Running head: Learning via Gaming: An Immersive Environment for Teaching Handwriting

Learning via Gaming: An Immersive Environment for Teaching Kids Handwriting

Bruce R. Maxim and Nicholas D. Martineau

Computer and Information Science

University of Michigan-Dearborn

Abstract

Immersive learning via animation, virtual experiments, and simulations is an attractive concept. As the complexity of educational content increases, its delivery methods and pedagogy must improve as well. While the efficacy of immersive environments for education and training is well established, their use with elementary and middle schools students is negligible. Hardware costs and long development times are two major factors impeding development of such environments for younger students. Computing technology, specifically tools for human machine interface development, has come a long way in the past few years. The authors are making use of this emerging technology to develop an immersive gaming environment for teaching handwriting to elementary school children using a tablet PC delivery system.

INTRODUCTION

There has been a fair amount of research on user interface design for children (Read, MacFalane, & Casey, 2003) (Read, MacFarlane, Gregory, 2004), error rates in text entry user interfaces (Read, MacFarlane, & Casey, 2002), interactive teaching environments using Tablet PC's (Willis & Muertschin, 2004) (Koile & Singer, 2006) and using games as tools to promote learning in and out of the classroom (Squire, 2005) (Aguilera & Mendiz,2003). There has been very little research done on combining all of these elements into a tool suitable for teaching children how to write.

A tablet PC comes equipped with a touch screen that is designed to work with a pen input device known as a stylus. Students can use the stylus to perform all user functions associated with an ordinary mouse, but more importantly, the Tablet PC allows students to use the stylus to write directly on the PC display screen. Interacting directly with the PC display screen provides users with an environment that is very easy to use. Studies suggest that young children may have problems using the standard QWERTY keyboard as a writing composition tool. There is some evidence that suggests children may write more easily using a Tablet PC stylus than by typing on a PC keyboard (Read et. al., 2002).

Some teachers feel that computer games can motivate students to maintain their attention on the goals of particular learning activities (Siemer & Angelides, 1995). Video games can teach hand-eye coordination, spatial relationships, and encourage exploratory experiences. Immersion in simulated environments has increased learning speed and retention for some tasks. Video games can engage players for two or more hours, yet these same students may lose interest in classroom activities after only fifteen minutes (Michael & Chen, 2006).

Many people feel that games may enhance the classroom environment by allowing the teacher to play a less dominate role and not being forced to serve as the sole judge of student performance (Siemer & Angelides, 1995). This allows teachers to assume a coaching role and guide students during learning tasks. Games can be a powerful and pervasive way to take learning outside the classroom (Squire, 1995).

When playing a computer game, children may notice the multimedia content in the game, have fun interacting with game elements, and observe how easy some things are to learn (Druin, 1999). There are several interface design issues that need to be addressed when creating game environments for children. Some of these include: creating consistent and predictable user dialogs, offering informative feedback, providing simple error handling, and reducing the user's short term memory load (Read et. al., 2004). Dealing with non-readers requires the use of audio and unambiguous video feedback and user guidance.

Children learn to write at different rates, in different ways, and with different capacities (Thurber, 1986). This makes it difficult to teach handwriting in a typical single teacher classroom. One mechanism for helping to monitor an individual student's progress in classroom setting might be to introduce intelligent tutoring system to guide the delivery of the necessary instructional activities. Intelligent tutoring systems go beyond the limitations of traditional computer-based training systems (Siemer & Angelides, 1995). Intelligent tutoring systems use information on a student's current and past performance to deliver customized content in a style best suited to the student's current instructional needs (Wegner,1987) (Kearsley, 1987).

SYSTEM DESIGN

The goal of this project is to create an immersive gaming environment using a Tablet PC that teaches children to write using the D'Nealian handwriting system (Thurber, 1986). While interacting with the game, students are taught to write new letters or words as a means of opening new reactive game elements. Students gain access to additional game features by writing required letters using the methods taught. An intelligent tutoring system provides feedback on the children's work, telling them what they did right or wrong, and how to correct their errors if any. As players progress through the letter worlds they are given access to entertaining mini-games to reward their mastering letter world objectives. Figure 1 shows a screenshot from the immersive game environment.

The system design contains three key modules: (1) an interactive gaming environment where kids play in a game environment to unlock animations and mini-games, (2) a student writing evaluation and feedback module, and (3) a teacher evaluation and feedback module. The current system is programmed using Microsoft C#, WPF, and the XNA library to ensure the ease installation on the player's home computer (which must a tablet PC).

The storyline begins with factory owned by Ollie the Octopus. Ollie is the chief engineer for Pickle's Printing Playground. Pickles Printing Playground or PPP for short specializes in creating words, using individual robot letters, for people all over the world. Ollie's job at PPP is to answer incoming phone calls, assemble the word orders, and send them out. One day, while busy at work, Ollie's tentacles knock over a cup of coffee that was sitting on the control panel. The coffee spills into the panel and causes the machine to malfunction. When this happens, the robot letters controlled by this machine escape into the playground behind the factory. Pickles job is to enter the

playground, find all of the individual robot letters, capture them by writing their grapheme correctly, and return them to the factory.

Figure 1 Student Screen Layout

When Pickle (the player's avatar) enters the game world, Pickle sees a world a map showing the order in which the letters are to be trapped. This order matches Thurber's sequence for teaching handwriting (Thurber, 1986). The player unlocks the path to each new letter by trapping the next letter in the sequence. For example Pickle cannot go past the "a" world arch until letter "a" has been captured.

Each letter world provides a unique experience to the player. Each letter world Pickle enters is filled with objects of that begin with that letter (i.e. the "g" world features a giraffe as

the main character, a globe, a gopher, gears, a ghost, and a gate to exit to the next letter world). After a player captures the letter “g”, by writing it correctly three times, each “g” world object will react to the Pickle’s touch by playing the object’s animation. The object’s name is presented on the screen and spoken each time the object is animated by Pickle’s touch.

The first time a player writes the letter he or she is allowed to trace a sample letter. The second time and third time a player writes a letter, he or she copies the letter without any step by step guidance. Once a letter has been drawn correctly players are allowed to access the mini-games. The mini-games serve to motivate students to complete levels and to provide practice opportunities on using the stylus and working with letter skills. Figure 2 shows the D’Nealian writing sequence for the letter “a”.

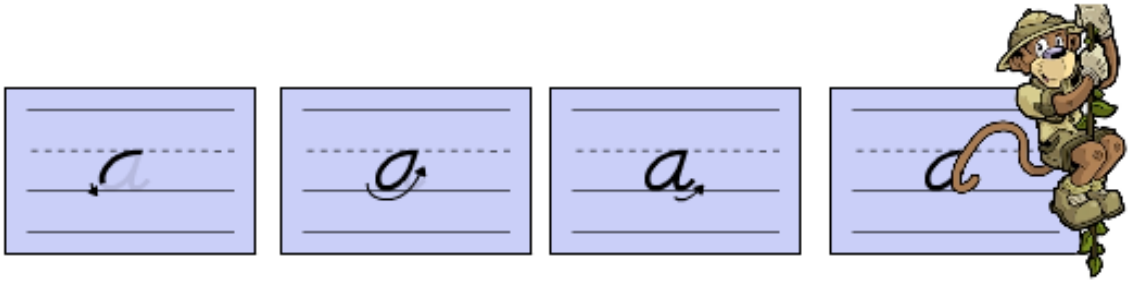

Figure 2 D’Nealian writing sequence for the letter “a”

If students are not able to complete the letter writing task they are given feedback and remedial instruction. For instance if a user is having problems writing the 'monkey tail' on the letter 'a' (the final stroke of the letter) instead of having them rewrite the letter as a whole over and over again, the system focuses on the part of the letter they are writing incorrectly, in this case the 'monkey tail'.

Figure 3 shows the architecture of the intelligent tutoring system. Our system incorporates a series of remediation rules that have been developed by both our research team

and Dr. Donald Thurber, the creator of the D'Nealian handwriting system (Thurber, 1986). These rules guide the users through the game and help them develop their handwriting skills. Some of these rules deal with generic problems such as letter size, shapes, and number of strokes used to write the letter. Some rules are specific to individual letters. The letter "a" should be written with three distinct strokes (around-down, up, and monkey tail). The letter "g" needs to end with a fish hook that extends below the line. The letter "d" needs to reach to the top of the first solid line on the writing surface

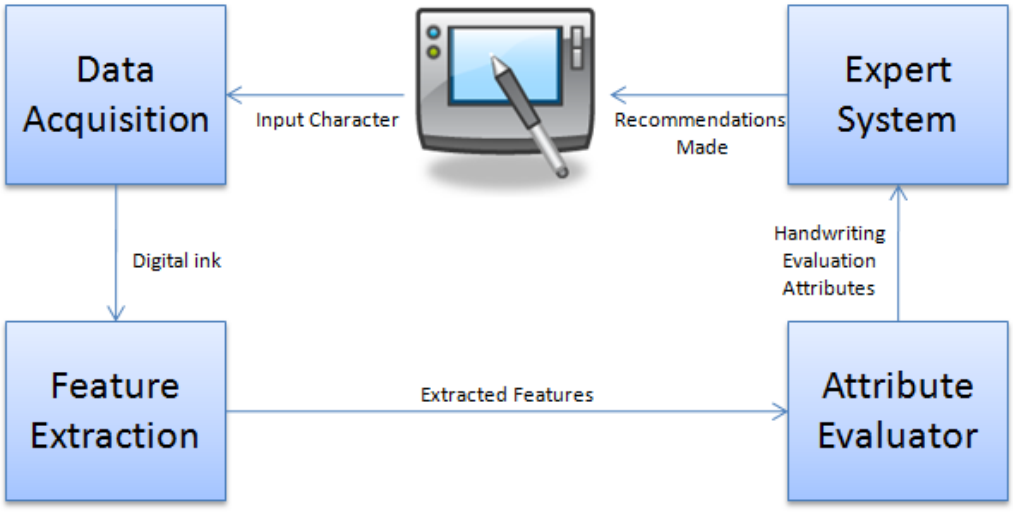

Figure 3 Tutoring System Architecture

. The remediation system allows the user to practice writing a particular letter stroke until he or she has done it correctly. Audio output and narrated animations are used to provide instructions and feedback to non-reading students. Part of the feedback given is one of a series of checkmarks. A blue checkmark shows that the letter has not been attempted. A green checkmark

means the letter is good, a yellow checkmark means the letter is not perfect and needs to be written again.

CURRENT STATUS

The authors created two working prototypes of the game environment. The first prototype was built using C# and Flash (Maxim, Patel, Martineau, & Schwartz, 2007). We had a great deal of trouble passing information gathered from the tablet PC ink system to flash and vice versa. In addition, using Flash slowed down the performance of the entire system. The second prototype was created using C#, XNA, and WPF. This proved to be a better platform for this particular application. It was relatively easy to interact with the ink system from C#. Using C# and WPF also made it very easy to transfer control in and out of the mini-games from the various letter worlds.

Sixteen senior and graduate level students, studying game design at the authors' institution, evaluated the prototype through play testing. These students used a five-point scale (1=poor, 5=excellent) to rate the prototype. Their average rating of the usability of the software is 4.9, the average reliability rating of the software is 4.9, the average entertainment value of the software is 4.5, and the quality of the game premise 4.7 (Maxim et. al., 2007).

Six elementary school teachers who teach writing skills to their students evaluated the second software prototype through active use of its intelligent tutoring capabilities. These teachers used a five-point scale (1=poor, 5=excellent) to rate this prototype. Their average rating of the ease of navigation for the software is 4.2, the average game playability rating of the software is 4.6, and the rating of the writing feedback module is 4.5.

FUTURE PLANS

The invetigators are seeking funding for the third year of this project and plan to develop a knowledcge acquistion component for this system. This will allow teachers to customize the content instruction and remediation activities provided by the system. In the current prorotype, each student error and remediation sequence is added to the system by a programmer.

The authors anitcipate formal testing with children once additional funding is obtained. Technical problems with the tablet PC, its development tools, and art asset creation delayed the project to the point that it is just now feasible to design experiments involving children in school settings.

In the near future, the system will be expanded to include animation for all twenty-six letters of the alphabet and several new mini-games will be created. Creating art assets for this system is a time consuming process. Students from a nearby art college are working to enhance the multimedia game library. The authors hope to use the expanded game to help students develop their initial reading skills.

ACKNOWLEDGEMENTS

The Henry Patton Center for Engineering Education and Practice at the University of Michigan-Dearborn provided the majority of the funding for this project. Additional funding was provided by Mandala Sciences and by the Verizon Foundation.

References

Aguilera, M. & Mendiz, A. (2003). Video Games and Education (Education in the Face of a "Parallel School"), *ACM Computers in Entertainment*, 1(1), 10-10.

Druin, A. (1999). Children as our technology design partners. *The Design of Children's Technology* (pp. 51–72). San Francisco, CA: Morgan Kaufmann.

Kearsley, G. (1987). *Artificial Intelligence and Instruction,* Reading, MA, Addison-Wesley.

Koile, K. & Singer, D. (2006). Improving learning in CS1 via tablet-PC-based in-class assessment. *Proceedings of the 2006 international Workshop on Computing Education Research*, 2(1), 119-126.

Maxim, B. R.; Patel, N.; Martineau, N.; & Schwartz, M. (2007, October), Work in Progress: Learning via Gaming – an Immersive Environment for Teaching Kids, In *Proceedings of 37th Annual Frontiers in Education Conference*, 37(1), T1B3-T1B4.

Michael, D. & Chen, S. (2006). *Serious Games: Games that Educate, Train, and Inform*. Indianapolis, IN: Thomson Course Technology.

Read, J. C., MacFarlane, S., & Gregory, P. (2004). Requirements for the design of a handwriting recognition based writing interface for children. *Proceedings of the 2004 Conference on interaction Design and Children: Building a Community*, 2004(1), 81-87.

Read, J. C., MacFarlane, S. J. and Casey, C. (2003). What's going on?: Discovering what Children understand about Handwriting Recognition Interfaces, In *Proceedings of the 2003 Conference on Interaction Design and Children,* 2003(1), 135-140.

Read, J. C., MacFarlane, S., and Casey, C. (2002). Oops! silly me! errors in a handwriting recognition-based text entry interface for children. *Proceedings of the Second Nordic Conference on Human-Computer Interaction*, 2(1), 35-40.

Siemer, J. and Angelides, M. C. (1995). Evaluating intelligent tutoring with gaming-simulations. *Proceedings of the 27th Conference on Winter Simulation*, 27(1), 1376-1383.

Squire, K.D. (2005). Changing the game: what happens when video games enter the classroom?. *Innovate*, Retrieved June 30, 2008 from: http://innovateonline.info/index.php?view=issue&id=9.

Thurber, D. N. (1986). *D'Nealian Home/School Activities – Manuscript Practice for Grades 1-3*, Good Year Books.

Wegner, E.(1987). Artificial *Intelligence and Tutoring Systems*, Los Altos, CA: Morgan Kaufman.

Willis, C.L., Muertschin, S.L. (2004). Tablet PC's as instructional tools or the pen is mightier than the board!. *Proceedings of the 5th Conference on Information Technology in Education*, 5(1), 153-159.