# Integrating Computer Science and Physical Education in Elementary Schools with Data Science Learning Modules Using Wearable Microcontrollers

Hyun Jeong
*Fordham University*
New York, NY, USA
sjeong18@fordham.edu

Jamie Payton
*Temple University*
Philadelphia, PA, USA
payton@temple.edu

Christine Julien
University of Texas at Austin
Austin, TX, USA
c.julien@utexas.edu

Darla Castelli
University of Texas at Austin
Austin, TX, USA
dcastelli@utexas.edu

***Abstract*—With limited time in the school day to meet required state-level objectives across subjects, it is challenging to meet the need to address declining physical activity levels and to incorporate additional computer science instruction in elementary schools. Project moveSMART uses a web-based platform to integrate opportunities for physical education with computer science and computational thinking (CS/CT) learning activities. Fourth grade students who underwent a single-day intervention experienced a significant improvement in their interest in coding and their perceptions of coders. This paper presents a series of three new lessons in the fifth grade moveSMART curriculum that allow students to create and improve a physical activity monitor. We employ a project-based learning approach to expose students to data analysis and machine learning in the framework of classification and pattern recognition.**

***Keywords—integrative curriculum, gamification, elementary school, physical activity, pervasive computing, BBC micro:bit***

## I. Introduction

Physical education is declining in schools, exemplified by the statistic that less than a quarter of 6 to 17-year-olds in the US receive the recommended daily hour of physical activity [1]. In addition, only 24% of students from low-income families participate in organized physical activity (PA), compared to 45% of their high-income counterparts; this discrepancy makes the lack of school-based organized physical activity more detrimental to underserved populations [2]. Children from low-income and ethnic minority communities commonly lack access to quality computer science and computational thinking (CS/CT) education due to social and structural barriers, including educator and parent stereotypes and a lack of reliable technology access in the household [3]. Physical education and computer science are subject areas that have long-lasting impacts on students' long-term well-being. Children who do meet physical activity guidelines are more likely to meet or exceed academic standards, and CS/CT education is found to positively correlate to higher college enrollment rates and improved problem-solving abilities [4].

Despite the known benefits of increasing PA and CS/CT education in the early stages of public education, it is challenging for elementary schools to address these issues, due to a crowded curriculum required to meet state learning objectives across academic subjects. Project moveSMART addresses this issue through an integrative curriculum that delivers computer science lessons contextualized within students' own physical activity data. Project moveSMART promotes increased PA and CS/CT instruction in alignment with learning standards in Texas Essential Knowledge and Skills (TEKS), as our partner schools are located in Austin, Texas.

In this paper, we propose a series of three data science lessons to be added to the existing fifth grade moveSMART curriculum, which consists of data analysis learning activities and programming tutorials which allow students to improve the accuracy of their activity monitor. Students use grade-level appropriate physical computing tools, including Microsoft MakeCode, a block-based programming environment for the micro:bit, and the built-in triaxial accelerometer. We provide an outline of learning objectives, the corresponding TEKS standards, and the suggested Computer Science Teachers Association (CSTA) standard alignment. We also explain the design choices made while creating the machine learning (ML) based human activity recognition algorithm used in the final learning activity.

## II. Background

### A. Project moveSMART Framework

Project moveSMART is a collaborative educational game that promotes increased PA and CS/CT while also delivering content aligned with state learning standards. The fifth grade moveSMART curriculum hosts a collaborative game that progresses through a visual map of US cities (Fig. 1.). After a PA session, each student self-reports their activity levels with a check-in box that contains a Raspberry Pi, an RFID reader, and four buttons (Fig. 2.). Based on the collective activity level translated into miles on the road, the class progresses through waypoints; in each waypoint, students complete learning modules that incorporate interdisciplinary curricular material (science, math, social studies, language arts, etc.) placed in the context of the waypoint. Students also complete a series of block-based programming tutorials introducing students to computer science concepts used to develop and improve a


Fig. 1. moveSMART journey through U.S. cities

2155-6814/22/$31.00 ©2022 IEEE
DOI 10.1109/MASS56207.2022.00105



2022 IEEE 19th International Conference on Mobile Ad Hoc and Smart Systems (MASS) | 978-1-6654-7180-0/22/$31.00 ©2022 IEEE | DOI: 10.1109/MASS56207.2022.00105

Fig. 2. Physical activity check-in box

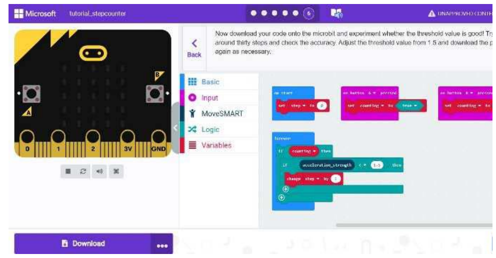


Fig. 3. Completed Learning Module 1

pedometer to track physical activity levels, supported by the BBC micro:bit, a small computer built for education.

The series of tutorials to program a wearable physical activity monitor culminates as students create a pedometer that records the step count and returns the step rate using mathematical operations and multiple conditionals. MakeCode, the block-based programming language associated with the micro:bit, has a native "on shake" block that returns "true" whenever the acceleration strength value is above the threshold of 1.5g. At this stage, students learn about the concept of acceleration through a short educational video, but exclusively use the abstracted "on shake" block rather than the raw data from the accelerometer.

### B. *Related Works and Motivation*

In the pilot run of our fourth grade moveSMART curriculum in elementary schools in Austin, Texas, focus group interviews revealed that students are interested in exploring data analysis-related questions, such as investigating the types and rates of their physical activity. Students also expressed interest in drawing charts to make data-based inferences and plan activity levels for the rest of the day.

With unprecedentedly large amounts of data available and data analysis methods expanding, "more aspects of the economy, society, and daily life will become dependent on data" [5]. A 2018 consensus report from the National Academies of Sciences, Engineering, and Medicine emphasizes the importance of data acumen, the "skills that all people need to be educated citizens in a data-driven world." [5] There are successful initiatives to teach data science to older elementary school students through integration with other science subjects like physics, such as the "Daily Do Sensemaking" activities created by the National Science Teaching Association (NSTA). For example, the "How Can We Use Data to Predict the Length of a Shadow?" lesson introduces data science as a discipline that collects, analyzes, and makes decisions based on large amounts of information. In addition, existing literature shows that physical activity is an effective tool to improve interest in CS, especially in underrepresented populations [6].

We believe that learning modules integrating PA, CS, and data science that guide students to build wearable activity monitors can increase students' motivation and success at learning computer science concepts and participating in in-school physical activities. The triaxial accelerometer native to the micro:bit provides ample data which can be used to teach analysis and CT skills at the fifth grade level, from simple numeric comparisons and graphing to pattern recognition and classification. The research area of using body-mounted inertial sensor data for human activity recognition is also well developed; therefore, we could package existing machine learning-based models into custom blocks for student use.

In the following sections, we introduce three data science-based learning modules: Calibrated Step Counter, Graphs and Patterns, and Machine Learning-based Activity Level Monitor. In each section, we provide a table that outlines learning objectives and the corresponding TEKS standards and suggested alignment with waypoint-relevant social studies content. In addition, we provide a sample version of each completed learning module.

## III. Learning Module 1: Calibrated Step Counter

This learning activity introduces students to the process of comparing and refining algorithms to obtain maximum accuracy (Table I). Students begin with a review of the concept of

TABLE I. LEARNING OBJECTIVES AND TEKS STANDARD ALIGNMENT: CALIBRATED STEP COUNTER

| Learning Objectives | Grade 5 TEKS Standards | CSTA Standards |
|---|---|---|
| Compare and order acceleration strength in g using inequalities | Math 2.(B). Number and Operations: compare and order two decimals to thousandths and represent comparisons using the symbols >, <, or = | |
| Determine optimal acceleration strength threshold for individual gait | Science 2.(B). Scientific Investigation and Reasoning: ask well defined questions, formulate testable hypotheses, and select and use appropriate equipment and technology | 1B-AP-08: Compare and refine multiple algorithms for the same task and determine which is the most appropriate.<br>1B-AP-10: Create programs that include sequences, events, loops, and conditionals. |
| Participate in a short active break session and monitor activity level with developed step counter | Physical Education 4.(D). define the principle of frequency, intensity, and time and describe how to incorporate these principles to improve fitness | |

acceleration through a short video and text, emphasizing the idea that acceleration "strength" is related to the intensity of movement. We provide a micro:bit program that shows the detected acceleration strength on the screen in g. Students are then instructed to shake the micro:bit at three different intensities in increasing order and observe the acceleration strength numbers shown on the screen. Students represent the pattern they notice between their shake intensity and acceleration strength using the inequality sign "<" and "positive/negative correlation." We relate this activity to the concept of intensity in exercise, and students are encouraged to consider questions about the intensity of exercise (e.g., "How fast is my heart beating?", "How out of breath am I?").

Students then learn about the mechanism of the "on shake" block they used in a previous step counter learning activity. Students design a scientific experiment to calibrate the step counter to maximize the pedometer accuracy for their own walking pattern. Students practice the TEKS Grade 5 Science skill of formulating testable hypotheses and arrive at the optimal acceleration strength threshold for their own gait through the scientific cycle of forming a hypothesis, testing with experiments, analyzing data, and reforming the hypothesis.

Fig. 3. shows a sample of the finished coding tutorial, in which the threshold for a step count is 1.5g. After downloading the program onto their micro:bit, students are encouraged to wear the micro:bit during an in-class "active break" session related to the waypoint's social studies content. Instructors are encouraged to discuss the course material with subject teachers to create an appropriate activity. One suggestion is a quiz activity in which the teacher asks a true or false question, and the students hop if the answer is true and crouch to the floor if the answer is false. Afterward, students report their activity levels recorded by the micro:bit; they are encouraged to wear the step counter over multiple active break sessions and analyze the trend of their step counts based on their activities.

## IV. Learning Module 2: Graphs and Patterns

This learning activity introduces students to data visualization (Table II). Students collect acceleration strength data using the micro:bit to draw a time series scatterplot on graphing paper and identify discrete points in terms of x and y pairs. Then students observe, analyze, and discuss several pre-generated acceleration graphs paired with specific physical

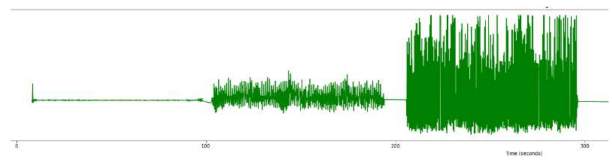

Fig. 4. Acceleration strength graphs: standing, walking, running

activities (Fig. 4.). We provide acceleration strength graphs obtained from everyday physical activities such as running, walking, climbing stairs, jumping, and dribbling a ball. Instructors guide students to recognize visual patterns and shape repetitions (e.g., peaks and troughs) in the graph, reinforcing the concept of pattern recognition in data science.

After that, students play a "guess my sport" modified charades activity as a class. In this activity, one student acts out a repetitive movement in sports, such as dribbling a basketball, swinging a baseball bat, and underhand serving in volleyball. The acceleration strength data is set up to display as a graph in real-time on a classroom digital screen, and the other students face away from the actor and towards the screen. As a class, the students attempt to guess the sport and determine whether the activity is a low or moderate-to-high intensity activity.

This learning activity solidifies students' understanding that pattern recognition is a significant part of data analysis. Students acknowledge the process of grouping similar—rather than identical—individual instances by recognizing patterns. For example, students may realize that the repetitive peaks in the graph for "running" are very similar in shape and size, though not identical.

## V. Learning Module 3: Machine Learning-based Activity Level Monitor

This learning activity introduces students to the application of machine learning by providing students with a machine learning-based human activity recognition algorithm (Table III). Many modern inertial sensor-based energy expenditure calculators use machine learning-based activity recognition algorithms due to their high versatility and reliability. Consequently, we believe that students will benefit from using the same tools at the culmination of their activity level monitor project. However, it is unrealistic for fifth grade students to develop a machine learning algorithm. It is technologically

TABLE II. Learning Objectives and TEKS Standard Alignment: Graphs and Patterns

| Learning Objectives | Grade 5 TEKS Standards | CSTA Standards |
| --- | --- | --- |
| Draw a graph with time and acceleration strength axes and identify points on the graph as pairs of data | Math 8.(B). Geometry and measurement: describe the process for graphing ordered pairs of numbers in the first quadrant of the coordinate plane | 1B-DA-06: Organize and present collected data visually to highlight relationships and support a claim. |
| Analyze time series acceleration graphs and visually recognize patterns from similar graphs paired with patterns of physical activity | Math 9.(B), (C). Data Analysis: represent discrete paired data on a scatterplot; and solve one- and two-step problems using data from a frequency table, dot plot, bar graph, stem-and-leaf plot, or scatterplot | |
| Collaborate as a class to recognize patterns from acceleration strength graphs to make evidence-based predictions regarding activity type | Physical Education 1.(K), (L) demonstrate competence in manipulative skills in dynamic situations such as overhand throw, catch, shooting, hand dribble, foot dribble, kick, and striking activities such as hitting a softball | |

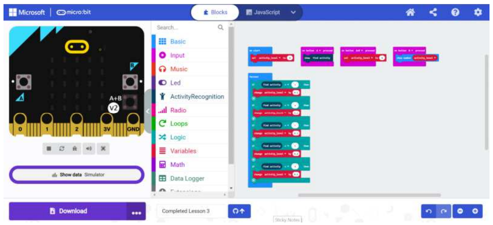

Fig. 5. Completed Learning Module 3

impractical to run an independent machine learning algorithm on a technologically limited device like a micro:bit, as well. Hence, we provide a pre-trained machine learning-based human activity recognition model via a custom "find activity" block in the MakeCode platform, which predicts the current activity being performed based on acceleration values collected in real time. Fig. 6. outlines the activity recognition model developed. While students are not expected to understand specific machine learning models such as the support vector machine we used to develop the custom block, we encourage students to understand machine learning as the process of pattern recognition based on similar data points. In the previous lessons, students analyze the time series acceleration strength data to recognize differences between the graphs for different activities. Students may recognize that the acceleration strength data points for "running" have more variation than "standing"; we instruct that the machine learning model also classifies activity classes by detecting similarities from time series features such as average and variance.

### *A. Student-facing programming tutorial*

At this stage, students know that physical activity can occur at different intensities, and are aware of methods to check PA intensity, such as using their pedometer or self-monitoring heart rate. Students then brainstorm the advantages and disadvantages of the activity monitor they developed: students are guided to recognize that while the pedometer algorithm is easy to calibrate and may provide a general idea of movement levels, it fails to account for different intensities in activity levels. We then introduce students to ML-based activity recognition, comparing it to Learning Module 2 in which students observed different time series acceleration strength graphs to recognize patterns. Students recognize that it is impractical for humans to observe large amounts of raw data and form predictions when needed; therefore, machine learning is a similar tool that can automatically recognize patterns from classified data points.

Then, students learn about the custom "find activity" block that we created for use in Project moveSMART learning activities. The block predicts the activity class that most closely resembles the acceleration data obtained by the micro:bit in real time. The block records acceleration every 0.1 seconds and returns the activity class based on the previous 20 data points. We provide five classes: standing, walking, running, climbing stairs, and dribbling ball. Students use the custom block to code a series of conditional statements which add different values to the "activity level" variable depending on how vigorous the activity is. Fig. 5. outlines a completed version of the activity monitor.

### *B. "Find activity" machine learning algorithm design*

To create the "find activity" block, we collected x, y, and z axes raw acceleration data every 0.1 seconds using the micro:bit, logging it in the flash drive, which can store up to around 15 minutes of data points. One 19-year-old female researcher fastened the micro:bit on her wrist using two rubber bands and performed the following five activity classes for 2 minutes each.

- Stand at 0 miles per hour (flat ground)
- Walk at 2.7 miles per hour (treadmill)
- Run at 5.4 miles per hour (treadmill)
- Climb stairs at the researcher's regular pace (stairs)
- Dribble a ball without significant leg movement (flat ground)

The recorded data was exported as a CSV file, then preprocessed and trained using the Python pandas data analysis library and scikit-learn machine learning library. The code was saved in both Jupyter Notebook and Google Colaboratory form for convenient reproduction.

In the preprocessing step, we used a sliding window of 20 data points, so that each window is 2 seconds, as previous literature using data from a single triaxial accelerometer to

TABLE III. LEARNING OBJECTIVES AND TEKS STANDARD ALIGNMENT: MACHINE LEARNING-BASED ACTIVITY LEVEL MONITOR

| Learning Objectives | Grade 5 TEKS Standards | CSTA Standards |
|---|---|---|
| Understand the basic concept of machine learning as in "finding patterns from looking at examples in data and applying it to new things to make predictions" | Math 9.(B), (C). Data Analysis: solve one- and two-step problems using data from a frequency table, dot plot, bar graph, stem-and-leaf plot, or scatterplot | |
| Use a pre-trained activity recognition program to code a rudimentary activity level monitor | Technology Applications 4. (C). Critical Thinking, Problem Solving, and Decision Making: evaluate student-created products through self and peer review for relevance to the assignment or task | 1B-AP-10: Create programs that include sequences, events, loops, and conditionals.<br>1B-IC-19: Brainstorm ways to improve the accessibility and usability of technology products for the diverse needs and wants of users. |
| Participate in multiple active break sessions over a period of time; monitor and increase own activity levels | Physical Education 4. (A), (B). Physical Activity and Health: A. relate ways that aerobic exercise strengthens and improves the efficiency of the heart and lungs; B. self-monitor the heart rate during exercise | |

TABLE IV. CONFUSION MATRIX

| | C | D | R | S | W |
|---|---|---|---|---|---|
| **C** | 315 | 0 | 0 | 0 | 0 |
| **D** | 0 | 356 | 0 | 0 | 0 |
| **R** | 0 | 3 | 327 | 0 | 0 |
| **S** | 0 | 0 | 0 | 393 | 0 |
| **W** | 4 | 1 | 0 | 0 | 318 |

C: climbing stairs, D: dribbling ball, R: running (5.4 mph on treadmill), S: standing (0 mph), W: walking (2.7 mph on treadmill)

predict similar activity classes reported high accuracy in window sizes between 1 and 3 seconds. A total of 12 features were extracted for each window: mean, standard deviation, minimum, and maximum values for the x, y, and z axes each. Then, the features were standardized by removing the mean and scaling to unit variance to center the data around 0. The mode of the individual data points' activity classes was chosen as the activity class of a particular window.

The preprocessed data resulted in 4479 data points, divided evenly among five activity classes, with each data point containing 12 features and an activity class tag. The dataset was then trained using a Support Vector Machine (SVM) with a linear kernel and a one versus one decision function. Based on a train-test split of 7:3, the model accuracy score was 0.9971, and the confusion matrix showed promising results as well (Table IV).

From the trained model, we recorded four vectors: the mean and standard deviation of the scaler for feature preprocessing, and the coefficients and intercepts for the hyperplanes used in the SVM. Then we used MicroPython, a lightweight version of Python that can be run on microcontrollers like the micro:bit, to recreate the decision algorithm. The preprocessing, feature extraction, standard scaling, and prediction functions were all programmed using the four extracted vectors from the trained model. While the program is running, the micro:bit collects acceleration data every 0.1 seconds and evaluates a feature set from a 20-data-point window, which it then uses to evaluate the current activity and stores in the custom block "find activity."

Below we explain the considerations made while creating the custom "find activity" block. All design choices were made considering the following aspects.

- Based on published specifications and our experimental data, we have found a limit of approximately 8kB of Python script that can be copied onto the micro:bit, unless we recompile MicroPython.
- This lesson may be mass-deployed into elementary school classrooms where teachers may have little to no experience in programming yet may wish to retrain the model based on data collected from actual students.
- Instructors may add a Fine Arts activity to the moveSMART curriculum, in which students design and use a 3D printer to create micro:bit holders, such as a wristband, ankle band, or hip belt.

*1) Location of the micro:bit on the body: wrist-mount*

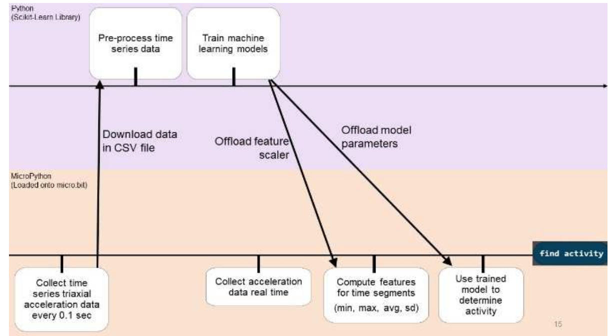


Fig. 6. Activity Recognition Model Flowchart

Most existing literature suggests that for an algorithm based on data from a single accelerometer, placing it on the hip maximizes accuracy, followed by the wrist [7]. During a trial run in which we trained the same model with hip-mount and wrist-mount data using stand/walk/run classes, the wrist-mount did indeed perform slightly worse. However, the accuracy scores were over 0.95 for both.

This learning activity depends on the student's commitment to wearing the micro:bit during physical activity. We believe the wrist-mount would lower student, teacher, and parent reluctance towards wearing the activity monitor because of the commercial prevalence of activity monitors such as the Fitbit and Apple Watch. Furthermore, a wristband-making activity is less time and resource-consuming than a belt-making activity, and a wristband will be more accommodating for students with mobility issues. Therefore, we determine that the slight loss in activity prediction model accuracy by choosing a wrist-mount is acceptable.

*2) Activity classes provided*

A basic energy expenditure calculator often considers only ambulation activities: standing, walking, and running. However, children's activity patterns are often sporadic and consist of isolated movements of different body parts. In addition to standing, walking, and running classes, we added the "climbing stairs" to account for activities not performed on a flat surface, and "dribbling ball" for isolated arm activity. In addition, basketball is a sport covered in the fifth grade PE curriculum, and students are expected to perform dribbling movements with relative ease, according to TEKS standards. We believe students' pre-existing understanding of the dribbling movement pattern will help reinforce the concept of pattern recognition in ML.

In addition, instructors are not limited to using only the five activity classes provided; they can train and export the model with new student data. We provide the source code for acceleration data logging as well as for machine learning in a Google Colaboratory platform, which only requires a free Google account. Using the given programs, instructors, even with little to no programming experience, can train an SVM model with custom-labeled activity classes. Hence, teachers can modify this learning activity to meet specific PE learning goals like "identifying the similarity between sports movements such as tennis underhand strokes and volleyball serves."

#### 3) Choice of preprocesssing method: time-series

Human activity recognition algorithms use either time domain features (e.g., mean, minimum, maximum, variance, range, interquartile range) or frequency domain features (e.g., entropy, kurtosis, skewness). While frequency domain features have been found to perform better in certain human activity recognition problems, such a method requires sinusoidal operations, calculus, and Fourier transforms. Since the math module in MicroPython is limited to basic arithmetic, frequency domain features cannot be calculated within the existing script limit. Even array operations like average and standard deviation must be coded from scratch; the deployed MicroPython program using time-domain features has a file size of 7.3kB, confirming our projection that there is little storage room to be spared.

#### 4) Choice of ML model: SVM with linear kernel

Among the machine learning models recommended for supervised learning of classified data, we selected a Support Vector Machine (SVM) with a linear kernel. This approach is appropriate for our context, as it is effective for activity recognition and is computationally simple so that it can run on the resource-constrained micro:bit device. Other algorithms such as the Naïve Bayes Classifier requires higher computational complexity while algorithms like k-Nearest Neighbors requires large amounts of storage and real-time computation.

## VI. Conclusion and Future Work

We have described a series of three integrated CS/CT and PE learning modules centered around developing an activity level monitor using the micro:bit and the block-based programming language MakeCode. These learning modules expand upon the existing activities by reinforcing the concept of pattern recognition in data science and exposing students to real-world uses of machine learning. We also provided an outline of design choices made while creating the ML algorithm to find a balance between algorithm accuracy and ease of use by students and instructors.

We plan to deploy the presented learning modules, along with the existing Project moveSMART curriculum, to a group of fifth grade school classrooms in the Fall 2022 semester. We will evaluate student interest, self-efficacy, and learning gains related to data science through pre- and post-lesson surveys. Through a continued Researcher-Practitioner Partnership, we seek to ensure the sustainability and scalability of the moveSMART curriculum, especially for the new data science learning activities, which elementary school personnel may be unfamiliar with.

## Acknowledgment

This material is based upon work supported by the National Science Foundation under Grant No. 2150152 and Grant No. 2031324.

## References

[1] R. Guthold, G.A. Stevens, L.M. Riley, F.C. Bull, “Global trends in insufficient physical activity among adolescents: a pooled analysis of 298 population-based surveys with May/June 2021 91·6 million participants,” The Lancet Child & Adolescent Health, vol. 4, no. 1, pp. 23-25, January 2020, doi: 10.1016/S2352-4642(19)30323-2.

[2] H.W. Kohl III and H.D. Cook, “Educating the student body: Taking physical activity and physical education to school,” 2013.

[3] D. R. and A. Swift, “U.S. minority students less exposed to computer science,” Gallup.com, 23-Sep-2021. [Online]. Available: https://news.gallup.com/poll/196307/minority-students-less-exposed-computer-science.aspx. [Accessed: 28-Jul-2022].

[4] E. Brown and R. S. Brown, "The Effect of Advanced Placement Computer Science Course Taking on College Enrollment," West Coast Analytics, March 2020.

[5] W. Martinez and D. LaLonde, “Data Science for Everyone Starts in Kindergarten: Strategies and Initiatives From the American Statistical Association,” Harvard Data Science Review, 2020.

[6] D. G. Kelley and P. Seeling, “Introducing underrepresented high school students to software engineering: using the micro:bit microcontroller to program connected autonomous cars,” Computer Applications in Engineering Education, vol. 28, no. 3, pp. 737-747, April 2020, doi: 10.1002/cae.22244.

[7] L. Bao and S. S. Intille. Activity recognition from user annotated acceleration data. In Proceedings of the 2nd International Conference on Pervasive Computing, pages 1–17, Apr 2004.