DOI: https://doi.org/10.53555/nncse.v3i2.431 **Publication URL: http://nnpub.org/index.php/CSE/article/view/431**

LEARNING OPPORTUNITIES AND CHALLENGES OF SENSOR-ENABLED INTELLIGENT TUTORING SYSTEMS ON MOBILE COMPUTING DEVICES

Luis Vazquez[1]*, Michael Proctor[2]
*[1]*University of Central Florida, USA*
[2]*University of Central Florida, USA*
*[1]*Email: luis@knights.ucf.edu* [2]*michael.proctor@ucf.edu*

***Corresponding Author: -**
Email ID: luis@knights.ucf.edu

Abstract: -

Long established on stationary devices like traditional personal computers, sensor-enabled intelligent tutoring system (ITS) technology on mobile computing devices (e.g. tablets, smartphones) hope to deliver a personalized tutoring experience tailored to student affect and educational state anywhere, anytime. To achieve these goals, sensor-enabled ITS must overcome technology changes introduced by the mobility of devices such as tablets and smartphones. After a brief contextual presentation and identification of the sensor-enabled ITS technology & research gaps on mobile computing devices, this article discusses opportunities and challenges of the mobile-computing device and proposes a sensor-enabled ITS prototype for mobile devices.

Keywords: - *intelligent tutoring systems, mobile learning, sensors, tablets, smartphones*

I. INTRODUCTION

Analysts expect that 2016 shipments of mobile computing devices (e.g. tablets, smartphones) to swamp traditional PC shipments nearly ten-fold, 2,239 million to 333 million [1]. But are learning system application developers taking full advantage of the potential mass market? Intelligent Tutoring Systems (ITS) attempt to provide instruction to individual users without the aid of a human instructor [2, p. 2], but Wu et al [3, p. 823] meta-analysis of 164 mobile learning studies between 2003 and 2010 indicated that 58% of the ITS and related learning applications on mobile devices focus simply on improving 'effectiveness' of learning with no mention of the use of the sensors that exist on mobile devices. The clear inference is that mobile device ITS applications lag far behind other applications (i.e. text messaging, games, Skype, etc) in the use of sensors (i.e. cameras, gesture recognition, GPS, etc) in their applications to respond to the individual user. In further contrast, ITS on personal computers (PC) utilize mobile sensors far more than mobile devices do. Specifically, ITS on PC's seek to determine the level of student's mastery of curriculum [2, p. 2] often employing sensors and artificial intelligence to infer student mood, affect and engagement so that ITS instruction may be uniquely tailored to a given student [4, p. 5]. Clearly there is significant opportunity to enhance ITS on mobile devices by incorporating sensors into ITS mobile applications.

Incorporation of sensors into ITS on PC's using windows, icons, menus and pointers (WIMP) interfaces took decades to occur [5, p. 22], [6]–[8]. Whether incorporating sensors into ITS on mobile devices face a similar decades-long climb is yet to be determined. Besides the obvious market size and profitability questions, additional challenges confront incorporating sensors into ITS applications on mobile devices to include technology, organizational and security [9], interface design [10], learning theory [11]–[13], and user adoption challenges [14]. For example, converting existing ITS applications on PC based operating systems onto mobile device face big conversion challenges in terms of the interface. Apple's iOS and Android on devices such as the 2007 iPhone and 2010 iPad reduced WIMP keyboard dependencies [15, p. 5] by emphasizing touch-screen finger-based navigation, gesture control, and voice commands now common in post-WIMP, mobile interfaces [16, p. 64]. Moreover, mobile devices continue to rapidly evolve with natural language capable personal assistants such as Microsoft's Cortana [17] or Apple Siri [18] among the many emerging post-WIMP, mobile interface usability enhancements available now and in forth coming generations.

II. INTELLIGENT TUTORING SYSTEMS

A. Intelligent Tutoring System Architecture

Traditionally, intelligent tutor architecture possesses four distinct sections: domain model, student model, tutoring model, and the user-interface model shown in Fig 1 [19, pp. 60–64], [20, p. ii]. The domain model encompasses specific data taught; the student model captures student knowledge level and possibly user learning styles, mood, and other personality traits; the tutoring model contains algorithms to improve learning; and the user interface model receives input and displays output [20, p. ii]. These four distinct sections trace to systems developed before 1990, and subsequent intelligent tutoring systems possess a genealogy traced to those earlier systems [21, p. 5]

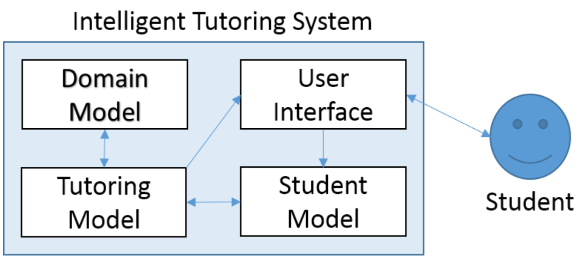

Fig 1 ITS Architecture [4]

B. Factors Motivating Adoption and Influencing Success of Intelligent Tutoring Systems

Factors motivating adoption and influencing the success of ITS may be grouped into five sub-categories.

1) **Tutoring Capability of Intelligent Tutoring Systems and its Impact on Student Learning, Affect and Engagement:** Bloom found one-to-one tutoring of a student by a human may result in a two standard deviation improvement over conventional methods [22, p. 4]. At a fundamental level, human instructors may pick up on emotional cues of their student and may adapt their teaching strategies accordingly [23, p. 127]. Thus, it's not surprising that some studies have shown that there is a strong relationship between affect and learning within an intelligent tutoring system and the emotional state of the student shouldn't be ignored [24, p. 4], [25, pp. 157–158]. Forbes-Riley and Rotaru [26, pp. 16–20] proposed that in the absence of affect a student may not experience any learning and may be disengaged. Laurel [27, p. 112] provides a popular characterization of engagement as, "a desirable, even essential, human response to computer-mediated activities" while O'Brien and Toms [28, p. 3] further breaks down engagement into a series of attributes including attention, novelty, interest, control, feedback and challenge. D'Mello et al. [29], D'Mello, Olney, Williams, & Hays [30] and Shanabrook, Arroyo, & Woolf, [31] indicate that the use of sensors to detect student's affect state may improve the effectiveness of an intelligent tutoring system [30, p. 389].
2) **Components in Learning Systems Interfaces***:* Interface components of intelligent tutors offer different capabilities of varying levels of suitability depending on instructional need. For example, a pen-based stylus allows free writing directly into the application and scores high in technology acceptance even though handwriting recognition is still a work in progress [32, p. 872]. In Anthony et al's [32, p. 881] handwriting ITS implementation, learners find entering

equations into the system to be more intuitive and are accomplished twice as fast as non-stylus keyboard and mouse entry means. Screen-based free-writing instructional domains, such as algebra equation solving ITS, may be suited for both the stylus interface, such as in Anthony et al's [32] algebra equation solving ITS, and post-WIMP interfaces as well.

3) **Cost Savings***:* From a cost perspective, long term cost savings of ITS as an alternative to human tutors also motivates ITS system development. Human tutors must be trained to tutor including interpersonal skills to reduce impatience with students and instructional strategies to lead the students towards their learning goal [33, p. 3]. In addition, human tutors need to maintain and update their skills given ever changing student profiles. Contrasting with human tutors, a sensor-based intelligent tutoring system that captures and responds to student affect may operate with the effectiveness of a well-trained tutor since they both react to the emotional state of the student [34, pp. 1–2]. Moreover, intelligent tutors at scale would be far more cost effective in the long-term to build and maintain than training and deploying legions of individual human tutors [35, pp. 213–214].

4) **Benefit, Convenience and Accessibility:** An intelligent tutor does not need to be a comprehensive solution to a problem in order to achieve beneficial results but may focus on teaching the "basics", leaving experienced humans to tutor advanced material [35, pp. 213–214]. Flexibility and accessibility to learning made possible by anytime, anywhere mobility provides additional motivation for mobile ITS research. Mobile intelligent tutoring systems would likely be more accessible than human tutors as well as deliver the right skills at the place and time right for the individual student.

5) **Different Target Audiences***:* While ITS advance, learning communities may vary in diffusion, and stage of technology solutions and acceptance [14, p. 178], [36, pp. 194–195], [37, p. 230]. As an example, Proctor and Marks [14, p. 178] found more widespread use of computer-based games for educational benefit among kindergarten through fifth grade communities then sixth through twelfth grade communities. Further, Proctor and Marks observed that educational game diffusion rates between the two communities had advanced at different rates and at the time of writing were in different Rogers Technology Adoption Curve stages [38]. Extrapolating the observed pattern of diffusion and stages of educational game to ITS, replacing human tutors with intelligent tutors in the desktop community and mobility community may be expected to be at different stages and diffuse at different rates.

C. Sensors in Intelligent Tutoring Systems

While not all methods of sensor monitoring used in stationary ITS WIMP devices are possible on post-WIMP devices, post-WIMP devices may offer a unique set of sensor capabilities not typically seen on stationary WIMP devices. ITS sensors monitoring may be passive and non-intrusive to active and intrusive or include combinations of sensors, each sensor combination with varying levels of usefulness [4, p. 5], [24], [29, pp. 3–4], [39, pp. 863–867], [40].

1) **Sensor Categories and Fusion of Sensors***:* Sensors tend to fit within three categories: cameras, haptic sensors, and body signal sensors. Cameras may observe and measure aspects of faces, eye gazes, and even emotion. Haptic sensors glean data related to user state by measuring touch and pressure, such as posture sensors, pressure mice and keyboards. Body signal sensors come in direct contact with the participant and passively measure body signals, such as skin conductance, heart rate, and electroencephalogram (EEG). The four most common types of sensors used in intelligent tutoring systems are cameras, haptic posture sensor, haptic pressure mouse/keyboard, and body skin conductance. Each sensor provides a specific interpretation of the environment with respect to its collected data. Traditional stationary devices including posture analysis seats, conductance bracelets, facial expression sensors, pressure mice, and blood pressure gauges [41, pp. 344–346]. Studies incorporating stationary ITS sensor devices attempt to use varying combinations in order to capture a larger set of user affects D'Mello [29, p. 9] notes that different sensors are better apt at reading different categories of emotion, such as face detection for confusion and delight, whereas posture sensors can more easily detect boredom and flow.

Through fusion and synergy of multiple sensor data streams the user state may be deduced. Over the past few years, analysis of that sensor data has evolved from offline analysis [24], to the use of judges and self-reporting users [29] and to the beginnings of real-time user affect analyses and reaction [25], [40]. Shen's [40] sensor fusion prototype highlighted the challenge of managing and maintaining the amount of data these sensors can deliver to the system. With the advancement of technology, San Diego [39] simplified data collection through collating multiple streams such as user gazes, screen interactions, and so forth. None the less, the increased data throughput forces having a sensible data management strategy that's relevant and useful to the researcher and to the system [39].

2) **Camera Sensor (observer):** Cameras provide an important direct view of the participant. That view may be utilized in multiple ways. For example, D'Mello's [29] used video recordings of the participants that experts used to determine the level of engagement. San Diego [39] also collected camera views of the participants but took advantage of eye tracking in order to track user's gazes. Recently, Bahreini [42] leveraged the idea of automated camera processing in an e-learning environment and developed a framework that interpreted a learner's emotion in real-time. Others have performed research in using cameras as a user-interface navigational tool instead of a mouse [43]. This type of technology could prove to be useful when in an environment where the use of a mouse would not be feasible [43].

3) **Keyboard, Posture Sensor, and Pressure Mouse (haptic):** Activity of a student at a keyboard may be used to infer levels mood and performance [4]. Posture sensors (seat sensors) built into the participant's seat do not require the participant to be aware of its existence. Woolf [24] used the Tekscan posture sensor system, where other researchers such as Mutlu [44] developed low-cost alternatives that produced comparable results in small settings. Both approaches are similar in design and try to calculate postures in order to make an inference on the user's state [24], [44]. Similar to posture sensors, pressure mice and keyboards attempt to infer the user's state by detecting the amount of pressure and force inflicted upon the mouse and keyboard [45]. In Arroyo's study [45], pressure mice were being used to determine the level of frustration, and Graesser [46] sought to incorporate pressure keyboards within the AutoTutor ITS.

4) **Skin Conductance Sensor (body signals):** Skin conductance sensors measure the electric variations on the participant's skin. These sensors are either worn on the hand [40] or the wrist [24]. The mobility of this particular sensor is dependent on its particular design. For example, Shen [40] obtains the skin conductance data from one of three sensors that are attached to the participant's fingers. Whereas Woolf [24] utilizes a glove to gather skin conductance. The premise of skin conductance sensors is to determine the level of "arousal" for the user [24]. Depending on the design of the skin conductance sensor, the sensor could be adapted to be portable, such as in Woolf's [24] study.

III. USE OF SENSORS IN MOBILE APPLICATIONS

Depending on the application, sensors may not be required to have an effective mobile ITS. For example, Sudoku ITS is a native Android app that teaches students how to play Sudoku without explicitly taking advantage of any sensor technologies [47, pp. 25–29]. Sudoku ITS tracks the length of user actions and the difficulty of the puzzle while maintaining a "user profile", which is simply a history of how many games the student has played. Once too much time has elapsed between actions, hints are automatically displayed for the user [47, pp. 29–30]. Despite application exceptions, sensors have the potential to enhance ITS effectiveness by monitoring and incorporating student affect in real-time into teaching strategies. A challenge is that the efficacy of a sensor driven intelligent tutoring in a mobile computing system will be confounded by many factors to include: availability of sensors, the dynamic nature and environmental factors associated with mobility, availability, access and limitations of interface components and their associated Application Programming Interfaces (API), acceptance of technology by users and communities, ever changing underlying device technologies, and the size and stability of markets sufficient to sustain mobile ITS developers.

In light of post-WIMP devices, availability of sensors for ITS design may change. For example, cameras are typically found on mobile devices and hence may be suitable for mobile ITS. Conversely using seat sensors in mobile ITS is problematic as seats may or may not be used in a mobile setting and if used, may be random. Skin conductance sensors may be applied to mobile applications as Bluetooth makes these devices feasible; however, previous research tying skin conductance readings to student level arousal associated with the ITS may be confounded by arousal readings related to physical activity associated to mobility itself [40, p. 183]. Tablets typically do not have mice and traditional keyboards undermining the use of mice and keyboard sensors in ITS mobility applications. Furthermore, the keyboard must be resting on a surface for a pressure surface to accurate provide feedback, which makes this sensor technology more suitable for stationary applications rather than mobile applications.

In terms of interface components, the proliferation of mobile devices has forced hardware manufacturers to add features and capabilities in attempts to differentiate themselves from others [48, p. 4]. Once a device manufacturer introduces a feature, other manufacturers quickly implement similar feature sets [49, Para. 1] until it becomes standard in all devices such as: frontfacing cameras, near field communication (NFC), infrared blasters, and front-facing speakers. These features let software developers create applications that weren't previously possible. For example, mobile applications that utilize NFC capabilities can now automate actions based upon what the NFC chip instructs the phone to do, such as change phone settings, report into social media websites, or download a business' contact information [50].

User communities running Android 4.3, iOS 5, or Windows 8, which now support Bluetooth SMART (or LE for Low Energy) allows for communication to a variety of different pulse and heart rate monitors [51]. Coupled with built-in, front facing cameras sufficient to the task, proposed intelligent tutoring systems may monitor eye retinas and facial expressions of the student. Pulse, heart rate, eye retinas, and facial expressions may be incorporated to better tailor ITS learning strategies to the individual. Mobile application, AttentiveLearner, shows early promise of utilizing a phone's camera to detect whether the student of a video learning class is still engaged in the course [52]. AttentiveLearner takes advantage of the LivePulse algorithm to implicitly sense the student's heart rate via the camera and auditory prompts to track whether the student is distracted [52]. Pham and Wang propose "providing intelligent learning interventions on mobile devices" as future work since the instructional material does not adapt to the student in real time [52].

Simple tutors like Soduku ITS, teaching a specific topic, such as the development of "ExploreIT!" [53, pp. 608–609] or "Math Tutor" [54, pp. 4–6] have limited success within their intended scope, but as Pham and Wang indicate there is a current lack of tailored use of sensors within ITS for post-WIMP mobile tablets that have the capability to monitor student affect and appropriately change their tutoring strategies.

IV. MOBILE INTELLIGENT TUTORING SYSTEM (MITS)

A. Mobile Application Design Considerations

The opportunity exists to take advantage of the potential of sensors available in tablets, such as Bluetooth heart rate monitors and the on-board camera to detect face and eye gazes, for enhancing tablet-based intelligent tutoring systems.

When designing learning applications or "apps" for a mobile device, the developers must choose between two implementation approaches: a web-based application or a native application [55, p. 1]. Web-based applications may be designed and developed with common web-based standards, such as HTML 5.0, CSS, and JavaScript [56, p. 77]. The idea behind a web-based application is that as long as the device has the appropriate web-browser that supports all these technologies, the user will be able to immediately interact with the service via the browser [55, p. 2]. Furthermore, software updates, such as bug fixes and newly developed functionality, may be updated instantly without any required user action [55, p. 2]. The other method of developing mobile applications is to develop them specifically for the native device. Native
applications follow the recommended look-and-feel of that specific operating system and may better interact with the resources on the device [57, p. 300]. Native apps need to be written using specific development environments including different language requirements and have specific (or recommended) guidelines that developers must follow [55, p. 2]. Therefore, unlike web-based applications, it takes a considerable amount of effort to learn the intricacies of device-specific development [58, p. 2]. When native apps are compared to their web-based counterparts, it may seem difficult to justify the extra expense and overhead, but there are inherent advantages in choosing native apps. With native apps, designers can design a user interface that is functional and aesthetically comfortable to use [57, p. 300]. These apps will match the 'look-and-feel' of other apps on the device and thus, will allow for a cohesive user experience [57, p. 300]. Furthermore, since native apps are directly communicating with the operating system, the app can take advantage of device-specific functionality such as creating shortcuts, use of the notification center, providing customized widgets, and so forth [57, p. 300]. Conversely, web-based applications are at the mercy of the standardization of Application Programming Interfaces (API) that allow for the web-based application to interact with the hardware resources via HTML5. As a result, native apps can directly access hardware resources that are currently available on the device as soon as the device is released [59, Para. 9]. Direct hardware access allows mobile ITS to be able to harness the power of hardware sensors for increased tutoring efficacy.

B. Design Rationale

With the few application-specific exceptions cited above, mobile ITS is clearly in the innovator stage of the Rogers Technology Adoption Curve. We propose that in order to best utilize the latest innovations in mobile technology, an ITS should be developed that leverages web-based and native app technologies. The Generalized Intelligent Framework for Tutoring (GIFT) is an existing open source intelligent tutoring framework system for which a sensor-based ITS may be developed [60, pp. 19–20]. Fig. **2** shows the proposed design of such an ITS that utilizes a native app which leverages camera and heart rate sensors in conjunction with the mobile application. GIFT is installed on the application server, which communicates and provides the intelligence for the native mobile app.

The proposed system would provide the power of an ITS, but allow for the flexibility of mobility. With the use of sensors, the ITS can also better adapt to the student and generate better tutor plans. A typical use case for this system would be for an undergraduate student, wishing to run through the tutoring app on her tablet, from within the classroom, park, or even a bus stop. The student would be able to resume her progress and each moment spent in the application allows for the system to better understand her mastery of the curriculum.

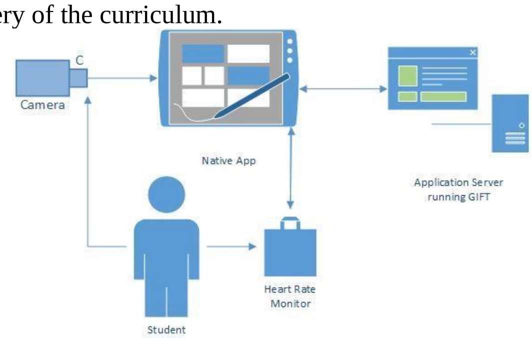

Fig. 2 Proposed Mobile ITS Design

C. mITS Prototype Implementation

In order to evolve the proof of concept into something tangible and demonstrable, a prototype was developed that incorporates the following functionality:

- Processing of facial detection to determine if user is facing the application
- Reading heart rate information from a heart rate monitor
- Integration and communication between an ITS application server (GIFT) and the mobile application

From a high-level design perspective, the mobile application mITS (Mobile Intelligent Tutoring System) prototype follows the design outlined by Fig. **2**, with a screenshot of the prototype depicted in Fig. **3**. The mobile application is currently running on a Nexus 7 (2013) Android tablet with version 5.1.1 of Android (Lollipop). Since the Nexus 7 sports a Qualcomm processor, the application is able to leverage the Snapdragon SDK which provides APIs that allow for facial feature detection. Once the user starts the application, the camera will begin to process its live video stream to detect faces and eyes. If the camera detects such features, then the user is notified via on screen text. The facial processing will continue throughout the life of the application.

Subsequently, the user pushes the "Start HR Sensor" button, which starts the heart rate monitor functionality, and the user is notified of the current reading once per second. The hardware required to support this would be any Bluetooth LE heart rate monitor and for this implementation the Scosche Rhythm+ was selected.

To connect to GIFT, the server must already be initialized and a course waiting for a client connection. From mITS, the user would push the "GIFT Connect" button, and interact with GIFT by pushing one of the three buttons. Each of these buttons have specific functions in the course provided by GIFT. For sake of simplicity, the prototype utilizes one of the sample courses which take three button inputs, where the third button informs GIFT to end the course. To show the message interaction between mITS and GIFT, the messages received from and sent to GIFT are presented to the user.

A finished implementation would hide these messages and process them in the background while domain-specific instructional material is presented to the user instead. Likewise, every second there will be a new heart rate reading sent by the heart rate monitor currently attached to the user's upper forearm. Both sets of sensor data is saved into timestamped files which can be used for further analysis. This data reporting infrastructure can be used by any sub component of the application.

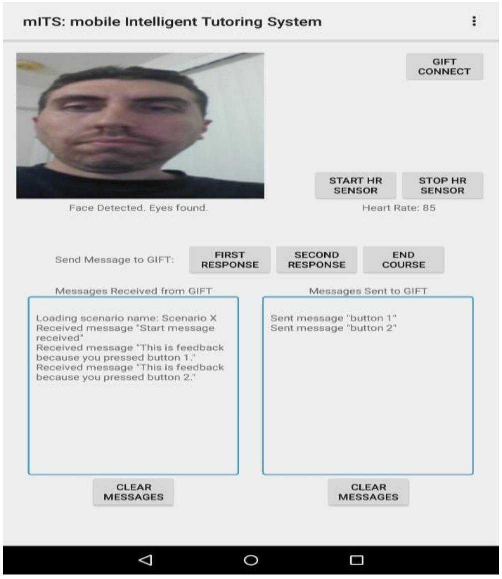

Fig. 3 mITS Prototype Screenshot

D. Future work and research

An immediate short-term goal is experimental involving conduct of a series of trials in order to obtain usability metrics on the prototype. The basic research question is, can the use of dedicated sensors as implemented by this prototype on a given mobile device improve an intelligent tutoring system's ability to tailor the curriculum in real-time? Challenges before one can enable a user to learn anywhere, anytime on a mobile device include incorporating the prototype application into curriculum and insuring application stability given intermittent internet connectivity. Analysis question include, do students that experience a sensor enhanced ITS application on a mobile device actually perform better than students using a ITS application on a mobile device that does not incorporate sensors? Other analysis questions include what are the differences in user acceptance and usability? The results of these trials will lead to other research questions which can be explored in future studies such as: posture implication, tablet orientation (accelerometer), external distractions, and so forth.

V. SUMMARY

To address both the lack of research on mobile ITS and how sensors can improve their effectiveness, the mITS prototype demonstrates that sensor technology may be incorporated into a ITS application on a mobile device. mITS successfully fuses two mobile sensors and provides a novel interface to GIFT courses. Follow-on research includes incorporation of mITS into curriculum and determining the level of contribution to student performance, acceptance, and usability. Clearly, application developers must consider trade-offs between both post-WIMP and convenience factors, as well as the challenges posed by a dynamic and ever evolving technology landscape. These challenges may be best addressed by leveraging a generalized framework for intelligent tutoring systems in order to support a mobile device. Future research efforts may expand upon the mITS prototype within a full course curriculum. Reusability of interfaces to sensors is essential to cost control and affordable customization. If successful, the opportunity for more widespread, cost effective, accessible, and educationally beneficial ITS's on mobile devices may begin to be realized.

REFERENCES

[1].Gartner, "Gartner Says Tablet Sales Continue to Be Slow in 2015," *Gartner*, 05-Jan-2015. [Online]. Available:

[2].http://www.gartner.com/newsroom/id/2954317.

[3].A. T. Corbett, K. R. Koedinger, and J. R. Anderson, "Intelligent tutoring systems," *Handb. Humancomputer Interact.*, pp. 849–874, 1997.

[4].W.-H. Wu, Y.-C. Jim Wu, C.-Y. Chen, H.-Y. Kao, C.-H. Lin, and S.-H. Huang, "Review of trends from mobile learning studies: A meta-analysis," *Comput. Educ.*, vol. 59, no. 2, pp. 817–827, Sep. 2012.

[5].Sottilare and Proctor, "Passively Classifying Student Mood and Performance within Intelligent Tutors.," *Educ. Technol. Soc.*, vol. 15, no. 2, pp. 101– 114, 2012.
[6].R. M. Baecker, "TIMELINES Themes in the early history of HCI—some unanswered questions," *interactions*, vol. 15, no. 2, pp. 22–27, 2008.
[7].T. L. Dimond, "Devices for reading handwritten characters," in *Papers and discussions presented at the December 9-13, 1957, eastern joint computer conference: Computers with deadlines to meet*, 1957, pp. 232–237.
[8].H. B. Goldberg, "Controller," 1117184, 17-Nov-1914.
[9].F. Lai, R. Luo, L. Zhang, X. Huang, and S. Rozelle, "Does computer-assisted learning improve learning outcomes? Evidence from a randomized experiment in migrant schools in Beijing," *Econ. Educ. Rev.*, vol. 47, pp. 34–48, Aug. 2015.
[10]. Y. C. Liu, Y.-A. Huang, and C. Lin, "Organizational factors' effects on the success of e-learning systems and organizational benefits: An empirical study in Taiwan," *Int. Rev. Res. Open Distrib. Learn.*, vol. 13, no. 4, pp. 130–151, 2012.
[11]. Y.-C. Chen, Y.-C. Lin, R. C. Yeh, and S.-J. Lou, "Examining Factors Affecting College Students' Intention to Use Web-Based Instruction Systems: Towards an Integrated Model.," *Turk. Online J. Educ. Technol.-TOJET*, vol. 12, no. 2, pp. 111–121, 2013.
[12]. J. S. Bruner, *Learning and thinking*, vol. 29. Harvard Educational Review, 1959. [12] J. S. Bruner, *The act of discovery*, vol. 31. Harvard Educational Review, 1961.
[13]. B. Dalgarno, G. Kennedy, and S. Bennett, "The impact of students' exploration strategies on discovery learning using computer-based simulations," *Educ. Media Int.*, vol. 51, no. 4, pp. 310–329, Oct. 2014.
[14]. M. D. Proctor and Y. Marks, "A survey of exemplar teachers' perceptions, use, and access of computer-based games and technology for classroom instruction," *Comput. Educ.*, vol. 62, pp. 171–180, Mar. 2013.
[15]. J. Schedeen, "The History of the Tablet PC," *IGN*, 01-Apr-2010.
[16]. A. van Dam, "Post-WIMP user interfaces," *Commun. ACM*, vol. 40, no. 2, pp. 63–67, Feb. 1997.
[17]. Microsoft, "Meet Cortana," 2014. [Online]. Available: http://www.windowsphone.com/en-us/how-to/wp8/cortana/meet-cortana.
[18]. "iOS - Siri - Apple," *Apple*, 2016. [Online]. Available: http://www.apple.com/ios/siri/.
[19]. M. A. Orey and W. A. Nelson, "Development principles for intelligent tutoring systems: Integrating cognitive theory into the development of computerbased instruction," *Educ. Technol. Res. Dev.*, vol. 41, no. 1, pp. 59–72, 1993.
[20]. R. Sottilare, A. Graesser, X. Hu, and H. Holden, Eds., *Design Recommendations for Intelligent Tutoring Systems*, vol. 1. 2013. [21] R. Nkambou, J. Bourdeau, and R. Mizoguchi, Eds., *Advances in intelligent tutoring systems*. Berlin: Springer, 2010.
[21]. B. S. Bloom, "The 2 sigma problem: The search for methods of group instruction as effective as one-to-one tutoring," *Educ. Res.*, vol. 13, no. 6, pp. 4– 16, 1984.
[22]. K. Porayska-Pomsta, M. Mavrikis, and H. Pain, "Diagnosing and acting on student affect: the tutor's perspective," *User Model. User-Adapt. Interact.*, vol. 18, no. 1–2, pp. 125–173, Feb. 2008.
[23]. B. Woolf, W. Burelson, and I. Arroyo, "Emotional intelligence for computer tutors," in *Workshop on Modeling and Scaffolding Affective Experiences to Impact Learning at 13th International Conference on Artificial Intelligence in Education*, 2007, pp. 6–15.
[24]. B. Woolf, W. Burleson, I. Arroyo, T. Dragon, D. Cooper, and R. Picard, "Affect-aware tutors: recognising and responding to student affect," *Int. J. Learn. Technol.*, vol. 4, no. 3, pp. 129–164, 2009.
[25]. K. Forbes-Riley, M. Rotaru, and D. J. Litman, "The relative impact of student affect on performance models in a spoken dialogue tutoring system," *User Model. User-Adapt. Interact.*, vol. 18, no. 1–2, pp. 11–43, 2008.
[26]. B. Laurel, *Computers as theatre*. Reading, Mass: Addison-Wesley Pub. Co, 1993.
[27]. H. L. O'Brien and E. G. Toms, "What is user engagement? A conceptual framework for defining user engagement with technology," *J. Am. Soc. Inf. Sci. Technol.*, vol. 59, no. 6, pp. 938–955, Apr. 2008.
[28]. S. D'Mello, A. Graesser, and R. Picard, "Toward an affect-sensitive AutoTutor," *Intelligent Systems, IEEE*, vol. 22, no. Intelligent Systems, IEEE, pp. 53–61, 2007.
[29]. S. D'Mello, A. Olney, C. Williams, and P. Hays, "Gaze tutor: A gaze-reactive intelligent tutoring system," *Int. J. Hum.-Comput. Stud.*, vol. 70, no. 5, pp. 377–398, May 2012.
[30]. D. H. Shanabrook, I. Arroyo, and B. P. Woolf, "Using touch as a predictor of effort: what the ipad can tell us about user affective state," in *User Modeling, Adaptation, and Personalization*, Springer, 2012, pp. 322–327.
[31]. L. Anthony, J. Yang, and K. R. Koedinger, "A paradigm for handwriting-based intelligent tutors," *Int. J. Hum.-Comput. Stud.*, vol. 70, no. 11, pp. 866– 887, Nov. 2012.
[32]. "Evidence That Tutoring Works." Department of Education, Washington, DC. Planning and Evaluation Service.; Corporation for National Service, Washington, DC, 2001.
[33]. R. W. Picard, S. Papert, W. Bender, B. Blumberg, C. Breazeal, D. Cavallo, T. Machover, M. Resnick, D. Roy, and C. Strohecker, "Affective learning— a manifesto," *BT Technol. J.*, vol. 22, no. 4, pp. 253–269, 2004.
[34]. K. VanLEHN, "The Relative Effectiveness of Human Tutoring, Intelligent Tutoring Systems, and Other Tutoring Systems," *Educ. Psychol.*, vol. 46, no. 4, pp. 197–221, Oct. 2011.
[35]. Proctor, T. Lucario, and C. Wiley, "Are Officers More Reticent of Games for Serious Training than Enlisted Soldiers?," *J. Def. Model. Simul. Appl. Methodol. Technol.*, vol. 5, no. 3, pp. 179–196, Jul. 2008.
[36]. H. K. Saleh, "Computer self-efficacy of university faculty in Lebanon," *Educ. Technol. Res. Dev.*, vol. 56, no. 2, pp. 229–240, Apr. 2008. [38] E. M. Rogers, *Diffusion of innovations*, 4th ed. New York: Free Press, 1995.

[37]. J. P. San Diego, J. C. Aczel, B. K. Hodgson, and E. Scanlon, “Digital approaches to researching learners’ computer interactions using gazes, actions, utterances and sketches,” *Educ. Technol. Res. Dev.*, vol. 60, no. 5, pp. 859–881, Oct. 2012.
[38]. L. Shen, M. Wang, and R. Shen, “Affective e-Learning: Using‘ Emotional’ Data to Improve Learning in Pervasive Learning Environment.,” *Educ. Technol. Soc.*, vol. 12, no. 2, pp. 176–189, 2009.
[39]. C. Frasson and P. Chalfoun, “Managing Learner’s Affective States in Intelligent Tutoring Systems,” in *Advances in Intelligent Tutoring Systems*, Springer, 2010, pp. 339–358.
[40]. K. Bahreini, R. Nadolski, and W. Westera, “Towards multimodal emotion recognition in e-learning environments,” *Interact. Learn. Environ.*, pp. 1–16, May 2014.
[41]. H. Ashtankar, P. Nagrale, Y. Timande, and I. Mandwi, “Mouse control using head movement,” *J. Int. Assoc. Adv. Technol. Sci.*, vol. 16, no. 4, p. 6, Mar. 2015.
[42]. B. Mutlu, A. Krause, J. Forlizzi, C. Guestrin, and J. Hodgins, “Robust, low-cost, non-intrusive sensing and recognition of seated postures,” in *Proceedings of the 20th annual ACM symposium on User interface software and technology*, 2007, pp. 149–158.
[43]. I. Arroyo, D. G. Cooper, W. Burleson, B. P. Woolf, K. Muldner, and R. Christopherson, “Emotion Sensors Go To School.,” in *AIED*, 2009, vol. 200, pp. 17–24.
[44]. A. C. Graesser, P. Chipman, B. C. Haynes, and A. Olney, “AutoTutor: An Intelligent Tutoring System With Mixed-Initiative Dialogue,” *IEEE Trans. Educ.*, vol. 48, no. 4, pp. 612–618, Nov. 2005.
[45]. Y. Zhuang and S. K. Cheung, “A Case Review on the Implementation of Intelligent Tutoring Systems on Mobile Devices,” in *Hybrid Learning and Continuing Education*, Springer, 2013, pp. 21–32.
[46]. ATKearney, “The Mobile Economy 2013,” 2013. [Online]. Available:
[47]. http://www.gsmamobileeconomy.com/GSMA%20Mobile%20Economy%202013.pdf.
[48]. N. Ekekwe, “When You Can’t Innovate, Copy,” *Harvard Business Review*, 24-May-2012. [Online]. Available: http://blogs.hbr.org/2012/05/when-youcant-innovate-copy/.
[49]. D. McFerran, “How to use NFC tags with your Android mobile phone,” *cNet*, 25-Apr-2012. [Online]. Available: http://www.cnet.com/how-to/how-touse-nfc-tags-with-your-android-mobile-phone/.
[50]. M. Casserly, “Personal health tech: can your iPhone or tablet replace your doctor?,” *TechAdvisor*, 29-Apr-2014. [Online]. Available: http://www.pcadvisor.co.uk/features/digital-home/3514084/how-monitor-your-health-with-gadgets/.
[51]. P. Pham and J. Wang, “AttentiveLearner: Improving Mobile MOOC Learning via Implicit Heart Rate Tracking,” in *Artificial Intelligence in Education*, vol. 9112, C. Conati, N. Heffernan, A. Mitrovic, and M. F. Verdejo, Eds. Cham: Springer International Publishing, 2015, pp. 367–376.
[52]. S. B. Blessing, J. S. Skowronek, and A.-A. Quintana, “ExploreIT! An Adaptive Tutor in an Informal Learning Environment,” in *Artificial Intelligence in Education*, 2013, pp. 607–610.
[53]. Z. Masood and R. Hoda, “Math Tutor: An Interactive Android-Based Numeracy Application for Primary Education,” presented at the Australasian User Interface Conference, 2014, vol. 150, pp. 3–10.
[54]. J. Stark, *Building iPhone apps with HTML, CSS, and JavaScript*, 1st ed. Beijing ; Sebastopol, CA: O’Reilly, 2010.
[55]. D. Gavalas and D. Economou, “Development Platforms for Mobile Applications: Status and Trends,” *IEEE Softw.*, vol. 28, no. 1, pp. 77–86, Jan. 2011.
[56]. H. Heitkötter, S. Hanschke, and T. A. Majchrzak, “Evaluating cross-platform development approaches for mobile applications,” in *Web Information Systems and Technologies*, Springer, 2013, pp. 120–138.
[57]. T. Wasserman, “Software engineering issues for mobile application development,” *FoSER 2010*, 2010.
[58]. M. Mahemoff, “HTML5 vs Native: The Mobile App Debate,” *HTML5 Rocks*, 03-Jun-2011. [Online]. Available: http://www.html5rocks.com/en/mobile/nativedebate/.
[59]. Sottilare, “Considerations in the Development of an Ontology for a Generalized Intelligent Framework for Tutoring,” in *International Defense and Homeland Security Simulation Workshop 2012*, 2012.