Access provided by:
National Cheng Chi University
Sign Out

Browse   My Settings   Get Help

Evaluation of Student's 3D Modeling Capability Based on Model Completeness and Usage Pattern in K-12 Classrooms

Publisher: IEEE   Cite This   Cite This   📄 PDF

5 Author(s)   Yi-Chieh Wu ; Wen-Hung Liao ; Chen-Yu Liu ; Tsai-Yen Li ; Ming-Te Chi   **View All Authors**

Export to
Collabratec   **Alerts**

30
Full
Text Views

Manage
Content
Alerts
Add to
Citation
Alerts
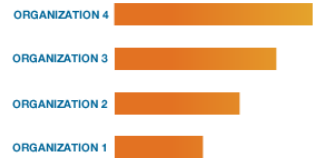
Abstract

Document Sections

I.   Introduction

II. Related Work

III. The Proposed Methodology

IV. Results and Discussion

V.   Conclusion

Authors

Figures

References

Keywords

Metrics

More Like This

Down
PDF

Abstract: As more schools incorporate 3D printing into their curriculum to stimulate the creativity of K-12 students with a learning-by-doing approach, it becomes crucial to unders... **View more**

Metadata
Abstract:
As more schools incorporate 3D printing into their curriculum to stimulate the creativity of K-12 students with a learning-by-doing approach, it becomes crucial to understand how users work with 3D modeling tools. In this paper, we aim to develop model and usage-pattern-related features to quantize students' performance on 3D modeling operation. The dataset is gathered from the Affiliated Experimental Elementary School (AEES) of National Chengchi University. Participants' operation log and finished work for specific 3D modeling software are recorded and analyzed. In all our lesson plans, students are required to create structurally stable and printable 3D models. Three modeling software with different levels of difficulty have been introduced and tested. The collected data include screen recording, software operation log, experts evaluation, and interviews with students, which are employed for subsequent qualitative evaluation as well as quantitative analysis. With our proposed approach, we are able to identify the key factors affecting students' learning experience and performance in terms of model completeness and usage pattern. Through these indicators, instructors can understand student's learning status of 3D modeling software more comprehensively.

≡ **Contents**

SECTION I.
Introduction

Due to advances in 3D printing technology and afford-ability of 3D printers, many teachers have introduced 3D printers and relevant lesson plans into K-12 classrooms. The common objective is to stimulate the creativity of K-12 students with a learning-by-doing approach. The process of 3D manufacturing usually includes the following stages: 1) constructing models using software or scanning devices, 2) editing and refining, and 3) 3D printing. In order to provide teachers and students a platform for online communication, idea exchange, and peer evaluation, Lin *et al.* developed the 3D Model Co-learning Space (3DMCLS) [1] in response to the needs of the course. When students have finalized their models, they can upload their works through the web interface. Uploading can also be achieved through the in-app function of Qmodel Creator [2]. Standard Template Library (STL) format is supported by 3DMCLS, which means that teachers can conveniently utilize this platform by choosing any modeling software that can export STL file. Fig. 1 shows models on 3DMCLS created by different modeling software, including Tinkercad, Qmodel Creator and 123D Design.
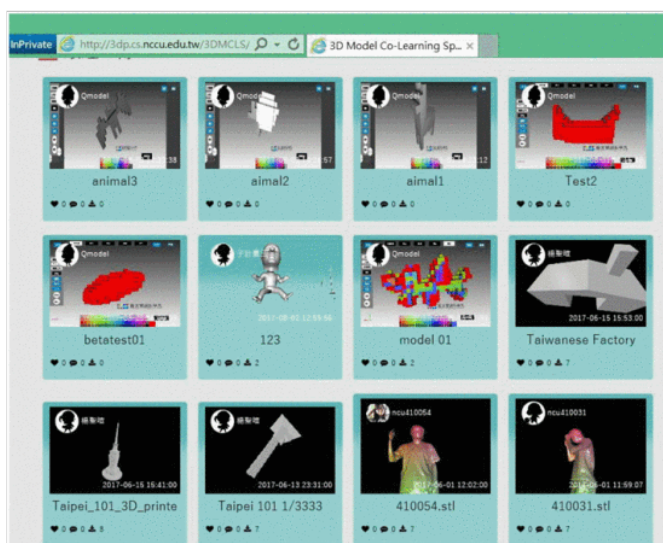

Figure 1.
Models created by various 3D software on 3DMCLS [1]

We have conducted research regarding 3D modeling usage patterns of Qmodel Creator and have identified indicators that are applicable to other evaluation process [3]. They are: Trial and Error Period (TEP), Implementation Period **(IP),** Effective Operating Period (EOP), and Degree of Detail (DoD). However, increasing choices for modeling software make the assessment more challenging. Furthermore, most educators based their evaluation on a subset of these indicators due to limited resources. Although we can easily obtain the log file of Qmodel Creator, we still need to spend efforts to examine students' operating procedure when other modeling applications are employed. In addition, we are not certain if the score given by teacher on a student's work is indeed related to the quality of the finished model itself. Hence, we need to develop quantitative indicators to better characterize the constructed 3D model.

In this study, we take multiple features of a model into consideration, including the number of connected components, the largest component ratio (L.C.R.) of a model, as well as shape (C_f), partition (C_p), and block-ratio (C_r) complexity which will be defined in the subsequent section. Compared with the Degree of Detail (DoD), these features are more representative of the model structure. For those finished models in the low-score group, the number of connected components is negatively correlated with their scores, while the L.C.R. of a model is positively correlated with the scores. The number of components and L.C.R. allow us to identify students who have problems when using thw modeling software and provide guidance for them in time. Moreover, we adopt video segmentation approach to make it easier to retrieve and label user operation indicators, so that the same features of user behavior can be observed and compared when different software are employed. According to the operation patterns, we roughly divide students into the following four categories: Novice with little interest, Novice with caution, Intermediate, and Advanced. The proposed classification and quantitative indicators for models will help teachers to track student's progress and identify aspects that require further improvement.

The remainder of this paper is organized as follows. In Section 2 we outline related works on model complexity and video segmentation approaches based on computer vision technology. Section 3 elucidates our proposed methodology and experimental procedure, as well as the indicators used for evaluation. In Section 4, we compare and discuss experimental results. Section 5 concludes this paper.

SECTION II.
Related Work

In our previous studies, we employ Degree of Detail (DoD) to quantize the fine-grain structure of 3D models. It works reasonably well on Qmodels. However, we detected an absurd condition with models created by other software such as Tickercad [4]. A flat 3D model such as the one shown in Fig. 2-(a) will exhibit a high DoD since the average side length is used as one of the parameters.
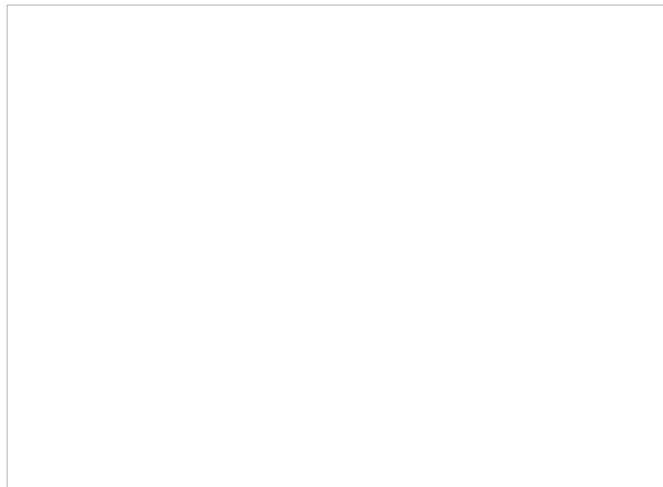

Figure 2.
Dod from tinkercad models. (a) 37.138 (unreasonable value) (b) 3.51 (normal value).

In order to extract more reliable structural information from 3D models, we surveyed literatures in quantifying the complexity of an entity. Fiolka *et al*. proposed SURE descriptor in 3D point clouds based on entropy [5]. Liao and Chen also calculate the complexity of images based on entropy using information-theoretic modeling for logo design analysis

We also exploit the learning process stages proposed by Dreyfus *et al.* [7], and develop indicators such as TEP, IP, and EOP, to mark users' actions in the past research. These indicators reflect a user's status and the period during each step of the operating procedure. When users operate modeling software (except for Qmodel Creator), we record the screen during the whole process. Background subtraction technique with an adaptive threshold has been devised to perform motion detection and provide the labeling function.

SECTION III.
The Proposed Methodology

In this section, we propose two coarse categories of quantitative indicators, namely, 3D model related features and usage pattern related features, to address performance evaluation of general 3D modeling task. We have conducted five 3D modeling software courses at AEES. The finished models and operating records are used as our dataset. We also have experts evaluation for these works, as well as face-to-face interviews with students.

A. 3d Model Related Features

For model analysis, we extract five features for model representation. They are: number of connected components, largest component ratio, shape complexity, partition complexity, and block-ratio complexity, respectively, as shown in Fig. 3.
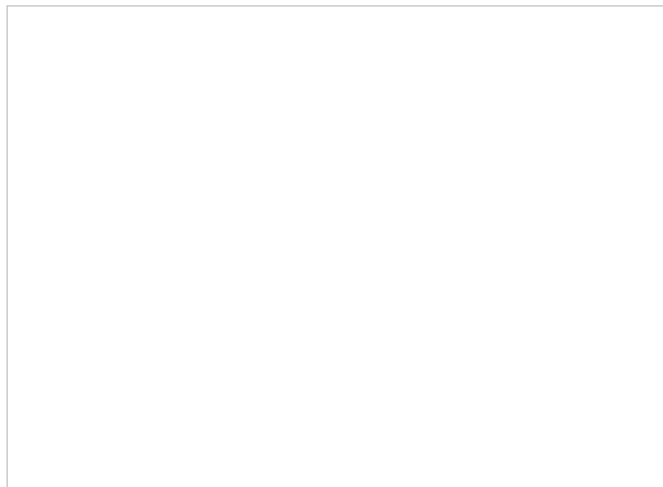

Figure 3.
Workflow of model analysis

1) The Number of Connected Components
A model with a large number of components will possess a higher risk of failure in printing and result in waste of materials. We also observe that it is usually created by inexperienced users.

2) Largest Component Ratio
We choose the component with the largest number of faces and compute the largest component ratio (L.C.R.) by Eq. 1. Model with larger ratio indicates a higher degree of completion.

$$\text{Largest component ratio} = \frac{\text{Number of largest component faces}}{\text{Number of total faces}} \quad (1)$$

View Source

3) Model Complexity
After the largest component has been retrieved, we obtain the bounding box of it and then divide each axis into 6 equal parts as unit. According to [6], complexity is directly related to entropy based on information theory. Therefore, we introduce [5] to calculate normal

vector entropy instead of intensity entropy. Firstly, we generate N vectors as histogram bins by dividing equidistant azimuth angles on a uniform sphere. In our implementation, the equidistant azimuth angle is set to 10 degrees. We then calculate each normal vector of the model by referencing above vectors, to build a histogram for normal entropy, as illustrated in Fig. 4-(a). Fig. 4-(b) visualizes all normal vectors on each vertex of this model.

Figure 4.
Normal vector entropy calculation using [5]. (a) Sphere are equally divided into N vectors, which labelled as blue arrows, for building histogram (b) the red parts are the normal vectors on vertices

We carry out the partition algorithm based on normalized entropy calculation as given in Eq. 2. Firstly, we segment the component into regions with similar entropy value. The partition can be applied according to directions of three axes. We compute the corresponding normalized entropies through each unit of axes. The normalized entropy is used to determine whether the partition should continue. If the minimum of the sum of the entropies of two partitioned blocks is smaller than a threshold, the segmentation process stops. Otherwise, the partition proceeds according to Eq. 3. In this paper, the threshold is set to 0.1.

$$\text{Entropy} = -\sum_i p_i log_2 p_i$$

$$\text{Maximum entropy}(bins) = -log_2 \frac{1}{bins} \tag{2}$$

$$NE = \frac{\text{Entropy}}{\text{Maximum entropy}}$$

$$E_{1i} = NE(PX(0,i)) + NE(PX(i+1, CX))$$
$$E_{2j} = NE(PY(0,j)) + NE(PY(j+1, CY))$$
$$E_{3k} = NE(PZ(0,k)) + NE(PZ(k+1, CZ))$$
$$P = Argmin(CY \times CZ \times Unit_y \times Unit_z \times E_{1i}, \tag{3}$$
$$CX \times CZ \times Unit_x \times Unit_z \times E_{2j},$$
$$CX \times CY \times Unit_x \times Unit_y \times E_{3k})$$
$$\forall i, \ j, \ k = 0 \sim 63$$

View Source

Once the partition is done, the complexity score can be computed according to Eq. 4. The first complexity measure is named partition complexity, denoted as C_p. If the volumes of the partitioned blocks are diverse, C_p of the model will be higher. The next feature shape complexity, denoted as C_f, is simply the summmation of all normailized entropies. The total number of partitions also reflects the complexity of a model. So we define and employ the feature block-ratio complexity, denoted as C_r. Fig. 5 shows these three complexity measures of two 3D models.

$$C_p = \text{Entropy}(Volume(Block_i))$$
$$C_f = \sum_i NE(Block_i) \tag{4}$$
$$C_r = \frac{R}{N}, \text{ where R } = \text{ Total number of partitions, N } = 64^3$$

Figure 5.
Complexity measures of two 3D models

B. Usage Pattern Related Features

The flowchart we used to analyze usage patterns is presented in Fig. 6. However, fully automatic processing of operation videos is still unattainable as user motion can be too slight to be detected. Instead, we implement a video segmentation system with adjustable threshold, and therefore video can be separated hierarchically. By comparing with the finalized model, labeler can focus on a video segment to judge whether the user is in the trial, implementation, or not an effective operation.
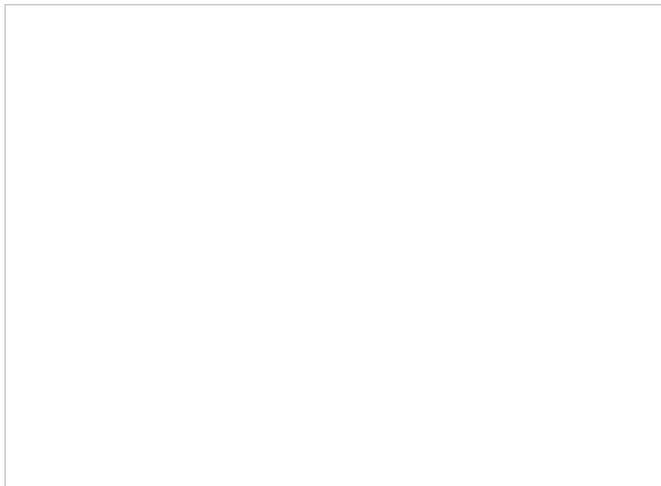
Figure 6.
Workflow of usage pattern analysis

We measure user's operation actions using these indicators from previous work [3]. Effective Operating Period (EOP) consists of an effective effective operating period that excludes daze, idle, or disturbed period, with the threshold defined as 5 seconds. Trial and Error Period (TEP) specifically refers to operations in the period which do not affect the final outcome, while Implementation Period **(IP)** is the duration of a set of operations which are resulted in the creation of a model. Based on these usage pattern metrics, we roughly categorize users into the following four categories,

1. Novice with little interest: Mean of TEP steps and mean of **IP** steps are both large.

2. Novice with caution: Mean of TEP steps is small, but mean of **IP** steps is large.

3. Intermediate: Mean of TEP steps is large, but mean of IP steps is small.

4. Advanced: Mean of TEP steps and mean of IP steps are both small.

C. Data Collection

We collaborate with the Affiliated Experimental Elementary School (AEES) of National Chengchi University to employ our lesson plan for teaching modeling software and 3D printing for one semester. The details are listed in Table I. These experiments involve 15 students including eight girls, seven boys, aged from 11 to 12. We obtain screen recordings (for Tinkercad and 123 Design) and log files (for Qmodel Creator) during all the courses.

Table I Lesson plan incorporated with courses in aees

D. Expert Evaluation

In order to assess the quality of 3D models objectively, we coordinate five researchers of our 3D printing project as expert and use three-point scale to evaluate models. The degree of mutual agreement of expert evaluation of 101 models in the dataset is calculated using Eq. 5, and the reliability is 0.818.

$$DMA_{AB} = \frac{2 \times M_{AB}}{N_A + N_B}$$
$$Reliability = \frac{N \times DMA_{average}}{1 + (N - 1) \times DMA_{average}} \quad (5)$$

View Source

E. Interview with Students

During the interview, we focus the following four issues for qualitative analysis. They are: What modeling software do you prefer to use? (Q1); Are you good at drawing? (Q2); Do you like art? (Q3); Students are asked to draw a picture of a river, a tree and a house with a pen on a paper in five minutes, which are scored into three classes.

SECTION IV.
Results and Discussion

A. Overall Distribution

In the beginning, we analyze models in the dataset and compare the results of model features as well as expert scores for each model shown in Fig. 7.
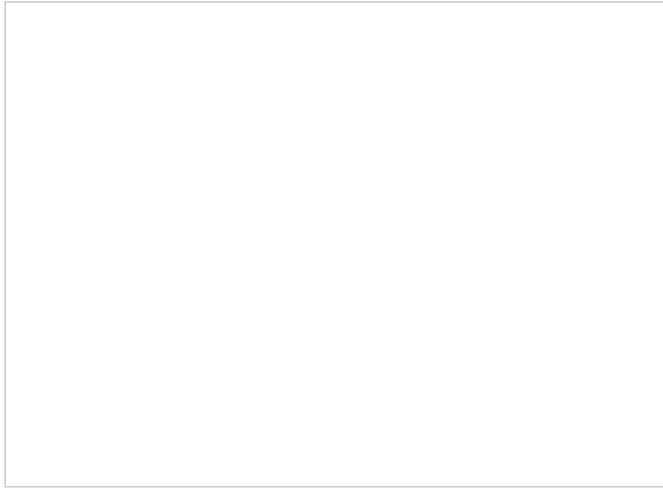
Figure 7.
Boxplot of distribution of model features and scores

The indicators of C_f, C_r, as well as DoD we developed in previous work have a larger interquartile range, which implies that these indicators have higher sensitivity to the differences between models.

We then check those models with extreme values. An example is shown in Fig. 8-(a). This model is not only structurally separated, but also possesses the lowest L.C.R, and the three complexity indicators are low as well. However, an extreme value of one indicator or two does not necessarily lead to a low score. Fig. 8-(b)(c) are examples with high score in extreme value area, whereas Fig. 8-(d)(e) are examples from the low score group. There is still a wide range of distribution of model complexity even for the group with similar scores. In addition, the quantitative values do indeed reflect the appearance of the model. To sum up, a model can be judged as low degree of completeness when the number of connected components is large, and the L.C.R., C_p, C_f, and C_r are small, a rule that can be applied for fast filtering in practice.
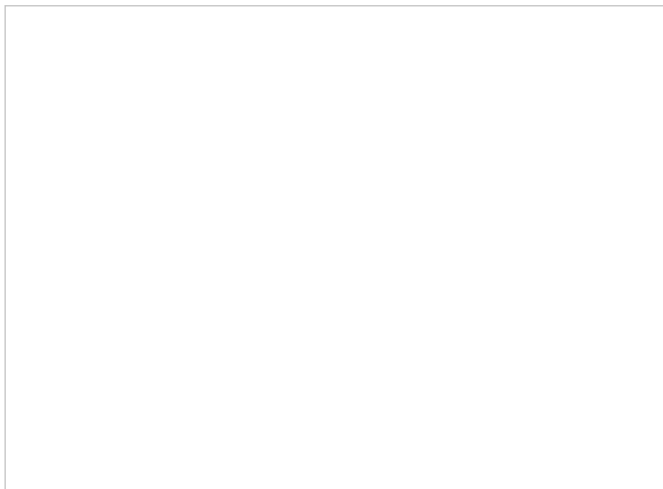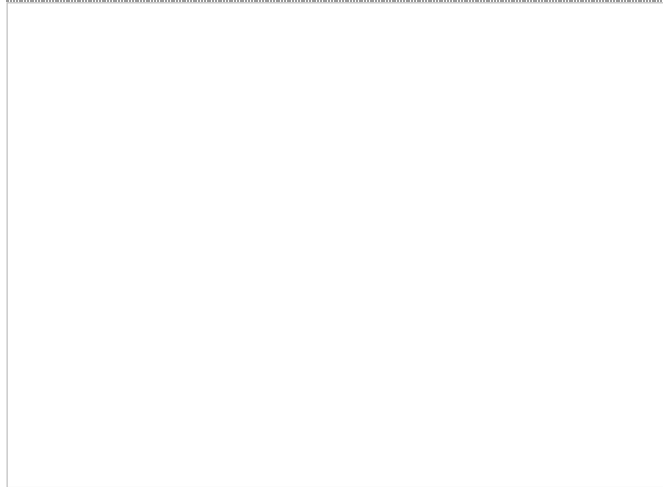
Figure 8.
Models with extreme values

B. Correlation with Score

In order to investigate factors related to the score of the model, we conduct three correlation analysis. Correlation coefficients including each model feature and measure of interviews are computed. Besides, in the second trial, we divide model data into three groups according to the software used, while in the third trial we divide data according to the

groups of the score, with threshold set to 7. The results are summarized in Table II.
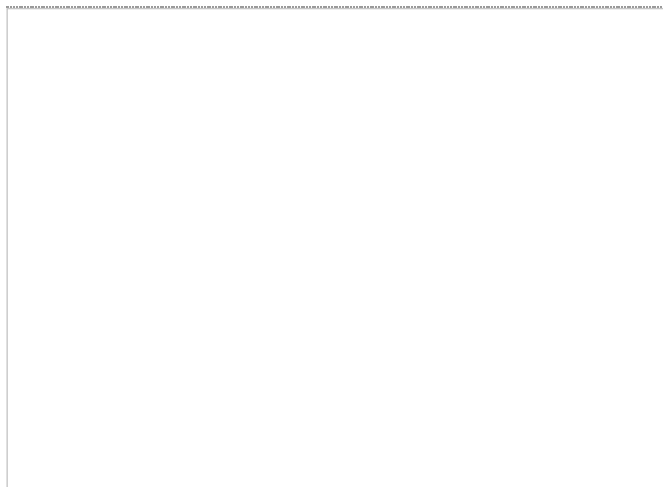
Table II Correlation with scores

The number of connected components is negatively correlated with their scores, and the L.C.R. of a model is positively correlated with the scores. But the correlation in the high-score group is different. We think it is because a low score model usually cannot be printed. If a model has ruled out key factors of failed printing, such as controlling a number of connected components to minimum, the model will possess a higher score.

In addition, the C_f, C_r, and DoD show an inconsistent degree of correlation in Trial 2. It may be related to the functions and properties of modeling software. In qualitative results, Q1 of interview gets weak correlation in all trials. We think that student's preferences and motivation are the keys to learning most of the time.

C. Difference between Groups of Score

We test the two score groups, as listed in Table III. C_r shows a significant difference in T-test. C_f of the two groups has a great gap but it is not significant. Therefore, we think the two features are appropriate to represent model complexity.

Table III Tests for high-score and low-score groups

D. User Categories Comparison

According to the definition of the 4 user categories, we identify an eligible case for each category, retrieve model and usage pattern features by user, as well as average and compare these quantitative values, as presented in Fig. 9.
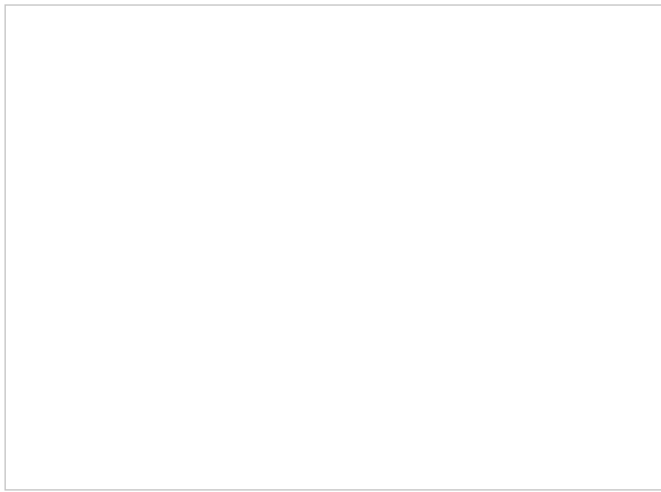
Figure 9.
Comparison among four user categories

Although we use usage patterns to select students, the features of the model reflect its creator's characteristics partially. The advanced student's models are the highest on the C_f item. It is interesting to note that the novice with caution student's works have the highest C_r, while the intermediate student's models have the lowest C_r. The novice with little interest student makes the largest number of connected components works, which contains the smallest ratio of largest component. We believe that teachers can better monitor the learning status of students by inspecting both model features and usage patterns.

SECTION V.
Conclusion

In this paper, three measures to quantize complexity of a 3D model, as well as usage pattern features are proposed to judge students' work and evaluate their performance regardless of the type of software used. The number of connected components and the L.C.R. can be used to automatically detect who needs more instructions. Besides, C_r and C_f are appropriate to represent model complexity. At last, different categories of students' work are compared. The result shows that the model-related features can be related to the properties of its creator partially. We believe that teachers can get a comprehensive understanding of students' learning progress by referencing the proposed features.

Authors	⌄
Figures	⌄
References	⌄
Keywords	⌄
Metrics	⌄