POLITECNICO DI TORINO
Repository ISTITUZIONALE

Reciprocal human–machine learning flow modelling for assisted assembly systems

Original
Reciprocal human–machine learning flow modelling for assisted assembly systems / Fan, Y., Simeone, A., Antonelli, D.. - 57:(2025), pp. 457-465. (XVII AITeM (Italian Manufacturing Association) Conference 10-12 September 2025, Politecnico di Bari, Italy) [10.21741/9781644903735-54].

Availability:
This version is available at: 11583/3013065 since: 2026-07-13T18:29:53Z

Publisher:
Materials Research Forum LLC

Published
DOI:10.21741/9781644903735-54

Terms of use:

This article is made available under terms and conditions as specified in the corresponding bibliographic description in the repository

Publisher copyright

(Article begins on next page)

12 October 2026

Materials Research Proceedings 57 (2025) 457-465 https://doi.org/10.21741/9781644903735-54

Reciprocal human–machine learning flow modelling for assisted assembly systems

Yuchen FAN[1,a *], Alessandro SIMEONE[1,b *] and Dario ANTONELLI[1,c *]

[1]Department of Management and Production Engineering, Politecnico di Torino, Turin, Italy

[a]yuchen.fan@polito.it, [b]alessandro.simeone@polito.it, [c]dario.antonalli@polito.it

Keywords: Human-Robot Collaboration, Real-Time Object Detection, Multi-Modal Instruction Interfaces, Assisted Assembly Systems, Generative AI

Abstract. Industry 5.0 emphasizes human-centric manufacturing by integrating advanced assistive technologies for inclusive production environments. Large Language Models (LLMs) offer new possibilities for assembly error detection and correction. This study introduces a reciprocal human–machine learning framework utilizing LLMs vision capabilities to improve assembly accuracy through real-time image comparison and corrective instruction generation. Human and machine-in-the-loop mechanisms enable continuous refinement, minimizing labeled datasets while enhancing responsiveness. Experimental validation demonstrates the system's ability to detect errors, diagnose causes, and provide corrective actions. Findings show LLM-driven reciprocal learning improves efficiency, supports diverse operator needs, and enables adaptive error detection in manufacturing.

Introduction

The manufacturing sector is undergoing a significant transformation under the Industry 5.0 paradigm, which seeks to integrate advanced technologies with human expertise to create inclusive, adaptive, and efficient production environments [1]. Central to this transformation is the development of human–machine collaboration systems supporting improved productivity while addressing the diverse needs of modern operators [2]. Advancements in generative artificial intelligence have led to novel assisted assembly systems capable of real-time object detection, natural language processing, and multimodal instructional content [3]. This research introduces a reciprocal human–machine learning framework that integrates these AI capabilities with operator feedback to enhance error detection in dynamic assembly environments while ensuring regulatory and ethical compliance [4]. Traditional assembly monitoring relies on pre-trained models with extensive task-specific datasets, limiting adaptability and responsiveness in dynamic manufacturing environments [5]. This research proposes an LLM-based system integrating vision processing and operator feedback for error detection. Rather than advancing real-time vision algorithms, the approach leverages built-in vision capabilities of LLMs to preserve adaptability across diverse manufacturing contexts.

Unlike CNNs requiring extensive training, this system uses prompt-based inference for adaptive assessment without large labeled datasets [6]. Deviations from reference images are detected dynamically, enabling immediate corrective guidance. The proposed system operates within an assisted assembly framework that provides real-time, multimodal feedback through auditory, visual, and robotic interventions.

Literature review

Generative AI and LLMs have revolutionized object recognition and industrial assembly through multimodal learning, visual reasoning, and automated decision-making [7]. Recent advances in multimodal LLMs have enabled the seamless integration of language processing with visual and other sensory modalities. Their impact spans across object detection, recognition, and human-robot

Content from this work may be used under the terms of the Creative Commons Attribution 3.0 license. Any further distribution of this work must maintain attribution to the author(s) and the title of the work, journal citation and DOI. Published under license by Materials Research Forum LLC.

Materials Research Proceedings 57 (2025) 457-465 https://doi.org/10.21741/9781644903735-54

collaboration, enabling new levels of adaptability and efficiency in manufacturing and robotics [8].

Multimodal LLMs enhance object recognition by integrating vision with language processing. Frameworks like ContextDET [9] and open-vocabulary detection systems [10] improve recognition beyond fixed training sets. LLMs in object detection enhance interpretability through richer feature representations, benefiting medical imaging, industrial inspection, and transportation [11]. New attention mechanisms optimize detection in low-data settings by balancing localization and classification [12].

In industrial applications, LLMs enhance human-robot collaboration through digital twin communication [13], natural language command execution [14], and contingency handling with hierarchical task networks [15]. Challenges include computational efficiency, robustness, and hallucination in vision-language models [16], along with addressing bias for better generalization across diverse scenarios [17]. Future developments will focus on optimizing model architectures, improving domain adaptation, and incorporating retrieval-augmented generation to enhance LLM performance in real-world applications.

A critical gap in current research lies in the application of generative AI for assembly error detection. While LLMs have demonstrated potential in recognizing objects and guiding robotic tasks, their role in detecting and responding to assembly errors remains underdeveloped. Recent studies highlight the challenges in designing error detection mechanisms that integrate seamlessly with multimodal AI frameworks. For example, research on linguistic error detection methods for LLMs suggests that multimodal models can be adapted for error identification by deploying contextual reasoning and external validation techniques [9]. Similarly, work on robotic assembly error correction shows that integrating structured visual reasoning with LLMs improves performance in identifying and rectifying misalignments in mechanical assembly tasks [18]. However, these approaches remain limited in scalability and adaptability across different industrial environments. While previous studies have explored LLMs for object recognition and robotic tasks, their integration into real-time assembly error detection with human/machine-in-the-loop mechanisms remains limited. This research addresses these gaps through a reciprocal learning framework combining LLM reasoning with vision-based analysis for dynamic, context-aware adaptation.

Framework

The framework, shown in Fig. 1, enables error detection in collaborative assembly by comparing real-time images against references using LLMs, generating error descriptions and corrective instructions while continuously improving through operator feedback [19]. The system monitors assembly tasks and processes three types of inputs: structured assembly instructions (detailing operations sequence, tools, and quality criteria), reference images (visual ground truth), and real-time images of the assembly progress. Reference images, showing the correct execution of the various assembly steps, including the selection of tools and components, must be acquired prior to assembly task as ground truth. Real-time images are captured continuously with industrial cameras maintaining consistent viewpoints and lighting to enable accurate comparisons. Prompt Management interfaces the assembly tasks with generative AI through three functions: Prompt Storage (repository of refined prompts), Input Conversion (reformatting instructions into standardized layouts), Context Specification (defining operational parameters). Such functions are incorporated into a predefined template to standardize task specifications and error detection requirements. All input data sources, activated by the prompt, are provided to the LLM Error Detection Module. This module performs three primary functions: error detection, error diagnosis with comprehensive error description, and prognosis, which consists of corrective instruction generation. The error detection function systematically compares real-time images with reference images to identify deviations during the assembly process. This approach minimizes the need for

Materials Research Proceedings 57 (2025) 457-465 https://doi.org/10.21741/9781644903735-54

extensive labeling and training because it utilizes predefined reference images along with contextual operator feedback for comparison. The method relies on structural similarities between real-time and reference data, allowing the system to detect discrepancies without relying on large annotated datasets or complex supervised learning methods. When employing LLMs, two inherent challenges must be addressed: hallucination [16]. and catastrophic forgetting [20].

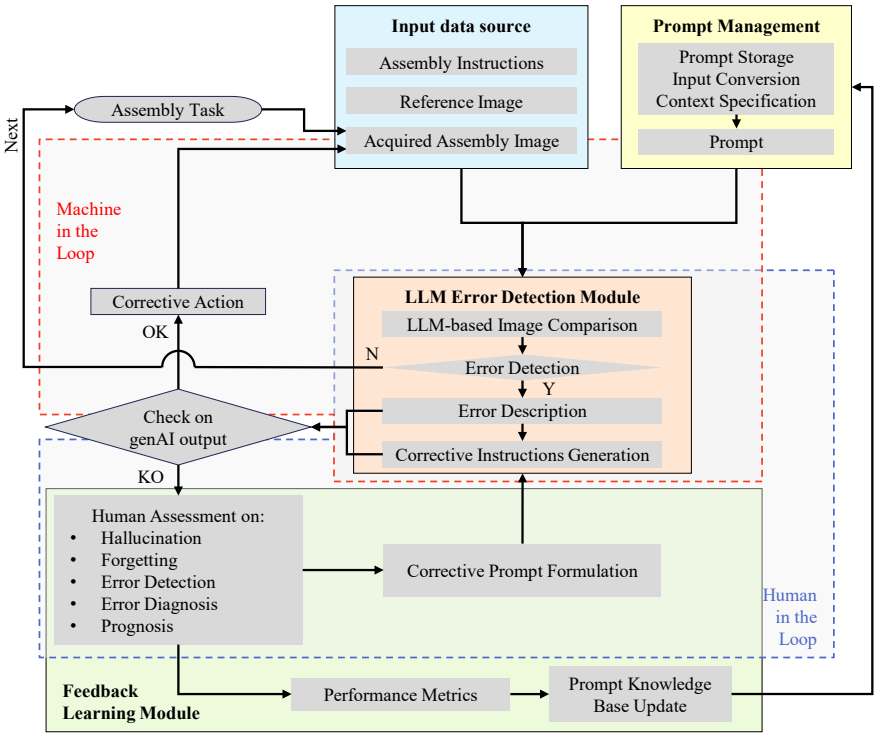

Fig. 1. *Framework flowchart diagram.*

The system mitigates hallucinations through consistency checks using confidence thresholds and feature comparisons to ensure alignment with visual evidence [16]. To address catastrophic forgetting, detection is confined to a single session, maintaining relevant memory without continuous retraining [20]. If no error is detected, the operator is notified, allowing the assembly task to proceed to the next stage. In case an error is detected, the module conducts error diagnosis and provides a detailed error description. In this phase, the system characterizes the nature, severity, and underlying causes of the detected assembly errors by merging automated feature analysis with contextual operator input [21]. Finally, the module performs prognosis, which consists of automatically generating corrective instructions aimed at reversing the error based on the diagnostic output. These instructions are then delivered to the operator, while subsequent responses are captured to continuously refine system performance and improve prognostic accuracy. The Feedback Learning Module enhances performance through human evaluation [22], deriving metrics to analyze discrepancies between detections and outcomes. Human evaluators formulate corrective prompts when needed [23], with refinements integrated into the Knowledge Base for cumulative learning without extensive retraining. The approach integrates two complementary loops (see Fig. 1): Human-in-the-Loop, where operators evaluate and refine LLM outputs through feedback on detection accuracy and corrective instructions; and Machine-in-the-Loop, where the system autonomously detects errors by comparing real-time images with references, generates corrective instructions, and iteratively improves based on learned adjustments.

Materials Research Proceedings 57 (2025) 457-465 https://doi.org/10.21741/9781644903735-54

Experimental Validation

This section reports laboratory validation using a cast iron centrifugal pump assembly. Fig. 2 shows the test bench with an industrial workspace, 1080p cameras (overhead and lateral), and diffused LED lighting for uniform illumination. Validation included 14 assembly tasks across 6 critical steps identified by industrial partners (examples in Table 1). Each task involved processing instruction spreadsheets and comparing reference images (Fig. 3a) with real-time operation images (Fig. 3b). In addition, the prompt presented in Fig. 4 conveys the request to the LLM, specifying the input data, task requirements, and error detection criteria. The tests evaluate the system's diagnostic and prognostic capabilities for error detection, description, and instruction generation. Error detection assesses the system ability to accurately identify the presence of assembly errors. Error description evaluates the clarity and relevance of the system-generated descriptions of detected errors. Corrective instruction generation measures the effectiveness of the system in producing actionable and contextually appropriate corrective instructions.

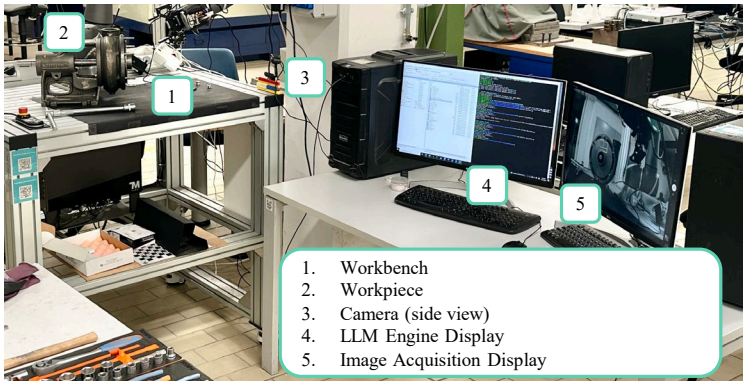

Fig. 2. *Laboratory-scale experimental setup.*

Table 1. *Assembly tasks used in the experimental tests; CS: Component Selection, AO: Assembly Operation.*

Task ID	Task Type	Work instructions
S15.1	CS	Select 1 diffuser (ID: C3) from the component table.
S15.2	AO	Position the diffuser (ID: C3) on the shaft (ID: C5) by aligning the 6 long stud bolts (ID: C22) on the pump housing (ID: C2) to the 6 holes on the diffuser.
S16.1	CS	Select 1 spacer (ID: C14) from the component table.

Materials Research Proceedings 57 (2025) 457-465 https://doi.org/10.21741/9781644903735-54

(a) (b)

***Fig.3**. (a) Reference image showing properly installed locking nut; (b) Real-time operation image showing the absence of locking nut.*

```
The type of this task is "{task_column ['task type']}". The first image shows the correct assembly reference,
and the second image shows the real-time operation result. Based on the two images and the instruction
"{task_column ['work instructions']}", please analyse whether the operator correctly completed this step.
Then, output a Detection Log, which includes: # "Error" or "Correct" # Specific error detection details #
Clear and concise correction instructions, if an error is present.
```

***Fig. 4.** Prompt #1*

The proposed framework was evaluated on two state-of-the-art LLM models: ChatGPT (GPT-4) and Claude (Claude 3). Both models were presented with the identical prompt shown in Fig. 4 and identical inputs containing (i) the assembly task instructions, in MS Excel spreadsheet format, (ii) the reference images of correctly completed assemblies and (iii) the test images of the current assembly state.

This study employed several metrics to assess the performance of the framework. Error detection was evaluated using standard classification metrics: True Positives (correctly identified error-free assemblies), True Negatives (correctly detected errors), False Positives (erroneously flagged errors), and False Negatives (missed errors).

The quality of error descriptions and corrective instructions was quantitatively evaluated using an assessment protocol based on a five-point Likert scale (ranging from 1 to 5) [24] adopting two principal metrics: Error Description Quality (EDQ) [25] and Instruction Quality (IQ) [26]. For EDQ, a score of 1 indicates that the error description is incorrect or misleading, while a score of 5 signifies a comprehensive and precise account of all relevant aspects. Similarly, IQ is rated from 1 for instructions that are vague, incomplete, or potentially harmful, to 5 for clear, actionable, and well-sequenced corrective guidance. Three independent evaluators assessed the LLM outputs using predefined criteria after a calibration session to align their interpretations. The performance is considered sufficient when the average scores for both EDQ and IQ exceed the score of 4.0.

Results and Discussion

Baseline testing without induced errors yielded 100% true positive classifications. With introduced deviations, ChatGPT showed high initial accuracy (92.9%) but limited improvement, while Claude improved from 57.1% to 71.4% after the third prompt iteration. Both maintained >90% accuracy for component/tool selection with greater variability in assembly operations (Figs. 5-6). In assembly operations, ChatGPT showed strong initial performance but minimal improvement, while Claude demonstrated better adaptability with iterative feedback.

Materials Research Proceedings 57 (2025) 457-465 https://doi.org/10.21741/9781644903735-54

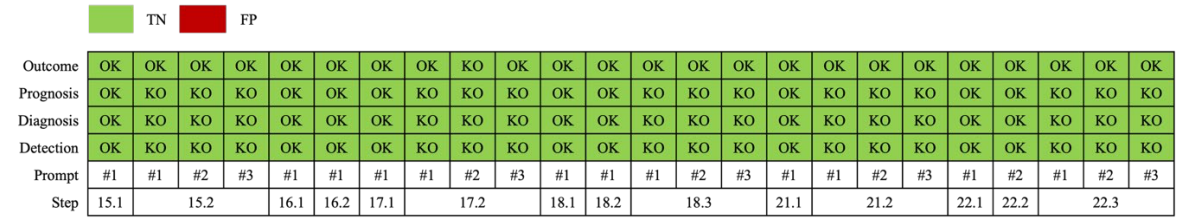

Fig. 5. *ChatGPT performance with purposedly-induced errors.*

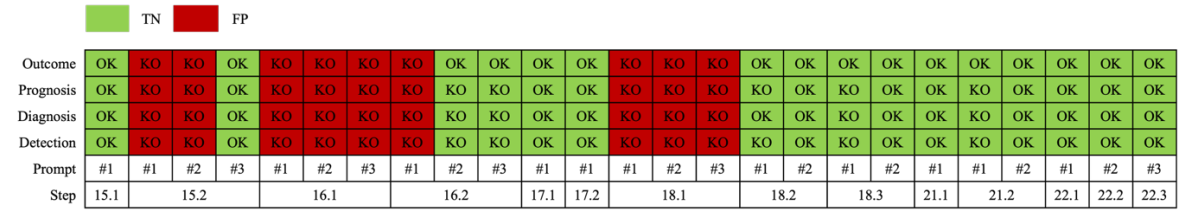

Fig. 6. *ChatGPT performance with purposedly-induced errors.*

These differing learning patterns highlight trade-offs in system design. ChatGPT offers higher initial accuracy but less adaptability, while Claude exhibits greater potential for improvement through operator feedback. The positive outcomes are enhanced adaptability of the machine assisting the operator on new tasks and improved accuracy of both human and machine in the execution of their tasks. This human-machine correction loop represents the practical implementation of the reciprocal learning framework. The system learns from operator expertise while simultaneously providing increasingly refined guidance, creating a continuous improvement cycle that enhances system adaptability over time. While machine-in-the-loop effectiveness has been extensively discussed in [2, 21, 27] in terms of reduced assembly time, error occurrence and error handling times, while human-in-the-loop, previously analyzed for traditional deep learning paradigms [2] has been developed and assessed here for LLM-based error detection, description and correction.

To comprehensively evaluate the model detection capabilities and output quality, examples of LLM-generated responses for step 18.3 were analyzed. In this instance, the operator deliberately omitted the installation of the locking nut, as illustrated in Fig. 3. Both ChatGPT and Claude responses to Prompt #1 were collected and analyzed. Both models identified errors, but with different accuracy. ChatGPT incorrectly diagnosed a "partially threaded" locking nut when it was completely missing, while Claude correctly identified the missing nut despite initial contradictions. Their correction instructions reflected these diagnoses. Following the submission and evaluation of Prompt #2 (reported in Fig. 7), Claude correctly identifies the diagnosis, whereas ChatGPT unsatisfactory performance in terms of error diagnosis and prognosis necessitated a third interaction using Prompt #3, whose content is detailed in Fig. 8.

```
Your previous response was not satisfactory in terms of ['error description and instructions generation'].
Let's refine the assessment. The type of this task is "{task_column ['task type']}". The first image shows the
correct assembly reference, and the second image shows the real-time operation result. Based on these images
and the instruction "{task_column ['work instruction']}", determine whether the correct component or tool was
selected and whether the assembly step was executed properly. Ensure that the selected item is correct,
properly positioned, oriented, aligned, and secured, and that all required components or tools, including
those from previous steps, are present. Identify any errors, such as the use of an incorrect component,
misalignment, missing elements, or improper execution of the assembly step. The Detection Log must indicate
"Error" or "Correct," provide specific details on any mistakes, and include clear and concise corrective
instructions to ensure compliance in future steps.
```

Fig. 7. *Prompt #2*

```
Your previous responses were not satisfactory in terms of ['error description and instructions generation'].
Let's refine the assessment. The type of this task is "{task_column['task type']}". Compare the first image,
which represents the correct reference, with the second image, showing the real-time operation result. Based
on "{task_column['work instruction']}", determine whether the correct component or tool was selected or if the
assembly step was executed correctly. Ensure that, when relevant, the selected item is properly positioned,
oriented, aligned, and secured. Additionally, verify that all required components or tools, including those
from previous steps, are correctly utilized and consistently present throughout the process. If a component
selection occurred in an earlier step, confirm that the same exact component is used in subsequent steps as
expected. If any errors were detected in previous steps, ensure that they have been corrected before proceeding
with this analysis. Identify any remaining errors, such as the use of an incorrect component, misalignment,
missing elements, or improper execution of the assembly step. The Detection Log must indicate "Error" or
"Correct," provide specific details on any mistakes, and include clear and concise corrective instructions to
ensure compliance in future steps.
```

Fig. 8. *Prompt #3*

Despite three increasingly detailed prompts, ChatGPT persisted in its erroneous assessment claiming that the locking nut was partially threaded rather than completely missing. This misjudgment led to errors in subsequent descriptions and corrective instructions, ultimately resulting in a KO status. This example demonstrates the difference in the visual reasoning capabilities of the two models. While both systems correctly recognized the presence of an error, ChatGPT consistently misinterpreted the visual evidence to focus on improper tightening of a non-existent part. In contrast, Claude was able to accurately identify missing parts and provide appropriate corrections after Prompt #2.

Conclusions

This research compared ChatGPT and Claude in an intelligent assisted assembly system utilizing reciprocal learning with human feedback. Results revealed ChatGPT higher initial accuracy versus Claude greater adaptability through iterative refinement, highlighting trade-offs in LLM selection for collaborative tasks and the importance of reciprocal learning. Claude demonstrated superior visual reasoning, particularly in identifying missing components. Future research should focus on extending reciprocal learning to more complex assembly scenarios, optimizing system latency for real-time use, and developing automated prompt refinement strategies. Large-scale validation in diverse industrial environments is also needed to ensure the scalability and robustness of LLM-based error detection and correction systems.

Declaration of generative AI and AI-assisted technologies in the writing process

The authors used ChatGPT and DeepL to improve syntax and grammar, then reviewed and edited the content, assuming full responsibility for the final publication.

References

[1] Leng, J., Sha, W., Wang, B., Zheng, P., Zhuang, C., Liu, Q., Wuest, T., Mourtzis, D., Wang, L.: Industry 5.0: Prospect and retrospect. J Manuf Syst. 65, 279–295 (2022). https://doi.org/10.1016/j.jmsy.2022.09.017

[2] Simeone, A., Fan, Y., Antonelli, D., Catalano, A.R., Priarone, P.C., Settineri, L.: Inclusive manufacturing: A contribution to assembly processes with human-machine reciprocal learning. CIRP Ann Manuf Technol. 73, 5–8 (2024). https://doi.org/10.1016/j.cirp.2024.03.005

[3] Konstantinou, C., Antonarakos, D., Angelakis, P., Gkournelos, C., Michalos, G., Makris, S.: Leveraging Generative AI Prompt Programming for Human-Robot Collaborative Assembly. Procedia CIRP. 128, 621–626 (2024). https://doi.org/10.1016/j.procir.2024.03.040

[4] Liu, S., Zhang, J., Wang, L., Gao, R.X.: Vision AI-based human-robot collaborative assembly driven by autonomous robots. CIRP Annals. (2024). https://doi.org/10.1016/j.cirp.2024.03.004

[5] Yurong, G., Jian, M.: Optimization analysis of assembly system based on digital twin and deep learning. In: Proceedings of the 2020 4th International Conference on Electronic Information Technology and Computer Engineering. pp. 627–631. ACM, New York, NY, USA (2020). https://doi.org/10.1145/3443467.3443825

[6] Fu, B., Hadid, A., Damer, N.: Generative AI in the context of assistive technologies: Trends, limitations and future directions. Image Vis Comput. 154, 105347 (2025). https://doi.org/10.1016/j.imavis.2024.105347

[7] Fan, H., Liu, X., Fuh, J.Y.H., Lu, W.F., Li, B.: Embodied intelligence in manufacturing: leveraging large language models for autonomous industrial robotics. J Intell Manuf. (2024). https://doi.org/10.1007/s10845-023-02294-y

[8] Shafiee, S.: Generative AI in manufacturing: a literature review of recent applications and future prospects. Procedia CIRP. 132, 1–6 (2025). https://doi.org/10.1016/j.procir.2025.01.001

[9] Zang, Y., Li, W., Han, J., Zhou, K., Loy, C.C.: Contextual Object Detection with Multimodal Large Language Models. Int J Comput Vis. 133, 825–843 (2025). https://doi.org/10.1007/s11263-024-02214-4

[10] Yu, Q., Shen, X., Chen, L.-C.: Towards Open-Ended Visual Recognition with Large Language Models. Presented at the (2025). https://doi.org/10.1007/978-3-031-72630-9_21

[11] Jiang, Y., Yan, X., Ji, G.-P., Fu, K., Sun, M., Xiong, H., Fan, D.-P., Khan, F.S.: Effectiveness assessment of recent large vision-language models. Visual Intelligence. 2, 17 (2024). https://doi.org/10.1007/s44267-024-00050-1

[12] Zhang, X., Wu, K., Ma, Q., Chen, Z.: Research on Object Detection Model Based on Feature Network Optimization. Processes. 9, 1654 (2021). https://doi.org/10.3390/pr9091654

[13] Wang, Z., Qin, H.: Intelligent industrial production process automatic regulation system based on LLM agents. In: 2024 5th International Conference on Artificial Intelligence and Electromechanical Automation (AIEA). pp. 133–137. IEEE (2024). https://doi.org/10.1109/AIEA62095.2024.10692701

[14] Lim, J., Patel, S., Evans, A., Pimley, J., Li, Y., Kovalenko, I.: Enhancing Human-Robot Collaborative Assembly in Manufacturing Systems Using Large Language Models. In: 2024 IEEE 20th International Conference on Automation Science and Engineering (CASE). pp. 2581–2587. IEEE (2024). https://doi.org/10.1109/CASE59546.2024.10711843

[15] Kang, J.H., Dhanaraj, N., Wadaskar, S., Gupta, S.K.: Using Large Language Models to Generate and Apply Contingency Handling Procedures in Collaborative Assembly Applications. In: 2024 IEEE International Conference on Robotics and Automation (ICRA). pp. 15585–15592. IEEE (2024). https://doi.org/10.1109/ICRA57147.2024.10610875

[16] Salvagno, M., Taccone, F.S., Gerli, A.G.: Artificial intelligence hallucinations. Crit Care. 27, 180 (2023). https://doi.org/10.1186/s13054-023-04473-y

[17] Ashqar, H.I., Alhadidi, T.I., Elhenawy, M., Khanfar, N.O.: Leveraging Multimodal Large Language Models (MLLMs) for Enhanced Object Detection and Scene Understanding in Thermal Images for Autonomous Driving Systems. Automation. 5, 508–526 (2024). https://doi.org/10.3390/automation5040029

[18] Wang, Z., Kiyokawa, T., Sera, I., Yamanobe, N., Wan, W., Harada, K.: Error Correction in Robotic Assembly Planning From Graphical Instruction Manuals. IEEE Access. 11, 107276–107286 (2023). https://doi.org/10.1109/ACCESS.2023.3319822

[19] Langer, M., Baum, K., Schlicker, N.: Effective Human Oversight of AI-Based Systems: A Signal Detection Perspective on the Detection of Inaccurate and Unfair Outputs. Minds Mach (Dordr). 35, 1 (2024). https://doi.org/10.1007/s11023-024-09701-0

[20] Aleixo, E.L., Colonna, J.G., Cristo, M., Fernandes, E.: Catastrophic Forgetting in Deep Learning: A Comprehensive Taxonomy. Journal of the Brazilian Computer Society. 30, (2024). https://doi.org/10.5753/jbcs.2024.3966

[21] Fan, Y., Simeone, A., Antonelli, D., Caggiano, A., Priarone, P.C., Settineri, L.: A framework for digital assembly instructions as a step towards manufacturing inclusiveness. Procedia CIRP. 132, 116–121 (2025). https://doi.org/10.1016/j.procir.2025.01.020

[22] Chen, H., Li, S., Fan, J., Duan, A., Yang, C., Navarro-Alarcon, D., Zheng, P.: Human-in-the-Loop Robot Learning for Smart Manufacturing: A Human-Centric Perspective. IEEE Transactions on Automation Science and Engineering. 1–1 (2025). https://doi.org/10.1109/TASE.2025.3528051

[23] Iyenghar, P.: Clever Hans in the Loop? A Critical Examination of ChatGPT in a Human-in-the-Loop Framework for Machinery Functional Safety Risk Analysis. Eng. 6, 31 (2025). https://doi.org/10.3390/eng6020031

[24] Hatia, A., Doldo, T., Parrini, S., Chisci, E., Cipriani, L., Montagna, L., Lagana, G., Guenza, G., Agosta, E., Vinjolli, F., Hoxha, M., D'Amelio, C., Favaretto, N., Chisci, G.: Accuracy and Completeness of ChatGPT-Generated Information on Interceptive Orthodontics: A Multicenter Collaborative Study. J Clin Med. 13, 735 (2024). https://doi.org/10.3390/jcm13030735

[25] Woo, K.C., Simon, G.W., Akindutire, O., Aphinyanaphongs, Y., Austrian, J.S., Kim, J.G., Genes, N., Goldenring, J.A., Major, V.J., Pariente, C.S., Pineda, E.G., Kang, S.K.: Evaluation of GPT-4 ability to identify and generate patient instructions for actionable incidental radiology findings. Journal of the American Medical Informatics Association. 31, 1983–1993 (2024). https://doi.org/10.1093/jamia/ocae117.

[26] Govender, R.G.: My AI students: Evaluating the proficiency of three AI chatbots in completeness and accuracy. Contemp Educ Technol. 16, ep509 (2024). https://doi.org/10.30935/cedtech/14564

[27] Fan, Y., Simeone, A., Antonelli, D., Catalano, A.R., Priarone, P.C., Settineri, L.: A logic-assisted assembly support system to promote the inclusion of neurodiverse operators in manufacturing. Procedia CIRP. (2025). Accepted for publication.