https://doi.org/10.36315/2025v2end018
p-ISSN: 2184-044X e-ISSN: 2184-1489 ISBN: 978-989-35728-8-7

EXPLORING THE POTENTIALS AND LIMITATIONS OF AI-BASED CAD DESIGN FOR MECHANICAL ENGINEERING

Michael Wendland
Dr., Department of Mechanical Engineering, Westphalian University of Applied Science, Bocholt (Germany)

Abstract

This paper investigates the integration of large language models (LLMs), specifically ChatGPT, with CAD software to support the generation of 3D models for mechanical engineering applications. A literature review of recent AI-to-CAD research identifies current capabilities and gaps regarding CAD integration, parametric modeling, and adherence to engineering norms. A proof-of-concept case study demonstrates how ChatGPT can generate parametric Python scripts for Fusion 360 across four mechanical test cases with varying complexity. The results show that AI-generated models can reproduce simple standardized components, support parametric design, and aid in educational settings by prompting users to explicitly define design intent. However, challenges persist with spatial reasoning, complex geometry, and integration depth. The study concludes that while current LLMs offer value for design automation and CAD education, significant development is needed to support complex and norm-driven engineering design workflows.

Keywords: *AI-assisted CAD design, AI design automation, LLM, engineering education.*

1. Introduction

The recent advances in Artificial Intelligence (AI) have impacted many engineering fields in academia as well as in industry. In both cases, AI is sometimes seen as a silver bullet to evade the ever-increasing time, cost and quality constraints. While this has still to be seen, the general potential for increasing efficiency and the overall level of automation remains undisputed.

This paper explores the publicly available AI-tool ChatGPT, which is a Large-Language-Model (LLM), in the context of Computer-Aided Design (CAD) for Mechanical Engineering. Since the traditional creation of 3D-CAD models for designing and manufacturing mechanical parts is a labor-intense manual process that requires extensive engineering domain knowledge and modelling skills, only limited automation is available and mostly restricted to predefined, standardised parts. But for the vast majority of all other mechanical parts, there is a strong need for automation and support in the CAD design process. This paper investigates these potentials and limitations of coupling a LLM with CAD-software to support design engineers. Three main research questions shall be answered:

1. Which research approaches with AI-tools for creating 3D-CAD models are currently available? What are advantages / limitations of the approaches for Mechanical Engineering?
2. How can a state-of-the-art LLM be used to help generate mechanical parts with CAD?
3. Which potentials does coupling a LLM with CAD-Software offer and which barriers exist?

2. Analysis of existing research for creating 3D-CAD models in Mechanical Engineering

Due to the fast development in AI-research, this analysis is not meant to be comprehensive but rather presents an excerpt of the current research state in the field of coupling AI2CAD and aims to give an overview. In order to evaluate existing approaches and AI-tools for their usage in the mechanical engineering domain, the following criteria with points ranging from 1-3 (0=no, 1=low, 2=medium, 3=high) are defined:

- *Integration Level CAD<->AI* (1=manual, 2=semi-automatic+manual input, 3=automatic)
- *Technical Maturity* (1=very basic, 2=still prototype, 3=professional tool)
- *Library of Mechanical Elements* (1= no library, 2=few elements, 3=comprehensive library)
- *Geometric Modification* (1=no change, 2=few changes, 3=complete parametric design)
- *Integration of Drawing and Part Norms* (1=no support, 2=few norms, 3=comprehensive)

In the first approach (Alrashedy et al., 2024), an iterative workflow is described that links a LLM with a vision-based feedback loop to generate code for a CAD environment. A text prompt is used to create an OpenSCAD model, which is then rendered and visually evaluated by a vision-language model. Discrepancies between the target and the generated geometry lead to corrective prompts, repeating the cycle until the model meets expectations. While the system automates certain steps, human intervention remains necessary for oversight. The approach is currently limited to simple parts and academic test cases. Domain-specific design logic, such as the use of standardized machine elements, is not addressed.

In the second approach (Khan et al., 2024), a method for generating CAD models based on natural language descriptions using a transformer-based architecture is proposed. The system receives a text prompt and returns a sequence of modeling operations that can be interpreted by a CAD backend. A dataset of paired CAD sequences and associated prompts was created for this purpose. The method focuses on reproducing geometric primitives and basic construction logic, such as extrusions or sketches. More complex constraints, tolerancing, or domain-specific functions are not covered. The model produces varying results depending on phrasing and manual post-processing is typically required. While the study highlights the feasibility of text-to-CAD translation, the technical depth remains limited and it does not meet the requirements of the mechanical design domain.

The third concept of Li et al. presents an approach for CAD model generation that combines GPT-4 with a scripted design logic (Li et al., 2024). Based on a textual description, the system generates a CAD script, primarily using Python and CadQuery, to produce simple 3D parts such as gears, nuts, and shafts. The framework includes visual inspection capabilities via image rendering, enabling limited model validation. The solution is restricted to elementary mechanical components with standard topology and does not generalize to complex parts or assemblies. The integration with CAD environments is indirect, and the system is not directly designed for iterative adaptation.

Mallis et al. (2024) propose a tool-augmented agent framework that connects a vision-language model with CAD operations via the FreeCAD Python API (Mallis et al., 2024). The system decomposes user prompts into sub-tasks, which are sequentially executed by external tools such as geometric modeling functions or knowledge retrieval components. The results, e.g., generated shapes or constraint feedback, are returned to the LLM for further processing. This approach allows iterative refinement of the model. From a technical viewpoint, this setup moves closer to real CAD workflows. Mechanical engineering knowledge like existing norms, load paths, standardized parts or interfaces are not embedded.

The fifth paper (Nelson et al., 2023) investigates the use of ChatGPT for generating OpenSCAD code to design microfluidic devices. The study follows a conversational workflow in which the user iteratively refines text prompts based on ChatGPT’s code outputs and their renderings. Several basic geometries are realized, including spiral channels and splitters, which are then 3D printed and tested for fluid flow. The authors highlight the system’s ability to recover from syntax errors and produce human-readable code. However, the setup offers currently no integration into other CAD software than OpenSCAD and is not applying any existing norms or standardized machine elements from the mechanical engineering domain.

The sixth research approach of Wang et al. does not focus on a single solution, but compares different models with regard to their usability and introduces a framework named CADFusion (Wang et al., 2025). This alternating training approach, switching between two stages, seeks to balance textual and visual learning signals. While the method shows promise in generating basic CAD models from textual descriptions, it currently lacks integration with standard CAD software and does not address complex design constraints or domain-specific requirements.

The last paper of Yavartanoo et al. proposes a pipeline that translates textual part descriptions into 3D CAD models by leveraging image generation and projection techniques (Yavartanoo et al., 2024). The system first uses a diffusion-based model to create isometric images from text prompts. These images are then converted into orthographic projections according to drawing norms (top, front and side view), which serve as the basis for reconstructing 3D geometry. While the approach effectively demonstrates the feasibility of generating simple mechanical components from textual descriptions, it currently lacks integration with standard CAD software and does not address complex design constraints or uses standardized machine elements or their design norms.

Figure 1 gives an overview of the selected research and summarizes the evaluation. As noted before, there is far more research available on LLM and CAD integration. However, the approaches shown above were selected because they are focused on practical examples and directly apply AI2CAD in the mechanical domain, which is the main target of this paper. More fundamental research which also covers other domains can e.g., be found in

- Ma et al. (2023) for assisting engineering designers via LLM in the concept phase
- Filippi (2023) for general concept generation in electronics
- Ege et al. (2024) for a discussion of strength and weaknesses of LLM against humans

Regarding the presented research and the first criteria of the integration level of AI and CAD, most approaches are basic with the exception of Mallis et al., Nelson et al. and Yavartanoo et al., because they link to existing CAD software (FreeCAD, OPENSCAD, Photo2CAD). Yet none of this software is considered in an industrial setting. Since all approaches represent current research topics, the technical maturity of the solutions ranges from basic to prototype-level at maximum. Regarding the mechanical engineering domain, none of the examined papers contains a library of standardized, mechanical elements that are typically used in an industrial or professional setting. Focusing on the possibility of parametric modification of the created CAD models, all approaches focus on creating unique models which are not directly suitable for parametric modification. Regarding the last criteria of norm integration, none of the approaches tried to integrate typical norms for standardized elements of the mechanical engineering domain.

Figure 1. Evaluated research approaches with AI-usage for CAD.

Research Approach \ Evaluation Criteria	Integration Level CAD<->AI	Technical Maturity of Solution	Library of Mechanical Elements	Parametric Modification	Integration of Norms
CADCode + Vision Feedback (Alrashedy et al., 2024)	●○○	●○○	●○○	●○○	○○○
Text2CAD (Khan et al., 2024)	●○○	●●○	●○○	●●○	○○○
LLM4CAD (Li et al., 2024)	●○○	●●○	●○○	●●○	○○○
CAD-Assistant (Mallis et al., 2024)	●●○	●●○	●○○	●●○	○○○
ChatGPT + OpenSCAD (Nelson et al., 2023)	●●○	●●○	○○○	●●○	○○○
CADFusion (Wang et al., 2025)	●○○	●●○	●○○	●●○	○○○
Text2CAD via Drawings (Yavartanoo et al., 2024)	●●○	●●○	○○○	●●○	○○○

To summarize, the identified research shows a clear gap for an AI-supported CAD-design process which takes the norms of the mechanical engineering domain into account and creates parametric models, which can be easily adapted afterwards to be reused. The used CAD software does not yet reflect professional solutions like Siemens NX, Autodesk Fusion/Inventor or Catia. Therefore, an approach for coupling a LLM with an industrial CAD software is presented in the next chapter.

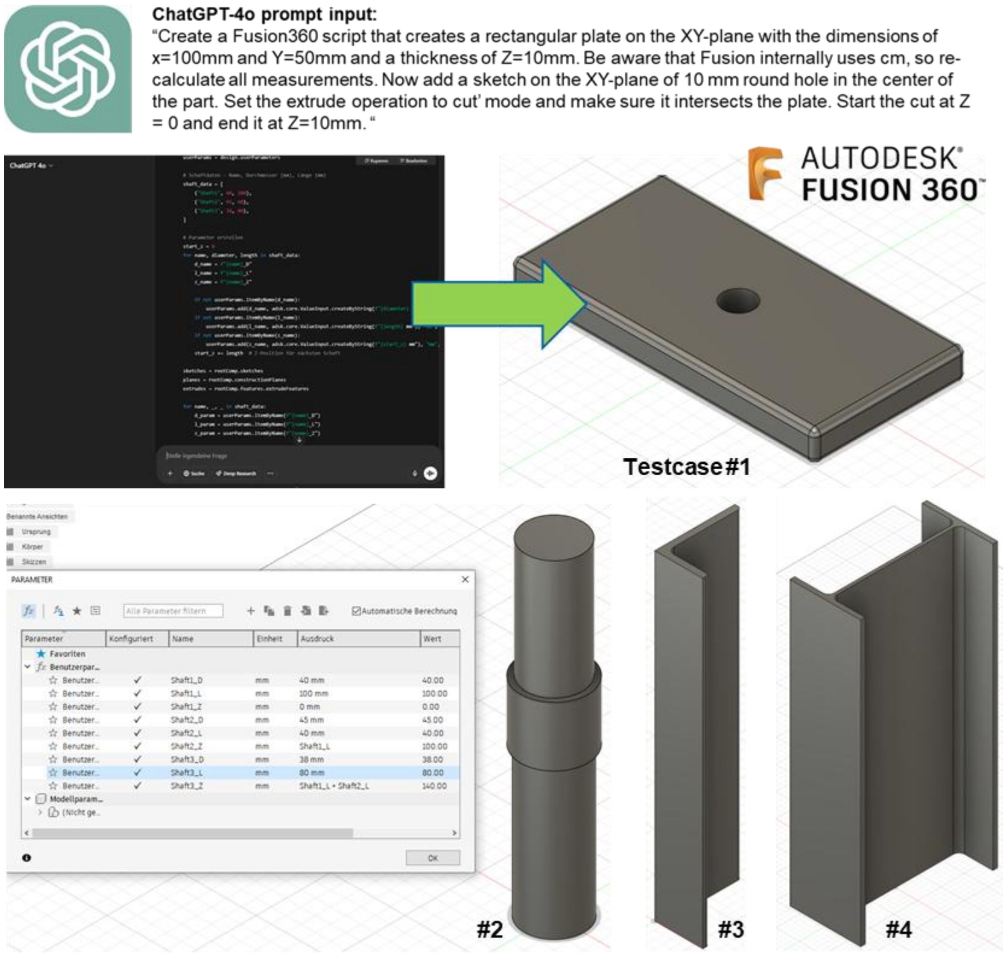
3. Approach of coupling a Large-Language-Model with CAD-design software

The real strength of coupling AI with CAD-design software is not to create unique models but to create 3D models that are parametric, which means they can be easily changed. Most experienced CAD users can create 3D models more quickly than "novices using AI" due to their spatial awareness and their understanding of the overall design and norms. When using LLMs, spatial awareness needs to be explicitly created for each element in order to locate them appropriately. This extra effort and the usual iterations to generate a bug-free, working script code by AI leads to a more time-consuming process. The huge advantage of employing AI is the possibility to create parametric 3D models that can be easily changed and reused many times, making the time investment more feasible.

For the chosen approach, the current "ChatGPT 4o" model has been selected together with Fusion 360 as the CAD software. The LLM is prompted to generate a Python script which can be directly executed in Fusion 360 and generates the CAD model. To evaluate the potentials and limits of this technique, four test cases have been created, which offer different levels of design complexity, elements and spatial arrangement.

Figure 2 shows the four test cases with varying complexity and the typical workflow with AI to generate the parametric CAD models. Test case 3 and 4 also integrate existing mechanical engineering norms for standardized parts and rely on the AI to identify the right dimensions of the parts using these norms. To showcase the potential of AI2CAD, the testcase #2 was prompted to be generated as a parametric design. This means that the design parameters like shaft diameter and length of each sections are not hardcoded with one value but rely on a parameter that can be changed afterwards (see figure 2, left side table). By changing i.e., "Shaft1_L" from 100mm to 50mm, the CAD model updates itself and the first section of the shaft is shortened.

Figure 2. Typical AI2CAD workflow and the selected four testcases.



The same additional prompt ("Make the design parametric") was successfully used on the third and fourth testcase and offers a huge advantage over standard hardcoded CAD models.

4. Identified potentials and limitations of the chosen approach

The evaluation of AI-supported CAD generation reveals several practical advantages, particularly for early-stage design tasks and instructional use in engineering education. One of the key potentials lies in the ability of LLMs to rapidly generate standardized sketches, such as cross-sections of profile systems, in accordance with common drafting norms. By simply stating the norm ID, the AI looks up the corresponding values for the sketch, which significantly accelerates the modelling of rule-based geometries. For simple to moderately complex components, parametric CAD scripts produced by the AI enable reproducible results and allow iterative refinement. Even multi-body parts can be generated successfully, provided that spatial relationships between features are clearly defined and consistently maintained.

An additional benefit is the possibility to preview the AI's geometric interpretation before finalizing the code. This mirrors the traditional process of sketching in engineering: visualizing intent before formal implementation. From a pedagogical perspective, the method proves instructive. It compels novice designers to articulate all relevant parameters of a part, making implicit knowledge explicit. For experienced users, this interaction can highlight common gaps in communication and serve as a didactic tool to better understand the difficulties faced by learners.

However, several limitations currently constrain the practicality. Complex components are difficult to describe in natural language with sufficient clarity, and the resulting prompts often lack the detail required for accurate reconstruction. It could be observed that as geometries become more intricate, the likelihood of misinterpretation by the AI increases. Decomposing the task into smaller steps and verifying each feature individually becomes necessary. The most frequent errors arise from incorrect spatial arrangement of features, suggesting that current AI models still lack robust 3D spatial reasoning capabilities.

5. Conclusion and outlook

The presented approach of coupling an AI with CAD software shows promising potential both for industrial and academic use cases that range from simple to medium complexity. One of the success factors is the spatial awareness of the model features. If the user is able to accurately describe the spatial arrangement, the AI design usually succeeds. Interestingly, this corresponds to by the author observed human performance levels of engineering students in academia. One decisive difference between novice and expert CAD students is the ability to understand and communicate the spatial arrangement of features in a model.

Furthermore, it could also be shown that existing mechanical engineering norms can be successfully built into the design process with AI. This provides great potential for creating designs that rely on standardized parts or norms. Even if the corresponding norm is not known by the user, the AI can be tasked to search for applicable norms to the given geometry, part or sketch, which massively helps novice CAD users and students alike. Moreover, the research revealed that the combination of AI & CAD can be used as teaching tool for novice CAD users or students, since it highlights missing information (e.g., dimensions or spatial arrangement) and allows the users to understand which information they have to explicitly share or express.

Further research is necessary and will be conducted with regards to the AI prompts in order to create a library of prompts for standard mechanical parts. Also, it is still unclear how a successful AI supported process could look like for more complex mechanical parts. Once these steps have been cleared, the path is open to generate complete modules or assemblies of multiple parts by AI.

References

Alrashedy, K., Tambwekar, P., Zaidi, Z., Langwasser, M., Xu, W., & Gombolay, M. (2024). *Generating CAD Code with Vision-Language Models for 3D Designs*. arXiv. Retrieved April 7, 2025, from https://arxiv.org/abs/2410.05340

Ege, D. N., Øvrebø, H. H., Stubberud, V., Berg, M. F., Elverum, C., Steinert, M., & Vestad, H. (2024). *ChatGPT as an inventor: Eliciting the strengths and weaknesses of current large language models against humans in engineering design*. arXiv. https://doi.org/10.48550/arXiv.2404.18479

Filippi, S. (2023). Measuring the Impact of ChatGPT on Fostering Concept Generation in Engineering Design. *Electronics, 16*, 3535.

Khan, M. S., Sinha, S., Sheikh, T. U., Stricker, D., Ali, S. A., & Afzal, M. Z. (2024). *Text2CAD: Generating Sequential CAD Models from Beginner-to-Expert Level Text Prompts*. arXiv. Retrieved April 7, 2025, from https://arxiv.org/abs/2409.17106

Li, Z., Srivatsan, R. A., McCormick, A., & Ingram, S. (2024). LLM4CAD: Multi-Modal Large Language Models for 3D CAD Generation. In *ASME IDETC/CIE Conference 2024*.

Ma, K., Grandi, D., McComb, C., & Goucher-Lambert, K. (2023). Conceptual Design Generation Using Large Language Models. In *ASME IDETC/CIE Conference 2023*.

Mallis, D., Karadeniz, A. S., Cavada, S., Rukhovich, D., Foteinopoulou, N., Cherenkova, K., Kacem, A., & Aouada, D. (2024). *CAD-Assistant: Tool-Augmented VLLMs as Generic CAD Task Solvers*. arXiv. Retrieved April 7, 2025, from https://arxiv.org/abs/2412.13810

Nelson, M. D., Goenner, B. L., & Gale, B. K. (2023). Utilizing ChatGPT to Assist CAD Design for Microfluidic Devices. *Lab on a Chip, 17*, 3778-3784.

Wang, R., Yuan, Y., Sun, S., & Bian, J. (2025). *Text-to-CAD Generation Through Infusing Visual Feedback in Large Language Models*. arXiv. Retrieved April 7, 2025, from https://arxiv.org/abs/2501.19054

Yavartanoo, M., Hong, S., Neshatavar, R., & Lee, K. M. (2024). *Text2CAD: Text to 3D CAD Generation via Technical Drawings*. arXiv. Retrieved April 7, 2025, from https://arxiv.org/abs/2411.06206