DO STUDENTS KNOW WHAT THEY THINK THEY KNOW? ASSESSING STUDENT CONFIDENCE IN A COMPUTER GRAPHICS COURSE

Timothy Urness
Department of Mathematics and Computer Science
Drake University
Des Moines, IA 50311
515 271-2118
timothy.urness@drake.edu

ABSTRACT

This paper describes an experiment to determine the extent students' confidence of their knowledge matches their ability. In other words, do students know what they *think* they know? This is done by comparing pre-course and post-course surveys about the learning objectives of the course to final exam answers. The primary goal is to determine if there are components of the course in which students have built up an overconfidence – they think they know the material more than they actually do or can demonstrate; or, conversely, an under-confidence – they are intimidated by the concept, but can actually perform the tasks to a greater level than their expectations. A secondary goal is to assess the effectiveness of the course in terms of the ability of the students to perform tasks related to the learning objectives and their perceptions. The experiment is done in the context of an introductory course on computer graphics. We found that the aggregate student confidence levels in the post-course exam were consistent with the corresponding final exam scores. However, students tended to have a slight overconfidence in material that was covered in the beginning of the semester.

INTRODUCTION

Metamemory or metacognition is the introspective process of reflecting on one's memory abilities. This self-awareness of memory allows students to assess their own learning. In essence, it allows students to acknowledge what they know and don't know. The ability to accurately determine the content that has not been mastered is a critical skill in learning.

Previous studies have shown that when asked about confidence levels of exam answers, students not familiar with a topic tend to perform poorly at discriminating between what they know and what they do not know. However, students that are familiar with the topic (such as after taking the course) have a higher percentage of correct answers, higher confidence, and a better discrimination of their confidence [1]. Furthermore, students with a high degree of confidence in their answers typically have higher exam scores [6]. Another study concluded that students with higher metacognitive abilities have better academic performances [7]. When asked after an exam to predict how well they did, students with higher grades have developed skills that help them more accurately evaluate their performances. The results of these studies indicate that students would be well served to sharpen their metacognitive skills – including the ability to accurately detect and depict what they know and what they don't know.

Computer graphics is often a very enticing course for students that have been already exposed to the glamorous manifestations of computer graphics such video games, special effects, and animated movies. Students enter the course with high expectations, and may not be prepared for the mathematical nature of the subject [5]. At small-to-medium sized universities, where only one course in computer graphics is offered, there is a challenge to balance the foundational computer graphics theory with technical details while supporting sound programming and software engineering practices [2, 4]. Prior to the course offering, we established learning objectives for the introductory computer graphics course in conjunction with the 2013 ACM-IEEE curriculum guidelines [3]. Our ultimate goal in this study is to determine if there are components of the course in which student's perceptions of their knowledge don't match their abilities. An additional goal is to assess the effectiveness of the course in terms of the ability of the students to perform tasks related to the learning objectives and to compare this to their perceptions of their knowledge.

METHODOLOGY

This paper describes an Introduction to Computer Graphics course with 32 students at Drake University, a small liberal arts university. The course consisted of 32 junior or senior computer science majors. Six of the 32 students were female (18.75%). In order to assess the students' knowledge level with respect to the course objectives, the students answered a survey that asked them to rate their knowledge level before the course began. Specifically, the survey asked students to rate on a scale from zero to five how well they could answer the questions correctly. Zero corresponds to no confidence. Five corresponds to very confident. The same survey was administered on the last day of the course. To determine the extent that students actually learned the material they reported to be confident in answering, graded final exam answers are compared to the survey responses.

The final exam included a wide range of questions with seven of the eighteen survey questions directly relating to items on the exam. These survey questions did not make up the entirety of the exam. We did this to keep the survey at a manageable length, as well as to ensure that the survey was not identical to the final exam, which could compromise the validity of the final exam as a measure of the students' knowledge of the course.

Survey Questions

Items that appeared on the final exam are denoted with an *.

1. Explain how the limits of human perception affect choices about the digital representation of analog signals.

2*. Describe an affine transformation and its representation as a matrix

3. What is an API? Describe how you would use an API in constructing a program.

4*. Describe the differences between lossy and lossless image compression techniques, for example as reflected in common graphics image file formats such as JPG, PNG, and GIF.

5*. Describe color models and their use in graphics display devices.

6. Describe the tradeoffs between storing information vs. storing enough information to reproduce the information, as in the difference between vector and raster rendering.
7. Describe how double buffering can remove flicker from animation.
8. Describe a process in which 3D polygons are rendered on a 2D screen.
9*. Describe the texture mapping process and the advantages of using texture mapping when creating computer-generated images.
10. Describe how programming in the large differs from individual efforts with respect to understanding a large code base, code reading, understanding builds, and understanding context of changes.
11*. Compare and contrast different shading algorithms for 3D computer graphics.
12. Identify and justify necessary roles in a software development team and discuss common behaviors that contribute to the effective functioning of a team in a software development setting.
13. Describe a method to estimate software development effort and explain how to best ensure its reliability.
14. Describe one or more design patterns that could be applicable to the design of a simple software system.
15*. Propose a suitable visualization design for a particular combination of data characteristics and application tasks.
16. Describe different visualization techniques as they can be applied to a variety of applications including representations of scientific, medical, and mathematical data.
17. Define a user-centered design process that explicitly takes account of the fact that the user is not like the developer or their acquaintances.
18*. What are homogeneous coordinates and why are they used in computer graphics?

RESULTS

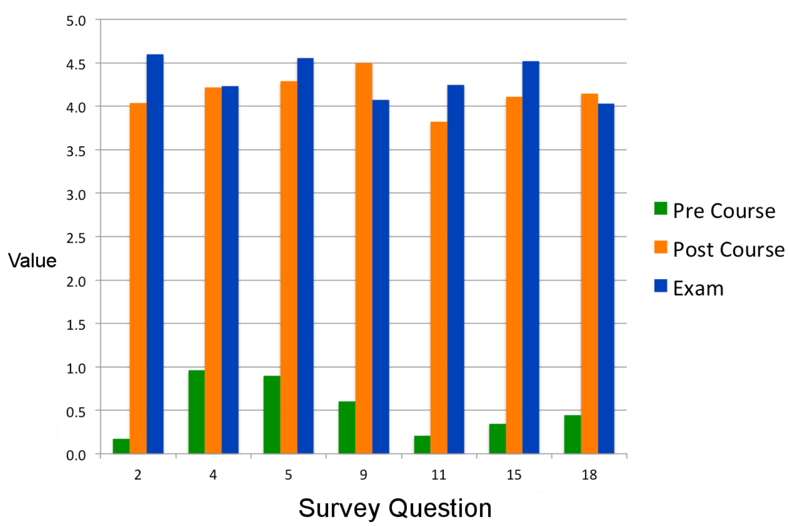

Figure 1: Values of the pre-course survey, post-course survey, and exam questions of the seven items on the surveys that appeared on the final exam.

Question Number	Pre-Course Average	Post-Course Average	Change
1	0.55	2.92	2.37
2	0.17	4.04	3.87
3	1.36	3.67	2.31
4	0.97	4.21	3.24
5	0.90	4.29	3.39
6	0.79	3.57	2.78
7	0.62	4.53	3.91
8	0.41	4.04	3.63
9	0.60	4.50	3.90
10	0.97	2.50	1.53
11	0.2	3.82	3.62
12	1.10	2.75	1.65
13	0.78	1.68	0.90
14	0.69	1.96	1.27
15	0.34	4.11	3.77
16	0.66	4.29	3.63
17	0.76	2.46	1.70
18	0.44	4.14	3.70

Table 1: Results from the Pre-course and Post-Course Surveys. Students were asked to rate on a scale from 0 to 5 on how confident they felt they could answer the questions correctly. There were 32 students in the course.

To compare the final exam questions to the survey results, we first calculated the average score for each of the questions. This gives a value between 0.00 and 1.00 (0% to 100%). The average score is then multiplied by 5 to give a range in [0,5], comparable to the range of the survey responses. These values are then compared to the post-course survey values in table 2.

Survey Question	Post-course Survey Mean Value	Normalized Mean Exam Score in Range [0,5]
2	4.04	4.60
4	4.21	4.23
5	4.29	4.56
9	4.50	4.07
11	3.82	4.25
15	4.11	4.52
18	4.14	4.03

Table 2: Survey questions and Exam Scores.

ANALYSIS

The comparison of the pre-course to post-course survey data indicates the students feel as if they learned the material related to these learning objectives. Comparison of the pre-survey and post-survey questions were statistically significant with a p-value < 0.001. Furthermore, the aggregate student confidence levels in the post-course exam were consistent with the corresponding final exam scores.

In general, students had a realistic confidence they could perform the learning objectives. Looking closer into the data, however, indicates that students were slightly over-confident in material covered at the beginning of the semester (question #9: texture mapping and question #18:homogeneous coordinates). This indicates that we weren't as effective in teaching these concepts as we had thought and/or students' working knowledge had atrophied while their confidence remained higher than their ability levels at the final exam. The most significant question in which students had *under-confidence* was question #2 (define an affine transformation). We suspect that the under-confidence result arises from the fact that the exam question was basic and straightforward. As students reviewed for the final exam, we believe this concept was easily covered and the simple nature of the question made the average answer higher than predicted during the last day of class.

Post-Semester Survey Analysis

The survey was made in preparation for the first day of the course. The intention was to teach the content of computer graphics in a fashion that emphasized elements of software engineering. However, due to a lack of time and an unexpected, time-consuming obligation during the semester, we resorted to teaching the computer graphics content in a more traditional fashion, with less emphasis on software engineering. As a result, several of the questions on the survey regarding software engineering were not fully addressed in the course (questions 10, 12, 13, 14, and 17). This provides us with an opportunity to assess the credibility of the survey responses: if the software engineering survey results saw comparable improvement to the computer graphics questions, we would be suspicious that the survey responses were flawed. The average value for these five software engineering questions on the post-course survey is 2.27 (out of 5). The average value for the graphics-related questions is 4.01.

The average initial response of the pre-semester survey for the five software engineering questions was 0.86, compared to a post-survey average of 2.27. We acknowledge that it is possible that students are inflating their confidence level because they felt they *should* have learned something in a post-semester survey for every question asked, even if the content was not part of the course itself. Another explanation is that over 25% of the students were also enrolled in a traditional software engineering course during the same semester. These skills have been learned in places other than in the computer graphics classroom.

CONCLUSION

It is important for students to be introspective about their abilities. Studies have shown that students that can accurately depict what they know and what they don't know have better academic performances. We set out to determine to what extent students know what they think they know by comparing pre-course and post-course surveys about the learning objectives of the course to final exam answers. In this case, students demonstrated that their expectations were very accurate, as the post-course survey results correspond to the exam results.

A secondary goal is to assess the effectiveness of the course in terms of the ability of the students to perform tasks related to the learning objectives and their perceptions. This also appears to be successful, with a few caveats. In general, students have a realistic confidence they could perform the learning objectives. Close analysis of the data indicates that students were slightly over-confident in material covered at the beginning of the semester (texture mapping and homogeneous coordinates). This implies that we weren't as effective in teaching these concepts as we had thought and/or students' working knowledge had atrophied while their confidence remained higher than their ability levels at the final exam.

The results of this study will impact future offerings of the course. Areas in which students were over-confident (texture mapping and homogeneous coordinates) will be covered in more detail, with more exercises and examples in class. We will also emphasize software engineering principles through computer graphics, including final exam questions that will hold students accountable for learning software engineering practices alongside applicable to computer graphics.

REFERENCES

[1]Caleon, I. S., Subramaniam, R. Do Students Know What They Know and What They Don't Know? Using a Four-Tier Diagnostic Test to Assess the Nature of Students' Alternative Conceptions, *Research In Science Education*, *40*(3), 313-337, 2010.

[2] Cunningham, S., Powers of 10: the case for changing the first course in computer graphics. In *Proceedings of the thirty-first SIGCSE technical symposium on Computer science education*, 46-49, 2000.

[3]Joint Task Force on Computing Curricula, Association for Computing Machinery (ACM) and IEEE Computer Society. *Computer Science Curricula 2013: Curriculum Guidelines for Undergraduate Degree Programs in Computer Science*, 2013.

[4]Lowther, J. L., Shene, C.-K., Rendering + modeling + animation + postprocessing = computer graphics, *Journal of Computing Sciences in Colleges,* (1), 20-28, 2000.

[5] Shesh, A., Toward a Singleton Undergraduate Computer Graphics Course in Small and Medium-sized Colleges. *ACM Transactions on Computing Education* 13(4), 17:1-17:21, 2013.

[6]Sinkavich, F. J., Performance and metamemory: Do students know what they don't know? *Journal Of Instructional Psychology*, *22*(1), 77-87,1995.

[7]Vadhan, V., Stander, P., Metacognitive ability and test performance among college students. *Journal Of Psychology*, *128*(3), 307-309, 1994.