AC 2011-545: EXPERIENCES OF TEACHING COMPUTER GAME AND MULTIMEDIA SEQUENCE COURSES IN THE ELECTRICAL AND COMPUTER ENGINEERING TECHNOLOGY PROGRAM

Li Tan, Purdue University North Central

DR. LI TAN is currently with the College of Engineering and Technology at Purdue University North Central, Westville, Indiana. He received his Ph.D. degree in Electrical Engineering from the University of New Mexico in1992. Dr. Tan is a senior member IEEE. His principal technical areas include digital signal processing, adaptive signal processing, and digital communications. He has published a number of papers in these areas. He has authored and co-authored three textbooks: Digital Signal Processing: Fundamentals and Applications, Elsevier/Academic Press, 2007; Fundamentals of Analog and Digital Signal Processing, Second Edition, AuthorHouse, 2008, and Analog Signal Processing and Filter Design, Linus Publications, 2009.

Experiences of Teaching Computer Game and Multimedia Sequence Courses in the Electrical and Computer Engineering Technology Program

Abstract

In this paper, we present our experiences for developing and teaching computer game and multimedia sequence courses in our electrical and computer engineering technology (ECET) program. The first course in the sequence is an introduction to computer games, where a software platform, Alice, is adopted to develop programming concepts and working knowledge of multimedia applications using audio, image, animation, and video data. The second course requires students perform multimedia programming using the well-known software, DirectX with a Microsoft Visual C++ platform. Students learn how to integrate graphics, animation and sound assets into interactive environments.

We find that the sequence courses stimulate students to achieve their learning objectives in the ECET curriculum via developing their computer game assignments and projects as a vehicle. Furthermore, the student retention is greatly improved. In fact, the sequence essentially establishes a bridge between computer game courses and a traditional digital signal processing (DSP) course offered in our ECET program. In this paper, we first outline the course content and present our pedagogy for teaching the sequence courses. Then, we examine course assessment and analyze outcomes of student learning effects in their upper level course study. The performance comparisons between the students who have completed the game and multimedia sequence courses and the traditional students who are used to do paper exercises and textbook reading, are addressed. Finally, an improvement of the sequence content is proposed.

I. Introduction

Pedagogical computer game-based teaching has recently received much attention, in the potential of computer games for increasing motivation, as effective teaching and learning tools, and enabling different learning experiences other than traditional teaching methods[1-5]. The advances of computer game-based teaching are in the areas of lower education[2] to higher education[3], American history, computer fundamentals, computer programming, science and engineering disciplines[3-4], and research in the use of this teaching approach[5]. The rationale using computer games for teaching may include the following facts[1-4]: 1. many students are less motivated in their traditional teaching learning environment; 2. computer related subjects seem more interesting; 3. many students have been exposed to a game environment and some of them are even computer hobbyists; 4. students also appreciate a flexible and collaborative learning environment. Most research outcomes have shown a positive learning perception of computer game-based learning approaches.

Similarly, in the age of computer technology, engineering technology students always find courses with computer game related applications more motivating and interesting than traditional courses taught using classical methods. As a mater of fact, many yang students (new generation) who have chosen their career path in engineering technology are greatly influenced by the current game industry. In addition, the majority of technology students also perceive a higher learning effect when working on their computer assignments and projects rather than weekly paper exercises and textbook reading. To take this fact into account to enhance our electrical and computer engineering technology (ECET) program as well as student retention, we have designed and implemented computer game and multimedia sequence courses in our curriculum. The first course is an introduction to computer games using the Alice platform[6-7] with a focus on programming concepts instead of programming languages. The second course in the sequence requires students perform multimedia programming using the well-known software, Direct X10[8], with emphasizing on real world applications.

We find that the sequence courses stimulate students to achieve their learning objectives in our ECET curriculum via developing their computer game assignments and projects as a vehicle. As a result, the sequence essentially establishes a bridge between computer game courses and a traditional digital signal processing (DSP) course with multimedia applications offered in our ECET program. In this paper, we first outline the sequence content and present our pedagogy for teaching the sequence courses. Then, we examine course assessment and analyze outcomes of student learning effects in their upper level course study. The performance comparisons with the pros and cons, between the students who have completed the game and multimedia sequence courses and the traditional students who are used to do paper exercises and textbook reading, are addressed. Finally, an improvement of the sequence content is also proposed.

II. Course Sequence Outline

Our computer game and multimedia sequence contains two courses each with a 17-week class schedule. The first course, which studies computer game software and hardware, is offered at the second semester in the sophomore year. The software Alice platform is adopted to develop programming concept and working knowledge of multimedia applications using audio, image, animation, and video data. The advantage of using this revolution platform (Alice) is as follows: 1. easy of program creation, where the Alice editing environment allows students to develop the syntax using “drag” and “drop” actions; 2. viewing interactive results, and three-dimensional graphics and animation actions; 3. teaching programming concepts without requiring programming languages such as C and C++ languages. With these features, any student in his/her sophomore year is able to take the course. The second multimedia programming course, offered at the first semester in the junior year, allows students to learn how to integrate graphics, animation and sound assets into reactive environments using the DirectX. Today, most PC games and multimedia applications on the market are constructed using the DirectX. Using this platform, students can experience the latest graphics technologies as well as practical world applications.

The sequence is designed with a motivation to improve student learning effects as well as student retention. Although the first course requires no prerequisite, it serves as a prerequisite for the second course. Each course has 4-credit hours and is conducted in a blended mode, in which half of class time (three hours per week) is allocated on site for the instructor’s lectures and students’ demonstrations, while the other half (three hours per week also) is a section for student development with an on-line help by the instructor. Figure 1 shows the flow chart of the computer game and multimedia programming sequence courses and its relation to other ECET courses.

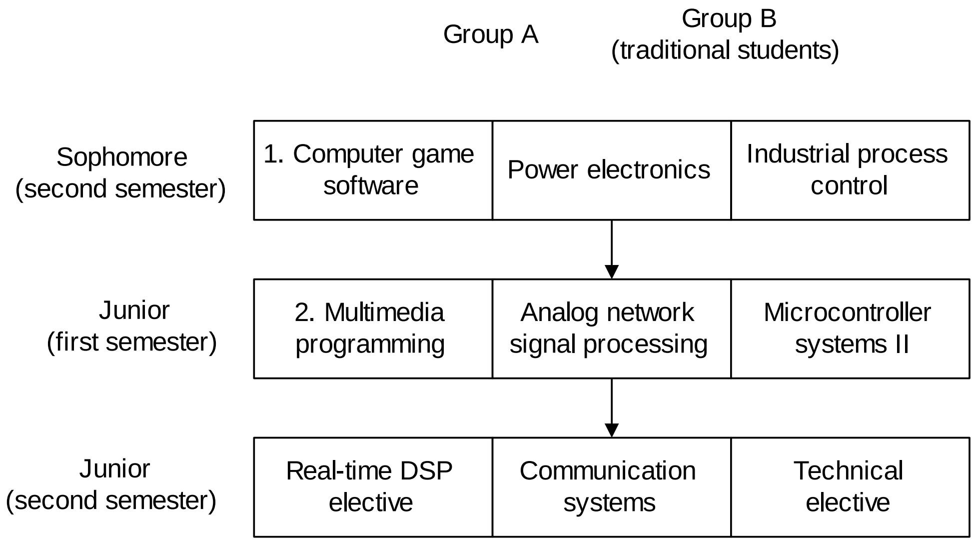


Figure 1. Flowchart of the sequence courses and the related courses.

As shown in Figure 1, we have two group students. Students in Group A choose computer game and multimedia sequence courses instead of industrial process control and microcontroller system II courses taken by traditional students in Group B. Note that both group students have completed the microcontroller system I and have basic programming skills before entering the second semester in their sophomore year. Hence, the first sequence course will further foster more advanced programming concepts used in game systems. In the second semester of the junior year, both group students will take communication systems course and an elective, chosen as the real-time DSP or the other technical elective. In the second sequence course, students equipped with C programming skills and concepts, are required to create the multimedia programming applications. Concurrently, they study a course tilted as analog network signal processing, in which the topics of Laplace transform, Fourier series, Fourier transform, and analog filter design are covered. After successfully completing both first-semester junior-level courses, students will have an opportunity to pursue the real-time DSP course, in which they study concepts of discrete-time systems, digital filtering and processing, and DSP related applications. These two sequence courses essentially support both real-time DSP and communication system courses, since the multimedia sequence course copy with processing of audio, image, and video data, and computer graphics.

A. Course Learning Outcomes

The first sequence course using the Alice platform covers the following key topics: (1) creating a simple animation program; (2) built-in functions and expressions, and control structures; (3) object-oriented programming; (4) events and event handling; (5) functions and controls; (6) recursion and repetition; (7) multimedia display systems and control devices; (8) basic audio and audio formats, image and image formats, graphics, color models, graphics formats, and video and video formats. Topics 1- 6 cover programming concepts in a textbook[6] while the rest topics prepare students with fundamentals and knowledge of multimedia[8] for the second course in the sequence. The course learning outcomes to satisfy the minimum requirements are developed and summarized in Table 1. Using the DirectX and Microsoft Visual C++ platform, the second multimedia sequence course will further study the following topics: (1) color and shading, models lighting with controls, and blending and obscuring objects; (2) rendering images with geometry, viewpoint, texture and lighting information; (3) textures, manipulating objects and projections, and meshes for positioning; (4) video and audio interleaving; (5) 2-dimension modeling; (6) 3-dimension animation; and (7) display and input controls. Table 2 includes the corresponding course learning outcomes.

Table 1. Course learning outcomes for the first sequence course.

O1. Create a simple animation program that includes the background scene, objects, and motion. O2. Create an animation program using built-in functions and expressions, and control structures. O3. Create an animation program using object-oriented programming. O4. Create a game simulation that involves with events and event handling. O5. Create an animation program using functions and controls. O6. Create an animation program using recursion. O7. Contrast display systems and input devices. O8. Characterize basic audio and audio formats, image and image formats, graphics, color models, graphics formats, and video and video formats. .

Table 2. Course learning outcomes for the second sequence course.

O1. Create simulation using color and shading, models lighting with controls, and effects with blending and fog. O2. Render images with geometry, viewpoint, texture and lighting information, O3. Create simulations with 2D, 3D textures, objects and projections and meshes. O4. Interleave video track and audio track. O5. Create 2D text and 2D modeling for display, create 3D animations. O6. Create simulations with multimedia display systems and control controls.

B. Laboratory Design and Teaching Pedagogy

Table 3 lists our labs in order to meet the course learning objectives in Tables 1-2. Again, the mappings between the developed labs and course learning outcome(s) are also included.

Table 3. List of labs and their mappings to the course learning outcomes for each course.

Sequence course I (Alice platform)	Outcomes in Table 1	Sequence course II (DirectX,	Outcomes in Table 2
1. Alice programming environment, build-in functions, expressions, control structures	(1), (2)	1. DirectX programming environment	(1)
2. Objected-oriented programming, Classes, objects, methods and parameters	(3)	2. The 2D Simulation development and animation	(2)
3.Event-driven programming	(4)	3. Text and fond rendering	(2), (3)
4. Programming using functions	(5)	4. 3D Primer and 3D basics	(3)
5. Programming using recursion	(6)	5. Shaders and effects	(1)
6. Research paper	(7), (8)	6. Direct 3D	(5)
Final project	All	7. Direct input and X input	(6)
		8. Direct sound	(4)
		Final project	all

For each course, our labs are divided into two parts: regular labs and a comprehensive project. We require no more than two (2) persons per group for the comprehensive project. At the end of the course, each group will demonstrate the functionality of their developed project meanwhile their project will be evaluated by the instructor and students from the other groups. We focus on illustrating our teaching pedagogy by briefly describing the features of labs and final developed projects.

The course is offered in a blended mode, that is, there are weekly three (3) hours for face time during which the instructor teaches the required programming concepts and leaves the rest of time for students to demonstrate their labs developed from the last period and continuously work on current assignments. Students are also required to read the related class material prior each lecture. For the left three (3) hours outside class, students will work by themselves. The instructor will provide an on-line help with the dedicated three hours period per week. Figure 4 displays a snapshot for a portion in lab 4 in the first sequence course, where students are introduced concepts of class, objects, methods and parameters in the lecture part and each student uses these elements to create a program outside the class face time. Figure 2 shows the corresponding simulation, in which a skater skates round a penguin and a duck, skates forward and backward, spins and jumps using the methods, classes, objects discussed in the text. In the next class meeting, after completion of new instructions, each student will demonstrate his/her work in class. In this interactive environment, they can share their learning experiences. Six regular labs are assigned in total and each lab is scheduled to be completed in two weeks. After students complete programming fundamentals, they will form a group to develop a comprehensive game project. The project will take about 4 weeks to complete. At the same time, four (4) lectures are given for the topics of game display systems and control devices as well as the topics of multimedia fundamentals such as audio, image, video coders and their data formats, and computer graphics[9-10]. An additional research lab is then assigned to each student to explore these concepts, which will serve as a foundation for the second course in the sequence.

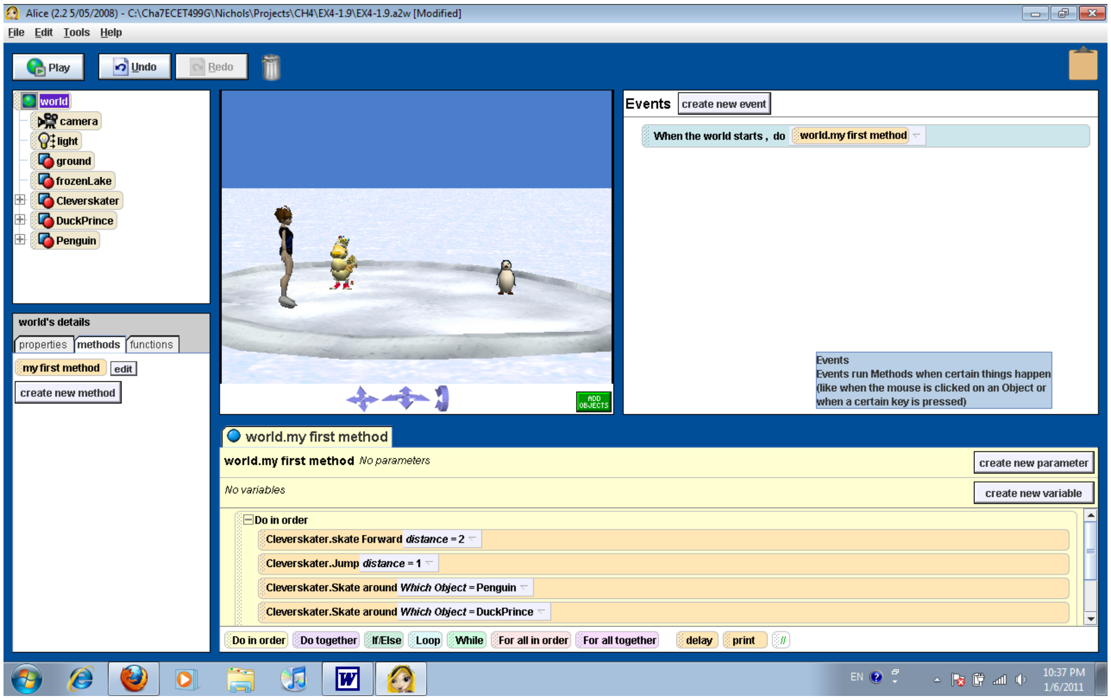
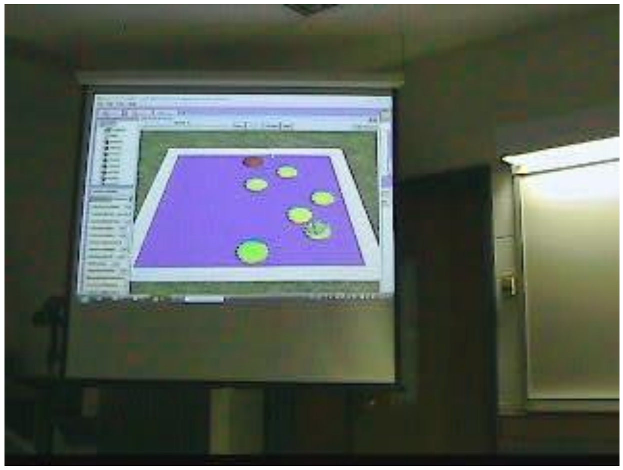


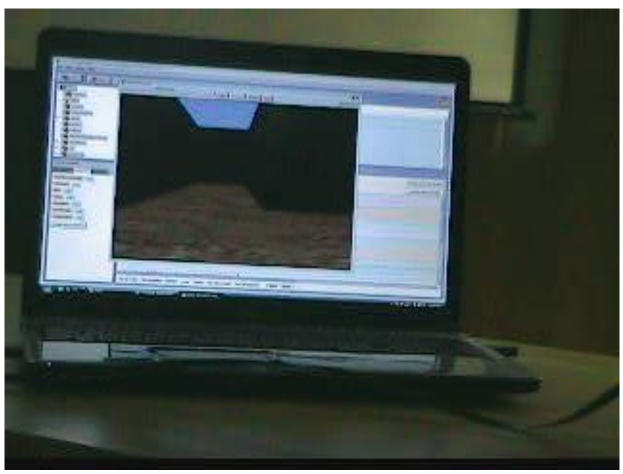
Figure 2. Sample program for the skater in lab 4.

Figure 3. Simulation example in lab 4.

Figure 4 shows the sample game projects developed by students. Each game is running in an infinite loop with a selection of a difficulty level companying with audio. Every final project is evaluated by the course instructor as well as students in other groups. Although in this course, C++ programming is not required, most concepts of object-oriented programming are covered.



(a) Frog crossing game
(Frog tries to jump across a pond)



(b) Wonderful maze of fun
(A person tries to walk out the maze)

Figure 4 Sample student projects developed in the first sequence course.

The second sequence course is mainly focused on multimedia applications using the DirectX, which is the premier game API for the Windows platform. The DirectX is based on a collection of code libraries, where a common set of functionality are provided for multimedia applications. It contains the following components[8]: DirectX Graphics (handling all the graphics processing), DirectXInput (all the user input via the API), Xinput (new additional component for interface such as Xbox 360 controller), DirectSound (adding sound and sound effects), and DirectSetup (setup for the finalized version). As shown in Table 2, the course is allocated with eight regular labs and a final project. The pedagogy for teaching this course is the same as the first sequence course. However, the second course is much more challenging, since students have to work with C++ programming in the Microsoft Visual C++ environment. Figure 5 demonstrates an example in lab 5, where students create a triangle with different vertex colors. The details of information about all regular labs can be found in reference[8]. Figure 6 shows a demonstration of a selected group project developed by students.

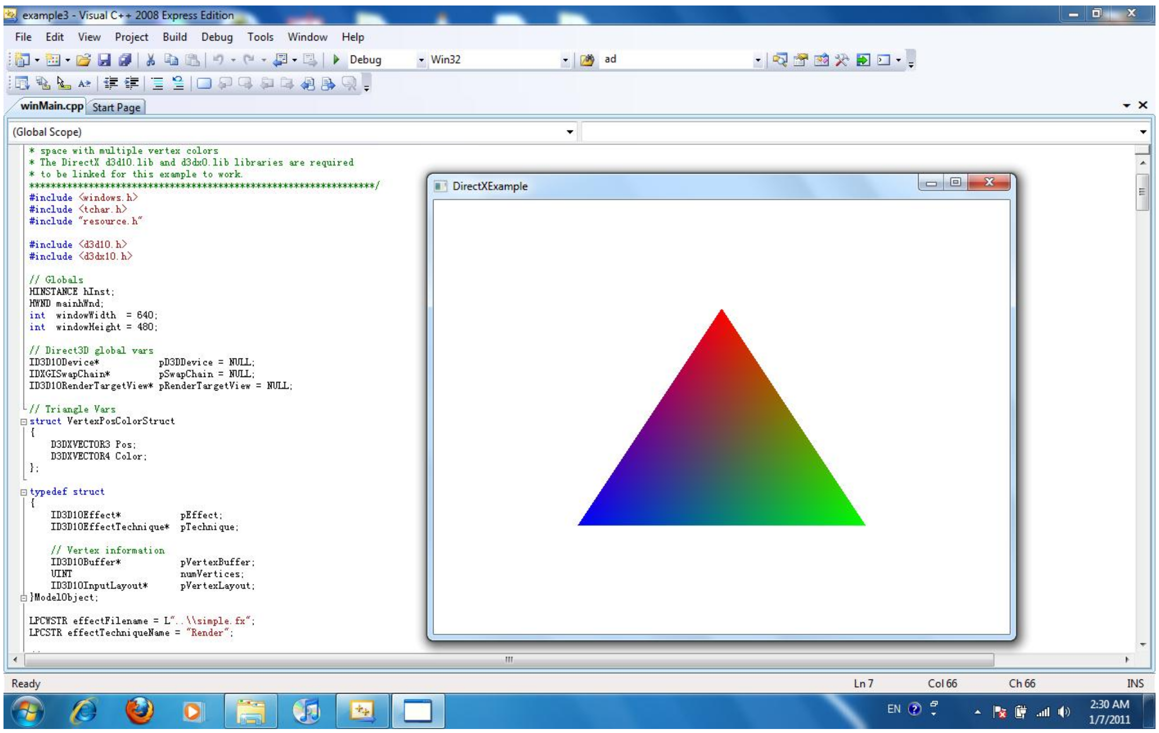
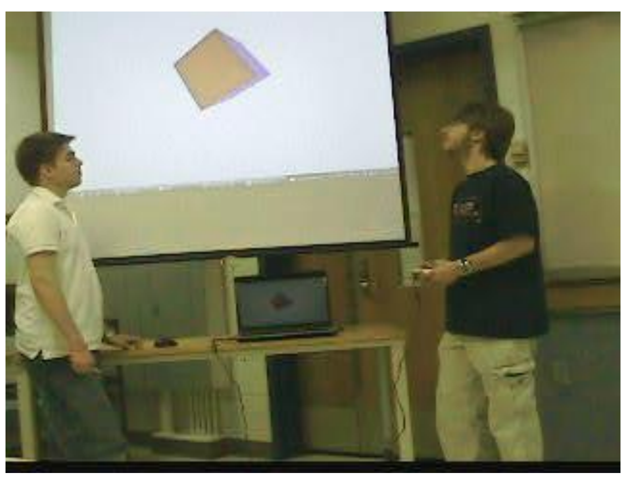


Figure 5. An example of a triangle with multiple vertex colors in lab 5.



Controlling the "cubik" with direction and rotating speed, size, colors, fading in/fading out effects, and sound effect

Figure 6. Selected student projects in the second sequence course.

III. Learning Outcome Assessment

The assessment results presented here are based on our collected data from teaching the sequence courses in fall 2008 and spring 2009. At the end of each semester, we conducted a student survey. This survey was given to each student to evaluate his/her achievement for each course learning outcome listed in Table 1 and Table 2, respectively. Students were asked to make the following five (5) choices: understand well, understand, somewhat understand, somewhat confused, and confused. For statistical purposes, the five choices were assigned the scores of 5, 4, 3, 2, 1, respectively. The average rating scores for all the course learning outcomes for each course are listed in Table 4.

Table 4. Student survey

Outcomes	O1	O2	O3	O4	O5	O6	O7	O8
1. Computer game	4.8	4.5	4.6	4.8	4.5	4.6	4.2	4.0
2. Multimedia	5	4.25	4.0	4.5	4.7	5.0		

From Table 4, we observed that

(1) The rating scores from student survey for both courses indicate that majority of students are satisfied with the courses. For the first course, the outcome 7 and 8 were rated low. This is due to the fact that students were required to investigate the game display systems and control devices via lectures and their research papers. Students were less motivated on the paper exercise. This is the area that may require an improvement.

(2) Outcome 3 in the second course was rated low as compared to others. We found that the topic deals with computer graphics using geometric transformations[10]. Our students were weak in this area. We will improve it by increasing more lecture time on this topic.

In addition, we also conducted another student survey regarding the useage of the Microsoft Visual C++ platform in the second course. A set of questions listed in Table 5 were given to students for evaluation. We also used five (5) choices for students to evaluate these questions: strongly agree, agree, somewhat agree, disagree, and strongly disagree. The corresponding rating scores were assigned as 5, 4, 3, 2, and 1. The average ratings were given in Table 6.

Table 5. Survey using the DirectX and Microsoft Visual C++ platform.

Q1. Do you feel you can significantly grasp concepts by working on multimedia processing coding?
Q2. I am excited to work on multimedia coding labs and projects?
Q3. Does multimedia coding improve your problem solving ability using C++?
Q4. Do you want to learn more on multimedia programming subjects and applications if possible in future?
Q5. Do you feel this course can help you for other course(s)?
Q6. Please give some suggestions for improving this course:

Table 6. Student survey for Visual C++ .

Question No.	Q1	Q2	Q3	Q4	Q5	Q6
Student Survey	4.5	5.0	4.25	4.75	4.25	N/A

The average rating scores shown in Table 6 indicated what follows.

(1) Students strongly agreed that they have learnt a great deal of multimedia applications via using the DirectX and Microsoft visual C++ platform as a vehicle.
(2) Most of the students remained excited about the course, since hands-on real-time laboratories motivated them. They had shown their continuous interests in learning more about multimedia programming and applications.
(3) Outcomes 3 and 5 were rated relatively low. These two outcomes are heavily relies on the student programming skills. We found that some students were struggling at the beginning of the class to familiarize with the Direct X and Visual C++ platforms. The student programming skill remains to be improved. On the positive site, students were eager to learn more multimedia application related subjects. In fact, about 75 % of students (Group A in Figure 1) chose to continue the real-time DSP course as their elective course (see Table 7 below), in which real-time speech and audio processing techniques using digital filters[11-12] are studied.
.
In order to validate the student learning effect, we kept track of students in Group A for their retention in the program and performance in the traditional DSP course. As shown in Table 7, after the first course using the Alice platform, more than 90% students remained interested in the sequence and continued the second multimedia programming course. Most students (75%) chose the real-time DSP course as their elective; and more than 90% students in Group A stayed in the ECET program. These results indicate that the designed sequence courses are effective to enhance the student retention. From our current data, only 40% students in Group B (see Figure 1) chose the DSP course as their elective. The program retention rate for students in Group B is also lower.

Table 7. Student retention rates

Student type	Entering 2nd course	Entering DSP course	Staying in ECET program
Group A	91.6%	75%	93 %
Group B	N/A	40%	81%

Table 8 compares the academic performance of Group A with traditional Group B in their later traditional DSP course (three hours lectures and three hours lab per week). Students were required to complete ten (10) homework assignments, three (3) exams, six (6) labs and one (1) group project[8]. There were 40% students in Group A while 60% students in Group B. The conclusions were drawn from comparing the overall averaging ratings of the homework assignments (paper exercises), exams, and labs and project each with a full scale of 100 points.

Table 8 Student performance comparisons of Group A over Group B.

Paper exercises	Exams	Labs and project
Group A: Average=73.4 (Std=11.8)	Group A: Average=80.1 (Std=8.5)	Group A: Average=97.3 (Std=3.3)
Group B: Average=88.3 (Std=3.1)	Group B: Average=87.6 (Std=4.3)	Group B: Average=92.9 (Std=4.9)

Although for paper exercises and exams, students in Group A, who completed the computer game and multimedia sequence, did not show better performance than the traditional students in Group B. However, for computer related labs and projects, we received the positive results, since students in Group B had an early exposure to computer solution problems and projects in their sequence courses.

As for the general academic performance in our ECET program, we observed the same trend. Students in Group A did not show better performance in comparison to traditional students in Group B in the traditional courses such as communication system II, advanced DSP, and feedback control systems, but they outperform in the computer related courses such as the microcontroller system II. Furthermore, as pointed in Table 7, students in Group A had a higher ECET program retention rate.

IV. Lessons Learned and Future Improvement

We successfully implemented the computer game and multimedia sequence in our ECET program. The sequence courses were motivational to our technology students. We received a better enrollment in the upper-level courses as well as a better retention rate in the ECET program. Although the skills in terms of paper and exercises used in traditional courses were not much improved, students in Group A outperform in hands-on assignments and computer related courses. We suggest that the sequence courses should serve as an option for students to balance their traditional courses in the ECET program.

Based on our experiences in teaching the computer game and multimedia sequence, we felt that each course contains well-established topics, with suitable lectures and laboratories. Students remain excited in area of multimedia programming and applications. According to our assessment results, the sequence can be improved in the following areas. For the first sequence course, one or two hands-on labs can be added to improve the topics of the game display systems, input devices, audio, image and video data formats. To smooth a transition from the first course to the second, a component for reviewing C++ programming at the beginning of the second course is suggested to strengthen student programming skills; to improve student understanding of the principles of geometry transformations, more lecture time should be allocated; finally students can be encouraged to develop more comprehensive and challenging projects.

VI. Conclusion

We have developed the computer game and multimedia sequence for the electrical and computer engineering technology program. From our course assessment, we have found that the student retention and computer programming skills are significantly enhanced. Students remain excitement about multimedia applications and increase their interests to take the real-time DSP course. Although we do not observe better results in terms of paper exercises and exams as compared to traditional students who used to do paper exercises and textbook reading, students demonstrate better performances in their computer related labs and projects.

Bibliography

1. G. Sindre, L. Natvig, M. Jahre, "Experimental Validation of the Learning Effects for a Pedagogical Game in Computer Fundamentals," IEEE Transactions on Education, Vol. 52, No. 1, pp. 10-18, February 2009.
2. A. Prensky, "Digital Game-based Learning, "ACM Computers in Entertainment," Vol. 1, No. 1, October 2003.
3. H. Jenkins, E. Klopfer, K. Squire, and P. Tan, "Entering the Education Arcade," ACM Computers in Education, Vol. 1, No. 1, October 2003.
4. L. Natvig, S. Line, A. Djupda, "Age of Computers: An Innovative Combination of History and Computer Game Elements for Teaching Computer Fundamentals," in Proc. 34th ASEE/IEEE Frontiers in Education Conference (FIE) 2004, Vol. 3, pp. S2F 1-6.
5. N. Whitton, "Motivation and Computer Game Based Learning," Proceedings Ascillite Singapore, 2007.
6. W. Dann, S. Cooper, R. Pausch, Learning to Program with Alice. Second Edition, Prentice Hall, 2008.
7. T. Gaddis, Starting Out with Alice: A Visual Introduction to Programming. First Edition, Pearson Education, Inc. 2008.
8. W. Jones, Beginning DirectX10. Premier/Thomson-Course, 2008.
9. Z. Li, M. Drew, Fundamentals of Multimedia, Prentice Hall, Upper Saddle River, NJ 07458, 2004.
10. H. Baker, Computer Graphics with OpenGL. Update Edition, Third Edition, Prentice Hall, 2004.
11. L. Tan, Digital Signal Processing: Fundamentals and Applications, Elsevier/Academic Press, 2007.
12. L. Tan,, J. Jiang, "Improving Digital Signal Processing Course with Real –Time Processing Experiences for Electrical and Computer Engineering Technology Students," ASEE Annual Conference, Louisville, Kentucky, June 2010.