“It Wasn’t As Bad As I Thought”: Exploring K-12 Students’ Experiences with Real-Time and Pre-Recorded Physiological Data

Author1 Author1
anon1@university.edu
Author1Institution
Author1Location, Author1Location

Author2 Author2
anon2@university.edu
Author2Institution
Author2Location, Author2Location

Author3 Author3
anon3@university.edu
Author3Institution
Author3Location, Author3Location

Author4 Author4
anon4@university.edu
Author4Institution
Author4Location, Author4Location

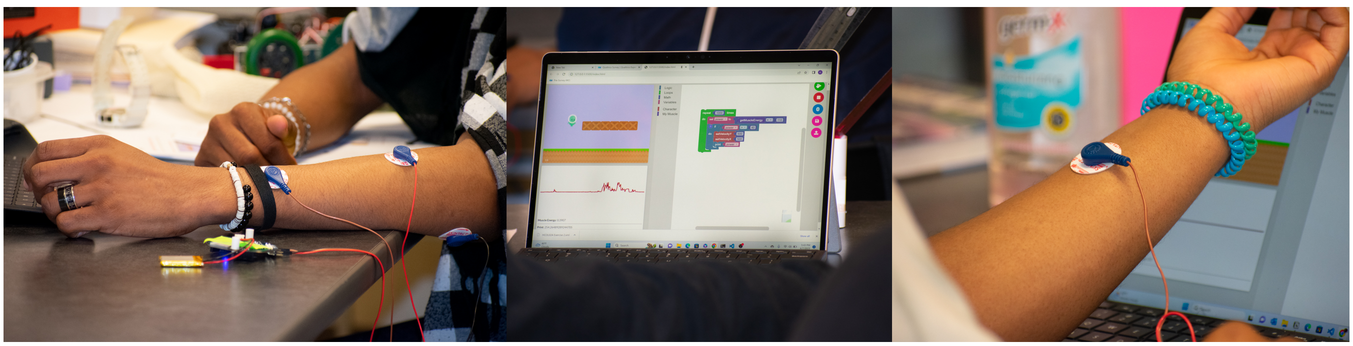

Figure 1: Students building simple apps that leverage physiological data

Abstract

Physiological sensing technologies are becoming increasingly accessible, prompting educators and researchers to explore hands-on approaches for introducing K–12 students to physiological computing. While prior efforts have engaged students using sensor-based activities, few empirical studies have examined how outcomes differ between using real-time physiological hardware and pre-recorded biosignal data. This paper presents findings from a study in which students engaged in two sets of activities: one activity using a consumer-grade hardware physiological sensor (i.e., OpenBCI Ganglion) and the other using pre-recorded physiological data (i.e. muscle activity) to build simple applications. Our observations revealed similar outcomes and student engagement across both approaches. Notably, students reported increases in self-efficacy and confidence regardless of whether they worked with real-time or pre-recorded data. These findings suggest that hardware-free implementations may offer similar benefits when teaching time-series data and signal processing concepts. We discuss the implications of these findings and reflect on the benefits and constraints of incorporating physiological sensors in high school computing classrooms.

CCS Concepts

• **Applied computing → Interactive learning environments**; • **Social and professional topics → Computing education**.

Keywords

Physiological Computing, Computer Science Education, Block-Based Programming, Electromyography (EMG)

ACM Reference Format:
Author1 Author1, Author2 Author2, Author3 Author3, and Author4 Author4. 2018. “It Wasn’t As Bad As I Thought”: Exploring K-12 Students’ Experiences with Real-Time and Pre-Recorded Physiological Data. In *Proceedings of Make sure to enter the correct conference title from your rights confirmation email (Conference acronym ’XX).* ACM, New York, NY, USA, 7 pages. https://doi.org/XXXXXXX.XXXXXXX

1 Introduction

Sensor-based technologies are becoming increasingly ubiquitous, with examples found in homes, vehicles, and a wide range of commercial settings. In recent years, wearable devices capable of measuring physiological data have also gained popularity, appearing

ACM ISBN 978-1-4503-XXXX-X/2018/06
https://doi.org/XXXXXXX.XXXXXXX

in smartwatches[1], glasses[2], and earphones[3]. As physiological sensing becomes more integrated into everyday life, there's a growing need to help young people engage with the technology not just as consumers, but as creators.

Researchers have recently explored various approaches to addressing this gap by designing tools that offer students hands-on experience with physiological sensing technologies [15, 18]. Educational tools have commonly focused on data from the brain (electroencephalogram – EEG), heart (electrocardiogram – ECG/EKG), and muscles (electromyogram – EMG) [12]. Prior relevant work has primarily examined how students engage with tools that use real-time physiological sensor data to develop applications or games that respond to changes in physiological state [19, 26]. However, there is limited research comparing the use of real-time data with pre-recorded physiological data in educational contexts.

To address this gap, we present a preliminary study that compares students' experiences in STEM activities using real-time (sensor-based) versus pre-recorded (non-sensor-based) physiological data.

2 Related Works

Over the past several decades, researchers have investigated various ways to integrate physiological sensing into educational contexts. One of the earliest approaches involved using physiological data to assess students' cognitive states during learning activities. For instance, prior studies have examined how physiological signals can be used to measure student engagement [1, 13, 17, 22, 29, 38, 44, 46, 47] and self-regulation [2, 3, 40]. In contrast, more recent research has focused on using physiological sensors to support hands-on learning experiences [19, 34].

Previous research suggests that engaging with authentic sensor data experiences [28] can enhance students' data literacy. As skills related to artificial intelligence, machine learning, and intelligent sensing grow increasingly important, understanding foundational concepts such as data preprocessing becomes even more critical. Several studies have examined how students engage with pre-recorded data in data science activities. For example, researchers have used pre-recorded environmental data [30, 39], fitness data [7, 32, 33], geological data [27], audio [11, 35] and simulation outputs [36, 37] to introduce key data science concepts. Researchers have also investigated the use of sensor-generated data in educational activities. This work spans a range of topics, including biology [20, 21], physiology [19, 34], sports [23, 31, 48], dance [5, 24], e-textiles [25], mobile computing [10], and data from physical environments [6, 14, 16, 41, 45].

Despite these advances, few studies have directly compared how students engage with real-time sensor data versus pre-recorded physiological data in educational programming contexts. This is a critical gap, especially for educators who must weigh cost, accessibility, and learning outcomes when deciding whether to include physical hardware in the classroom. To address this gap, this study compared the experiences of K-12 students during real-time and non-real-time physiological computing activities using a block-based programming interface. We contribute preliminary insights on how each modality affects students' self-efficacy, programming confidence, and motivation. Additionally, this work extends our current understanding of students' general responses to creating programs that leverage muscle activity data.

[1]https://www.fitbit.com
[2]https://www.ray-ban.com/usa/ray-ban-meta-ai-glasses
[3]https://www.emotiv.com/products/mn8

3 Methods

We conducted a user study to examine students' experiences with sensor-based and non-sensor-based approaches to learning programming and physiological computing. This study aims to address the following research questions. **(RQ1):** To what extent does working with **physiological-sensors** affect student's **self-efficacy** compared to **non-sensor approaches**? **(RQ2):** To what extent does working with **physiological-sensors** affect student's **programming confidence** compared to **non-sensor** approaches? **(RQ3):** What is the impact of using **physiological sensors** on students' motivation compared to a **non-sensor-based** method?

3.1 Study Procedures

The X Institutional Review Board (IRB) approved this study before recruitment. Our group worked with local high school educators to explore our research questions. A total of 37 students (M=17, F=19, gender fluid=1) participated in the studies. Participants had an average age of 16.3 years (SD = 0.83), ranging from 15 to 18 years old. We employed a within-subjects design and counterbalanced the condition order to mitigate potential order effects. Participants engaged in two activity conditions: (1) a non-sensor condition utilizing pre-recorded physiological data and (2) a sensor-based condition involving real-time data collection (Figure 1). Each session began with a pre-survey and a brief overview of the corresponding activity. The primary distinction between the two conditions was whether students worked with live physiological signals captured via sensors or analyzed previously recorded data.

3.1.1 Non-Sensor Condition (External and Pre-recorded Data). During the non-sensor condition, students interacted with a custom application featuring an interactive visual programming environment designed to support educational activities using pre-recorded physiological data (Figure 2). The application was inspired by block-based programming tools such as Scratch and Snap. A central feature of the environment was the ***data block category***, which allowed students to experiment with common signal processing techniques, including mean removal, absolute value computation, and band-pass filtering (Figure 3). These operations are frequently used in the analysis of time-series data such as audio and physiological signals. The environment also included a ***get signal block*** that provided access to pre-recorded electromyography (EMG) data reflecting variations in arm muscle activity. To support data exploration, a ***plot block*** was included to dynamically update an interactive graph. In addition to supporting pre-recorded physiological data, the tool used in this condition also featured external data integrations, including links to music (Spotify Charts), sports (ScoreStream), and weather (Weather.gov). These additions were informed by initial student feedback and aimed to increase engagement by incorporating familiar, real-world data sources. Students learned to capture and visualize data from these sources using blocks such as ***create***

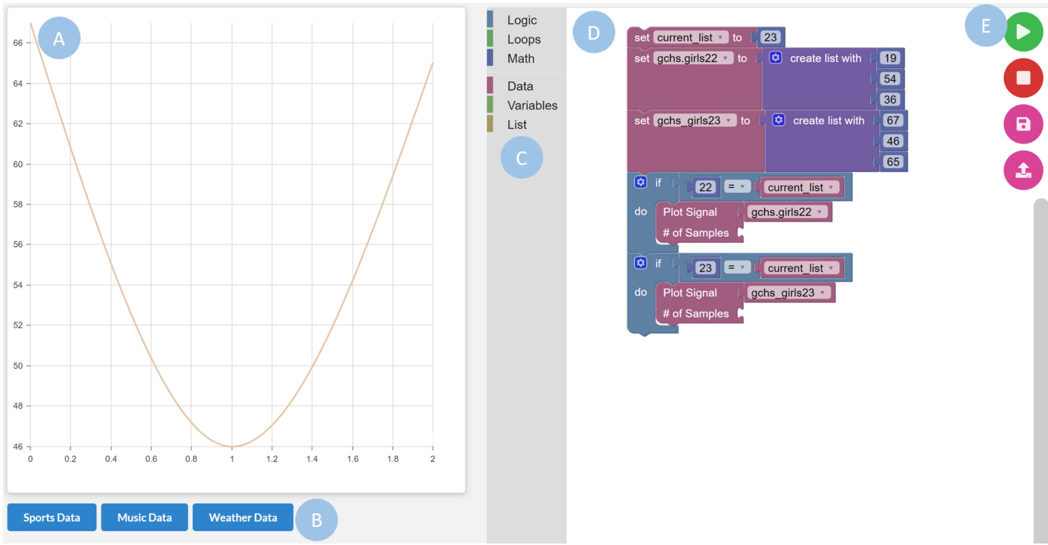

Figure 2: Example program created by student that plots different list data. (A) Interactive Graph (B) External Data Links (C) Block Toolbox (D) Scripting Pane (E) Action Buttons. (Non-Sensor)

list with, ***set variable***, and ***plot***. The activity began with students plotting simple, custom datasets derived from the external sources. Following this, they completed an exercise focused on visualizing and processing raw EMG data. This session lasted approximately 60 minutes.

3.1.2 Sensor Condition (Real-Time Data). To compare sensor-based and non-sensor-based approaches, we developed a custom tool that provided students with hands-on experience using physiological sensors (Figure 4). This tool was designed to closely mirror the application used in the non-sensor condition, with key enhancements to support real-time sensor interaction. Specifically, the sensor-based tool included: (1) a ***real-time data block*** that updates in real-time with electromyography (EMG) muscle activity data, (2) a ***game environment*** featuring a character, obstacles, and collectibles, (3) ***real-time visual signal feedback***, and (4) a menu option to establish a ***Bluetooth connection*** between the sensor device and a computer. To reduce potential confounding variables, all other design elements were kept consistent with the non-sensor version.

Real-time functionality was implemented using the Web Bluetooth API[4]. We used OpenBCI’s Ganglion device to capture real-time EMG data[5]. Incoming EMG signals were rectified and filtered using standard procedures from prior work on muscle-computer interfaces, including a 4th-order Butterworth low-pass filter with a 3 Hz cutoff frequency [43]. During the sensor-based activity, students were guided through exercises in which they built simple interactive games that allowed them to control a character using muscle activity from their arms. Each session lasted approximately 60 minutes.

3.2 Data Collection

This study examined students’ **self-efficacy**, **programming confidence**, **motivation**, and **user experience** following each condition. To support our analysis, we employed the following data collection methods.

3.2.1 Self-Efficacy. Self-efficacy scores were collected using pre- and post-surveys based on a previously validated questionnaire [8], with modifications to tailor the items specifically to physiological computing technologies. Responses were measured on a seven-point Likert scale.

3.2.2 Programming Confidence. The programming confidence questions ask students how confident they were with various programming concepts such as (Q1) variables, (Q2) sequence, (Q3) logic structures (IF), (Q4) methods, (Q5) lists, and (Q6) encapsulation. Responses were measured on a seven-point Likert scale.

3.2.3 Motivation. The Intrinsic Motivation Inventory instrument was used to measure interest, competence, and effort [42]. This measure was only collected during the post-survey. We used the Intrinsic Motivation Inventory instrument to measure interest (Q1-Q7), competence (Q8-Q13), and effort/importance (Q14-Q18).

3.2.4 Usability. Usability for each condition was assessed using the System Usability Scale (SUS) [4], a widely used and validated standardized instrument. Participants completed the SUS as part of the post-survey, following the recommended format and scoring guidelines.

3.2.5 Open-Ended Questions. Open-ended survey questions were included to gain deeper insights into students’ experiences with each condition. These questions were designed to elicit reflections on both the creative aspects and challenges of the activities. For example, students were asked to describe what they perceived to be an **interesting** aspect of their project and how they implemented it

[4]https://developer.mozilla.org/en-US/docs/Web/API/Web_Bluetooth_API
[5]https://shop.openbci.com/products/biosensing-starter-bundle

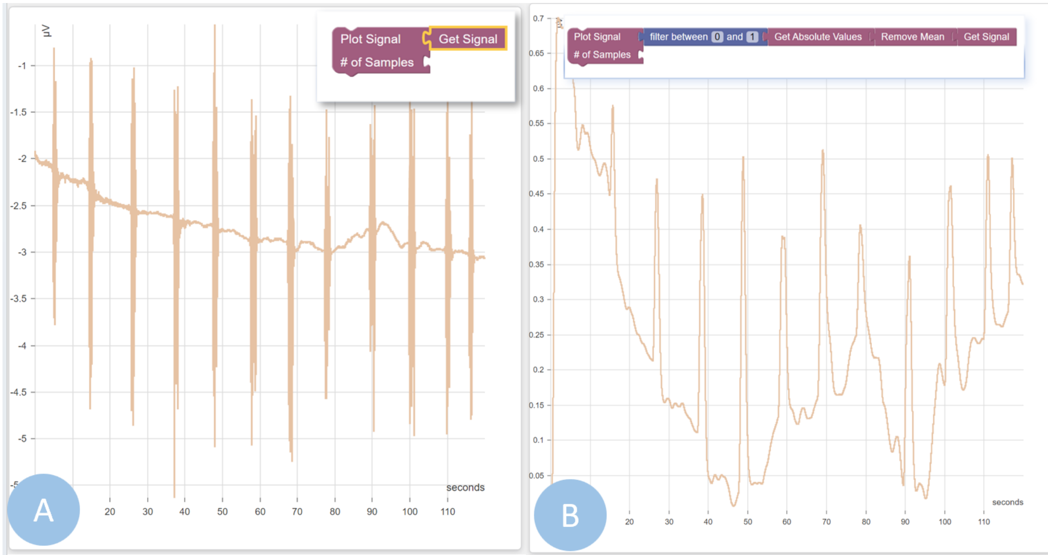

Figure 3: (A) Plotting raw pre-recorded EMG signal (B) Using data blocks to process raw EMG data. (Non-Sensor/Pre-Recorded)

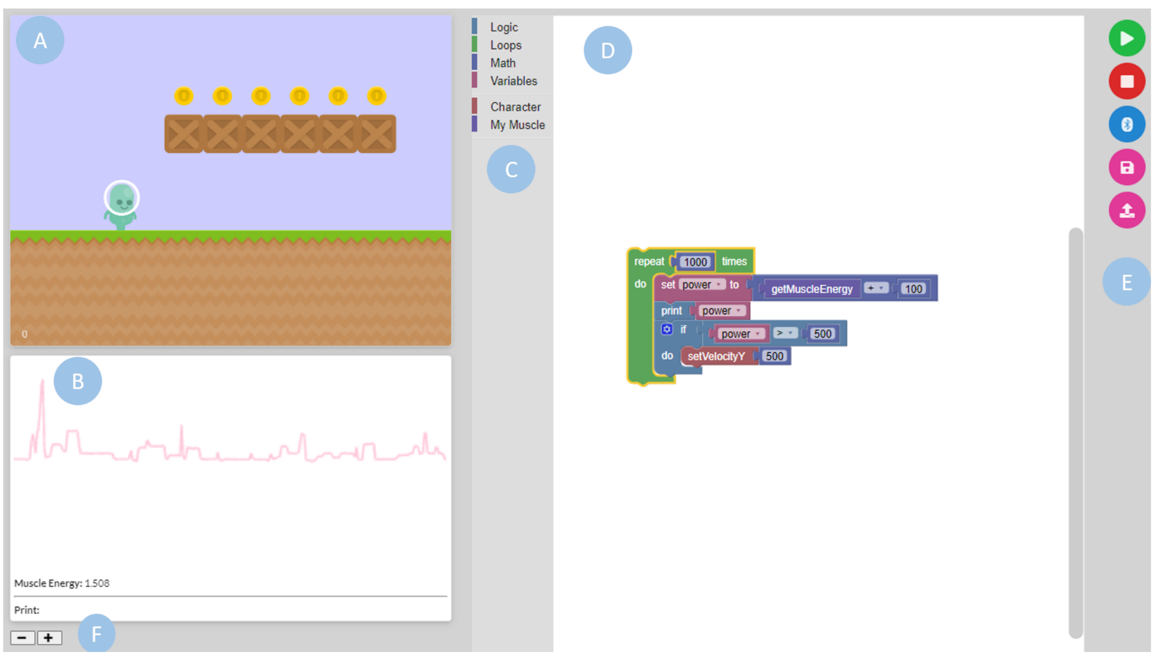

Figure 4: (Real-Time Sensor) Example program created by a student that makes a character jump when the arm muscle is activated. (A) Game Feedback (B) Visual signal feedback. (C) Block Toolbox (D) Scripting Pane (E) Action Buttons. (F) Signal zoom controls.

("Write down an interesting thing that you did in your project and how you made it happen"), as well as to identify any **challenges** they encountered ("What things were hard when making your project?").

3.3 Data Analysis

3.3.1 Quantitative Data. All quantitative data were analyzed using R. The Wilcoxon signed-rank test was used to evaluate changes in self-efficacy scores between pre- and post-surveys, as well as to compare responses between the sensor and non-sensor conditions. Data normality was assessed using both visual inspection and the Shapiro-Wilk test. Additionally, skewness and kurtosis statistics

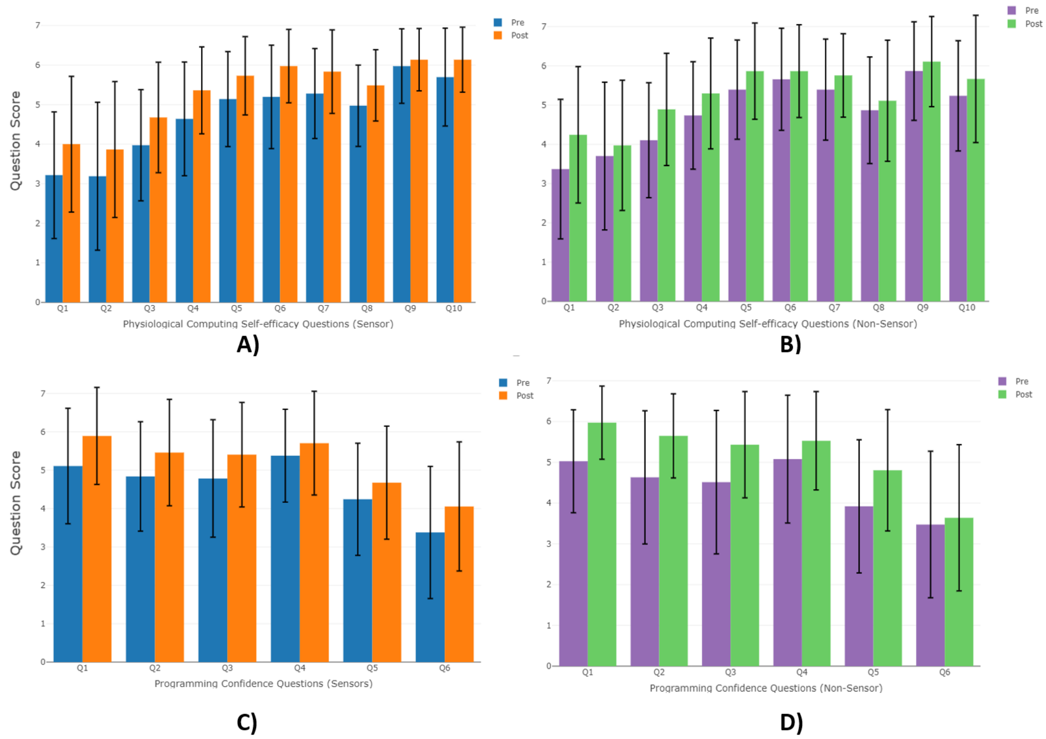

Figure 5: Self-efficacy scores for the Sensor (A) and Non-Sensor (B) conditions. Programming confidence scores for the Sensor (C) and Non-Sensor (D) conditions.

were examined to further evaluate the distributional characteristics of the data.

3.3.2 Qualitative Data. Analysis of the open-ended survey responses was informed by Grounded Theory [9]. A deductive coding approach was used to identify recurring topics and themes, helping to reduce potential bias during interpretation. The resulting codes were organized into broader thematic categories that reflected the core insights shared by participants.

4 Results

4.1 Self-Efficacy

We did not find a significant difference (p>0.05) in self-efficacy between the non-sensor (M = 52.84, SD = 9.9) and sensor (M = 52.8, SD = 7.6) conditions (Figure 5). **However, we did observe a significant difference in self-efficacy between the pre and post-survey responses for both the sensor (Pre(M = 46.2, SD = 9.7), Post(M = 52.8, SD = 7.6), p = 0.034) and non-sensor condition (Pre(M = 48.2, SD = 9.3), Post(M = 52.84, SD = 9.9), p = 0.046).**

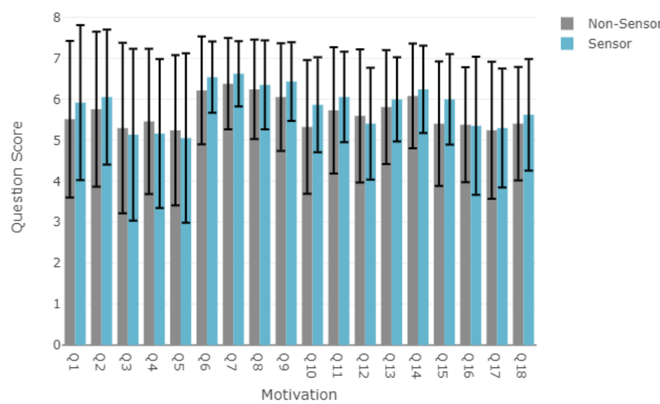

Figure 6: Physiological Computing Self-Efficacy (Non-Sensor v. Sensor)

4.2 Programming Confidence

Surveys administered during the study assessed changes in students’ programming confidence. Overall confidence was calculated as the average score across six items targeting core programming concepts, including variables, sequences, logic structures, functions, lists, and encapsulation. We did not find a significant difference (p>0.05) in programming confidence between the non-sensor (M = 30.6, SD = 5.8) and sensor (M = 31.1, SD = 6.8) conditions.

4.3 Interest, Motivation, and Usability

We did not find a significant difference (p>0.05) in interest between the sensor (M = 43.7, SD = 6.05) and non-sensor (M = 41.4, SD = 7.2) conditions (Figure 6). Furthermore, no differences were observed in competence (Sensor(M = 34.8, SD = 5.2), Non-Sensor(M = 33.9, SD = 7.7)) or effort/importance (Sensor(M = 26.4, SD = 5.5), Non-Sensor(M = 26.7, SD = 6.15)). No significant difference (p>0.05) in usability was found between the sensor condition (M = 80.0, SD = 16.8) and the non-sensor condition (M = 75, SD = 22.1). Prior research suggests that SUS scores above 68 are considered above average, indicating that both conditions were rated as having above-average usability.

4.4 Open-Ended Question Themes

4.4.1 Interesting Ideas. During the sensor-based condition, the most recurring theme was students exploring how real-time muscle activity could control game elements, particularly character movement. For example, one participant shared: "using my muscle energy to make things jump and run." Another noted: "one thing that was interesting to me was controlling the character by the grip of your hand." These responses reflect students' interest in how physiological input could influence on-screen actions. In contrast, during the non-sensor condition, students frequently explored the use of real-world data sources and variables to drive visualization. One student explained: "I like how you could make as many different variables as you liked." Another wrote: "Made a list of the scores from our basketball teams games." These reflections highlight how students found meaning in using external datasets to create interactive, data-driven visualizations.

4.4.2 Challenges. In the sensor-based condition, the most common theme related to difficulty centered on controlling and interpreting muscle signals. For example, one participant shared: "Trying to get a range of energy to accurately make it jump, move, and stop at the right times." Another noted challenges with physical setup, stating: "Figuring out the correct placement of the wires. I put right colors in the correct area, but I think it could be more specified." In contrast, challenges during the non-sensor condition were primarily related to understanding programming concepts and using block-based tools. One participant remarked: "Understanding which variable to use" was hard, while another commented: "Getting used to adding new blocks." These responses reflect how each condition introduced distinct cognitive and technical challenges.

5 Discussions

Observations from this study suggest that both sensor-based and non-sensor-based activities can offer positive hands-on experiences in educational settings. Specifically, no significant differences in self-efficacy were observed between the sensor and non-sensor conditions (RQ1). While prior research has highlighted the benefits of using physical sensors to enhance hands-on learning experiences [19, 34], these findings indicate that non-sensor approaches may have comparable effects on students' self-efficacy.

Findings related to programming confidence (RQ2) suggest minimal differences between sensor-based and non-sensor-based approaches. This may indicate that, when designed effectively, activities using pre-recorded sensor data can support programming confidence in ways comparable to real-time sensor interactions. However, further research with larger sample sizes and a broader range of sensor types is needed to validate this observation. Additionally, students reported similar levels of interest, motivation (RQ3), and usability across both conditions, further reinforcing the observed similarities between the sensor and non-sensor approaches.

When asked about aspects they found interesting, students in the sensor-based condition often highlighted the interactive elements of the activity. In contrast, responses in the non-sensor condition focused more on programming concepts such as the use of variables and lists. Students also more frequently commented on plotting and data visualization features during the non-sensor condition. These findings suggest that sensor-based approaches may be most effective when the goal is to emphasize interactivity. Conversely, non-sensor approaches may be more suitable when the focus is on reinforcing fundamental programming concepts.

Additional research is needed to better understand the differences between sensor-based and non-sensor-based approaches. While the goal of this study was to provide a preliminary evaluation of both methods, we acknowledge several potential threats to validity. In particular, future studies should examine whether similar findings emerge when incorporating a wider range of sensor types and larger sample sizes. Further studies should also explore whether similarities are also observed when using tools that feature text-based programming instead of block-based programming.

6 Conclusion

This paper presents a preliminary study examining both sensor-based and non-sensor-based approaches for engaging students with physiological sensing. Findings suggest that students had comparable experiences across both conditions. While the sensor-based activities appeared to promote interest through interactive features, the non-sensor activities encouraged greater focus on fundamental programming concepts. Further research is needed to determine whether these patterns hold when using different types of sensors and tools.

7 Acknowledgements

References

[1] Marvin Andujar and Juan E Gilbert. 2013. Let's learn! enhancing user's engagement levels through passive brain-computer interfaces. In *CHI'13 Extended Abstracts on Human Factors in Computing Systems*. 703–708.

[2] Alissa N Antle, Leslie Chesick, Aaron Levisohn, Srilekha Kirshnamachari Sridharan, and Perry Tan. 2015. Using neurofeedback to teach self-regulation to children living in poverty. In *Proceedings of the 14th International Conference on Interaction Design and Children*. 119–128.

[3] Alissa N Antle, Elgin-Skye McLaren, Holly Fiedler, and Naomi Johnson. 2019. Evaluating the impact of a mobile neurofeedback app for young children at school and home. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*. 1–13.

[4] John Brooke. 1996. SUS-A quick and dirty usability scale. *Usability evaluation in industry* 189, 194 (1996), 4–7.

[5] Francisco Enrique Vicente Castro, Kayla DesPortes, William Payne, Yoav Bergner, and Kathleen McDermott. 2022. Ai+ dance: Co-designing culturally sustaining

curricular resources for ai and ethics education through artistic computing. In *Proceedings of the 2022 ACM Conference on International Computing Education Research-Volume 2*. 26–27.
[6] Alexandra Gendreau Chakarov, Quentin Biddy, Jennifer Jacobs, Mimi Recker, and Tamara Sumner. 2020. Opening the Black Box: Investigating Student Understanding of Data Displays Using Programmable Sensor Technology.. In *ICER*. 291–301.
[7] Cynthia Carter Ching, Mary K Stewart, Danielle E Hagood, and Roxanne Naseem Rashedi. 2016. Representing and reconciling personal data and experience in a wearable technology gaming project. *IEEE Transactions on Learning Technologies* 9, 4 (2016), 342–353.
[8] Deborah R. Compeau and Christopher A. Higgins. 1995. Computer self-efficacy: Development of a measure and initial test. *MIS quarterly* (1995), 189–211.
[9] Juliet Corbin and Anselm Strauss. 2008. Basics of qualitative research: Techniques and procedures for developing grounded theory. (2008).
[10] Matthew H Dabney, Brian C Dean, and Tom Rogers. 2013. No sensor left behind: enriching computing education with mobile devices. In *Proceeding of the 44th ACM technical symposium on Computer science education*. 627–632.
[11] Shelly Engelman, Brian Magerko, Tom McKlin, Morgan Miller, Doug Edwards, and Jason Freeman. 2017. Creativity in authentic STEAM education with EarSketch. In *Proceedings of the 2017 ACM SIGCSE Technical Symposium on Computer Science Education*. 183–188.
[12] Stephen H Fairclough. 2010. Physiological computing: interfacing with the human nervous system. In *Sensing emotions*. Springer, 1–20.
[13] Alessio Ferrari, Thaide Huichapa, Paola Spoletini, Nicole Novielli, Davide Fucci, and Daniela Girardi. 2024. Using voice and biofeedback to predict user engagement during product feedback interviews. *ACM Transactions on Software Engineering and Methodology* 33, 4 (2024), 1–36.
[14] Bjørn Fjukstad, Nina Angelvik, Maria Wulff Hauglann, Joachim Sveia Knutsen, Morten Grønnesby, Hedinn Gunhildrud, and Lars Ailo Bongo. 2018. Low-cost programmable air quality sensor kits in science education. In *Proceedings of the 49th ACM Technical Symposium on Computer Science Education*. 227–232.
[15] Jérémy Frey, Renaud Gervais, Stéphanie Fleck, Fabien Lotte, and Martin Hachet. 2014. Teegi: tangible EEG interface. In *Proceedings of the 27th annual ACM symposium on User interface software and technology*. ACM, 301–308.
[16] Alexandra Gendreau Chakarov, Mimi Recker, Jennifer Jacobs, Katie Van Horne, and Tamara Sumner. 2019. Designing a middle school science curriculum that integrates computational thinking and sensor technology. In *Proceedings of the 50th ACM Technical Symposium on Computer Science Education*. 818–824.
[17] Jamie Gorson, Kathryn Cunningham, Marcelo Worsley, and Eleanor O’Rourke. 2022. Using electrodermal activity measurements to understand student emotions while programming. In *Proceedings of the 2022 ACM Conference on International Computing Education Research-Volume 1*. 105–119.
[18] Bryan Hernandez-Cuevas, William Egbert, Andre Denham, Ajay Mehul, and Chris S Crawford. 2020. Changing Minds: Exploring Brain-Computer Interface Experiences with High School Students. In *Extended Abstracts of the 2020 CHI Conference on Human Factors in Computing Systems*. 1–10.
[19] Bryan Y Hernández-Cuevas, Myles Lewis, Wesley Junkins, Chris S Crawford, Andre Denham, and Feiya Luo. 2025. PhysioML: A Web-Based Tool for Machine Learning Education with Real-Time Physiological Data. In *Proceedings of the 56th ACM Technical Symposium on Computer Science Education V. 1*. 485–491.
[20] Zahid Hossain, Engin W Bumbacher, Alice M Chung, Honesty Kim, Casey Litton, Ashley D Walter, Sachin N Pradhan, Kemi Jona, Paulo Blikstein, and Ingmar H Riedel-Kruse. 2016. Interactive and scalable biology cloud experimentation for scientific inquiry and education. *Nature biotechnology* 34, 12 (2016), 1293–1298.
[21] Sherry Hsi. 2017. Science Thinking for Tomorrow Today. *@ Concord* 21, 2 (2017).
[22] Jin Huang, Chun Yu, Yuntao Wang, Yuhang Zhao, Siqi Liu, Chou Mo, Jie Liu, Lie Zhang, and Yuanchun Shi. 2014. FOCUS: enhancing children’s engagement in reading by using contextual BCI training sessions. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. 1905–1908.
[23] Stephanie T Jones, JaCoya Thompson, and Marcelo Worsley. 2020. Data in Motion: Sports as a site for expansive learning. *Computer Science Education* 30, 3 (2020), 279–312.
[24] Brian Jordan, Nisha Devasia, Jenna Hong, Randi Williams, and Cynthia Breazeal. 2021. PoseBlocks: A toolkit for creating (and dancing) with AI. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 35. 15551–15559.
[25] Yasmin B Kafai, Deborah A Fields, Debora A Lui, Justice T Walker, Mia S Shaw, Gayithri Jayathirtha, Tomoko M Nakajima, Joanna Goode, and Michael T Giang. 2019. Stitching the loop with electronic textiles: Promoting equity in high school students’ competencies and perceptions of computer science. In *Proceedings of the 50th ACM technical symposium on computer science education*. 1176–1182.
[26] Seokbin Kang, Leyla Norooz, Vanessa Oguamanam, Angelisa C Plane, Tamara L Clegg, and Jon E Froehlich. 2016. SharedPhys: Live physiological sensing, whole-body interaction, and large-screen visualizations to support shared inquiry experiences. In *Proceedings of the the 15th international conference on interaction design and children*. 275–287.
[27] Steven C Kerlin, Scott P McDonald, and Gregory J Kelly. 2010. Complexity of secondary scientific data sources and students’ argumentative discourse. *International Journal of Science Education* 32, 9 (2010), 1207–1225.
[28] Melissa K Kjelvik and Elizabeth H Schultheis. 2019. Getting messy with authentic data: Exploring the potential of using data from scientific research to support student data literacy. *CBE—Life Sciences Education* 18, 2 (2019), es2.
[29] Nataliya Kosmyna and Pattie Maes. 2019. AttentivU: an EEG-based closed-loop biofeedback system for real-time monitoring and improvement of engagement for personalized learning. *Sensors* 19, 23 (2019), 5200.
[30] Ari Krakowski, Eric Greenwald, and Natalie Roman. 2022. " That’s What Science Is, All This Data:" Coding Data Visualizations in Middle School Science Classrooms. In *Proceedings of the 53rd ACM Technical Symposium on Computer Science Education-Volume 1*. 36–42.
[31] Vishesh Kumar and Marcelo Worsley. 2023. Scratch for sports: athletic drills as a platform for experiencing, understanding, and developing AI-driven apps. In *Proceedings of the AAAI conference on artificial intelligence*, Vol. 37. 16011–16016.
[32] Victor R Lee, Joel Drake, and Kylie Williamson. 2015. Let’s get physical: K-12 students using wearable devices to obtain and learn about data from physical activities. *TechTrends* 59, 4 (2015), 46–53.
[33] Victor R Lee and Maneksha DuMont. 2010. An exploration into how physical activity data-recording devices could be used in computer-supported data investigations. *International Journal of Computers for Mathematical Learning* 15 (2010), 167–189.
[34] Myles Lewis, Pranay Joshi, Wesley Cade Junkins, Vincent Ingram, and Chris S Crawford. 2025. PhysioBots: Engaging K-12 Students with Physiological Computing and Robotics. In *Proceedings of the Extended Abstracts of the CHI Conference on Human Factors in Computing Systems (CHI EA ’25)*. Association for Computing Machinery, New York, NY, USA, Article 434, 8 pages. doi:10.1145/3706599.3720106
[35] Zifeng Liu, Shan Zhang, Maya Israel, Robert Smith, Wanli Xing, and Victor Minces. 2025. Engaging K-12 students with flow-based music programming: An experience report on its impact on teaching and learning. In *Proceedings of the 56th ACM Technical Symposium on Computer Science Education V. 1*. 708–714.
[36] Tom Moher. 2006. Embedded phenomena: Supporting science learning with classroom-sized distributed simulations. In *Proceedings of the SIGCHI conference on human factors in computing systems*. 691–700.
[37] Katherine Perkins, Wendy Adams, Michael Dubson, Noah Finkelstein, Sam Reid, Carl Wieman, and Ron LeMaster. 2006. PhET: Interactive simulations for teaching and learning physics. *The physics teacher* 44, 1 (2006), 18–23.
[38] Andreas Trier Poulsen, Simon Kamronn, Jacek Dmochowski, Lucas C Parra, and Lars Kai Hansen. 2017. EEG in the classroom: Synchronised neural recordings during video presentation. *Scientific reports* 7, 1 (2017), 43916.
[39] Prerna Ravi, Robert Parks, John Masla, Hal Abelson, and Cynthia Breazeal. 2024. "Data comes from the real world": A Constructionist Approach to Mainstreaming K12 Data Science Education. In *Proceedings of the 2024 on ACM Virtual Global Computing Education Conference V. 1*. 271–274.
[40] Anke V Reinschluessel and Regan L Mandryk. 2016. Using positive or negative reinforcement in neurofeedback games for training self-regulation. In *Proceedings of the 2016 annual symposium on computer-human interaction in play*. 186–198.
[41] Mike Richards, Marian Petre, and Arosha K Bandara. 2012. Starting with Ubicomp: using the SenseBoard to introduce computing. In *Proceedings of the 43rd ACM technical symposium on Computer Science Education*. 583–588.
[42] Richard M Ryan. 1982. Control and information in the intrapersonal sphere: An extension of cognitive evaluation theory. *Journal of personality and social psychology* 43, 3 (1982), 450.
[43] T Scott Saponas, Desney S Tan, Dan Morris, and Ravin Balakrishnan. 2008. Demonstrating the feasibility of using forearm electromyography for muscle-computer interfaces. In *Proceedings of the SIGCHI conference on human factors in computing systems*. 515–524.
[44] Natalia Silvis-Cividjian, Joshua Kenyon, Elina Nazarian, Stijn Sluis, and Martin Gevonden. 2024. On Using Physiological Sensors and AI to Monitor Emotions in a Bug-Hunting Game. In *Proceedings of the 2024 on Innovation and Technology in Computer Science Education V. 1*. 429–435.
[45] Hussel Suriyaarachchi, Paul Denny, and Suranga Nanayakkara. 2022. Scratch and sense: Using real-time sensor data to motivate students learning Scratch. In *Proceedings of the 53rd ACM Technical Symposium on Computer Science Education-Volume 1*. 983–989.
[46] Daniel Szafir and Bilge Mutlu. 2012. Pay attention! Designing adaptive agents that monitor and improve user engagement. In *Proceedings of the SIGCHI conference on human factors in computing systems*. 11–20.
[47] Daniel Szafir and Bilge Mutlu. 2013. ARTFul: adaptive review technology for flipped learning. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. 1001–1010.
[48] Abigail Zimmermann-Niefield, R Benjamin Shapiro, and Shaun Kane. 2019. Sports and machine learning: How young people can use data from their own bodies to learn about machine learning. *XRDS: Crossroads, the ACM Magazine for students* 25, 4 (2019), 44–49.