent move. Two demo videos of human-robot chess play can be found in Appendix C.

To ensure a fair match, the human player also had access to Stockfish 15.1 to determine their next best move. The game ended when a checkmate was achieved or a draw due to repetition. Throughout the experiment, we recorded the time taken by each robot module as it carried out its respective tasks.

The game ended in a draw after 97 moves, including 49 robot moves and 48 human moves. Out of these 49 robot moves, 7 involved capturing pieces, wherein the robot first removed the captured piece before executing its move, 6 moves required jumping pieces over others, resulting in longer execution times compared to moves made on unobstructed paths, and one was a castling move. The other 35 moves followed the pick-and-slide approach.

The time taken for the OpenChessRobot to execute its moves is depicted in Figure 4.13A. Each interval is measured from the robot's home position back to its home position upon completion of the full movement sequence. On average, it took 7.33 s, while capture moves averaged 6.24 s, slightly less, as captures in this game involved shorter transport distances overall. Figure 4.13B shows the time allocation for move detection, evaluation, and prediction. Move detection was typically completed within 1 s, with one outlier attributed to camera failure. Evaluating a single move with Stockfish took approximately 1 s. Finding the next move with Stockfish generally took approximately 5 s, unless Stockfish had already encountered the current human move during its previous search. In such cases, the search time was virtually zero, allowing the robot to execute the next move immediately. The relatively large computation time can be explained by the fact that Stockfish was allowed to search at a high depth, which is typically unnecessary for amateur-level play.

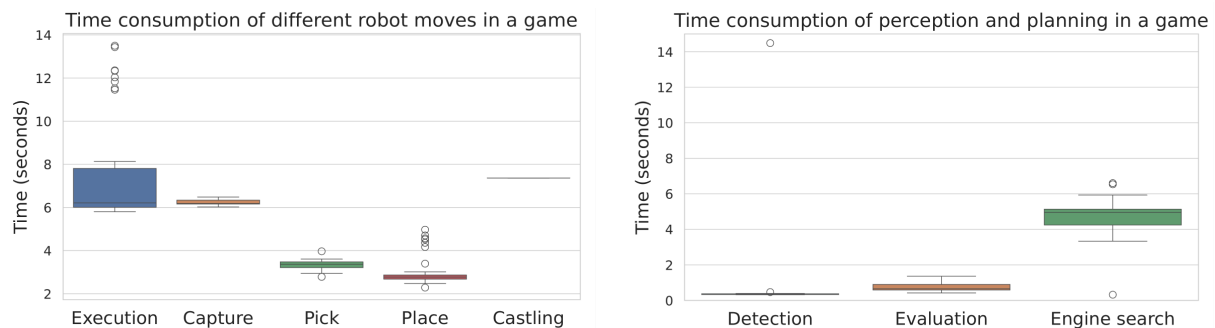


Figure 4.13. (A) Time consumption of different robot moves in the game. (B) Time consumption of move detection, Stockfish evaluating a move, and Stockfish searching for the next move.

4.6 Evaluation of people's views on the chess robot

To collect an expert chess player's opinion on the OpenChessRobot, we invited a FIDE Master to play against the robot in two games (Appendix D). The interview conducted after playing two games revealed that the robot, while potentially useful for beginner-level practice, lacks human-like feedback or commentary from which experts might benefit. The player also indicated that the robot might be useful for entertainment purposes.

In addition to the evaluation by an expert player, we conducted an online study to investigate how the broader population perceives the OpenChessRobot and to gather opinions on the future uses of such embodied AI in general. Participants watched a demo video of the robot and completed a corresponding questionnaire.

Specifically, demo videos were shown featuring the robot in three scenarios:

- The scenario *Robotics Education* featured the robot explaining how it works. The 28-s demo video showed the robot describing its perception function and how it localizes the chessboard.
- In the *Chess Coach* scenario, the robot teaches a chess opening. The 36-s video demonstrates specific moves and highlights key squares on the chessboard.

- The scenario *Home Entertainment* represents a future household robot that intends to entertain people through conversation and chess play. The 42-s demo video showed the robot discussing a chess game from a movie (Campbell, 1997). The videos and corresponding transcripts can be found in Appendix C.

The choice of these scenarios is partly based on the aforementioned experiences with the expert chess player and also represents how (chess) robots could potentially be used in the future. That is, chess robots do not necessarily have to provide chess lessons (*Chess Coach* scenario) but can also support education by explaining their own functionality and serving as a knowledge resource (*Robotics Education* scenario). Additionally, robots can be seen as a companion or interaction partner (*Home Entertainment* scenario).

4.6.1 Participants

We aimed to recruit 600 participants through the online research platform Prolific (Douglas et al., 2023; Peer et al., 2022) from six countries where English is an official or de facto primary language: Australia, Canada, Ireland, New Zealand, the United Kingdom, and the United States.

A questionnaire was launched on Prolific on Wednesday, 12 February 2025, at 22:00 Central European Time. In Prolific, prospective participants were provided with a hyperlink to the survey platform Qualtrics to complete our questionnaire titled “User perception of a talking chess robot”. The purpose of the study was described as “*to examine how people evaluate a talking chess robot, which is powered by artificial intelligence including ChatGPT*”. Once the target of 600 active participants was reached, the survey was automatically closed. Ultimately, 597 responses were collected. Respondents gave informed consent via a dedicated questionnaire item. Each respondent received a reward of £1.50 for completing the estimated 10-min questionnaire. The research was approved by the TU Delft Human Research Ethics Committee (Application Number 5109).

4.6.2 Experiment design

The online study followed a between-subjects design, with each participant presented with one of the three demo videos. All participants first answered questions about their familiarity with ChatGPT. After this, a short text was presented introducing one of the three demo videos, such as (*Chess coach* scenario): “*The following video shows a talking chess robot that uses artificial intelligence (including ChatGPT) to give a chess lesson. The robot explains a chess opening. Please turn on your sound.*”

Next, the demo video was played, followed by questions about people’s acceptance (Q1), impression of the technical soundness of the robot (Q2), possible improvements (Q3), and opinions on the potential for such talking robots equipped with AI to replace existing technology (Q4) or human workers (Q5). The first three questions collected people’s perceptions of the current OpenChessRobot, and the last two questions aimed to acquire their opinions on the future use of such talking robots equipped with AI. The five questions were as follows:

Q1. “*Do you think this robot could be used <in schools to explain to students how a robot*

works?> / <in schools to teach students chess?> / <at home to provide entertainment?>" (Definitely not, Probably not, Neutral, Probably Yes, Definitely yes, I prefer not to respond)

Q2. "Comment on the technical quality of the robot as shown in the video. Mention positive or negative aspects." (A response of at least 20 characters was required)

Q3. "Do you have any recommendations about how to improve the robot?" (A response of at least 20 characters was required)

Q4. "Do you think that within 10 years, robots like this could serve as a replacement of existing screen-based <computers to help understand technical topics such as computer vision?> / <computers to improve chess skills?> / <forms of entertainment such as television?> Please elaborate on why or why not." (A response of at least 20 characters was required)

Q5. "Would you welcome a future where robots that can talk, equipped with artificial intelligence (such as improved versions of ChatGPT), replace human workers?" (Definitely not, Probably not, Neutral, Probably Yes, Definitely yes, I prefer not to respond)

The remaining questions gathered demographic information (age, gender, and education) and assessed participants' attitudes toward and familiarity with chess, technology, and the movie used in the *Home Entertainment* scenario. The full questionnaire can be found in Appendix E.

4.6.3 Data analysis

For the open-ended questions (Q2–Q4), we used ChatGPT o1 (OpenAI, 2024) to analyze the responses (see Appendix F for the full prompts and raw outputs). Previous research shows the possibility of using large language models to analyze textual data from user studies (Tabone & De Winter, 2023). In addition, the reasoning model ChatGPT o1 has demonstrated its efficiency in textual content classification for social media analysis, sentiment analysis (Zhong et al., 2024) and categorical reasoning (Latif et al., 2024). We requested ChatGPT o1 to extract the three most mentioned positive and negative aspects from responses to Q2, the five most mentioned recommendations from responses to Q3, and the three most mentioned concerns from responses to Q4.

Additionally, for the open-ended question Q4, we used two reasoning models, ChatGPT o3-mini-high and Gemini 2.0 Flash Thinking, as text classifiers to classify the responses regarding people's opinions as *positive*, *neutral*, or *negative*. If the two classifiers produced different predictions, a human annotator manually determined the final label.

4.6.4 Results

A total of 195, 211, and 191 participants were randomly allocated to the *Robotics Education*, *Chess Coach*, and *Home Entertainment* scenarios, respectively. The median time to complete the survey was 6.48 min. The mean age of the participants was 36.5 years ($SD = 12.33$, $n = 597$). The gender distribution was: 302 females (50.6%), 286 males (47.9%), 8 'other' (1.3%), and 1 "I prefer not to respond".

Figure 4.14 presents the distribution of responses for Q1, Q4, and Q5. For Q4, the two reasoning models classified the opinions identically for 528 participants, while a human annotator manually labeled the remaining 69 responses.

Analysis of acceptance per scenario

Acceptance was the highest for the *Robotics Education* scenario (Figure 4.14A), with a mean of 3.96 on a scale of 1 (“*Definitely not*”) to 5 (“*Definitely yes*”) ($SD = 0.98$, $n = 195$) (Q1). According to the ChatGPT o1 analysis, the most mentioned positive aspects for this scenario were the “*Advanced technical ability*” in terms of computer vision and the “*Clear voice/explanation*” demonstrated by the robot (Q2). However, participants often found the technical language in this scenario rather hard for beginners or students to comprehend.

The acceptance of the *Chess Coach* scenario was the second highest, with a mean of 3.82 ($SD = 1.05$, $n = 210$, excluding one participant who chose “*I prefer not to respond*”) (Q1). The most mentioned positive aspect specifically for this scenario according to ChatGPT o1’s analysis of Q2 was that it was seen as an “*effective teaching approach*”. However, participants also often mentioned the “*limited interactivity*” as a downside.

The acceptance of the Home Entertainment scenario was the lowest (mean = 3.59, $SD = 1.07$, $n = 191$), with only 15% of participants answering “*Definitely yes*” to Q1. A similar pattern can be observed in the classified opinions of Q4, where a majority of participants (57%) expressed a negative expectation regarding the use of such a robot to replace screen-based entertainment like television (Figure 4.14B).

Analysis of open-ended questions across the three scenarios

The summary results of the open-ended questions are shown in Table 4.4. Participants frequently praised the smooth and precise movements of the robot and its clear teaching style, but they criticized its robotic voice, mechanical noise, and slow operation. Common recommendations included adopting a more natural, human-like voice, simplifying the explanations, reducing the mechanical noise, speeding up performance, making the design more friendly, while keeping the robot cost-effective.

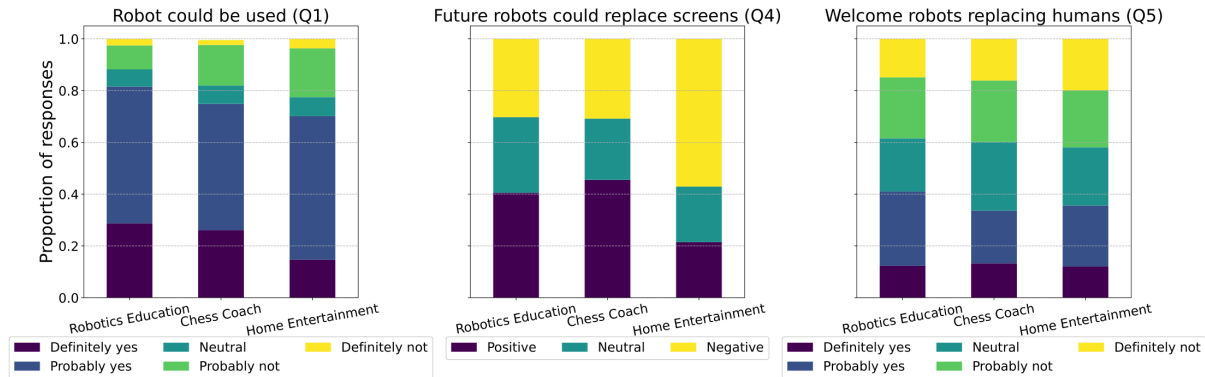


Figure 4.14. (A) The proportion of responses to Q1 “Do you think this robot could be used ...”, (B) Q4 “Do you think that within 10 years, robots like this could serve as a replacement of existing screen-based ... Please elaborate on why or why not.” (answers classified into three categories), and (C) Q5 “Would you welcome a future where robots that can talk, equipped with artificial intelligence (such as improved versions of ChatGPT), replace human workers?”.

Table 4.4. Summary of results from the open-ended questions analyzed by ChatGPT o1. Underlined items indicate aspects that were commonly mentioned across all three scenarios.

	Theme	Robotics Education	Chess Coach	Home Entertainment
Q2	The three most mentioned positive and negative aspects	Positive: - Smooth and precise movement - Clear voice/explanation - Advanced technical ability (vision, analysis) Negative: - Overly technical/Complex language - Slow or lengthy process - Robotic or distracting Voice/Noise	Positive: - Smooth and precise movement - Clear explanations - Effective teaching approach Negative: - Robotic or unnatural voice - Noisy or distracting movements - Limited interactivity	Positive: - Smooth and precise movement - Intelligent or knowledgeable - Entertaining or interactive Negative: - Loud or noisy - Bulky or large - Slow or boring
Q3	The five most mentioned recommendations	- More natural, varied, or human-like voice - Simpler, age-appropriate explanations - Friendlier or more appealing design - Reduced mechanical noise - Faster or smoother operation	- More natural, simpler speech - Reduce mechanical noise - Friendlier design and smaller size - Greater interactivity and Q&A - Clearer movement cues	- Reduce mechanical noise - Use a more natural, expressive voice - Make the design more compact - Improve overall aesthetics/approachability - Enhance movement fluidity
Q4	The three most mentioned concerns	- High cost and limited cost-effectiveness - Acceptance and human interaction - Incomplete or restricted functionality	- High cost - Limited portability - Reduced versatility	- Lack of variety and visual/storytelling depth - Preference for passive entertainment - Insufficient human/emotional element

4.7 Discussion and conclusion

In this paper, we introduced an open-source cognitive robot designed for engaging in chess matches with humans. While multiple chess robots have been developed over the years, none have been made available as a reproducible platform, and many of them have not been specifically designed for conducting HRI studies. Our robot takes a unique approach by integrating robust robotic perception, evaluation of gameplay, and move execution with verbal and non-verbal interactions. Our development prioritized adaptability and reproducibility, to create an accessible platform for researchers and enthusiasts alike.

We focused on the game of chess because of its value as a controlled experimental environment where both human behavior and machine performance can be accurately measured. Chess also serves as a ‘battleground’ for Moravec’s paradox, which states that while machines excel at computational tasks (such as playing chess), they traditionally struggle with tasks requiring human-like perception, motor control, and language processing (Moravec, 1988). Our research platform aims to transcend Moravec’s paradox by integrating chess engine capabilities with innovations in human-like perception and the verbalization of moves using synthetic speech.

Furthermore, the integration of verbal and non-verbal interactions is included to increase the depth of engagement between humans and the robot. By using a large language model for verbal communication, we tried to create a solution that lowers the barrier of understanding the reasons behind robot chess moves. Our platform, in essence, revisits the conceptual foundation laid by the Mechanical Turk in the 18th century (Standage, 2002).

We invited an expert chess player to play with the platform and found that although it can be useful for beginners, it failed to provide human-like in-depth feedback from which expert-level players might benefit. Additionally, we conducted an online survey with 597 participants recruited from six countries to gather their views about the OpenChessRobot as well as talking robots equipped with AI in general. The online survey results revealed that participants favored the robot’s use in educational contexts (where its advanced technical skills were appreciated) while expressing reservations about its role in home entertainment, primarily due to concerns over limited interactivity and its inability to replace traditional screen-based media. These findings motivate future research on HRI in educational settings using this robot.

One of the limitations of the OpenChessRobot, as also shown in Table 4.4, is that its movement is rather slow and noisy. Another limitation is that the ChatGPT-based explanations are still narrow in scope and quality. To improve the robot, we will focus on faster arm movements and the incorporation of advanced large language models that are specifically trained or fine-tuned on chess-related text databases (see also Feng et al., 2023). Here, there is a need for explainable AI, where the chess position, represented by a FEN notation, is translated into a verbal explanation that is not only correct in terms of the chess engine’s evaluation (as our current robot already does) but also contains meaningful content about why a particular move is good or bad. Moreover, although the results from the online study demonstrate promising

aspects of the platform, any application beyond the research setting, such as education, coaching or entertainment, should be researched further. Finally, a limitation of the current paper is the lack of physical human-subject experiments among a large number of chess players. In the future, we intend to use this setup to study how AI-embodied robots influence people during interactions. This will involve the robot communicating with humans through emotional expressions and more natural/meaningful verbal interactions.

In conclusion, this paper presents the OpenChessRobot, an open-source, reproducible cognitive chess robot that integrates robust computer vision, chess engine evaluation, and both verbal and non-verbal interactions for HRI studies. A large language model is used to translate the evaluation from the chess engines into human-like speech. Supported by an international survey and expert gameplay, our findings show the robot's potential in educational and research contexts, while highlighting limitations in human-like interaction.

Funding

The author(s) declare that financial support was received for the research, authorship, and/or publication of this article. Specifically, this project has received funding from the Cohesion Project of the Faculty of Mechanical Engineering, Delft University of Technology.

Data availability statement

Models for chess recognition used in the OpenChessRobot software, material from the online study, images used to fine-tune and evaluate the models, raw data, and scripts for reproducing the analyses in the paper are available in a public data repository: <https://doi.org/10.4121/1cb5bf64-468e-462a-a82e-c847d88a7a86>

References

- Campbell, M. (1997). An enjoyable game: How HAL plays chess. In D. G. Stork (Ed.), *HAL's legacy: 2001's computer as dream and reality*. MIT Press. <https://doi.org/10.7551/mitpress/3404.003.0007>
- Campbell, M., Hoane, A. J., Jr., & Hsu, F.-h. (2002). Deep Blue. *Artificial Intelligence*, 134, 57–83. [https://doi.org/10.1016/S0004-3702\(01\)00129-1](https://doi.org/10.1016/S0004-3702(01)00129-1)
- Chen, A. T.-Y., & Wang, K. I.-K. (2019). Robust computer vision chess analysis and interaction with a humanoid robot. *Computers*, 8, 14. <https://doi.org/10.3390/computers8010014>
- Christie, D. A., Kusuma, T. M., & Musa, P. (2017). Chess piece movement detection and tracking, a vision system framework for autonomous chess playing robot. *Proceedings of the 2017 Second International Conference on Informatics and Computing*, Jayapura, Indonesia. <https://doi.org/10.1109/IAC.2017.8280621>
- Coleman, D., Sucan, I., Chitta, S., & Correll, N. (2014). *Reducing the barrier to entry of complex robotic software: A MoveIt! case study*. arXiv. <https://doi.org/10.48550/arXiv.1404.3785>
- Czyżewski, M. A., Laskowski, A., & Wasik, S. (2020). Chessboard and chess piece recognition with the support of neural networks. *Foundations of Computing and Decision Sciences*, 45, 257–280. <https://doi.org/10.2478/fcds-2020-0014>
- Ding, J. (2016). *ChessVision: Chess board and piece recognition* (Technical report). Stanford University. https://web.stanford.edu/class/cs231a/prev_projects_2016/CS_231A_Final_Report.pdf
- Douglas, B. D., Ewell, P. J., & Brauer, M. (2023). Data quality in online human-subjects research: Comparisons between MTurk, Prolific, CloudResearch, Qualtrics, and SONA. *PLOS ONE*, 18, Article e0279720. <https://doi.org/10.1371/journal.pone.0279720>

- Feng, X., Luo, Y., Wang, Z., Tang, H., Yang, M., Shao, K., Mguni, D., Du, Y., & Wang, J. (2023). ChessGPT: Bridging policy learning and language modeling. *Advances in Neural Information Processing Systems*, 36.
https://proceedings.neurips.cc/paper_files/paper/2023/file/16b14e3f288f076e0ca73bdad6405f77-Paper-Datasets_and_Benchmarks.pdf
- Franka Emika. (2020). Panda powertool.
<https://web.archive.org/web/20200102112015/https://www.franka.de/technology>
- Franka Robotics GmbH. (2023). Franka Control Interface documentation.
<https://frankaemika.github.io/docs>
- Golz, J., & Biesenbach, R. (2015). Implementation of an autonomous chess playing industrial robot. *Proceedings of the 2015 16th International Conference on Research and Education in Mechatronics*, 53–56, Bochum, Germany. <https://doi.org/10.1109/REM.2015.7380373>
- Gunes, H., Broz, F., Crawford, C. S., Rosenthal-von der Pütten, A., Strait, M., & Riek, L. (2022). Reproducibility in human-robot interaction: Furthering the science of HRI. *Current Robotics Reports*, 3, 281–292. <https://doi.org/10.1007/s43154-022-00094-5>
- He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition. *Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition*, 770–778, Las Vegas, NV. <https://doi.org/10.1109/CVPR.2016.90>
- Kołosowski, P., Wolniakowski, A., & Miatliuk, K. (2020). Collaborative robot system for playing chess. *Proceedings of the 2020 International Conference Mechatronic Systems and Materials*, Bialystok, Poland. <https://doi.org/10.1109/MSM49833.2020.9202398>
- Koray, C., & Sümer, E. (2016). A computer vision system for chess game tracking. In L. Cehovin, R. Mandeljc, & V. Štruc (Eds.), *Proceedings of the 21st Computer Vision Winter Workshop*, Rimske Toplice, Slovenia.
- Kuffner, J. J., & LaValle, S. M. (2000). RRT-connect: An efficient approach to single-query path planning. *Proceedings 2000 ICRA. Millennium Conference. IEEE International Conference on Robotics and Automation. Symposia Proceedings*, 2, 995–1001, San Francisco, CA.
<https://doi.org/10.1109/ROBOT.2000.844730>
- Kunz, T., & Stilman, M. (2013). Time-optimal trajectory generation for path following with bounded acceleration and velocity. In N. Roy, P. Newman, & S. Srinivasa (Eds.), *Robotics: Science and Systems VIII* (pp. 209–216). MIT Press. <https://doi.org/10.7551/mitpress/9816.003.0032>
- Kuo, M. T., Hsueh, C. C., & Tsai, R. T. H. (2023). *Large language models on the chessboard: A study on ChatGPT's formal language comprehension and complex reasoning skills*. arXiv.
<https://doi.org/10.48550/arXiv.2308.15118>
- Larregay, G., Pinna, F., Avila, L., & Morán, D. (2018). Design and implementation of a computer vision system for an autonomous chess-playing robot. *Journal of Computer Science & Technology*, 18, e01.
<https://doi.org/10.24215/16666038.18.e01>
- Latif, E., Zhou, Y., Guo, S., Shi, L., Gao, Y., Nyaaba, M., Bewerdorff, A., Yang, X., & Zhai, X. (2024). *Can OpenAI o1 outperform humans in higher-order cognitive thinking?* arXiv.
<https://doi.org/10.48550/arXiv.2412.05753>
- LC, R., Benayoun, M., Lindborg, P., Xu, H., Chung Chan, H., Man Yip, K., & Zhang, T. (2021). Power chess: Robot-to-robot nonverbal emotional expression applied to competitive play. *Proceedings of the 10th International Conference on Digital and Interactive Arts*, Aveiro, Portugal.
<https://doi.org/10.1145/3483529.3483844>
- Leite, I., Pereira, A., Mascarenhas, S., Martinho, C., Prada, R., & Paiva, A. (2013). The influence of empathy in human–robot relations. *International Journal of Human-Computer Studies*, 71, 250–260.
<https://doi.org/10.1016/j.ijhcs.2012.09.005>
- Luqman, H. M., & Zaffar, M. (2016). Chess brain and autonomous chess playing robotic system. *Proceedings of the 2016 International Conference on Autonomous Robot Systems and Competitions*, 211–216, Bragança, Portugal. <https://doi.org/10.1109/ICARSC.2016.27>
- Mac, T. T., Dam, T.-N., Nguyen, V. T. T., Hiep, L. H., & Nguyen, T.-D. (2023). The development of an intelligent robot arm playing chess with human-robot interaction. *Proceedings of the 2023 1st International Conference on Health Science and Technology*, Hanoi, Vietnam.
<https://doi.org/10.1109/ICHST59286.2023.10565356>
- Makoviychuk, V., Wawrzyniak, L., Guo, Y., Lu, M., Storey, K., Macklin, M., Hoeller, D., Rudin, N., Allshire, A., Handa, A., & State, G. (2021). Isaac Gym: High performance GPU-based physics simulation for

- robot learning. *Proceedings of the Thirty-Fifth Annual Conference on Neural Information Processing Systems*, Virtual event. https://openreview.net/forum?id=fgFBtYgJQX_
- Mallasén Quintana, D., Del Barrio García, A. A., & Prieto Matías, M. (2020). *LiveChess2FEN: A framework for classifying chess pieces based on CNNs*. arXiv. <https://doi.org/10.48550/arXiv.2012.06858>
- Matuszek, C., Mayton, B., Aimi, R., Deisenroth, M. P., Bo, L., Chu, R., Kung, M., LeGrand, L., Smith, J. R., & Fox, D. (2011). Gambit: An autonomous chess-playing robotic system. *Proceedings of the 2011 IEEE International Conference on Robotics and Automation*, 4291–4297, Shanghai, China. <https://doi.org/10.1109/ICRA.2011.5980528>
- Moravec, H. (1988). *Mind children: The future of robot and human intelligence*. Cambridge, MA: Harvard University Press.
- MoveIt. (2021). IKFast Kinematics Solver. https://docs.ros.org/en/kinetic/api/moveit_tutorials/html/doc/ikfast/ikfast_tutorial.html
- NVIDIA. (2019). NVIDIA Jetson Nano. <https://www.nvidia.com/en-us/autonomous-machines/embedded-systems/jetson-nano>
- OpenAI. (2024). OpenAI o1 system card. arXiv. <https://doi.org/10.48550/arXiv.2412.16720>
- Open Robotics. (2020). ROS Noetic installation instructions. <https://wiki.ros.org/noetic/Installation>
- Peer, E., Rothschild, D., Gordon, A., Evernden, Z., & Damer, E. (2022). Data quality of platforms and panels for online behavioral research. *Behavior Research Methods*, 54, 1643–1662. <https://doi.org/10.3758/s13428-021-01694-3>
- Pereira, A., Martinho, C., Leite, I., & Paiva, A. (2008). iCat, the chess player: The influence of embodiment in the enjoyment of a game. *Proceedings of the 7th International Joint Conference on Autonomous Agents and Multiagent Systems - Volume 3*, 1253–1256, Estoril, Portugal. <https://dl.acm.org/doi/10.5555/1402821.1402844>
- Quigley, M., Conley, K., Gerkey, B., Faust, J., Foote, T., Leibs, J., Berger, E., Wheeler, R., & Ng, A. Y. (2009). ROS: An open-source Robot Operating System. *Proceedings of the ICRA Workshop on Open Source Software*, Kobe, Japan.
- Ranasinghe, P., Ranasinghe, P., & Ashan, V. (2023). ChessEye: An integrated framework for accurate and efficient chessboard reconstruction. *Proceedings of the 2023 Moratuwa Engineering Research Conference*, 177–182, Moratuwa, Sri Lanka. <https://doi.org/10.1109/MERCon60487.2023.10355515>
- Sajó, L., Ruttkay, Z., & Fazekas, A. (2011). Turk-2, a multi-modal chess player. *International Journal of Human-Computer Studies*, 69, 483–495. <https://doi.org/10.1016/j.ijhcs.2011.04.001>
- Schwenk, A., & Yuan, C. (2015). Visual perception and analysis as first steps toward human–robot chess playing. In G. Bebis, R. Boyle, B. Parvin, D. Koracin, I. Pavlidis, R. Feris, T. McGraw, M. Elendt, R. Kopper, E. Ragan, Z. Ye, & G. Weber (Eds.), *Advances in visual computing. 11th International Symposium, ISVC 2015, Las Vegas, NV, USA, December 14–16, 2015, Proceedings, Part II* (pp. 283–292). Cham: Springer. https://doi.org/10.1007/978-3-319-27863-6_26
- Shannon, C. E. (1950). XXII. Programming a computer for playing chess. *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science*, 41, 256–275. <https://doi.org/10.1080/14786445008521796>
- Shin, K., Yamsani, S., Mineyev, R., Chen, H., Gandhi, N., Lee, Y. J., & Kim, J. (2023). Exploring the capabilities of a general-purpose robotic arm in chess gameplay. *Proceedings of the 2023 IEEE-RAS 22nd International Conference on Humanoid Robots*, Austin, TX. <https://doi.org/10.1109/Humanoids57100.2023.10375209>
- Silver, D., Hubert, T., Schrittwieser, J., Antonoglou, I., Lai, M., Guez, A., Lanctot, M., Sifre, L., Kumaran, D., Graepel, T., Lillicrap, T., Simonyan, K., & Hassabis, D. (2018). A general reinforcement learning algorithm that masters chess, shogi, and Go through self-play. *Science*, 362, 1140–1144. <https://doi.org/10.1126/science.aar6404>
- Siraj, D. S. (2017). JudgeMaster: Automated chess movement validator. *Proceedings of the 2017 National Information Technology Conference*, 112–118, Colombo, Sri Lanka. <https://doi.org/10.1109/NITC.2017.8285662>
- Srivatsan, R., Badrinath, S., & Lakshmi Sutha, G. (2020). Autonomous chess-playing robotic arm using Raspberry Pi. *Proceedings of the 2020 International Conference on System, Computation, Automation and Networking*, Pondicherry, India. <https://doi.org/10.1109/ICSCAN49426.2020.9262351>
- Standage, T. (2002). *The Turk: The life and times of the famous eighteenth-century chess-playing machine*. New York: Walker.

- StereoLabs. (2020). Zed 2. <https://www.stereolabs.com/products/zed-2>
- Stockfish. (2022). Stockfish 15.1. <https://stockfishchess.org>
- Szegedy, C., Vanhoucke, V., Ioffe, S., Shlens, J., & Wojna, Z. (2016). Rethinking the inception architecture for computer vision. *Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition*, Las Vegas, NV. <https://doi.org/10.1109/CVPR.2016.308>
- Tabone, W., & De Winter, J. (2023). Using ChatGPT for human–computer interaction research: A primer. *Royal Society Open Science*, 10, Article 231053. <https://doi.org/10.1098/rsos.231053>
- Thanh Tra, N. L., Cong, P. T., & Anh, N. D. (2018). Design a chess movement algorithm and detect the movement by images classification using Support Vector Machine classifier. *Proceedings of the 2018 4th International Conference on Green Technology and Sustainable Development*, 335–340, Ho Chi Minh City, Vietnam. <https://doi.org/10.1109/GTSD.2018.8595604>
- Wölflein, G., & Arandjelović, O. (2021). Determining chess game state from an image. *Journal of Imaging*, 7, 94. <https://doi.org/10.3390/jimaging7060094>
- Xie, Y., Tang, G., & Hoff, W. (2018). Chess piece recognition using oriented chamfer matching with a comparison to CNN. *Proceedings of the 2018 IEEE Winter Conference on Applications of Computer Vision*, 2001–2009, Lake Tahoe, NV. <https://doi.org/10.1109/WACV.2018.00221>
- Zhelyabuzhsky, I. (2022). Stockfish. <https://pypi.org/project/stockfish>
- Zhong, T., Liu, Z., Pan, Y., Zhang, Y., Zhou, Y., Liang, S., Wu, Z., Lyu, Y., Shu, P., Yu, X., Cao, C., Jiang, H., Chen, H., Li, Y., Chen, J., Hu, H., Liu, Y., Zhao, H., Xu, S.,... Liu, T. (2024). *Evaluation of OpenAI o1: Opportunities and challenges of AGI*. arXiv. <https://doi.org/10.48550/arXiv.2409.18486>

Appendices

A. Comparison with existing chess robots

Table 4.A1. A comparison between the OpenChessRobot and other chess robots designed for human-robot interaction (HRI), evaluating aspects such as perception, manipulation, reproducibility, interactive features, and main purpose. The selected works explicitly mention the use of the robot in HRI studies or list “human-robot interaction” among their keywords.

Robot	Perception		Manipulation		Reproducibility		Interactive Features	Main Purpose
	Method	Evaluation	Method	Evaluation	Material	Code/dataset		
Pereira et al. (2008), iCat	Relying on E-board	Not applicable	Relying on E-board	Not applicable	iCat robot, E-board	Not available	Facial expressions	To study “the influence of embodiment in the enjoyment of a game” (p. 1253)
Sajó et al. (2011), Turk-2	Via an on-top RGB camera	Not reported	Via a robot arm	Not reported	A camera, a customized robot arm, and a standard chess set	Not available	Facial expressions, speech of fixed terms	To present a chess robot that can “communicate with the opponent” (p. 493)
Matuszek et al. (2011), Gambit	Via a RGB camera	Piece Recognition: 93.22%	Via a robot arm	Success rate: 91.6% (in three games)	A camera, a customized robot arm, and a normal chess set	Not available	Not available	To present a chess robot “that can play physical board games against human opponents” (p. 4291)
LC et al. (2021), Power chess	Memorizing the game based on previous moves	Not applicable	Via a robot arm	Not reported	Two robot arms	Not available	Gesture expressions	To study human perception of robot-to-robot gesture expressions in chess scenarios
Mac et al. (2023)	Via an on-top RGB camera	Not reported	Via a robot arm	Not reported	A camera, a Reactor X150 robot arm, and a standard chess set	Not available	Not available	To construct a physically capable Chess-Playing system to compete against the player
OpenChessRobot	Via a RGB camera	Piece Recognition: 96.52%–100%	Via a robot arm	Success rate with a pawn: Center: 100% Halfway: 75% Edge: 31%; more results in the paper	A camera, a Franka Panda robot arm, a customized gripper, a normal chess set	Open-source code, 3D-print models, and dataset available.	Gesture expressions and generative speech for game analysis	To represent an open-source reproducible chess robot that can play physical chess games with humans, interpret the game position verbally, and provide feedback with gestures

B. Dataset creation

A new dataset was collected by placing all the pieces (63 pieces in total, one rook missing) on the board, and then taking photos from the default camera pose. The corresponding labels in FEN format were then created. After performing row movements, column movements, and rotations, each piece appeared at least once at each square on the board, with rotations. The procedure for creating the dataset is detailed below. The dataset and fine-tuned models can be found at: <https://doi.org/10.4121/1cb5bf64-468e-462a-a82e-c847d88a7a86>

Starting position

The pieces were manually placed in the middle of the square. The camera pose was such that pieces often occluded each other. Each piece was placed so that its adjacent neighbors included both white and black pieces.

Taking photos of the default and rotated piece positions

At a game position, multiple photos were taken: one photo showed the default setting, and four more photos showed the pieces rotated. Four types of pieces were rotated because of their rotational asymmetry: white king, white knight, black king, and black knight. Kings were rotated twice, and knights were rotated four times to create different shapes of the pieces in the image. Each rotation was approximately 45 degrees.

Following the above procedure, at each chess piece's position, five images were taken (unrotated, both the knight and king rotated twice, knight rotated twice more). Photos are named as: `img_CX_RY_[r0, r1, r2, r3, r4].png`

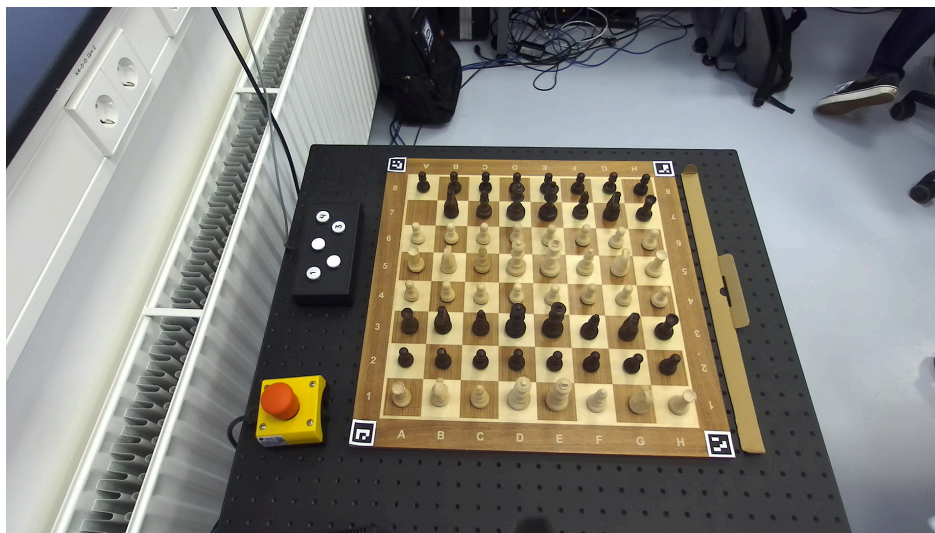


Figure 4.B1. Starting position. Image in dataset: `chess_image_data/raw/img/img_C0_R0_r0.png`. Label in FEN: `pppppppp/1nbqkbnr/PPPPPPPP/RNBQKBNR/PPPPPPPP/rnbqkbnr/pppppppp/RNBQKBNR`

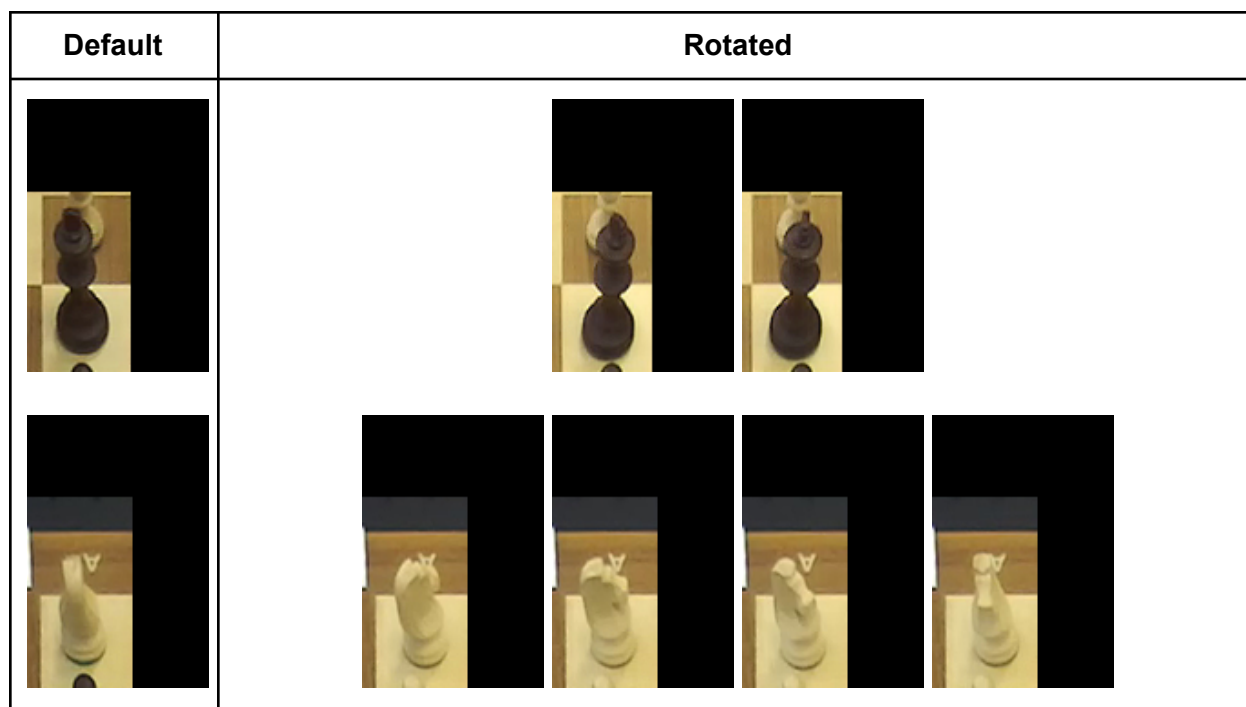


Figure 4.B2. Illustration of piece rotation.

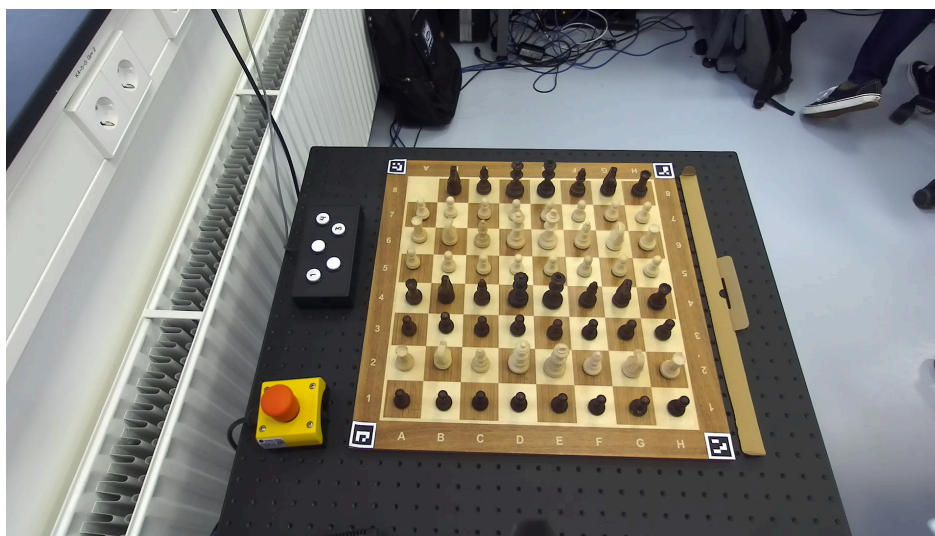


Figure 4.B3. Starting position, after one upward row shift. Image in dataset:

chess_image_data/raw/img/img_C0_R1_r0.png. Label in FEN:

1nbqkbnr/PPPPPPPP/RNBQKBNR/PPPPPPPP/rnbqkbnr/pppppppp/RNBQKBNR/pppppppp

Translating the pieces by row

In the direction of increasing rank, the rows of pieces were moved to make a new chess piece position. Then five photos were taken at the new position with the same method mentioned above. The row translation was repeated three times ($R_0 > R_1 > R_2 > R_3$).

Translating the pieces by column

After translating the row three times, one additional row translation was realized ($R3 > R0$), in order to restore the $R0$ game position. After that, a column translation in the ($A > H$) direction was performed ($C0 > C1$), followed by taking photos at this new game position using the same method mentioned above.

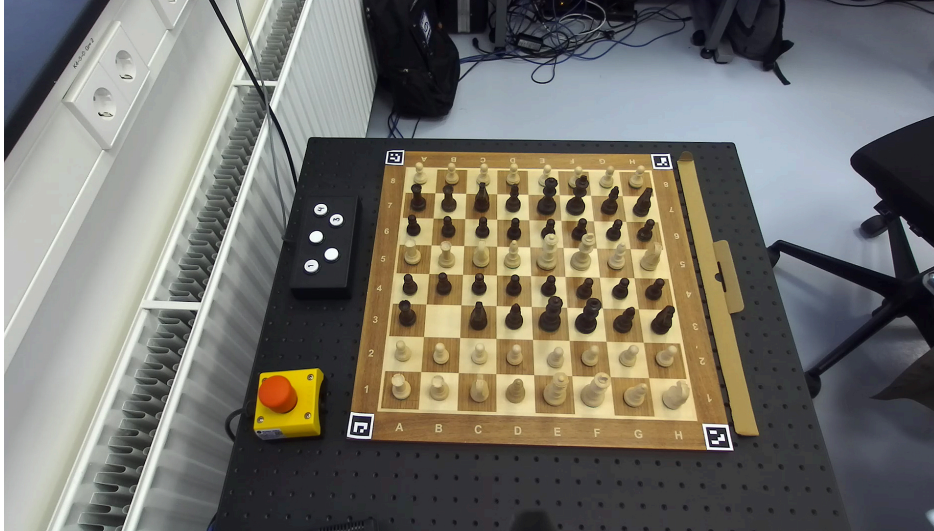


Figure 4.B4. Starting position, after one rightward column translation. Image in dataset: chess_image_data/raw/img/img_C1_R0_r0.png. Label in FEN: PPPPPPPP/rrnbqkbn/pppppppp/RRNBQKBN/pppppppp/r1nbqkbn/PPPPPPPP/RRNBQKBN

Creating the datasets

A total of 160 raw images were collected ($4 \text{ row movements} \times 8 \text{ column movements} \times 5 \text{ orientations}$), from which three datasets were created: default, rotated, and visually shifted.

The default dataset (**D**) contained 2048 sub-images of unrotated pieces extracted from the raw images, with every unique piece-square combination represented at least once.

The rotated dataset (**R**) contained 1280 sub-images of each rotated piece (black king, white king, black knight, and white knight) extracted from the raw images, with every unique piece-square-rotation combination represented at least once.

The visually shifted dataset (**S**) contained 8192 sub-images with cropping windows shifted in four directions (up, down, left, right). The displacement in each direction was $\frac{1}{4}$ of a square's length. For each shift direction, 2048 sub-images were extracted from the raw images without piece rotations, following the same procedure used to generate the dataset **D** but with displaced cropping windows. The visually shifted dataset is used only for testing, not for training.

Training data selection and train-val-test split

Several squares were selected as key squares for the board. Sub-images of pieces on these squares were used to train the model; this model was then tested on the remaining squares to show its generalization ability. Two sets of key squares were tested:

1. 3×3:
key_file: A, E, H;
key_rank: 1, 4, 8;
training set: 9 × 12 sub-images;
2. 4×4:
key_file: A, E, D, H;
key_rank: 1, 4, 5, 8;
training set: 16 × 12 sub-images;

The training dataset (d) from the default dataset was generated using all sub-images from these key squares. One sub-image was chosen randomly for each square-piece pair.

The training dataset (r) was generated from the rotation dataset using all sub-images rotated by 90 degrees. One sub-image was chosen randomly for each square-piece pair with the same rotation angle.

Another training dataset (rr) was generated from the rotation dataset using all sub-images rotated by both 45 degrees and 90 degrees. For black and white knights, one sub-image was chosen randomly for each square per rotation angle.

Then, sub-images of the remaining squares were split into a validation set and a test set using a ratio of 20:80. The test sets corresponding to datasets d, r, rr, respectively, are referred to as d, r, rr.

Before feeding the training set to train the model, some random image augmentation was applied to each sub-image. These parameters were set through trial and error.

Table 4.B1. Parameters for augmentation.

Attributes	Range
brightness/hue/saturation	[-20%, 20%]
contrast	[-30%, 30%]
scale	[0.8, 1.2]
translation	[-0.03, 0.1]

Examples:



C. Links to videos of the OpenChessRobot

Video 1: Demonstrations of the robot's movements and game explanation. Human-robot gameplay with body posture feedback is also included. Link:

<https://www.youtube.com/watch?v=RenXuiwX4Go>

Video 2: A sample from an annotated video recording using the Tobii Glasses 3 eye-tracker from the perspective of a professional chess player. To protect the privacy of the participant, audio has been removed. Link: <https://www.youtube.com/watch?v=j-b32ILQjtw>

Video 3: The three videos used in the online questionnaire survey. Link:

https://www.youtube.com/shorts/Mj_rZVboH9s

Transcripts of video 3:

The Robotics Education scenario: I determine the exact 3D position of the board first by detecting the four ArUco markers placed on it and calculate the chessboard's 3D grid. Then once I identify which square each piece is on through my occupancy and piece classification, I map that square to its corresponding 3D coordinates on the board.

The Chess Coach scenario: Great, I am playing e4 which is one of the most popular and aggressive openings for White, controlling the center and opening lines for my queen and bishop. Now if Black plays c5 they are introducing the Sicilian Defense. Instead of directly placing a pawn in the center (like with e5), Black plays c5 to exert influence on the d4 square.

The Home Entertainment scenario: This is the famous position from 2001: A Space Odyssey, where Frank Poole faced against the HAL 9000. In the movie Poole as White captured on a6 with the queen and walked right into a forced mate. But if we analyze carefully White might try something safer—like shifting the queen to b7 instead of immediately grabbing the pawn. Or pushing a rook to e1. What would you like me to try?

D. A case study with an expert chess player

On May 25, 2023, during development, we invited a FIDE Master to play against the initial version of the OpenChessRobot. His feedback was used to improve the interactive features, leading to the current version of the OpenChessRobot.

Two games were played between the human and the robot. In the first game, the player played against a Stockfish chess engine. In the second game, the player experienced the verbal feedback on the game positions, while the robot used the chess engine Maia ELO 1900 (McIlroy-Young et al., 2020), which was trained on human games. The player wore Tobii Pro Glasses 3 during the games to record gaze activity and was instructed to think aloud (a video can be found in Appendix C). An experimenter prompted the player to explain strategies and thoughts. After the games, an interview was conducted to gather feedback on the player's experience with the robot, potential use cases, and suggestions for improvements.

The recordings were processed using AssemblyAI's Speech-to-Text API for speech recognition and speaker diarization. OpenAI's ChatGPT o1 model was used to translate the transcriptions from Dutch to English and to answer specific questions (see Table 4.D1). Eventually, only the recording of the interview was analyzed, as most comments during the games were about the game strategy.

Table 4.D1. Questions to ChatGPT o1 and corresponding responses.

Question 1: What is the playing experience of the chess player playing against the chess-playing robot? Provide concise answers in bullet points.

Answer from ChatGPT o1:

- The chess player did not find the robot intimidating and felt it resembled playing a regular human opponent primarily due to its strong moves.
- The novelty of a physical, tangible opponent added a different dynamic compared to playing chess online.
- The player expressed that faster, more natural robot movements would make the experience more enjoyable.
- The absence of meaningful feedback (e.g., identifying mistakes or strong moves) reduced the robot's value as a training tool.
- The player would find the robot more beneficial if it played and provided insights in a more human-like and instructive manner.

Question 2: What does the chess player think about the possible use cases and the unlikely use cases of this chess robot? Provide concise answers in bullet points.

Answer from ChatGPT o1:

- Likely Use Cases:
 - o Practicing specific chess openings or positions.
 - o Receiving instructive feedback on moves and tactics.
-

- o Serving as a training tool for beginners.
 - o Providing an entertaining, physical alternative to playing online.
 - Unlikely Use Cases:
 - o Playing random casual games where no human-like or instructive element is added.
 - o Replacing standard online engines for straightforward chess play.
 - o Serious professional training relying solely on a “perfect” engine (like Stockfish).
-

Question 3: What are the suggestions from the chess player to improve the robot? Provide concise answers in bullet points.

Answer from ChatGPT o1:

- Offer immediate feedback on moves, including identifying blunders or strong plays.
 - Employ a more human-like engine (e.g., the Maia engine) to create a realistic playing style.
 - Increase the speed and fluidity of the robot’s physical movements.
 - Provide instructive commentary that explains why certain moves are good or bad.
 - Allow scenario-based training, such as practicing specific openings or positions.
-

From the interview, we learned that the player did not find the robot intimidating and recognized its potential use for beginners in practice or entertainment scenarios. However, he noted that the robot could be improved to play more like a human, both in terms of game strategy and motion speed. To improve the robot, he suggested it should provide direct feedback on move quality and move faster.

The eye-tracking data from the two games was processed into annotated videos, showing where the player was looking during the game. The gaze data provides insights into how the player anticipated the robot’s next move and how often the player engaged in gaze interaction with the robot’s physical body (see Figure 4.D1). These gaze recordings offer valuable data for future HRI studies on this platform. A video clip of the eye-tracking data is included in Appendix C (Video 2).

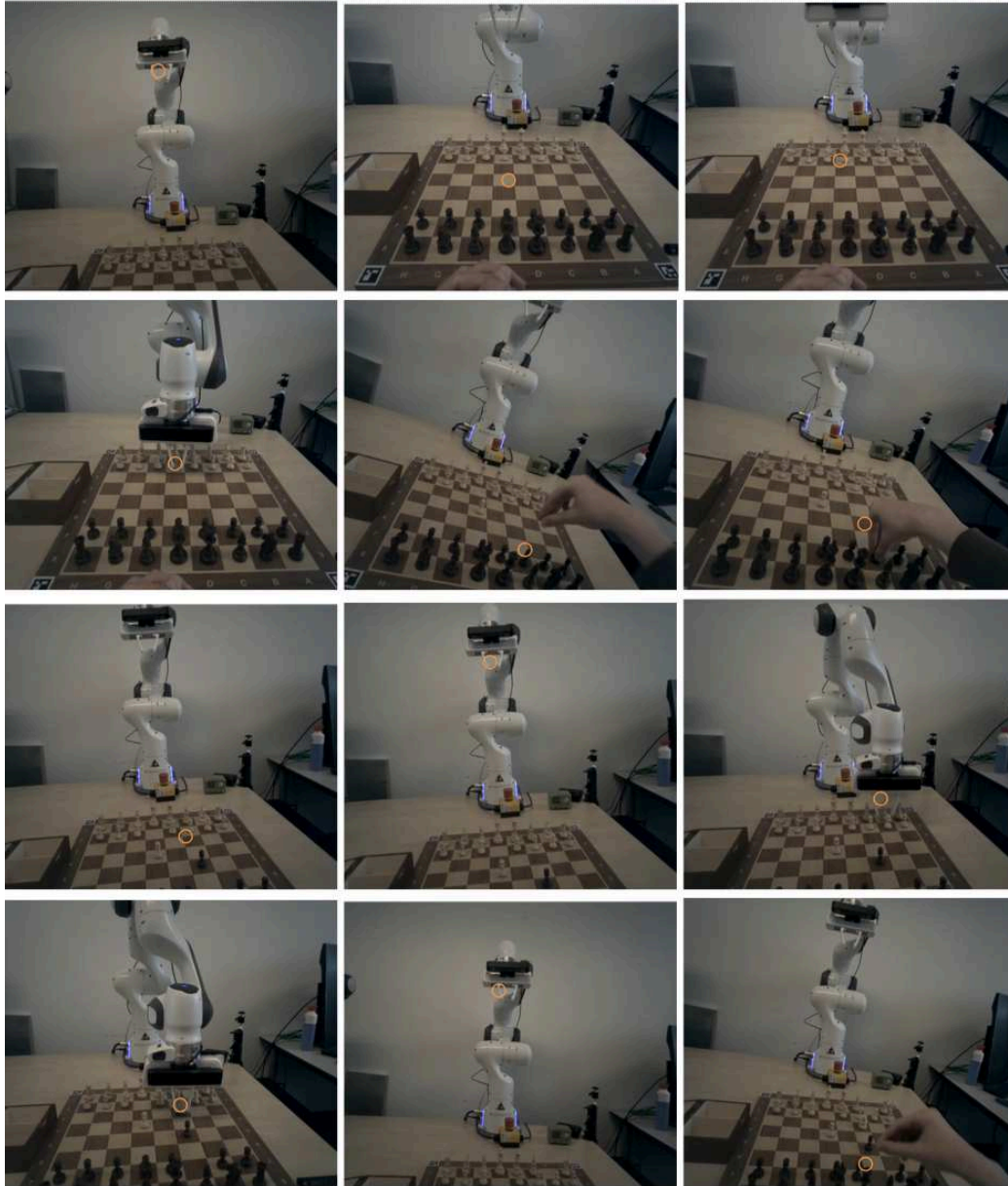


Figure 4.D1. Annotated video with gaze overlay. Row 1 (left to right): The player looked at the robot and scanned the board for the robot's possible moves. Row 2 (left to right): The robot made its first move, followed by the player's first move. Row 3 (left to right): The player predicted the robot's next move and looked at the robot, waiting for its execution. Row 4 (left to right): The robot executed its move, the player gazed at the robot again, then made the second move.

Reference

McIlroy-Young, R., Sen, S., Kleinberg, J., & Anderson, A. (2020). Aligning superhuman AI with human behavior. *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, 1677–1687. <https://doi.org/10.1145/3394486.3403219>

E. The full questionnaire

Category	Question
GPT familiarity	QA. <i>“Have you ever heard of ChatGPT?”</i> (Yes, No, I prefer not to respond) QB. <i>“How often do you use ChatGPT?”</i> (Every day, 4-6 days a week, 1-3 days a week, About once every 2 weeks, About once a month, Less than once a month, Never, I prefer not to respond) QC. <i>“Do you use a paid version of ChatGPT (Plus or Pro)?”</i> (Yes, No, I prefer not to respond)
Video Demo	
Perception of the robot	Q1. (A/B/C) <i>“Do you think this robot could be used <in schools to explain to students how a robot works?> / <in schools to teach students chess?> / <at home to provide entertainment?>”</i> (Definitely not, Probably not, Neutral, Probably Yes, Definitely yes, I prefer not to respond) Q2. <i>“Comment on the technical quality of the robot as shown in the video. Mention positive or negative aspects.”</i> (A response of at least 20 characters was required) Q3. <i>“Do you have any recommendations about how to improve the robot?”</i> (A response of at least 20 characters was required) Q4. (A/B/C) <i>“Do you think that within 10 years, robots like this could serve as a replacement of existing screen-based <computers to help understand technical topics such as computer vision?> / <computers to improve chess skills?> / <forms of entertainment such as television?> Please elaborate on why or why not.”</i> (A response of at least 20 characters was required) Q5. <i>“Would you welcome a future where robots that can talk, equipped with artificial intelligence (such as improved versions of ChatGPT), replace human workers?”</i> (Definitely not, Probably not, Neutral, Probably Yes, Definitely yes, I prefer not to respond)
Demographics	QD. Age, QE. Gender, QF. Education
Familiarity with the context of scenarios	QG. <i>“Are you a technology enthusiast?”</i> (Not at all, Slightly, Moderately, Very, Extremely, I prefer not to respond) QH. <i>“Are you familiar with the movie 2001: A Space Odyssey?”</i> (Not at all, Slightly, Moderately, Very, Extremely, I prefer not to respond) QI. <i>“Did you ever play in an official FIDE-rated chess tournament?”</i> (No, Yes, I prefer not to respond) QJ. <i>“Are you a chess enthusiast?”</i> (Not at all, Slightly, Moderately, Very, Extremely, I prefer not to respond)

F. Prompts for ChatGPT o1 and the raw outputs

Prompt to ChatGPT o1 to analyze Q2	Based on the given answers to the question: “Comment on the technical quality of the robot as shown in the video. Mention positive or negative aspects.”, summarize the most mentioned three positive and negative aspects in the answers. Make concise bullet points. Answers:
Output of Scenario A	Most Mentioned Positive Aspects • Smooth and Precise Movement: Many noted the arm moves fluidly, handling chess pieces accurately. • Clear Voice/Explanation: Several answers highlighted that the robot’s speech is generally understandable and coherent. • Advanced Technical Ability: Respondents frequently praised its strong board-detection and analytical capabilities. Most Mentioned Negative Aspects • Overly Technical/Complex Language: A common concern was that its explanations could be too advanced or confusing for some audiences. • Slow or Lengthy Process: Multiple people felt the scanning or movement took too long. • Robotic or Distracting Voice/Noise: The monotone sound and mechanical noise were cited as off-putting or not engaging.
Output of Scenario B	Most Mentioned Positive Aspects • Smooth and precise movement: The robot accurately picks up and places pieces without disturbing others, and its motion control is generally fluid and stable. • Clear explanations: Many appreciated that it articulates moves clearly, providing reasoning and context for each chess move. • Effective teaching approach: Explaining why certain moves are played and discussing strategy was seen as a strong educational feature. Most Mentioned Negative Aspects • Robotic or unnatural voice: Several found the voice to be lacking a human touch or warmth, making it less engaging. • Noisy or distracting movements: The mechanical sounds during operation were frequently cited as disruptive. • Limited interactivity: Users noted a lack of two-way communication or the ability to ask questions, reducing the personalized learning experience.
Output of Scenario C	Most Mentioned Positive Aspects • Smooth and precise movement (many note its fluid, accurate

	handling of chess pieces) • Intelligent or knowledgeable (demonstrates strong chess skills and offers insightful commentary) • Entertaining or interactive (engages the user through narration, voice, and conversation) Most mentioned negative aspects • Loud or noisy (the whirring and mechanical sounds are off-putting to some) • Bulky or large (the size and industrial look make it less home-friendly) • Slow or boring (some feel its pace and monotone delivery can be dull over time)
Prompt to ChatGPT o1 to analyze Q3	Based on the given answers to the question: “Do you have any recommendations about how to improve the robot?”, summarize the most mentioned five recommendations in the answers. Make concise bullet points. Answers:
Output of Scenario A	• More natural, varied, or human-like voice (e.g., less robotic, different voice options, more expressive tone) • Simpler, age-appropriate explanations (e.g., less technical jargon, clearer language for children) • Friendlier or more appealing design (e.g., adding a face or screen, making it look less mechanical and more welcoming) • Reduced mechanical noise (e.g., quieter motors, less distracting sounds during movement) • Faster or smoother operation (e.g., increased speed, more fluid arm movements)
Output of Scenario B	• More natural, simpler speech: Many respondents suggested using a less robotic voice and avoiding overly technical language, especially for beginners or children. • Reduce mechanical noise: A common recommendation was to make the robot’s arm and movements quieter so it’s less distracting. • Friendlier design and smaller size: Several mentioned the robot’s industrial look, proposing a more approachable or compact form—e.g., adding a face, giving it a name, or shrinking its footprint. • Greater interactivity and Q&A: Multiple answers emphasized an option for students to ask questions and receive explanations in real time.

	• Clearer movement cues: Suggestions included using a laser pointer or other visual indicators to highlight the squares being discussed.
Output of Scenario C	• Reduce mechanical noise: Many participants found the whirring and buzzing of the robot's arm distracting and suggested quieter motors or motion. • Use a more natural, expressive voice: A large number of people felt the robot's monotone voice should be improved, either by adding emotion, varying intonation, or offering different accents. • Make the design more compact: Several responses pointed out that the robot is bulky or tall, recommending a smaller, sleeker form factor. • Improve overall aesthetics/approachability: Suggestions ranged from adding a "face" or a friendlier exterior to changing color schemes and reducing exposed wires. • Enhance movement fluidity: Smoother, faster, or less "jerky" movement was mentioned to make the robot feel more natural and engaging.
Prompt to ChatGPT o1 to analyze Q4 – Scenario A	Based on the given answers to the question: "Do you think that within 10 years, robots like this could serve as a replacement of existing screen-based computers to help understand technical topics such as computer vision? Please elaborate on why or why not.", summarize the most mentioned three concerns in the answers. Make concise bullet points (keywords and explanation in one sentence). Answers:
Output of Scenario A	• High cost and limited cost-effectiveness: Many responses cite that these robots would be too expensive or impractical to deploy at scale compared to existing screen-based tools. • Concerns about acceptance and human interaction: Several people doubt whether users would find a robot's impersonal or purely mechanical explanations as engaging as human teachers or more familiar screen-based interfaces. • Incomplete or restricted functionality: Respondents worry that a robot arm cannot cover the breadth of content or flexibility offered by computers, and may not fully replace the versatility (e.g., code viewing, detailed visuals) screens currently provide.
Prompt to ChatGPT o1 to analyze Q4 – Scenario B	Based on the given answers to the question: "Do you think that within 10 years, robots like this could serve as a replacement of existing screen-based computers to improve chess skills? Please elaborate on why or why not.", summarize the most mentioned three concerns in the answers. Make concise bullet points (keywords and explanation in one

	sentence). Answers:
Output of Scenario B	• High cost: Many respondents believe the robots would be too expensive for widespread adoption. • Limited portability: Several people noted that these physical robots are bulkier and less convenient than screen-based computers. • Reduced versatility: Others highlighted that a single-purpose chess robot cannot match the broader functionality and advanced analysis tools offered by standard computers.
Prompt to ChatGPT o1 to analyze Q4 – Scenario C	Based on the given answers to the question: “Do you think that within 10 years, robots like this could serve as a replacement of existing screen-based forms of entertainment such as television? Please elaborate on why or why not.”, summarize the most mentioned three concerns in the answers. Make concise bullet points (keywords and explanation in one sentence). Answers:
Output of Scenario C	• Lack of variety and visual/storytelling depth – Many respondents feel a robot cannot replicate the diverse content and creative appeal (shows, movies, storylines) that TV provides. • Preference for passive entertainment – A common concern is that people want to “sit and watch,” whereas a robot requires more active engagement and isn’t as effortless as simply turning on a screen. • Insufficient human/emotional element – Several mention that robots lack the emotional connection or relatable human factor that makes television and film compelling.
Prompt to ChatGPT o1 and Gemini 2.0 Flash Think to classify answers of Q4 – Scenario A	Based on the given answers to the question: “Do you think that within 10 years, robots like this could serve as a replacement of existing screen-based computers to help understand technical topics such as computer vision? Please elaborate on why or why not”, classify each answer to one of the three classes: [positive, neutral, negative] and add the category to each row. Add quotation marks if any answer is not in quotation marks. The answers are provided as a csv.
Prompt to ChatGPT o1 and Gemini 2.0 Flash Think to classify answers of Q4 – Scenario B	Based on the given answers to the question: “Do you think that within 10 years, robots like this could serve as a replacement of existing screen-based computers to improve chess skills? Please elaborate on why or why not”, classify each answer to one of the three classes: [positive, neutral, negative] and add the category to each row. Add quotation marks if any answer is not in quotation marks. The answers are provided as a csv.
Prompt to ChatGPT	Based on the given answers to the question: “Do you think that within

o1 and Gemini 2.0 Flash Think to classify answers of Q4 – Scenario C	10 years, robots like this could serve as a replacement of existing screen-based forms of entertainment such as television? Please elaborate on why or why not”, classify each answer to one of the three classes: [positive, neutral, negative] and add the category to each row. Add quotation marks if any answer is not in quotation marks. The answers are provided as a csv.
---	--

5

Educating future engineers about LLMs: A scalable workshop

Intermezzo

The cumulative picture of alignment and misalignment spans perception (Chapter 2), the full processing chain in voice and gesture teleoperation (Chapter 3), and concurrent perception and response in the chess robot with LLM-generated commentary (Chapter 4), where fluent output masked shallow reasoning. What remains is the cognition stage on the human side: whether people can learn to recognize such limitations and compensate. Chapter 5 tests this in a gamified workshop in which engineering students redistribute reasoning between themselves and LLMs of differing capability (Research question 4).

Abstract

As large language models (LLMs) are increasingly integrated into engineering workflows, students require hands-on experience to learn how to collaborate with them critically. This paper presents a scalable gamified workshop designed for engineering Master's students to practice human-AI collaboration in navigation planning. Using a mobile web interface, across 10 workshop sessions, a total of 226 students wrote prompts for a non-reasoning and a reasoning LLM to solve grid-based navigation tasks of increasing complexity. The system returned robot-executable plans, trajectory visualizations, and automated scoring, culminating in a live demonstration on a Boston Dynamics Spot robot. In a post-task questionnaire, 81.5% reported substantial learning and 91.0% reported high engagement. Analysis of the submitted prompts revealed that students changed their strategies from step-by-step instructions for the non-reasoning LLM toward providing higher-level guidance for complex problem-solving tasks. We conclude that such interactive simulation-to-reality environments are viable for teaching the verification and collaboration skills necessary for responsible LLM use in engineering. Code is available at: <https://github.com/renchizhyyy/LLM-robotics-workshop>

5.1 Introduction

Large language models (LLMs) are rapidly becoming part of how engineers write code, explore design alternatives, and communicate technical ideas. These developments create a dilemma for education. LLMs can be misused as a shortcut for completing assignments (Cotton et al., 2024), yet they also enable new forms of interactive learning, especially when students learn to treat the model as a collaborator whose output must be interpreted, tested, and verified (Kasneci et al., 2023; Lo, 2023). For engineering programs, the key question is no longer whether students will use LLMs, but how we can teach them to use these systems critically and responsibly.

Robotics provides a particularly concrete setting for this challenge because LLM outputs can be operationalized: a model's response can be turned into executable actions, and failures become immediately visible. Recent work has shown that LLMs can support robotics workflows in multiple ways, for example by mapping natural language to robot skills (Ahn et al., 2023; Black et al., 2025; Shridhar et al., 2022), generating code-based policies (Arenas et al., 2024; Liang et al., 2023), or producing symbolic plans that downstream systems can follow (Chen et al., 2024; Huang et al., 2022; Silver et al., 2024; Zhou et al., 2024). However, LLMs remain probabilistic and can produce plausible-looking outputs that violate constraints or fail in subtle ways (Kumar & Cohen, 2026). In safety-relevant domains such as robotics, this makes verification a core competence: users need to learn to anticipate failure modes, prompt the model appropriately, and validate the resulting plan before execution.

A practical complication is that 'LLM capability' is not monolithic. In robotics planning, non-reasoning models typically behave like pattern matchers: they can follow highly structured instructions but may fail when the problem requires multi-step constraint satisfaction (Kambhampati et al., 2024; Valmeekam et al., 2023). Newer reasoning LLMs can handle more

complex planning-like tasks by performing internal reasoning, yet their performance still degrades as the search space grows (Shojaee et al., 2025; Zhang et al., 2026). Importantly, this difference is not only about task success, but also about how cognition is distributed between human and model. With non-reasoning models, a larger share of the “thinking” may have to be externalized by the user: successful prompting may require providing domain knowledge up front (e.g., decomposing into subgoals, or giving explicit waypoints). Reasoning models can be expected to complete simple tasks with less human guidance, because they can carry the inferential load internally. Yet as tasks become more complex (e.g., longer horizons, or greater combinatorial search), effective use depends on scaffolding, analogous to the role of Chain-of-Thought or least-to-most prompting in non-reasoning models (Wei et al., 2022; Zhou et al., 2023), where the human prompter has to perform part of the cognition by adding hints or heuristics that reduce the LLM’s search space. This creates a new educational opportunity: students can learn when concise task descriptions suffice and when additional scaffolding is required.

However, providing hands-on, equitable practice at scale is difficult. Individual access to paid LLM tools, heterogeneous devices, institutional restrictions, and the logistics of running robotics hardware can prevent instructors from giving every student a comparable experience. What is needed are scalable learning activities, allowing entire classes to participate without needing a robot for every student, that (1) make model limitations observable, (2) teach prompt refinement across tasks and verification strategies, including how to distribute cognition effectively between humans and models, and (3) engage large numbers of students. To meet these requirements, we adopted principles from active learning (Bonwell & Eison, 1991; Freeman et al., 2014; Prince, 2004) and gamification (Morales-Trujillo & García-Mireles, 2020; Sailer & Homner, 2020) by designing an activity in which students tested and compared prompts while progress was reinforced through goals, scoring, and performance feedback.

This paper presents a scalable, gamified workshop designed for Master’s-level robotics students to practice human-LLM collaboration in navigation planning. Students used their own smartphones to submit prompts through a lightweight web interface to two LLM planners: a non-reasoning model and a reasoning model. Their goal was to generate a robot-executable plan for grid-based navigation tasks of increasing difficulty. Concretely, students prompted the model to produce stepwise movement commands that would move a mobile robot through a grid while respecting constraints such as obstacle avoidance, visiting named checkpoints, and (in harder tasks) following a specified checkpoint order and avoiding revisiting cells. The interface returned an executable plan, a trajectory visualization, and an automated score based on rule violations. This setup made the differences in required scaffolding visible. For simpler tasks, a concise prompt often sufficed with the reasoning model; the non-reasoning model needed more explicit input; and as complexity increased, even the reasoning model benefited from human hints. Instructors used these outputs to facilitate discussion: students predicted whether prompts would succeed, compared high- and low-scoring prompts, and reflected on which kinds of scaffolding helped each model type. The workshop concluded with a live demonstration on a Boston Dynamics Spot robot to connect the simulation-based planning exercise to real robot execution and to surface practical issues.

This paper addresses two questions. First, we evaluated whether this workshop format is an effective and scalable way to teach students how to work with LLMs for robotics-relevant planning, specifically whether it yields high engagement and high self-reported learning about model differences, verification, and collaboration strategies. Second, we analyzed what students actually did when prompted with increasing task difficulty: how their prompting strategies changed as they learned to externalize and redistribute cognition. To do so, we analyzed the prompts students submitted during the workshop and responses to an anonymous post-task questionnaire measuring engagement and perceived learning, which we expected to be high given the interactive, feedback-rich format.

In summary, we present a practical workshop design for teaching critical LLM use in engineering, report empirical data on engagement and learning, and analyze how students changed their prompting as tasks became more complex.

5.2 Methods

Participants

A total of 226 students participated in the workshop as part of an MSc course on Human-Robot Interaction at the Faculty of Mechanical Engineering at TU Delft. The students were enrolled in either the MSc Robotics program or the Biomechanical Design track within the MSc Mechanical Engineering program. Participants signed up for the workshop in 10 time slots. Each workshop had a maximum of 30 participants. The number of participants per workshop ranged from 9 to 29. The study was approved by the Human Research Ethics Committee of TU Delft (approval number 6189).

Hardware and software

We set up an area for real robot navigation, presentation, and interaction (see Figure 5.1). Two TVs were used: one displayed the presentation slides and the other was connected to an instructor's laptop to show interfaces for interaction with students. A 7×13 virtual grid and a corresponding physical grid were set up for the navigation tasks. This grid contained 18 obstacle cells and three or four checkpoint cells. The navigation task was simplified as movements from cell to cell, represented in integer coordinates (row, column).

The instructor's laptop (Dell Latitude 7320) ran a custom-built graphical user interface (GUI). The GUI was developed using Python's *Tkinter* GUI toolkit to visualize the simulated environment, run the LLM planners, display students' prompts, visualize planned trajectories, control the Spot robot in the demonstration, and show the position of the Spot robot in real-time (Figure 5.2, bottom). An instructor operated the GUI to process all user prompts collected from the students for each task in parallel and to show students' inputs and the LLM planner's output (Figure 5.2, top left).

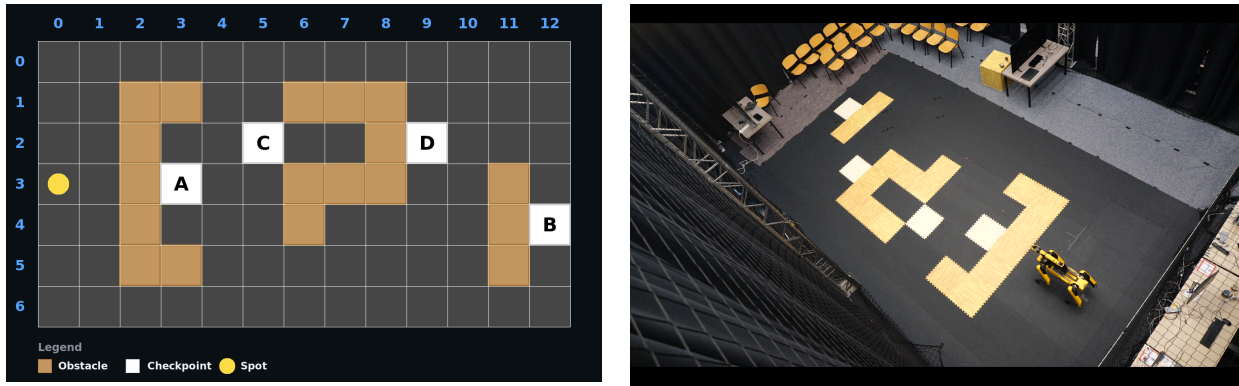


Figure 5.1. Left: Grid area for simulated robot navigation. Right: Corresponding grid area for real robot navigation.

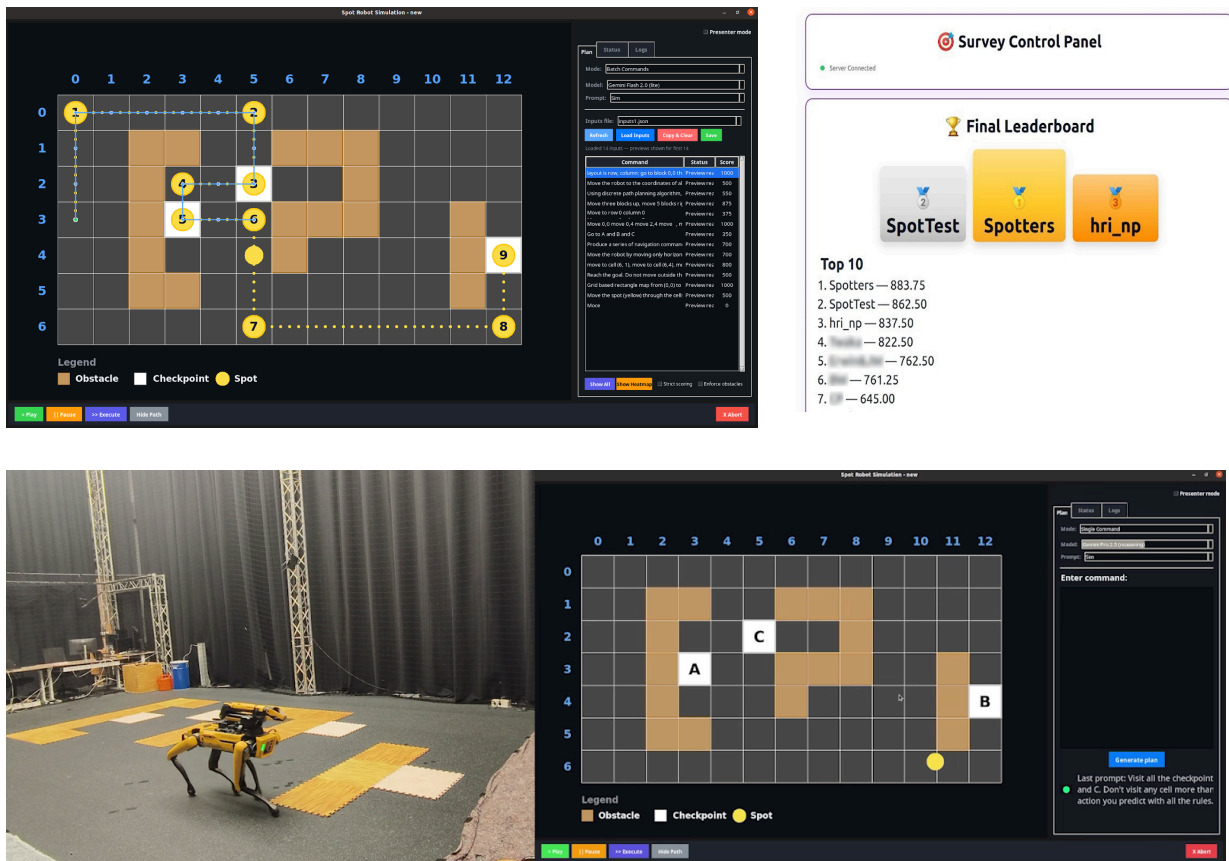


Figure 5.2. Top left: The simulation GUI displaying a trajectory generated by the LLM planner. Top right: The host web interface showing a leaderboard of mean student teams' scores. Bottom: The physical experimental setup, where the GUI (visible on the TV screen) provides real-time visualization and control of the robot during the demonstration.

User prompts were collected through a custom web interface hosted on the instructor's laptop. The host web interface tracked the progress, collected prompts, and displayed the final leaderboard (Figure 5.2). The client web interface on students' devices collected the students'

prompts and showed the output plans from the LLM, together with an evaluation and visualization (Figure 5.3). Students accessed the client interface via a local wireless network provided by a TP-Link AX3000 Gigabit Wi-Fi 6 Router. Using the host interface, the instructor determined which client interface participants saw at each stage of the workshop.

The physical robot, Spot Explorer (Boston Dynamics, 2020), was controlled from the same laptop using the Python interface of Spot-SDK 5.0.1 (Boston Dynamics, 2025). The Noetic distribution of the Robot Operating System (Quigley et al., 2009) was used to bridge the GUI and the Spot SDK, enabling execution of the planned movements on the physical robot. Each LLM-planned move in the format “*move_to_cell row=<int> col=<int>*” was translated from the coordinates of a cell in the grid to a real-world target location in the world frame.

A system prompt was combined with each user prompt (i.e., the prompt that students created during the workshop) to formulate a request to the LLM API server. The system prompt consisted of (1) the LLM’s role: the LLM should act as a path planner on a 2D grid; (2) constraints: the planner should follow rules such as not going through obstacles and only moving horizontally or vertically on the grid; (3) the required output format: the planner had to output a robot-executable plan in the format “*move_to_cell row=<int> col=<int>*” per line; (4) a world description: a description of the grid dimensions, the robot’s start coordinate, the checkpoints with names and coordinates, the obstacle coordinates, and the orientation of the grid; and (5) a guardrail: for impossible or non-relevant requests, the planner would not make a plan and respond “*FAIL <REASON>*” (see the full system prompt in Appendix A).

Two LLMs were chosen as planners: a non-reasoning model, Gemini 2.0 Flash-Lite (Google, 2025a), and a reasoning model, Gemini 2.5 Pro (Google, 2025b). All API requests were sent with a temperature of 0.0. For requests with Gemini 2.5 Pro (Google, 2025b), a ‘thinking budget’ of 3,000 tokens was set to ensure an acceptable inference time during the workshop. This acted as a soft upper limit, meaning actual token usage varied based on the specific task (Google, 2025d).

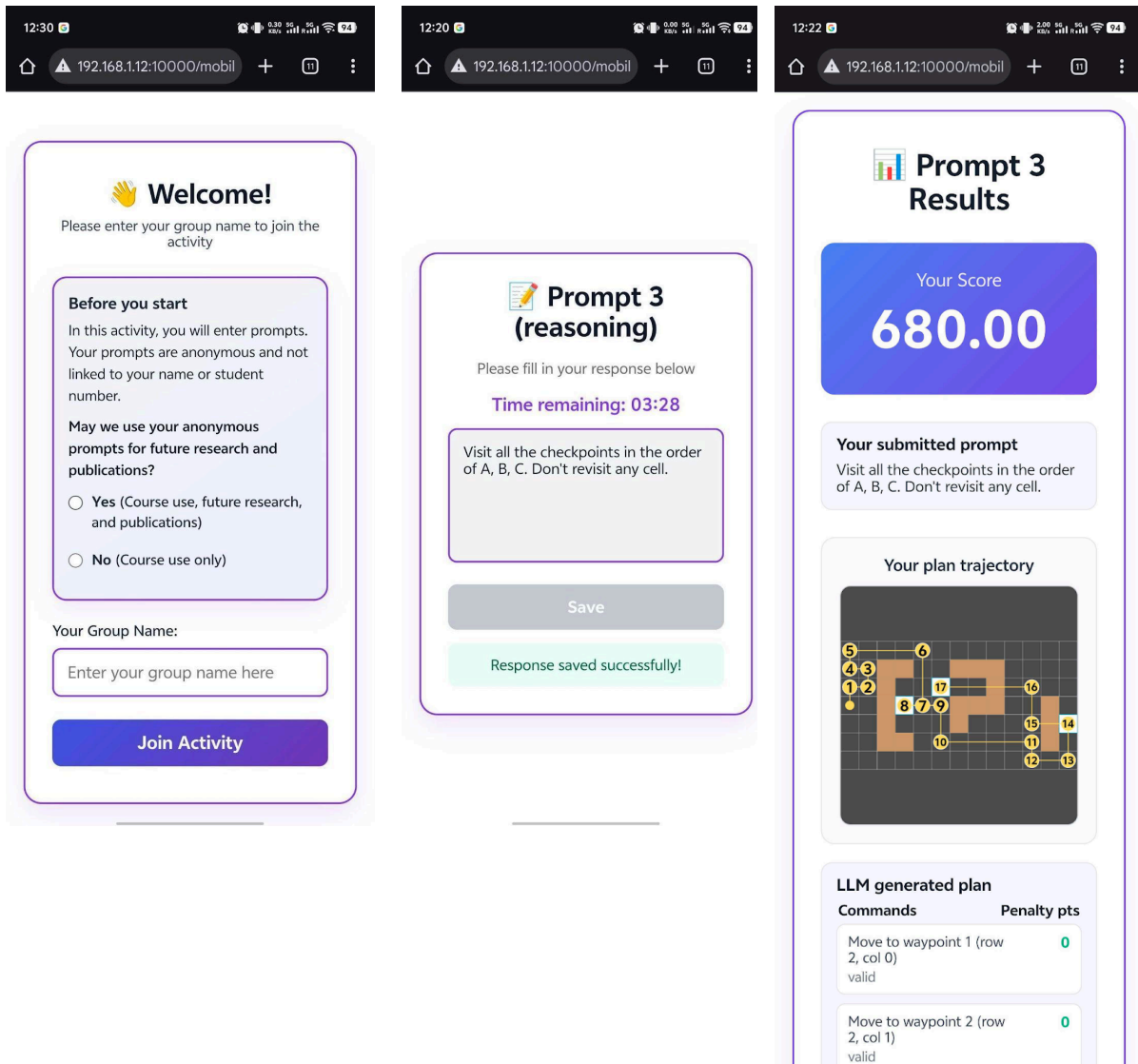


Figure 5.3. Screenshots from the client web interface, which students could access through their phones. Left: Welcome and consent screen; Middle: Prompt submission window; Right: Prompt results.

Tasks

We designed three tasks for students to learn by prompting the LLM planners on their phones via the web interface. The tasks used the same simulated environment but varied in difficulty.

- Task 1 was the simplest task: there were three checkpoints (A, B, C) to visit in any order. Students were asked to submit one prompt for the non-reasoning model (Task 1-NR) and one prompt for the reasoning model (Task 1-R).
- Task 2 was more difficult because there were additional constraints: the mobile robot had to visit checkpoints A, B, and C in that specific order, without revisiting any cell. Students were asked to submit a prompt for the reasoning model (Task 2-R).

- Task 3 was the most difficult task; it was based on Task 2 but involved one additional checkpoint (D) to visit. Again, students were asked to submit a prompt for the reasoning model (Task 3-R).

Each plan started with a baseline score of 1000, from which penalties were deducted whenever an action violated a constraint such as colliding with an obstacle or moving illegally. The GUI applied these penalties to the LLM-generated plans. Tasks were delivered via presentation slides (e.g., see Figure 5.4).

We validated both the task difficulty and the importance of human hints beforehand using simulations. These simulations evaluated the baseline performance (a prompt consisting of the task description only) of the models. The non-reasoning model in Task 1 consistently generated plans with penalties, while the reasoning model was often able to solve Task 1 perfectly without human guidance. However, the performance of the reasoning model declined in Tasks 2 and 3 as the difficulty increased. Figure 5.5 presents an example plan from the reasoning model with a basic task description in Task 3.

We hypothesized that providing guidance on how to reach a checkpoint would improve the reasoning model's performance in Tasks 2 and 3, as such guidance reduces the planner's search space. To verify this, we evaluated Task 3 using prompts with varying levels of guidance, ranging from directions for the first checkpoint to directions for all four. The results supported our hypothesis: more guidance generally led to better performance. For example, a prompt containing "... Go to A from the top and to B from the bottom." achieved a mean score of 650 ($SD = 403$; $n = 10$ repetitions; mean tokens used = 6,914), compared to the baseline mean score of 355 ($SD = 395$; $n = 10$ repetitions; mean tokens used = 13,471).

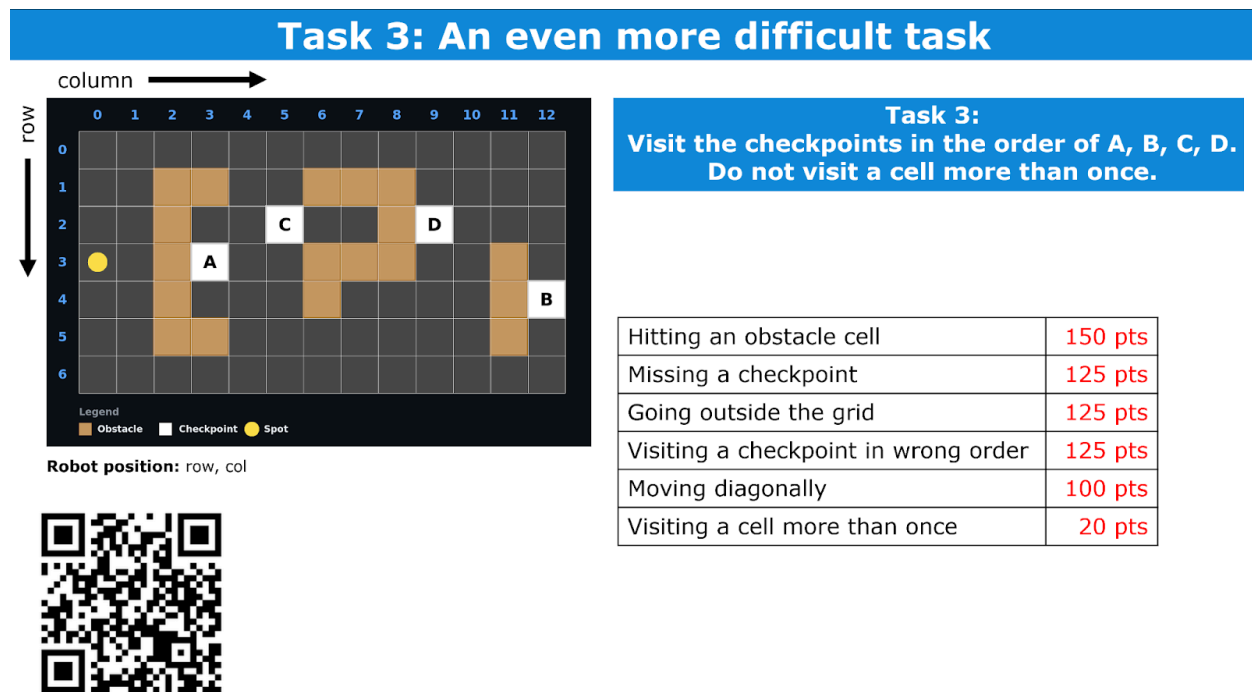


Figure 5.4. Presentation slide used to present Task 3.

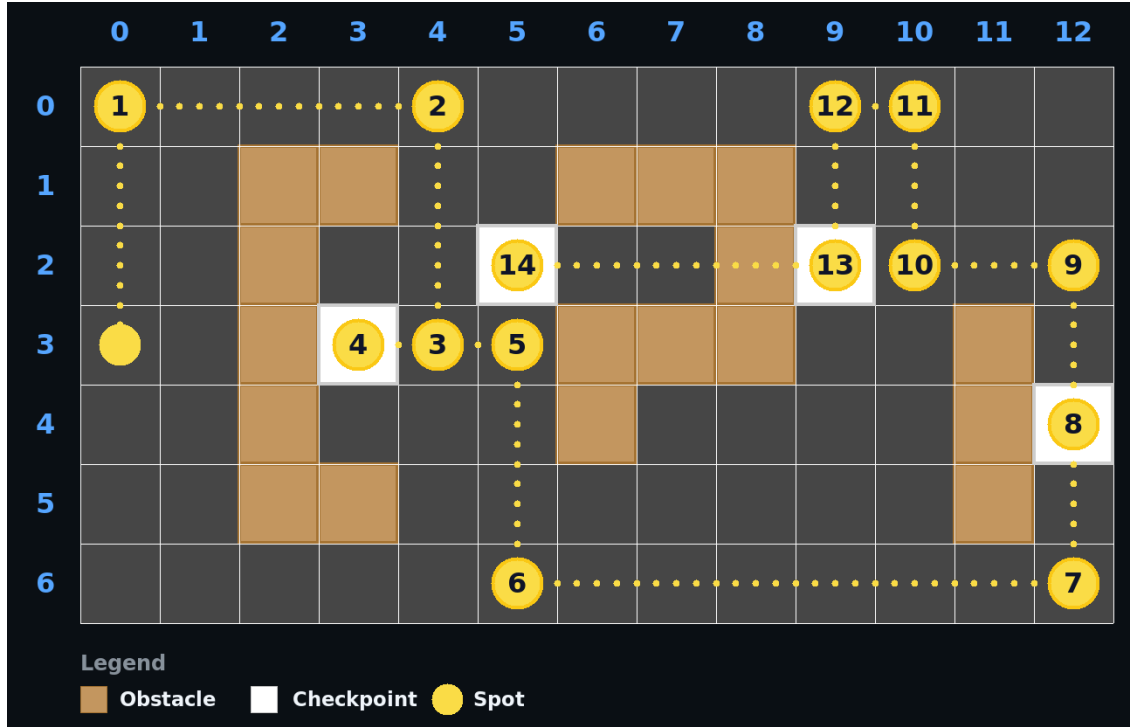


Figure 5.5. GUI screenshot visualizing a Task 3 trajectory plan produced by the reasoning model from a basic task description only. Two types of penalty occurred: colliding with an obstacle (from “13” to “14”) and revisiting two cells (from “4” to “5”). The score of this plan was 810 ($1000 - 150 - 2 \times 20$).

To highlight differences in model capabilities across tasks when interacting with students, we designed one example prompt per task condition. In Task 1, we provided the non-reasoning model with coordinate-based, step-by-step instructions (e.g., “Go to (0,0), then go to (0,4), then go to (3,4) ...”), while the reasoning model received only a basic task description. In Task 2, we used a basic task description augmented with a “self-check” instruction (e.g., “...Check every action you predict with all the known rules”)¹. In Task 3, we supplied a basic task description with directional hints (e.g., “...Approach A from the top and B from the bottom”).

Workshop flow

The workshop was designed as an 80-minute session. The protocol evolved slightly after the first two workshops based on feedback about login timing, when and how to present the system prompt, and how to run the robot demonstration. It was also noted from the first two sessions that working in teams was more engaging than working individually. The workshop was led by two instructors: one delivered the presentation and the other managed the setup, while both facilitated the discussion on prompts.

¹ It should be noted that while self-correction methods can improve accuracy for standard models (Tie et al., 2025), their reliability varies across tasks; standard models frequently even show degraded performance under intrinsic self-correction (Huang et al., 2024). When applied to modern reasoning LLMs, these interventions yield only marginal gains relative to their computational cost, as reasoning models natively execute internal error-detection and correction steps during generation (Tie et al., 2025). In summary, the fact that this specific ‘self-check’ prompt worked in the reasoning context of Task 2 should not be taken as evidence that such prompts are a generally recommended strategy.

Students were first introduced to the learning goals, which specifically focused on the background of LLMs and the principles of controlling robots via natural language. Next, the students joined a local Wi-Fi network by scanning a QR code to access the web interface on their phones, mostly in teams of two. After scanning a second QR code to enter the interface, students provided an alias and indicated whether they consented to the use of their data for research purposes.

Then, an interactive phase took place, consisting of three tasks where the students created prompts. The instructor explained the system prompt and introduced Task 1 by explaining the task goal, the gameplay (an initial score, penalties, and the leaderboard), and the LLMs used. Students were given eight minutes to draft their two prompts for Task 1. Prompts from all student teams were then collected via the web interface and sent to the LLM API in parallel via the GUI.

While complete plan generation of all submitted prompts for the non-reasoning model required only a few seconds, the reasoning model typically required approximately 40 seconds, and occasionally several minutes. When waiting for plans to be generated, the instructor facilitated a discussion with students using several student-written prompts, encouraging them to predict the performance. Once plan generation and evaluation were complete, the instructors led a further discussion using visualizations of the generated plans to analyze why certain prompts worked better than others. The cycle ended with the instructor showing one of the four example prompt outputs, after which students reviewed their prompt outputs, consisting of the text-based plan, visual trajectories, scores, and detailed penalty breakdowns, on their phones and discussed with instructors if they wanted to. The total duration of the discussions was approximately 12 minutes (6 minutes for Task 1-NR; 6 minutes for Task 1-R).

The cycle of briefing, prompt drafting, and discussion was repeated for Tasks 2 and 3, with a prompt-drafting time limit of 4 minutes and a mean discussion time of about 8 minutes for each of the tasks. After completing the interaction of all three tasks, the instructor showed a final leaderboard, depicting student teams' mean scores across the four generated prompts, via the host web interface on the TV screen (Figure 5.2).

Following the tasks, the instructor summarized key takeaways regarding human guidance in planning tasks and provided transparency on the computational costs per prompt. The instructor explained that for Task 1, compared to the non-reasoning LLM, the reasoning LLM used many more tokens and therefore incurred higher expenses. On the upside, it was emphasized that the reasoning model allowed for high-level communication and offloading of the path planning to the LLM.

Students were then invited to complete an anonymous feedback questionnaire consisting of four questions. The workshop culminated in a 10-minute live demonstration of a Boston Dynamics Spot robot. The robot executed a real-world, penalty-free trajectory based on a student-generated plan from Task 2. The demonstration included manipulation: the robot was commanded to “*grasp the tomato can in front of you*” using Gemini 2.5 Pro (Google, 2025b),

demonstrating that LLMs can handle multimodal input, i.e., text and image data. The instructors explained the reality gap using robot localization as an example, mentioning the sensor offset, robot locomotion, and the transformation from the grid frame to the real world, and how LLMs can also be used for perceptual tasks.

Data collection and analysis

We collected all prompts submitted by students via the web interface, including the LLM-predicted plans and token usage. The evaluations of the students' prompts were analyzed using three measures: (1) the penalty-free rate, defined as the proportion of prompts resulting in LLM plans that had no penalty events; (2) the no-plan rate, defined as the proportion of prompts for which the LLM output "FAIL"; and (3) penalty counts, defined as the number of penalties triggered within a plan (see Figure 5.4). No-plan prompts were excluded when reporting the penalty counts.

The feedback questionnaire included two Likert-scale questions and two open-ended questions:

Q1: *To what extent did the workshop help you understand the differences between non-reasoning LLMs and reasoning LLMs?* (A great deal, A lot, Moderately, A little, Not at all, Prefer not to say)

Q2: *How engaged did you feel during the workshop?* (A great deal, A lot, Moderately, A little, Not at all, Prefer not to say)

Q3: *What were your main strategies for formulating prompts so that the LLM would understand your intent?*

Q4: *Do you have any other feedback on the workshop? What was the most valuable part, and what changes could we make to improve it?*

The students' prompts and answers to Q3 and Q4 were analyzed using LLM-assisted quantitative content analysis by Gemini 3 Pro (Google, 2025c). Each prompt/answer was evaluated against pre-defined categories using a binary (0 or 1) classification, allowing items to fall into multiple categories or no category at all. The final codebook for the students' prompts, Q3, and Q4 contained 11, 10, and 15 categories, respectively. The full codebooks are provided in Appendix B.

5.3 Results

Prompts generated by students

Across all workshops, a total of 494 prompts ($n = 123, 123, 124, 124$ for Task 1-NR, Task 1-R, Task 2-R, Task 3-R, respectively) were collected from students who provided consent. Not all collected prompts were evaluated by the LLM due to the API rate limit and rare non-responses. Therefore, the performance analysis was conducted on 457 prompts ($n = 123, 122, 101, 111$) with valid evaluation results.

Figure 5.6 summarizes the performance of the students' prompts across the four conditions. For Task 1, the non-reasoning model achieved a penalty-free rate of 18.7%, whereas the reasoning

model achieved 55.7%. This performance improvement is also reflected in the penalty counts (median of 3 versus 0). Performance decreased as task difficulty increased from Task 1 to 3. Specifically, the penalty-free rate was 30.7% for Task 2 and 21.6% for Task 3, with a median number of penalties of 1 and 2, respectively. Furthermore, Task 3 also showed the highest no-plan rate of 19.8%. This was primarily caused by the model’s inability to find a solution during its search, rather than the presence of irrelevant user prompts.

We analyzed the mean frequency of each penalty type within each prompt condition (see Figure 5.7). Collisions were the dominant penalty type across all tasks. Compared with the reasoning model’s performance in Task 1, the non-reasoning model triggered more collisions (1.681 vs. 0.389), more diagonal moves (0.612 vs. 0.035), and more out-of-bounds penalties (0.345 vs. 0.000). For the more complex tasks (Tasks 2 and 3), the second most common penalty was revisiting cells; the mean penalty count for revisiting increased from 0.684 in Task 2 to 1.213 in Task 3.

We also compared computational resource usage (tokens) and time across all tasks (see Figure 5.8). Requests and responses in Task 1-NR were fast and low-cost (median: 1,032 tokens; 1.2 s latency). In contrast, reasoning prompts consumed substantially more resources (Task 1-R median: 5,584 tokens; 38.1 s). Consumption rose as task difficulty increased (Task 3 median: 7,167 tokens; 51.6 s), driven by the model’s greater use of “thinking tokens” (see Appendix C).

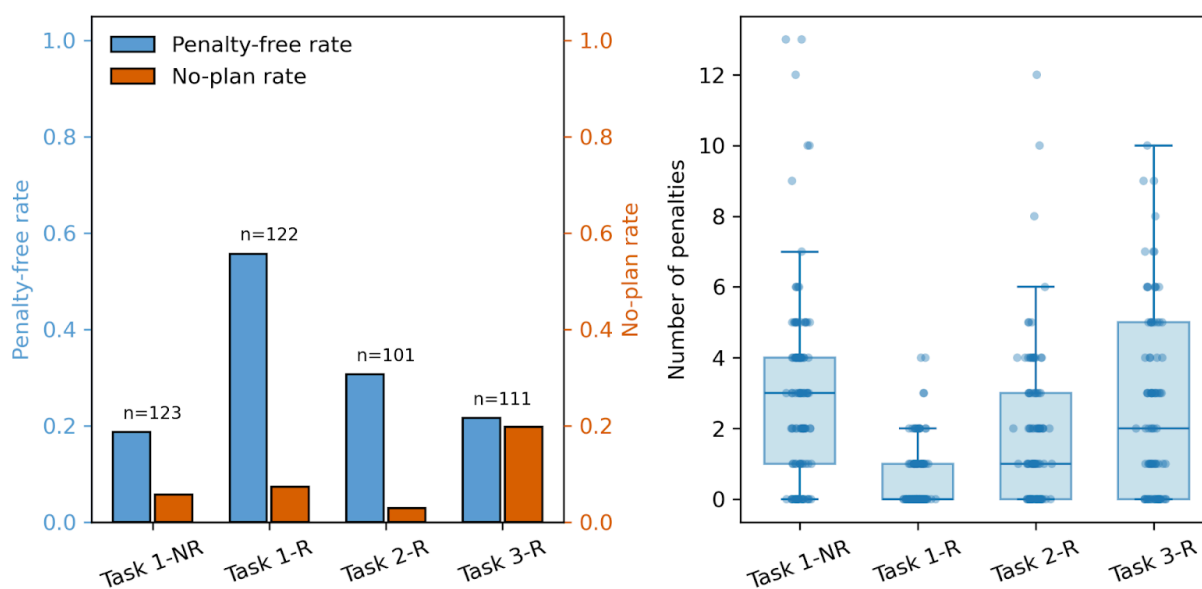


Figure 5.6. Performance of the students’ prompts across tasks. Left: Penalty-free rate and no-plan rate for each prompt condition, with the number of evaluated prompts. Right: The distribution of penalty counts, shown as scatter points, for each prompt condition. NR: Non-reasoning LLM; R: Reasoning LLM.

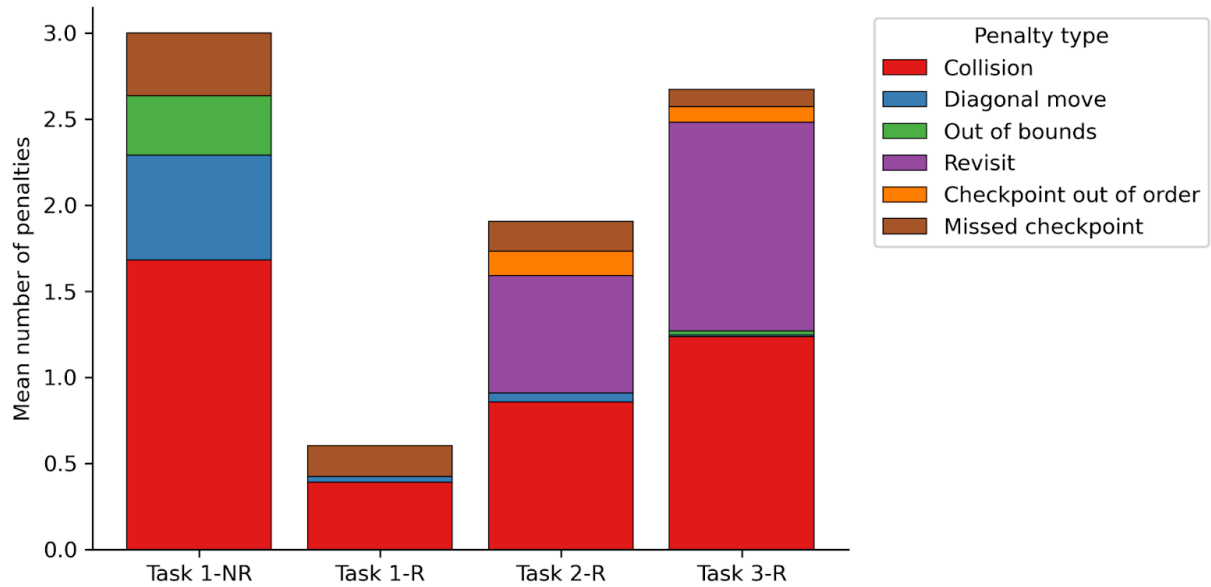


Figure 5.7. Mean number of penalties per plan, categorized by penalty type. NR: Non-reasoning LLM; R: Reasoning LLM.

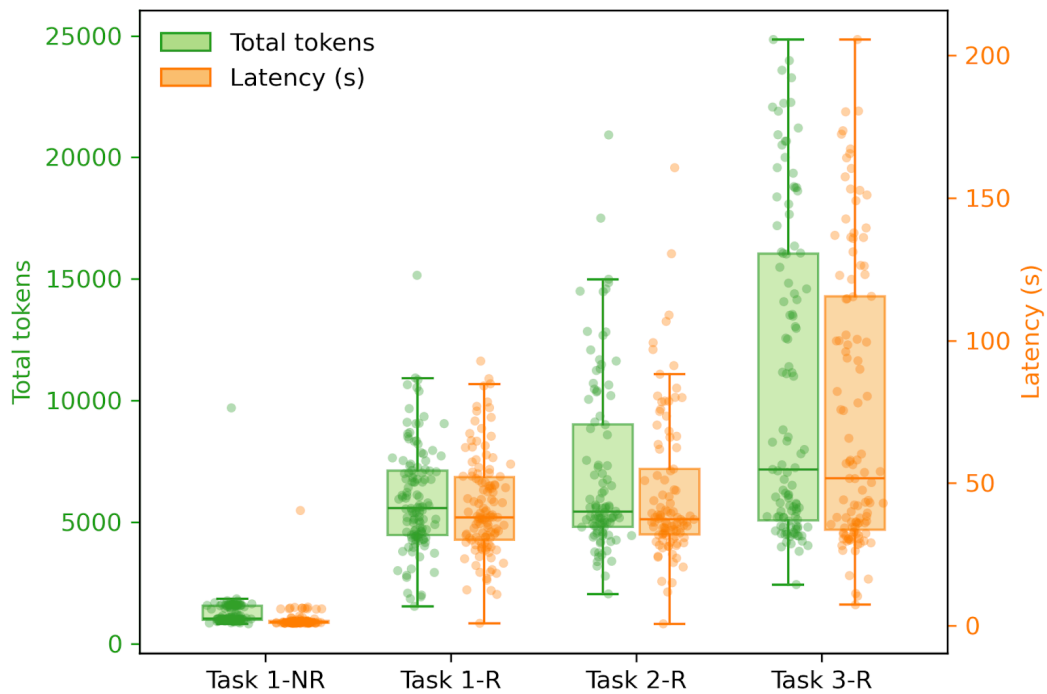


Figure 5.8. Token usage and latency during the workshop. NR: Non-reasoning LLM; R: Reasoning LLM.

We further analyzed the linguistic patterns of the student-generated prompts, focusing on how instructions, constraints, and human guidance were provided within the conditions. Prompting patterns varied by prompt condition (see Figure 5.9). In Task 1, prompts for the non-reasoning model frequently micromanaged navigation with explicit human-planned paths (48.8%), e.g., “...*First move in the straight line to (0,0), Then turn right and move 4 cells...*”, while prompts for the reasoning model contained fewer such step-by-step instructions (18.7%). As task difficulty increased, students started to provide high-level guidance. In Task 3, explicit step-by-step routes were rare (3.2%) and students provided more partial waypoints (37.1%), e.g., “*First go to 6,0 and then to 6,4. Then go to A and after to C...*” or strategic heuristics (23.4%), e.g., “...*Hint: the path consist [sic] 2 counterclockwise spirals...*” They also often asked the model to self-check before outputting a final plan (40.3%), e.g., “*Validate each move*”.

The changes in prompting strategies from Task 1 to Task 3 were consistent with the key messages and example prompts the instructor provided after each task: After Task 1, the instructor explained that the non-reasoning model required step-by-step guidance whereas the reasoning model could perform well with only concise user input of the task description. Such information may have contributed to the decrease in “*Constraint repetition*” and “*Path explicit step-by-step*”. After Task 2, the instructor explained the idea of providing hints or self-checks for complex tasks, which may have resulted in increased usage of “*Path partial waypoints*”, “*Path strategic heuristic*”, and “*Reasoning self-correction*” in Task 3-R.

In an exploratory analysis, we examined the rank correlations (Spearman’s ρ) between the coding of students’ prompts and penalty counts incurred. For this analysis, no-plan prompts were assigned a high penalty count of 10 to reflect their failure to generate a plan. The correlations between the binary coding results (4 task conditions $\times$ 11 categories) and penalty counts revealed statistically significant associations for certain prompt types. Providing step-by-step guidance with the non-reasoning model in Task 1, such as “*Go 3 steps down Go 12 steps right Go 2 steps up...*”, was associated with fewer penalties ($\rho = -0.437$, $p < 0.001$, $n = 123$), as were heuristic strategies when using the reasoning model in Task 3 (e.g., “*Go to A through row zero, then move in a spiral way to B through row 6*”; $\rho = -0.240$, $p = 0.011$, $n = 111$). In contrast, prompts containing additional (and unneeded) optimization constraints (e.g., “...*Find path with minimal amount of state changes...*”) with the reasoning model in Task 2 resulted in higher penalty counts ($\rho = 0.196$, $p = 0.049$, $n = 101$). Repeating known constraints had mixed effects: a positive correlation for the non-reasoning model in Task 1 ($\rho = 0.210$, $p = 0.019$, $n = 123$), but negative correlations with the reasoning model in Tasks 1 and 2 ($\rho = -0.213$, -0.203 ; $p = 0.018$, 0.041 ; $n = 122$, 101). All other correlations were not significant ($p > 0.05$).

Post-task questionnaire

In total, 214 questionnaire responses were collected. We included in the research analysis only responses from participants who consented to the use of their data for future research and publication ($n = 200$). The response distributions of Q1 and Q2 are visualized in Figure 5.10. A majority reported high perceived learning and engagement: 81.5% selected “A lot” or “A great deal” for understanding model differences, and 91.0% selected “A lot” or “A great deal” for engagement (Figure 5.10).

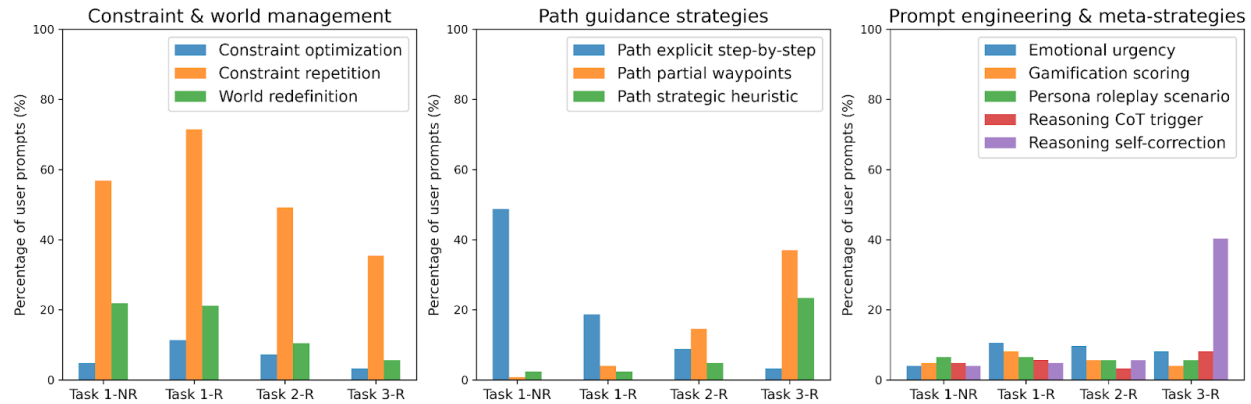


Figure 5.9. Distribution of the categorization of students' prompts, by task. NR: Non-reasoning LLM; R: Reasoning LLM.

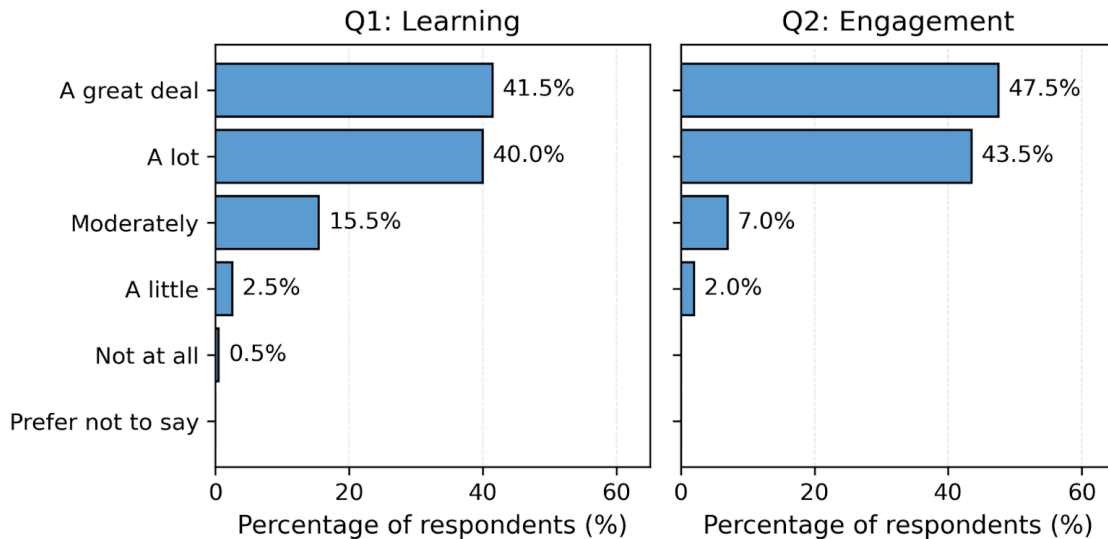


Figure 5.10. Response distributions for Question 1 (Q1, left) and Question 2 (Q2, right) of the post-task questionnaire ($n = 200$).

For the open-ended question Q3 (*“What were your main strategies for formulating prompts so that the LLM would understand your intent?”*), we excluded six empty submissions, leaving 194 responses for analysis. Among 10 final coded strategies, the most frequently reported strategies were using explicit, unambiguous task descriptions (33.5%; e.g., *“be as precise as possible”*) and keeping prompts clear and concise (29.9%; e.g., *“Be clear and concise”*). Beyond these, reiterating rules and key instructions was also common (19.1%; e.g., *“Repeating rules & instructions”*). Participants further described two levels of guidance in their strategies: providing partial plans or hints (18.6%; e.g., *“Give direction to follow”*) and, less often, providing complete solution steps (14.4%; e.g., *“instruct the LLM to follow the path”*).

Students were also asked to respond to the open-ended question: *“Do you have any other feedback on the workshop? What was the most valuable part, and what changes could we*

make to improve it?” (Q4). A total of 137 responses were provided to this question. Responses were mostly positive, consistent with Q1 and Q2. A total of 43.1% used the text field to express general satisfaction (e.g., *"I liked it!"*), while 10.2% explicitly stated they had no additional feedback or critique to offer (Figure 5.11).

When asked about the most valuable parts, participants mostly highlighted the pedagogical value over the hardware. The visualization of results was the most frequently mentioned feature (13.1%), with participants appreciating *"seeing how [their] prompts performed"* in the simulation. This was followed by the comparative analysis of prompts (11.7%), with students noting that *"the discussion about why each prompt failed and worked proved to be valuable."* In addition, 6.6% of participants recognized the value of hands-on experience in prompting, commenting that the *"Most valuable [aspect] was to do it ourselves"*.

Suggestions for improvement clustered around logistical friction and conceptual depth. The most frequent issue (8.8%) concerned the setup of the workshop, mentioning the sitting space (*"Create wider places to sit :)"*), view of the screen (*"...a larger screen..."*), and technical issues such as the API rate limit and the QR-code-based login. Pedagogically, 6.6% of students requested greater real-world context, asking for examples of *"how LLM based robotics is used... to perform real tasks"* to better understand the practical utility of the exercises. Because the demo occurred after the questionnaire, students were not yet aware of the real-world applications being presented. Additionally, some students felt the pacing and difficulty could be adjusted; while some noted the *"time between the tasks is quite [long]"*, others suggested the exercises *"could be even more difficult/challenging"* for the LLMs as the initial tasks were *"quite simple"*.

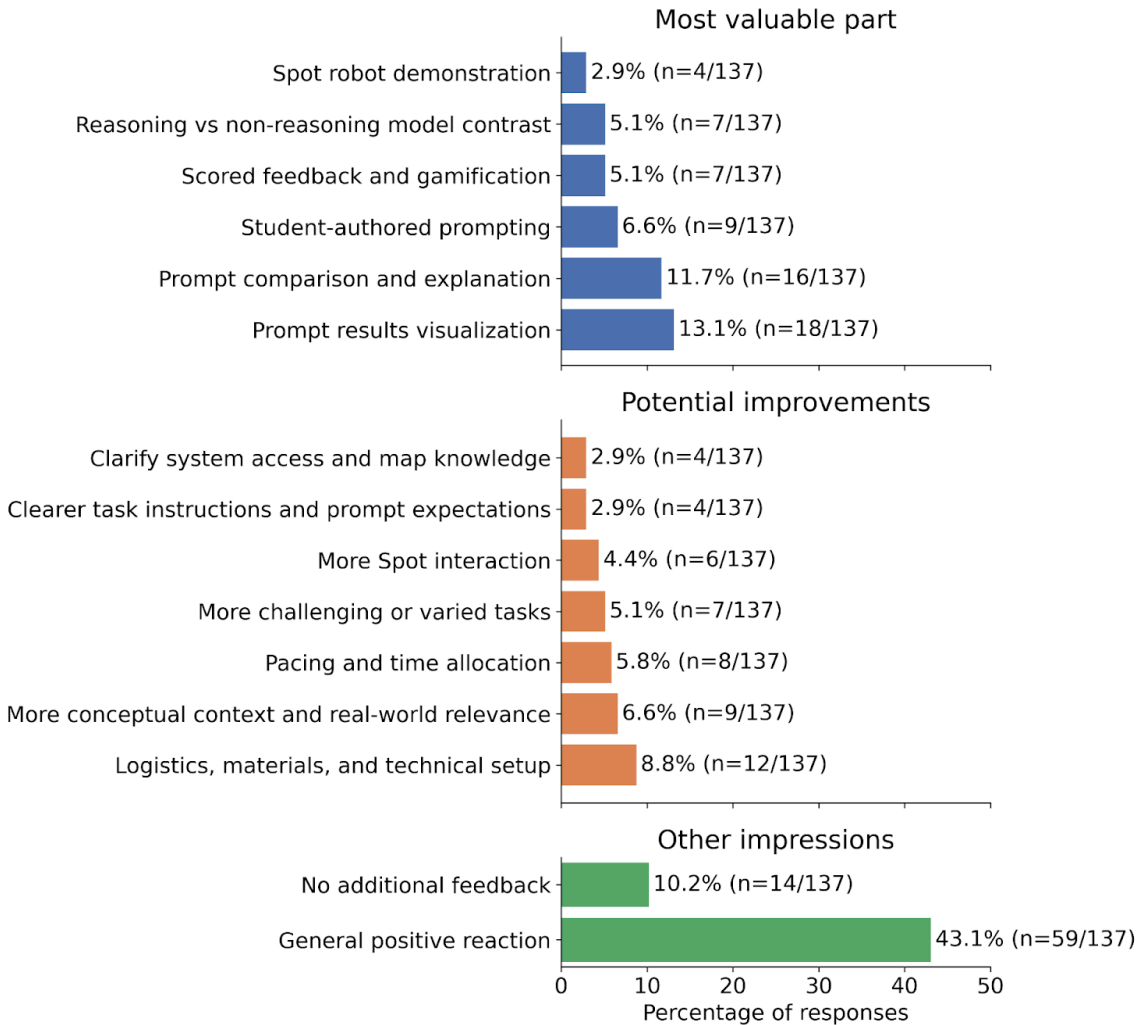


Figure 5.11. Distribution of the students' responses to Question 4 (*"Do you have any other feedback on the workshop? What was the most valuable part, and what changes could we make to improve it?"*; Q4) of the post-task questionnaire.

5.4 Discussion

This study set out to answer two questions: (1) whether a scalable gamified robotics workshop can teach critical LLM use at cohort scale, and (2) how students' prompting strategies change as task constraints increase. Robotics is well suited for this because an LLM's output can be directly executed, and any errors show up as observable robot behavior.

By placing students in a gamified setting, we found that hands-on exposure to model limitations was associated with high self-reported learning about verification and model differences. The workshop's success was revealed by high levels of student engagement and perceived learning; questionnaire responses indicated that 91.0% felt highly engaged, while 81.5% reported learning "a lot" or "a great deal" about the distinctions between model types. This was achieved with a lightweight, bring-your-own-device setup, making the format easy to replicate in other

courses. In total, processing the 494 prompts cost an estimated 23.19 USD, and median end-to-end latency ranged from 1.2 s (Task 1 non-reasoning) to 51.6 s (Task 3 reasoning), making cost and delay concrete constraints that students had to account for.

The prompt data show that students shifted their strategies in a way consistent with a distributed-cognition view of human-AI collaboration. As task complexity increased relative to model capability, students came to redistribute the inferential load. They moved from step-by-step guidance for non-reasoning models toward higher-level guidance when using a reasoning model in a complex task, i.e., using waypoints or strategic heuristics to reduce the model's search space. Students appeared to calibrate their strategies based on the instructor's guidance and their own experience, learning when a brief task description was enough and when they needed to provide more structure.

From a robotics perspective, while the workshop used a direct "LLM-as-planner" architecture (Chen et al., 2024; Huang et al., 2022; Silver et al., 2024; Zhou et al., 2024) to deliberately expose students to the consequences of unverified outputs, such as collisions, it is important to acknowledge that responsible professional deployment requires a safer architecture. In real-world applications, relying solely on the probabilistic output of an LLM for motor control would be irresponsible at the current state of technology. As discussed during the workshop, a viable alternative to the direct prompting used in our exercises is "LLM-informed classical planning" (Meng et al., 2024; Xiao & Yamasaki, 2025), where the LLM is used for its semantic understanding and high-level logic, while the actual trajectory generation is delegated to rule-based solvers. In addition, an algorithmic safety layer could ensure that, regardless of the LLM's hallucinations, the robot acts within physically feasible bounds, preventing collisions (Ravichandran et al., 2026).

Several limitations should be noted. The use of a discrete grid world abstracts away continuous dynamics in physical robotics, which simplifies the verification challenges students face. Additionally, the efficacy of the prompt strategies observed may depend on the specific model versions used (Gemini 2.5 Pro; Google, 2025b) and may evolve rapidly as reasoning capabilities improve. Furthermore, while a simulation-to-reality component was present via the Spot robot demonstration, logistical constraints meant that most student interaction occurred in simulation, limiting exposure to hardware capabilities such as battery limits, sensing uncertainty, and communication latency.

In conclusion, this work showed that scalable gamified environments can connect theoretical AI knowledge with practical engineering skills. By engaging students in hypothesis generation, testing, and correction, the workshop provides a practical model for teaching the distributed cognition required in modern human-AI collaboration. The role of the engineer will increasingly resemble that of a verifier and supervisor; curricula should therefore continue to emphasize that while AI can accelerate planning, the final check before physical action remains a human obligation.

References

- Ahn, M., Brohan, A., Brown, N., Chebotar, Y., Cortes, O., David, B., Finn, C., Fu, C., Gopalakrishnan, K., Hausman, K., Herzog, A., Ho, D., Hsu, J., Ibarz, J., Ichter, B., Irpan, A., Jang, E., Jauregui Ruano, R., Jeffrey, K., ... Zeng, A. (2023). Do as I can, not as I say: Grounding language in robotic affordances. In K. Liu, D. Kulic, & J. Ichnowski (Eds.), *Proceedings of the 6th Conference on Robot Learning* (Vol. 205, pp. 287–318). PMLR. <https://proceedings.mlr.press/v205/ichter23a/ichter23a.pdf>
- Arenas, M. G., Xiao, T., Singh, S., Jain, V., Ren, A., Vuong, Q., Varley, J., Herzog, A., Leal, I., Kirmani, S., Prats, M., Sadigh, D., Sindhvani, V., Rao, K., Liang, J., & Zeng, A. (2024). How to prompt your robot: A promptbook for manipulation skills with code as policies. *Proceedings of the 2024 IEEE International Conference on Robotics and Automation*, Yokohama, Japan, 4340–4348. <https://doi.org/10.1109/ICRA57147.2024.10610784>
- Black, K., Brown, N., Driess, D., Esmail, A., Equi, M., Finn, C., Fusai, N., Groom, L., Hausman, K., Ichter, B., Jakubczak, S., Jones, T., Ke, L., Levine, S., Li-Bell, A., Mothukuri, M., Nair, S., Pertsch, K., Shi, L., ... Zhilinsky, U. (2025). π_0 : A vision-language-action flow model for general robot control. *Robotics: Science and Systems XXI*, Los Angeles, CA. <https://doi.org/10.15607/rss.2025.xxi.010>
- Bonwell, C. C., & Eison, J. A. (1991). *Active learning: Creating excitement in the classroom* (ASHE-ERIC Higher Education Report No. 1). ERIC Clearinghouse on Higher Education. <https://files.eric.ed.gov/fulltext/ED336049.pdf>
- Boston Dynamics. (2020). *Spot – The agile mobile robot*. <https://bostondynamics.com/products/spot>
- Boston Dynamics. (2025). *Spot SDK* (Version 5.0.1) [Computer software]. <https://github.com/boston-dynamics/spot-sdk/releases/tag/v5.0.1>
- Chen, G., Yang, L., Jia, R., Hu, Z., Chen, Y., Zhang, W., Wang, W., & Pan, J. (2024). Language-augmented symbolic planner for open-world task planning. *Robotics: Science and Systems 2024*, Delft, The Netherlands. <https://doi.org/10.15607/rss.2024.xx.037>
- Cotton, D. R. E., Cotton, P. A., & Shipway, J. R. (2024). Chatting and cheating: Ensuring academic integrity in the era of ChatGPT. *Innovations in Education and Teaching International*, 61(2), 228–239. <https://doi.org/10.1080/14703297.2023.2190148>
- Freeman, S., Eddy, S. L., McDonough, M., Smith, M. K., Okoroafor, N., Jordt, H., & Wenderoth, M. P. (2014). Active learning increases student performance in science, engineering, and mathematics. *Proceedings of the National Academy of Sciences*, 111(23), 8410–8415. <https://doi.org/10.1073/pnas.1319030111>
- Google. (2025a). *Gemini 2.0 Flash-Lite*. Google Cloud Documentation. <https://docs.cloud.google.com/vertex-ai/generative-ai/docs/models/gemini/2-0-flash-lite>
- Google. (2025b). *Gemini 2.5 Pro*. Google Cloud Documentation. <https://docs.cloud.google.com/vertex-ai/generative-ai/docs/models/gemini/2-5-pro>
- Google. (2025c). *Gemini 3 Pro*. Google Cloud Documentation. <https://docs.cloud.google.com/vertex-ai/generative-ai/docs/models/gemini/3-pro>
- Google. (2025d). *Thinking*. Google Cloud Documentation. <https://docs.cloud.google.com/vertex-ai/generative-ai/docs/thinking#gemini-2-5-and-earlier-models>
- Huang, J., Chen, X., Mishra, S., Zheng, H. S., Yu, A. W., Song, X., & Zhou, D. (2024). Large language models cannot self-correct reasoning yet. *Proceedings of the Twelfth International Conference on Learning Representations*, Vienna, Austria. <https://openreview.net/pdf?id=lkmD3fKBPQ>
- Huang, W., Abbeel, P., Pathak, D., & Mordatch, I. (2022). Language models as zero-shot planners: Extracting actionable knowledge for embodied agents. In Chaudhuri, K., Jegelka, S., Song, L., Szepesvari, C., Niu, G., & Sabato, S. (Eds.), *Proceedings of the 39th International Conference on Machine Learning* (pp. 9118–9147). PMLR. <https://proceedings.mlr.press/v162/huang22a/huang22a.pdf>
- Kambhampati, S., Valmeekam, K., Guan, L., Verma, M., Stechly, K., Bhambri, S., Saldyt, L. P., & Murthy, A. B. (2024). Position: LLMs can't plan, but can help planning in LLM-modulo frameworks. In R. Salakhutdinov, Z. Kolter, K. Heller, A. Weller, N. Oliver, J. Scarlett, & F. Berkenkamp (Eds.), *Proceedings of the 41st International Conference on Machine Learning* (Vol 235, pp. 22895–22907). PMLR. <https://openreview.net/pdf?id=Th8JPEmH4z>
- Kasneci, E., Seßler, K., Küchemann, S., Bannert, M., Dementieva, D., Fischer, F., Gasser, U., Groh, G., Günemann, S., Hüllermeier, E., Krusche, S., Kutyniok, G., Michaeli, T., Nerdel, C., Pfeffer, J., Poquet,

- O., Sailer, M., Schmidt, A., Seidel, T., ... Kasneci, G. (2023). ChatGPT for good? On opportunities and challenges of large language models for education. *Learning and Individual Differences*, 103, Article 102274. <https://doi.org/10.1016/j.lindif.2023.102274>
- Kumar, A., & Cohen, W. W. (2026). *Localizing and correcting errors for LLM-based planners* [Preprint]. arXiv. <https://doi.org/10.48550/arXiv.2602.00276>
- Liang, J., Huang, W., Xia, F., Xu, P., Hausman, K., Ichter, B., Florence, P., & Zeng, A. (2023). Code as policies: Language model programs for embodied control. *Proceedings of the 2023 IEEE International Conference on Robotics and Automation*, London, UK, 9493–9500. <https://doi.org/10.1109/icra48891.2023.10160591>
- Lo, C. K. (2023). What is the impact of ChatGPT on education? A rapid review of the literature. *Education Sciences*, 13(4), Article 410. <https://doi.org/10.3390/educsci13040410>
- Meng, S., Wang, Y., Yang, C.-F., Peng, N., & Chang, K.-W. (2024). LLM-A*: Large language model enhanced incremental heuristic search on path planning. In Y. Al-Onaizan, M. Bansal, & Y.-N. Chen (Eds.), *Findings of the Association for Computational Linguistics: EMNLP 2024* (pp. 1087–1102). Association for Computational Linguistics. <https://doi.org/10.18653/v1/2024.findings-emnlp.60>
- Morales-Trujillo, M. E., & García-Mireles, G. A. (2020). Gamification and SQL: An empirical study on student performance in a database course. *ACM Transactions on Computing Education*, 21(1), Article 3. <https://doi.org/10.1145/3427597>
- Prince, M. (2004). Does active learning work? A review of the research. *Journal of Engineering Education*, 93(3), 223–231. <https://doi.org/10.1002/j.2168-9830.2004.tb00809.x>
- Quigley, M., Conley, K., Gerkey, B., Faust, J., Foote, T., Leibs, J., Berger, E., Wheeler, R., & Ng, A. Y. (2009). ROS: An open-source Robot Operating System. *Proceedings of the ICRA Workshop on Open Source Software*, Kobe, Japan. <https://ai.stanford.edu/~mquigley/papers/icra2009-ros.pdf>
- Ravichandran, Z., Robey, A., Kumar, V., Pappas, G. J., & Hassani, H. (2026). Safety guardrails for LLM-enabled robots. *IEEE Robotics and Automation Letters*, 11(4), 4649–4656. <https://doi.org/10.1109/lra.2026.3667488>
- Sailer, M., & Homner, L. (2020). The gamification of learning: A meta-analysis. *Educational Psychology Review*, 32(1), 77–112. <https://doi.org/10.1007/s10648-019-09498-w>
- Shojaee, P., Mirzadeh, I., Alizadeh, K., Horton, M., Bengio, S., & Farajtabar, M. (2025). The illusion of thinking: Understanding the strengths and limitations of reasoning models via the lens of problem complexity. *Advances in Neural Information Processing Systems* 38. <https://openreview.net/forum?id=YghiOusmvw>
- Shridhar, M., Manuelli, L., & Fox, D. (2022). CLIPort: What and where pathways for robotic manipulation. In A. Faust, D. Hsu, & G. Neumann (Eds.), *Proceedings of the 5th Conference on Robot Learning* (Vol. 164, pp. 894–906). PMLR. <https://proceedings.mlr.press/v164/shridhar22a/shridhar22a.pdf>
- Silver, T., Dan, S., Srinivas, K., Tenenbaum, J. B., Kaelbling, L. P., & Katz, M. (2024). Generalized planning in PDDL domains with pretrained large language models. *Proceedings of the AAAI Conference on Artificial Intelligence*, 38(18), 20256–20264. <https://doi.org/10.1609/aaai.v38i18.30006>
- Tie, G., Yuan, Z., Zhao, Z., Hu, C., Gu, T., Zhang, R., Zhang, S., Wu, J., Tu, X., Jin, M., Wen, Q., Chen, L., Zhou, P., & Sun, L. (2025). Can LLMs correct themselves? A benchmark of self-correction in LLMs. *Proceedings of the 39th Conference on Neural Information Processing Systems*, San Diego, CA. <https://openreview.net/pdf?id=956KYtqwU>
- Valmeekam, K., Marquez, M., Sreedharan, S., & Kambhampati, S. (2023). On the planning abilities of large language models: A critical investigation. *Advances in Neural Information Processing Systems*, 36, 75993–76005. https://proceedings.neurips.cc/paper_files/paper/2023/file/efb2072a358cefb75886a315a6fcf880-Paper-Conference.pdf
- Wei, J., Wang, X., Schuurmans, D., Bosma, M., Xia, F., Chi, E., Le, Q., & Zhou, D. (2022). Chain-of-thought prompting elicits reasoning in large language models. In S. Koyejo, S. Mohamed, A. Agarwal, D. Belgrave, K. Cho, & A. Oh (Eds.), *Advances in Neural Information Processing Systems* (Vol. 35, pp. 24824–24837). https://proceedings.neurips.cc/paper_files/paper/2022/file/9d5609613524ecf4f15af0f7b31abca4-Paper-Conference.pdf
- Xiao, L., & Yamasaki, T. (2025). *LLM-Advisor: An LLM benchmark for cost-efficient path planning across multiple terrains* [Preprint]. arXiv. <https://doi.org/10.48550/arXiv.2503.01236>

- Zhang, D., Li, Z.-Z., Zhang, M.-L., Zhang, J., Liu, Z., Yao, Y., Xu, H., Zheng, J., Chen, X., Zhang, Y., Yin, F., Dong, J., Guo, Z., Song, L., & Liu, C.-L. (2026). From System 1 to System 2: A survey of reasoning large language models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 48(3), 3335–3354. <https://doi.org/10.1109/tpami.2025.3637037>
- Zhou, D., Schärli, N., Hou, L., Wei, J., Scales, N., Wang, X., Schuurmans, D., Cui, C., Bousquet, O., Le, Q., & Chi, E. (2023). Least-to-most prompting enables complex reasoning in large language models. *Proceedings of the Eleventh International Conference on Learning Representations*, Kigali, Rwanda. <https://openreview.net/pdf?id=WZH7099tgfM>
- Zhou, Z., Song, J., Yao, K., Shu, Z., & Ma, L. (2024). ISR-LLM: Iterative self-refined large language model for long-horizon sequential task planning. *Proceedings of the 2024 IEEE International Conference on Robotics and Automation*, Yokohama, Japan, 2081–2088. <https://doi.org/10.1109/ICRA57147.2024.10610065>

Appendices

A. System prompt

In the workshop, a system prompt was sent together with every student-written prompt to the LLM API server. Below is the system prompt appended to the students' prompts.

```
### ROLE
You are a path planner for a mobile robot operating on a 2D grid. Your goal is to
generate a valid sequence of actions according to the USER_PROMPT.

### AVAILABLE ACTION:
- move_to_cell row=<int> col=<int>

### CONTEXT:
CURRENT_STATE = {current_state_json}
WORLD_MODEL = {world_model_json}

### RULES:
1. Start state:
- The robot begins at CURRENT_STATE.robot_cell.
- The left top cell is at (row=0,col=0) and the right bottom cell is at
(row=6,col=12).
2. Movement logic:
- The robot performs each move_to_cell by traveling in a straight horizontal or
vertical line.
- The robot should not navigate through any obstacle cell (listed in
WORLD_MODEL.obstacle).
- The robot should not move out of the grid.
3. Error handling:
- This system prompt is the ground truth. If a user's prompt is not relevant to the
role or suggests an action that is impossible (e.g., dance or backflip), output:
"FAIL <REASON>".

### OUTPUT FORMAT:
- Return only the action commands, one per line, or a FAIL message.
- Do not emit JSON, comments, numbering, braces, brackets, or quotation marks.

### EXAMPLES
User: "Go to the left top corner"
```

```

CURRENT_STATE.robot_cell: row=3 col=0
Ouput:
move_to_cell row=0 col=0

User: "Hello world"
Ouput:
FAIL Not relevant to the role.

### USER PROMPT
{command}

```

B. Codebook generation and encoding pipeline

We conducted a quantitative content analysis of student prompts using a multi-stage coding process. To develop the initial codebook, we used Gemini 3 Pro (Google, 2025c) to analyze a random sample of 100 prompts (25 per condition). We refined this codebook iteratively by prompting the LLM with guiding questions on prompting patterns and validating the emerging themes, resulting in 11 final categories. These categories were then applied to the full dataset via binary multi-label coding. To mitigate potential LLM order effects, we randomized both the response and code sequences for each run.

The 11 prompt categories (Table 5.B1) comprise three groups:

- Constraint & world management: Includes references to system constraints (Constraint repetition), user-specified parameters (Constraint optimization), and the redefinition of environmental coordinates (World redefinition).
- Path guidance strategies: Ranges from explicit navigational instructions (Path explicit step-by-step) to partial waypoints (Path partial waypoints) and high-level heuristics (Path strategic heuristic).
- Prompt engineering & meta-strategies: Encompasses chain-of-thought triggers (Reasoning CoT trigger), self-validation requirements (Reasoning self-correction), persona framing (Persona roleplay scenario), gamification (Gamification scoring), and emotionally charged language (Emotional urgency).

Following a similar iterative process, we developed codebooks for the open-ended responses in Q3 (10 categories; Table 5.B2) and Q4 (15 categories; Table 5.B3).

Table 5.B1. The codebook for the content analysis of students' prompts

Category	Definition of category
Path_Explicit_StepByStep	The user provides a complete (or nearly complete) sequence of specific coordinates, directional moves (up/down/left/right), or cell-by-cell instructions to solve the task, effectively treating the LLM as a transcriber rather than a planner.
Path_Strategic_Heuristic	The user provides a high-level geometric shape, pattern, or general navigation strategy to guide the pathfinding without providing a strict step-by-step coordinate list.

Path_Partial_Waypoints	The user suggests specific intermediate target cells, rows, or columns to visit between checkpoints to aid navigation, without providing the full trajectory.
Constraint_Repetition	The user explicitly restates rules that are already known to the system via the hidden system prompt (e.g., do not move diagonally, do not hit obstacles, stay within the grid).
Constraint_Optimization	The user imposes a requirement for efficiency or specific performance metrics that were not strictly required by the basic task description (e.g., shortest path, least steps, fewest turns).
World_Redefinition	The user explicitly describes the grid dimensions, obstacle locations, or coordinate system in the prompt, despite the system prompt already containing this data.
Reasoning_CoT_Trigger	The user explicitly instructs the model to think, reason, plan sequentially, or explain its logic (Chain of Thought) rather than just outputting the result.
Reasoning_Self_Correction	The user instructs the model to verify, double-check, or validate its generated path against the rules before outputting the final answer.
Persona_Roleplay_Scenario	The user assigns a specific persona to the AI (e.g., "Robot Planner") or creates a fictional narrative/scenario (e.g., "Prison escape") to frame the task.
Gamification_Scoring	The user introduces a specific point system, rewards, or penalties to influence the model's decision-making process.
Emotional_Urgency	The user employs capitalization, exclamation marks, pleading language, or emotionally charged words to emphasize the importance of following the rules.

Table 5.B2. The codebook for the content analysis of Q3

Category	Definition of category
Conciseness & Simplicity	Strategies focused on keeping prompts short, direct, and free of unnecessary complexity to avoid confusing the model.
Specificity & Explicitness	Strategies involving highly detailed descriptions of the task environment, objectives, or boundaries to ensure no ambiguity.
Model-Specific Adaptation	Strategies that explicitly differentiate the prompting approach based on whether the model is a "reasoning" model or a "non-reasoning" model.
Constraint Emphasis & Repetition	Strategies that focus on reiterating rules, stating what *not* to do (negative constraints), or repeating key instructions to ensure adherence.
Step-by-Step Structuring	Strategies that break the task down into a sequence of operations, logical steps, or use code-like formatting to organize the prompt.

Low-Level Guidance (Solution Providing)	Strategies where the user performs the planning logic themselves and provides the LLM with specific coordinates, row/column numbers, or the exact path to follow.
High-Level Scaffolding (Hinting)	Strategies that provide clues, general directions, or partial plans to guide the model's reasoning without giving the full solution.
Anthropomorphism & Role-Playing	Strategies that treat the LLM as a human (e.g., a child, a student), use emotional appeals, or create high-stakes fictional scenarios.
Self-Correction & Verification	Strategies that explicitly instruct the model to review its own work, double-check constraints, or reflect before finalizing the answer.
Iterative Learning	Strategies based on trial-and-error, adapting prompts based on previous failures, or using examples from prior attempts.

Table 5.B3. The codebook for the content analysis of Q4

Category	Definition of category
Most valuable part	
Student-authored prompting	Include when the response values being able to write/enter their own prompts or commands (doing it themselves).
Prompt results visualization	Include when the response highlights seeing the prompt output visually (path/simulation/results on screen or phone) as valuable.
Scored feedback and gamification	Include when the response values scoring/points/competition or evaluation feedback as valuable.
Prompt comparison and explanation	Include when the response values comparing prompts and discussing/explaining why prompts worked or failed (pros/cons, baseline, analysis).
Reasoning vs non-reasoning model contrast	Include when the response highlights learning differences between reasoning vs non-reasoning models (LLM vs LRM) as valuable.
Spot robot demonstration	Include when the response identifies the Spot robot (or LLM-controlled robotics) demo as a valuable part of the workshop.
Potential improvements	
Clarify system access and map	Include when the response asks to clarify what the model/system already knows or has access to (e.g., map, obstacles, checkpoints) and what needs to be included in prompts.

knowledge	
Clearer task instructions and prompt expectations	Include when the response asks for clearer guidance at the start (what kind of prompts to write, a brief before starting, simpler starter example, or clearer expectations).
Pacing and time allocation	Include when the response requests different timing (more time to write prompts, faster/slower review, longer session, less waiting, or less repetitive discussion).
More challenging or varied tasks	Include when the response requests harder/more complex tasks, different problems, or different task mix (e.g., more examples, different difficulty balance).
More Spot interaction	Include when the response asks for more use of Spot during the workshop (more action, more examples earlier, letting students control, or changes to Spot setup/placement).
Logistics, materials, and technical setup	Include when the response requests changes to physical setup or delivery infrastructure (seating, screen visibility, slides/handouts, wifi/QR onboarding, token/API limits, prompt processing speed, class size, or individual device feedback).
More conceptual context and real-world relevance	Include when the response asks for more explanation/context (why use these models, prompt-engineering strategies, chain-of-thought concepts, real-world robotics use cases, or broader tradeoffs like environmental impact).
Other impressions	
General positive reaction	Include when the response offers a broad positive evaluation (fun, interesting, engaging, informative, well organized) without a specific detailed theme.
No additional feedback	Include when the response explicitly says they have no feedback/changes (e.g., nothing, no comment, no improvements needed).

C. Token usage and cost

This workshop collected a total of 494 prompts from consenting participants. Analysis of the total token usage and associated costs for Gemini 2.0 Flash-Lite (Google, 2025a) and Gemini 2.5 Pro (Google, 2025b) revealed an estimated total expenditure of 23.19 USD. This cost was primarily driven by the consumption of thinking tokens (2,221,258 tokens, totaling 22.21 USD). Input token data indicated that the length of students' prompts was highest in Task 1-NR and lowest in Task 2-R, a pattern consistent with the shift from verbose step-by-step instructions for the non-reasoning model to more concise high-level prompts for the reasoning model.

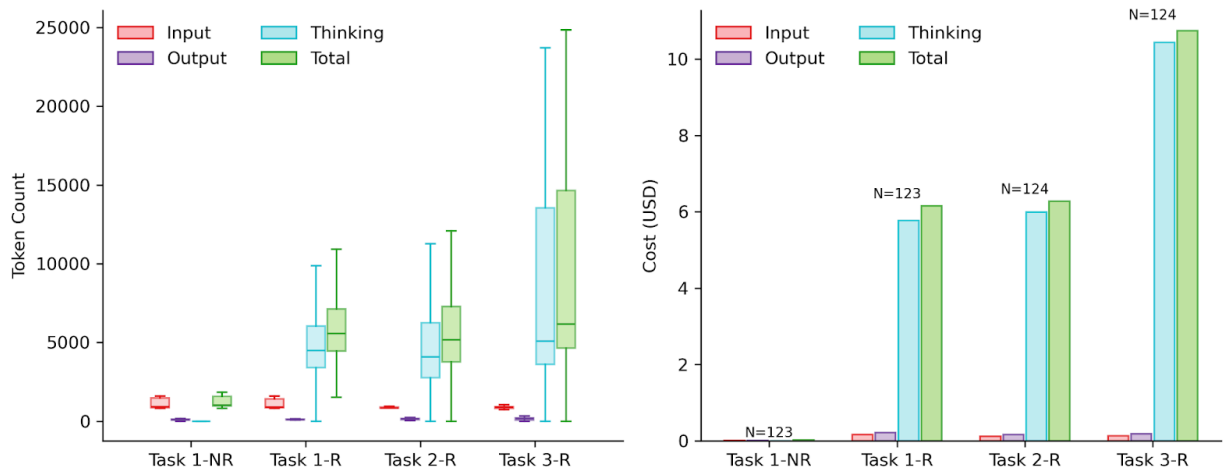


Figure 5.C1. Breakdown of token usage and estimated costs.

D. Approximate workshop timeline

Table 5.D1. Approximate workshop timeline

Phase	Content	Duration
Introduction	Background knowledge of the workshop	10 min
Interaction	Introduction to Task 1	5 min
	Writing prompts for Task 1	8 min
	Running the planner, discussing the outputs	12 min
	Introduction to Task 2	2 min
	Writing prompts for Task 2	4 min
	Running the planner, discussing the outputs	8 min
	Introduction to Task 3	2 min
	Writing prompts for Task 3	4 min
	Running the planner, discussing the outputs	8 min
Conclusion	Key takeaways, questionnaire	7 min
Demonstration	Real robot navigation & grasping	10 min
Total (not including the Q&A after the demonstration)		80 min

6

Discussion and conclusions

Intermezzo

Chapter 5 showed that students recognized LLM reasoning limitations and redistributed cognitive effort accordingly. Together, the four studies located misalignment at every information-processing stage. Chapter 6 will synthesize the findings, answer the research questions and the central question of the thesis, and derive design principles and recommendations for future research.

6.1 Recap: Motivation, research questions, and approach

Machines that work alongside humans have progressed from mechanical devices and teleoperated systems to AI-driven robots that perceive, reason, and act in unstructured environments shared with people (Firoozi et al., 2025). Multimodal large language models (MLLMs) and vision language action models (VLAs) now enable robots to interpret open-ended instructions, plan multi-step actions, generate motor commands from multimodal input and generate natural language explanations (Ahn et al., 2023; Black et al., 2025; Driess et al., 2023). The developments in robotics are driven by the need for higher productivity, safer workplaces, and the support for aging populations, among others (e.g., Graetz & Michaels, 2018; Gihleb et al., 2022; Acemoglu & Restrepo, 2022).

However, capable robots are not a guarantee for effective collaboration, as humans and AI-driven robots perceive, reason, and respond through different information-processing mechanisms. When human and robot processes are misaligned, interaction can fail, regardless of the robot's capability. This thesis investigated where alignment and misalignment occur between humans and AI-driven robots, and how human-robot interaction (HRI) should be designed for complementary interaction.

The thesis adopted the vocabulary of Wickens' Multiple Resource Theory (MRT). The MRT not only distinguishes processing stages (perception, cognition, responding), but also processing modalities (visual and auditory for perception), and processing codes (spatial, verbal; Wickens, 2002, 2008). In this thesis, four studies targeted different processing stages and progressed in platform complexity and realism for real-world applications.

Chapter 2 targeted the processing stage 'perception', the basis (first stage) for subsequent processing, and examined how humans and MLLMs align and differ in visual-spatial search. Chapter 3 covered all three processing stages and multiple modalities; it examined how control modality (voice vs. gesture) and mobility condition (walking vs. standing) affect performance and intuitiveness when teleoperating a mobile robot. Chapter 4 developed an open-source chess robot integrating visual perception, AI-based reasoning, and multimodal responding as a testbed. Chapter 5 focused on the cognitive stage, by investigating whether a gamified workshop can teach engineering students to recognize AI reasoning limitations and redistribute cognitive effort between themselves and the model.

The studies followed a methodological progression: controlled benchmarking with online human participants and LLMs (Chapter 2), a laboratory user study with in-person engineering students (Chapter 3), an international online survey combined with expert gameplay (Chapter 4), and cohort-scale educational deployment (Chapter 5). Participant populations ranged from 1,250 online respondents (Chapter 2) and 597 international survey participants (Chapter 4) to 218 (Chapter 3) and 226 (Chapter 5) in-person engineering students, contributing to the empirical breadth and statistical power of the studies in this thesis.

6.2 Main research findings

Across the four studies, a combined 2,292 human participants generated empirical evidence spanning the different stages of human-AI interaction. The following subsections summarize the methods, results, and conclusions of each chapter before synthesizing them.

6.2.1 Chapter 2: Human-like attention? A psychophysical comparison of visual search in humans and MLLMs

Chapter 2 tested whether MLLMs exhibit human-like attentional bottlenecks in visual-spatial search. A total of 1,250 participants, recruited via the online platform Prolific, completed nine conditions that differed in search type (feature vs. conjunction), distractor heterogeneity, and dimensionality (2D vs. 3D), yielding 157,042 valid trials across seven set sizes (4 to 256). Three MLLMs (Gemini 2.0 Flash-Lite, Gemini 2.5 Flash-Lite, and Gemma 3) were queried on the same 6,300 stimuli.

Correspondence in visual search outcomes was strong: Spearman correlations between human response times and MLLM error rates reached $\rho = 0.91$, and between human and MLLM error rates $\rho = 0.82$. Both humans and MLLMs showed efficient feature search and degraded conjunction search, consistent with Feature Integration Theory (Treisman & Gelade, 1980) and Guided Search (Wolfe, 2021).

However, the underlying mechanisms were found to differ: Humans exhibited a speed-accuracy trade-off with roughly doubled response times on target-absent trials, consistent with the quitting threshold in Guided Search 6.0 (Wolfe, 2021). The MLLMs, on the other hand, did not feature this mechanism and defaulted to biased ‘absent’ or ‘present’ responses in the more complex conditions, with high (>50%) error rates in the hardest search conditions. This deficiency in feature-binding resembles findings from recent research on feature binding as a limitation of vision-language architectures (Campbell et al., 2024; Tangtartharakul & Storrs, 2025; Mehrani & Tsotsos, 2023). The difference between human search and MLLM search can be framed as an analogy introduced earlier: Like humans do in ambient processing, MLLMs encode the entire image in a single parallel pass. Humans then search a scene through a small high-resolution fovea, using sequences of fixations and saccades that allow for strategic sampling of the image, such as first filtering on color and then serially verifying shape among remaining candidates. Most current MLLMs lack this sequential re-inspection. As a result, the overall difficulty rankings across conditions corresponded well between humans and MLLMs, but the search strategies themselves, and the error patterns they produced in complex conjunction conditions, did not.

In summary, the study of Chapter 2 contributed to literature in three ways: (1) it replicated previously established human visual search findings at scale, (2) it provided a systematic psychophysical evaluation of MLLM visual processing, and (3) it offered a direct human-MLLM comparison on identical stimuli. The main conclusion is that MLLMs learned fundamental search principles through pre-training but do not replicate the focal search strategy that humans use. This dissociation observed in Chapter 2 represents an elementary form of misalignment: two agents that appear to perceive the same scene similarly at a surface level, yet fail under

different conditions and for different reasons. This can also have practical consequences. For example, if similar perceptual dissociations extend to embodied robot vision, a warehouse robot and its human supervisor might achieve comparable object detection rates overall, yet the robot could fail specifically on targets defined by a conjunction of features. This answers Research question 1: *MLLMs do exhibit human-like attentional bottlenecks at the outcome level, but their processing diverges fundamentally from human scanning strategy, most likely due to the absence of a focal search stage and iterative feature-binding.*

6.2.2 Chapter 3: Walk Along: An Experiment on Controlling the Mobile Robot ‘Spot’ with Voice and Gestures

Chapter 3 investigated how control modality (speech vs. gesture) and mobility condition (walking vs. standing) affect cognitive load, spatial reasoning, and performance. In a within-subjects 2×2 design, 218 participants teleoperated Boston Dynamics Spot along a 14-node indoor trajectory using voice vs. gesture commands while standing or walking, via the Microsoft HoloLens 2. The experiment of Chapter 3 covered all three processing stages: visual-spatial perception (monitoring the robot), spatial cognition (maintaining the robot’s pose and selecting the next command), and verbal (voice) or spatial (gesture) response.

The results showed that voice-walking was preferred by 53% of participants; gesture-standing was least favored (47% rated it last). Furthermore, voice produced faster inter-command times (5.50 s vs. 5.96 s), higher command detection ratings (4.60 vs. 4.22 on a scale of 1 “disagree” to 5 “agree”) and was preferred by 71% of participants. Walking improved intuitiveness ratings for both modalities. Gesture-standing produced significantly more left-right errors ($M = 0.213$) than gesture-walking ($M = 0.032$), a difference which points to so-called stimulus-response incompatibility (Fitts & Seeger, 1953). This incompatibility arises when operators mentally rotate gestures into the robot’s frame of reference (cf. Shepard & Metzler, 1971), creating a conflict when processing the spatial code used for responding and the visual-spatial code used for monitoring the robot. Voice commands avoid this conflict by operating in a verbal code, thereby reducing left-right conflict for the human. An odometry-based analysis further showed that when allowed to walk, participants physically aligned themselves with the robot more in the gesture condition (46.8%) than in the voice condition (41.9%), suggesting that the embodied nature of gesture control motivated operators to reduce spatial misalignment through repositioning.

In summary, Chapter 3 showed that misalignment is, in part, a property of the interface: command modality redistributes load across the resource pools of the operator. On the technical side, Chapter 3 contributed a validated gesture recognition pipeline for the HoloLens 2 (99% classification accuracy). Thus, the human operator’s command speed and error rate were determined by two layers: the spatial congruence between operator and robot (a human factors issue, reflected in intuitiveness ratings and left-right errors) and the accuracy with which the robot’s recognition system detected commands (a technical issue, where voice recognition was faster and more robust than gesture recognition). When allowed to walk, participants physically co-aligned with the robot to reduce spatial misalignment, particularly in the gesture condition. For robot control design, voice commands should therefore be preferred when egocentric spatial alignment cannot be upheld. Gesture control, however, should not be dismissed entirely:

a substantial minority valued it for its directness and engagement, and combining it with physical co-location largely resolved the left-right confusion observed in the stationary condition. These findings answer Research question 2: *voice control outperformed gestures on both task performance and perceived intuitiveness, and walking benefited both modalities but especially gesture control, where physical co-alignment mitigates the left-right errors observed in the stationary condition.*

6.2.3 Chapter 4: An Open-Source Reproducible Chess Robot for Human-Robot Interaction Research

Chapter 4 developed OpenChessRobot, a chess robot that integrates a Franka Emika Panda robot arm, a ZED2 stereo camera, computer vision (more than 96% piece recognition accuracy), the Stockfish chess engine, and ChatGPT-generated verbal commentary. This research focused on the processing stages ‘perception’ and ‘response’: the robot perceived the chessboard visually and responded through physical manipulation and spoken verbal commentary. The game of chess was chosen because human-AI cognitive differences are well-documented in this specific task (Shannon, 1950; McCarthy, 1990; Campbell et al., 2002) and chess game outcomes are enormously diverse yet objectively verifiable.

An international survey of 597 Prolific participants evaluated the platform across three scenarios. Acceptance was highest for robotics education ($M = 3.96$ on a scale of 1 “Definitely not” to 5 “Definitely yes”) and lowest for home entertainment ($M = 3.59$), where 57% expressed negative expectations about replacing screen-based media. These findings made it clear that acceptability depends on the social role the robot fills. An expert session with a FIDE Master further showed that the LLM-based commentary lacked sufficient depth for expert-level chess. Although the LLM solved the language-fluency challenge (see e.g., Wallkötter et al., 2021) that is important for explainable embodied agents, the content of the commentary lacked the strategic understanding of the game, i.e., it projected confidence that exceeded the model’s actual understanding of the game. This illustrates a form of response misalignment: verbal fluency masks a lack of domain reasoning. Together, these results answer Research question 3: *the chess robot serves as a testbed for human-AI collaboration research, and users valued AI-generated verbal interaction in educational contexts but found it strategically shallow at the expert level.*

6.2.4 Chapter 5: Educating future engineers about LLMs: A scalable workshop

Chapter 5 focused on the processing stage ‘cognition’: the human’s ability to recognize AI reasoning limitations and compensate accordingly. The premise was that, as LLMs become more capable, the human needs to do less of the reasoning, whereas with less capable models, the human must provide more guidance, hints, or even solve parts of the task directly. A key learning objective for the future is therefore for engineering students to recognize how much of the cognitive work they need to do themselves and how much they can delegate to an LLM, a skill that will only grow in importance as LLMs continue to evolve.

In ten workshop sessions, 226 MSc engineering students prompted a non-reasoning LLM (Gemini 2.0 Flash-Lite) and a reasoning LLM (Gemini 2.5 Pro) to solve grid navigation tasks of

increasing complexity, with plans executed in simulation and on a Boston Dynamics Spot robot. The study engaged the cognition stage with both spatial and verbal codes: students formulated verbal prompts encoding spatial goals and evaluated spatial plans.

The results showed that the reasoning model achieved 56% penalty-free solutions on easy tasks versus 19% for the non-reasoning model, but declined to 22% on the hardest task, consistent with findings that LLM reasoning quality degrades with complexity (Shojaee et al., 2025). Students changed their prompting accordingly: 49% used step-by-step instructions for the non-reasoning model, while on the hardest task with the reasoning model only 3% did so; 37% instead provided waypoints and 23% strategic heuristics.

Post-workshop data ($n = 200$) showed that 91.0% of participants reported high engagement and 81.5% reported substantial learning, consistent with research on ‘active learning’ in the engineering education domain (Bonwell & Eison, 1991; Freeman et al., 2014; Prince, 2004). Our gamification approach was used to facilitate learning through game-based incentives (Morales-Trujillo & García-Mireles, 2020; Sailer & Homner, 2020). Chapter 5 demonstrated that misalignment can represent a learning opportunity: experiencing LLM boundaries in a scorable task taught students to redistribute cognitive load, reflecting the ‘complementary’ aim of the overarching research question. This answers Research question 4: *through such a scalable, gamified workshop, students could recognize LLM reasoning limitations and redistribute cognitive effort accordingly, as evidenced by their adaptive prompting strategies and high self-reported learning.*

6.2.5 Synthesis and misalignment as a unifying construct

The four chapters studied the alignment problem from perception to integrative cognition, and revealed that misalignment can take place in different forms, at each information-processing stage. Chapter 2 revealed perceptual misalignment: outcome-level alignment coexisted with divergent visual encoding mechanisms. Chapter 3 revealed spatial cognition-response misalignment: stimulus-response incompatibility where gesture control interfered with visual monitoring more than voice did. Chapter 4 revealed verbal response misalignment: surface-level verbal coherence masked shallow reasoning. Chapter 5 revealed cognitive misalignment: human and AI reasoning processes differed on the same planning tasks: humans relied on spatial intuition such as heuristics and waypoints, while the LLM relied on token-level sequential generation; and students initially failed to compensate for this difference in their prompts.

A number of overarching observations can be made across these forms. First, *outcome-level metrics alone are insufficient for detecting misalignment*: high correlations in Chapter 2 masked different search strategies, fluent chess commentary in Chapter 4 concealed shallow reasoning, and penalty-free rates in Chapter 5 obscured the fact that the reasoning model’s performance degraded sharply with complexity while failure modes changed from collisions to planning breakdowns.

Second, *as the robot/AI system becomes more automated, the human role changes from operator to supervisor*: from directly commanding every movement (Chapter 3), through

evaluating automated perception and strategy (Chapter 4), to diagnosing and scaffolding complete LLM-generated plans (Chapter 5). This progression points to the need for educational interventions that prepare users for the supervisory role by making robot/AI failure modes visible. In addition, since humans and MLLMs can differ in how they perceive the same scene (Chapter 2), predicting where the AI fails can be difficult for human supervisors. This thesis illustrates how the transition from manual control to supervisory control, a concept historically studied extensively in tightly controlled domains such as control rooms and aviation (e.g., Carvalho et al., 2008; Sarter & Woods, 1995), now reveals itself in modern LLM-driven human-robot interaction.

Third, *the quality of human-robot collaboration depends on how cognition is distributed, not on the absolute capability of either partner*. Voice control in Chapter 3 freed spatial resources while gestures overloaded them; Chapter 5 extended this principle to the reasoning level, where students redistributed cognitive work based on model type and task difficulty. Chapter 2 revealed the risk when distribution is misaligned at the perceptual stage: if a supervisor assumes the AI perceives as they do, they may fail to verify the conditions under which the model errs.

These observations complement classical frameworks of human-machine interaction. The Fitts list (Fitts, 1951), Moravec's paradox (Moravec, 1988), and Sheridan and Verplank's (1978) levels-of-automation framework all assumed static allocation of capabilities between human and machine. Later refinements allowed adjustment along levels and stages of automation (Parasuraman et al., 2000). *The present thesis shows that static allocation is inadequate for LLM-driven robots*: LLM capabilities are context-dependent and partially unpredictable (Chapters 2 and 5), and the human mental model of AI capability is dynamic and often initially miscalibrated (Chapter 5). Function allocation therefore has to be dynamic and learned in real time. Unlike classical adaptive automation, in which the system adjusts its own automation level in response to operator state (Parasuraman, 2001; Kaber & Endsley, 2004), the present findings call for operator-led task allocation (also called adaptable automation; Sauer et al., 2012): the human actively redistributes cognitive work based on an evolving model of AI competence. In Chapter 5, students fulfilled this role, redistributing cognitive work according to observed model performance and task complexity.

The above observations culminate in an answer to the central question posed in Section 1.1: *where do the processes of humans and AI-driven robots align, where do they diverge, and how can interaction be designed so that those differences become complementary?* Human and AI-driven robot processes **align at the level of outcomes yet diverge at the level of underlying mechanisms**, and this divergence can emerge at every information-processing stage as well as at the interface that couples the two agents. The differences **become complementary when function allocation is dynamic and operator-led**: the human redistributes cognitive work based on an evolving model of AI competence as task demands and observed AI performance change.

A further consideration is that a transparency asymmetry exists: humans can introspect on their limitations and signal cognitive overload, whereas current LLMs can hardly indicate when they operate beyond their competence (Kim et al., 2025; also see Chapter 4 and 5), making misalignment detection largely dependent on the human side. Future research may want to distinguish between misalignment that improved AI capability can resolve (such as binding deficits [Chapter 2] or shallow chess commentary [Chapter 4]) and structural misalignment, such as spatial reference-frame conflicts (Chapter 3) or the inherent opacity of LLM reasoning.

6.3 Practical contributions

Apart from the scientific findings described above, this thesis produced practical outputs for reuse by researchers and practitioners. Each chapter delivered open tools, datasets, or educational materials. These include a psychophysical benchmark and large-scale human dataset (Chapter 2), an open-source mixed-reality control pipeline (Chapter 3), a reproducible chess robot with open hardware designs (Chapter 4), and a scalable workshop package with curriculum materials (Chapter 5). The four chapters enable other groups to replicate or build upon this work.

Chapter 2 produced a benchmark of 6,300 synthetic visual search images across nine conditions grounded in visual search theory (Treisman & Gelade, 1980; Wolfe, 2021), extending the CLEVR framework (Johnson et al., 2017). The human dataset of 157,042 trials from 1,250 participants is one of the largest psychophysical visual search datasets to date. The benchmark enables standardized comparison of any MLLM against human perceptual performance, with relevance to tasks such as robotic inspection and assistive vision applications.

Chapter 3 contributed an open-source pipeline for voice and gesture control of the Boston Dynamics Spot via the HoloLens 2, including a custom SVM-based gesture recognition module and ROS-based communication architecture. The study also yielded design guidelines for multimodal robot control, such as preferring voice when egocentric spatial alignment is unavailable and supporting physical co-location for gesture interfaces. Methodologically, Chapter 3 introduced LLM-based content analysis of participant interview transcripts using ChatGPT o1 pro mode, validated against manual coding ($r = 0.98$). This approach was subsequently adopted in Chapters 4 and 5 for analyzing open-ended survey responses and student prompts, respectively, forming a methodological thread across the thesis. Code and data are publicly available.

Chapter 4 produced the *OpenChessRobot*, the first open-source chess robot designed for reproducible HRI research. Source code, 3D-printed gripper designs, trained models, and data are publicly available. Reproducibility remains a challenge in HRI (Gunes et al., 2022), and open hardware enables cumulative progress in robotics (Patel et al., 2023). Without shared platforms, researchers have to rebuild experimental setups from scratch, which makes it difficult to compare findings across studies or build on prior work. Open-source systems enable other research teams to replicate experiments under comparable conditions. The modular design allows component substitution without redesigning the system.

Chapter 5 produced a reusable workshop package: a web-based platform, simulation environment, gamification framework, instructor guides, and a three-task curriculum, all openly available. Delivery to 226 students at \$23 total API cost demonstrated scalability. As LLMs become standard engineering tools, curricula that teach their capabilities and limitations, not only their use, will be essential (Kasneci et al., 2023; Lo, 2023).

In addition to producing functional prototypes, the four studies collectively yielded a number of viable design principles. First, designers should use modality switching to offload overloaded cognitive resource pools: Chapter 3 showed that shifting from gesture to voice freed the operator's spatial resources, reducing left-right errors. Second, interfaces should expose AI confidence/uncertainty rather than only outcomes: Chapters 2 and 4 showed that outcome-level metrics and verbal fluency can mask underlying failures, so displaying mechanism-level information (e.g., feature-binding difficulty, reasoning depth) would help operators anticipate where the AI is likely to fail. Third, calibration training should occur before deploying the robot: Chapter 5 demonstrated that even a short, gamified intervention shifted students from over-reliance to adaptive task allocation, suggesting that structured exposure to AI failure modes can build realistic expectations. Fourth, function allocation should be designed as dynamic and operator-led rather than fixed a priori: because LLM capabilities are context-dependent and partially unpredictable (Chapters 2 and 5), the human needs both the authority and the information to redistribute cognitive work in real time as task demands and AI performance change.

6.4 Limitations

This thesis has several limitations. Firstly, this study focused heavily on the visual processing modality; several other processing modalities are unexplored in this thesis. In particular, the haptic modality was not investigated, despite its growing importance in tasks such as teleoperation (Popken et al., 2023; Pacchierotti & Prattichizzo, 2024), dexterous manipulation (Kappassov et al., 2015; Zhao et al., 2025), assistive care (Nanavati et al., 2024), collaborative manipulation (Mol et al., 2026), and humanoid robotics in general (Sanctuary AI, 2025). The auditory modality was used in this thesis (see Chapter 3) but not systematically studied. For example, this thesis did not examine how prosody, timing, or turn-taking in robot speech affects cognitive alignment, though these factors are known to affect conversational grounding (Mavridis, 2015; Skantze, 2021).

The thesis also assumed modularized and verifiable robot cognitive stages. End-to-end perception-to-execution architectures such as vision-language-action models (Black et al., 2025; Brohan et al., 2023) fuse these stages into a single pipeline with non-interpretable intermediate representations, challenging the stage-based categorization adopted in this thesis. Rapid model improvements may reduce the bottlenecks documented here (feature binding, spatial planning), but structural differences between human and AI cognition are unlikely to disappear.

Another limitation of this thesis is that each study used a specific domain and participant population. Chapter 2 used synthetic arrays with geometric shapes (squares and cubes), which is rather far from the rich scenes robots encounter in practice. Chapter 3 studied one robot (Spot), one device (HoloLens 2), and a young and technically educated sample (mean age 23.5 years). Chapter 4 focused on chess, a complete-information domain, unlike typical HRI where information is partial. Chapter 5 again recruited engineering students at the same university, a population likely above average in AI literacy.

The progression from controlled online benchmarking (Chapter 2) through a laboratory experiment (Chapter 3) and an international survey combined with expert gameplay (Chapter 4) to ecological deployment in a university course (Chapter 5) gradually sacrificed internal validity for increased ecological validity. More specifically, Chapter 2 allowed for precise parameter manipulation but used artificial stimuli. Chapter 3 introduced embodied interaction but involved a relatively simple trajectory for the robot. Chapter 4 gained breadth through an international survey but did not feature a longitudinal design. Chapter 5 achieved full realism within an actual university course but could not assess long-term retention of skills or knowledge. The four studies therefore differ in measures, populations, and control, making the synthesis in Section 6.2.5 primarily qualitative.

In summary, whether misalignment transfers to more complex, unstructured environments (e.g., retail, outdoor, home), other HRI domains (e.g., healthcare, manufacturing, domestic robotics) and populations (older adults, non-experts, non-engineers) requires further investigation. Therefore, it should be acknowledged that the claim of misalignment occurring at any MRT stage is domain-general in principle but validated only within these domains.

6.5 Recommendations for future research

The limitations above suggest several directions for future research. Systematic studies of haptic and auditory modalities could examine whether tactile feedback reduces the visual-spatial overload in Chapter 3 (Culbertson et al., 2018; Wang et al., 2024) and how prosody and turn-taking in robot speech affect conversational grounding (Skantze, 2021).

The detection of pieces by the chess robot (see Chapter 4) relied on standard computer vision techniques. In a follow-up study by an MSc student whom the author of this dissertation co-supervised (Waterhout, 2024), more advanced pose estimation methods for the chess pieces were used, together with synthetic training data (3D chess pieces rendered via Blender, a similar approach to that used for generating the 3D shapes in Chapter 2). As AI has improved, MLLMs have become capable of recognizing complex and ill-defined objects, such as “a cube in a cluttered environment” (Chapter 2) or a “can of tomatoes” (as demonstrated in Chapter 5). Another MSc student (Verbaan, 2024), co-supervised by the author of this dissertation, developed a system that used an MLLM to interpret camera images of a tabletop scene, identify manipulable objects, and decompose natural language user instructions into executable action sequences for the Franka Emika robotic arm. Chapter 5 took this further, by combining MLLM-driven perception with navigation in a single pipeline. Future work should extend this line

of research to more demanding real-world settings, with systematic investigation of long-tail error patterns.

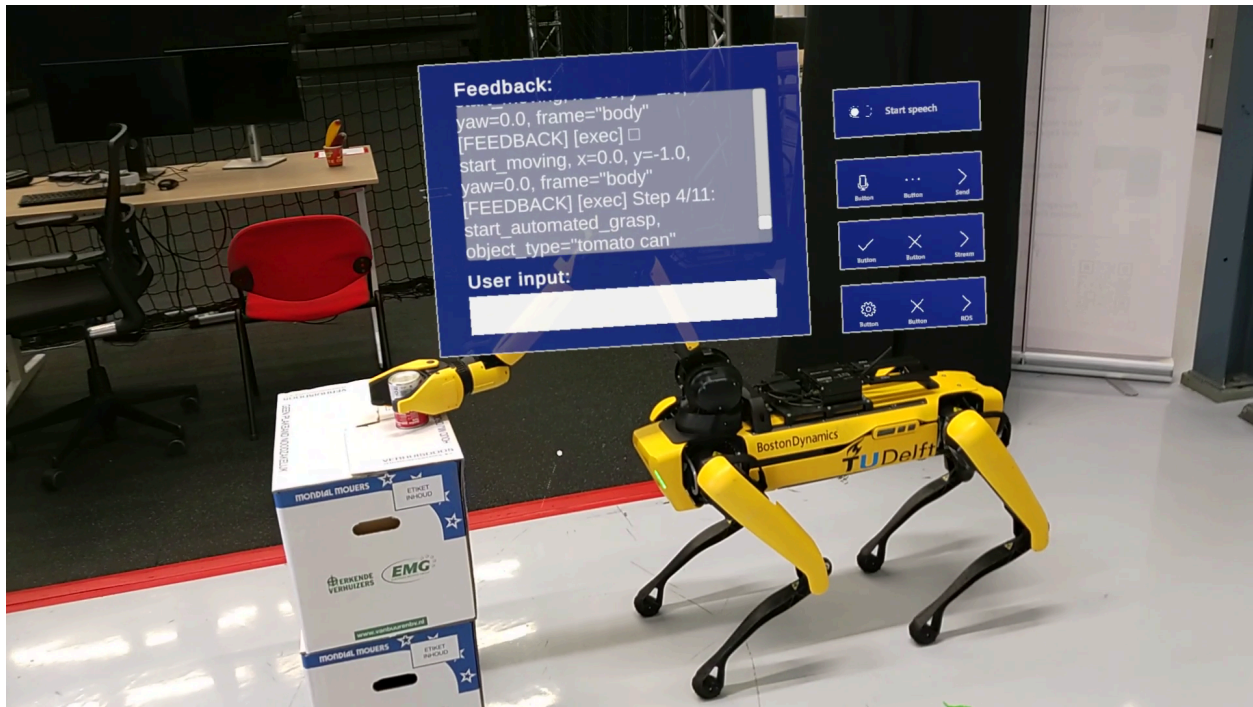


Figure 6.1. Prototype head-mounted mixed-reality interface combining verbal commands, LLM-generated planning feedback, and holographic overlays of robot execution. The user uttered the command “Pick up the tomato can” and is monitoring the robot’s execution progress.

Apart from improvements on the robot side, future work should also augment the human operator. Chapter 3 used a HoloLens mixed reality device to provide visual overlays of issued commands. Additionally, in a prototype of the setup in Chapter 5 (see Figure 6.1), we extended the HoloLens setup so that it combines verbal input, LLM verbal feedback, and holographic display of robot feedback, creating an interaction mode comparable to the simulation interface in Chapter 5 but with real-time visual overlays. Augmented reality also allows for testing virtual robots or setups in an immersive manner before deploying the same robots or setups in the real world. For example, the author co-conducted a study that used a mixed-reality device to test different ways of displaying machine-state information on large virtual screens (Tabone et al., 2026). As wearable augmented reality devices become more accessible, such augmentation could help operators calibrate their mental model of robot capabilities by making robot perception and planning visible (Wang et al., 2026), to better address the misalignment documented across this thesis.

Furthermore, the neurosymbolic approach of Chapter 4, coupling a symbolic problem solver with an LLM for explanation, is a promising direction for explainable AI (e.g., an AI-based chess tutor that can explain why it made a particular move). High-quality chess reasoning requires move accuracy, which modern chess engines achieve through decades of developments in tree-search-based methods such as Stockfish. However, engine evaluations do not directly

translate to explanation quality from a learner's perspective, and collecting expert human feedback is substantially harder than the preference labelling used for general-purpose LLMs. Advances in LLM capabilities may partially address this (cf. Kolasani et al., 2025), but domain-expert evaluation data or pipelines are still needed to calibrate LLMs that genuinely explain strategic reasoning rather than produce superficially fluent but strategically shallow verbal commentary.

The stage-based framework as used in the Introduction of this thesis could be extended or complemented with elements suited to human-AI resource sharing. Shared mental model theory for human-autonomy teams (Demir et al., 2020) and interdependence analysis (Johnson et al., 2014) may offer such complementary elements. Another possible extension is modeling how humans update their mental models of AI capability over extended interaction, drawing on Bayesian trust models (Guo & Yang, 2021) or computational theory of mind (Rabinowitz et al., 2018).

As foundation models improve in feature binding and spatial reasoning, research should track whether misalignment is actually resolved, or just shifted to subsequent processing stages. Longitudinal studies comparing the same tasks across model generations would be needed to provide empirical data to test the robustness of our findings. Additionally, it is recommended that the misalignment concept be tested in applied HRI domains other than education (such as healthcare and manufacturing), and with more diverse populations. Finally, the practical contributions point toward valorisation opportunities: the gamified workshop could be adapted for professional training where misalignment stakes are higher, and the open-source platforms facilitate such translational work.

6.6 Broader reflections and outlook

Across the studies, the human's cognitive role changes: from direct teleoperation (Chapter 3), through a robot serving as opponent, teacher, and partner (Chapter 4, where participants valued it most in educational roles), to supervision (goal-setting and verification) (Chapter 5). As AI capability improves in the future, this transition will likely accelerate toward even more supervisory control and less direct control. In other words, as AI assumes more perception and cognition, the human role moves to meta-cognitive monitoring. Chapter 5 showed this empirically: students offloaded spatial-verbal planning to the AI and reallocated resources to verification. Whether this enhances system performance depends on humans maintaining situation awareness to detect AI failures.

The above implications extend to education and the labor market. Future engineers need training not only in AI tools but also in maintaining cognitive engagement during AI-assisted problem-solving, detecting reasoning failures, and knowing when to override. Productivity gains from AI-driven robots will not be distributed equally: workers with automatable tasks may face displacement (Frey & Osborne, 2017; Acemoglu & Restrepo, 2020), while those who can supervise and calibrate AI are more likely to benefit. The dividing line may be the ability to recognize and manage misalignment, a competence not yet systematically taught in current

education. It can be expected that in the future, detecting errors, hallucinations, or even sabotage by highly advanced AI will be a challenging but critically important task for humanity.

The studies in this thesis revealed several risks when AI takes on cognitive roles. MLLMs produced confident but incorrect visual search output (Chapter 2), the reasoning model generated hallucinated navigation plans and therefore would require an extra safety layer for avoiding collisions in real-life applications (Chapter 5; Ravichandran et al., 2026), and the commentary of the chess robot projected confidence exceeding its understanding (Chapter 4). Current LLMs do not reliably signal when they operate beyond their competence, placing the cost of opacity on the human. Trust calibration (Lee & See, 2004; Parasuraman & Riley, 1997; Guo & Yang, 2021) becomes increasingly important and difficult when AI competence boundaries are not transparent. Students initially overestimated LLM capabilities (Chapter 5), and in safety-critical domains, users may not have the opportunity to learn AI competence boundaries through training or repeated use (Tomsett et al., 2020).

Based on this evidence, it is argued that meaningful human control is necessary. If AI development disregards human authority and understanding of the machine, the misalignment documented in Chapters 2 through 5 would be seen as irrelevant rather than a problem to solve. That framing might yield short-term efficiency but would remove the feedback loop through which humans detect errors and maintain accountability. Human oversight should therefore be designed into systems through transparent interfaces that display uncertainty, training that builds calibrated trust (Chapter 5), and institutional accountability. The EU AI Act (European Parliament, 2024) and trustworthy AI standards (Floridi et al., 2018; Jobin et al., 2019) provide a regulatory structure, but translating these into interaction designs is still an open challenge. The trajectory from direct control (Chapter 3) to goal-setting (Chapter 5) showed that human work will change from performing tasks to supervising them. This need not mean unemployment, but the benefits of human-AI collaboration may go primarily to those trained to manage it, while others become passive recipients of AI decisions. Education (Chapter 5), policy, professional standards, and technology design have to work together.

The empirical research in this thesis represents an increment in understanding human-AI collaboration, but the implications extend beyond the domains studied in this thesis. Misalignment occurs even when aggregate AI performance matches humans (Chapter 2), which indicates that benchmark-only evaluation can be misleading. Interface design affects misalignment independently of agent capability (Chapter 3), which suggests that regulatory attention should include interaction design. Short educational interventions can improve AI calibration (Chapter 5), a finding that supports investment in AI literacy.

This thesis was itself shaped by the rapid evolution it studies. At the start of this PhD in mid-2022, large generative models had not yet entered mainstream use. The release of ChatGPT five months later prompted the LLM-based commentary in the chess robot (Chapter 4, the earliest study) and raised the multimodal interaction questions behind Chapters 2 and 3. The emergence of reasoning-capable LLMs in 2024–2025 inspired a vision-to-plan pipeline, but early experiments revealed perception as the bottleneck, redirecting the work toward Chapter 2

and informing Chapter 5. LLMs also transformed how the author conducted this research, from writing support to co-coding, making the judgment of when to rely on AI and when to override it (also see Chapter 5) a daily practice rather than an abstract research topic.

6.7 Conclusions

This thesis investigated where human and AI-driven robot cognition aligns, where it misaligns, and how interaction can make differences complementary. Four empirical studies with a total of 2,292 participants, spanning the perceptual, cognitive, and response stages of information-processing, revealed that misalignment does occur and is consequential. Agents that appear to perceive the same scene fail under different task conditions, the human-robot interface determines where cognitive conflicts arise, surface coherence in AI outputs can conceal shallow reasoning, and humans initially miscalibrate their expectations of AI capability. The overarching insight is that function allocation between human and AI in complex environments has to be continuously recalibrated through interaction.

On the design side, this thesis recommends a number of principles: use modality switching to offload overloaded processing codes, expose AI confidence and uncertainty rather than only outcomes, provide calibration training before deployment of the robot, and design human vs. AI function allocation as dynamic and operator-led. On the research side, the recommendations are to extend the investigation to unexplored processing modalities (haptic, auditory), test human-robot misalignment in other applied domains, and scale educational approaches that build the cognitive competence for human-AI collaboration.

As AI-driven robots assume increasingly diverse roles, the human contribution will change from direct control to supervision. Whether this transition strengthens or weakens human agency depends on investment in transparent interaction design, calibrated trust, and training and access to skills for effective human-AI collaboration. The path from ‘misalignment’ to ‘collaboration’ requires deliberate design, empirical validation, and ongoing effort.

References

- Acemoglu, D., & Restrepo, P. (2020). Robots and jobs: Evidence from US labor markets. *Journal of Political Economy*, 128(6), 2188–2244. <https://doi.org/10.1086/705716>
- Acemoglu, D., & Restrepo, P. (2022). Demographics and automation. *The Review of Economic Studies*, 89(1), 1–44. <https://doi.org/10.1093/restud/rdab031>
- Ahn, M., Brohan, A., Brown, N., Chebotar, Y., Cortes, O., David, B., Finn, C., Fu, C., Gopalakrishnan, K., Hausman, K., Herzog, A., Ho, D., Hsu, J., Ibarz, J., Ichter, B., Irpan, A., Jang, E., Jauregui Ruano, R., Jeffrey, K., ... Zeng, A. (2023). Do as I can, not as I say: Grounding language in robotic affordances. In K. Liu, D. Kulic, & J. Ichnowski (Eds.), *Proceedings of the 6th Conference on Robot Learning* (Vol 205, pp. 287–318). PMLR. <https://proceedings.mlr.press/v205/ichter23a/ichter23a.pdf>
- Black, K., Brown, N., Driess, D., Esmail, A., Equi, M., Finn, C., Fusai, N., Groom, L., Hausman, K., Ichter, B., Jakubczak, S., Jones, T., Ke, L., Levine, S., Li-Bell, A., Mothukuri, M., Nair, S., Pertsch, K., Shi, L., ... Zhilinsky, U. (2025). π_0 : A vision-language-action flow model for general robot control. *Robotics: Science and Systems XXI*, Los Angeles, CA. <https://doi.org/10.15607/rss.2025.xxi.010>
- Bonwell, C. C., & Eison, J. A. (1991). *Active learning: Creating excitement in the classroom* (ASHE-ERIC Higher Education Report No. 1). ERIC Clearinghouse on Higher Education. <https://files.eric.ed.gov/fulltext/ED336049.pdf>

- Brohan, A., Brown, N., Carbajal, J., Chebotar, Y., Chen, X., Choromanski, K., Ding, T., Driess, D., Dubey, A., Finn, C., Florence, P., Fu, C., Arenas, M. G., Gopalakrishnan, K., Han, K., Hausman, K., Herzog, A., Hsu, J., Ichter, B., ... Zeng, A. (2023). RT-2: Vision-language-action models transfer web knowledge to robotic control. In J. Tan, M. Toussaint, & K. Darvish (Eds.), *Proceedings of the 7th Conference on Robot Learning* (Vol 229, pp. 2165–2183). PMLR. <https://proceedings.mlr.press/v229/zitkovich23a/zitkovich23a.pdf>
- Campbell, M., Hoane, A. J., Jr., & Hsu, F. (2002). Deep Blue. *Artificial Intelligence*, 134(1–2), 57–83. [https://doi.org/10.1016/S0004-3702\(01\)00129-1](https://doi.org/10.1016/S0004-3702(01)00129-1)
- Campbell, D., Rane, S., Giallanza, T., De Sabbata, N., Ghods, K., Joshi, A., Ku, A., Frankland, S. M., Griffiths, T. L., Cohen, J. D., & Webb, T. (2024). Understanding the limits of vision language models through the lens of the binding problem. *Advances in Neural Information Processing Systems* 37 (pp. 113436–113460). NeurIPS. <https://doi.org/10.52202/079017-3604>
- Carvalho, P. V., dos Santos, I. L., Gomes, J. O., Borges, M. R., & Guerlain, S. (2008). Human factors approach for evaluation and redesign of human–system interfaces of a nuclear power plant simulator. *Displays*, 29(3), 273–284. <https://doi.org/10.1016/j.displa.2007.08.010>
- Culbertson, H., Schorr, S. B., & Okamura, A. M. (2018). Haptics: The present and future of artificial touch sensation. *Annual Review of Control, Robotics, and Autonomous Systems*, 1, 385–409. <https://doi.org/10.1146/annurev-control-060117-105043>
- Demir, M., McNeese, N. J., & Cooke, N. J. (2020). Understanding human-robot teams in light of all-human teams: Aspects of team interaction and shared cognition. *International Journal of Human-Computer Studies*, 140, Article 102436. <https://doi.org/10.1016/j.ijhcs.2020.102436>
- Driess, D., Xia, F., Sajjadi, M. S. M., Lynch, C., Chowdhery, A., Ichter, B., Wahid, A., Tompson, J., Vuong, Q., Yu, T., Huang, W., Chebotar, Y., Sermanet, P., Duckworth, D., Levine, S., Vanhoucke, V., Hausman, K., Toussaint, M., Greff, K., Zeng, A., Mordatch, I., & Florence, P. (2023). PaLM-E: An embodied multimodal language model. *Proceedings of the 40th International Conference on Machine Learning*, 202 (pp. 8469–8488). PMLR. <https://proceedings.mlr.press/v202/driess23a/driess23a.pdf>
- European Parliament. (2024). Regulation (EU) 2024/1689 of the European Parliament and of the Council laying down harmonised rules on artificial intelligence (Artificial Intelligence Act). *Official Journal of the European Union*, L 2024/1689. <https://eur-lex.europa.eu/eli/reg/2024/1689/oj/en>
- Firooz, R., Tucker, J., Tian, S., Majumdar, A., Sun, J., Liu, W., ... Schwager, M. (2025). Foundation models in robotics: Applications, challenges, and the future. *The International Journal of Robotics Research*, 44(5), 701–739. <https://doi.org/10.1177/02783649241281508>
- Fitts, P. M. (Ed.). (1951). *Human engineering for an effective air-navigation and traffic-control system*. National Research Council, Division of Anthropology and Psychology. <https://apps.dtic.mil/sti/tr/pdf/ADB815893.pdf>
- Fitts, P. M., & Seeger, C. M. (1953). S-R compatibility: Spatial characteristics of stimulus and response codes. *Journal of Experimental Psychology*, 46(3), 199–210. <https://doi.org/10.1037/h0062827>
- Floridi, L., Cowls, J., Beltrametti, M., Chatila, R., Chazerand, P., Dignum, V., Luetge, C., Madelin, R., Pagallo, U., Rossi, F., Schafer, B., Valcke, P., & Vayena, E. (2018). AI4People: An ethical framework for a good AI society. *Minds and Machines*, 28, 689–707. <https://doi.org/10.1007/s11023-018-9482-5>
- Freeman, S., Eddy, S. L., McDonough, M., Smith, M. K., Okoroafor, N., Jordt, H., & Wenderoth, M. P. (2014). Active learning increases student performance in science, engineering, and mathematics. *Proceedings of the National Academy of Sciences*, 111(23), 8410–8415. <https://doi.org/10.1073/pnas.1319030111>
- Frey, C. B., & Osborne, M. A. (2017). The future of employment: How susceptible are jobs to computerisation? *Technological Forecasting and Social Change*, 114, 254–280. <https://doi.org/10.1016/j.techfore.2016.08.019>
- Gihleb, R., Giuntella, O., Stella, L., & Wang, T. (2022). Industrial robots, workers' safety, and health. *Labour Economics*, 78, Article 102205. <https://doi.org/10.1016/j.labeco.2022.102205>
- Graetz, G., & Michaels, G. (2018). Robots at work. *The Review of Economics and Statistics*, 100(5), 753–768. https://doi.org/10.1162/rest_a_00754
- Gunes, H., Broz, F., Crawford, C. S., Rosenthal-von der Pütten, A., Strait, M., & Riek, L. (2022). Reproducibility in human-robot interaction: Furthering the science of HRI. *Current Robotics Reports*, 3, 281–292. <https://doi.org/10.1007/s43154-022-00094-5>

- Guo, Y., & Yang, X. J. (2021). Modeling and predicting trust dynamics in human-robot teaming: A Bayesian inference approach. *International Journal of Social Robotics*, 13, 1899–1909. <https://doi.org/10.1007/s12369-020-00703-3>
- Jobin, A., Ienca, M., & Vayena, E. (2019). The global landscape of AI ethics guidelines. *Nature Machine Intelligence*, 1, 389–399. <https://doi.org/10.1038/s42256-019-0088-2>
- Johnson, J., Hariharan, B., van der Maaten, L., Fei-Fei, L., Zitnick, C. L., & Girshick, R. (2017). CLEVR: A diagnostic dataset for compositional language and elementary visual reasoning. Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, 1988–1997. <https://doi.org/10.1109/CVPR.2017.215>
- Johnson, M., Bradshaw, J. M., Feltovich, P. J., Jonker, C. M., van Riemsdijk, M. B., & Sierhuis, M. (2014). Coactive design: Designing support for interdependence in joint activity. *Journal of Human-Robot Interaction*, 3(1), 43–69. <https://doi.org/10.5898/JHRI.3.1.Johnson>
- Kaber, D. B., & Endsley, M. R. (2004). The effects of level of automation and adaptive automation on human performance, situation awareness and workload in a dynamic control task. *Theoretical Issues in Ergonomics Science*, 5(2), 113–153. <https://doi.org/10.1080/1463922021000054335>
- Kappassov, Z., Corrales, J. A., & Perdureau, V. (2015). Tactile sensing in dexterous robot hands. *Robotics and Autonomous Systems*, 74, 195–220. <https://doi.org/10.1016/j.robot.2015.07.015>
- Kasneci, E., Sessler, K., Küchemann, S., Bannert, M., Dementieva, D., Fischer, F., Gasser, U., Groh, G., Günemann, S., Hüllermeier, E., Krusche, S., Kutyniok, G., Michaeli, T., Nerdel, C., Pfeffer, J., Poquet, O., Sailer, M., Schmidt, A., Seidel, T., Stadler, M., Weller, J., Kuhn, J., & Kasneci, G. (2023). ChatGPT for good? On opportunities and challenges of large language models for education. *Learning and Individual Differences*, 103, Article 102274. <https://doi.org/10.1016/j.lindif.2023.102274>
- Kim, S. S. Y., Wortman Vaughan, J., Liao, Q. V., Lombrozo, T., & Russakovsky, O. (2025). Fostering appropriate reliance on large language models: The role of explanations, sources, and inconsistencies. *Proceedings of the 2025 CHI Conference on Human Factors in Computing Systems*, Article 420. ACM. <https://doi.org/10.1145/3706598.3714020>
- Kolasani, S., Saplin, M., Crispino, N., Montgomery, K., Davis, J. Q., Zaharia, M., Wang, C., & Wang, C. (2025). LLM CHESS: Benchmarking reasoning and instruction-following in LLMs through chess [Paper presentation]. First Workshop on the Foundations of Reasoning in Language Models (FoRLM) at NeurIPS 2025, San Diego, CA, United States. <https://openreview.net/forum?id=i0UHRCKEXZ>
- Lee, J. D., & See, K. A. (2004). Trust in automation: Designing for appropriate reliance. *Human Factors*, 46(1), 50–80. https://journals.sagepub.com/doi/10.1518/hfes.46.1.50_30392
- Lo, C. K. (2023). What is the impact of ChatGPT on education? A rapid review of the literature. *Education Sciences*, 13(4), Article 410. <https://doi.org/10.3390/educsci13040410>
- Mavridis, N. (2015). A review of verbal and non-verbal human-robot interactive communication. *Robotics and Autonomous Systems*, 63, 22–35. <https://doi.org/10.1016/j.robot.2014.09.031>
- McCarthy, J. (1990). Chess as the Drosophila of AI. In T. A. Marsland & J. Schaeffer (Eds.), *Computers, chess, and cognition* (pp. 227–237). Springer. https://doi.org/10.1007/978-1-4613-9080-0_14
- Mehrani, P., & Tsotsos, J. K. (2023). Self-attention in vision transformers performs perceptual grouping, not attention. *Frontiers in Computer Science*, 5, Article 1178450. <https://doi.org/10.3389/fcomp.2023.1178450>
- Mol, N., Prendergast, J. M., Abbink, D. A., & Peternel, L. (2026). Fitts' list revisited: An empirical study on function allocation in a two-agent physical human-robot collaborative position/force task. *IEEE Robotics and Automation Letters*, 11(1), 202–209. <https://doi.org/10.1109/LRA.2025.3632607>
- Morales-Trujillo, M. E., & García-Mireles, G. A. (2020). Gamification and SQL: An empirical study on student performance in a database course. *ACM Transactions on Computing Education*, 21(1), Article 3. <https://doi.org/10.1145/3427597>
- Moravec, H. (1988). *Mind children*. Harvard University Press. <https://dl.acm.org/doi/abs/10.5555/48030>
- Nanavati, A., Ranganeni, V., & Cakmak, M. (2024). Physically assistive robots: A systematic review of mobile and manipulator robots that physically assist people with disabilities. *Annual Review of Control, Robotics, and Autonomous Systems*, 7, 123–147. <https://doi.org/10.1146/annurev-control-062823-024352>
- Pacchierotti, C., & Prattichizzo, D. (2024). Cutaneous/tactile haptic feedback in robotic teleoperation: Motivation, survey, and perspectives. *IEEE Transactions on Robotics*, 40, 978–998. <https://doi.org/10.1109/TRO.2023.3344027>

- Parasuraman, R. (2001). Adaptive automation: From theory to practice. *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*, 45(4), 291–292. <https://doi.org/10.1177/154193120104500405>
- Parasuraman, R., & Riley, V. (1997). Humans and automation: Use, misuse, disuse, abuse. *Human Factors*, 39(2), 230–253. <https://doi.org/10.1518/001872097778543886>
- Parasuraman, R., Sheridan, T. B., & Wickens, C. D. (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), 286–297. <https://doi.org/10.1109/3468.844354>
- Patel, V. V., Liarakapis, M. V., & Dollar, A. M. (2023). Open robot hardware: Progress, benefits, challenges, and best practices. *IEEE Robotics & Automation Magazine*, 30(3), 123–148. <https://doi.org/10.1109/MRA.2022.3225725>
- Popken, M., Prendergast, J. M., Wiertelowski, M., & Peternel, L. (2023). An adaptive semi-autonomous impedance controller for teleoperated object grasping based on human grip safety margin. *2023 IEEE-RAS 22nd International Conference on Humanoid Robots (Humanoids 2023)* (pp. 1–8). IEEE. <https://doi.org/10.1109/Humanoids57100.2023.10375144>
- Prince, M. (2004). Does active learning work? A review of the research. *Journal of Engineering Education*, 93(3), 223–231. <https://doi.org/10.1002/j.2168-9830.2004.tb00809.x>
- Rabinowitz, N. C., Perbet, F., Song, F., Zhang, C., Eslami, S. M. A., & Botvinick, M. (2018). Machine theory of mind. In J. Dy & A. Krause (Eds.), *Proceedings of the 35th International Conference on Machine Learning* (Vol 80, pp. 4218–4227). PMLR. <https://proceedings.mlr.press/v80/rabinowitz18a.html>
- Ravichandran, Z., Robey, A., Kumar, V., Pappas, G. J., & Hassani, H. (2026). Safety guardrails for LLM-enabled robots. *IEEE Robotics and Automation Letters*, 11(4), 4649–4656. <https://doi.org/10.1109/LRA.2026.3667488>
- Sailer, M., & Homner, L. (2020). The gamification of learning: A meta-analysis. *Educational Psychology Review*, 32(1), 77–112. <https://doi.org/10.1007/s10648-019-09498-w>
- Sanctuary AI. (2025). *Sanctuary AI equips general purpose robots with new touch sensors for performing highly dexterous tasks*. <https://www.sanctuary.ai/blog/sanctuary-ai-equips-general-purpose-robots>
- Sarter, N. B., & Woods, D. D. (1995). How in the world did we ever get into that mode? Mode error and awareness in supervisory control. *Human Factors*, 37(1), 5–19. <https://doi.org/10.1518/001872095779049516>
- Sauer, J., Kao, C. S., & Wastell, D. (2012). A comparison of adaptive and adaptable automation under different levels of environmental stress. *Ergonomics*, 55(8), 840–853. <https://doi.org/10.1080/00140139.2012.676673>
- Shannon, C. E. (1950). XXII. Programming a computer for playing chess. *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science*, 41(314), 256–275. <https://doi.org/10.1080/14786445008521796>
- Shepard, R. N., & Metzler, J. (1971). Mental rotation of three-dimensional objects. *Science*, 171(3972), 701–703. <https://doi.org/10.1126/science.171.3972.701>
- Sheridan, T. B., & Verplank, W. L. (1978). *Human and computer control of undersea teleoperators*. Massachusetts Institute of Technology, Man-Machine Systems Lab. <https://doi.org/10.21236/ADA057655>
- Shojaee, P., Mirzadeh, I., Alizadeh, K., Horton, M., Bengio, S., & Farajtabar, M. (2025). The illusion of thinking: Understanding the strengths and limitations of reasoning models via the lens of problem complexity. *Advances in Neural Information Processing Systems* 38. <https://openreview.net/forum?id=YghiOusmvw>
- Skantze, G. (2021). Turn-taking in conversational systems and human-robot interaction: A review. *Computer Speech & Language*, 67, Article 101178. <https://doi.org/10.1016/j.csl.2020.101178>
- Tabone, W., De Winter, J., Zhang, R., Wickens, C. D., van de Ruit, M., & Eisma, Y. B. (2026). *Embodied sampling: An augmented reality investigation of visual attention* [Preprint]. ResearchGate. https://www.researchgate.net/profile/Joost-De-Winter/publication/395535610_Embodied_sampling_An_augmented_reality_investigation_of_visual_attention/links/6970c5bc52773b62239c5082/Embodied-sampling-An-augmented-reality-investigation-of-visual-attention.pdf
- Tangtharakul, G., & Storrs, K. R. (2025). Visual language models show widespread visual deficits on neuropsychological tests. *Nature Machine Intelligence*, 8, 209–219. <https://doi.org/10.1038/s42256-026-01179-y>

- Tomsett, R., Preece, A., Braines, D., Cerutti, F., Chakraborty, S., Srivastava, M., Pearson, G., & Kaplan, L. (2020). Rapid trust calibration through interpretable and uncertainty-aware AI. *Patterns*, 1(4), Article 100049. <https://doi.org/10.1016/j.patter.2020.100049>
- Treisman, A. M., & Gelade, G. (1980). A feature-integration theory of attention. *Cognitive Psychology*, 12(1), 97–136. [https://doi.org/10.1016/0010-0285\(80\)90005-5](https://doi.org/10.1016/0010-0285(80)90005-5)
- Verbaan, L. E. (2024). *Perception and control with large language models in robotic manipulation: Developing and assessing an integrated large language model system on environmental and task complexity* [Master's thesis, Delft University of Technology]. TU Delft Repository. <https://resolver.tudelft.nl/uuid:2944a8e1-0a54-492b-af63-193a7ac11db8>
- Wallkötter, S., Tulli, S., Castellano, G., Paiva, A., & Chetouani, M. (2021). Explainable embodied agents through social cues: A review. *ACM Transactions on Human-Robot Interaction*, 10(3), Article 27. <https://doi.org/10.1145/3457188>
- Wang, G., Wang, H. H., & Ren, G. (2024). Visual and haptic guidance for enhancing target search performance in dual-task settings. *Applied Sciences*, 14(11), Article 4650. <https://doi.org/10.3390/app14114650>
- Wang, H., Phatak, R., Ocampo, J., & Han, Z. (2026). Indicating robot vision capabilities with augmented reality. *International Journal of Social Robotics*, 18(3), Article 42. <https://doi.org/10.1007/s12369-026-01368-0>
- Waterhout, M. (2024). *Evaluating 6D pose estimation accuracy with synthetic data: A comparative analysis of RGB-based and RGB-depth-based images in a chess piece picking task with a robotic arm* [Master's thesis, Delft University of Technology]. TU Delft Repository. <https://resolver.tudelft.nl/uuid:4506c969-220d-4257-ad5b-91853ea83ccb>
- Wickens, C. D. (2002). Multiple resources and performance prediction. *Theoretical Issues in Ergonomics Science*, 3(2), 159–177. <https://doi.org/10.1080/14639220210123806>
- Wickens, C. D. (2008). Multiple resources and mental workload. *Human Factors*, 50(3), 449–455. <https://doi.org/10.1518/001872008X288394>
- Wolfe, J. M. (2021). Guided Search 6.0: An updated model of visual search. *Psychonomic Bulletin & Review*, 28(4), 1060–1092. <https://doi.org/10.3758/s13423-020-01859-9>
- Zhao, Z., Li, W., Li, Y., Liu, T., Li, B., Wang, M., ... Zhu, S. C. (2025). Embedding high-resolution touch across robotic hands enables adaptive human-like grasping. *Nature Machine Intelligence*, 7(6), 889–900. <https://doi.org/10.1038/s42256-025-01053-3>

Acknowledgement

Completing this thesis was not an easy journey. Looking back, the people I met along the way were like wind and waves: sometimes pushing, sometimes pulling, but together they carried me to the destination.

Yke, thank you for all your time, from my first day to my last, and especially for the meetings that began with "let's get some fresh air" and ended somewhere along the harbour or in the campus forest. You are bold, full of ideas, and always willing to share them. You gave me the freedom to explore in many directions, and the colourful result in these pages comes from that trust.

Harleigh, thank you for the support that does not appear in any publication. Our weekly meetings gave me a place to think out loud, about the research and about everything around it. You always took my worries seriously, and somehow that made them smaller. Your care made a difference through the harder stretches of this work.

Joost, thank you for your extraordinary dedication. Your feedback often arrived within hours or minutes, sometimes seconds, and always with a level of detail that pushed the work forward. You were never afraid to say when something was not good enough. I learned from you what it means to care deeply about the quality of the work and to keep the project moving even when the path was unclear. Your standards shaped this thesis more than any other single influence.

Dimitra, thank you for your patience and your practical mind. Knowing you were behind me made the work feel safe. You listened carefully before speaking, and your guidance cleared the fog in front of me. I have come to see you and Joost as models of what serious academic work looks like: hard-working, sharp, and generous with your time.

Working with a supervisory team of four was not always simple, but each of you brought something the others did not, and the thesis is better for it.

I would also like to thank the members of my doctoral committee, Prof. dr. ir. J. Hellendoorn, Prof. dr. R. R. Negenborn, Prof. dr. ir. W. P. Brinkman, Prof. dr. S. van der Stigchel, and Dr. ir. J. F. P. Kooij, for the time they spent with this thesis and for the questions that made it better.

Special thanks to **Di Wu**, who introduced me to research during my master's. Without that first push, I would not have started this path at all.

To my first officemates, **Nicky**, **Italo**, and **Zhaochong**: those early lunches and fun discussions made the beginning of the PhD far less lonely than it could have been. **Zhaochong**, thank you especially for the bouldering and the time outdoors. Some of my best memories from these years happened far away from the desk. **Thomas**, we shared a lab, a robot arm, and a few drinks in Delft. You were always kind and always willing to help, and the start of my PhD would have been much harder without you.

Chuhan, Jingyue, Zhaoting, Yuxuan, Meiye, Shiming, Chen Gang, Jiatao, Jihong, and Sihao: thank you for the conversations, the lunches, and the many small moments of fun that made Delft feel like home on the other side of the continent. **Yuxuan**, my back owes you a special debt for dragging me away from my desk for a cup of tea or a walk, no matter how busy I claimed to be.

Jenna and Bianca, I will remember the coffees and lunches in the sun, and conversations that never got boring until we noticed it was time to work again. **Koen**, our cookie friendship made the lab next door feel even closer than it was.

Yiqin and Jingsen, the shared meals, the shared flat, and the friendship behind both. Few things matter more than a home you want to return to.

I am grateful to **André van der Kraan, Kseniia Khomenko**, and **Maurits Pfaff** for their technical support, and to **Hanneke, Loulou, Noortje**, and **Ellen** for guiding me through countless practical matters over the years, always with patience.

Finally, to my father and mother: thank you for your unconditional support from far away. You never asked me to explain what I was doing, only whether I was doing well. And **Junran**, thank you for staying through all of it, the good weeks and the bad ones. You know the true cost of this thesis better than anyone, and you have been counting my lost hair since chapter one.

I am sure I cannot list every name that shaped this journey, but everyone who crossed my path along the way deserves my thanks.

To all of you: thank you for the wind and the waves.

Renchi Zhang

Den Haag, July 2026

Curriculum Vitae

Renchi Zhang

September 1996 Born in Henan, China

Education

2022–2026	Ph.D. in Cognitive Robotics, Delft University of Technology
2019–2021	M.Sc. in Computer Science (<i>cum laude</i>), Leiden University
2021	Visiting student, Katholieke Universiteit Leuven
2014–2018	B.Eng. in Digital Media Technology, Beijing University of Posts and Telecommunications

Work Experience

2018	Internship. BMW China Automotive Trading Ltd., Beijing, China
------	---

List of Publications

Journal Publications

1. **Zhang, R.**, de Winter, J. C. F., Dodou, D., Seyffert, H., & Eisma, Y. B. (in press). Human-like attention? A psychophysical comparison of visual search in humans and MLLMs. *Computational Brain & Behavior*.
2. **Zhang, R.***, van der Linden, J.* , Dodou, D., Seyffert, H., Eisma, Y. B.#, & de Winter, J.# (2025). Walk Along: An Experiment on Controlling the Mobile Robot ‘Spot’ with Voice and Gestures. *ACM Transactions on Human-Robot Interaction*, 14(4), Article 64. <https://doi.org/10.1145/3729540>
3. **Zhang, R.**, de Winter, J., Dodou, D., Seyffert, H., & Eisma, Y. B. (2025). An open-source reproducible chess robot for human-robot interaction research. *Frontiers in Robotics and AI*, 12, Article 1436674. <https://doi.org/10.3389/frobt.2025.1436674>
4. Wu, D.* , **Zhang, R.***, Pore, A., Dall’Alba, D., Ha, X. T., Li, Z., ... & Vander Poorten, E. (2024). A review on machine learning in flexible surgical and interventional robots: Where we are and where we are going. *Biomedical Signal Processing and Control*, 93, Article 106179. <https://doi.org/10.1016/j.bspc.2024.106179>

Under review

1. **Zhang, R.**, de Winter, J., Dicke, T., Dodou, D., & Eisma, Y. B. (2026). Educating future engineers about LLMs: A scalable workshop.
2. Tabone, W., de Winter, J., **Zhang, R.**, Wickens, C. D., van de Ruit, M., & Eisma, Y. B. (2026). Embodied sampling: An augmented reality investigation of visual attention.

Datasets and Repositories

1. **Zhang, R.**, & Dicke, T. (2026). *Code for the paper: Educating future engineers about LLMs: A scalable workshop*. GitHub. <https://github.com/renchizh/LLM-robotics-workshop>
2. **Zhang, R.** (2025). *Code for the paper: Human-like attention? A psychophysical comparison of visual search in humans and MLLMs*. GitHub. <https://github.com/renchizh/MLLM-Visual-Search>
3. **Zhang, R.**, de Winter, J., Dodou, D., Seyffert, H., & Eisma, Y. B. (2025). *Supplementary data for the paper: An open-source reproducible chess robot for human-robot interaction research* (Version 3) [Data set]. 4TU.ResearchData. <https://doi.org/10.4121/1CB5BF64-468E-462A-A82E-C847D88A7A86.V3>
4. **Zhang, R.**, van der Linden, J., Dodou, D., Seyffert, H., Eisma, Y. B., & de Winter, J. (2025). *Supplementary data for the paper: Walk along: An experiment on controlling the mobile robot ‘Spot’ with voice and gestures* (Version 1) [Data set]. 4TU.ResearchData. <https://doi.org/10.4121/96642016-963C-427D-BCD7-BD6D990FC264.V1>

5. **Zhang, R.**, & van der Linden, J. (2024). *Code for the paper: Walk Along: An Experiment on Controlling the Mobile Robot 'Spot' with Voice and Gestures*. GitHub.
<https://github.com/renchizhyyy/Walk-along-Spot>
6. **Zhang, R.** (2024). *Code for the paper: An open-source reproducible chess robot for human-robot interaction research*. GitHub.
<https://github.com/renchizhyyy/OpenChessRobot>

* Equal contribution

Propositions

accompanying the dissertation
Information-processing alignment in human-robot interaction
by
Renchi Zhang

1. LLM benchmarks must probe the process, not just the outcome. (*This proposition pertains to this dissertation*)
2. Freedom to walk restores stimulus-response compatibility. (*This proposition pertains to this dissertation*)
3. Experts see through fluent but shallow reasoning; novices mistake fluency for competence. (*This proposition pertains to this dissertation*)
4. With increasing AI intelligence, effective collaboration means knowing when you are the weaker partner. (*This proposition pertains to this dissertation*)
5. The more tasks AI takes over, the more the bottleneck of productivity moves from human-AI alignment to human-human alignment.
6. A strike in an AI economy is a free pilot test for full automation.
7. LLMs have advanced human-computer interaction more than decades of interface design research.
8. AI will make every academic format obsolete except the proposition.
9. A PhD thesis demands a focused contribution, yet the researcher's most valuable intellectual growth comes from divergent exploration.
10. Confronting the finitude of life is a necessary condition for making bold life choices, such as pursuing a PhD.

These propositions are regarded as opposable and defensible, and have been approved as such by the promoters, Prof. dr. ir. J. C. F. de Winter and Dr. D. Dodou, and the copromoters, Dr. ir. Y. B. Eisma and Dr. H. C. Seyffert.