Neuroergonomic Signatures of Improved Human–Robot Collaboration in LLM-Supported Industrial Workflows

Jose A. Trapero[1*], Yannick Robin[1], Eric Wagner[2], Steffen Knapp[2], Martina Lehser[2], Daniel J. Strauss[1]

[1*]Systems Neuroscience and Neurotechnology Unit, Faculty of Medicine, Saarland University, Homburg/Saar, 66421 & School of Engineering, htw saar, Saarbrücken, 66117, Saarland, Germany.
[2]Embedded Robotics Lab, School of Engineering, htw saar, Saarbrücken, 66117, Saarland, Germany.

*Corresponding author(s). E-mail(s): jose.trapero@uni-saarland.de;
Contributing authors: robin@snn-unit.de; eric.wagner@htwsaar.de;
steffen.knapp@htwsaar.de; martina.lehser@htwsaar.de;
daniel.strauss@uni-saarland.de;

Abstract

The Industry 5.0 is hindered by the communication bottleneck between humans and collaborative robots (cobots), because of the rigid programming requirements, which prevent dynamic and unexpected workflow changes. This study introduces and validates a Large Language Model (LLM)-enabled framework for natural language robot task planning to improve communication between humans and robots while simultaneously reducing the mental load of the worker. Utilizing a resource-efficient one-shot prompt engineering strategy, the system allows operators to adapt cobot trajectories via conversational input. We evaluated this dynamic workflow against a traditional static baseline in a simulated assembly environment, employing a comprehensive multimodal neuroergonomic assessment. Results demonstrate that the LLM framework achieved high technical efficacy, generating executable code within 1–2 prompts. Crucially, the combination of physiological metrics (heart rate, blink rate), behavioral data (task error rate), and subjective workload (NASA Task Load Index) revealed a significant reduction in operator strain and an increase in process reliability in the LLM-assisted condition. These findings provide empirical evidence that LLMs can democratize cobot programming while improving the physiological

well-being of the workforce, thereby compelling a human-centric approach to flexible automation.

Keywords: Industry 5.0, Human-centric, Large language models (LLMs), Collaborative robot, Intelligent collaboration

1 Introduction

The transition from Industry 4.0 (I4.0) to Industry 5.0 (I5.0) represents a fundamental realignment of industrial values, moving from a technology-driven approach to a value-driven one (Panagou et al., 2024). While I4.0 focused on the connectivity of smart factories to optimize production metrics, I5.0 emphasizes human-centricity and resilience. It reintroduces the human agent into the production loop, not merely as a supervisor, but as the central pillar of the ecosystem. The "Operator 5.0", the operator within I5.0, is envisioned as a "Smart Operator," augmented by technologies like exoskeletons, augmented reality (AR), and intelligent assistants that reduce physical load and cognitive stress (Rahman et al., 2024).

This shift is driven by the recognition that while automation excels at repetitive, high-speed execution, it lacks the cognitive flexibility, critical thinking, and adaptability inherent to human intelligence (Loizaga et al., 2023; Panagou et al., 2024; Rahman et al., 2024). Consequently, the future of high-value manufacturing lies in Human-Robot Collaboration (HRC), where the strengths of both agents are leveraged symbiotically (Pluchino et al., 2023; Trapero et al., 2025). In this context, collaborative robots, or "cobots," are designed to operate in shared workspaces with humans, relying on integrated safety mechanisms such as force limitation and speed monitoring rather than the traditional physical safety cages that defined industrial robotics (Rahman et al., 2024).

Despite the theoretical promise of I5.0, the practical realization of seamless HRC is hindered by significant operational bottlenecks (Loizaga et al., 2023; Pluchino et al., 2023; Rahman et al., 2024). One of the most critical of these is the communication and programming bottleneck (El Zaatari et al., 2019; Merlo et al., 2025). Cobots, although designed to work safely alongside humans, have traditionally required expert programming, hindering wider accessibility for novice workers (El Zaatari et al., 2019; Kranti et al., 2024). When a production line faces an unexpected event, such as a supply chain disruption requiring a sudden change in assembly protocol, the robot cannot adapt autonomously (El Zaatari et al., 2019; Merlo et al., 2025). It requires the intervention of a specialized programmer, creating downtime and forcing the manufacturing line to wait idle (Colabianchi et al., 2024), thereby increasing cognitive stress for the programming personnel who need to integrate the changes quickly, and naturally, a considerable loss of money for the company (Bokrantz et al., 2016; Chang et al., 2012). Addressing this operational constraint, conversational programming is emerging as a possible solution: systems that can parse human input, grasp context, and craft corresponding programs interactively (Kranti et al., 2024). By functioning as "translators," LLMs can interpret natural language commands from non-expert

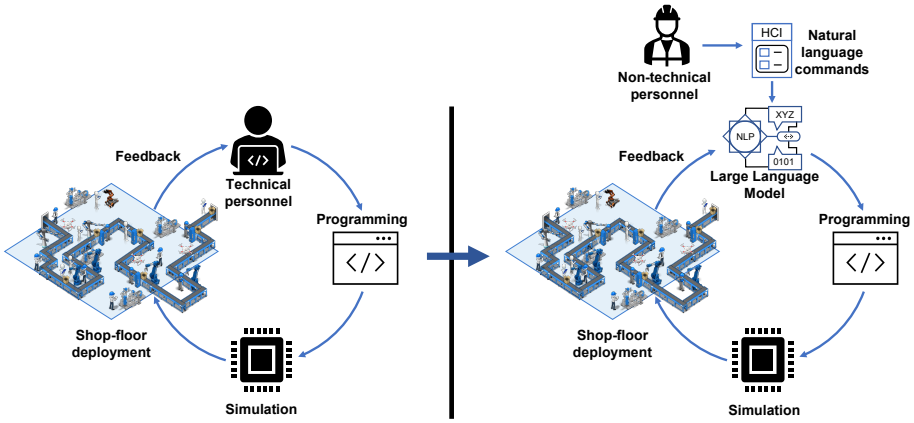


Fig. 1: Schematic of the programming transition. (Left) Traditional workflow: When an unexpected change occurs on the assembly line, specialized personnel must intervene to program, simulate, and upload the new routine. This process induces downtime and excludes the non-expert operator from the control loop due to programming interfaces. (Right) Proposed LLM-based workflow: Non-technical operators can issue natural language commands via a standard keyboard interface. The LLM translates this text into executable code utilizing a one-shot template Chen et al. (2025). A simulation is then presented on the user's display for validation. Upon acceptance, the new routine is integrated and executed in the production cycle.

workers and convert them into executable robotic code, thereby enabling a LLM-based conversational programming paradigm (Colabianchi et al., 2024; Kranti et al., 2024). Figure 1 illustrates this shift from the traditional, expert-dependent workflow to the proposed LLM-driven approach. This capability promises to facilitate cobot programming, keeping the manufacturing worker in the loop and enhancing system resilience (Merlo et al., 2025).

Recent research has begun to operationalize this potential, demonstrating the technical efficacy of LLMs in various manufacturing contexts. For instance, Kranti et al. (2024) validated the use of Large Code Models for synthesizing cobot programs from conversational inputs, while Lim et al. (2024) proposed a framework using LLMs to bridge the communication gap via natural language voice commands. Similarly, Merlo et al. (2025) leveraged LLMs' common-sense reasoning to facilitate human-in-the-loop action replanning and failure mitigation. However, a critical limitation persists in the current body of knowledge: previous studies have predominantly focused on implementation feasibility and technical robustness, such as code generation accuracy or system latency, rather than the granular impact on the human operator. When human factors are evaluated, as seen in Colabianchi et al. (2024), the assessment is often limited to average production metrics or to post-hoc subjective questionnaires such as the NASA Task Load Index (NASA-TLX). Rarely do these studies go as deep as to employ a comprehensive neuroergonomic assessment that triangulates these subjective reports

with objective neuromarkers. Consequently, the extent to which these systems benefit workers physically and cognitively remains underexplored. Addressing this gap, this paper's research question is set as, first, whether it is feasible to implement an LLM for manufacturing task planning to address unexpected changes in the production line, and second, whether this integration tangibly improves industrial workflow performance and human factors as measured by objective physiological data.

To evaluate if our system meets this standard, we employ a comprehensive multimodal neuroergonomic framework (Zakeri et al., 2023). We utilize photoplethysmography (PPG) to estimate Heart Rate (HR), a metric of autonomic nervous system activity (W. Li et al., 2022; Sriranga et al., 2023; Taelman et al., 2009), and analyze eye blink rate as a correlate of cognitive load (Biondi et al., 2023; Marquart et al., 2015; Pluchino et al., 2023). To complement these physiological measures, we integrate the NASA-TLX (Hart, 2006) for subjective workload assessment and track behavioral error rates. The latter serves a dual purpose: quantifying direct productivity improvements while acting as an indirect indicator of operator state, as a less-stressed individual is likely to commit fewer errors (Loizaga et al., 2023; Ricci et al., 2025; Thinnes et al., 2025).

2 Materials and Methods

In total, 29 participants (mean age 27.1, standard deviation 2.5 years, 11 female) with no known neurological or psychiatric disorder participated in an HRC paradigm. After a detailed explanation of the procedure, all participants were required to provide their written consent. The study was conducted at the Neuroergonomic Digital Factory Saar of htw saar and Saarland University in Saarbrücken, Germany. All measurements were conducted in accordance with the Declaration of Helsinki. The local ethics committee approved the study (application: 95/21 Ärztekammer des Saarlandes, Medical Council of the Saarland).

2.1 Experimental Setup

The experiment employed a within-participant design to evaluate the impact of LLM-based adaptation on operator workload. Participants collaborated with a UR5-CB3 cobot to perform repeated sorting operations across four stations.

2.1.1 Roles within the Workspace

The collaborative workflow integrated four distinct roles. The participant was responsible for physically populating the workpiece slots according to instructions displayed on a screen (seen in Fig. 4). Working alongside the participant, the cobot managed the automated workpiece handling. At active stations, the cobot retrieved completed workpieces and transferred them to a designated green frame, relieving both the operator and a human assistant of manual lifting and transport duties. The assistant, present in both conditions, logged the assembled workpieces and manually retrieved them during cobot-unaware tasks. Finally, during the LLM-enabled workflow, the LLM served as a high-level reasoning engine and translator. It bridged the communication gap by

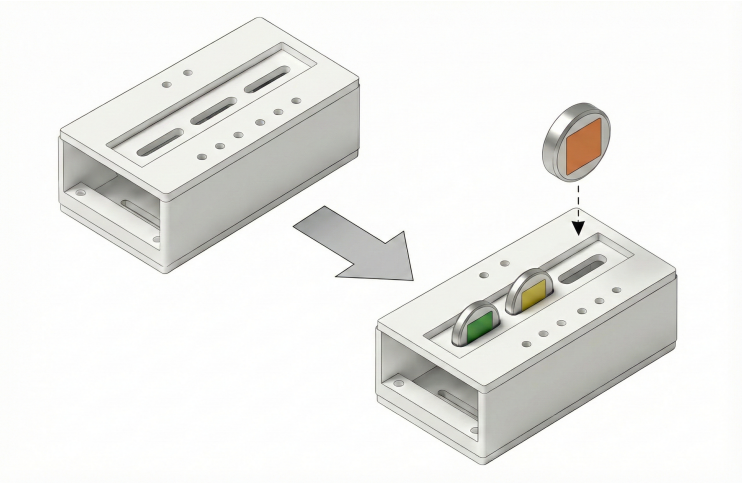


Fig. 2: Isometric view of the workpieces. The left shows the empty housing, and the right shows the housing with button cell batteries being inserted into the slots. The small circular marks on the top surface serve as orientation indicators for the operator, ensuring the workpiece is correctly positioned (two marks on the left, six on the right) and not reversed.

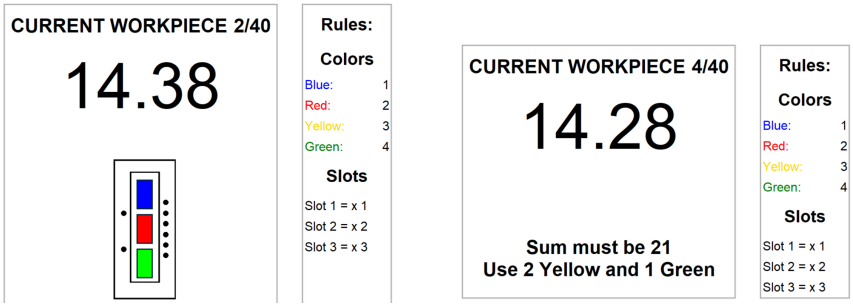
converting the operator's natural-language requests into executable URScript Python code, enabling seamless adaptation without specialized programming knowledge.

2.1.2 Task Types: Cobot-Aware vs. Cobot-Unaware

At each station, the operator performed one of two distinct task types, both of which required the participant to fill cell batteries, as illustrated in Fig. 2.

The first type, the cobot-aware task (or image replication task), represented a fully integrated system (Fig. 3.A). Because the system was programmed to manage this station, it was simulated to handle the background machine tasks, namely the necessary arithmetic calculations. Consequently, the human's task was significantly simplified: they only needed to populate the workpiece slots to match a displayed random color configuration. The rule-set section of the display was irrelevant, and the cobot automatically handled the physical retrieval.

The second type, the cobot-unaware task (or cognitive calculation task), simulated a scenario where the system lacked the programming for a station, forcing a manual fallback. This introduced a much higher cognitive and physical load. Lacking system orchestration, the display provided only "raw" instructions. Participants had to perform mental arithmetic strictly adhering to a complex rule set (Fig. 3.B) to achieve a target score. Furthermore, because the cobot was unaware of the station, it remained idle, forcing a human assistant to manually retrieve the workpieces for data logging. All task directives were presented via the instruction interface shown in Fig. 4.



(a) Image-matching task (b) Math-based task

Fig. 3: Instruction interface for the two experimental conditions. (a) Image Replication Task: The participant must populate the workpiece slots to match the displayed color configuration. The rules section is ignored for this task. (b) Cognitive Calculation Task: The participant performs mental arithmetic to achieve a target score, requiring adherence to the provided rule set. In both conditions, a countdown timer limits the trial to 15 s, after which the workpiece is removed.

2.1.3 Experimental Conditions and the Role of the LLM

The study compared two conditions to highlight the advantage of LLM-enabled conversational programming.

In the static condition (No-LLM), the cobot was rigidly pre-programmed with knowledge of only two of the four stations. The workflow alternated: the first two stations presented simple "cobot-aware" tasks with active cobot assistance, while the remaining two stations simulated an unexpected sequence change. The robot remained idle, forcing the operator into the complex "cobot-unaware" arithmetic tasks and manual retrieval. In a traditional setup, resolving this bottleneck would require an expert programmer to rewrite the system's code to include the new stations.

In the LLM-assisted condition, this bottleneck was bypassed using a type-based natural-language interface powered by OpenAI's GPT-4o-mini (Hurst et al., 2024). Prior to the block start, the operator could simply inform the system of the new requirements. For example, a user could type: "We need to use all four stations for the assembly today, please update the sequence to include stations 3 and 4." Unlike standard speech recognition or simple command-line interfaces, which require exact syntax, the LLM utilized contextual reasoning to parse this high-level intent into runnable Python code. It translated the non-expert human text request into a specific, executable background script that enables all 4 stations. This in-real-time adaptation effectively converted the entire workflow into four simplified "cobot-aware" color-matching tasks (Fig. 5), keeping the cobot active at every station and eliminating the need for traditional reprogramming.

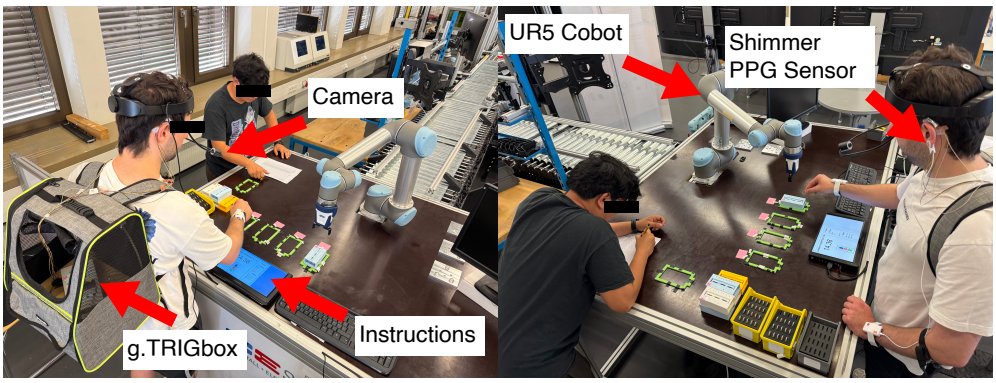


Fig. 4: Experimental Setup. (a) Left: The participant wears a head-mounted camera and a backpack housing a trigger box and an amplifier. Task instructions are displayed on the screen, and an assistant annotates the completed assembly. (b) Right: The participant populates the workpiece according to the assigned pattern; meanwhile, the UR5 cobot waits for the timer to finish. Once the 15-s timer finishes, the workpiece is removed from the work station. A PPG sensor is attached to the left earlobe.

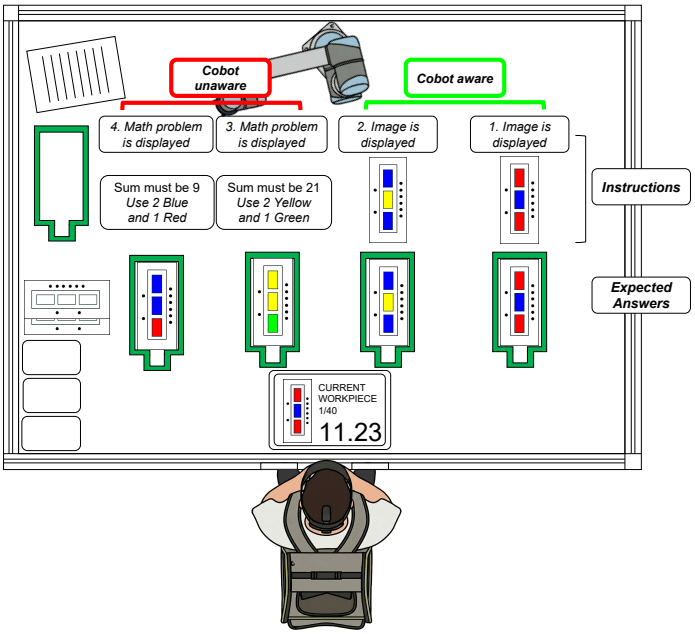
To maintain consistent temporal pressure throughout the experiment, a countdown timer limited the operation time at each station to 15 seconds. Once the timer finished, the workpiece was immediately removed from the workspace, pushing the participant to transition directly to the next station. In both conditions, the cycle of four stations was repeated 10 times, resulting in a total duration of 10 minutes per condition.

As mentioned in Fig. 6, the experimental design employed a counterbalanced block ordering. Thus, participants were randomly assigned to perform either the LLM-assisted or the No-LLM condition first.

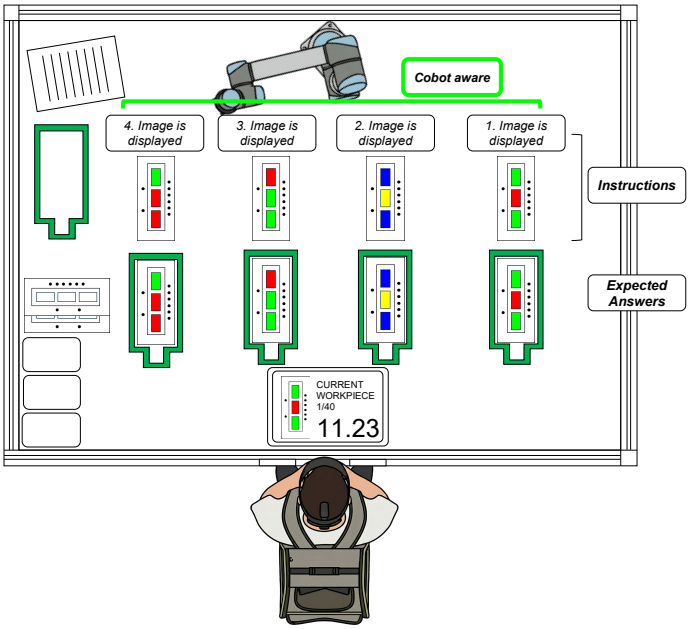
2.2 LLM Integration

To facilitate natural language control, we developed a custom GUI that integrates the GPT-4o-mini model (Hurst et al., 2024) via the OpenAI API. Operators type high-level text commands into the GUI using the physical keyboard shown in Fig. 4. Upon pressing the enter key, the text input is transmitted to the system, which translates it into executable Python scripts using a system prompt engineering strategy. The model is initialized as a robot control expert persona and provided with the complete Scripter class interface alongside a "one-shot" working example (Chen et al., 2025). This context primes the LLM to utilize only valid function calls and adhere to our specific threading and synchronization frameworks. An illustration of the command-to-code workflow is shown in Fig. 1.

Upon receiving a user prompt, the system queries the API with a temperature setting of 0 to maximize determinism and reproducibility. The resulting response is automatically saved as a local Python file, ready for execution by the robot's interpreter. However, before any code is executed, the system initiates a custom analysis routine developed using Python's Abstract Syntax Tree (ast) module. This routine



(a) No-LLM-assisted condition example



(b) LLM-assisted condition example

Fig. 5: View of the paradigm setup for an arbitrary set of the first 4 tasks for both conditions. (a) In the No LLM-assisted condition, the first two stations have cobot-aware tasks (image matching), and the other two stations have cobot-unaware tasks (math-based). (b) For the LLM-assisted condition, all four stations have cobot-aware tasks, meaning only image-matching tasks. Participant repeats the 4-station cycle 10 times in both conditions.

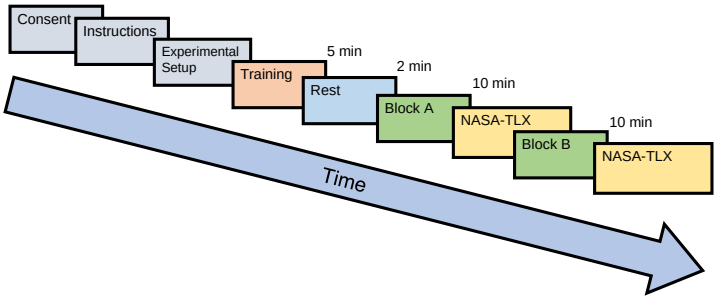


Fig. 6: Measurement plan. A training, a rest stage, and two different blocks were performed. The blocks included a randomly chosen condition and were counterbalanced across participants.

parses the generated script to validate syntactical correctness and reconstruct the logic flow. The routine also traverses the syntax tree to extract dynamic workpiece-indexing commands and displays a text-based preview of the sequence. This allows the operator to visually verify the specific cycle repetitions and movement sequences, serving as a human-in-the-loop safeguard to prevent unintended behaviors before the physical robot moves.

2.3 Data Acquisition

Trigger data were acquired at a sampling rate of 4,800 Hz using a trigger box (g.TRIGbox, gtec, Austria) connected to a biosignal amplifier (g.USBamp, g.tec, Austria) controlled via a Simulink interface. PPG signals were recorded at a sampling rate of 128 Hz using an earlobe PPG sensor (Shimmer3 GSR+, Shimmer Research, Ireland) with an earlobe clip sensor fixed at the participants' left earlobe. No online filtering was applied. To guarantee temporal alignment across these heterogeneous hardware systems, the Lab Streaming Layer (LSL) middleware (Kothe et al., 2025) was utilized as the central synchronization data hub. Each data source was paired with a dedicated software interface acting as an LSL outlet, ensuring all streams shared a unified clock at the point of acquisition. The g.USBamp and g.TRIGbox signals were broadcast to the network via a MATLAB-based LSL outlet, while the Shimmer PPG data was pushed via a custom C# executable. This synchronization architecture extended to the visual data. The video feed from the head-mounted camera (Rokoko Headcam, Rokoko) was integrated into the LSL network via a custom Python script. This LSL Python script captured the video frame numbers and pushed data at 60 Hz with a resolution of 1024 x 768 pixels.

From these synchronized streams, we extracted specific psychophysiological metrics to quantify the interaction's impact. HR was derived from the PPG signal to serve as a marker of autonomic nervous system activity. Theoretically, increased workload triggers the sympathetic nervous system, elevating HR, whereas lower workload conditions are associated with parasympathetic dominance (Taelman et al., 2009). Concurrently, the head-mounted video feed allowed for the analysis of blink rate as an

ocular metric of mental workload. While current literature presents a nuanced debate over the specific drivers of blink rate variability, studies suggest that higher cognitive load generally correlates with an increased blink rate (Biondi et al., 2023; Marquart et al., 2015; Pluchino et al., 2023; Ricci et al., 2025).

Subjective ratings were acquired upon the completion of each experimental block (refer to Fig. 6), in which participants were administered the NASA-TLX (Hart, 2006) questionnaire. The NASA-TLX is a widely accepted instrument for quantifying subjective operator workload, predicated on the assumption that the overall perceived load is a multidimensional construct (Hart, 2006). This study specifically analyzed subjective mental demand, frustration, and effort as the most pertinent metrics for assessing the cognitive cost and potential stress induced by the different workflows. Finally, task error rate was recorded as an objective measure of performance accuracy. A research assistant manually inspected the assembled piece after each trial to verify the correct color ordering of the batteries, as illustrated in Fig. 4.

2.4 Data Processing

2.4.1 HR Estimation

To analyze the mean HR, we extracted inter-beat intervals from the PPG signal. The raw data first underwent linear detrending, followed by a second-order Butterworth bandpass filter with cutoff frequencies at 0.5 Hz and 4 Hz. We defined the analysis window to span from 10 seconds prior to the second trigger to 300 seconds after it. We restricted the analysis to this 5-minute interval for two primary reasons. First, it adheres to the gold standard for short-term HR recordings established by recommendations of the Task Force of the European Society of Cardiology and the North American Society of Pacing and Electrophysiology (Malik et al., 1996). Second, this window focuses on the sustained cognitive effort phase, excluding potential non-stationary effects due to fatigue or resignation in the later stages of the recording. The computation of RR intervals followed the standard guidelines (Berkaya et al., 2018), which exclude intervals shorter than 300 ms or longer than 3000 ms. Additionally, all of the RR intervals that changed by more than 400 ms with respect to the previous valid RR interval were removed. Finally, we corrected ectopic, unmarked, and motion-artifact peaks. These steps yielded the clean normal-to-normal (NN) intervals used to compute the mean HR.

2.4.2 Blink Rate Detection

We determined the blink rate using a comprehensive, semi-automated multi-stage pipeline. The process began with data segmentation, in which the video footage was trimmed to match the exact time window used for the PPG signal analysis. This window was mapped to the camera's LSL timestamps and the corresponding video frame numbers, enabling us to isolate the precise start and end frames for each participant and experimental condition. Once the video segments were defined, we processed the data using the MediaPipe FaceMesh model (Lugaresi et al., 2019) to extract 468 distinct 3D facial landmarks per frame. Before analysis, these landmarks were algorithmically stabilized using orthogonal Procrustes analysis (Schönemann, 1966) to align

the detected head pose to a canonical reference model, neutralizing rotation and translation effects. Subsequently, the landmarks were scale-normalized to adjust for varying distances from the camera. Using these refined landmarks, we calculated the mean Eye Aspect Ratio (EAR), a standard metric for quantifying eye openness (Soukupova & Cech, 2016), for each individual frame.

The blink detection phase involved loading the precomputed EAR signal and utilizing the SciPy *find_peaks* algorithm to locate potential blinks. This step included smoothing the signal with a Savitzky-Golay filter to minimize high-frequency noise while retaining the specific morphological shape of a blink (Al-gawwam & Benaissa, 2018), and establishing a baseline using a median filter. Peaks were then identified based on their prominence, width, and slope. To ensure data integrity, these automated detections were manually verified using a custom MATLAB script that displayed the synchronized video and EAR signal, enabling correction of false positives or negatives. Finally, the confirmed blink count was normalized by the window duration to yield the blink rate (blinks/min).

2.5 Statistical Analysis

Statistical analyses comparing the no-LLM and LLM-assisted conditions were conducted using R (version 4.3.3). We employed one-tailed paired samples t-tests with a significance threshold of $\alpha = 0.05$. Normality was assessed using the Shapiro-Wilk test. In cases where this assumption was violated ($p < 0.05$), we utilized the Wilcoxon signed-rank test. To control for family-wise error rates across the subjective NASA-TLX dimensions, we applied the Holm-Bonferroni correction for multiple comparisons.

3 Results

All statistical comparisons reported in this section were conducted as one-tailed tests, with the alternative hypothesis specifying that the values in the LLM-assisted condition would be lower than those in the no LLM-assisted condition. Results are summarized in Table 1.

3.1 Performance Metrics

The analysis first examined the task error rates and interaction efficiency. The Shapiro-Wilk test indicated that the paired differences for error rates followed a normal distribution ($W = 0.942, p = 0.112$). A paired t-test confirmed a significant reduction in error rates for the LLM-assisted condition ($M = 1.03, SD = 1.57$) compared to the manual condition ($M = 27.85, SD = 7.90$), $t = 17.768, p < 0.001$. This represents a 96% decrease in error rates. Additionally, the efficiency of the LLM interaction was observed to be high, requiring an average of only 1.2 prompts ($SD = 0.5$) to generate the correct code.

3.2 Subjective Workload

Subjective workload was assessed across three dimensions: effort, mental demand, and frustration. Normality testing revealed that the difference scores for effort ($W = 0.902, p = 0.011$) and mental demand ($W = 0.924, p = 0.038$) deviated significantly from normality, whereas frustration scores were normally distributed ($W = 0.936, p = 0.080$).

Consequently, the Wilcoxon signed-rank test was applied to the non-normal dimensions. Participants reported significantly lower effort in the LLM condition ($M = 35.35, SD = 24.09$) compared to the baseline ($M = 73.28, SD = 20.89$), corresponding to a 52% reduction ($V = 0.000, p < 0.001$). Similarly, mental demand was 61% lower with LLM support ($M = 28.79, SD = 16.57$) than without ($M = 73.28, SD = 16.22$), a significant difference ($V = 0.000, p < 0.001$). For frustration, the paired t-test demonstrated a significant 66% decrease in the LLM condition ($M = 20.69, SD = 22.15$) compared to the no LLM condition ($M = 60.52, SD = 29.44$), $t = 8.598, p < 0.001$.

3.3 Physiological Measures

To objectively assess operator state, mean HR and blink rate were analyzed. The Shapiro-Wilk test indicated that the paired differences for both mean HR ($W = 0.917, p = 0.026$) and blink rate ($W = 0.860, p = 0.002$) violated the assumption of normality.

Using the Wilcoxon signed-rank test, mean HR was found to be significantly lower during the LLM condition ($M = 90.11, SD = 10.88$) compared to the no LLM condition ($M = 93.18, SD = 12.33$), $V = 56, p < 0.001$, as illustrated by the boxplots in Fig. 9.A. Furthermore, as seen in Fig. 9.B, the blink rate was significantly reduced in the LLM condition ($M = 16.16, SD = 11.09$) relative to the baseline ($M = 18.04, SD = 10.03$), $V = 74, p = 0.002$.

4 Discussion

The integration of an LLM into the collaborative assembly workflow demonstrated high technical efficacy, particularly in translating natural language intent into executable robot commands. The results indicate that the generalized "common sense" reasoning capabilities inherent in large pre-trained foundation models, specifically GPT-4o-mini (Hurst et al., 2024), effectively bridge the semantic gap between human operators and robotic control systems without the necessity for computationally expensive model fine-tuning. This aligns with recent findings that leveraging LLM common-sense reasoning can successfully facilitate human-in-the-loop action replanning (Merlo et al., 2025). Notably, by utilizing a one-shot prompt engineering strategy conditioned on a single structural anchor (Chen et al., 2025), the system consistently yielded executable code within 1 to 2 prompts. This efficiency offers a significant architectural advantage over established frameworks like Code as Policies (Liang et al., 2023) and ProgPrompt (Singh et al., 2022), which typically rely on token-heavy few-shot prompting (providing multiple in-context examples) to guarantee syntactic

Table 1: Descriptive statistics and hypothesis testing results comparing the no LLM- vs. LLM-assisted conditions. Significance is denoted by p-values < 0.05.

Measure	**No LLM** $M(SD)$	**LLM** $M(SD)$	**Normality** (W)	**Test Used**	**Statistic**	p**-value**
Performance						
Error Rate	27.85 (7.90)	1.03 (1.57)	0.942	Paired t-test	$t = 17.768$	< 0.001
NASA-TLX						
Effort	73.28 (20.89)	35.35 (24.09)	0.902*	Wilcoxon	$V = 0.000$	< 0.001
Mental Demand	73.28 (16.22)	28.79 (16.57)	0.924*	Wilcoxon	$V = 0.000$	< 0.001
Frustration	60.52 (29.44)	20.69 (22.15)	0.936	Paired t-test	$t = 8.598$	< 0.001
Physiological						
Mean HR (bpm)	93.18 (12.33)	90.11 (10.88)	0.917*	Wilcoxon	$V = 56$	< 0.001
Blink Rate (blinks/min)	18.04 (10.03)	16.16 (11.09)	0.860*	Wilcoxon	$V = 74$	0.002

*Indicates deviation from normality ($p < 0.05$).

Fig. 7: Boxplot shows the task error rates during the two conditions (*** denotes $\alpha < 0.001$).

correctness. Our findings suggest that modern foundation models have evolved sufficiently to infer control logic from a single template, matching the high reliability standards reported in recent comparative benchmarks for code generation in HRI (Sobo et al., 2025; Wang et al., 2025). This validates the hypothesis that context-rich, one-shot prompting is now a viable, resource-efficient alternative to few-shot methods for handling logic-based assembly adjustments.

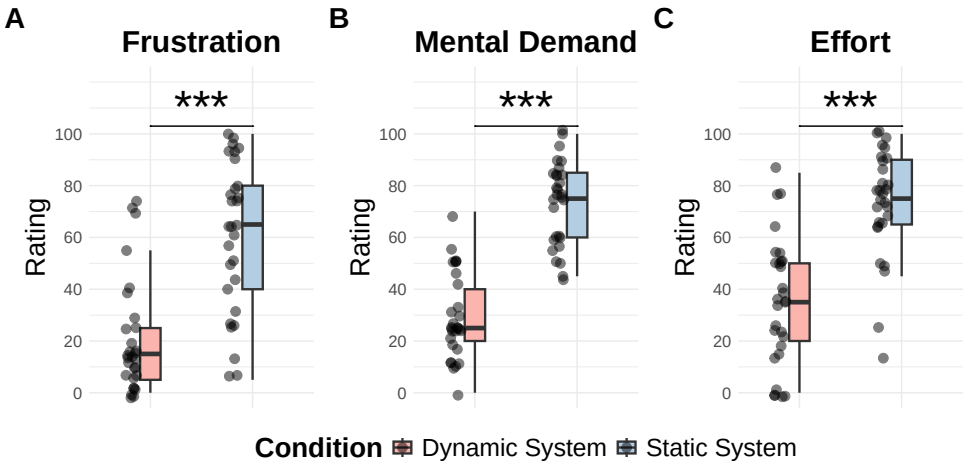


Fig. 8: Boxplots of the NASA-TLX scores across dimensions: a) Frustration, b) Mental Demand, and c) Effort (*** denotes $\alpha < 0.001$).

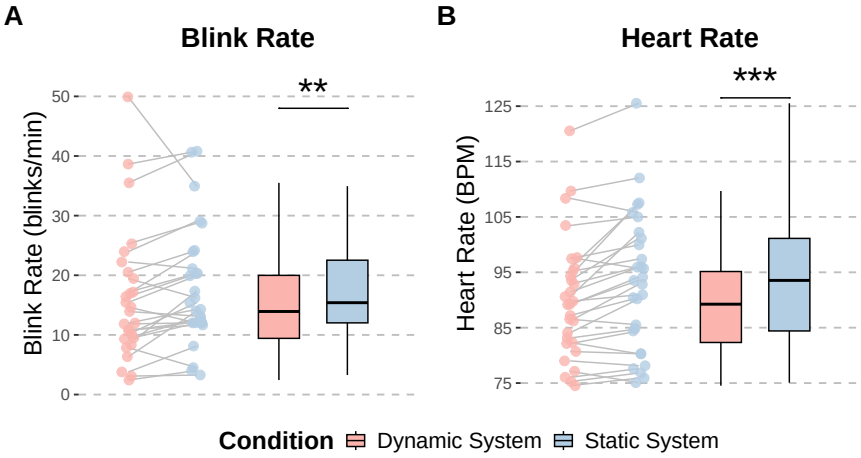


Fig. 9: Comparison of physiological metrics. (a) shows the mean HR analysis, while (b) displays the blink rate. (** denotes $\alpha < 0.01$, *** denotes $\alpha < 0.001$).

Furthermore, the application of lower temperature settings proved critical in minimizing the inherent stochasticity of the generative model, ensuring that the outputs remained syntactically compliant with the robot's interpreter. However, this reliability was largely supported by the underlying control architecture, which abstracted complex movements into pre-defined subfunctions. Our results suggest that for structured manufacturing tasks, foundation models are highly effective when used to sequence

these high-level actions. The results' accuracy also suggests the technical feasibility of using generative AI to rapidly adapt cobots to new task sequences without requiring specialized programming knowledge, a well-documented barrier in traditional cobot interaction (El Zaatari et al., 2019; Liu et al., 2024). These technical results translated directly into a perceived improvement in the working environment (refer to Fig. 8). Subjective assessments using the NASA-TLX indicated that the LLM-assisted condition was significantly more comfortable and mentally less demanding for participants than the static, unassisted condition.

This reduction in subjective workload was corroborated by the battery assembly error rate, which served as the objective task performance metric. In the no LLM condition, designed to simulate a traditional, rigid production workflow, the cobot could not assist with unprogrammed stations. Consequently, participants were forced to revert to manual, "cobot-unaware" tasks involving arithmetic computations to maintain workflow continuity. The reduction in error rates in the LLM condition, as shown in Fig. 7, confirms that empowering the robot to adapt to unexpected workflow changes significantly reduces the cognitive burden on the human operator. This observation is consistent with recent neuroergonomic evaluations in collaborative manufacturing, which identify high mental workload as a primary antecedent to productivity loss and operational errors (Caiazzo et al., 2025). By shifting the interaction modality from rigid programming to flexible natural language, the system reduces the cognitive strain typically observed during task transitions (Lim et al., 2024). This aligns with recent findings (Colabianchi et al., 2024), who demonstrated that LLM-based intelligent assistants significantly lower subjective cognitive workload. Furthermore, this error reduction corroborates the failure mitigation capabilities observed in recent human-in-the-loop frameworks, where common-sense reasoning agents actively validate user intent to prevent execution failures (Merlo et al., 2025).

These performance reports of increased precision were also supported by objective validation, specifically through mental workload metrics. The static condition, characterized by high manual compensation, elicited higher physiological markers of cognitive load than the LLM-assisted interaction. Specifically, the participants exhibited increased sympathetic nervous system activation, reflected by a higher mean HR during the unsupported tasks, a physiological response linked to cognitive load in human-robot interaction contexts (W. Li et al., 2022; Sriranga et al., 2023; Taelman et al., 2009). Furthermore, the analysis of eye metrics showed a significant increase in blink rate during the more mentally demanding cobot-unaware manufacturing tasks. This aligns with I5.0 human factors characterization, connecting increased blink rate to heightened cognitive load (Biondi et al., 2023; Marquart et al., 2015; Pluchino et al., 2023; Ricci et al., 2025). The convergence of subjective feedback and performance metrics with these objective physiological signals provides robust evidence that natural language programming does not merely improve usability but fundamentally alters the physiological cost of the interaction, promoting a more sustainable working environment (Panagou et al., 2024).

Moreover, the workflow presented in this paradigm exemplifies the co-pilot paradigm, where the human remains in the loop not as a manual laborer, but as a strategic verifier of plans generated by the AI (Leng et al., 2022). This shifts the

human-machine interaction from a dynamic defined by rigid code to a collaborative partnership mediated by natural language, a core requirement for the human-centric approach of Industry 5.0 (Ricci et al., 2025). This capability is essential for modern manufacturing environments where high-mix, low-volume production requires frequent, rapid reconfiguration of assembly lines that traditional programming interfaces cannot support efficiently (Leng et al., 2022; Rahman et al., 2024).

Despite these advantages, several technical limitations regarding physical grounding and system architecture must be addressed. The current implementation relies on a text-based LLM, which lacks intrinsic physical grounding. The model does not "see" the workspace and relies entirely on the semantic description provided in the prompt. This abstraction introduces the risk that the model may generate logically sound but spatially infeasible trajectories if the set of spatial constraints is not respected. Furthermore, the reliance on an external, cloud-based API introduces latency and data privacy concerns that may be prohibitive for real-time control loops or proprietary manufacturing data. While the latency observed was acceptable for high-level task planning, it precludes the use of this architecture for real-time trajectory modulation. A viable solution for future research is the deployment of localized open-source LLMs, such as LLaMA (Dubey et al., 2024), which would mitigate privacy risks and reduce latency by keeping inference on-premise, ensuring data sovereignty (Golpayegani et al., 2024).

Finally, regarding safety and transparency, while the system incorporated physical guardrails, such as defined dead zones to prevent collisions, there remains a gap in the visualization of intent. The current system outputs text-based coordinate plans, which require the operator to mentally map the sequence to the manufacturing task for verification. This lack of a visual simulation or digital twin animation prior to execution hinders intuitive verification (Leng et al., 2022), increasing the mental effort required to confirm safety, a critical factor given that increased cognitive load is directly linked to higher risks of occupational injury (Bonsang & Caroli, 2021).

Future iterations of this framework should integrate a visualizer that renders the LLM-generated path in a 3D environment (C. Li et al., 2022; Liu et al., 2024; Ong et al., 2020). Such immersive interfaces would better support human attention mechanisms (Strauss et al., 2025, 2024), allowing the human operator to visually validate the digital assistant's (Colabianchi et al., 2024) suggestion before physical execution. Additionally, the current implementation is functionally constrained to predefined positions. This discretization limits the system's flexibility, preventing the execution of more complex trajectories required for more flexible assembly scenarios. To address this, future versions must expand the LLM's capabilities to generate dynamic paths for arbitrary coordinates (Singh et al., 2022; Wang et al., 2025). Integrating such visualization tools and flexible motion planning with localized, physically grounded models represents the next necessary step toward a fully resilient, intelligent, and human-centric manufacturing system.

5 Conclusions

This study establishes the viability of a neuroergonomically validated framework for integrating LLMs in HRC. While prior research has largely focused on the technical feasibility of code generation, our work provides holistic validation of the human-in-the-loop experience, offering empirical evidence that shifting from rigid programming to natural language interaction fundamentally alters the physiological cost of manufacturing tasks. Results indicate that a one-shot prompt engineering strategy provides a computationally efficient means to adapt robot workflows, reducing the reliance on extensive model fine-tuning. By analyzing in parallel objective neurophysiological markers with behavioral performance, results also suggest that this architectural framework safeguards operator well-being, mitigating the cognitive saturation that typically precedes operational errors in highly dynamic production environments.

For the wider domain of I5.0, these findings suggest a pathway toward the democratization of advanced robotics. By validating a system that functions effectively with a generic foundation model and minimal prompting, we lower the barrier to entry into cobotics, enabling a paradigm shift where operators act as strategic co-pilots rather than technical troubleshooters. Crucially, this shift offers a mechanism to optimize the cognitive ergonomics of the manufacturing floor. By modulating the level of LLM support, it becomes possible to calibrate the operator's cognitive load to the optimal performance zone described by the Yerkes-Dodson Law (Yerkes & Dodson, 1908), thereby mitigating the risks of both cognitive overload and the vigilance decrement associated with under-stimulation. However, to fully transition this framework from a controlled experimental setting to the factory floor, future architectures must bridge the gap between semantic understanding and physical reality. The next generation of this system must prioritize data sovereignty through localized AI deployment and operational transparency through immersive digital twin visualizations. Addressing these constraints will evolve the system from a flexible programming tool into a fully resilient, trustworthy, and human-centric industrial partner.

From an I5.0 perspective, these findings illustrate a pathway toward lowering the barrier to entry for Small and Medium-sized Enterprises (SMEs) seeking to adopt flexible automation. By eliminating the requirement for specialized machine learning expertise or rigid code-based programming (Colabianchi et al., 2024; Kranti et al., 2024), this system democratizes access to advanced robotics.

Acknowledgements. The authors would like to thank Fabian F. Hernandez for helping during data acquisition and early discussion on the topic.

Declarations

Some journals require declarations to be submitted in a standardised format. Please check the Instructions for Authors of the journal to which you are submitting to see if you need to complete this section. If yes, your manuscript must contain the following sections under the heading 'Declarations':

- Funding: This work was supported by the German Federal Ministry of Education and Research under Grant 13FH630KX1 and by the European Union and the state

of Saarland (European Regional Development Fund, ERDF), project Center for Digital Neurotechnologies Saar–CDNS.

- Conflict of interest/Competing interests (check journal-specific guidelines for which heading to use): The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work presented in this paper.
- Ethics approval and consent to participate: All measurements were conducted following the Declaration of Helsinki. The local ethics committee approved the study (application: 95/21 Ärztekammer des Saarlandes, Medical Council of the Saarland).
- Consent for publication: Not applicable
- Data availability: On request
- Materials availability: Not applicable
- Code availability: On request
- Author contribution: On request

If any of the sections are not relevant to your manuscript, please include the heading and write 'Not applicable' for that section.

References

Al-gawwam, S., & Benaissa, M. (2018). Robust Eye Blink Detection Based on Eye Landmarks and Savitzky–Golay Filtering. *Information*, *9*(4), 93,

Berkaya, S.K., Uysal, A.K., Sora Gunal, E., et al. (2018). A survey on ECG analysis. *Biomedical Signal Processing and Control*, *43*, 216–235,

Biondi, F.N., Saberi, B., Graf, F., et al. (2023). Distracted worker: Using pupil size and blink rate to detect cognitive load during manufacturing tasks. *Applied Ergonomics*, *106*, 103867,

Bokrantz, J., Skoogh, A., Ylipää, T., et al. (2016). Handling of production disturbances in the manufacturing industry. *Journal of Manufacturing Technology Management*, *27*(8), 1054–1075,

Bonsang, E., & Caroli, E. (2021). Cognitive Load and Occupational Injuries. *Industrial Relations: A Journal of Economy and Society*, *60*, 219–242,

Caiazzo, C., Djapan, M., Savkovic, M., et al. (2025). Evaluating Mental Workload and Productivity in Manufacturing: A Neuroergonomic Study of Human–Robot Collaboration Scenarios. *Machines*, *13*(9), 783,

Chang, Q., Liu, J., Xiao, G., et al. (2012). The Costs of Downtime Incidents in Serial Multi-Stage Manufacturing Systems. *Journal of Manufacturing Science and Engineering*, *134*, 021016,

Chen, B., Zhang, Z., Langrené, N., et al. (2025). Unleashing the potential of prompt engineering for large language models. *Patterns*, *6*(6), ,

Colabianchi, S., Costantino, F., & Sabetta, N. (2024). Assessment of a large language model based digital intelligent assistant in assembly manufacturing. *Computers in Industry*, *162*, 104129,

Dubey, A., Jauhri, A., Pandey, A., et al. (2024). The Llama 3 Herd of Models. arXiv:2407.21783

El Zaatari, S., Marei, M., Li, W., et al. (2019). Cobot programming for collaborative industrial tasks: An overview. *Robotics and Autonomous Systems*, *116*, 162–180,

Golpayegani, F., Chen, N., Afraz, N., et al. (2024). Adaptation in Edge Computing: A Review on Design Principles and Research Challenges. *ACM Transactions on Autonomous and Adaptive Systems*, *19*(3), 19:1–19:43,

Hart, S.G. (2006). Nasa-Task Load Index (NASA-TLX); 20 Years Later. *Proceedings of the human factors and ergonomics society annual meeting* (Vol. 50, pp. 904–908).

Hurst, A., Lerer, A., Goucher, A.P., et al. (2024). Gpt-4o system card. arXiv:2410.21276

Kothe, C., Shirazi, S.Y., Stenner, T., et al. (2025). The lab streaming layer for synchronized multimodal recording. *Imaging Neuroscience*, *3*, IMAG.a.136,

Kranti, C., Hakimov, S., & Schlangen, D. (2024). Towards no-code programming of cobots: Experiments with code synthesis by large code models for conversational programming. arXiv:2409.11041

Leng, J., Sha, W., Wang, B., et al. (2022). Industry 5.0: Prospect and retrospect. *Journal of Manufacturing Systems*, *65*, 279–295,

Li, C., Zheng, P., Li, S., et al. (2022). AR-assisted digital twin-enabled robot collaborative manufacturing system with human-in-the-loop. *Robotics and Computer-Integrated Manufacturing*, *76*, 102321,

Li, W., Li, R., Xie, X., et al. (2022). Evaluating mental workload during multitasking in simulated flight. *Brain and Behavior*, *12*, e2489,

Liang, J., Huang, W., Xia, F., et al. (2023). Code as Policies: Language Model Programs for Embodied Control.
arXiv:2209.07753

Lim, J., Patel, S., Evans, A., et al. (2024). Enhancing human-robot collaborative assembly in manufacturing systems using large language models. *2024 ieee 20th international conference on automation science and engineering (case)* (pp. 2581–2587).

Liu, C., Tang, D., Zhu, H., et al. (2024). An augmented reality-assisted interaction approach using deep reinforcement learning and cloud-edge orchestration for user-friendly robot teaching. *Robotics and Computer-Integrated Manufacturing*, *85*, 102638,

Loizaga, E., Eyam, A.T., Bastida, L., et al. (2023). A Comprehensive Study of Human Factors, Sensory Principles, and Commercial Solutions for Future Human-Centered Working Operations in Industry 5.0. *IEEE Access*, *11*, 53806–53829,

Lugaresi, C., Tang, J., Nash, H., et al. (2019). Mediapipe: A framework for building perception pipelines.
arXiv:1906.08172

Malik, M., Bigger, J.T., Camm, A.J., et al. (1996). Heart rate variability: Standards of measurement, physiological interpretation, and clinical use. *European Heart Journal*, *17*, 354–381,

Marquart, G., Cabrall, C., & De Winter, J. (2015). Review of Eye-related Measures of Drivers' Mental Workload. *Procedia Manufacturing*, *3*, 2854–2861,

Merlo, E., Lagomarsino, M., & Ajoudani, A. (2025). A Human-in-The-Loop Approach to Robot Action Replanning Through LLM Common-Sense Reasoning. *IEEE Robotics and Automation Letters*, *10*, 10767–10774,

Ong, S.K., Yew, A.W.W., Thanigaivel, N.K., et al. (2020). Augmented reality-assisted robot programming system for industrial applications. *Robotics and Computer-Integrated Manufacturing*, *61*, 101820,

Panagou, S., Neumann, W.P., & Fruggiero, F. (2024). A scoping review of human robot interaction research towards Industry 5.0 human-centric workplaces. *International Journal of Production Research*, *62*, 974–990,

Pluchino, P., Pernice, G.F.A., Nenna, F., et al. (2023). Advanced workstations and collaborative robots: Exploiting eye-tracking and cardiac activity indices to unveil senior workers' mental workload in assembly tasks. *Frontiers in Robotics and AI*, *10*, ,

Rahman, M.M., Khatun, F., Jahan, I., et al. (2024). Cobotics: The Evolving Roles and Prospects of Next-Generation Collaborative Robots in Industry 5.0. *Journal of Robotics*, *2024*, 2918089,

Ricci, A., Ronca, V., Capotorto, R., et al. (2025). Understanding the Unexplored: A Review on the Gap in Human Factors Characterization for Industry 5.0. *Applied Sciences*, *15*, 1822,

Schönemann, P.H. (1966). A Generalized Solution of the Orthogonal Procrustes Problem. *Psychometrika*, *31*(1), 1–10,

Singh, I., Blukis, V., Mousavian, A., et al. (2022). ProgPrompt: Generating Situated Robot Task Plans using Large Language Models.
arXiv:2209.11302

Sobo, A., Mubarak, A., Baimagambetov, A., et al. (2025). Evaluating LLMs for Code Generation in HRI: A Comparative Study of ChatGPT, Gemini, and Claude. *Applied Artificial Intelligence*, *39*(1), 2439610,

Soukupova, T., & Cech, J. (2016). Eye Blink Detection Using Facial Landmarks. *21st computer vision winter workshop (cvww)* (Vol. 1, pp. 1–8).

Sriranga, A.K., Lu, Q., & Birrell, S. (2023). A Systematic Review of In-Vehicle Physiological Indices and Sensor Technology for Driver Mental Workload Monitoring. *Sensors*, *23*, 2214,

Strauss, D.J., Francis, A.L., Schäfer, Z., et al. (2025). Understanding speech in "noise" or free energy minimization in the soundscapes of the anthropocene. *Frontiers in Neuroscience*, *19*, ,

Strauss, D.J., Francis, A.L., Vibell, J., et al. (2024). The role of attention in immersion: The two–competitor model. *Brain Research Bulletin*, *210*, 110923,

Taelman, J., Vandeput, S., Spaepen, A., et al. (2009). Influence of Mental Stress on Heart Rate and Heart Rate Variability. *4th european conference of the international federation for medical and biological engineering* (pp. 1366–1369).

Thinnes, D., Francis, A.L., Sayman, V., et al. (2025). Neuroergonomics in Digital Operating Rooms: Applying the Two-Competitor Model of Attention to the Surgical Context. *IEEE Transactions on Human-Machine Systems*, *55*, 765–776,

Trapero, J., Thinnes, D., Wagner, E., et al. (2025). Haptic Vest-Attention Assistance for Outside Field-of-View Guidance and Enhanced Human–Robot Interaction. *IEEE Transactions on Industrial Informatics*, *PP*, Early Access,

Wang, W., Rong, Y., Li, Y., et al. (2025). LLM-Driven Corrective Robot Operation Code Generation with Static Text-Based Simulation. arXiv:2512.02002

Yerkes, R.M., & Dodson, J.D. (1908). The relation of strength of stimulus to rapidity of habit-formation. *Journal of Comparative Neurology and Psychology*, *18*(5), 459–482,

Zakeri, Z., Arif, A., Omurtag, A., et al. (2023). Multimodal Assessment of Cognitive Workload Using Neural, Subjective and Behavioural Measures in Smart Factory Settings. *Sensors (Basel)*, *23*, 8926,