Media Computation Projects for High School Students

Dr. Suxia Cui

Electrical and Computer Engineering Department

Prairie View A&M University

Dr. Yonggao Yang, Dr. Akhtar Lodgher

Computer Science Department

Prairie View A&M University

Abstract

This paper presents part of the work from a NSF sponsored three-year Broadening Participation in Computing (BPC) project which utilizes several software tools to teach high school students basic computer knowledge and attract them to the computing discipline. The main carrying vehicle of this project is teaching modules that feasible for high school students. On the first year's implementation, modules utilizing Alice and 3D game-engine Vizard were successfully developed. This article shares our second-year experience of involving high school students in media computing projects. Media Computation is an approach to introduce computing through manipulating various media data, including image files, audio files, and video files. Three categories of modules were developed and introduced to students to manipulate digital image, audio and video data in summer training and main semester high school class visit. Students are able to experience the changing color of an image, reposition and resize the image just by typing several lines of computer instructions in the Jython Environment for Students interface. Students are very excited when they can make their own movies by linking still images frame by frame. Also sound manipulation is one of the most favorable experiments among all the media computation modules. Students like the way of making weird sound effect, adding sound to pictures, and making their own music to enjoy. In this paper, successful approaches are shared with survey data to support the finding.

Background

There is no doubt that computer is one of the fastest growing industry. Computer techniques have changed the world in every aspect [1]. Nationally, the demand for computer software engineers, computer information system, and applications is among the top fast growing occupations with an increasing percentage of 34% [2]. But over the past decade, student interest in computing has declined dramatically. For example, the percentage of college freshmen indicating intent to major in computing has declined overall by 70% in the last decade; for women, the decline was 80% [3]. It was observed that one crucial reason of this decline is due to the unfamiliar of computing disciplines before they get in touch with computing core courses at their junior or senior college year [4-6]. To face this challenge, educators from universities collaborate with teachers and administrative from K-12 to update curricula and design various recruit activities to attract youngsters at an early age to computing disciplines. Several successful projects all show the success of using multimedia and game-like environment in teaching youngsters computer [6-11]. Here the authors would like to share the experience of one educational project entitled "Building Computing Aptitude, Confidence, and Engagement for Students (Computing ACES)" funded by NSF focuses on recruitment (award # CNS-0940561 from March 1, 2010 to Feb 28, 2013).

The goal of this Computing ACES project is to increase the number of high school students who enroll as undergraduates in computer science and computer engineering from groups underrepresented in the computing disciplines, including women. The project objectives are to (1) Engage high school students' interest in computing disciplines and strengthen their aptitude in computing; (2) Increase high school students' awareness of careers in computing disciplines aligned with their areas of interest and dispel commonly held myths about computing careers, especially for women; and (3) Encourage and assist qualifying students to apply to college in a

computing discipline, particularly in computer science and computer engineering. This three-year project targets secondary school business computing and technology courses where students are taught how to use computer technology but not necessarily the computer programming behind it. The carrying vehicles are the multimedia and game-like teaching modules developed by university faculty members. The software environment chosen for the teaching modules are: Alice [12], Vizard [13], and Media Computation [14]. Several interactive activities are (or going to be) involved to deliver the modules to local high school students. The major project activities include:

(1) High school teachers attend a four-day summer workshop in university to get familiar with the software and teaching modules which will be used for university faculty high school class visit in the following academic year.

(2) University faculty visits high school classes: University faculty and student assistants will go to high school classes to introduce Alice, media computing, and Vizard game-like teaching modules every other week throughout fall and spring semesters.

(3) High school students come to university monthly to tour computer science and computer engineering labs. Parents are invited to university for a Computer Festival in annual National Engineers Week. Each summer, a 5-day Computing Fiesta is going to host high school students for more fun activities.

After the first year's kick off, the collaboration among educators from different levels was established through summer high schools teachers' training, discussion and surveys. Game-like teaching modules with Alice and Vizard were developed and introduced to high school students. Positive feedback was obtained through surveys and observation [15]. The second year of implementation, all the successful activities were continued, and the new teaching modules from Media Computation were the focus of curriculum development. It was previous observed that moving from Alice, a drag-and-drop environment to a text-based Media Computation software environment, like DrJava [16], is very challenging for students, especially at middle school or high school levels. This time, a relatively easier environment under Python, JES (Jython Environment for Students) [17], was selected. A brief illustration of the Media Computation teaching modules are introduced in below content with survey results and conclusion.

Media Computation Teaching Modules

Among all the teaching modules developed for Media Computation under JES, we can group them into three categories: image manipulation, creation of sound, and generate video. Each category has at least five individual small projects to be completed through high school class. The textbook "Introduction to Computing and Programming in Python" [18] is used as reference. Since the media computation modules will be introduced to high school class in fall 2011 and spring 2012semester, they were tested in 2011 summer student workshop to see the effectiveness. The 5-day Computing Fiesta was conducted from June 13 to June 17, 2011. Total participants were 15 high school students. Besides one full day of media computation project, Alice and Vizard projects were also revisited one day before and after media computation respectively. University faculty also arranged campus tour, talk and presentation, meet with recruit experts, and financial aid guidance to provide high school students a broader understanding of higher education. The students got the first hand experience through the below teaching modules, and

the surveys were obtained from this Computing Fiesta regarding the Media Computation portion. Here we take one example from each of the category:

1. Image manipulation
   The image projects combined the basic introduction of how digital images are formed, how to change the colors, scale, rotate, and display them. One of the examples is to create sunset scene by reduce the green and blue portion of the image. The sample code and results are shown in Fig. 1 and Fig. 2.
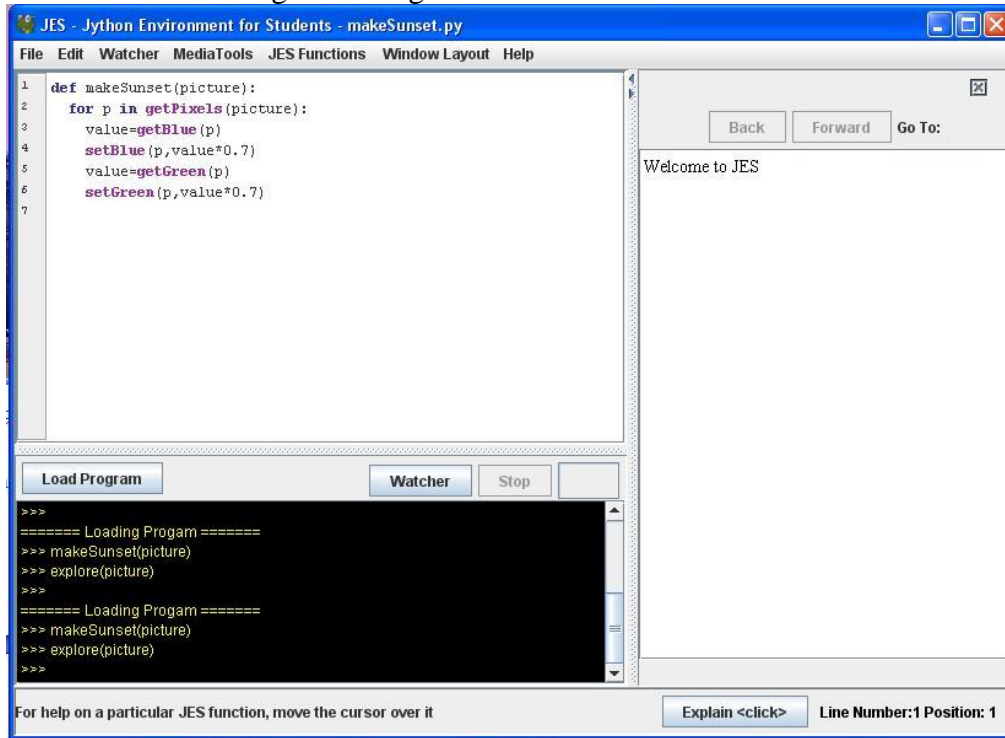

Fig. 1 Sample program "makeSunset" code runs in JES environment

Fig. 2 Original image (left) and the result (right) of "makeSunset" effect

2. Creation of sound

In the sound project, students learned the concept that physically sounds are waves of air pressure and the shape of the waves, their frequency, and their amplitude all impact what we perceive in the sound. In the hands-on practice, they can adjust the volume of the sound, merge different sounds files together, produce echo, and play a sound backward. Fig. 3 illustrates the original sound file wave and the reversed sound wave. Students were able to observe the signal change at the same time of listened to the changes from the effect.
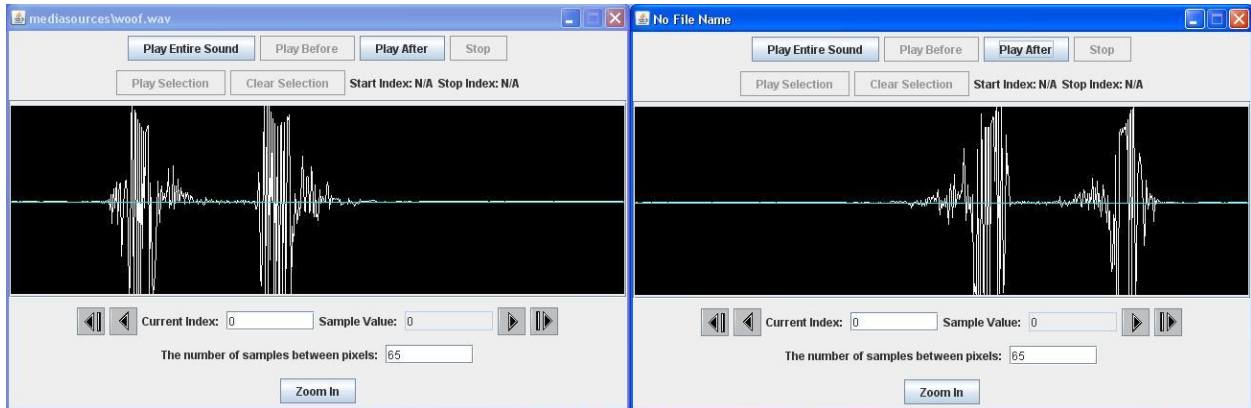

Fig. 3 The waveform of the original (left) and reversed (right) sound

One of the most interesting projects is to create their own music by playing MIDI (Musical Instrument Digital Interface) notes. JES has developed a "playNote" function that takes three parameters as input for a MIDI sound. The first one is the MIDI note correspond to keys (like 62 is for middle C, 62 is for D, and so on, see Table 1 for details), the second is the duration of the sound in milliseconds, and the third is the intensity ranging from 0 to 127. Example given is the first two measures of "Twinkle Twinkle":

```
def song():
        playNote(60, 200, 127)
        playNote(60, 200, 127)
        playNote(67, 200, 127)
        playNote(67, 200, 127)
        playNote(69, 200, 127)
        playNote(69, 200, 127)
        playNote(67, 400, 127)
```

Then students were able to create their music piece based on the information in Table 1 and the knowledge learned. It was very enjoyable to share one's own music with friends. This project was well received by workshop participants.

Octave	Note Numbers											
	C	C#	D	D#	E	F	F#	G	G#	A	A#	B
-1	0	1	2	3	4	5	6	7	8	9	10	11
0	12	13	14	15	16	17	18	19	20	21	22	23
1	24	25	26	27	28	29	30	31	32	33	34	35
2	36	37	38	39	40	41	42	43	44	45	46	47
3	48	49	50	51	52	53	54	55	56	57	58	59
4	60	61	62	63	64	65	66	67	68	69	70	71
5	72	73	74	75	76	77	78	79	80	81	82	83
6	84	85	86	87	88	89	90	91	92	93	94	95
7	96	97	98	99	100	101	102	103	104	105	106	107
8	108	109	110	111	112	113	114	115	116	117	118	119
9	120	121	122	123	124	125	126	127				

Table 1 MIDI notes correspond to keys

3. Generate video
   Videos are sequences of pictures linked together and persistence of vision make it work for our visual system. As the scheme is more complicated, and the programming codes are relatively longer than the previous two projects, instructor chose to provide part of the code, only let students type in key instructions. Thus students were able to create simple movies by themselves. One example is to show a text on a screen. While the program is running, the text is moving from right to left. One sample frame is shown in Fig. 4. Students were very creative on presenting their own sentences, and tried to change different display features of the movie.

Fig. 4 Sample frame of showing text in a movie

Survey Results and Analysis

After the summer students' workshop on media computation, surveys were taken to observe the feedbacks from total of 15 high school participants. The survey was anonymous and with five straightforward questions on the newly introduced media computation technique, and also asked them among all the three modules—image, sound, and video—which one they liked most. The survey is provided in appendix. The survey results are illustrated below from Fig. 5 to Fig. 10:
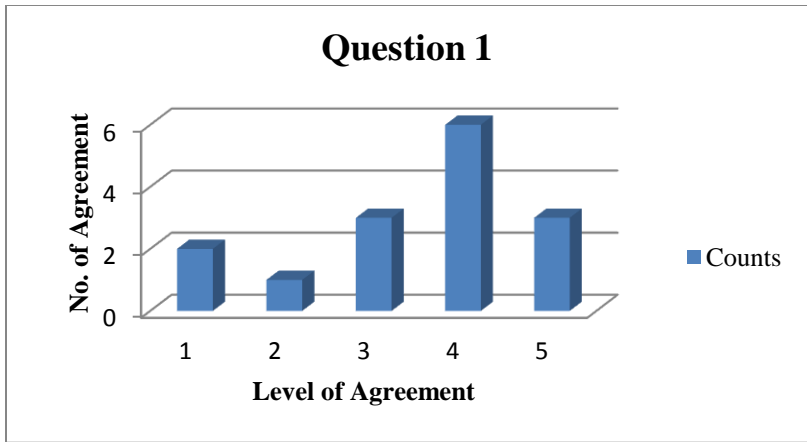
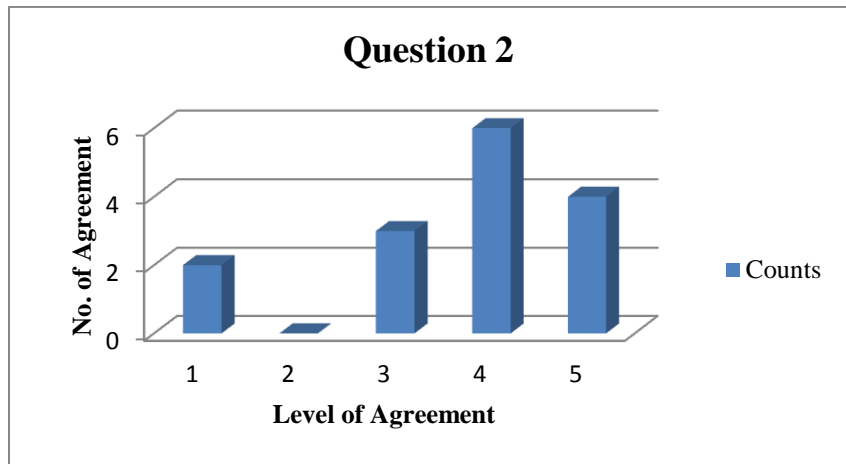Fig. 5 Survey results of question "I found JES environment friendly and easy to use."

Fig. 6 Survey results of question "Media computation did help me understand computer technology."
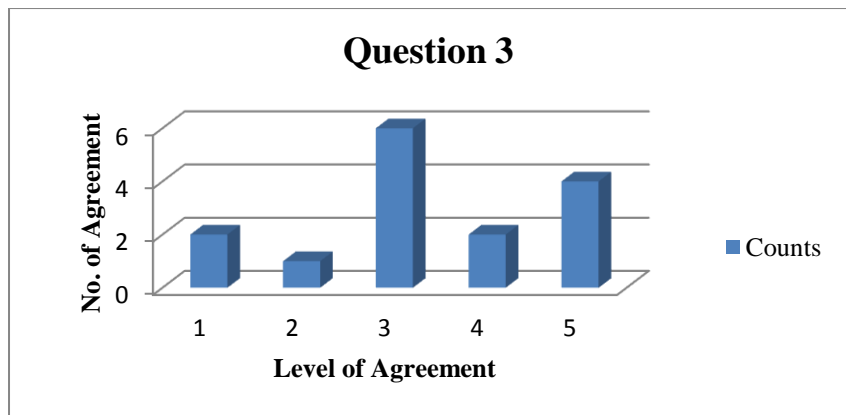

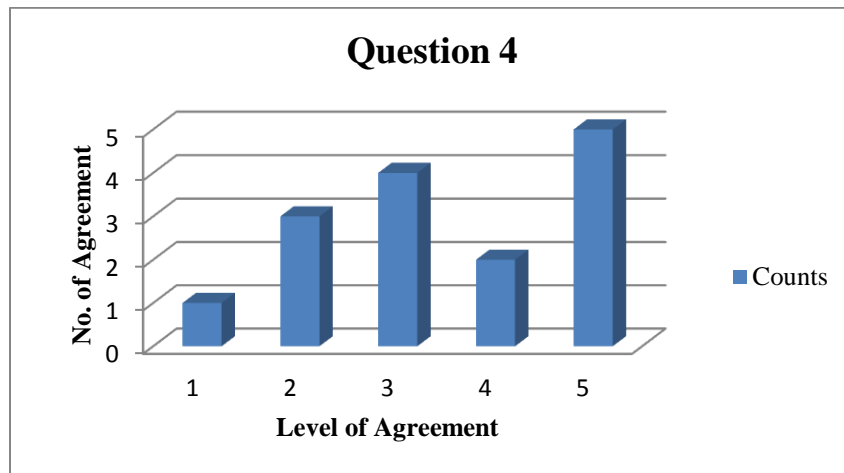Fig. 7 Survey results of question "Media computation increased my curiosity about computer programming."

Fig. 8 Survey results of question "I would like to learn more about media computation projects."
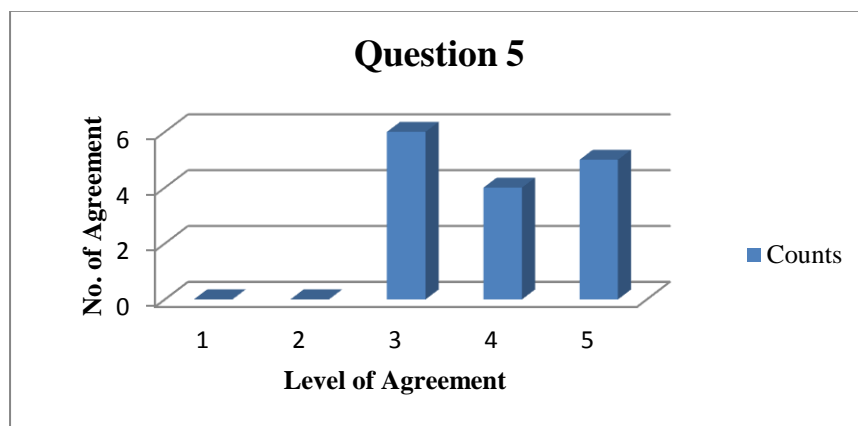

Fig. 9 Survey results of question "After this workshop, I believe I will be able to manipulate my own multimedia files."
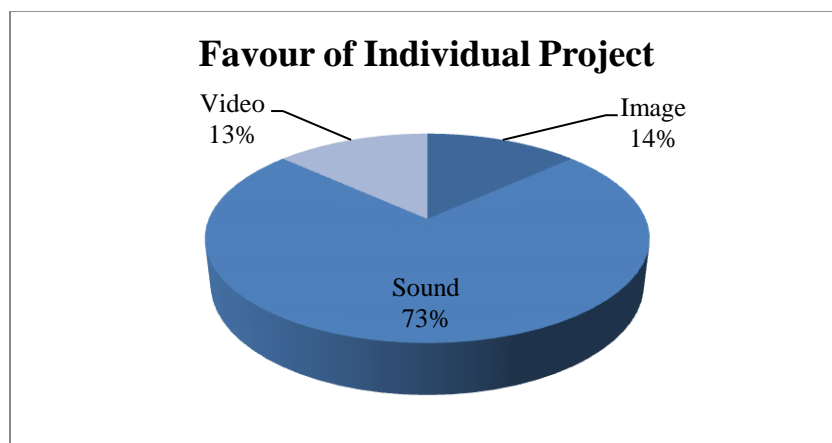

Fig. 10 Survey results for question "Which lab you like most?"

The above survey results proved positive feedback from participating students. As the students group is from high school 9 to12 grade, most of them did not have any computer or programming background. This media computation training was scheduled one day after Alice training, so the transition from "drag-and-drop" style to "text-based" programming environment was investigated. About two out of the fifteen students responded negatively on the "friendly and easy to use" of JES, but majority of the students were satisfied with it. And two thirds of them rated "Media computation did help me understand computer technology" with agreement level 4 and higher. It is also positively shown that all of them were confident about manipulating their own multimedia files after the workshop. To choose "Which lab you like most?", eleven out of fifteen students chose "sound" project which included making sound effects and music. It is similar to the authors' finding in another project focused on incoming freshmen in the College of Engineering where students claimed "computer programming language is a universal language just as music" [4-5]. Another reason for this group of students to favor sound project is because it asked students to make their own music which is more personalized. Other two projects mainly used instructor provided media files since the summer training took place at university, students did not have their own multimedia files with them. This suggested university faculty to update the teaching manuals for major semester high school visit, so students were able to upload their own media files from their school accounts and do the manipulation. Then the favor levels of image and video projects are expected to increase.

Under "Other comments", one student addressed: "I have always been interested in the creation of video games, but haven't thought much of programming. It has been changed now." This comment may reflect the feelings of most workshop participants. It was obvious that students realized the programming environment change and they were facing the challenge of real text-based programming. Also for educators, it is very necessary to introduce both types of programming software and compare them at the early stage to the students. So students will have a better understanding of programming and be motivated to figure out more by themselves.

Conclusions

Though previous study has shown that there exist barriers of moving from Alice based drag-and-drop style to text-based programming environment for certain groups of students, these media computation projects conclude that manipulate visual and audio files are attractive to high school students. Most of the participating high school students were able to conquer the obstruction and reached the goal of these educational projects. Students' surveys also support this positive result.

As it is an ongoing project, the survey data were mainly took from the summer training when media computation was first introduced to this group of high school students, it is very valuable to advise authors to adjust their future teaching modules to fit for this generation's learning steps. For similar approaches, the suggestions are listed below:
1. Media computation is an appropriate tool to introduce computer programming to beginners. It could be directly adopted to college level courses, but to be introduced to high school or middle school level students, it is suggested to have a pre-introduction like

using Alice's drag-and-drop style before teach learners to type lines and lines of their programs.
2. While designing multimedia teaching modules, motivate students' creativities by using their own media files—sound or image.
3. To impact students from all background, a variety of activities should be planned to engage students. Examples are class visits, summer workshops, competitions and so on.
4. In order to have a better recruit results through newly developed educational modules, university faculty should have good collaboration with high school teachers and administration. It is a good asset to have them provide more feedbacks and experience on the design and deliver of new educational modules. This BPC project established a structure that summer workshops and surveys are also conducted to high school teachers, which guarantees a more feasible step will be taken for future development.

Future Work

This paper presents an on-going educational project supported by NSF. The goal is to recruit and retain more students to computing disciplines. Among several successful activities have been taken place for the past two academic years of this three year project, the paper focuses on the development and testing of media computation teaching modules. Three teaching modules: image, sound and video, were introduced during summer workshops. Student surveys were collected. In the future, the teaching modules will be revised and introduced to major semester high school classes. More students will be impacted through this project. It is expected to have a more influential result at the end of this academic year.

Bibliography

[1] A. Aravind, "Students recruitment and retention in computer science: panel discussion", *in proceedings of the 16th Western Canadian Conference on Computing Education (WCCCE'11)*, pp. 43-44, May 6-7, 2011, University of Northern British Columbia, Prince George, BC.

[2] National Center for Education Statistics, *Career and technical education in the United States: 1990 to 2005*, Institute of Education Sciences, Editor. 2008, US Department of Education: Washington, DC. pp. 287.

[3] American College Testing Program (2010), *The Condition of College and Career Readiness*.

[4] S. Cui, Y. Wang, Y. Yang, F. M. Nave, and K. T. Harris, "Connecting Incoming Freshmen with Engineering through Hands-On Projects" *American Journal of Engineering Education*, accepted to appear.

[5] S. Cui, Y. Wang, F. M. Nave, and K. T. Harris, "Teach Computer Techniques through Multimedia," *Proceedings of the 2010 ASEE Northeast Section Conference*, Boston, MA, May 7-8 2010. http://acamedics.com/proceedings/aseene2010/papers/faculty_papers/80.pdf

[6] S. Cui, Y. Wang, Y. Yang, and S. Koay, "Revamp Computer Education with Multimedia and Game", *Proceedings of the 2010 ASEE annual conference & exposition*, Lousville, Kentucky, June 20-23, 2010.

[7] W. Dann, S. Cooper, and B. Ericson, "Exploring Wonderland: Java Programming Using Alice and Media Computation", *Prentice Hall, 1st edition,* ISBN: 0136001599.

[8] C.M. Gorriz, and C. Medina, "Engaging girls with computers through software games", *Communications of the ACM*, 2000, vol. 43, no. 1, p. 8.

[9] C. Kelleher, R. Pausch, and S. Kiesler, "Storytelling Alice motivates middle school girls to learn computer programming", *in Proceedings of Computer/Human Interaction*. Apr. 28 – May 3, 2007, San Jose, p. 1455-1464.

[10] J.P. Cohoon, "An introductory course format for promoting diversity and retention", *in Proceedings of the 38th SIGCSE technical symposium in Computer science education,* 2007. p. 395-399.

[11] M. Guzdial, "Education:Teaching computing to everyone", *Communications of the ACM, May 2009,* vol. 52, no. 5, p. 31-33.

[12] http://www.alice.org

[13] http://www.worldviz.com/products/vizard/

[14] http://mediacomputation.org

[15] S. Cui, Y. Yang, A. Lodgher, and K. Phelps, "Strengthen Computing Discipline Recruitment with Educators Collaboration", *Proceedings of the 2011 ASEE Gulf Southwest Annual Conference*, Houston, TX, Mar. 9-10, 2011. http://www.tech.uh.edu/aseegsw2011/papers/T3A-3.pdf

[16] http://www.drjava.org/

[17] http://coweb.cc.gatech.edu/mediaComp-teach/26

[18] M. Guzdial, and B. Ericson, "Introduction to Computing and Programming in Python: A Multimedia Approach", *Prentice Hall*, 2nd edition, ISBN: 978-0-13-606023-9.

Appendix – Survey Questions

Media Computation Questionnaire

Consider each of the following statements. Please indicate which category best describes your agreement with each statement. (Use 5 as the most agree and 1 as the least agree.)

Statements	Level of Agreement				
	1	**2**	**3**	**4**	**5**
1. I found JES environment friendly and easy to use.	❒	❒	❒	❒	❒
2. Media computation did help me understand computer technology.	❒	❒	❒	❒	❒
3. Media computation increased my curiosity about computer programming.	❒	❒	❒	❒	❒
4. I would like to learn more about media computation projects.	❒	❒	❒	❒	❒
5. After this workshop, I believe I will be able to manipulate my own multimedia files.	❒	❒	❒	❒	❒

Which lab you like most?

1. Image

2. Sound

3. Video

Other comments: