EFFECTIVENESS OF MULTIMEDIA IN LEARNING & TEACHING DATA STRUCTURES ONLINE

Sahalu JUNAIDU
Information & Computer Science Department
King Fahd University of Petroleum & Minerals
Dhahran, KINGDOM OF SAUDI ARABIA

ABSTRACT

Online electronic education is now being widely accepted as a major viable component of higher education. This is fuelled by the emergence of worldwide information and computer communications technologies. However, online education is not being adopted in science and engineering subjects as widely as in other fields because of the idiosyncrasies of some science and engineering-based courses.

For online engineering education to be broadly accepted and utilized, the quality of online courses must, amongst other things, be comparable to or better than those of traditional face-to-face classroom education. This paper explores and reports on the importance of creating multimedia-rich course content and the important role that animations can play in creating a successful online learning experience.

Results of our study on an online data structures course over five years offerings show that students consistently perform much better in questions requiring application of material taught in carefully animated algorithms. These results should carry over to other educational environments.

Keywords: Multimedia; e-Learning; data structures

INTRODUCTION

The acceptance of online electronic education in colleges, universities and corporate organizations is now pervasive. This is made possible largely by the emergence and rapid development in worldwide information and computer communications technologies. The initial skepticism with which online electronic education was greeted is now waning away. We are now witnessing not only the offering of a course or two online in traditional universities but the establishment of full-fledged degree programs online and online universities (Phoenix, 2006; Cardean, 2006; Colorado, 2006).

Even with these developments, online electronic courses in science and engineering are not as widespread as courses in other disciplines in higher education. The reasons for this are that, science and engineering education has, traditionally, been content-centered, design-oriented, and is permeated by the development of problem-solving skills (Bourne, 2005). It is further argued that some of the special needs of undergraduate science and engineering education have not been well served by methods of online education.

Specifically, laboratories are a mainstay of engineering education, as are mathematical foundations and design tools. Laboratories (Grose, 2003; Peterson, 2002) are notably difficult to provide online because of the traditional desire for the direct operation of instruments. Similarly, course materials that require significant use of mathematics have not been as easy to implement as topics that require only text-based discussion (Bourne, 2005). For online science and engineering education to be broadly accepted and utilized;

- **the quality of online courses must be comparable to or better than the traditional classroom,**
- **courses should be available when needed and accessible from anywhere by any number of learners, and**
- **topics across the broad spectrum of engineering disciplines should be available. One way of meeting the first requirement is through the use of multimedia in creating interactive courseware that gives learner control leading to potentially better learning experiences.**

The potential of multimedia in education does have a theoretical foundation. Bagui (1998) and Daniels (1995) summarized the theory of multi-channel communication in support of the potential for multimedia. According to this theory, humans have several channels by which data is communicated. If information is presented via two or more of these channels, there will be additional reinforcement and, consequently, greater retention, thereby improving learning (Ellis, 2004). Further support for the potential benefits of multimedia is offered by research in learning styles. McCarthy (1997) explored learning styles and identified four distinct approaches to learning: the *feeler*, the *analyzer*, the *doer*, and the *creator*. A multimedia approach presents the potential to address these different approaches to learning, as was suggested by the research of (Riding and Grimley, 1999).

This paper explores the effectiveness of multimedia in helping students learn in an online undergraduate Data Structures course at our university. By the time of this study, the course has been offered completely online for four years, except for the laboratory component of the course which was instructor lead. I make use of Ellis's model for testing the effectiveness of multimedia in this study. Ellis's model is discussed in the next section. The rest of the paper details the purpose of the study, the algorithms selected for the study, the student population, data collection and analysis, and conclusions of the study.

ELLIS'S MODEL FOR MULTIMEDIA EFFECTIVENESS

Our work follows the model of (Ellis, 2004) for establishing precisely the value of multimedia in enhancing learning. According to this model, any study of the effectiveness of multimedia as a tool to enhance learning must specify:

- **learning in a manner that is consistent with accepted learning theory**
- **the student population under consideration**
- **the subject matter being studied**
- **which media elements are being studied, at what level of interactivity, and toward what end**

A typical Data Structures course like ours normally covers level 2 (comprehension), level 3 (application), level 4 (analysis) and level 5 (synthesis) competency levels of the Bloom's taxonomy (Bloom, 1956). Level 2 competency is covered in the requirement for learners to be able to describe common applications for each data structures, describe how the data structures are allocated and used in memory, and explain the use of big O to describe the amount of work done by an algorithm.

Level 3 competency is covered in the requirement for learners to be able to perform tasks illustrated in our animations. They should also be able to write programs that use arrays, strings, linked lists, stacks, queues, hash Tables, trees, and graphs. Level 4 competency is covered in the requirement for learners to be able to compare alternative implementations of data structures with respect to performance and also learners' ability to compare and contrast the costs and benefits of dynamic and static data structures implementations. Level 5 competency is covered in the requirement for learners to be able to choose the appropriate data structure for modeling a given problem. For the purpose of the study in this paper, the aspect of learning under consideration is learners' ability to quickly acquire information and to apply the newly acquired information to solve problems. We measure this learning quantitatively through performance in examination and qualitatively through focus interviews and learners' written responses from our survey.

Data structures is an introductory foundation course and is typically taken early in the curriculum to give students sufficient grounding required in intermediate and higher level courses. The student population for this study consists of BS students of Computer Science and Computer Engineering programs at our university. Our data structures course was designed around and meets fully the CC2001 curricular requirements (ACM, 2001).Multimedia elements in our courseware include images, linear and non-linear animations and Java applets, voice narration, an arrow indicator highlighting current focus point, and well-placed interactive self-check exercises. Specifically, our study focuses on the use of interactive animations to raise learners' interest level, enhance understanding and increase memorability. We use Macromedia's applications suite to animate, at the lowest granularity level, complex algorithms that are otherwise cumbersome to teach in the face-to-face style using a chalkboard.

THE STUDY

Rationale of Animations Use

Learning theorists state that to reach an objective or to acquire a skill, the learner must be actively involved through practice to cognitively incorporate it into long-term memory. The interaction or "doing the objective" helps the learner reach the objective and recall the information, skill, or behavior that was learned (Dick & Carey, 1990). Furthermore, Wolfgram (1994) states, "People only remember 15 percent of what they hear and 25 percent of what they see, but they remember 60 percent of what they interact with". The aim of our study is to investigate the effectiveness of a multimedia-rich Data Structures courseware in enhancing students learning. The multimedia elements in our courseware include images, linear and non-linear animations and Java applets, voice narration, an arrow indicator highlighting current focus point, and well-placed interactive self-check exercises.

Several dozen studies indicate that computer-based multimedia can improve learning and retention of material presented during a class session or individual study period, as compared to "traditional" lectures or study materials that do not use multimedia (Bagui, 1998; Fletcher, 2003; Kozma, 1991; Mayer, 2001). Furthermore, a number of studies have suggested that student satisfaction and motivation is higher in courses that use multimedia materials (Astleitner & Wiesner, 2004; Yarbrough, 2001). Multimedia draws upon more than one of the five human senses, utilizing the two fundamental senses vital for information reception – sight and sound. Due to motion and sound, it can also spark attention, interest and motivation in the process (Mohler, 2001). Rich multimedia combined with planned interactions that give control to the user, as are salient in our courseware, are expected to yield good pedagogical dividends. We set out to study and quantify these dividends in this research work.
Before proceeding, we first outline the algorithms that we focus on in our studies and the aspect of learning we intend to measure.

Algorithms Studied

The objective of a data structures course, in general, is to enhance students' algorithm design, analysis, application, and implementation skills through the coverage of linear and non-linear data structures. These skills are fundamental to a successful career and advanced studies in computer-related fields. The study in this paper focuses on four aspects: trees, graphs, hashing and applications of data structures in data compression and memory management. The competency targeted in our study is students' ability to demonstrate learning by *applying* the concepts learned as exhibited in their ability to *perform* specific tasks.

In addition to recognizing the suitability of B-trees as storage structures, in contrast to data structures like the AVL trees, students' should be able to perform the basic operations of insertion and deletion in various kinds of trees including B-trees. A major learning outcome in teaching graphs is that students should be able to apply knowledge of graph traversals to solve a wide-range of graph algorithms. While understanding the trade-offs between hashing and retrieval operations in other data structures, the students' should be able to apply basic number-theoretic knowledge in building and implementing hash Tables. In the Applications part of our course, we use animations as pedagogic tools to teach data compression, memory management and some recursive algorithms. We measure learning in this paper based on students' ability to effectively demonstrate Bloom's Level 3 competency, application of data structures knowledge to solve problems in this case.

According to Michigan (2006), the instruction needed to effectively deliver the knowledge concerned with ability to perform tasks requires that the task be described to the learner, in terms of what the end result will be, the steps involved to produce the results, and the order in which the steps must be performed. Traditionally, data structures and algorithms like the ones highlighted above are typically taught by the instructor having to illustrate them with drawings on the chalkboard.

Some of these algorithms can be cumbersome to teach this way even for the seasoned instructor. Furthermore, the instructor's productivity within a single class time will be minimized and that such drawings will have to be redone by the instructor in other classes. To overcome these limitations and to increase the potential for learn ability of the course material, we developed and animated carefully selected examples to teach data structures and algorithms.

The next section outlines the design and implementation of our animations. The animation development process, the tools used and the estimated development time for each animation are also provided. Subsequent sections report the instructional value of our animations based on students' performances in assignments, examinations and also on qualitative interviews.

DEVELOPING THE ANIMATIONS

An important issue in designing and developing rich-media animations is the selection of an appropriate authoring system. Based on our experience, an authoring system must be characterized by the following elements for it to be effective:

- **Usability**
- **Foreign file import**
- **Granularity**
- **Streaming capability**

An authoring system should be user-friendly and should support a rich collection of functions that give as much control to the user as possible. A good authoring system should be compatible with and support importing images and graphics created using other applications. It should enable the user to develop animations at the lowest level of granularity. It should support a streaming capability that enables a developed animation to be compiled and broken down into smaller units for easier deployment over the Internet/intranet.

Our investigations in (Junaidu, 2004), found that Macromedia's applications bundle provides a convenient and effective environment for animations development. Our animations are developed using the applications in this bundle. Selection and design of animation elements in our course were done collaboratively among the course development team during the detailed design phase of the course (Junaidu, 2004).Our animations on trees carefully and systematically exemplified all traversals, insertion and deletion cases. In the case of B-trees the associated splitting and merging of nodes, where applicable, are shown clearly. The animations proceed gently with indicators showing nodes underflow and nodes overflow before proceeding to carry out the merging and splitting processes, respectively. The animation took about 30 hours to design and develop. This includes the time to develop a voice narration that explains the animations and to synchronize the animation movements with the voice narration. The animation runs to completion within about 15 minutes without user interaction.

It can take more time to preview if the user decides to pause at intervals or less time to preview if the user disables the voice narration as is typical during revision. For graph algorithms, we animated the Dijkstra's single-source shortest paths algorithm, the Prim's and Kruskal's minimum spanning tree algorithms and others. The Dijkstra's algorithm we animated, in particular, carries out the computations in phases while maintaining a tabular record of the current shortest distances from the origin vertex, the vertices inspected so far, the current vertex to be selected etc. This algorithm is highly cumbersome to teach by drawing these elements on the board. This animation took about 40 hours to design and implement. Students can conveniently preview this animation within 11 minutes with high potential for raising interest level and increased memorability.

Our animations on hashing carefully exemplify and graphically illustrate the concept of collision and collision resolution in open addressing showing how hash Table locations are probed systematically using the standard collision resolution schemes. Cichelli's algorithm for building minimal perfect hash Tables, by performing exhaustive searches, is also carefully animated showing and indicating, step-by-step, the concept of backtracking that is otherwise difficult to grasp.In the applications part of the course, we developed animations illustrating data compression and decompression using Huffman coding, data compression and decompression using the LZ77, LZ78 and LZW of the Lempel-Ziv family of encoding algorithms. There are also animations on memory management interactively showing the *acquire* and *release* operations of the memory manager while highlighting internal and external fragmentation of memory storage. As in the other algorithms there is a significant time investment in the design and development of the animations while students' preview times for the different algorithms range between 4 to 8 minutes.

Sample Population

In order to contextualize results of our study and in following (Ellis, 2004), it is necessary to specify the students' population under consideration in our study. Our students' population consists of young adult undergraduate students of BS computer science and computer engineering. The course is core for all the students so the relevance of the course material is well recognized by the students. However, the relevance of the course being online has not been as clear in the minds of the students over the semesters. All students were required to take the course online with no face-to-face option.

Furthermore, a greater majority of the students are not as matured as online learning requires. That is, the students are largely instructor dependent who saw little or no need to have full control over the educational process. The students are generally from above-average families with many of them having their jobs already secured. We should also mention that the Data Structures course was the only course taken online by the students in our on-campus university.

DATA COLLECTION and ANALYSIS

Data is collected from major examinations, final examinations and homework assignments for all students' who took the course from the 2001/2002 academic year to the 2005/2006 academic year. There were two major examinations and a final examination in each year. There were at least five assignments in each year. The first major examination is mostly programming-based covering material on object-oriented design patterns, linear data structures, analysis of linear data structures and recursion. The second major examination is based on tree and graph algorithms with extensive algorithmic animations. The final examination is comprehensive covering material on hashing and applications of data structures in addition to the material in the major examinations.The data collected is used to compare the relative performance, of the same students' group in each semester, in different portions of the same online course. The course portions are divided into two main parts; those with extensive algorithmic animations and those with mere voice narration and simple animations explaining concepts or program segments. Assessment of the major learning outcome in this study, to apply learned concepts to solve problems, was conducted using students' examinations and assignments results. Performances were compared at various levels in each semester.

Comparisons were made first at the level of the examinations (among the majors and the final) and at the level of the assignments. At the second level, performance comparisons were made at topics level and, at the third level, performance comparisons were made on the kind of question asked on the individual topics. On graph traversals, for example, students could be asked to analyze a given algorithm, to implement a certain algorithm or to apply a specified algorithm on a given graph.

Our questions typically cover all these aspects and students' performances vary significantly in these.

RESULTS

Standard statistical analysis procedures were used to compare the performance of the students on the course material that was intensive in animating algorithms and the material that did not involve much animation. Table 1 shows the descriptive statistics for the cumulative mean performances for about 700 students for the five year period under study. Results for the individual students' groups are similar except for the occasional showing of outliers, among performances in assignments or examinations, over the semesters.

Table: 1 shows results for five assignments and three examinations. Assessment material for Assignments 1 and 2, and Major 1 were unclassified because there were no significant algorithmic animations there. These were included to provide for top-level comparisons and for completeness.

Table: 1
Assessment Results Summary

Assessment element	Material	Mean	Classified Mean	Std Deviation
Assignment 1	Unclassified	65.64	-	32.38
Assignment 2	Unclassified	59.29	-	28.08
Assignment 3	Animated algorithms	73.59	88	10.33
	Other material		58	22.27
Assignment 4	Animated algorithms	75.4	91	5.97
	Other material		60	13.53
Assignment 5	Animated algorithms	74.5	90	6.37
	Other material		61	19.13
Major Exam 1	Unclassified	60.71	-	15.97
Major Exam 2	Animated algorithms	68.98	79.8	9.19
	Other material		57.6	17.58
Final Exam	Animated algorithms	65.06	75	7.06
	Other material		55	11.81

Material for assignment 1 is purely programming-based on object-oriented design patterns where students implement visitors and enumerators over containers.

Similarly, assignment 2 is programming-based covering material on linked lists, stacks, queues with some analytic questions on algorithm analysis. Major examination 1 is based on assignments 1 and 2 material. Animations on this material include illustrations on linked list operations in addition to an arrow indicator and textbox highlighter that compliment the voice explanations. Students' mean performances in these assignments are comparatively lower than those in assignments 3, 4 and 5 even though the programming material is supported by extensive practice in the instructor-led laboratory component of the course.Assignment 3 covers material on trees involving many elaborate animations like those illustrating tree traversals, AVL rotations and B-Tree operations.

Except for simple programming questions, assignment 4 typically requires students to perform tasks that involve 'executing' the Dijkstra's, Prim's, Kruskal's and other graph algorithms that have been carefully animated. Assignment 5 is on building hash Tables and data compression, algorithms that have, again, substantial investment on animations. Major examination 2 is based on material on trees and graphs while the final examination is comprehensive covering all material in the course with more emphasis on hashing, data compression and memory management.

Results of major examination 2 usually average higher than that of the final examination which, in turn, averages higher than that of major examination 1. The only plausible explanation for this is the effectiveness of the animations as the students' themselves attest.Based on the summarized data in Table: 1, we performed a t-test for the comparison of means of the students' performances in questions with elaborate animations and in those without or with very light animations. Paired samples t tests indicated that the difference in performance between the categorized questions in the course material is significant ($p<0.05$, $t = -13.828$).

In our study in (Junaidu, 2004b) on the usability and appropriateness of our animations in three online courses, there was an overwhelming response in support of the utility of the animations, especially by the students of the data structures course. Face-to-face discussion with selected students, mostly who have already completed the course, confirmed those results. Common observations were that the course material was too much, that the algorithm course should be canceled (since most of it has already been covered in the data structures course, according to some students), that the voice explanations of some of the units should be re-recorded to remove background noise.

Apart from these issues, the utility of the animations recorded an excellent rating by the students (Junaidu, 2004b, Junaidu, 2004a).

SUMMARY and CONCLUSIONS

Online science and engineering education does not enjoy as wide acceptability as other academic fields. This paper reviewed arguments proposing solutions to wide acceptability of online science and engineering education including the need to develop online courseware with comparable or higher quality than those of traditional classroom.

After a brief review of the theoretical foundation of the potential of multimedia for enhancing instruction, this paper conducted a study on the effectiveness of multimedia in learning and teaching an undergraduate data structures course.

Following Ellis's model, we conducted a study on how effective media-rich animations are in helping students learn and apply various algorithms in an online data structures course. Our media elements include text, graphics, voice narrations, interactive self-check quizzes requiring significant learners' active participation in the learning process. Our students' population in the study was a group of junior undergraduate students and the assessment instruments in the study were students' performance in assignments and examinations.

Results of our study over five years of offering the course online show that students consistently perform much better in questions related to the demonstration of understanding and the application of algorithms that have been carefully animated. Students' comments have often been that they found the course much easier after the first major examination and as they study the much more heavily animated components of the course thereafter.

In addition to improve learn ability of the course material, as is evidenced by students' performance, there are also instructional and curricular benefits afforded by our media-rich animations. Instructors in our university and some of its affiliated colleges now find it easier to teach using our courseware, thanks to the animations.

Students' also appraise the repeaTable nature of the animations; as they preview them over and over again until they master the algorithms being illustrated.

A benefit to the curriculum is the standardization of the course material, regardless of the instructor teaching the course and regardless of the offering mode (face-to-face or online).We should point out that while multimedia can be used to achieve the benefits highlighted above, it must not be used to dazzle the learner and should only be used in the presentation of concepts where practical, applicable, and valid.

It is also important to keep in mind that a poorly developed and/or executed use of multimedia can do more harm than good (Ludwig, Daniel, Froman and Mathie, 2004). Superfluous use of multimedia may induce disorientation and *cognitive overload* that could interfere with learning rather than enhance learning (Mayer, Heiser, & Lonn, 2001).

Acknowledgement

This paper would not have been possible without the support of our university. I extend my gratitude to my colleagues with whom we designed, developed and facilitated the delivery of this course online.

I acknowledge the support of my colleagues who read drafts of this document and provided constructive suggestions for improvement. Lastly, I'm deeply indebted ICS 202 students over the years especially those with whom I conducted several friendly chat sessions regarding pedagogic, management and students issues related to the success of the online course.

BIODATA and CONTACT ADDRESSES of AUTHOUR

Sahalu JUNAIDU is an Assistant Professor of Computer Science at King Fahd University of Petroleum & Minerals, Saudi Arabia. He received his MSc from Queen Mary and Westfield College, University of London (1992) and his Ph.D. from St. Andrews University, Scotland (1998). His areas of interest include parallel computing, programming languages, Computer Science Education and e-Learning. He has been actively involved in the development, delivery and policy development for online course material for the past eight years.

Sahalu JUNAIDU
Information & Computer Science Department
King Fahd University of Petroleum & Minerals
P O Box 1136, Dhahran 31261
Kingdom of Saudi Arabia
Email: sahl@kfupm.edu.sa

REFERENCES

ACM (2001). *ACM/IEEE-CS Computing Curricula 2001*. Electronic version available at http://www.acm.org/sigcse/cc2001

Astleitner, H., & Wiesner, C. (2004). An integrated model of multimedia learning and motivation. *Journal of Educational Multimedia and Hypermedia, 13,* 3-21.

Bagui, S. (1998). Reasons for increased learning using multimedia. *Journal of Educational Multimedia and Hypermedia, 7,* 3-18.

Bloom, B.S., (Ed.) (1956). *Taxonomy of educational objectives: the classification of educational goals: Handbook I*, cognitive domain. Longmans, New York, 1956.

Bourne, J., Harris, D.A, & Mayadas, F. (2005). Online Engineering Education: Learning Anywhere, Anytime; *Journal of Engineering Education* Vol. 94, No. 1; January 2005.

Cardean (2006). Carden University. http://www.cardean-mba-online.com/, Accessed last on May 8, 2006.

Colorado (2006). Colorado Technical University Online. http://www.ctudegreeonline.com, Accessed last on May 8, 2006.

Daniels, L. (1995). Audio vision: Audio-visual interaction in desktop multimedia. In Beauchamp, D.G., Braden, R.A., and Griffin, R.E. (Eds.). *Imagery and visual literacy: Selected readings from the annual conference of the international visual literacy association* (26th, Tempe, Arizona, October 12-16, 1994). (ERIC Document No. ED380056, pp. 57-63).

Dick, W. & Carey, L. (1990). *The systematic design of instruction*. U.S.: Harper Collins Publishers.

Ellis, T. (2004). Animating to Build Higher Cognitive Understanding: A Model for Studying Multimedia Effectiveness in Education, *Journal of Engineering Education*, Vol. 93, No. 1, 2004, pp. 59–64.

Fletcher, J. D. (2003). Evidence for learning from technology-assisted instruction. In H. O'Neil Jr., F., and Perez, R. S. (Eds.), *Technology applications in education: A learning view* (pp. 79-99). Mahwah, NJ: Lawrence Erlbaum Associates.

Grose, T. K. (2003). Can Distance Education be Unlocked? *Prism*, Vol. 12, No. 8, April 19–23, 2003. Available from http://www.prismmagazine.org/april03/unlocked.cfm

Junaidu, S. et al. (2004). *Technology Based Education in KFUPM*, Final Project Report, Deanship of Academic Development, King Fahd University of Petroleum & Minerals, Dhahran 31261, Saudi Arabia.

Junaidu, S. & Al-Ghamdi, J. (2004a). *Comparative Analysis of F2F and Online Course Offerings: KFUPM Experience*. International Journal of Instructional Technology and Distance Learning (IJITDL), April 2004.

Junaidu, S. (2004b). *Use of Internet for Online Course Delivery: A Case study*, International Conference of Information & Computer Science, November 28-30, 2004, King Fahd University of Petroleum & Minerals, Dhahran Saudi Arabia.

Kozma, R. (1991). Learning with media. *Review of Educational Research, 61,* 179-211.

Ludwig, T.E., Daniel, D.B., Froman, R. & Mathie, V.A. (2004). *Using Multimedia In Classroom Presentations: Best Principles*, Pedagogic Innovations Task Force, Society for the Teaching of Phsychology.

Mayer, R. E. (2001). *Multimedia learning.* New York: Cambridge University Press.

Mayer, R. E. & Moreno, R. (2002). Animation as an aid to multimedia learning. *Educational Psychology Review, 14,* 87-99.

Mayer, R. E., Heiser, J. & Lonn, S. (2001). Cognitive constraints on multimedia learning: When presenting more material results in less understanding. *Journal of Educational Psychology, 93,* 187-198.

McCarthy, B. (1997). A tale of four learners: 4MAT's learning styles. *Educational Leadership*, *54*(6), 46-52.

Michigan (2006). Online Instructional Design, Michigan Virtual University, http://standards.mivu.org/overview. Last accessed: April 13, 2006.

Mohler, J.L. (2001). Using Interactive Multimedia Technologies to Improve Student Understanding of Spatially-Dependent Engineering Concepts. *Grahpicon*, 2001.

Peterson, G.D. & Feisel, L. D. (2002). e-Learning: The Challenge for Engineering Education, *Proceedings, e-Technologies in Engineering Education, A United Engineering Foundation Conference*, Davos, Switzerland, 11– 16 August, 2002, *http://services.bepress.com/eci/etechnologies/* , pp.164–169.