Development and Pilot Validation of Computational Tools for Gear Design and Manufacturing Workflows

Nico O. Aspra[1],* Michael A. Alagaban[1], Mary Jane C. Urbano[1].
*Corresponding Author: nico.aspra@bicol-u.edu.ph

[1] Department of Mechanical Technology
College of Industrial Technology, Bicol University
Legazpi City, Philippines

Abstract—Manual computation of gear cutting and shaft design parameters is a routine requirement in mechanical technology instruction and shop-floor practice, yet its repetitive nature introduces costly error risks in both contexts. This paper presents the development and pilot validation of three web-based computational modules for spur gear geometry, dividing head indexing, and torsional shaft sizing, implemented in Python using the Streamlit framework and deployed on Hugging Face Spaces. Unlike general-purpose calculators or static reference tables, the tool suite integrates symbolic equation solving, a multi-parameter input structure, a built-in materials library, and dual-representation indexing output within a single workflow-oriented interface grounded in established mechanical design standards. A pilot study with $N = 54$ Mechanical Technology students evaluated tool outputs against independently derived reference solutions using a binary correctness scheme across nine required output items. Results indicate an overall mean accuracy rate of 84.6%, with per-module rates of 89.6%, 75.3%, and 87.0% for the spur gear, indexing, and shaft modules respectively. Most participants completed all tasks without assistance, with a median total completion time of 190 seconds. System Usability Scale scores yielded a mean of 62.78, while Likert-scale ratings for perceived usefulness, ease of use, applicability, and reliability ranged from 4.09 to 4.51. A weak accuracy-usability correlation ($r = 0.136$) indicated that errors were attributable primarily to output labeling ambiguity rather than interface navigation difficulty. Findings support the computational validity of the tools and their readiness for laboratory instruction integration, pending targeted interface refinements.

Keywords—computational tools, gear design, manufacturing

I. Introduction

Accurate computation is fundamental to machining and mechanical design, particularly in tasks involving involute spur gear geometry, dividing head indexing for gear cutting operations, and torsional shaft sizing. These calculations rely on precise application of standard design relations, consistent unit handling, and careful interpretation of results, as outputs directly inform machining setup decisions and component design [1]. In instructional settings, students in mechanical technology courses routinely perform these computations as preparatory exercises before laboratory work. In professional practice, machinists and designers use equivalent procedures for setup planning, dimensional verification, and ensuring components remain within allowable stress and deformation limits [2], [3].

While manual calculation remains essential for conceptual learning, its repetitive application in laboratory and production contexts introduces practical limitations. Common error sources include incorrect formula selection, arithmetic mistakes, and inconsistent unit conversion. These errors are particularly costly in subtractive machining, where a mistake can result in material scrap, machine downtime, and rework [4]–[6]. For instructors, verifying student outputs across varied parameter sets is time-intensive, and the absence of a consistent computational reference increases the likelihood of undetected errors propagating into machining decisions [7].

Existing computational aids such as general-purpose calculators, spreadsheet templates, and reference tables partially address these limitations but are not designed around the specific workflows of gear cutting and shaft design instruction. They typically lack guided input sequences, embedded parameter validation, integrated output formatting aligned with standard machining reporting, and instructional context that connects computed values to their physical meaning [7], [8]. No prior work has specifically developed and pilot-validated a modular, web-accessible computational tool suite targeting both gear cutting and torsional shaft design simultaneously within a single instructional workflow.

This paper presents the development and pilot validation of three web-based computational modules for spur gear geometry, dividing head indexing, and torsional shaft sizing, grounded in established mechanical design standards [1] and deployed through an accessible browser interface. The contribution lies in the integration of validated computation logic, workflow-oriented interface design, and structured task-based pilot validation against known reference solutions, evaluated with $N = 54$ Mechanical Technology students. Results demonstrate a mean overall accuracy of 84.6% across all task items, with findings identifying specific interface refinements needed to reduce input and recording errors.

II. Related Work

The integration of digital tools into engineering and technology education has been widely examined as a means of improving instructional effectiveness and bridging the gap between theoretical knowledge and practical application. Simulation-based environments have demonstrated value in supporting design validation without physical prototyping [9], while interactive applications and virtual laboratory systems have been developed specifically for undergraduate engineering instruction through guided computation and visualized feedback [10], [11]. Research also consistently identifies a gap between graduate competencies and industry expectations, motivating the development of instructional tools closely aligned with authentic engineering tasks and decision-making workflows [7], [12], [13]. At the same time, emerging technologies such as generative design and AI-assisted optimization continue to reshape engineering practice, though challenges in access, implementation, and instructional preparation remain [8].

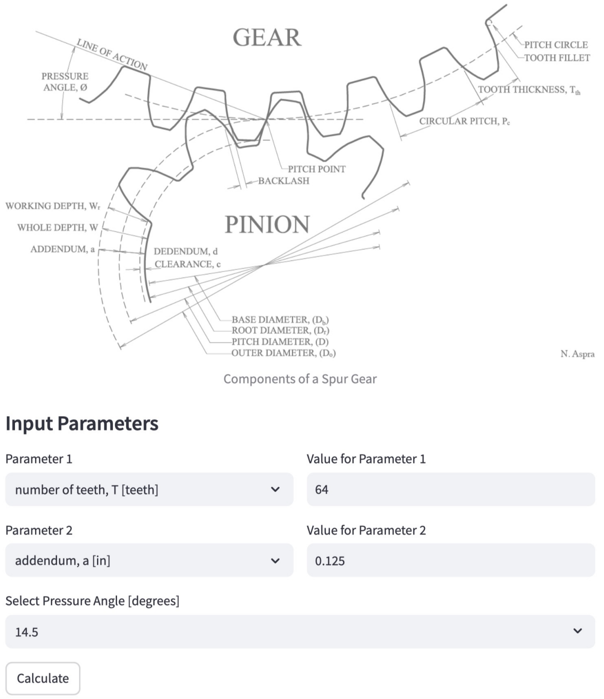
In the domain of gear cutting, prior work has addressed computational and educational needs from both industrial and academic perspectives. Industrial gear manufacturing systems have emphasized the importance of precision and repeatability in high-volume production [4], while theoretical treatments of gear cutting geometry have established the kinematic and parametric foundations underlying spur gear generation [14], [15]. At the educational level, virtual gear design environments and multi-user laboratory simulators have demonstrated the feasibility of digitally supported gear computation for undergraduate learners [10], [11]. For shaft design and torsional analysis, classical analytical methods and simulation-based approaches have been applied to deformation modeling and material selection [16], [17], while physical torsion testing devices and design project-based instruction have been used to reinforce conceptual understanding of torsional behavior [18], [19].

Despite these contributions, no prior work has developed and pilot-validated a unified, modular, web-accessible computational tool suite targeting both gear cutting and torsional shaft design simultaneously within a single instructional workflow. Existing tools are either domain-specific without cross-module integration, industrially oriented without instructional alignment, or not validated against reference solutions through structured task-based testing with defined participant groups. This work addresses that gap by presenting a three-module computational tool suite grounded in established design standards, deployed through an accessible web interface, and validated through a structured pilot study with $N = 54$ Mechanical Technology students using binary correctness scoring against known reference solutions.

III. Tool Development

The three computational modules were developed in Python using the Streamlit framework, grounded in a structured consolidation of established mechanical design relations and machining references [1]–[3], [14], [15], [20]. Each module was structured around a defined set of input, intermediate, and output variables with standardized naming conventions, consistent unit handling, and fixed-decimal output formatting. The modules were deployed as independent web applications on Hugging Face Spaces, accessible through any standard browser without local installation, with access during pilot testing facilitated through dedicated QR codes printed on the task sheet.

A. Spur Gear Geometry Module



Components of a Spur Gear

Input Parameters

Parameter 1: number of teeth, T [teeth]
Value for Parameter 1: 64
Parameter 2: addendum, a [in]
Value for Parameter 2: 0.125
Select Pressure Angle [degrees]: 14.5

Fig. 1. Spur gear geometry module showing the embedded gear diagram, two-parameter input selectors, pressure angle selection, and computed output table.

The spur gear geometry module accepts any two of thirteen standard spur gear parameters as inputs and solves for all remaining values, including diametral pitch P_d, number of teeth T, pitch diameter D, circular pitch P_c, addendum a, dedendum d, clearance c, module M, tooth thickness T_{th}, whole depth W, working depth W_r, outside diameter D_o, and root diameter D_r. The module solves the following system of governing relations simultaneously using symbolic computation [1]:

$$P_d = \frac{T}{D} = \frac{\pi}{P_c}, \quad a = \frac{1}{P_d}, \quad M = 25.4\,a, \quad T_{th} = \frac{P_c}{2} \tag{1}$$

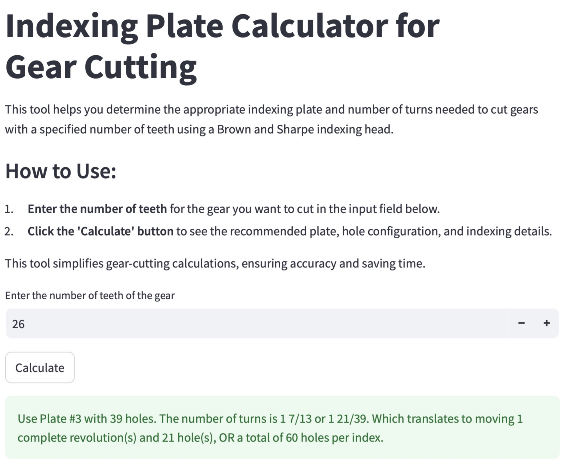
The dedendum coefficient is selected based on the specified pressure angle: $d = 1.25/P_d$ for 20° and 25°, and $d = 1.157/P_d$ for 14.5° and 22.5° [1]. The remaining outputs follow as:

$$c = d - a, \quad W = a + d, \quad W_r = 2a \tag{2}$$

$$D_o = D + 2a, \quad D_r = D - 2d \tag{3}$$

The interface presents two labeled dropdown selectors for parameter selection and a pressure angle selector. Upon submission, the symbolic solver resolves all unknowns and displays the full output set in a structured table. An embedded spur gear diagram supports correct parameter identification. Input validation rejects non-physical entries, zero inputs, and duplicate parameter selections prior to computation.

B. Dividing Head Indexing Module



Indexing Plate Calculator for Gear Cutting

This tool helps you determine the appropriate indexing plate and number of turns needed to cut gears with a specified number of teeth using a Brown and Sharpe indexing head.

How to Use:

1. **Enter the number of teeth** for the gear you want to cut in the input field below.
2. **Click the 'Calculate' button** to see the recommended plate, hole configuration, and indexing details.

This tool simplifies gear-cutting calculations, ensuring accuracy and saving time.

Enter the number of teeth of the gear

26

Use Plate #3 with 39 holes. The number of turns is 1 7/13 or 1 21/39. Which translates to moving 1 complete revolution(s) and 21 hole(s), OR a total of 60 holes per index.

Fig. 2. Indexing plate calculator showing the tooth count input and the dual-representation output: mixed-number crank movement and equivalent total hole count per index.

The dividing head indexing module supports gear cutting setup for a standard 40:1 indexing head. Given a desired number of gear teeth N, the required crank movement per index is computed as:

$$n = \frac{40}{N} \tag{4}$$

The result is decomposed into complete revolutions and a fractional remainder r/q in lowest terms. The module searches the standard Brown and Sharpe plate library for a hole circle count H satisfying $H \equiv 0 \pmod{q}$, then computes the integer hole advancement h:

$$h = \frac{r}{q} \times H \tag{5}$$

The output presents both a mixed-number crank instruction and an equivalent total hole count per index. For the validation task ($N = 26$), the module yields Plate 3 with a 39-hole circle, one complete revolution plus 21 holes advanced, or equivalently 60 total holes per index. This dual representation is an intentional design decision to reduce setup ambiguity, as misinterpretation of the fractional crank movement was the most frequently observed source of error during pilot testing, discussed further in Section V.

C. Torsional Shaft Sizing Module

The torsional shaft sizing module determines the minimum required diameter of a solid circular shaft under pure torsion [20]. For a solid shaft with polar moment of inertia $J = \pi d^4/32$, the required diameter based on allowable shear stress τ and angle of twist ϕ is:

$$d = \frac{2\tau L}{G\phi} \tag{6}$$

where L is shaft length, G is the material shear modulus, and ϕ is expressed in radians. The interface guides users through sequential inputs: material selection from a built-in library that automatically supplies G for common engineering materials including steel, aluminum, copper, titanium, brass, cast iron, stainless steel, nickel, magnesium, zinc, bronze, and carbon fiber; applied shear stress in MPa; allowable angle of twist in degrees; and shaft length in mm. Unit conversion is handled internally. For the validation task, a titanium shaft ($G = 45$ GPa, $L = 500$ mm, $\tau = 45$ MPa, $\phi = 1°$) yielded a required diameter of 57.30 mm.

IV. Validation Methodology

A. Research Design

The validation followed a single-arm pilot design in which all participants used the computational tools to complete structured tasks, with no control group or baseline manual calculation condition. This design is appropriate for a first-iteration tool validation study where the primary objective is to establish computational correctness and initial usability evidence prior to broader deployment, rather than to compare performance against an alternative condition [7]. The absence of a control group is acknowledged as a limitation and is addressed in the recommendations for future work.

B. Participants

Pilot testing was conducted with $N = 54$ Mechanical Technology students from Bicol University. Participants completed the tasks using their own devices, accessed through QR codes printed on a structured task sheet. Of the 54 participants, 52 used mobile phones and 2 used tablets, reflecting the browser-based accessibility of the deployed tools across device types.

C. Task Design

Three computation tasks were designed to represent typical gear cutting and shaft design scenarios encountered in instructional and practice-oriented settings. Task 1 required participants to compute five spur gear geometry parameters — diametral pitch, pitch diameter, circular pitch, dedendum, and working depth — given the number of teeth, addendum, and pressure angle. Task 2 required participants to determine the indexing plate number, crank turns per index expressed as a mixed number, and number of holes to advance per index for a specified gear tooth count using a standard 40:1 dividing head. Task 3 required participants to determine the required shaft diameter for a titanium shaft given its length, applied shear stress, and allowable angle of twist. Participants were instructed to use only the computational tools accessible via QR code, enter the provided parameters, interpret the outputs, and record their answers on a structured task sheet. Manual calculation was explicitly not required.

D. Correctness Scoring

Computational correctness was assessed by comparing each recorded participant output against a pre-computed reference solution derived independently from the consolidated design relations described in Section ??, verified against Machinery's Handbook [1] and standard design references [3], [20] prior to testing. Each required output item was scored using a binary scheme: a score of 1 was assigned if the participant's recorded value matched the reference value within the specified reporting precision of two decimal places, and a score of 0 was assigned otherwise. Item-level correctness scores were summed and divided by the total number of required outputs to produce an overall accuracy rate per participant, where a value of 1.0 represents full agreement with all reference solutions across all three tasks. This scheme produced nine binary correctness indicators per participant, corresponding to five outputs from Task 1, three from Task 2, and one from Task 3.

E. Task Performance Indicators

In addition to correctness, four task performance indicators were recorded for each task. Completion time was recorded in seconds from task initiation to final answer entry. Task completion status was recorded as yes or no. Assistance level was recorded using a three-level ordinal scale: None, indicating the participant completed the task independently; Minor, indicating the participant required brief clarification such as parameter identification or output interpretation; and Major, indicating the participant required substantial guidance to complete the task. Retake count was recorded as the number of times a participant re-entered inputs before finalizing an answer. These indicators provided supplementary evidence of task efficiency and interface usability beyond correctness alone.

F. Post-Task Evaluation

After completing all three tasks, participants answered a structured post-task questionnaire comprising two components. The first component was the System Usability Scale (SUS), a standardized ten-item instrument scored on a five-point Likert scale [21]. SUS responses were scored using the standard procedure: odd-numbered items were scored as the response value minus one, even-numbered items were scored as five minus the response value, the adjusted scores were summed, and the total was multiplied by 2.5 to yield a score on a 0 to 100 scale, where higher scores indicate greater perceived usability. The second component consisted of ten additional Likert-scale items grouped into four constructs aligned with the Technology Acceptance Model [22]: perceived usefulness (3 items), perceived ease of use (3 items), applicability (2 items), and reliability (2 items). Composite means were computed for each construct by averaging the corresponding item scores. All responses used a five-point scale ranging from 1 (Strongly Disagree) to 5 (Strongly Agree). Descriptive statistics were used to summarize all quantitative indicators, and no inferential statistical tests were applied given the pilot sample size and single-arm design.

V. Results

A. Computational Correctness

Per-item correctness rates across all nine required outputs are shown in Fig. 3. Task 1 outputs achieved consistently high correctness rates ranging from 81.5% to 92.6%, indicating that the spur gear geometry module produced reference-consistent outputs across the majority of participant entries. Task 3 correctness was similarly high at 87.0%. Task 2 showed a more differentiated pattern: the plate number item reached 87.0% and the crank turns item reached 98.1%, while the holes-per-index item recorded the lowest correctness rate across all outputs at 40.7%. Post-task review of participant responses indicates that this error predominantly reflected misinterpretation of the output label, with many participants recording the crank turns value in the holes-per-index field rather than the computed hole count. This finding suggests a labeling and output clarity issue rather than a computation error in the tool itself. The overall mean accuracy rate across all participants and all nine items was 84.6% (median = 88.9%), with 42 of 54 participants achieving an accuracy rate of 80% or above.

B. Task Performance

The majority of participants completed all three tasks without requiring assistance. Aggregated across all tasks, 58.0% of task attempts were completed with no assistance, 32.1% required minor clarification such as parameter identification or output interpretation, and 9.9% required major guidance. The assistance pattern was consistent across tasks, as shown in Fig. 4, with Task 3 recording a slightly higher proportion of major assistance cases, likely

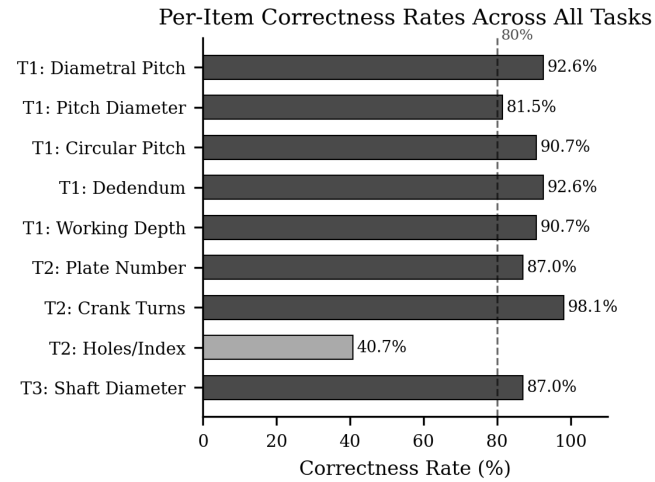


Fig. 3. Per-item correctness rates across all nine required task outputs. The dashed line indicates the 80% threshold. Items below threshold are shown in gray.

reflecting unfamiliarity with material property inputs and unit expectations in the shaft module. Retake counts were low across all tasks (mean ≤ 0.15 per task), suggesting that most participants finalized their answers without repeated input re-entry. Median completion times were 60 seconds for Task 1, 83 seconds for Task 2, and 47 seconds for Task 3, with Task 2 showing the widest variation, consistent with the greater cognitive load of interpreting the dual-representation indexing output.

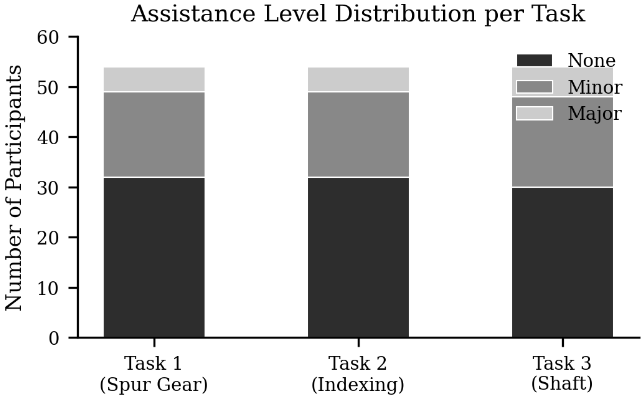


Fig. 4. Assistance level distribution per task. The majority of task attempts were completed without assistance across all three modules.

C. Usability and Acceptance

SUS scores ranged from 42.5 to 100.0, with a mean of 62.78 and a median of 58.75, placing the overall group-level usability rating in the "OK" band of the standard SUS grading scale [21]. The distribution shown in Fig. 5 indicates considerable spread, with 18 participants scoring below 50, 20 between 50 and 70, 11 between 70 and 85, and 5 at 85 or above. This spread reflects variation in participant familiarity with web-based tools and mobile browser interfaces, and is consistent with the expected range for a first-iteration deployment without prior user onboarding. In contrast, the four Likert-scale constructs aligned with the Technology Acceptance Model [22] yielded more uniformly favorable ratings: perceived usefulness (mean = 4.51), perceived ease of use (mean = 4.31), applicability (mean = 4.09), and reliability (mean = 4.17), all on a five-point scale. These results indicate that despite moderate SUS scores, participants broadly perceived the tools as useful, applicable to their instructional context, and reliable in their computational outputs. The Pearson correlation between SUS scores and overall accuracy rates was weak ($r = 0.136$), suggesting that accuracy was not strongly dependent on perceived usability, and that errors were more likely attributable to output interpretation and recording issues than to interface navigation difficulty.

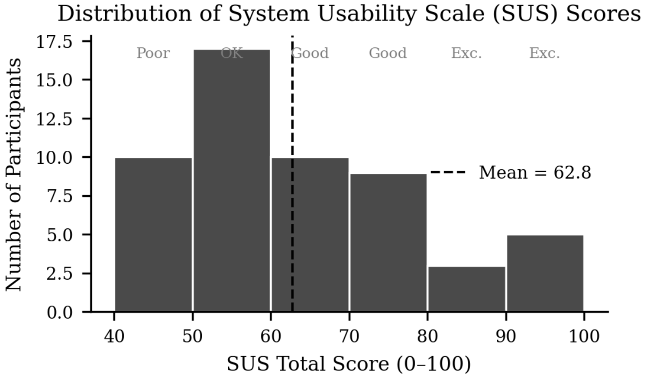


Fig. 5. Distribution of SUS total scores ($N = 54$). The dashed line indicates the group mean of 62.78. Informal SUS grade bands are annotated above each bin range.

The relationship between overall accuracy and SUS score is shown in Fig. 6. The scatter plot and fitted trend line indicate a weak positive correlation ($r = 0.136$) between perceived usability and task accuracy, suggesting that higher usability ratings were not a strong predictor of correctness. This finding implies that the accuracy errors observed in this pilot were more likely attributable to output labeling ambiguity and recording confusion than to difficulty navigating the interface. Notably, several participants who rated the tool highly in usability still recorded low accuracy, further supporting the interpretation that interface improvements targeting output clarity and label specificity are the more productive focus for the next development iteration.

VI. Conclusion

This paper presented the development and pilot validation of three web-based computational modules for spur gear geometry, dividing head indexing, and torsional

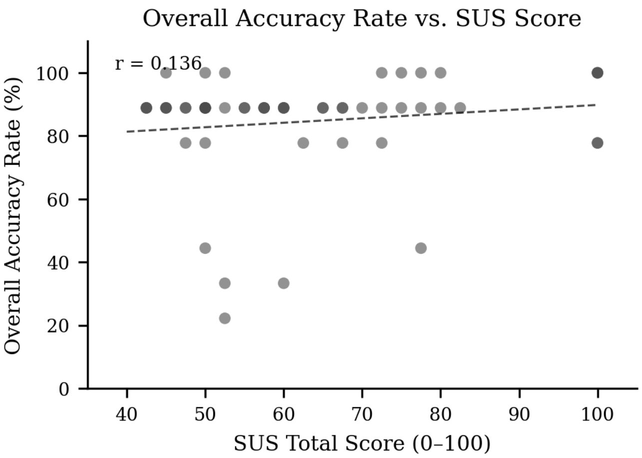


Fig. 6. Scatter plot of overall accuracy rate versus SUS total score with fitted trend line. The weak correlation ($r = 0.136$) indicates that accuracy errors were largely independent of perceived usability.

shaft sizing, implemented in Python using the Streamlit framework and deployed on Hugging Face Spaces. The tool suite was grounded in established mechanical design standards and validated against independently derived reference solutions. A pilot study with $N = 54$ Mechanical Technology students demonstrated an overall mean accuracy rate of 84.6%, with per-module rates of 89.6%, 75.3%, and 87.0% for the spur gear, indexing, and shaft modules respectively. The majority of participants completed all tasks with no assistance and low retake counts, and Likert-scale ratings for perceived usefulness, ease of use, applicability, and reliability ranged from 4.09 to 4.51, indicating strong user acceptance.

The most critical finding was the 40.7% correctness rate for the holes-per-index output, traced to output labeling ambiguity rather than a computation error. Combined with a mean SUS score of 62.78 and a weak accuracy-usability correlation ($r = 0.136$), this points to output clarity and label specificity as the primary targets for the next development iteration. Future work should extend validation to industry practitioners and incorporate a baseline manual calculation condition to enable direct performance comparison.

References

[1] E. Oberg, F. D. Jones, H. L. Horton, and H. H. Ryffel, Machinery's Handbook, 30th. Connecticut: Industrial Press, 2016.

[2] J. R. Walker and B. Dixon, Machining Fundamentals, 10th. Tinley Park, IL: The Goodheart-Willcox Company, Inc., 2018.

[3] K. H. Moltrecht, Machine Shop Practice. New York, NY: Industrial Press, 1981.

[4] Y. Nishimura, Y. Yanase, M. Hashitani, M. Fujita, and M. Misaki, "Gear cutting and grinding machines and precision cutting tools developed for gear manufacturing for automobile transmissions,mitsubishi heavy industries technical review vol.43 no.3(2006)," 2008, [Online]. Available: https://api.semanticscholar.org/CorpusID:51802245.

[5] S. Kalpakjian and S. Schmid, Manufacturing Engineering and Technology, 9th. Pearson, 2025, isbn: 9780138309466.

[6] O. Ezeanyim, U. O. Onwurah, O. Ndubuisi, and C. Okpala, "An evaluation of actual costs of rework and scrap in manufacturing industry," vol. 2, pp. 612–618, Apr. 2015.

[7] D. Trevas and J. Tester, "Improving machine design instruction by developing computational design tools," in 2019 Pacific Southwest Section Meeting Proceedings, ASEE Conferences, 2019, doi: 10.18260/1-2--31831.

[8] I. Magomedov, E. Belashova, and M.-D. Bersanov, "Review article: Enhancing the power of artificial intelligence in mechanical design," E3S Web of Conferences, vol. 402, I. Tanaino and T. Dzholdosheva, Eds., p. 03042, 2023, issn: 2267-1242, doi: 10.1051/e3sconf/202340203042.

[9] S. Rasu, "Impact of simulation tools on mechanical design validation," Journal of Engineering and Applied Sciences Technology, pp. 1–4, Sep. 2022, issn: 2634-8853, doi: 10.47363/jeast/2022(4)e135.

[10] E.-S. S. Aziz, Y. Chang, S. K. Esche, and C. Chassapis, "A multi-user virtual laboratory environment for gear train design," Computer Applications in Engineering Education, vol. 22, no. 4, pp. 788–802, Feb. 2013, issn: 1099-0542, doi: 10.1002/cae.21573.

[11] S. Tumkor, M. Fetvaci, and I. Fidan, "Development and implementation of a virtual gear design and simulation tool for undergraduate education," in 2003 Annual Conference Proceedings, ASEE Conferences, 2003, pp. 8.409.1–8.409.11, doi: 10.18260/1-2--11435.

[12] S. Zeidan, "An effective framework for bridging the gap between industry and academia," vol. 11, pp. 454–461, Apr. 2020.

[13] P. Sharma, "Bridging the skill gap between the academia and industry: A study on lovely professional university," Interantional Journal of Scientific Research in Engineering and Management, vol. 08, no. 05, pp. 1–5, May 2024, issn: 2582-3930, doi: 10.55041/ijsrem33553.

[14] S. P. Radzevich, Gear Cutting Tools: Fundamentals of Design and Computation. CRC Press, Feb. 2010, isbn: 9780429152467, doi: 10.1201/9781439819685.

[15] S. Radzevich, Gear Cutting Tools: Science and Engineering. Routledge, Oct. 2017, isbn: 9781351712224, doi: 10.1201/9781315178042.

[16] H. Truong, E. Odom, C. Egelhoff, and K. Burns, "Using modern engineering tools to efficiently solve challenging engineering design problems: Analysis of the stepped shaft," in 2011 Spring ASEE Middle Atlantic Section Conference Proceedings, ASEE Conferences, 2011, doi: 10.18260/1-2-1153-50053.

[17] J. F. Kayode, J. O. Akinyoola, T. Ogedengbe, A. A. Noiki, S. A. Afolalu, and A. O. Akinola, "Analysis of torsional deformation in an elongated shaft," in 2024 International Conference on Science, Engineering and Business for Driving Sustainable Development Goals (SEB4SDG), IEEE, Apr. 2024, pp. 1–8, doi: 10.1109/seb4sdg60871.2024.10630420.

[18] T. Teranishi, A. Saimoto, and K. Ushijima, "Development of teaching material on torsional rigidity," Journal of JSEE, vol. 71, no. 5, pp. 37–41, 2023, issn: 1881-0764, doi: 10.4307/jsee.71.5_37.

[19] H. Stalford, "Hands-on method for teaching design of mechanical components course," in 2014 ASEE Annual Conference & Exposition Proceedings, ASEE Conferences, 2014, pp. 24.661.1–24.661.21, doi: 10.18260/1-2--20552.

[20] A. D. Deutschman, Machine Design: Theory and Practice. New York: McMillan Publishing Co. Inc., 1975.

[21] J. Brooke, "Sus: A quick and dirty usability scale," in Usability Evaluation in Industry, P. W. Jordan, B. Thomas, B. A. Weerdmeester, and A. L. McClelland, Eds. Taylor & Francis, 1996, pp. 189–194.

[22] F. D. Davis, "Perceived usefulness, perceived ease of use, and user acceptance of information technology," MIS Quarterly, vol. 13, no. 3, pp. 319–340, 1989, doi: 10.2307/249008.