Educational Robots: Three Models for the Research of Learning Theories and Human-Robot Interaction

Sheila Tejada, Neil Traft, Malcolm Hutson, Harold Bufford, Matthew Dooner, Joshua Hanson, Anthony Radler, George Mauer

Department of Electrical Engineering & Computer Science
211 Stanley Thomas Hall, Tulane University, New Orleans, LA 70118
Dr. Sheila Tejada (tejada@eecs.tulane.edu)
Malcolm Hutson (lhutson@tulane.edu)
Neil Traft (ntraft@tulane.edu)
Harold Bufford (hbufford@tulane.edu)

Abstract

In the summer of 2006, the Tulane Electrical Engineering and Computer Science department teamed up with UNO graduate students in education and developmental psychology to conduct experiments in the education of young children using robotic agents. Students from the "AI Robotics" class used the Sony AIBO robotic dog to design three different educational systems geared primarily toward young physically and mentally challenged children. We consulted with UNO graduate students concerning pre-kindergarten education and visited a special education institution, The Chinchuba Institute. The Chinchuba Institute educates hearing- and speech-impaired children, along with children with mental disabilities. We returned with impressive results, designing three systems that focus on methods of communication with the student and usability for the student and teacher. We encountered many challenges in designing for the pre-K and K-12 classroom, such as making the robot interaction robust and intuitive to a child.

Section I - Introduction

The applicability of robotics in an educational environment is an emerging field of study which seeks to introduce robotics into the educational system in order to further the abilities of educators to instruct students. While research in human-computer interaction has delivered methods and approaches to the design and usability testing of computer usage, it has focused mainly on graphical user interfaces or multi-modal interfaces where vision is the main component. It has left untouched issues of physical robots as companions or assistants to communicate with computers [3]. This leaves a gap between the fields of robotics and education where the robot will be directly interacting with the students or educators to aid in instruction. The robot itself can serve several roles. First, the robots can act as an instructor, thus helping an educator to affectively convey material. Second, the robot can also be the subject of learning, and third the robot can act as an assistant to the instructor. Since the field of robotics spans many disciplines working with a robot can be a wonderful learning experience. Robots ground ethereal computation to palpable physicality, and as a result they have the potential to excite and inspire children and adults in a way that desktop computers cannot[17].

In the summer of 2006 Tulane University's Electrical Engineering and Computer Science department sought to find the best of both of these areas in a course entitled AI Robotics. The focus of the course was to learn about programming robots by integrating them into the educational system. The motivation for using the educational system came from the fact that there is always a shortage of good educational materials. In particular there is limited material with an 'AI flavor' [12]. Working with the educational system would give students access to AI and robotics at an early age. Actively involving teachers as well as prospective engineering students provides a two-fold approach for increasing the number of well-prepared students [15].

In order to accomplish this we collaborated with the Chinchuba Institute, whose mission is to educate children who are speech and language impaired. The goal was to begin the development of an application the Institute could use to aid in the education of their students.

The class selected the Sony AIBO robotics platform for use with this project. The AIBO's asthetical value, along with the Remote Framework SDK, provided by Sony, made a good fit for this project. The AI Robotics class would have to learn how to program the AIBOs and then apply this to education, thus giving this class multiple facets.

The first step the class took was to develop a wrapper class for the Remote Framework SDK, which is written in Microsoft Foundation C++. This allowed for bringing all the functionality of the SDK into a single class which would then be used for further development.

At this point the students were split into three groups, and each would select their own emphasis in education and develop an application accordingly. This paper is organized as follows: Section II discusses in detail the three projects; the AIBO Educators' Tactile Interface, the AIBO Quizzer, and the Cassandra Platform. Finally, Section III is Conclusions and Future Work.

Section II – Results of Three Project Teams

The AIBO Educators' Tactile Kinesthetic Interface for Intelligent Tutoring

The AIBO Educators' aim to explore the possibilities of using robots as teaching tools for young children. This project specifically targets the preschool and kindergarten age groups to teach vocabulary and memory through an integration audio, visual, and tactile feedback. More specifically, we would like to develop tools for the education of children with speech and language disorders. As opposed to the traditional connotation of *education robotics* where children are taught to use robots, our intention is to use the Sony AIBO ERS-7 robotic dog as a facilitator and motivation to an educational game.

While there is significant documentation that intelligent tutoring systems (ITS) can achieve significant gains over traditional teaching methods, little of this work has occurred with preschool aged children. Projects such as PACT [11] in geometry, ANDES [7] in physics, and SHERLOCK [14] in electronics demonstrate the potential of ITSs, but the methods employed by these systems would not be effective with preschoolers. This is in part due to the fact that these children have short and limited attention spans, requiring of constant stimulation and instant feedback. Additionally, young children typically respond better to tactile/kinesthetic feedback than to the visual and auditory methods employed by most successful ITSs. Thus, the researchers note a need for an intuitive tactile interface to facilitate the communication between the child and the robot that could be integrated with an ITS.

When evaluating any educational technology, it is important to determine if the tool is developmentally appropriate and consistent with how the child learns at his current developmental stage [16]. To incorporate a system as complex as a robot intelligent tutor, many steps need to be taken in the planning stage to insure that the final product will be useful, as its design criteria will differ dramatically from common intelligent tutoring systems and research robotics. Early childhood learning environments should thoughtfully incorporate a variety of ways for children to see, hear, touch, and connect with their surroundings. The "learning readiness" required to sit and learn in kindergarten is not taught, but is developed by numerous concrete interactions with the world [6]. Unfortunately, this is often ignored and rote skills such as worksheets and drills are forcefully substituted resulting in negative dispositions to such work as they continue school [10].

We began by developing a framework for communicating with and controlling the robot as a puppet. No readily available package provided the robustness and simplicity that we desired. Our solution, the myBO wrapper, encapsulates the routine methods of communicating wirelessly to AIBO, providing us full control over the robot via a wireless interface from a PC. Written in Microsoft VC++, the myBO wrapper uses the Sony Remote Framework SDK (RFW). We can stream generated speech or other audio to AIBO to imitate the dog speaking, or we can capture audio from AIBO's microphones. Using the Skitter [21] key frame motion editor, we can create and preview complex dance routines with the aid of a 3D simulation of the dog. The compiled output handles the details of the motor positions and timing, giving us full control of AIBO's 19 degrees of freedom for motion control as well as all of his LED displays. Lastly, we have the ability to poll the state of any of the robot's sensors.

With these educational guidelines and robot framework, our early system consisted of a simple word identification game administered by the robot. Responses were registered by a touchscreen interface as well as by touching salient areas on the robots body. Testing of the early prototype was conducted with children at the Chinchuba Institute's KidSpeak Learning Center in Gretna, Louisiana. Chinchuba's KidSpeak Learning Center provides early intervention services to children age two through five with a focus on teaching deaf children and children with speech and language disorders to speak intelligibly and to understand the speech of others. Their disorders can come from many different conditions. The ineffectiveness of the game and its interface was immediately apparent. The game did not hold the children's attentions. Encouraged to touch the robot, they were more interested in the robot as a toy. The children were notably forceful and destructive when interacting with our robots. The challenge of building a robot that is rugged, gentle, and interactive is a difficult one. Prone to breakage, the Sony AIBO could not be used as a source of tactile input for young children.

After a redesign, we retrofitted a Microsoft Xbox Dance Dance Revolution (DDR) style game pad to accept the tactile input from the children. The dance pad consists of a grid 3 columns wide by three rows tall of low-tech pressure sensors. The pads are inexpensive to replace and rugged enough to endure the use of energetic young children. Microsoft's Xbox game system is natively USB and freely available drivers allow the game pads to be read from Windows software as joysticks. An opaque fabric overlay was sewn that fits snugly over the pad. Over each pressure sensor, an image of a farm animal was printed onto the fabric.

In the new version of our simple game system, an AIBO named Woofie teaches the children vocabulary through a Simon-style memory game. Woofie stands on a table at eye level with the children while the pad is placed on the floor between the child and the robot. Employing an array of lifelike motions, Woofie speaks a sequence of animal names accompanied by their sounds, and the children repeat the sequence back to Woofie by stepping on the squares of the pad interface containing pictures of the corresponding animals(Figure 1). The sequence of sounds increases in length until a mistake is made. The AIBO provides immediate feedback to the gameplay by performing motions set to music that personify either a positive or negative reaction. The motions and dances become increasingly more complex to provide a system on increasing rewards as the child becomes more proficient. To maintain a series of appropriate responses, a simple algorithm is used to choose the motion and speech responses given in the game based on the skill level of the player.

Figure 1: Woofie and her modified DDR pad.

In our informal testing to date, it seems intuitive for young children to make a tactile connection with their answer. The gross motor movement of stepping is far less concentration intensive for a preschooler that the fine motor task of pressing a response on a touchscreen. Moreover, it is a full body task for child, resulting in a more engaging experience. An unforeseen consequence of the floor based interface is that it implicitly suggested to the child where to stand and decreased their tendency to walk forward to and handle the robot. Our test children appear very entertained by the robot's dances as well as the game itself.

The system, including the robot communication and the tactile interface, was designed with the capability to be easily ported to an ITS. In our future work, we plan to improve upon our tactile interface by constructing a step interface with dynamic content. This will be accomplished by floating clear acrylic panels over LCD screens. Responses will still be registered by stepping on the content. In combination, we will develop a full fledged ITS to pair with this new interface that will be capable of modifying the content.

AIBO Questioner & Quiz Builder

The primary motivation in designing traditional educational software to be run in tandem with robots has to do with the general attitude of interest that robots create in children. Their ability to engage a child's attention is empirically confirmed by many studies ([9], [13], [20]), whether that child tends to interact aggressively, gently, or not at all (some children were initially apprehensive of AIBO and hesitant to approach it). In one study, statistical results tabulated from 80 children under the age of 6 interacting individually with AIBO show a large range of attitudes from affection to apprehension to attempts at communication [9]. Also, the authors of [20] observed a need for robots' behavior to be adaptable or changeable over time if they are to sustain a child's attention over repeated exposure. This diversity of behavior presents the largest challenge to developing an *effective* robot tutoring platform (i.e. a platform that actually succeeds in imparting knowledge better than traditional methods).

We decided then to develop a question-and-answer program than ran scripted questions series that the teachers could create themselves. The AIBO and accompanying touch screen would provide the visual and auditory expression, while also acting as touch sensors. Pictures could be displayed while AIBO either played an action, or spoke a question out loud (Figure 2). The text of the question could also be displayed to the screen, or AIBO could play a pre-recorded WAV file. At each stage of the game AIBO could play actions and questions, rewarding children with a dance for correct questions, etc.

With these constraints we set our sights on developing a system that would provide an attention grabbing, simple, and robust environment for the children, while providing a powerful, and intuitive development tool for the teacher. After study of Learner-Centered Design and usability principles, we propose a versatile system for the AIBO platform that can be scripted largely by experts in education. The AIBO Quiz Builder and AIBO Questioner represent a promising model for integration into classrooms and use for research of HRI and e-learning theories in general.

In addition to *effectiveness, efficiency* is also identified by the International Standards Organization [8] as a general principle characterizing software *usability*. Our goal of developing a practical application involving robots which could be feasibly integrated into curriculum demands that our interface be as *usable* as possible, for the teacher as well as the student. As means to the end of integration into a classroom setting, and to encourage experimentation by experts in education, we have taken many principles of usability, User-Centered Design (UCD), and Learner-Centered Design (LCD) into consideration. Developing learning tools presents many unique challenges, given the large diversity of personalities in students ([9], [18], [20]) and their attitudes toward robots ([9], [20]), such as the students that we worked with at The Chinchuba Institute. A description of the interface and design methodology we have decided upon follows.

AIBO Questioner Interface: Firstly, the system interaction must be simple for the teacher as well as the child. As a result we based our AIBO Questioner interface on some important principles as dictated by study [1], namely: “Do not complicate the users’ task” and “Design for responsiveness” [1]. The levels of simplicity, and what they imply, vary drastically. The child’s response is both unrefined and unpredictable and is often mixed with other undesired reactions. The system interaction must account for these if it is to interact usefully with the child.

The following design criteria aids in developing such interactions: First, we have provided large displays and response regions, as per ISO 9241 guidelines [8], and the guidelines of LCD listed in study [2], particularly in the category of “presentation efficiency.” These response regions (or buttons) can include auditory, visual, and touch as forms of communication, and catch a child’s often misplaced responses. This inclusion of varied media also works toward goals of “hypermediality” as listed by [2] and, in general, takes advantage of our electronic platform for teaching as much as possible. The other idea behind such varied input regions is to aid in catching errors as well as provide continual re-prompting for the child. Pop-up notifications and a “Tip Box” are mechanisms to prevent and work through errors, a requirement of “efficiency of application’s proactivity” [2]. These utilities, along with Quiz Builder, make up easy-to-use platform tools that study [2] also suggests in an e-learning system.

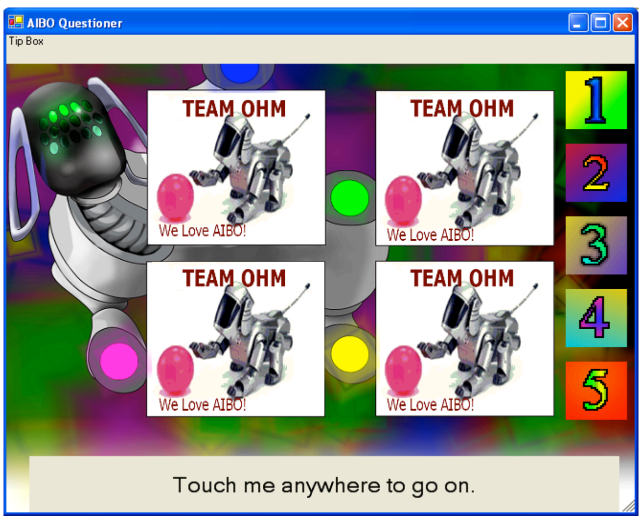

Figure 2: AIBO Quizzer touchscreen interface.

AIBO Quiz Builder Interface: As concerns the Quiz Builder interface, the teacher interaction must also be kept simple to accommodate for teachers’ wide range of experience with computers and technology, and to allow easy integration into the classroom. Study [4] stresses some issues that need to be addressed for the widespread integration of robots into education. These needs include: the need to raise awareness among educators, the need to produce low-cost robots, the need for a wide range of applications compatible with a single robot, and the need for teacher resources that integrate the robotic tools directly with curriculum material.

Obviously, all teachers will have varying degrees of technological literacy, and this has proven to be a difficulty for encouraging e-learning techniques, but this is not the only problem with bringing computers and robots into the classroom. Firstly, as noted by [19], experts in HCI and in education rarely commune or study each others’ work, and new theories of learning are absent in the vocabulary of most e-learning software developers. Secondly, programs that teachers don’t like simply won’t be used. Study [23] investigates reasons why teachers often neglect to integrate innovative and useful new software into their lesson plans. Some of the largest obstacles to integration were (1) difficulties in appropriating and setting up the proper computers, and (2) time consumed learning about the new software and altering their current curriculum. Despite an agreement between all 13 participants of the study that the software is convenient and useful, less than half of them actually used the program in class after a 2-year trial period. Thus, there is an extremely high need for these programs to be intuitive, usable, and ready-to-deploy, for even given all these traits it is less than guaranteed that an educator will actually make frequent use of it.

Luckily, in contrast to the child, the teacher can understand instructions, so providing help through tips and help menus, while making the interface easy to read will aid the novice user. To this end, the Quiz Builder interface (Figure 3) adopts the general “wizard” type of format currently in widespread use in computer interfaces, as well as following many principles of UCD especially the ISO guidelines in [8]. Also, the Questioner interface itself includes many informational cues to inform the teacher of the status of the program and AIBO in the form of pop-up notifications, as well as an unobtrusive “Tip Bar” that displays information on the functions of buttons as one rolls over them with one’s mouse.

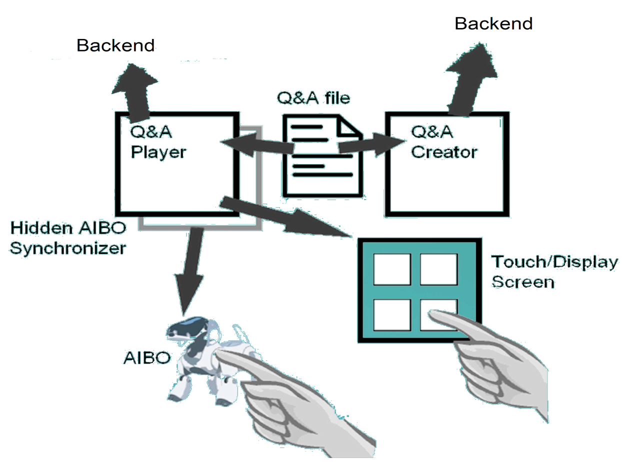

Figure 3: Components of the AIBO Quizzer system.

Making the program adaptable to a unique child’s needs adds complexity to a system that inherently needs simplicity. To keep it simple, the program needs to work even when the teacher does not want to specify the details, as observed in study [23]. As the teacher grows comfortable with the flow of the program, they can explore more details, expanding the potential of their creations. This makes the AIBO Questioner *scalable*, a property stressed by studies such as [19]. This, along with the rest of the aforementioned design criteria, all works to incorporate as many *efficiency* and *effectiveness* heuristics of usability as possible.

Some Conclusions: However, simply taking these usability heuristics into account is not yet sufficient for design of an electronic educational system. Many studies in Learner-Centered Design stress the relative youth of the field and lack of provable, statistical results. Most researchers in LCD agree that (1) development of educational software should include a period of Systematic Usability Evaluation (SUE) and multiple cycles of redesign and reanalysis ([2], [19], [18], [22], [23]) and (2) much more research and statistical surveying must be done on e-learning software to determine the true differences between electronic platforms and traditional classroom methods ([19], [18], [22]). In this respect, our project is unique from others in that it actually *opens* avenues for this direction of research.

The AIBO Quiz Builder wizard would, we hope, be usable by any expert in learning theories. One aim is at bridging the current gap of communication between learning theorists and HCI/HRI theorists. The other important use for such a customizable program is for research. Quiz Builder allows educators to create greatly varied applications for the robot system, and it is written on AIBO, a platform already widely used for cheap research. Putting this functionality in the hands of the teacher allows for much more experimentation and feedback in a field that greatly necessitates it, the field of Learner-Centered Design.

By providing an easy scripting tool that can be mastered by a wide range of educators, in addition to collecting research on students' performance, we hope to continue promoting and improving the interface by getting feedback from many teachers in a large variety of settings. Moreover, if a scriptable platform of this type were to become widely available, quizzes and tutorials compatible with AIBO could become as numerable as PowerPoint presentations on the internet. This, in addition to AIBO's relatively low price compared to other robotic platforms, would make it a prime candidate for easy integration into classrooms [23]. We believe that this kind of scriptable robotic platform is possibly the most promising solution to what [4] presents as the largest obstacles to robots in the classroom: the need for widespread, inexpensive robots with a wide range of applications.

The Cassandra "Robot Chat" Platform

The Cassandra platform was developed with the motivation that interactions with robots at any age can be a learning experience. The primary goal was to develop a system in a short period of time that people could interact with using natural language by using "off the shelf" components. This system could then be used to get kids and adults excited about robotics and spur their motivation to learn more about the subject.

During our visit to Chinchuba Institute's Kid Speak many of the children tried to speak with the robot. This was met with poor results. The current state of Speech Recognition Technology will not allow for the type of interaction these kids were looking for. Although, the software we developed for the AIBO included text-to-speak (TTS) functionality. During our time with the children we would type in a text string, and have the robot speak the string. This went over very well with the children. We even had success with several of them typing into the computer to hear the robot speak what they wanted it to.

The focus of the Cassandra platform was formed out of that interaction. The platform would use a text-based input, and then the robot would respond verbally to what was said. Not by simply repeating, but instead as if it had its own personality. In order to accomplish this goal a form of natural language processing would be necessary. After a short survey of many different natural language processing techniques the Artificial Intelligence Markup Language (AIML) was selected. The reference character ALICE [5] won the 2000 and 2001 Lobener prizes, and is used as a starting point for new developers. The core features of the language is the easy ability to create conversational agents through the process of pattern matching [5].

The Rebecca Implementation of AIML was chosen because it was developed in MFC, as was our wrapper class, which would add to the ease of implementation into the system. Once Rebecca was successfully integrated with our AIBO software, Cassandra began to take form.

Development in AIML of a custom personality was necessary. A demographic was developed for Cassandra and tags were added and modified to fit this model. This gave Cassandra the illusion of an understanding of its situation as a robot helping to educate the public about robots.

Once this was accomplished we had the backbone of the system, its ability to speak. Next we needed to further the believability of the robot as being an actual entity. A female TTS voice was selected as Cassandra was supposed to be a girl, and the attributes were modified slightly to give her a soft voice. Custom motions were developed in Skitter™ to allow Cassandra's mouth and body to move as she spoke a response.

Finally we did not want to platform to be a text input only. Though this was the driving force behind the platform we wanted small additions using image recognition to make the platform more real. Facial tracking was added so that if the robot was to see a face in its field of vision with would center the face in its field. This gave the illusion of looking at a person as it responded.

We also used several of the internal semantics to generate responses. The first was that of happy. If the robots sensors would be touched several time as if you were petting the dog the happy semantic would be returned. This was captured and one of several random responses would be spoken. These responses were meant to convey that the dog was enjoying being petted. The second semantic was that of exercise. When the robot would not move for a period of time the value of the exercise semantic would begin to rise. When it crossed certain thresholds (30 and 60) Cassandra would express that she is tired of standing in the same place, through a variety of random responses. When the value crossed 90 Cassandra would change her current posture, thus resetting the value back to zero.

The Cassandra platform is an excellent starting point for a variety of research in the field of HRI. It is primarily classified under the sociable partner paradigm; however the original development was to use that paradigm in an education context. Future work with Cassandra will focus on developing an emotional model for the platform so that when Cassandra responds it can have an emotional inflection based on a history of inputs to the system. Speech recognition will also be integrated into the platform, but the text based input will not be dropped, and a system of switching between the two when the speech recognition is not working correctly witll be developed as well.

Section III – Conclusions and Future Work

Despite different focuses in application design, the three project teams discovered many of the same challenges to creating educational robots and came to some similar conclusions. First and foremost each group experienced challenges in developing for the unpredictability of children, especially with the rather fragile AIBO. This was particularly challenging because of our work with the special needs children at the Chinchuba Institute, and caused each team to pay special attention to methods of robust communication and sensor input.

In addition we experienced difficulty with the usability of the Remote Framework SDK, which turned us on to the importance of the openness of a system and the coherence of its code. This led us to consider not only human-robot interaction, but also *programmer*-robot interaction as a major factor in the success of a robotic platform. This has become especially significant in the course of our particular study because of the difficulty with integrating new technology into classrooms and the need for robots with a widespread variety of educational software.

Another feature crucial in the design and integration of educational systems is their *usability*. Each of the teams took much effort to develop software that would be as simple for the teacher and as easy to install as possible.

We have proposed here the design of three practical, usable applications of robots in education. We believe these represent promising models for research in education as well as HRI, and we intend to continue surveying and working with teachers in New Orleans, collecting feedback, and expanding the designs of these systems.

References

[1] Adams, Julie A. (2002). Critical Considerations for Human-Robot Interface Development. *AAAI Fall Symposium: Human Robot Interaction Technical Report*, FS-02-03. Nov. 2002, p. 1-8.

[2] Ardito, C., De Marsico, M., Lanzilotti, R., Levialdi, S., Roselli, T., Rossano, V., Tersigni, M. (2004). *Proceedings of the Working Conference on Advanced Visual Interfaces*, Gallipoli, Italy. p. 80-84. New York, NY: ACM Press.

[3] Bartlett, B., Estivill-Castro, V., Seymon, S., Tourky, A. “Robots for Pre-orientation and Interaction of Toddlers and Preschoolers who are Blind” (2003) *Proc. of the Australasian Conference on Robotics & Automation.*

[4] Cooper, M., Keating, D., Harwin, W., and Dautenhahn, K. (1999). Robots in the Classroom - Tools for Accessible Education. In Buhler, C. and Knops, H., (Eds.), Proc. *AAATE Conference 1999, The 5th European Conference for the Advancement of Assistive Technology, November, Dusseldorf/Germany*, p. 448-452. IOS Press.

[5] Coursey, K. 2004, Living in CyN: Mating AIML and Cyc together with Program N, Available: http://www.cyc.com/doc/white_papers.

[6] Elkind, D. Miseducation: Preschoolers at risk. New York: W.W. Norton, 1987.

[7] Gertner, A. S., and VanLehn, K. ANDES: A Coached Problem-Solving Environment for Physics. In Intelligent Tutoring Systems: Fifth International Conference, ITS 2000, eds. G. Gautheier, C. Frasson, and K. VanLehn, 133–142. New York: Springer, 2000.

[8] ISO 9241: Ergonomics Requirements for Office Work with Visual Display Terminal (VDT), 1998.

[9] Kahn, P. H., Jr., Friedman, B., Perez-Granados, D. R., Freier, N. G. (2004, April). Robotic Pets in the Lives of Preschool Children. *CHI '04 extended abstracts on Human factors in computing systems*, Vienna, Austria. p. 1449-1452. New York, NY: ACM Press.

[10] Katz, L.G. (1993). *Dispositions as educational goals.* Urbana, IL: ERIC Clearinghouse on Elementary and Early Childhood Education. ED 363454. Available: http://www.ericdigest.org/1994/goals.htm.

[11] Koedinger, K. R., Anderson, J. R., Hadley, W. H., and Mark, M. A. Intelligent Tutoring Goes to School in the Big

City. Journal of Artificial Intelligence in Education, 1997. 8(1): 30–43.

[12] Krose, B., Bogaard, R., and Hietbrink, N. *Programming robots is fun: RoboCup jr*. 2000. In Proceedings of Belgium-Netherlands AI Conference 2000, 2000.

[13] Leinhardt, G. and Crowley, K. (2002). Objects of Learning, Objects of Talk: Changing Minds in Museums. To appear in: Paris, S. (ed). *Multiple Perspectives on Children's Object-Centered Learning.* Mahwah, NJ: Lawrence Erlbaum Associates.

[14] Lesgold, A.; Lajoie, S.; Bunzo, M.; and Eggan, G. SHERLOCK: A Coached Practice Environment for an Electronics Troubleshooting Job. In Computer-Assisted Instruction and Intelligent Tutoring Systems, eds. J. H. Larkin and R. W. Chabay, 201–238. Hillsdale, N.J.: Lawrence Erlbaum, 1992.

[15] Mountain, Jeffrey R., and Wells, Robert L. "Engaging K-12 Students and Teachers Using Competitive Interactive Design" *In Proceedings of Frontiers in Education 2002*, November 2002.

[16] National Association for the Education of Young Children. Technology and young children - ages 3-8 [Position statement]. Washington, D.C. 1996. Available: http://www.naeyc.org/about/positions/pdf/PSTECH98.PDF

[17] Nourbakhsh, I.R. Robots and Education in the classroom and in the museum: On the study of robots, and robots for study.In *Proceedings, Workshop for Personal Robotics for Education.* IEEE ICRA 2000.

[18] Poulymenakou, A., Vassilopoulou, K., and Zaharias, P. (2002). Designing On-line Learning Courses: Implications for Usability.*Scientific Journal on Applied Information Technology*, 2002(1).

[19] Poulymenakou, A., Vassilopoulou, K., and Zaharias, P. (2002). On-line Learning Courses: A Review and Usability Attributes. *World Conference on E-Learning in Corp., Govt., Health., & Higher Ed. 2002*(1), p. 1056-1062.

[20] Salter, T. Dautenhahn, K. Bockhorst, R. (2004, Sept.). *RO-MAN 2004: 13th IEEE International Workshop on Robot and Human Interactive Communication*, September 20-22, 2004, Kurashiki, Okayama Japan. p. 563-568. Piscataway, NJ: IEEE Press.

[21] Skitter website. Available: http://www.dogsbodynet.com/skitter.html

[22] Soloway, E., Jackson, S. L., Klein, J., Quintana, C., Reed, J., Spitulnik, J., Stratford, S. J., Studer, S., Jul, S., Eng, J., & Scala, N. (1996). Learning Theory in Practice: Case Studies of Learner-Centered Design. *ACM CHI '96 Human Factors in Computer Systems*, Vancouver, B.C. p. 189-196. New York, NY: ACM Press.

[23] Weston, Timothy J. (2005). Why Faculty Did—and Did Not—Integrate Instructional Software in Their Undergraduate Classrooms. *Innovative Higher Education, Vol. 30, No. 2*, 2005. p. 99-115.