The Future of Engineering Education
2024 Annual Conference & Exposition
Oregon Convention Center
Portland, OR . June 23 - 26, 2024
ASEE
Paper ID #41865

Directions in Automating CAD Modeling Assessment

Dr. Derek M. Yip-Hoi, Western Washington University

Dr. Yip-Hoi is currently a professor in Manufacturing Engineering at the Department of Engineering and Design at Western Washington University. Previously, he served on the faculties of the University of the West Indies - St. Augustine, the University of Michigan - Ann Arbor, and the University of British Columbia. His research interests lie in the areas of CAD, geometric and solid modeling, machining and CNC, engineering design and ethics, and machine design.

Jack P. Wilson, North Carolina State University

Jack Wilson is a Ph.D. student in the Edward P. Fitts Department of Industrial and Systems Engineering at NC State, focusing on Advanced Manufacturing and Systems Analytics and Optimization.

Challenges in Automated CAD Modeling Assessment

Abstract

Automating the assessment of CAD models has been the focus of significant research efforts. One focus of this has been in its application to grading in support of training of engineering students in 3D parametric modeling skills and practices. However, there continue to be significant challenges in producing broadly acceptable tools of practice due to the complexities involved in creating a CAD model and in identifying formal evaluation criteria that robustly capture whether skills have been acquired. Of interest is whether tools can be developed that provide more robust formative assessment of a modeling activity. This contrasts with summative assessment approaches which largely benefits the assessor in reducing grading times by evaluating the result but can miss important tendencies in a student designer that might need to be corrected. For this to be feasible better metrics that reflect how a modeling activity is progressing not just with respect to realizing a final shape goal, but also in capturing design intent and meeting best practices is needed. In this paper some of the challenges of evaluating 3D CAD modeling efficacy are explored. These challenges increase with the level of complexity desired in the result which can range from just creating a final 3D shape to capturing design intent and finally skill at incorporating best practices. A case study of a capstone modeling project given to students in an introductory CAD class is used to illustrate these challenges. This example also highlights the difficulties encountered with assessing more open-ended modeling experiences when students are given less guidance and have many more options that they can use in satisfying the modeling requirements. A simple case study is also presented to demonstrate the viability of collecting a more complete set of assessment metrics during a modeling activity. A discussion of how access to a richer set of metrics might lead to a better understanding of modeling tendencies is presented.

Introduction

The challenge of assessing the skills of a student in a 3D CAD modeling activity is one that is well known to instructors in the field. Classes with large numbers of students completing numerous assignments over the course of a semester presents a volume of work that is time consuming to grade consistently. More importantly there is a question of whether the model being graded, the final result of completing an assignment, captures enough information to truly reflect the skills of the student being assessed. Then, if it does, how is that information extracted, interpreted, and correlated with a specific deficiency in the skills of a student that an instructor can proactively address. It can be argued that for this to be most effective it needs to include more information during the activity when the student is building the model so that their modeling decisions are more holistically understood. The busy nature of engineering classes means that students often do not have the time to rework skill acquisition type problems like those in CAD modeling based on feedback for improvement. Repetition helps, but this can be frustrating for those students who have adequately grasped the skill.

Unless an instructor is carefully monitoring each student as they complete a modeling exercise, a practical impossibility in a large CAD class, they will not get a clear picture of how deftly a student is practicing a skill. Two students can arrive at the same result with markedly different efforts. The ease with which information can be deleted or changed without any record of this being retained by the model, can mask the true process followed by a student to arrive at a result. An additional complication is that as more modeling know-how is acquired by a student, the number of paths they can follow to get to a result increases. Instructors can constrain the choices possible in an assignment by requiring that certain practices be observed. For example, it might be required that the hole feature be used for diameters up to a certain value, or that fillets be created as a feature rather than using an arc edge on a sketch that will be extruded or revolved.

All of this complicates assessment of skills let alone developing approaches for automation. To further study and illustrate the challenges faced, metrics extracted from a complex part modeled as part of a capstone assignment in an introductory CAD class will be presented and discussed. The capstone assignment was chosen as it most completely captures the culmination of skills acquisition from the class. Students are given limited instructions and are expected to demonstrate an appropriate use of modeling practices learned from the more structured training and weekly assignments that proceeded it. This will be followed by a simpler example used to demonstrate the potential for expanding the range of metrics that can be used to assess how effectively a modeling assignment is completed.

Some Relevant Reported Research

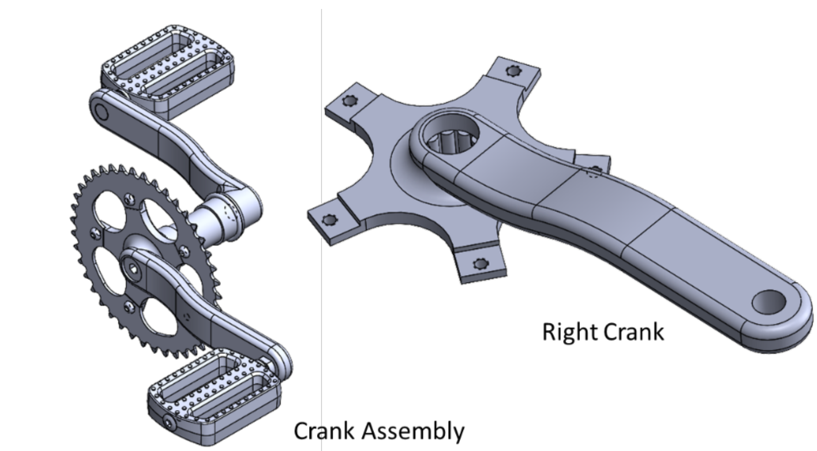
A significant body of past work exists around automating CAD model assessment. Most of this work has focused on the summative approach that analyzes a result with the goal of scoring a student's work and providing ex post facto feedback on the differences with the expected result. One focus of this research is on evaluating 2D drawings generated from 3D CAD models. Hekman et al. [1] describe their experiences with a system that extracts geometric information from an Autodesk DXF file submitted by students and scores its accuracy by comparison with the expected result. Their method was developed using LabVIEW with a second version implemented to support a student receiving automated feedback that they can use to improve their grade before a final submission. Likewise, Bryan [2] directly utilizes information in an Autodesk DXF file and geometric reasoning to compare a students work with the desired result. The method offers both offline assessment of work as well as online assessment. The latter uses a server with a Python implementation to analyze work in-process and display results through a browser that provides immediate feedback about a submitted drawing. Applying a similar strategy, Ingale et al. [3] have also developed an approach for providing automated feedback though focused more narrowly on how well a student creates sectional view drawings in a spatial visualization class. Other researchers utilize images created from drawings as the basis for assessment. Younes et al. [4] describe a system called ViTA which is CAD system independent that utilizes computer vision techniques to identify mistakes in drawings. Such mistakes include missing structural features and incorrect use of drawing properties such as section crosshatching, colors, and line types. Their results show 100% success and less than 1 minute to grade 500 submissions. However, they admit that greater complexity in the drawings limit its effectiveness. Their system utilizes a target solution created by the instructor for comparison. Khaleel et al. [5] propose the use of a deep convolution neural network (DCNN) implemented using the Cloud to provide Software as a Service (SaaS) to grade 2D CAD drawings. They generate images for deep learning by combining an RGB export of an Autodesk DXF drawing files submitted by a student with the instructor's solution key. Training of the neural network utilizes 1500 such images. Their solution implemented using a Cloud-based SaaS yielded an average 91% average accuracy in assigning a grade of A to E to a student's work. The author's use this success as motivation for applying their method to larger datasets.

The need to automate assessment of 3D parametric CAD has long been recognized as important and a more difficult undertaking than assessing 2D drawings. Early work by Baxter et al. [6] discuss the importance of this automation in the context of a CAD course where a large number of students and challenges finding instructional staff is a major motivation. They highlight the basic mechanisms for developing such tools using the API programming interface and Visual Basic that come with SolidWorks®. However, they do not present a final implementation only a strategy for developing the system. Branoff et al. [7] in a similar fashion highlight a range of grading strategies that can be applied to constraint-based CAD activities but do not get into the question of automation of these. Renu et al. [8] describe the use of a shape similarity algorithm based on a tessellated representations of a solid model and its mass properties to assess the similarity of a student's submission and the correct result. Use of a tessellation makes their method CAD system independent. Their approach of performing a shape similarity analysis and using mass properties seems to be the norm for what is available in practice.

Graderworks® [9] a product of Garland Industries is one such commercially available solution for the SolidWorks® CAD application that compares a student's model to a solution using volume, material and center of mass together with confirmation of the presence of fully constrained sketches. It also generates what is referred to as a *Composite Shape Score* using a stochastic process to account for variability when determining a match. Garland et al.[10] used this system to perform a statistical analysis on over 5200 models from different assignments to compare its capabilities to that of a human grader. Their results showed that the automatic grader was more accurate and repeatable when compared to teaching assistants using the four metrics mentioned previously. They conclude that this is because of the large amount of information present given the quantity of models. A human grader cannot grade this bulk of work as reliably as automation. Graders can also be found embedded within training LMSs. Ault et al. [11]describe their experiences using the grading engine within the Precision LMS® developed by PTC for instructors to assess proficiency in CAD modeling skills. According to the authors "The automatic grading algorithm can be based on the presence or absence of various feature types, feature count, dimensional values within specified features, order of feature creation, global part properties such as mass, volume, or location of the center of gravity". They were unable to accomplish their original goal of comparing its capabilities to a human grader due to unplanned differences in the grading criteria. However, their overall experience was that "the system is capable of assessing strategic knowledge as well as procedural CAD "skills", depending on the specific criteria selected by the instructor for assessment." SolidProfessor® [12] also provides an LMS which has the capability to confirm and document completion of exercises in training classes. These mostly use comparisons of stored solution values for mass properties or key dimensions to determine correctness. Their main limitation of these types of comparisons is that they can return false negatives with small numerical differences. Their use is limited to assignments that are guided where a student is expected to follow a set of instructions to arrive at a result. Their reliability falls off for more complex modeling assignments where a student maybe asked to reproduce a component from a drawing with more general guidelines to follow.

Challenges with Automating Approaches

Automating the assessment of modeling skills parallels the level of complexity incorporated into the model. This starts with just capturing a desired geometry in the shape that is modeled. Typically, this is not enough to support engineering product development functions where changes to the model are mandatory and best practices that support downstream uses of the model must be incorporated. Assessing the model at each level of complexity presents different challenges and the opportunity to use different automation techniques. To better understand these difficulties a capstone modeling assignment from an introduction to CAD class was evaluated. Figure 1 shows the overall assignment, a bicycle crank assembly, the right crank which is the component evaluated, and a table summarizing its structure and mass. 23 student submissions were used in the study. The component was modeled using Solidworks® 2022.



********** BODY DATA **********		
Body Number:	1	
Body Name:	Right Crank	
Mass:	0.3423	kg
Number of Features:	14	
Number of Sketches:	7	
Number of Extruded Bosses:	3	
Number of Extruded Cuts:	3	
Number of Revolved Cuts:	1	
Number of Holes:	2	
Number of Chamfers:	1	
Number of Fillets:	2	
Number of Circular Patterns:	2	
Number of Sketch Constraints	86	
Number of Sketch Geometries	41	

Figure 1. Capstone Modeling Assignment Component

<u>Model Shape and Mass Properties:</u> As discussed in the review of reported research, the use of a model's shape and mass properties is one of the most common forms of modeling assessment. Volume, surface area, mass and center of gravity measures are easily extracted by CAD systems and these values can be compared with the expected results taken from the desired solution. Figure 2 clearly shows the difficulty that can be encountered in using this approach for assessment. The results from the capstone assignment showed significant divergence from the solution that the instructor used to generate the drawings (red marker and line) that were given to the students as guidance for the assignment. A more careful examination of the work submitted during grading showed that many of the reasons for differences could be accounted for by errors from misinterpreting the drawing due to inherent ambiguities that are difficult to prevent in 2D views of more complex shapes. This includes

subtle differences that can occur based on how cosmetic features such as fillets and chamfers are interpreted and applied. Differences in interpreting and reproducing the shape of the spline interface used to connect the pedal to the bottom bracket bearing also contributed to differences in the final mass. Even if a student were to perfectly duplicate the instructor's modeling strategy, numerical errors when comparing floating point numbers makes it impossible to guarantee a correct comparison with an expected value. So, mass (volume, surface area) comparison approaches can generate false negatives when there is no discernible discrepancy in the result much to the frustration of a student who ends up wasting valuable time trying to correct a mistake that does not exist.

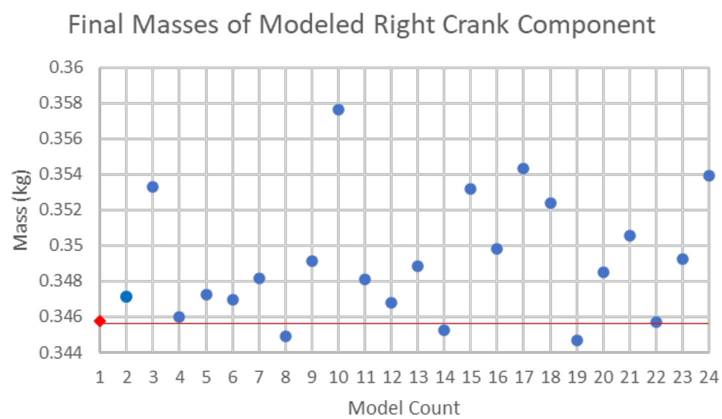


Figure 2. Mass Variation in Capstone Modeling Assignment

- Modeling History and Complexity: The modeling history is captured using a feature tree in modern parametric, feature-based CAD systems. It encapsulates a process used by the designer to create a model and so adds more information to an assessment beyond just the final shape. The challenge with these methods is that unless a modeling problem is highly constrained the process of arriving at a solution can be highly varied. Different choices of features and their order can yield the same final shape. However, they can be useful in verifying that specific modeling decisions have been made. For example, confirming if a particular type of feature is used in the model and are they added at a suitable stage in the modeling process. These would have to be stated requirements that a designer would need to meet in the modeling activity. Even with these guidelines there remain numerous modeling strategies that can be followed to arrive at a solution. Using the same crank arm capstone example, Figure 3 illustrates this using the strategies taken by three of the student designers. Designers 'A' and 'B' start by modeling the chainring end of the crank, though using different strategies themselves. 'A' takes an approach that strongly favors the use of additive features throughout, while 'B' includes the use of subtractive features to produce the shape of the five arms. Neither approach is wrong, and they both use around the same number of features (8 versus 9) to arrive at approximately the same shape. Most of the student designers' strategies aligned more with that of 'C' which starts with the crank arm before moving onto the chainring end. To demonstrate the variation that occurred across the entire class, automation was developed using the Visual Basic for Applications macro interface to Solidworks® to roll back the design history to each step where a feature was added, and the mass of the in-work model measured. The plots in the figure illustrate the results with the one on the left tracking the change in mass of each of the 23 models, and on the right a second representation showing the variation in mass that occurred at each modeling step. It's clear that over the first 12 steps there is significant variability in the masses accounted for by different modeling strategies. The convergence after the 12th step around a final mass value (0.3423 kg in the instructor's model) indicates the point at which all students have transitioned to adding cosmetic features such as fillets and chamfers which have a smaller impact on the mass. It is possible to use this in-process mass history by comparing it to that of the instructor's solution to determine how closely a student's strategy matches. In addition to the magnitudes, increases in mass indicate the use of additive features, and decreases subtractive. However, the question of whether such an assessment is valuable is predicated on whether the instructor can provide the student with enough guidelines so that they can be expected to closely match a desired solution. Unfortunately, this would defeat the purpose of a capstone assignment where students are assessed partly on their ability to develop their own effective strategy.

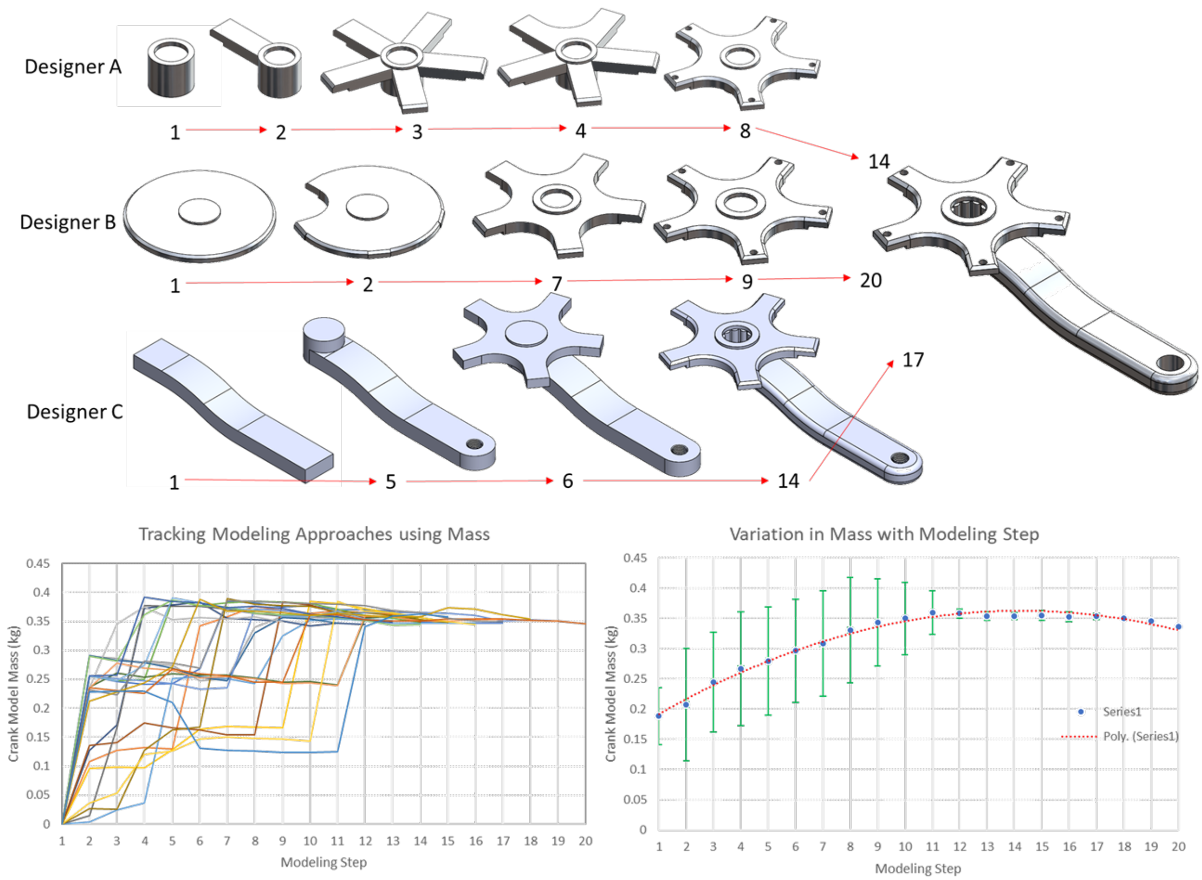


Figure 3. Tracking the Incremental Mass of a Component for Assessing Modeling Proficiency

In order to track complexity in modeling, the same automation also extracted from sketches the number of pieces of geometry and constraints used in building sketches. The results for the 23 students are summarized in Figure 4 sorted from the fewest to the most with the averages indicated by the horizontal lines. Also superimposed are the number of features and sketches used in the model. When compared to the instructor's solution (see table in Figure 1) almost all the students used more geometry and constraints when building sketches, and many significantly more. Since sketching is considered the most challenging skill to develop for modeling, this is probably not surprising. The trend also indicates that students who used more pieces of geometry also used more constraints which is again not surprising. The overlay of feature and sketch use was to observe whether there was any correlation between more complex sketches and fewer sketches and features. Except for the extreme cases (1 for fewest, 22 and 23 for most) none is obvious. This information might prompt the instructor to pay more attention to those students with above average use of geometry and constraints on their sketches (i.e. 15 to 23).

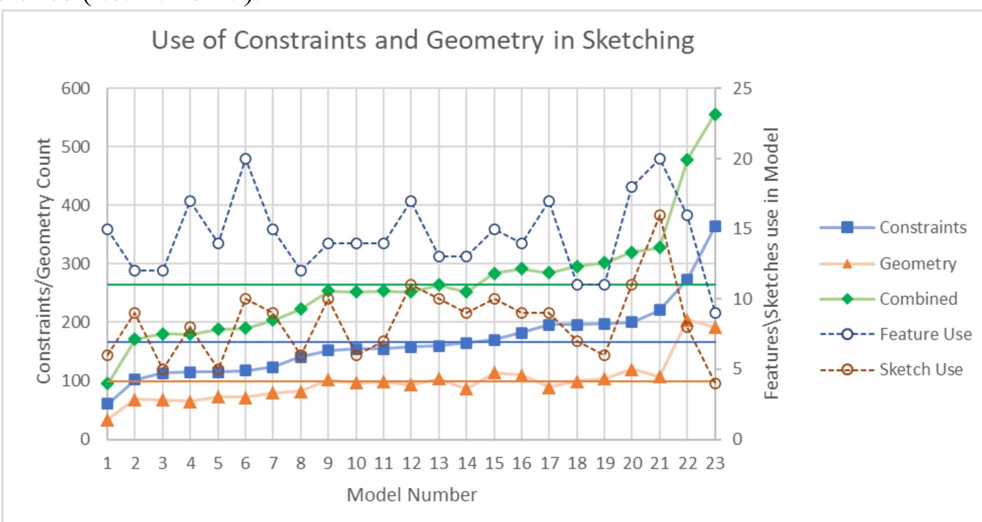


Figure 4. Sketch Constraint and Geometry Count as a Measure of Modeling Complexity

- Using Design Intent: This level of assessment evaluates whether a designer has created a model that can be parametrically modified. As with the previous shape and mass property methods it is an assessment of the final model and not the process used to create it. The automation of this assessment can be accomplished through the use of global or user parameters that the designer must incorporate into their model. These can be used to drive parameters on sketches or in defining features. Assessment automation can vary these global parameters and evaluate the impact on the final model. The simplest of these evaluations would be whether the model breaks within a stipulated range of values. Beyond this, testing the result goes back to using a shape and mass property method that performs a comparison with values from the desired solution, and faces the same challenges described previously. Without constraints on the modeling process design intent can be satisfied while using very different modeling strategies.

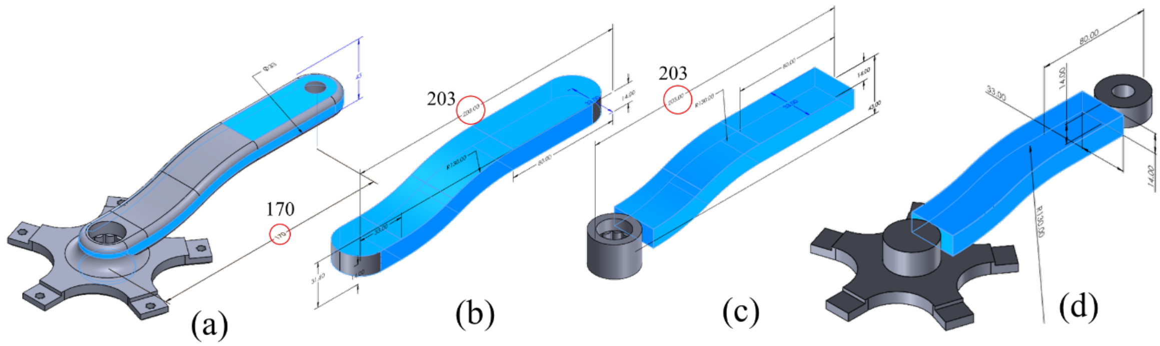


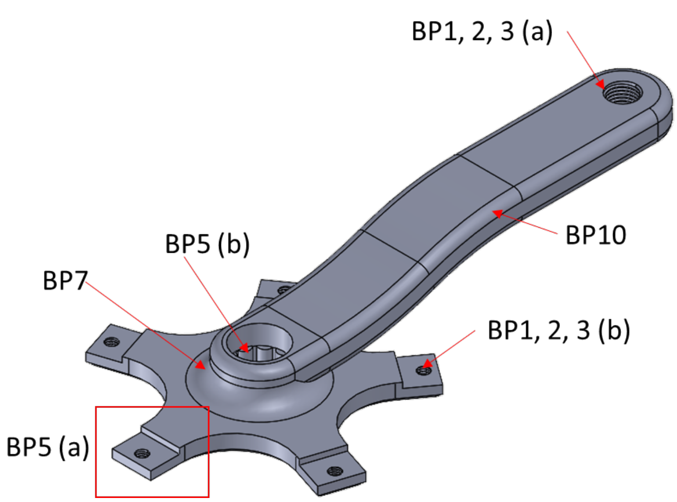
Figure 5. Challenges Satisfying Design Intent with Different Strategies

Figure 5 shows different modeling strategies applied by 3 students in creating the capstone right crank model which present different levels of difficulty in capturing the design intent of controlling the length of the crank arm. (a) shows the most efficient approach used in the desired solution where the dimension between the center of the pedal and bearing holes (170 mm) is controlled on a sketch. That sketch dimension can be linked directly to a global parameter that can be varied with automation to confirm the design intent is present. In (b) the designer chose to create the rounded ends using fillets after creating a base feature using a sketch extruded from the side. To capture the design intent, the length dimension must be defined using a relation that adds the end diameter to the global variable. The same is true in (c) with the added complication of ensuring that the two features over which the design intent must now be managed remain in contact with matching width and diameter dimensions. The example in (d) requires managing the design intent over three features. As with the modeling strategies none of these approaches is technically wrong if the length of the arm changes without breaking the model assuming the assignment did not explicitly require that the most efficient approach be the solution. Experience has shown that it's important to teach students the importance of paying attention to design intent as they develop their modeling strategy and not trying to add it as an afterthought.

- Using Best Practices: The final level of assessment is determining whether best practices have been incorporated into a model. This raises the question of what are considered "best practices" and are these universally agreed upon as such. What is clear from practice is that CAD systems provide the flexibility for end-users to enforce rules in how a model is created by their modelers and engineers to support their product development needs and that these can vary across disciplines. For example, the way a model is created for a component that is machined would apply a different modeling strategy to a similar shape that might be cast or forged. For certain manufacturing domains, CAD systems have encapsulated best practices by providing the designer with application specific features to model with. Examples of these include sheet metal, composites, weldment and mold design. Assessment of models created using such applications is typically easier since the constraints enforced using these features lead to expected manufacturable shapes and parametric characteristics in the final model. Examples of these would be the presence of a draft feature on the side walls of a molded component, or the bend radius between two walls in a sheet metal part. CAD systems that implement application specific modeling add-ins typically include analysis tools that can be used to assess these features in a final model. These can be appropriately tied to assessment automation. Outside of these application areas there are some broadly accepted generic best modeling practices that a designer can be expected to use. These include but are not limited to the following:

1. Use a hole feature for circular pockets under 1" in diameter (machinability).
2. Use a hole feature instead of circle on a sketch for a through hole.
3. Use a hole feature when a thread is included.
4. Use a hole feature when a counterbore or countersink is included.
5. Pattern large arrays using features rather than with geometry on a sketch.
6. Use a mirror feature for symmetrical parts.
7. Use the fillet and chamfer features on 3D model edges over sketch fillets and chamfers.
8. Include fillets and chamfers before shelling.
9. Draft before shelling.
10. Fillet and chamfer as late as possible.

Assessing the use of best practices is again a process of comparing with an expected result where they have been correctly applied. Best practice usage rates for the capstone right crank modeling assignment were assessed and are summarized in Figure 6. This shows that a significant number of the 23 students (almost 40%) did not correctly follow the best practices they were taught in earlier assignments about how and when to use the hole feature. Almost all avoided using feature patterning for the five arms and the teeth in the spline connector in favor of building more complicated sketches. All correctly applied the fillet feature over sketch filleting for the arm edges though about 25% did so at an early stage when the net shape of the model was yet to be completed. Experience has shown that students tend to be biased towards using modeling skills they learn earlier in a course when their enthusiasm maybe higher and the amount of time they can put into learning that skill greater. This tends to favor use of sketching skills and can bias their judgement on applying a best practice such as mirroring or patterning features. The ability to judge when a best practice is needed without it been explicitly required by the instructor in an assignment is an advanced skill that takes time and practice to develop.



Best Practice	Usage Rate
BP1 (a,b)	61%
BP2 (a,b)	61%
BP3 (a,b)	61%
BP5 (a)	13%
BP5 (b)	4%
BP7	100%
BP10	74%

Figure 6. Best Practice Usage in Capstone Right Crank Modeling Assignment

Using Complete Modeling Activity Data

One obvious distinction in the metrics that can be used for assessment is whether they are extracted from the final result, or the process used to arrive at the result. The latter information is considered to reside in the design history of the model captured in a feature tree. However, this is an incomplete picture of the process followed to arrive at a result. It's a snapshot of a strategy to build the component, but not a full picture of the steps taken by the designer. Missing are any changes in strategy that may have occurred along the way and information about how much time was spent on different modeling activities. This missing information is valuable in assessing how efficiently a student designer may have completed a modeling task and in identifying activities where the effort was unexpectedly high. This would help differentiate between those cases where two students might arrive at the same result using the same recorded strategy but where one spent significantly more time and rework to get there. Identifying those students and tracking these metrics over time is potentially a useful guide for instructors to assist those struggling in acquiring modeling skills. A major question to using this information is whether it can even be extracted from a modeling session.

- Overview of Implementation: To examine this question, the authors investigated the CATIA® V5R18 software by Dassault Systemes. CATIA® contains several features for recording session data that generates log files, the most important being the *Statistics* feature that is enabled in the *Options* menu. As shown in Figure 7, the system logs statistics for the time spent in workbenches (e.g. Sketcher, Part Design, Assembly Design), time spent using specific commands (e.g. feature definition, sketch constraints/geometry/operations) in those workbenches, and the session statistics [13].

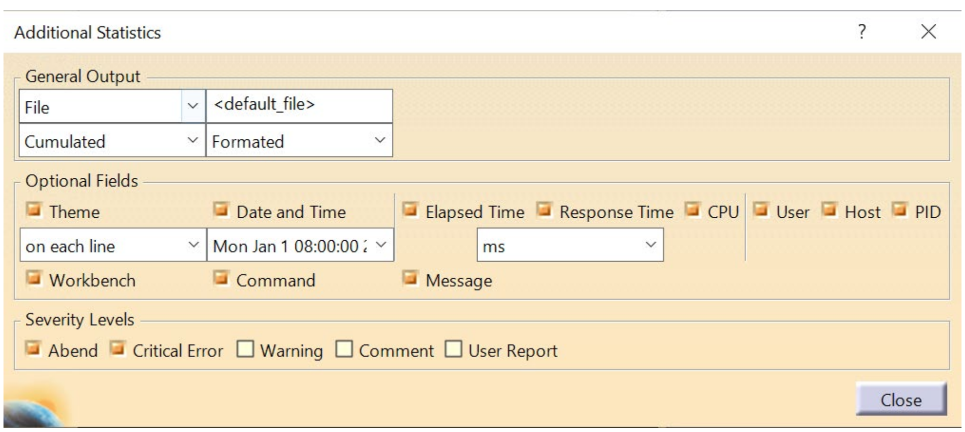


Figure 7. Statistics that Can be Generated during a CATIA Session

Statistics are saved to three log files:

- *Command log* – contains entries for the workbench and specific commands used as well as the elapsed time.
- *Session log* – contains entries for the total elapsed time and computational usage for the entire CATIA session.
- *Workbench log* – contains entries for the elapsed time in each workbench.

The most important log is the Command log. This contains the granular information about each command executed during a modeling session including a time stamp.

It was discovered that important information needed to properly parse the log could not be generated given the options available in the Statistics dialog. Surprisingly this included the model's name in which a command was executed (file name), the in-work object (a feature or sketch), or even what was selected. So, a user could move back and forth between models that were open in a CATIA® session and the Command log would not identify in which model the command was executed. As such, there was no way to unambiguously identify what a user was working on using just the CATIA® log files.

As a work around, the authors developed an additional *Working log* that gathered data simultaneously with the system log files while CATIA® is running. This working log was designed to specifically track the file name of the active model where a command is executed and the in-work object. The log was created by running a background process that watches for the CATIA® startup process and then logs the in-work object and its filename approximately every 2 seconds. The Python packages *WMI*, *win32com*, and *pyvba* were used for watching the startup process and for interacting with CATIA®. Table 1 summarizes the schema of the Command and Working log files with the critical data that can be used to interpret a designer's actions highlighted. The *time* entry (highlighted in green) is used to synchronize the information in both files.

Table 1. Header descriptions for the command and working log files.

Header	In Command log?	In Working log?	Description
them	X	X	Theme (COMMAND or WORKING)
time	X	X	Time of the action
elps	X		Elapsed time spent in the command
rtim	X		Response time (elapsed time minus user wait time)
cpus	X		CPU time spent in the command
user	X	X	Username associated with the computer profile
host	X	X	Host machine name
upid	X		Universal program identification number
Session	X		Session status (Start or End)
Command	X		Internal command name
NLS	X		External command name
Origin_Header	X		Origin header
mode	X		Command mode (Foreground or Background)
CurrentWorkbench	X		Internal current workbench name
NLS_CurrentWorkbench	X		External current workbench name
CurrentWorkshop	X		Internal current workshop name
NLS_CurrentWorkshop	X		External current workshop name
filename		X	Name of the file
inwork		X	Current in-work object name
inworktype		X	Current in-work object type
sel		X	Currently selected object name
seltype		X	Currently selected object type

The CATIA® log files are all stored in one location determined by each user's settings. There are also options for each CATIA® session to generate its own file or to aggregate the information in a single file. The latter option was used to avoid the challenge of having to parse and synchronize multiple log files. However, there is a file size limit that needs to be appropriately set which if exceeded will lead to a copy being created and the log file being reset. Though not included in the current implementation it would be a simple solution to automate tracking the Command log file size and extracting information before a reset occurs.

From the two log files, time-based modeling performance metrics can be determined. A full list of such metrics that can be tracked is given in Table 2. These include efficiency metrics such as the *Sketching Efficiency* and *Modeling Efficiency* which can be used to determine whether a particular student designer is within the expected norms for a modeling assignment.

Table 2. Modeling Performance Metrics

Metric Name	**Variable**	**Description**	**Filters (If Applicable) and Calculation**
Start Date	D_s	When the model is first opened	Minimum of datetime
End Date	D_e	When the model is last closed	Maximum of datetime
Duration of Modeling Effort	d_m	Time spent working on model	$D_e - D_s$
Total Modeling Time	T_m	Cumulative time while model was open in CATIA	Maximum of time_passed
Effective Modeling Time	T_{em}	Time while sketcher workbench is open, or a feature is being created.	∑elps
Modeling Efficiency	ε_M	Percentage Ratio of Effective Modeling time to Total Modeling Time	$T_{em}/T_m * 100$
Total Sketching Time	T_s	Time during which the Sketcher Workbench is open.	∑elps | NLS_CurrentWorkbench == Sketcher
Total Feature Creation Time	T_f	Time during which a feature (Pad, Pocket, Shaft, Groove etc.) is being created	∑elps | Origin_Header == PrtCfg
Sketcher Effort	e_s	Percentage ratio of the Total Sketching Time to the Effective Modeling Tim	$T_s/T_{em} * 100$
Effective Sketching Time	T_{es}	Time during which Sketching commands are being executed	Origin_Header == CS0WKS
Sketching Efficiency	ε_S	Percentage Ratio of Effective Sketching time to Total Sketching Time	$T_{es}/T_s * 100$
Sketcher Geometry Time	T_{sg}	Effective sketching time devoted to creating geometry	∑elps | Origin_Header == CS0WKS & NLS != Constraint
Sketcher Geometry Effort	e_{sg}	Percentage ratio of Sketcher Geometry Time to the Effective Sketching Time	$T_{sg}/T_{es} * 100$
Sketcher Constraint Time	T_{sc}	Effective sketching time devoted to creating constraints	∑elps | Origin_Header == CS0WKS & NLS == Constraint
Sketcher Constraint Effort	e_{sc}	Percentage ratio of Sketcher Constraint Time to the Effective Sketching Time	$T_{sc}/T_{es} * 100$
Sketch Geometric Elements	N_{sg}	Number of geometric elements created in sketches	∑i | Origin_Header == CS0WKS & NLS != Constraint
Sketch Constraint Elements	N_{sc}	Number of constraint elements created in sketches	∑i | Origin_Header == CS0WKS & NLS == Constraint
Sketch Elements	N_e	Total number of elements created in sketches	$N_{sg} + N_{sc}$

Sketch Deletions	N_d	Number of deletions performed during sketching.	∑i | NLS_CurrentWorkbench == Sketcher & NLS == Delete
Sketch Percentage Deletion Rate	R_d	Percentage ratio of deletions to the number of sketcher elements created	$N_d/(N_{sg} + N_{sc}) * 100$
Feature Creation Effort	e_f	Percentage ratio of the Total Feature Creation Time to the Effective Modeling Time	$T_f/T_m * 100$

- Case Study: A pilot study was conducted to evaluate the potential capability to collect and parse the information collected using the previously described methodology. Three desingers were given two entry-level models to create in CATIA®. These models shown in Figure 8 were selected for their different use of modeling features. The strategy for the *Transition Base* component favors use of prismatic extruded features such as Pads and Pockets. The *V-Pulley* strategy favors use of rotational features such as shafts and grooves. The data for each was collected using the implementation previously described.

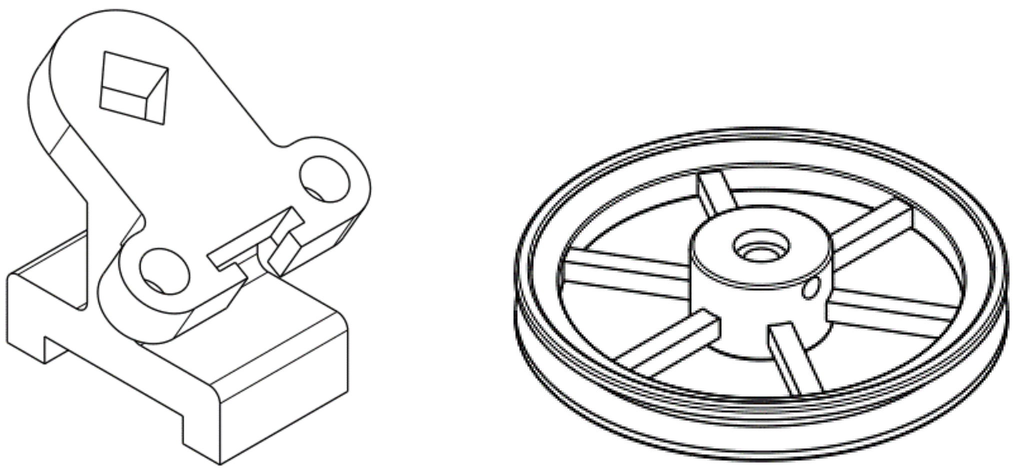


Figure 8. The Transition Base (left) and V-Pulley (right) part models used in the study.

The final values for variables are calculated from the resulting data according to the filters and calculations listed in Table 2. For example, to evaluate the total sketching time (T_s), the filter (NLS_CurrentWorkbench == Sketcher) should be applied to the entries, with the sum of the elapsed time (*elps*) of the filter's result yielding the desired value. To evaluate specific files or users, the appropriate grouping constraints should be applied. From further inspection of Table 2 the metrics can be largely divided into two groups. These groups impact:

- *Modeling time and efficiency* – These metrics can be used to observe the holistic objectives of the modeling process.
- *Sketching metrics* – These metrics are used to investigate the sketching efficacy and efficiency. The final model should be very dependent on the sketch geometry, as it defines the foundational structure of the object.

Table 3. Results for some of the modeling time and efficiency metrics.

Designer	File	Total Modeling Time (hh:mm:ss)	Effective Modeling Time (hh:mm:ss)	Modeling Efficiency (%)
1	Transition Base	00:15:54	00:14:30	91.19
	V-Pulley	00:26:00	00:19:26	74.77
2	Transition Base	00:09:40	00:08:02	82.99
	V-Pulley	00:11:40	00:10:56	93.69
3	Transition Base	00:28:18	00:19:46	69.83
	V-Pulley	00:18:26	00:16:41	90.49

Using the data collected from the three designers engaged in the case study, the modeling time and efficiency were calculated and are shown in Table 3. From this limited test data, it is possible to observe how these metrics might be used to identify a novice designer who may be struggling. For example, the lowest efficiency was 69.83% by Designer 3 for the *Transition Base*. Repeated instances of lower efficiency across modeling assignments can be used to alert an instructor to a student who might be struggling with acquiring a modeling skill.

Table 4. Results for some of the sketching metrics.

Designer	File	Sketcher Geometry Time (hh:mm:ss)	Sketcher Geometry Effort (%)	Sketcher Constraint Time (hh:mm:ss)	Sketcher Constraint Effort (%)	Combined Time (hh:mm:ss)
1	Transition Base	0:01:40	24.39	0:05:10	75.61	0:06:50
	V-Pulley	0:01:54	20.27	0:07:29	79.73	0:09:23
2	Transition Base	0:01:35	32.93	0:03:47	67.07	0:05:22
	V-Pulley	0:02:26	35.26	0:04:29	64.74	0:06:55
3	Transition Base	0:03:56	86.72	0:00:36	13.28	0:04:32
	V-Pulley	0:03:05	82.35	0:00:40	17.65	0:03:45

In a similar fashion looking at the sketching metrics summarized in Table 4, an interesting difference can be observed in the relative time spent by Designer 3 in creating sketch geometry versus creating sketch constraints when compared to the other two. Unlike

Designers 1 and 2, a significant amount of time is spent creating sketcher geometry in comparison to adding constraints. Examination of the models show different strategies with Design 3 favoring simpler sketches spread over more features. This is illustrated in Figure 9. More complex sketches would require significantly more time to fully constrain particularly for a novice. Interestingly, this potential advantage in more efficient sketching did not yield the best overall modeling time for Designer 3 as can be seen from the *Total Modeling Time* in Table 3. This time includes the effort taken to first develop a modeling strategy i.e. the features that will be used and the sequencing. It may be that Designer 3 took more time to develop this when planning their strategy with the goal of using simpler sketches before starting to model them.

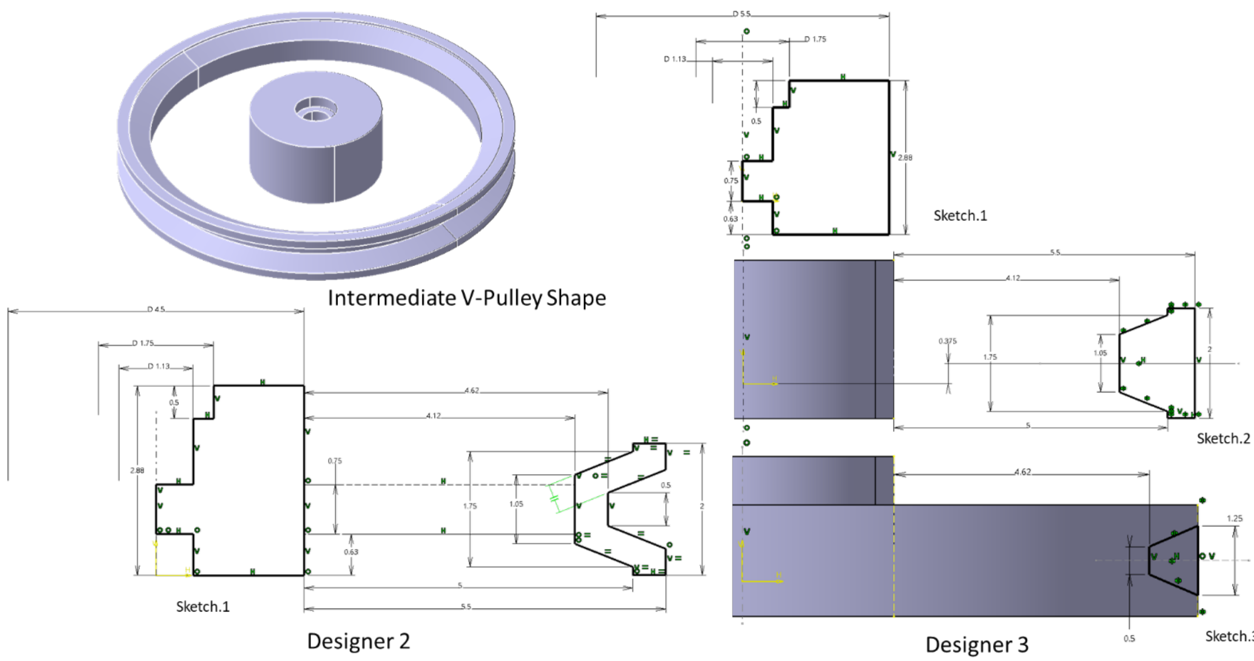


Figure 9. Modeling Strategies with Different Sketch Complexities

Discussion

Current automated tools for CAD modeling skill assessment have limitations. They work best when the modeling assignment is highly constrained by the instructor which limits the range of modeling strategies a student can use to arrive at the answer. This typically means simpler assignments which is what any training pedagogy starts with to develop a novice's skill. In this sense they are valuable particular in environments where large numbers of students need to be assessed. However, consideration must be given to the limitations of these tools particularly those that directly compare a metric such as mass or volume extracted from a student's work to that of the expected solution. Experience in using such graders built into SolidProfessor® [12] and StudyCAD® [14], [15] by the authors have shown that students need a backup mechanism when auto-grading is used to assist them in correcting their work before a final solution is submitted. False negatives can be generated due to reasons that are not relevant to the modeling skill being assessed. Use of a teaching assistant or a peer-review are potential approaches to provide this safety net though timeliness of feedback becomes an issue. Another approach that has been adopted by the authors is to provide a defeatured part body of the solution within a starter file that is used by the student to model their answer in an overlapping part body. Missing or dimensionally incorrect features can be visualized to prompt self-correction. It's also easier for the instructor to hone in on errors when providing help. One drawback of this approach is that it limits the practice of reading and interpreting views in drawings which can be a secondary learning outcome in a CAD class as is the case for the capstone example discussed earlier.

As illustrated earlier, more complex assignments where students are expected to do more independent thinking on their modeling strategy, capture design intent, and use appropriate best practices are much more challenging to automate their assessment. It might even be true that because more complex assignments such as a capstone project are meant to evaluate overall mastery rather than a specific skill, that automation is not appropriate to use in these cases. Regardless looking at how a student approaches a more complex modeling assignment where they are expected to independently apply what they have previously practiced raises some intriguing questions. For example, why did the three student designers decide on the strategy they followed in Figure 3? There was a split decision where two designers chose to model the chain ring end (more complex) first and the other the pedal end which was indicative of the majority. Are there other not so obvious modeling decision points across the many different strategies followed? With a larger sample of student solutions, can patterns of decision-making be identified that lead to categories that group students based on how they approach a modeling problem? This type of information might be useful to an instructor in recognizing the tendencies of each student as they approach a modeling problem.

There is of course the question of whether a tendency in of itself is a poor modeling practice that needs to be corrected. For example, Figure 9 illustrates two tendencies that students might present if asked to model this shape. The solution on the left might suggest a tendency to jump into the sketcher to try and accomplish as much as possible as quickly as possible. While the solution on the right might suggest a student who does more planning by visualizing different feature decomposition strategies before starting the modeling activity. As mentioned earlier, there is also a tendency for some students to lock into modeling skills they learn earlier in a course which might bias those students who have fallen behind towards overusing the sketching tool. As an assignment neither solution would be wrong unless the instructor gave guidance that would influence use of one over the other e.g. using the simplest sketches possible favoring Designer 3's solution or using the fewest features possible favoring Designer 2's solution. However, there is value to developing within engineers the skill of planning as opposed to jumping into in a problem without much forethought. Understanding of how these tendencies might play out in a modeling assignment can help instructors provide better instructions that emphasize a preferred tendency.

On the question of using more complete modeling activity data, the case study presented shows the feasibility of obtaining this data using a leading CAD system. Some of the automation for doing this developed in Python can potentially be adapted for use with other CAD systems. However, this would depend on whether those systems generated log files with detailed command instructions tagged with a time stamp. For example, the Solidworks Rx® app allows users to generate log data from sessions that captures the session duration and user interface command sequence in a performance log file. This is typically used to report crashes of the system during modeling. Further investigation of this is needed to determine if it contains the desired data to replicate the automation developed for CATIA®. Although generalized conclusions are difficult to draw from the simple case study presented, it does show that a mechanism for capturing information that more completely tracks a designer's activities is possible. This can be used as another tool for extracting information about the modeling process to monitor and assess modeling skills development. For instance, one might expect the sketcher effort to correlate closely with modeling efficiency if the model is heavily sketch dependent. A heatmap, like the one shown in Figure 10, can be used to reveal tendencies in a particular modeling exercise. Although not conclusive due to the limited data, one might question why sketcher effort and sketcher geometry effort have a low correlation. This might be the case as discussed earlier if more of the designers favored use of complex sketches over user of simpler features, as two of the three designers did in this particular case. Finally, getting feedback from students on the collection of this additional data, and whether its analysis helps them perform better by them taking corrective action during the modeling process, is critical to the development and acceptance of grading automation. Due to the transition from away from CATIA® this will have to wait until the automation described is redeveloped using the Solidworks® platform.

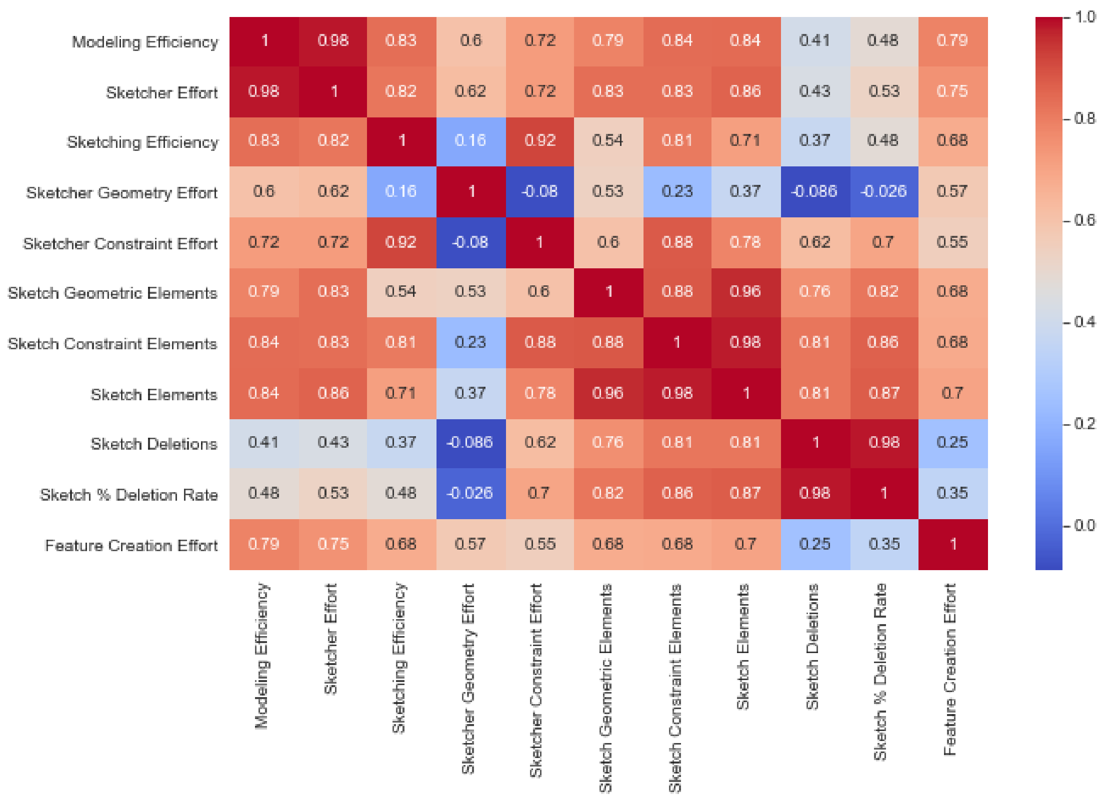


Figure 10. A heatmap of all the captured metrics listed in Table 2.

Conclusion

Automated assessment of CAD modeling student activities is an indisputable efficiency benefit for instructors. Methods are available and some implemented in available tools including CAD training LMSs to do this, but are limited in their effectiveness. Challenges exist in reliability particularly if applied to more complex models and in extending the assessment beyond a student's ability to arrive at a final shape that matches a solution. Automation becomes much more challenging when the process by which a student arrives at a solution is to be evaluated. This is even more so when limited instructions are provided as in a capstone modeling assignment when what is being partly assessed is the ability of the student to independently make decisions about modeling strategies that include capturing design intent and use of modeling best practices. A case study has been used in this paper to highlight some of these challenges. This shows that for a complex model many strategies are possible and that in the absence of guidelines, which is typical of a capstone assignment, an instructor may find it difficult to justify penalizing a strategy that does not match what they might have generated as their preferred solution. This case study and some exploratory work on extracting a more complete set of performance metrics that considers time spent on activities, can be useful in identifying tendencies amongst students on a particular assignment and in general. Knowing such modeling tendencies and whether they should be reinforced or discouraged, has the potential to be an additional tool for an instructor to use to help students develop better CAD skills. This broadens the role of assessment automation beyond what an instructor currently does when grading assignments, to extracting new insights into the processes that students use when modeling which can impact the way instructors approach assessment.

References

[1] K. Hekman and M. Gordon, "Automated Grading of First Year Student CAD Work," in *2013 ASEE Annual Conference & Exposition Proceedings*, Atlanta, Georgia: ASEE Conferences, Jun. 2013, p. 23.235.1-23.235.11. doi: 10.18260/1-2--19249.

[2] J. A. Bryan, "Automatic grading software for 2D CAD files," *Comput. Appl. Eng. Educ.*, vol. 28, no. 1, pp. 51–61, 2020, doi: 10.1002/cae.22174.

[3] S. Ingale, A. Srinivasan, and D. Bairaktarova, "CAD Platform Independent Software for Automatic Grading of Technical Drawings," presented at the ASME 2017 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference, American Society of Mechanical Engineers Digital Collection, Nov. 2017. doi: 10.1115/DETC2017-67612.

[4] R. Younes and D. Bairaktarova, "ViTA: A flexible CAD-tool-independent automatic grading platform for two-dimensional CAD drawings," *Int. J. Mech. Eng. Educ.*, vol. 50, no. 1, pp. 135–157, Jan. 2022, doi: 10.1177/0306419020947688.

[5] M. F. Khaleel, M. A. Sharkh, and M. Kalil, "A Cloud-based Architecture for Automated Grading of Computer-Aided Design Student Work Using Deep Learning," in *2020 IEEE Canadian Conference on Electrical and Computer Engineering (CCECE)*, Aug. 2020, pp. 1–5. doi: 10.1109/CCECE47787.2020.9255825.

[6] M. Guerci and D. Baxter, "Automating An Introductory Computer Aided Design Course To Improve Student Evaluation," in *2003 Annual Conference Proceedings*, Nashville, Tennessee: ASEE Conferences, Jun. 2003, p. 8.258.1-8.258.9. doi: 10.18260/1-2--11479.

[7] T. Branoff, E. Wiebe, and N. Hartman, "Integrating Constraint Based Cad Into An Introductory Engineering Graphics Course: Activities And Grading Strategies," presented at the 2003 Annual Conference, Jun. 2003, p. 8.726.1-8.726.10. Accessed: Jun. 23, 2023. [Online]. Available:

https://peer.asee.org/integrating-constraint-based-cad-into-an-introductory-engineering-graphics-course-activities-and-grading-strategies

[8] R. Renu, A. Garland, S. Grigg, J. Minor, and N. Yasmin, "Improving Engineering Graphics Grading Using a Shape Similarity Algorithm: An Initial Investigation," in *Volume 3: 18th International Conference on Advanced Vehicle Technologies; 13th International Conference on Design Education; 9th Frontiers in Biomedical Devices*, Charlotte, North Carolina, USA: American Society of Mechanical Engineers, Aug. 2016, p. V003T04A018. doi: 10.1115/DETC2016-59597.

[9] "Graderworks," Garland Industries LLC. Accessed: Jan. 20, 2024. [Online]. Available: https://garlandindustriesllc.com/index.php/graderworks

[10] A. P. Garland and S. J. Grigg, "Evaluation of Humans and Software for Grading in an Engineering 3D CAD Course," presented at the 2019 ASEE Annual Conference & Exposition, Jun. 2019. Accessed: Jun. 23, 2023. [Online]. Available: https://peer.asee.org/evaluation-of-humans-and-software-for-grading-in-an-engineering-3d-cad-course

[11] H. Ault and A. Fraser, "A Comparison of Manual vs. Online Grading for Solid Models," in *2013 ASEE Annual Conference & Exposition Proceedings*, Atlanta, Georgia: ASEE Conferences, Jun. 2013, p. 23.31.1-23.31.13. doi: 10.18260/1-2--19045.

[12] "SolidProfessor | Build Skills with CAD & CAM Online Courses," SolidProfessor. Accessed: Jan. 20, 2024. [Online]. Available: https://www.solidprofessor.com/

[13] "Statistics." Accessed: Jan. 25, 2024. [Online]. Available: http://catiadoc.free.fr/online/bascugen_C2/bascustatistics0000.htm

[14] "StudyCAD." Accessed: Feb. 02, 2024. [Online]. Available: https://www.studycad.com/

[15] D. M. Yip-Hoi and J. G. Welch, "Enhancing a Blended Learning Approach to CAD Instruction Using Lean Manufacturing Principles," presented at the 2015 ASEE Annual Conference & Exposition, Jun. 2015, p. 26.656.1-26.656.18. Accessed: Feb. 06, 2022. [Online]. Available: https://peer.asee.org/enhancing-a-blended-learning-approach-to-cad-instruction-using-lean-manufacturing-principles