

13th CIRP Global Web Conference (CIRPe 2025)

Automated Vision-Based Quality Control of Robotic Assembly in HMLV Manufacturing

Alexander Geiser^{*a}, Alexander Fuhrmann^a, Mekdes Hailu^b, Saif Hashem^a, Martin Benfer^a,
Gisela Lanza^a

^a *wbk Institute of Production Science, Karlsruhe Institute of Technology (KIT)*

^b *Department of Mechanical Engineering, Korean Advanced Institute of Science and Technology (KAIST)*

* Corresponding author E-mail address: alexander.geiser@kit.edu

Abstract

Robotics has advanced significantly, but small and medium-sized enterprises face challenges in adopting it in high-mix, low-volume environments. Primarily due to the manual programming required for robot integration and the time it takes for a process to run stably and produce high-quality output without rejections. The high scrap rate in unstable processes leads to low productivity, which is detrimental to competitiveness. While computer vision technologies offer some flexibility, they often require large, manually labelled datasets and only provide binary quality feedback or error classes. This paper proposes an approach to automate robotic processes and quality control using synthetic data. Computer-aided design data and assembly sequences initiate object recognition training. A quality model is automatically derived from the data, which is used to evaluate the manufactured quality during the process and to continuously improve the process via iterative learning control. The goal of this procedure is to allow for a change of variant without the need for time-consuming re-teaching by a robotics expert.

© 2025 The Authors. Published by Elsevier B.V.

This is an open access article under the CC BY license (<http://creativecommons.org/licenses/by/4.0/>)

Peer review under the responsibility of the scientific committee of the CIRPe 2025

Keywords: Robotic Assembly, High Mix Low Volume, Quality Control, ILC, 3D Point Cloud, Anomaly Detection, Synthetic Data

1. Introduction

In the face of global competition and the ongoing trend toward customization, automation plays a key role in modern manufacturing. It is causing a shift from traditional mass production to high-mix, low-volume (HMLV) production. This new model requires a high degree of flexibility and adaptability [1], [2], [3]. Although robots have been established as flexible hardware platforms, programming them in HMLV environments remains a significant challenge [1], [4]. To reduce the significant manual programming required for robotic assembly, skill libraries have been introduced [5], [6]. However, parametrizing these skills is not trivial [2], [3]. Manual parametrization, as well as self-learning systems such as reinforcement learning approaches, often results in significant scrap rates in the early stages of production [6], [7]. In HMLV manufacturing, this is particularly problematic due

to low volumes and high material costs [3], [8]. Search algorithms, such as spiral search, are used to reduce the amount of scrap through unsuccessful trials at the cost of production time [1], [6]. To utilize a search algorithm as a fallback option, the system must be able to recognize when a state has been reached that can be classified as “good” [6], [9]. A significant shortcoming of current robot programming is its inability to automatically detect and describe errors. Due to factors such as the sim-to-real gap, drift, and systematic deviations in the product and resources, e.g., caused by a batch change or system maintenance, manual reteaching (fine-tuning) is still necessary [1], [5]. This still causes machine downtime and requires the use of expensive experts [3], [5], [10]. Many automated approaches treat error detection as a binary problem, checking, for example, only for the presence of an object [9], [11], [12], [13]. Methods that can distinguish between different error classes often require a large amount of input data for training.

This data may contain errors and is often manually labeled [6], [11], [14]. In order for error detection to be used in HMLV scenarios for quality control strategies, errors must be detected automatically and described in a way that the control algorithm can understand. Current robotics approaches attempt to implement this adaptive capability via iterative learning control (ILC) [15], [16]. ILC methods are designed to optimize recurring tasks through repeated execution. They use the results of previous attempts to improve control commands for subsequent ones [17]. However, existing ILC implementations are often limited in the context of HMLV manufacturing due to a lack of repetition [18].

This work introduces a conceptual framework to improve the ability to automatically derive quality characteristics and adaptively correct errors, even within HMLV scenarios. To achieve this, errors are not described for each case; instead, they are automatically detected, and what is considered "incorrect" is deducted based on existing standards and conventions. The general tolerances specified in ISO 2768, standard in mechanical engineering, or the designer's specifications in CAD serve as a starting point for threshold values. The quality achieved can then be evaluated based on these quality areas. A perception module is required to assess quality based on this information. This module describes the geometric relationships between components and assemblies. CAD files are used to derive target parameters, which are then visually checked in the process using an RGB-D camera. The quality information obtained is then used to apply ILC. To effectively use ILC in an HMLV environment, it is applied at the skill level, where individual skills in the skill library exhibit sufficient similarity, even between product variants. This ensures that frequent repetition of skills has a positive effect, even with small batch sizes of individual product variants. This enables the continuous improvement of assembly processes. This approach can independently react to systematic disturbances through autonomous quality description, recording and control. This reduces plant downtime, the need for expert intervention, scrap and increases efficiency.

2. Related Work

This chapter outlines the current state of research in three key areas that are crucial for developing an adaptive, quality-assured assembly system. Firstly, it examines approaches to visual defect detection that identify deviations in the assembly process. Second, it presents iterative learning control (ILC) as a method that enables robots to enhance their performance over time and prevent such errors in the future. Thirdly, it analyses the importance of geometric tolerances in defining quality standards and their application in robot control. Together, these three areas of research provide a foundation for overcoming the given challenges.

2.1. Visual Error Detection

Visual error detection in automated quality control is often performed using deep learning-based models trained on either 2D images or 3D data such as point clouds [9], [19], [20], [21], [22], [23]. A trend in recent literature is the use of object detection frameworks, such as YOLO [19], [24], [25]. One weakness of this approach is the precise detection and evaluation of critical factors, such as geometrical tolerances, orientation, or subtle defects like slight misalignments, that occur during assembly scenarios in HMLV environments, beyond simple binary detection of anomalies. This binary error detection can be approached using supervised learning methods, i.e., machine learning algorithms, which are applied when a predefined set of defect types is available in advance. This reliance on predefined classes, however, is a major downside [23]. In HMLV settings, supervised models require frequent retraining and are inherently limited in their ability to generalize to previously unseen errors, which can make them less suitable for dynamic or flexible manufacturing systems. Unsupervised approaches offer a compelling alternative, particularly for assemblies where the range of possible malfunctions is either unknown or too vast to define. These methods typically compare input data to a reference, representing normality and flagging deviations above a certain threshold as potential anomalies [9], [19], [23], [26]. Training-based and untrained methods are employed to model normality, but they risk overlooking minor anomalies or poorly reconstructing complex surfaces [19], [23]. One key limitation is the inability to identify the specific type of malfunction or error [23]. Both the supervised and unsupervised approaches typically require substantial amounts of training data, the collection of which can be labor-intensive and time-consuming [19], [23]. To mitigate this challenge, the synthetic generation of training samples can be used, significantly reducing the manual effort involved in data acquisition [12], [20]. A relevant untrained method is the Iterative Closest Point (ICP) algorithm. This geometric matching enables precise detection of misalignments or positional errors by directly comparing the scanned part to its ideal configuration, offering a label-free alternative for identifying assembly defects and requiring no prior training or datasets [27], [28]. However, a major downside of this method is the extensive calculation time needed for real-time detection [19].

2.2. Iterative Learning Control

Iterative Learning Control (ILC) is a technique that improves the performance of systems executing the same task repeatedly. The core principle is to use information from previous iterations to update the control signal for the subsequent one. The general update law can be expressed as (1):

$$u_{k+1} = u_k + \Delta(u_k, e_k) \quad (1)$$

Here, u_k represents the control input for the k -th iteration, e_k is the measured tracking error from that same iteration (i.e., the

difference between the desired and actual output), and $\Delta(u_k, e_k)$ is the correction term and is a function of the error and input recorded over the previous trial. Through this iterative refinement, the system's output is driven closer to the desired target with each repetition. [17] ILC has emerged as a robust method for improving task execution in robotics through repeated trial-and-error, particularly in structured and repetitive settings. It has been widely used to enhance the precision of trajectory tracking, force control, and contact-based operations such as screwing and insertion [29]. Early applications focused on high-volume manufacturing tasks, where ILC demonstrated rapid convergence to high accuracy, often outperforming classical feedback control by correcting repeatable error components between iterations. However, standard ILC approaches assume identical task repetition, which poses a limitation in HMLV production environments [30]. In such settings, product variability and low repetition rates hinder the convergence needed for classical ILC to be effective. To address this, recent studies propose modularizing ILC at the skill level, allowing learned corrections for tasks like gripping, pick-and-place, or alignment to be reused across variants [31]. These skill-level approaches align well with modern robotics frameworks that decompose tasks into reusable, parameterized primitives. By enabling localized learning within individual skills rather than complete trajectories, such methods maintain quality while accommodating variation.

2.3. Tolerance in Robotic Assembly

Compliance with geometric tolerances is a crucial factor in determining the quality and functionality of mechanical assemblies. Traditionally, tolerance analysis has been considered statistically and separately from direct robot control, as demonstrated in early work on representing geometric deviations using matrix transformations. [32] Newer approaches aim to make tolerance information directly usable for robot planning from digital product data. A promising approach is the extraction of semantic Geometric Dimensioning and Tolerancing (GD&T) information from standardized CAD exchange formats such as STEP AP242. Such data can form the basis for automatic tolerance assignment and assembly analysis, which significantly reduces manual effort. This information is used to plan robot paths and evaluate the feasibility of assembly tasks, such as pick-and-place operations, taking tolerance chains into account. [8], [33], [34], [35] In practical applications, the focus is often on solving specific, tolerance-critical tasks. A classic example is the peg-in-hole process, where even minimal deviations can lead to failure. Methods have been developed for this purpose that operate based on contact condition estimates or passive alignment principles for extremely tight clearances. The automation of robot planning using product and manufacturing information is also an active field of research aimed at closing the gap between design and manufacturing. [36], [37], [38]

Although these studies highlight the progress made in using tolerance information for robot assembly control, integration into a closed-loop control system that detects deviations and corrects the process adaptively remains a key challenge. Especially for HMLV scenarios, a framework is needed that uses tolerance assessments not only for error detection but also as continuous feedback for learning-based control approaches such as ILC.

3. Research Approach

This chapter outlines the research approach, which is divided into four interconnected modules. These modules form a closed loop for an adaptive robotic assembly system that operates autonomously in dynamic manufacturing environments.

3.1. Perception Module

The initial phase of this approach focuses on the precise and automated detection of the environment and components. Automatic initial calibration is performed at the commencement of the process chain, wherein the robot base is detected via point cloud matching using a Zivid RGB-D camera. This process, as shown in [39], establishes a foundational reference for subsequent measurements and movements. To address the challenge of acquiring extensive annotated datasets in HMLV manufacturing, synthetic RGB-D datasets are automatically generated from CAD data within NVIDIA Isaac Sim (Fig. 1). The content of the dataset is structured according to the BOP standard, the Benchmark for 6D Object Pose Estimation. These virtual datasets are then utilized for training deep learning object recognition networks, facilitating the development of robust models without the need for laborious manual data collection. The virtually trained networks are subsequently deployed in the real production environment for rapid detection of approximate component positions and intermediate states, represented as bounding



Fig. 1. Output from the simulation environment; Top left: Virtual RGB image, top right: Depth information in grayscale, bottom left: ground truth information, bottom right: visualization of the automatically labeled image with randomized object count, object position, lighting, and material properties.

boxes. This serves as an initial estimate, reducing the search space for more precise methods. A focused Region of Interest (RoI) is then defined based on the specific task, encompassing the relevant objects, such as the component to be assembled and the intermediate state. This reduction of the RoI significantly minimizes computational effort and enhances processing speed. For highly accurate position determination, 3D matching methods, such as the Iterative Closest Point (ICP) algorithm, are employed. The coarse position obtained from the initial object detection serves as the starting point for ICP, thereby accelerating convergence and improving accuracy in determining the exact positions of components and intermediate states. The complete process, including all steps, can be seen in Fig. 2. Following each assembly step, a new image is captured using the RGB-D camera. This image serves to select objects for the subsequent assembly step and to evaluate the success of the preceding assembly. To precisely isolate the newly assembled component and determine its position, the point clouds are segmented by utilizing the difference formation. The pre-assembly point cloud, lacking the component, is compared to the post-assembly point cloud, which includes the component. The resulting difference represents the newly added component at its current position. Subsequently, the isolated points of the component are subtracted from the intermediate state. This is followed by matching the CAD data of both the component and the intermediate assembly within the segmented point clouds, with the expected pose serving as the initial starting position for ICP (Fig. 3). This enables the precise determination of the component's position relative to the intermediate assembly.

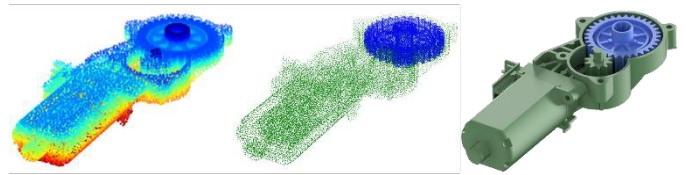


Fig. 2. Process for determining the position of the newly assembled component in the current intermediate assembly. From left to right: Point cloud > Segmented point cloud > CAD data match

3.2. Quality Module

The quality model enables the robotic system to objectively assess assembly success and quantify deviations from the expected standard. The actual transformation (rotation and translation) as seen in equation (2) is calculated from the precisely determined positions of the component and the intermediate assembly, derived from sensor data.

$$c_{T_i} = \begin{pmatrix} n_x & o_x & a_x & t_x \\ n_y & o_y & a_y & t_y \\ n_z & o_z & a_z & t_z \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad (2)$$

The target transformation (rotation and translation) is automatically extracted directly from the CAD data. The difference between the target and actual transformations is then calculated in the value comparison and converted into conventional descriptions of position tolerances, such as offsets in X, Y, Z, and rotational deviations, with the equations shown in (3) and (4).

$$c_{T_{Diff}} = \begin{pmatrix} n_x & o_x & a_x & t_x \\ n_y & o_y & a_y & t_y \\ n_z & o_z & a_z & t_z \\ 0 & 0 & 0 & 1 \end{pmatrix} = \begin{pmatrix} czcy & czsysx - szcx & czsycx + szsx & x \\ szcy & szsysx + czcx & szsycx - czsx & y \\ -sy & cysx & cycz & z \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad (3)$$

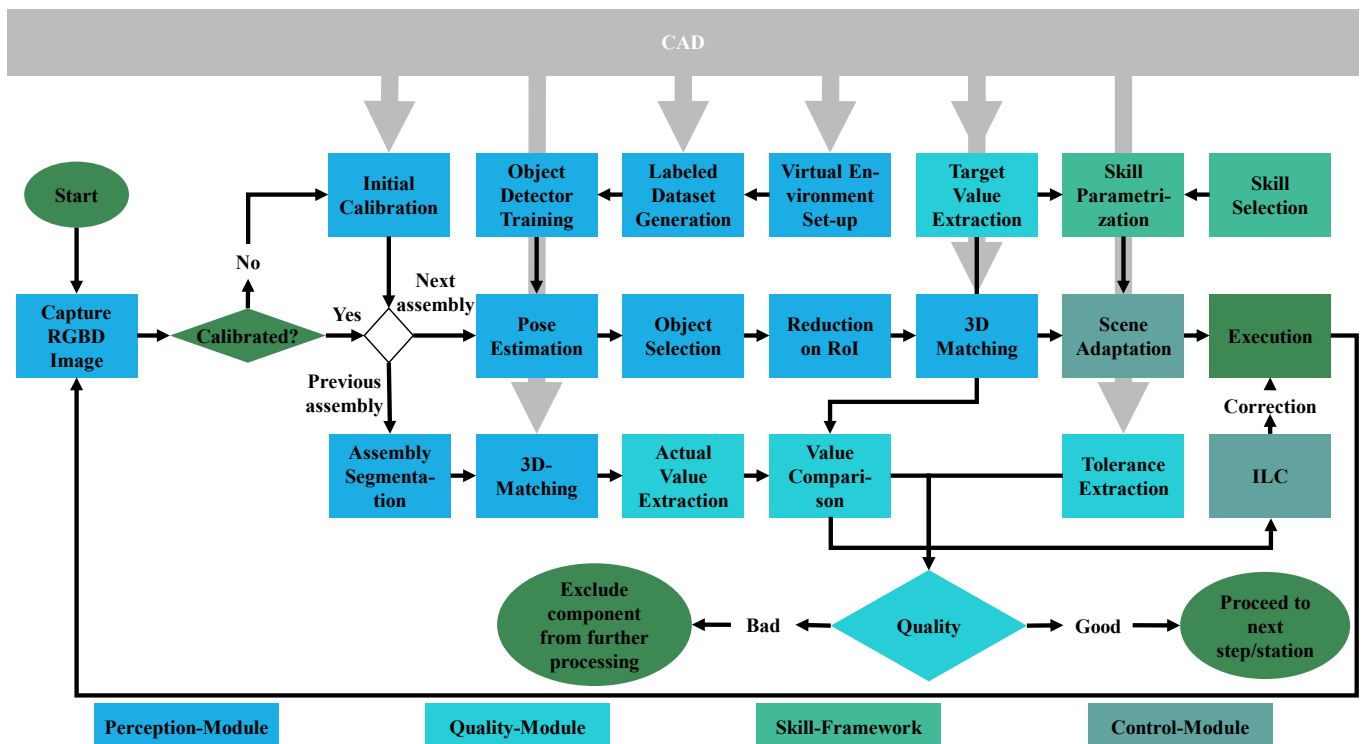


Fig. 3. The flowchart shows how the presented concept improves the next assembly step iteratively based on deviations from the previous step. It also shows how changes in the scene and systematic errors can be addressed. The necessary information is automatically derived from the components' CAD data

$$\begin{aligned}
 x = t_x & \quad \varphi_x = \text{atan2}(\pm o_z, \pm a_z) \\
 y = t_y & \quad \varphi_y = \text{atan2}(\mp n_z, \pm \sqrt{n_x^2 + n_y^2}) \\
 z = t_z & \quad \varphi_z = \text{atan2}(\pm n_y, \pm n_x)
 \end{aligned}
 \quad (4)$$

Threshold values for decision-making are derived from commonly used mechanical engineering standards and conventions, such as ISO 2768, or designer specifications embedded within the CAD data. These thresholds determine whether the assembly quality is sufficient to proceed with the process, facilitating early error detection and preventing the production of scrap. In contrast to binary error definitions prevalent in current research, these metrics provide continuous feedback, which is subsequently utilized for ILC, even when the assembly falls within tolerance.

3.3. Skill Framework

To effectively manage the complexity of robot programming in HMLV environments and maximize the benefits of ILC, a skill framework is employed. Complex assembly processes are decomposed into discrete skills, each describing an elementary robot operation that exhibits similarity across different products and variants. The granularity of these skills ensures that their underlying operations are sufficiently homogeneous across diverse products and variants, which is crucial for practical ILC application, even with small batch sizes. The limited complexity of each skill allows for positive influence through ILC. Controllable parameters are identified at the skill level, designated for optimization by ILC. The initial parametrization of the skills is set according to the target values and subsequently adapted to the scene based on the results of the perception module.

3.4. Control Module

The control mechanism is a critical element that endows the robotic system with adaptive behavior. Error feedback from the quality model, even when within tolerance, is transmitted to the Iterative Learning Control (ILC). This distinguishes the presented approach from existing ILC methods, which often learn exclusively from explicit error events. This enables continuous improvement and the early correction of systematic errors, such as inaccuracies in robot positioning accuracy or deviations in initial calibration. While ILC typically thrives on frequent repetitions, which are uncommon in HMLV environments due to small batch sizes, this challenge is addressed by applying ILC at the individual skill level. Since process skills, by their limited complexity and scope, recur across different products and variants, ILC can be deployed more effectively. Since a skill is primarily related to the resource, the inaccuracies of the resource can be learned over many iterations. This information remains valid even in the face of frequent changes to the overall product or its variants. Learning at the skill level enables the robotic system to generalize acquired knowledge to similar skills in other

products and variants, leading to a significant reduction in programming effort and rapid adaptation to new scenarios.

4. Conclusion

This work presents a concept for advancing robotic assembly in HMLV manufacturing, combining CAD-driven perception, standards-based quality evaluation, and skill-level ILC. By providing robots with iterative, non-binary feedback based directly on the designer's intentions, it can be possible to implement a control strategy that achieves flexible automation. The goal is to minimize plant downtime, the need for expert intervention and scrap, even with high product variety by reducing the need for re-teaching. The system continuously learns and compensates for deviations without the need to create a system model. The work primarily focuses on assembly situations in which assembly tolerances are more important than component tolerances. However, if the process fails completely, for instance, if the component tips over or falls out after assembly, the perception module cannot provide meaningful feedback to the control module. But the approach primarily focuses on improving object quality rather than initializing the process. However, even in the intended scenarios, significant challenges remain, including limitations in sim-to-real transfer, the completeness and reliability of CAD-encoded tolerances, and the generalizability of skill abstractions across diverse assemblies. Addressing these open questions will be essential for progressing from promising prototypes to robust, real-world solutions. Nevertheless, this research represents a step towards self-improving, data-driven robotics for flexible factories of the future. The modules presented will be developed in the next stage. The perception module of the framework is currently in the development and integration phase. A demonstrator will be built and test components developed for validation purposes. Results and evaluations will be published in future papers.

Acknowledgements

This research and development project is funded by the Federal Ministry of Research, Technology and Space (BMFTR) within the ‘‘SME-innovative: The Future of Value Creation’’ funding measure (funding number 02K24K021) and managed by the Project Management Agency Karlsruhe (PTKA). The authors are responsible for the content of this publication.

References

- [1] G. Gorjup, G. Gao, A. Dwivedi, and M. Liarokapis, ‘‘A Flexible Robotic Assembly System Combining CAD Based Localization, Compliance Control, and a Multi-Modal Gripper,’’ *IEEE Robot. Autom. Lett.*, vol. 6, no. 4, pp. 8639–8646, Oct. 2021, doi: 10.1109/LRA.2021.3113626.
- [2] M. H. Arbo, Y. Pane, E. Aertbelien, and W. Decre, ‘‘A System Architecture for Constraint-Based Robotic Assembly with CAD Information,’’ in *2018 IEEE 14th International Conference on Automation Science and Engineering (CASE)*, Munich: IEEE, Aug. 2018, pp. 690–696. doi: 10.1109/COASE.2018.8560450.

- [3] S. Kurscheid and R. Daub, “Automated Robot Skill Parameterization Based On Task Specifications,” in *2024 IEEE 29th International Conference on Emerging Technologies and Factory Automation (ETFA)*, Padova, Italy: IEEE, Sept. 2024, pp. 01–04. doi: 10.1109/ETFA61755.2024.10711102.
- [4] L. C. Sørensen, R. S. Andersen, C. Schou, and D. Kraft, “Automatic parameter learning for easy instruction of industrial collaborative robots,” in *2018 IEEE International Conference on Industrial Technology (ICIT)*, Feb. 2018, pp. 87–92. doi: 10.1109/ICIT.2018.8352157.
- [5] T. Lohi, S. Soutukorva, and T. Heikkilä, “Programming of Skill-Based Robots,” in *2024 IEEE 19th Conference on Industrial Electronics and Applications (ICIEA)*, Kristiansand, Norway: IEEE, Aug. 2024, pp. 1–7. doi: 10.1109/ICIEA61579.2024.10664981.
- [6] A. Lämmle et al., “Simulation-based Learning of the Peg-in-Hole Process Using Robot-Skills,” in *2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Oct. 2022, pp. 9340–9346. doi: 10.1109/IROS47612.2022.9982212.
- [7] B. Tang et al., “AutoMate: Specialist and Generalist Assembly Policies over Diverse Geometries,” Aug. 01, 2024, arXiv: arXiv:2407.08028. doi: 10.48550/arXiv.2407.08028.
- [8] S. K. Mohammed, M. H. Arbo, and L. Tingelstad, “Using semantic Geometric Dimensioning and Tolerancing (GD &T) information from STEP AP242 neutral exchange files for robotic applications,” *Int. J. Interact. Des. Manuf. IJIDeM*, vol. 18, no. 9, pp. 6587–6603, Nov. 2024, doi: 10.1007/s12008-023-01242-7.
- [9] V. Dosljak, I. Jovančević, and J.-J. Orteu, “Visual inspection of complex mechanical assemblies based on Siamese networks for 3D point clouds,” in *Sixteenth International Conference on Quality Control by Artificial Vision*, J.-J. Orteu and I. Jovančević, Eds., Albi, France: SPIE, July 2023, p. 40. doi: 10.1117/12.2692751.
- [10] D. Bargmann, W. Kraus, and M. F. Huber, “Enabling Maintainability of Robot Programs in Assembly by Extracting Compositions of Force- and Position-Based Robot Skills from Learning-from-Demonstration Models,” in *2024 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Abu Dhabi, United Arab Emirates: IEEE, Oct. 2024, pp. 9227–9234. doi: 10.1109/IROS58592.2024.10802802.
- [11] X. Zhu, H. Manamasa, J. L. Jiménez Sánchez, A. Maki, and L. Hanson, “Automatic assembly quality inspection based on an unsupervised point cloud domain adaptation model,” *Procedia CIRP*, vol. 104, pp. 1801–1806, 2021, doi: 10.1016/j.procir.2021.11.304.
- [12] C. Huber, D. Knoll, and M. Guthe, “Fully-Synthetic Training for Visual Quality Inspection in Automotive Production,” Mar. 12, 2025, arXiv: arXiv:2503.09354. doi: 10.48550/arXiv.2503.09354.
- [13] F. P. Basamaklis, A. C. Bavelos, D. Dimosthenopoulos, A. Papavasileiou, and S. Makris, “Deep object detection framework for automated quality inspection in assembly operations,” *Procedia CIRP*, vol. 115, pp. 166–171, 2022, doi: 10.1016/j.procir.2022.10.068.
- [14] N. Karaklas, A. Papavasileiou, and S. Makris, “Empowering Precision-Guided Automotive Assembly Operations: A Flexible Robot Vision Framework,” *Procedia CIRP*, vol. 127, pp. 50–55, Jan. 2024, doi: 10.1016/j.procir.2024.07.010.
- [15] L. Halt, F. Nagele, P. Tenbrock, and A. Pott, “Intuitive Constraint-Based Robot Programming for Robotic Assembly Tasks,” in *2018 IEEE International Conference on Robotics and Automation (ICRA)*, Brisbane, QLD, Australia: IEEE, May 2018, pp. 520–526. doi: 10.1109/ICRA.2018.8462882.
- [16] A. Vemula, W. Sun, M. Likhachev, and J. A. Bagnell, “On the Effectiveness of Iterative Learning Control,” in *Proceedings of The 4th Annual Learning for Dynamics and Control Conference*, PMLR, May 2022, pp. 47–58. Accessed: June 12, 2025. [Online]. Available: <https://proceedings.mlr.press/v168/vemula22a.html>
- [17] C. T. Freeman, E. Rogers, J. H. Burridge, A.-M. Hughes, and K. L. Meadmore, “Iterative Learning Control—An Overview,” in *Iterative Learning Control for Electrical Stimulation and Stroke Rehabilitation*, in SpringerBriefs in Electrical and Computer Engineering, London: Springer London, 2015, pp. 3–16. doi: 10.1007/978-1-4471-6726-6_2.
- [18] L. Xu, W. Zhong, J. Lu, F. Gao, F. Qian, and Z. Cao, “Learning of Iterative Learning Control for Flexible Manufacturing of Batch Processes,” *ACS Omega*, vol. 7, no. 23, pp. 19939–19947, May 2022, doi: 10.1021/acsomega.2c01741.
- [19] J. Du, C. Tao, X. Cao, and F. Tsung, “3D vision-based anomaly detection in manufacturing: A survey,” *Front. Eng. Manag.*, vol. 12, no. 2, pp. 343–360, June 2025, doi: 10.1007/s42524-025-4189-9.
- [20] X. Zhu, P. Mårtensson, L. Hanson, M. Björkman, and A. Maki, “Automated assembly quality inspection by deep learning with 2D and 3D synthetic CAD data,” *J. Intell. Manuf.*, vol. 36, no. 4, pp. 2567–2582, Apr. 2025, doi: 10.1007/s10845-024-02375-6.
- [21] M. Demetgul, Q. Zheng, I. N. Tansel, and J. Fleischer, “Monitoring the misalignment of machine tools with autoencoders after they are trained with transfer learning data,” *Int. J. Adv. Manuf. Technol.*, vol. 128, no. 7, pp. 3357–3373, Oct. 2023, doi: 10.1007/s00170-023-12060-2.
- [22] X. Pan and T. Y. Yang, “3D vision-based bolt loosening assessment using photogrammetry, deep neural networks, and 3D point-cloud processing,” *J. Build. Eng.*, vol. 70, p. 106326, July 2023, doi: 10.1016/j.jobte.2023.106326.
- [23] Q. Yang and R. Guo, “An Unsupervised Method for Industrial Image Anomaly Detection with Vision Transformer-Based Autoencoder,” *Sensors*, vol. 24, no. 8, 2024, doi: 10.3390/s24082440.
- [24] L. Huo, Y. Liu, Y. Yang, Z. Zhuang, and M. Sun, “Review: Research on product surface quality inspection technology based on 3D point cloud,” *Adv. Mech. Eng.*, vol. 15, no. 3, p. 16878132231159523, 2023, doi: 10.1177/16878132231159523.
- [25] F. Stamer, R. Jachemich, S. Puttero, E. Verna, and M. Galetto, “Integrative inspection methodology for enhanced PCB remanufacturing using artificial intelligence,” *12th CIRP Glob. Web Conf. CIRPe 2024*, vol. 132, pp. 227–232, Jan. 2025, doi: 10.1016/j.procir.2025.01.038.
- [26] B. Zhao et al., “PointCore: Efficient Unsupervised Point Cloud Anomaly Detector Using Local-Global Features,” Mar. 04, 2024, arXiv: arXiv:2403.01804. doi: 10.48550/arXiv.2403.01804.
- [27] P. J. Besl and N. D. McKay, “A method for registration of 3-D shapes,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 14, no. 2, pp. 239–256, 1992, doi: 10.1109/34.121791.
- [28] J. Zhang, Y. Yao, and B. Deng, “Fast and Robust Iterative Closest Point,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 44, no. 7, pp. 3450–3466, 2022, doi: 10.1109/TPAMI.2021.3054619.
- [29] K. L. Sorensen, W. Singhose, and S. Dickerson, “A controller enabling precise positioning and sway reduction in bridge and gantry cranes,” *Spec. Issue Award Win. Appl.*, vol. 15, no. 7, pp. 825–837, July 2007, doi: 10.1016/j.conengprac.2006.03.005.
- [30] F. Boeren, A. Bareja, T. Kok, and T. Oomen, “Frequency-Domain ILC Approach for Repeating and Varying Tasks: With Application to Semiconductor Bonding Equipment,” *IEEE/ASME Trans. Mechatron.*, vol. 21, no. 6, pp. 2716–2727, Dec. 2016, doi: 10.1109/TMECH.2016.2577139.
- [31] Y. Li, X. Zhou, J. Zhong, and X. Li, “Robotic Impedance Learning for Robot-Assisted Physical Training,” *Front. Robot. AI*, vol. 6, Aug. 2019, doi: 10.3389/frobt.2019.00078.
- [32] D. E. Whitney, O. L. Gilbert, and M. Jastrzebski, “Representation of geometric variations using matrix transforms for statistical tolerance analysis in assemblies,” *Res. Eng. Des.*, vol. 6, no. 4, pp. 191–210, Dec. 1994, doi: 10.1007/BF01608399.
- [33] P. Haghghi, P. Mohan, N. Kalish, P. Vemulapalli, J. J. Shah, and J. K. Davidson, “Toward Automatic Tolerancing of Mechanical Assemblies: First-Order GD&T Schema Development and Tolerance Allocation,” *J. Comput. Inf. Sci. Eng.*, vol. 15, no. 041003, Aug. 2015, doi: 10.1115/1.4030939.
- [34] P. Mohan, P. Haghghi, P. Vemulapalli, N. Kalish, J. J. Shah, and J. K. Davidson, “Toward Automatic Tolerancing of Mechanical Assemblies: Assembly Analyses,” *J. Comput. Inf. Sci. Eng.*, vol. 14, no. 041009, Oct. 2014, doi: 10.1115/1.4028592.
- [35] B. Tipary and G. Erdős, “Tolerance analysis for robotic pick-and-place operations,” *Int. J. Adv. Manuf. Technol.*, vol. 117, no. 5–6, pp. 1405–1426, Nov. 2021, doi: 10.1007/s00170-021-07672-5.
- [36] H. Lee, S. Park, K. Jang, S. Kim, and J. Park, “Contact State Estimation for Peg-in-Hole Assembly Using Gaussian Mixture Model,” *IEEE Robot. Autom. Lett.*, vol. 7, no. 2, pp. 3349–3356, Apr. 2022, doi: 10.1109/ra.2022.3146949.
- [37] J. Takahashi, T. Fukukawa, and T. Fukuda, “Passive Alignment Principle for Robotic Assembly between a ring and a shaft with Extremely Narrow Clearance,” *IEEE/ASME Trans. Mechatron.*, pp. 1–1, 2015, doi: 10.1109/tmech.2015.2448639.
- [38] F. M. Proctor, G. Van Der Hoorn, and R. Lipman, “Automating Robot Planning Using Product and Manufacturing Information,” *Procedia CIRP*, vol. 43, pp. 208–213, 2016, doi: 10.1016/j.procir.2016.02.139.
- [39] L. Li, X. Yang, R. Wang, and X. Zhang, “Automatic Robot Hand-Eye Calibration Enabled by Learning-Based 3D Vision,” *J. Intell. Robot. Syst.*, vol. 110, no. 3, p. 130, Sept. 2024, doi: 10.1007/s10846-024-02166-4.