Information technology for automated static balancing of rotating parts in CAD environment*

Anatolii Hordieiev[1,*,†], Olga Pavlova[1,†], Victor Horokhovskyi[2,†], Nazar Kostiuk[2,†] and Houda El Bouhissi[3,†]

[1] Khmelnytskyi National University, Instytutska 11, 29000, Khmelnytskyi, Ukraine
[2] Ivano-Frankivsk National Technical University of Oil and Gas, Ivano-Frankivsk, Ukraine
[3] LIMED Laboratory, Faculty of Exact Sciences,University of Bejaia, 06000, Bejaia, Algeria

Abstract

The paper presents an information technology for automated static balancing of rotating parts in the CAD environment. The proposed system integrates mass-property extraction through the SolidWorks API, center-of-gravity offset analysis, counterweight parameter calculation, automatic geometry generation, and iterative verification within a unified CAD-based workflow. The system is implemented as a modular solution that includes CAD data acquisition, balancing computation, geometry synthesis, verification, and reporting modules. Experimental validation on a rotating part model demonstrated that the proposed approach reduces balancing time, improves accuracy, and minimizes human-induced errors compared with traditional manual balancing procedures. The developed system supports digital engineering principles by transferring static balancing from experimental post-processing to the design stage.

Keywords

automated static balancing, rotating parts, CAD automation, information technology, counterweight generation, digital engineering, center of gravity, design automation [1]

1. Introduction

The rapid development of digital engineering and intelligent design technologies has significantly transformed modern mechanical design processes. Contemporary computer-aided design (CAD) systems are no longer limited to geometric modeling; they increasingly serve as integrated platforms for simulation, analysis, optimization, and automation of engineering workflows. In this context, the integration of computational algorithms directly into CAD environments represents an important step toward intelligent and automated design systems [1].

One of the critical engineering problems in rotating machinery design is static balancing. Even minor deviations of the center of mass relative to the axis of rotation may lead to increased vibration, noise, mechanical wear, and reduced operational reliability. Traditionally, static balancing is performed either manually during the design stage or experimentally after prototype manufacturing. Both approaches are time-consuming and prone to human error.

With the advancement of CAD APIs and embedded scripting environments, it becomes possible to shift balancing procedures from experimental and manual operations to automated computational processes integrated directly within the digital model. This approach aligns with the principles of Industry 4.0, smart manufacturing, and model-based engineering, where digital twins and automated design tools reduce development time and improve product quality [4, 6].

* IntelITSIS-2026: 7th International Workshop on Intelligent Information Technologies & Systems of Information Security, July 3, 2026, Khmelnytskyi, Ukraine

1* Corresponding author.
† These authors contributed equally.
aigordeev54@ukr.net (A. Hordieiev); pavlovao@khmnu.edu.ua (O. Pavlova); sertgoroh@gmail.com (V. Horokhovskyi); nazkost92@gmail.com (N. Kostyuk); houda.elbouhissi@gmail.com (H. El Bouhissi)
0000-0003-4494-4348 (A. Hordieiev); 0000-0001-7019-0354 (O. Pavlova); 0009-0002-0676-0038 (V. Horokhovskyi); 0000-0001-6223-6412 (N. Kostyuk); 0000-0003-3239-8255 (H. El Bouhissi)

Despite the availability of advanced CAE systems for dynamic simulation, many industrial tasks require lightweight, cost-effective, and task-specific automation tools embedded directly into commonly used CAD environments. In this regard, SolidWorks provides a programmable API interface and VBA-based macro support, enabling the development of custom information systems for automated engineering analysis.

This paper presents an information system for automated static balancing of rotating parts in the CAD environment. The proposed solution integrates mass property analysis, imbalance detection, counterweight calculation, and automatic 3D model generation within a unified algorithmic framework. The system architecture, computational logic, and experimental validation are described. Comparative analysis demonstrates significant improvements in balancing accuracy, reduction of processing time, and minimization of human-induced errors.

2. Literature analysis

During the study, an analysis of the recent scientific publications has been conducted. The selected sources cover CAD-based modeling, design automation, simulation systems, digital manufacturing, rotor balancing, and digital twin technologies.

The purpose of this analysis is to identify current research trends, determine technological gaps, and position the proposed solution within the contemporary digital engineering landscape. Recent research in digital engineering demonstrates a growing integration of intelligent algorithms and automated decision-making mechanisms into engineering workflows. Modern CAD environments increasingly serve not only as geometric modeling tools but also as platforms for embedded analytics, simulation, and automation.

Several studies emphasize the transition toward intelligent manufacturing systems. The study [9] highlights the development of interactive digital manufacturing environments, where design and production processes are interconnected within unified digital platforms. Similarly, the study [12] discusses digital twin technologies that transfer monitoring and correction mechanisms from physical systems to virtual models, supporting real-time analytical control.

The integration of programmable logic within CAD systems has also been investigated. The study [6] demonstrates CAD-based programming approaches for automated mechanical assembly, showing how parametric modeling and embedded computational logic can enhance process repeatability. The study [11] further confirms the importance of structured CAD modeling strategies for enabling higher levels of automation and design consistency.

Workflow integration across CAD and CAM environments has also been investigated. The study [3] presents a streamlined design-to-manufacturing pipeline combining AutoCAD, SolidWorks, and SolidCAM. Although this work focuses primarily on production efficiency, it supports the broader concept of digital process integration within unified engineering systems.

The study [16] investigated the application of the SolidWorks software environment for determining mass, kinematic, and dynamic characteristics of objects with complex volumetric geometric configurations.

A subsequent study [17] developed a methodological framework for determining kinematic characteristics of moving objects with complex spatial configurations. The research formalized procedures for mass property evaluation and motion analysis within CAD, strengthening the theoretical basis for computational modeling of rotating systems.

Artificial intelligence techniques are increasingly embedded in manufacturing processes. The study [14] proposes a real-time method for detecting wood surface defects using the YOLOv8 deep learning model. This work integrates computer vision and CNC machining parameter adjustment to enhance surface quality and reduce processing defects. While this work focuses on intelligent visual inspection rather than CAD-based structural correction, it illustrates the broader trend of embedding algorithmic decision-making directly into engineering workflows. It confirms the feasibility and industrial relevance of integrating computational intelligence into traditionally manual processes.

Mechanical balancing methodologies remain largely rooted in conventional engineering approaches. The study [5] investigates industrial rotor balancing systems from a mechanical perspective, focusing on balancing principles and hardware-based correction methods. However, such approaches do not incorporate API-driven automation or direct geometric synthesis within CAD environments.

The literature review demonstrates that although significant progress has been achieved in CAD modeling, digital manufacturing, and mechanical system optimization, there remains a clear gap in the development of lightweight, API-integrated information systems for automated static balancing directly within CAD platforms. Existing studies either focus on mechanical balancing theory, CAD-based simulation, or workflow optimization, but do not combine algorithmic imbalance detection, automated counterweight computation, and 3D model generation into a unified information system architecture.

Therefore, the proposed information system for automated static balancing in the SolidWorks CAD environment represents a relevant and novel contribution aligned with current trends in digital engineering, intelligent design automation, and Industry 4.0 development.

3. System architecture

The proposed solution is implemented as an information system embedded into the SolidWorks CAD environment and aimed at automated static balancing of rotating parts at the digital design stage. The system follows a modular architecture that combines CAD mass-property acquisition through the SolidWorks API, algorithmic imbalance estimation, counterweight parameter synthesis, automatic geometric generation of the corrective element, and iterative verification based on updated mass properties.

3.1. Architectural overview

The system is built as a CAD-integrated automation module (VBA macro) that communicates with the CAD model via the SolidWorks API [14]. Its architecture is organized into three functional layers:

1. Data Acquisition Layer (CAD or API Interface). This layer retrieves model-dependent parameters from SolidWorks, including mass, center of gravity (CoG) coordinates, reference coordinate system, and rotation axis definition. It also manages model regeneration and updates after geometric modifications.
2. Computation Layer (Balancing Core). This layer contains the core algorithm responsible for imbalance detection and counterweight parameter computation. The primary inputs are CoG offset relative to the target rotation axis and user-defined design constraints (allowed placement zones, counterweight type, material assumptions). The outputs include the required counterweight mass and placement vector.
3. Geometry Synthesis and Validation Layer. Based on computed parameters, the system automatically generates a parametric 2D sketch and builds a 3D counterweight feature (with mounting elements if needed). After insertion, the system recomputes mass properties and verifies whether the updated CoG meets the balancing tolerance. If the tolerance is not satisfied, the procedure iterates by adjusting counterweight parameters.

3.2. Software components

The implementation is structured using an object-oriented approach to ensure reusability and clear separation of responsibilities.

Three key software components are defined: Balancer module (clsBalancer) performs imbalance estimation, computes counterweight mass and placement parameters, and checks balancing criteria; Geometry builder (clsGeometryBuilder) constructs parametric sketches and generates 3D

counterweight geometry (cylindrical, disk, sector forms), including optional mounting features; Controller (clsController) orchestrates the workflow, manages UI inputs, calls the computational core, triggers geometry creation, and initiates verification loops. This structure enables a standardized “compute – build - verify” cycle and supports extension to different rotating parts and counterweight configurations.
The automated balancing workflow can be described as the following sequence:

1. Load the assembly model and select the reference coordinate system and axis of rotation.
2. Extract mass properties (mass and CoG) using SolidWorks API calls.
3. Compute the CoG offset relative to the rotation axis and estimate the static imbalance magnitude.
4. Determine counterweight parameters (mass and placement) under design constraints.
5. Generate counterweight sketch and build 3D geometry automatically.
6. Insert and rebuild the model, then recompute mass properties.
7. Verify whether the CoG offset is within tolerance; if not, adjust parameters and repeat steps 4–7.
8. Output a statically balanced model ready for downstream CAD tasks.

In conventional mechanical design practice, the static balancing of rotating components is typically performed as a partially manual and experimentally supported procedure. The traditional workflow begins with CAD modeling of the part geometry, followed by manual estimation and placement of balancing features. After prototype manufacturing, experimental balancing is conducted using physical test setups. If imbalance remains, redesign iterations are required, which increase development time and cost. In this approach, balancing is largely separated from the digital design stage and relies on post-model validation.

In contrast, the proposed information system integrates static balancing directly into the CAD design environment. The redesigned workflow begins with standard CAD modeling but immediately proceeds to automated extraction of mass properties using the SolidWorks API. The system then performs algorithmic imbalance computation based on the detected center-of-gravity offset relative to the selected axis of rotation. Positioning diagram of the proposed architecture is presented in Figure 1.

Following this analysis, the system automatically synthesizes corrective counterweight geometry according to predefined design constraints. The generated geometry is inserted into the model, and mass properties are recalculated within the same CAD environment. If necessary, the system iterates until the balancing tolerance is satisfied. The balanced model can then be transferred to downstream CAE or CAM processes without requiring separate experimental correction stages.

The key distinction between the traditional and the proposed workflow lies in the transfer of the balancing procedure from manual and experimental stages to a fully automated, CAD-integrated computational process. This transformation reduces design iteration cycles, minimizes human error, and supports the principles of model-based engineering and digital manufacturing.

Figure 2 demonstrated the proposed in this research Modular architecture of the automated static balancing information system. The architecture is organized into five interconnected functional modules that collectively ensure a complete automated balancing workflow, from data acquisition to verification and reporting.

The CAD API Module serves as the interface between the information system and the SolidWorks environment. It is responsible for accessing the active CAD model, extracting mass properties, retrieving center-of-gravity (CoG) coordinates, identifying the selected rotation axis, and triggering model rebuild operations. This module provides standardized data structures that are used by subsequent computational components. By encapsulating all interactions with the SolidWorks API, the system ensures separation between geometric data access and algorithmic processing.

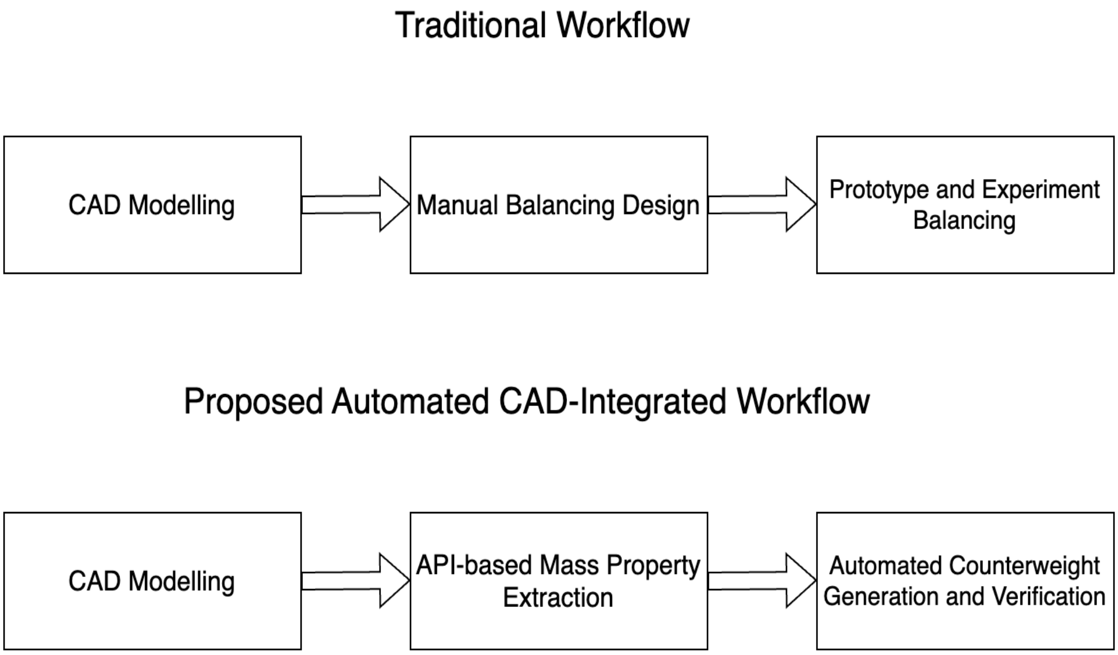

Figure 1: Positioning of the proposed automated balancing system within the CAD-based engineering workflow.

The Balancing Computation Module represents the analytical core of the system. Based on the CoG offset relative to the defined axis of rotation and user-defined design constraints, this module estimates the magnitude and direction of static imbalance. It then computes the required counterweight parameters, including corrective mass, placement radius, and angular position. The computation process may incorporate geometric and material constraints to ensure feasibility within the existing model. This module operates independently of specific geometric implementation details, thereby maintaining modularity and extensibility.

The Geometry Generation Module transforms computed counterweight parameters into parametric CAD features. It automatically creates sketches and generates corresponding three-dimensional geometry, such as cylindrical, disk-shaped, or sector-based counterweights. The generated geometry is integrated directly into the existing model configuration. This module ensures that the corrective geometry is consistent with design constraints and can include mounting features when required.

The Verification Module performs post-generation validation of the updated model. After counterweight insertion and model rebuild, mass properties are recalculated via the CAD API Module. The residual CoG offset is evaluated against predefined tolerance criteria. If the balancing condition is not satisfied, the system triggers an iterative correction cycle by returning updated parameters to the Balancing Computation Module. This iterative mechanism ensures convergence toward the desired static equilibrium state.

The Reporting Module provides structured documentation of the balancing process. It aggregates computational results, iteration metrics, tolerance values, and final model characteristics. The module generates logs or formatted reports that can be used for experimental validation, traceability, and further engineering analysis. This functionality enhances reproducibility and supports integration into digital engineering pipelines.

3.3. Algorithmic implementation

The automated static balancing procedure is implemented as an iterative algorithm embedded within the SolidWorks CAD environment. The objective of the algorithm is to minimize the offset of the center of gravity (CoG) relative to the predefined axis of rotation by automatically generating and adjusting corrective counterweight geometry. The algorithm operates on mass properties extracted from the CAD model via the SolidWorks API. The balancing condition is satisfied when the distance between the CoG projection and the rotation axis does not exceed a predefined tolerance value ε.

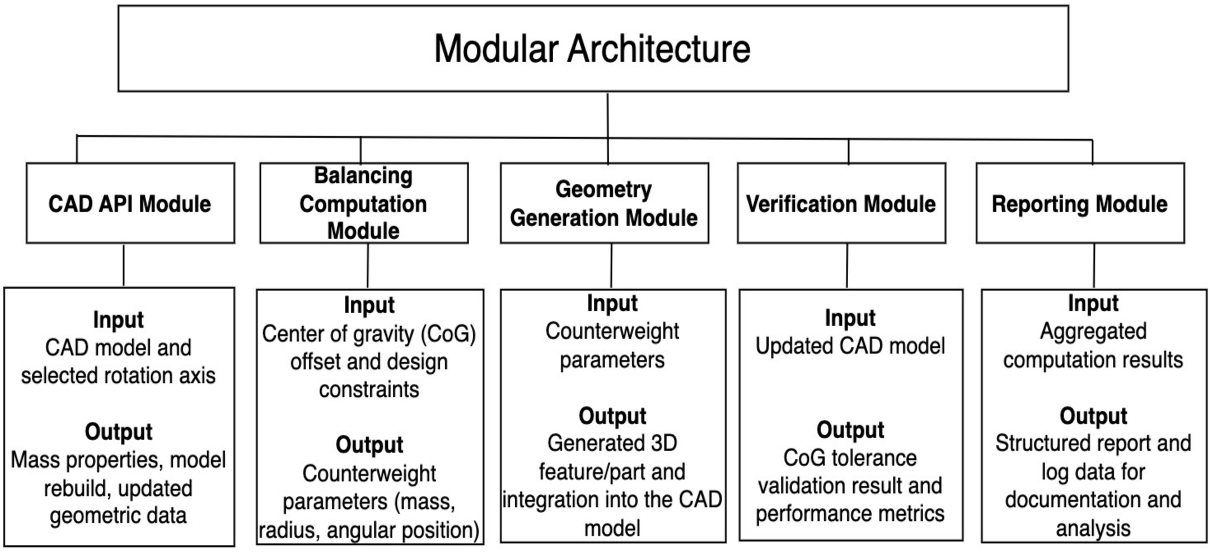

Figure 2: Modular architecture of the proposed automated static balancing information system.

Let: M - total mass of the rotating part, $\vec{r_{\mathrm{CoG}}}$ - position vector of the center of gravity, $\vec{r_{\mathrm{axis}}}$ - projection of CoG onto the rotation axis, $\Delta\vec{r}=\vec{r_{\mathrm{CoG}}}-\vec{r_{\mathrm{axis}}}$ - imbalance vector, ε - allowable tolerance.

The balancing objective is described by the formula 1:

$$|\Delta r|\leq\varepsilon \tag{1}$$

If the condition is not satisfied, the required corrective mass m_c and its placement radius R_c are determined such that the moment generated by the counterweight compensates the detected imbalance. The general balancing principle is based on static equilibrium of moments and is denoted by the equation 2:

$$M \cdot \Delta r = m_c \cdot R_c \tag{2}$$

From which the corrective mass can be estimated by the formula 3:

$$m_c = \frac{M \cdot \Delta r}{R_c} \tag{3}$$

The algorithm incorporates geometric constraints and available placement regions when selecting R_c. After computing the counterweight parameters, the system automatically generates the corresponding CAD geometry and recalculates mass properties.

The process repeats until convergence. The balancing algorithm is proposed in Figure 3.

The implemented algorithm possesses the following properties:

1. Deterministic convergence under static balancing assumptions,
2. Full integration with CAD mass property calculations,
3. Independence from external simulation environments,
4. Iterative tolerance-based stopping criterion,
5. Extendability to multi-plane or dynamic balancing cases.

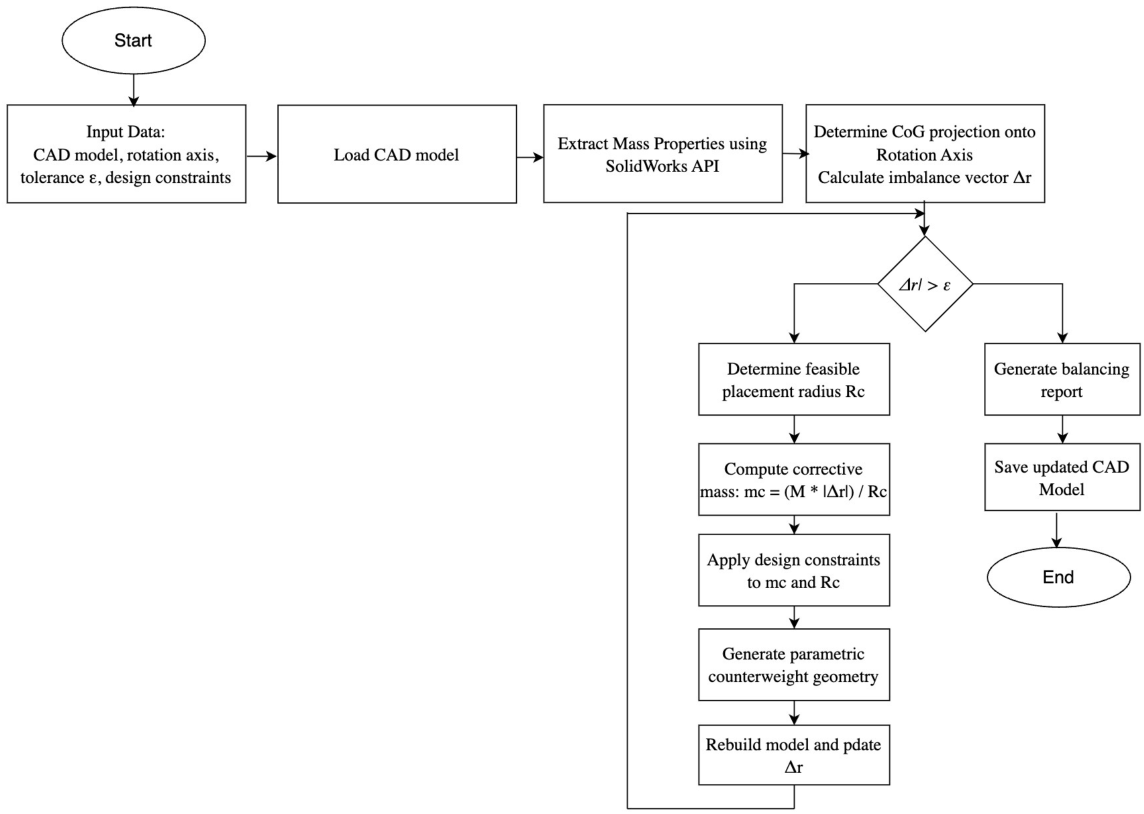

Figure 3: Fraphic implementation of the balansing algorithm.

This algorithmic structure ensures repeatability, minimizes human-induced errors, and shifts balancing from an experimental post-processing stage to a computational design-stage procedure within the digital engineering workflow.

3.4. Computational complexity analysis

From a computational perspective, the proposed automated balancing system operates as an iterative CAD-integrated optimization procedure. The overall computational complexity depends primarily on three factors: extraction of mass properties through the SolidWorks API, geometric regeneration of corrective features, iterative convergence toward the balancing tolerance.

Let: *n* - number of geometric entities in the CAD model, *k* - number of balancing iterations, *g(n)* - complexity of CAD model rebuild operations.

The mass property extraction stage performs API-based queries over the CAD model and can be approximated by the Formula 4:

$$T_{\text{mass}}(n) = O(n) \tag{4}$$

since the system traverses geometric and material entities required for mass-property evaluation. The counterweight computation stage consists of direct algebraic calculations and constraint evaluation. Its complexity is constant and can be denoted by the equation 5:

$$T_{\text{compute}} = O(1) \tag{5}$$

The most computationally expensive stage is geometric regeneration and CAD rebuild, which complexity depends on model topology and feature dependencies (Formula 6):

$$T_{\text{rebuild}}(n) = O(g(n)) \tag{6}$$

Therefore, the total complexity of the iterative balancing process can be expressed by the Formula 7:

$$T_{\text{total}}(n, k) = O(k \cdot (n + g(n))) \tag{7}$$

where k is typically small due to deterministic convergence of the balancing algorithm. Experimental evaluation demonstrated that convergence is generally achieved within 2–3 iterations for practical engineering models. From a systems engineering perspective, the proposed architecture minimizes computational overhead by: avoiding external CAE solvers, performing calculations directly within the CAD environment, using lightweight API-based mass-property extraction and limiting iterative geometry reconstruction only to corrective features.

As a result, the computational cost remains sufficiently low for interactive engineering workflows and real-time design-stage correction tasks.

4. Results and discussion

To validate the proposed automated static balancing information system, an experimental study was conducted within the SolidWorks CAD environment using a crankshaft model as a representative rotating component. The initial CAD model was intentionally unbalanced, exhibiting a measurable center-of-gravity (CoG) offset relative to the predefined rotation axis. Mass properties were extracted using the SolidWorks API, and imbalance magnitude was calculated. The automated balancing algorithm was then executed. The system computed the corrective counterweight parameters, generated parametric 3D geometry, integrated it into the model, and iteratively recalculated mass properties until the predefined tolerance threshold was satisfied.

Performance was evaluated using residual CoG displacement, execution time, and iteration count. Additionally, results were compared with a traditional manual balancing approach performed under similar conditions. The automated system successfully reduced the CoG offset to within tolerance limits and eliminated observable trajectory deviation in motion simulation at 1500 rpm. The balancing process required significantly less time compared to manual correction and demonstrated improved repeatability. The quantitative results of the conducted experiment are proposed in Table 1.

Table 1
Comparative performance evaluation

Metric	Manual Balancing	Automated System	Improvement
Average time per part (min)	15	1	~93% reduction
Average time per assembly (min)	30	4	~87% reduction
Residual CoG deviation (mm)	0.05	0.001	98% improvement
Error rate (%)	14	1	Significant reduction
Iterations required	(manual)	2-3	Deterministic convergence

The automated system demonstrated substantial efficiency gains while maintaining higher balancing precision and eliminating subjective design adjustments.

The obtained results confirm that integrating static balancing algorithms directly into the CAD environment transforms balancing from a manual corrective activity into a deterministic computational design procedure. Unlike traditional approaches, where balancing is performed after geometry completion or physical prototyping, the proposed system embeds imbalance detection and correction within the digital modeling phase. This integration aligns with model-based engineering principles and reduces dependency on experimental validation.

From a systems perspective, the modular architecture ensures separation between data acquisition, computation, geometry synthesis, and verification. This design enables scalability and potential extension to dynamic balancing or multi-plane correction scenarios. The experimental evaluation demonstrates that algorithmic automation significantly reduces design time while improving precision and repeatability. The convergence-based iterative mechanism guarantees tolerance satisfaction without manual intervention. Overall, the developed information system represents a practical implementation of intelligent CAD automation within the broader context of digital engineering and Industry 4.0 paradigms.

5. Conclusion

This paper presented an information system for automated static balancing of rotating parts in the SolidWorks CAD environment. The proposed system integrates mass-property extraction, imbalance computation, automatic counterweight generation, and iterative verification into a single CAD-based workflow.

The developed approach transfers static balancing from manual and experimental procedures to the digital design stage. Experimental validation showed that the system reduces balancing time, improves accuracy, and minimizes human-induced errors compared with traditional manual balancing.

Future research will focus on extending the system to dynamic and multi-plane balancing, integrating optimization methods, and connecting the proposed CAD-based solution with digital twin and sensor-based monitoring platforms.

Declaration on Generative AI

During the preparation of this work, the authors used Grammarly in order to: Grammar and spelling check. After using these tools, the authors reviewed and edited the content as needed and take full responsibility for the publication's content.

References

[1] P. Patpatiya, A comparative motion study of mated gears on AutoCAD and SOLIDWORKS, in: Advances in Augmented Reality and Virtual Reality, Springer, 2022, pp. 57–72.

[2] I. M. Sudana, I. M. Rajendra, M. Yusuf, I. G. N. S. Waisnawa, Design of an ergonomic rotating steering system on an electric cart based on RULA analysis (Rapid Upper Limb Assessment) and using SolidWorks software.

[3] G. Arifaj, R. Likaj, B. Berisha, Streamlining design workflow: Converting AutoCAD to SolidWorks, utilizing SolidCAM, and cost analysis integration, in: Proceedings of the International Conference on Textile Conference & Conference on Engineering and Entrepreneurship, Springer Nature Switzerland, 2023, pp. 77–85.

[4] O. Y. Rudyk, O. I. Sokolov, Application of information technologies on the example of SolidWorks, Naukovi Pratsi 336 (2022).

[5] S. Adhikari, Design of Rotor Balancing System for Industrial Purpose, Doctoral dissertation, Pulchowk Campus, 2022.

[6] P. Kyratsis, A. Tzotzis, J. P. Davim, Automated assembly of mechanical systems: A case study, in: CAD-Based Programming for Design and Manufacturing, Springer Nature Switzerland, 2025, pp. 53–68.

[7] K. B. Mustapha, Practical Finite Element Simulations with SOLIDWORKS 2022: An Illustrated Guide to Performing Static Analysis with SOLIDWORKS Simulation, Packt Publishing, 2022.

[8] A. Nashat, A. Morsi, M. M. Hassan, M. Abdelrahman, Ballbot simulation system: Modeling, verification, and gym environment development, in: Proceedings of the 11th International Conference on Intelligent Computing and Information Systems (ICICIS), IEEE, 2023, pp. 198–204.

[9] G. C. Vosniakos, P. Katsaros, I. Papagiannoulis, E. Meristoudi, Development of robotic welding stations for pressure vessels: Interactive digital manufacturing approaches, International Journal on Interactive Design and Manufacturing 16 (1) (2022) 151–166.

[10] S. Bozherikov, H. Uzunov, S. Dechkova, M. Tsoneva, Analysis and optimization of epicyclic mechanisms with mutually meshed satellites for engineering and industrial applications, International Journal of Applied Mechanics and Engineering 30 (2) (2025) 62–88.

[11] I. Huić, P. Kosec, T. Martinec, S. Škec, Comparison of CAD modeling approaches in the design of custom dental abutments, Computer-Aided Design and Applications 23 (4) (2025) 428–450.

[12] B. Tripathi, N. Srivastava, A. K. Tiwari, Execution of fully automated coal mining face with transparent digital twin self-adaptive mining system, in: Simulation Techniques of Digital Twin in Real-Time Applications: Design Modeling and Implementation, 2024, pp. 183–194.

[13] A. Roshan, R. Dhaouadi, CAD modeling and simulation of a large quadcopter with a flexible frame, in: Proceedings of the Future Technologies Conference, Springer International Publishing, 2022, pp. 860–879.

[14] V. Tkachuk, S. Kostiuk, O. Pavlova, V. Alekseiko, M. Kostiuk, Real-time method of detecting wood defects using YOLOv8 model, CEUR Workshop Proceedings 3974 (2025) 107–115.

[15] SOLIDWORKS API Documentation. Available at: https://help.solidworks.com (accessed February 17, 2026).

[16] V. O. Gorohovsky, A. I. Gordeev, Application of the SolidWorks software product to determine the mass, kinematic and dynamic characteristics of the motion of objects of complex volumetric geometric configuration, Bulletin of Khmelnytsky National University. Technical Sciences (6) (2023) 415–420.

[17] V. O. Gorohovsky, A. I. Gordeev, Creation of a methodology for determining the kinematic characteristics of moving objects of complex volumetric configuration, Bulletin of Khmelnytsky National University. Technical Sciences (4) (2024) 482–490.
[18] SOLIDWORKS API Documentation. Available at: https://help.solidworks.com (accessed May 16, 2026).
[19] SOLIDWORKS Developer Portal. Available at: https://www.solidworks.com/developer-portal (accessed May 16, 2026).
[20] J. D. Camba, M. Contero, P. Company, Parametric CAD modeling: An analysis of strategies for design reusability, Computer-Aided Design 134 (2021) 102984.
[21] E. Santos, C. Marto, A. Gonçalves-Coelho, Integration of knowledge-based engineering systems within CAD platforms, Advanced Engineering Informatics 51 (2022) 101474.
[22] R. Andriiv, V. Senkivskyy (2025). Information technology for predicting the reliability level of text messages. *Computer Systems and Information Technologies*, 2025, (2), 65–71. https://doi.org/10.31891/csit-2025-2-7.