第 48 卷 第 5 期

七月 2023

Diqiu Kexue - Zhongguo Dizhi Daxue Xuebao/Earth Science Journal of China University of Geosciences

https://dqkx-periodicals.com

Vol. 48 No. 5

July 2023

https://doi.org/10.3799/dqkx.2023.784

Human Activity Recognition using Inertial and Stretch Sensors: Enhancing Performance with a Deep Learning Network Incorporating Aggregation Residual Transformation.

Jackson Harper, Madison Sinclair

Department of Computer Science, Faculty of Sciences, University of Missipie

Received: 29 June 2023

Revised: 09 July 2023

Accepted: 18 July 2023

Published: 23 July 2023

Abstract: With the rise of artificial intelligence, sensor-based human activity recognition (S-HAR) is increasingly being employed in healthcare monitoring for the elderly, fitness tracking, and patient rehabilitation using smart devices. Inertial sensors have been commonly used for S-HAR, but wearable devices have been demanding more comfort and flexibility in recent years. Consequently,there has been an effort to incorporate stretch sensors into S-HAR with the advancement of flexibleelectronics technology. This paper presents a deep learning network model, utilizing aggregation residual transformation, that can efficiently extract spatial–temporal features and perform activityclassification. The efficacy of the suggested model was assessed using the w-HAR dataset, which included both inertial and stretch sensor data. This dataset was used to train and test five fundamental deep learning models (CNN, LSTM, BiLSTM, GRU, and BiGRU), along with the proposed model.The primary objective of the w-HAR investigations was to determine the feasibility of utilizing stretch sensors for recognizing human actions. Additionally, this study aimed to explore the effectiveness of combining data from both inertial and stretch sensors in S-HAR. The results clearly demonstrate the effectiveness of the proposed approach in enhancing HAR using inertial and stretch sensors. The deep learning model we presented achieved an impressive accuracy of 97.68%. Notably, our method outperformed existing approaches and demonstrated excellent generalization capabilities.

Keywords: human activity recognition; inertial sensor; stretch sensor; low-power wearable device; deep residual learning network

1. Introduction

As wearable devices such as smartwatches and bracelets become more prevalent, it becomes possible to effortlessly record and analyze the daily movements of the human body without any limitations imposed by the surrounding environment [1]. The issue of recognizing human behavior using wearable sensors has been extensively studied. The main goal is to differentiate between various types of physical activities performed by individuals by analyzing time series data collected from these sensors [2]. A considerable amount of research has been conducted to develop advanced data processing methods, classification techniques, and machine learning (ML) models for implementing human activity recognition (HAR) in human-to-human interactions, human-to-machine interfaces, and automated systems [3].

Applications that recognize human activity are valuable for monitoring elderly health- care, tracking fitness, and aiding patient rehabilitation. To identify user activity, applications of HAR require ongoing sensor data, and recent advancements in sensor technology have facilitated the widespread use of HAR applications in daily life. Previous literature indi- cates that HAR can be categorized into two main types: video-based and sensor-based activity recognition [4,5].

Video-based HAR (V-HAR) involves extracting features of human activity from im- ages and video streams captured by cameras placed in the environment where humans are present [6]; although this approach provides a more intuitive understanding of the complexities involved in the task at hand, its applicability is limited to specific scenarios and cannot be used in unstructured environments due to its heavy reliance on external factors such as lighting and camera placement [7]. To overcome these limitations, sensor- based human activity recognition (S-HAR) has been developed. This approach utilizes data from multiple wearable sensors to identify, understand, and evaluate human behavior. Ini- tially, sensor technology was mainly used to analyze human gait and joint movements for healthcare diagnosis and rehabilitation purposes [8,9]. However, advancements in sensor technology have resulted in significant improvements in key aspects such as accuracy, size, and production costs [10].

The use of inertial sensors in HAR has proven to be effective [11]. However, wearable devices incorporating these sensors face several limitations [12]. The inclusion of inertial sensors often results in bulky devices that do not fit well on the skin and are uncomfortable due to rigid materials used in their construction. Additionally, the accuracy of these sensors in identifying activities heavily relies on motion velocity. To address these challenges, researchers have focused on developing flexible sensors made from lightweight and pli- able materials that are less affected by motion velocity [13]. One example is the stretch sensor, which shows promise for application in wearable technology. These sensors can be attached to garments to collect data related to body movements, respiration, and cardiac activity [14,15].

In S-HAR, conventional approaches involve using ML techniques to manually extract characteristics from sensor data [16]. These characteristics typically consist of mathematical or fundamental measurements, such as means, medians, and standard deviations. However, selecting the most relevant set of features requires domain-specific expertise and knowledge. However, these methods have limited recognition performance, as they rely on handcrafted features that depend on human experts [17]. To address this limitation, deep learning (DL) approaches have been proposed to automatically extract informative features from raw sensor data.

The contribution of this work is following:

1. To overcome this challenge, we propose a DL network with aggregation residual transformation called the ResNeXt model that can classify human activities based on inertial and stretch sensor data with satisfactory results.
2. The experiment also revealed that utilizing data from the stretch sensor yields im- proved recognition of various human actions compared to utilizing the initial sensor.

The rest of this article is organized as follows. Section 2 introduces recent related works. Section 3 describes the details of the proposed model. Section 4 shows our experimental results. Research findings are discussed in Section 5. Finally, Section 6 draws a conclusion of this work and shows challenging future works.

2. Related Works

3. Materials and Methods

3.1. Materials

The 23 Lithuanian students from the Faculty of Informatics ranged in age from 18 to 25 years. Variations were noted within this range, with a few older persons falling slightly outside of it. The gender distribution among the student cohort favored male students, who had a larger representation than females. Notably, the sample includes a handful of transgenders, contributing to the group's diversity of gender identities. All 23 students in the sample were pursuing programming-related specialties. This emphasis demonstrates their unique interest in learning about creating, building, and maintaining software systems. The majority of the participants were from Lithuania, which corresponded to the location of Kaunas University of Technology.

The 49 Polish respondents were 1st-year students of IT in Business and IT & Econo- metrics attending the Introduction to Programming course at the Faculty of Economics, Finance, and Management of University of Szczecin. Almost all of them were 19 years old; 1/4 of them were female, and 3/4 male. Alike the students at Kaunas University of Technology, they have a passion for developing software systems, yet in contrast to them, they are more focused on enterprise information systems rather than software in general.

3.2. Pareto-Optimized Gamified Programming Task Selection Model

Designing a Pareto-optimized gamified programming task selection model (Figure 1) for adaptive personalized learning involves

the creation of a multi-objective optimization model that seeks to balance multiple competing factors such as the learner's interests, abilities, task difficulty, novelty, and relevance to the curriculum. In this context, Pareto optimization refers to a state of allocation where it is impossible to make any one individual better off without making at least one individual worse off.
The model has the following key components:

1. Programming task bank: A repository of programming tasks, each classified according to their difficulty level, related topic, required skills, estimated completion time, etc.
2. Learner profile: A dynamic profile for each learner capturing their programming skill level, areas of interest, learning pace, historical performance on tasks, preferred learning style, etc.

3. Gamification elements: Incorporation of game design elements such as points, badges, leaderboards, achievement tracking, feedback, progress bars, storyline, etc.

Formally, the model can be described as follows:
Let L = {L1, L2, . . . , Ln} be the set of learners.
Each learner Li is represented as a tuple (idi, sli, ini, pi, hi, sti), where:
idi is the id,
sli is the skill level,
ini is the set of interests,
pi is the learning pace,
hi is the history of completed tasks,
sti is the learning style.
Let T = {T1, T2, . . . , Tm} be the set of tasks.
Each task Tj is represented as a tuple (idj, dj, tj, sj, timej), where:
idj is the id,
dj is the difficulty level,
tj is the topic,
sj is the set of required skills,
timej is the estimated completion time.
Let A : L × T → 2T be the assignment function, where 2T is the power set of T. This function assigns to each learner a set of tasks, i.e., A(Li) = {Ti1, Ti2, . . .} ⊆ T. Let O : L × T → 2T be the Pareto optimization function, defined as:
O(Li, T) = {Tj ∈ T | there does not exist Tk ∈ T such that Tk is better than Tj for Li}.
Let U : L × T × R → L be the update function, defined as:
U(Li, Tj, per f ormance) = Li′, where Li′ is the updated learner profile based on the performance on task Tj.
Here, L is the set of learners, each represented as a tuple of parameters. T is the set of tasks, each represented as a tuple of parameters. A is the assignment function that maps each learner to a set of tasks. O is the Pareto optimization function that assigns a learner a subset of tasks for which there are no better alternatives. U is the update function that updates a learner's profile based on their performance on a task.
The model starts by initializing the learners and tasks. Each learner is then assigned a set of tasks that are Pareto optimized for them. The Pareto optimization process involves finding the balance between different objectives (learner's interests, skill level, pace, etc., vs. the task's difficulty, skills required, topic relevance, etc.) to maximize the learning outcome. The learner's profile is updated after they complete a task based on their performance, and the task assignment process can be repeated as necessary. Gamification elements can be added to the model to increase learner engagement.

3.3. Implementation

Our goal was to fill a substantial knowledge gap on the potential benefits and inade- quate use of Progressive Web Applications (PWAs) in the education sector, particularly for educational programming. The FGPE+ model provides programmers with a unique and engaging mobile user experience (see Figure 2). To create an effective and pleasant learning environment, it blends the ideas of Pareto optimization, gamification, and programming exercises. The PWA-based mobile-compatible solution has a clean and straightforward user experience that facilitates navigation and interaction. The interface is designed in a modern, minimalist style, with an emphasis on usability and clarity. When new users start the app, they are met with a full onboarding experience that introduces them to the features and capabilities of the FGPE+ model. Within the app, users are encouraged to create individualized profiles. They can choose their preferred programming language (Javascript, Python, Java, C#, Cpp), skill level, and areas of interest. This data allows the FGPE+ model to personalize exercise recommendations to the user's specific requirements and goals.

1. The exercise abstract class represents a programming exercise. It has attributes such as id (exercise identifier), difficulty (difficulty level of the exercise), and learningOut- comes (a list of learning outcomes associated with the exercise). It provides meth- ods to access these attributes and defines three virtual methods: evaluateObjec- tiveWeights(to evaluate the objective weights of the exercise), calculateObjectiveVal- ues() (to calculate the objective values of the exercise), and compareTo() (to

compare two exercises based on their objective values).
2. ParetoExercise class represents an exercise that includes objective values. It inherits from the Exercise class and has an additional attribute called objectiveValues, which is a map that stores objective values for the exercise. It provides methods to obtain and set objective values for specific objectives and defines the dominates() method to check whether it dominates another ParetoExercise based on their objective values.
3. ExerciseSelector abstract class serves as the base class for the exercise selection al- gorithm. It has attributes exercises (a list of exercises to select from) and objec- tiveWeights (a map that holds the weights of different objectives). It provides meth- ods to select exercises, sets the exercises and objective weights, and defines six virtual methods that outline different steps of the algorithm: evaluateExercises() (to evaluate the exercises based on objectives), paretoOptimization() (to perform Pareto opti- mization on the exercises), diversityEnhancement() (to enhance the diversity of the exercise set), gamificationIntegration() (to integrate gamification elements into the ex- ercises), personalization() (to personalize the exercise selection), and evaluationAnd- FeedbackLoop() (to evaluate and refine the exercise selection based on feedback).
4. MyExerciseSelector. This class represents a specific implementation of ExerciseS- elector. It adds an additional attribute called the threshold (a threshold value for evaluation) and overrides the paretoOptimization() and evaluationAndFeedback- Loop() methods to provide custom implementation based on the defined threshold.
5. Objective class represents an objective to optimize in the exercise selection. It has an attribute name (the name of the objective) and provides a method to access the name.

The FGPE+ concept adds gamification aspects throughout the app to make the learning experience more interesting. For completing workouts, attaining milestones, and obtaining high scores, users gain points, badges, and virtual gifts. This gamified method encour- ages competitiveness, incentive, and ongoing progress. The app monitors and shows the progress and performance data of users. Users may check their completion rates, accuracy, and time required to complete each activity. The model offers customized reports and insights to assist users in identifying their own strengths, shortcomings, and opportunities for progress. The FGPE+ paradigm encourages user social engagement. Users may join groups, participate on discussion boards, and work together to solve puzzles. Users may also compare their performance to that of others, encouraging healthy rivalry and infor- mation exchange. Users obtain fast feedback on their workout answers, which helps them understand and improve from their mistakes. The software offers thorough explanations, code critiques, and advice to help users improve their programming skills. Users may also seek assistance from mentors or experienced programmers within the app.

Here is the pseudo code of the FGPE+ approach:

Inputs: List of exercises (ExerciseList); exercise attributes (e.g., learning outcomes, difficulty levels); objective weights (e.g., learning outcomes, engagement).

Outputs: Pareto optimal exercise set (ParetoSet).

Procedure: 1. DefineExerciseSelectionCriteria(); 2. CollectExerciseData(); 3. ParetoOp-
timization(); 4. DiversityEnhancement(); 5. GamificationIntegration(); 6. Personalization();
7. EvaluationAndFeedbackLoop().

Procedure DefineExerciseSelectionCriteria(): Specify criteria for selecting exercises based on objectives and gamification elements; define attributes to consider, such as learn- ing outcomes, difficulty levels, or programming concepts.

Procedure CollectExerciseData(): Gather information on exercises, including attributes and gamification elements; store exercise data in a suitable data structure.

Procedure ParetoOptimization(): Apply multi-objective optimization algorithm (e.g., NSGA-II, SPEA2) on the exercise data; generate a set of Pareto optimal solutions considering the defined objectives and weights.

Procedure DiversityEnhancement(): Enhance diversity in the selected exercise set; apply niching or crowding techniques to avoid redundancy and promote variety.

Procedure GamificationIntegration(): Integrate gamification elements into the selected exercises; consider elements such as points, badges, levels, leaderboards, or social interac- tion features.

Procedure Personalization(): Allow learners to personalize exercise selection based on preferences, prior knowledge, or skill levels; implement adaptive algorithms for dynami- cally adjusting exercise difficulty or sequence.

Procedure EvaluationAndFeedbackLoop(): Continuously evaluate the effectiveness of the exercise selection model; collect user feedback, learning outcomes, and engagement metrics; refine and improve the algorithm based on evaluation results.

Output ParetoSet: Return the set of exercises representing the Pareto optimal solutions.

4. Results

4.1. Evaluation of the PWA Version of the FGPE PLE Platform by Mobile Device Users

As the main purpose of redeveloping the FGPE PLE platform as a PWA was to make it more suitable for mobile device users, the first part of its evaluation was aimed at assessing to what extent we have succeeded. Although this could be evaluated with both self-report and behavioral measures [76], following the findings of [77] showing that the interpretation of the latter is not always straightforward and could be misleading, we decided to go with the former.

Apart from assessing the general students' attitude to using the FGPE PLE platform on a mobile device (Q1), we also strived to assess whether the learning place makes a difference in the mobile use of the platform (Q2 and Q3), as well as whether the mobile users still feel the need to use the PC version at all (Q4):

(Q1): How do you generally rate the mobile version of the FGPE PLE platform?

Answer range: 1 (bad)–5 (excellent).

(Q2): Do you think the mobile version of the FGPE PLE platform makes sense for students learning at home?

Answer range: 1 (bad)–5 (excellent).

(Q3): Do you think the mobile version of the FGPE PLE platform makes sense for students who study on the go to school/work?

Answer range: 1 (bad)–5 (excellent).

(Q4): Do you think it is possible to learn to write code only using the mobile version of the FGPE PLE platform—and without using the PC version at all?

Answer range: 1 (bad)–5 (excellent).

The results are summarized in Figure 4. The answers to Q1 demonstrate the over- whelmingly positive response to the mobile version of FGPE PLE in our study. The answers to Q2 and Q3 indicate the students see it as a convenient learning tool at home and during commuting to school/work (more so for the latter than the former). The answers to Q4 show that the students do not believe that programming can be learned only by using FGPE PLE on their mobile devices.

4.2. User Experience Analysis

In order to obtain a more detailed picture of how the users view the mobile version of FGPE PLE, the User Experience Questionnaire (UEQ) has been employed [78]. It is a well-established instrument used to evaluate the user experience of a product or service, designed to provide developers with a quick and straightforward way to assess the user experience of their product, be it a website, a software application, or any other kind of interactive system. It has been previously used for The Evaluation of User Experience on Learning Management Systems [79]. The UEQ measures six different scales:

- Attractiveness: covers the overall impression of the product, including whether it is pleasant or enjoyable to use.
- Perspicuity: measures how easy it is for users to understand how to use the product.
- Efficiency: evaluates the perception of how efficiently users can complete tasks using the product.
- Dependability: measures how reliable and predictable users find the product.
- Stimulation: evaluates how exciting and motivating the product is to use.
- Novelty: assesses whether the design of the product is creative and innovative and whether it meets users' expectations.

The questionnaire itself consists of 26 pairs of opposing adjectives (such as "compli- cated" vs. "easy"), and respondents rate their experience with the product on a 7-point Likert scale between these extremes (see Figure 5). The scores from these pairs of ad- jectives are then used to calculate scores on the six scales listed above. This provides a comprehensive and nuanced understanding of how users perceive the product or service.

Students from Poland and Lithuania provided answers to the User Experience Ques- tionnaire (UEQ), reflecting how they used the FGPE+ model. Each input indicates a score for a particular aspect of the user experience, such as appeal, perspicacity, effectiveness, reliability, stimulation, and innovation (Figure 6). These replies come from many student groups; therefore, they represent the distinctive perspectives and experiences of these various cohorts.

The availability of data from two countries created an opportunity for a cross-country analysis. This analysis aimed at identifying discrepancies and/or parallels in the assessment that could be linked to differences in pedagogy and cultural backgrounds of the students taught in various countries. Here is a brief analysis of the results:

- Attractiveness: The Lithuanian group had a higher average score (mean = 1.4141, std = 0.6081) compared to the Polish group (mean = 0.6607, std = 0.5785). This indicates that the Lithuanian students found the learning environment more attractive and appealing than the Polish students.
- Perspicuity: The Lithuanian group also scored higher (mean = 1.4914, std = 0.7604) than the Polish group (mean = 0.5580, std = 0.7732), suggesting that the Lithuanian students found the learning environment more clear and understandable.
- Efficiency: Again, the Lithuanian group's score was higher (mean = 1.5193, std = 0.6714) than the Polish group (mean = 0.2727, std = 0.6650), indicating that the Lithuanian students found the learning environment more efficient for achieving their tasks.
- Dependability: The Lithuanian group had a higher mean score (mean = 1.2462, std = 0.8367) than the Polish group (mean = 0.7047, std = 0.7909), indicating they found the learning environment more reliable and dependable.
- Stimulation: The Lithuanian group scored slightly higher in this dimension (mean = 1.4972, std = 0.6408) than the Polish

group (mean = 1.2548, std = 0.7053). This means that the Lithuanian students found the learning environment slightly more exciting and motivating.

- Novelty: Lastly, the Lithuanian group scored higher in terms of novelty (mean = 1.3701, std = 0.6361) compared to the Polish group (mean = 0.5738, std = 0.6177). This suggests that the Lithuanian students found the learning environment more innovative and creative.

By comparing the data, it is evident that Lithuanian students often score higher on the UEQ than Polish students. Such results indicates that, globally, Lithuanian students may have found the FGPE PLE mobile version easier to use. However, it is important to note that a thorough analysis is difficult without a complete understanding of the issues that correlate to the presented data. It may simply reflect differing cultural views, educational backgrounds, or degrees of knowledge with comparable systems rather than necessarily implying that the approach featuring mobile learning supported with FGPE is more frequently accepted in Lithuania. The lower results in the Polish student group may indicate an opportunity for the further development of the FGPE PLE in some areas to make it more flexible and advantageous for a wider range of users, but it could also be a reflection of various standards or expectations.

4.3. Knowledge Evaluation Survey: FGPE Approach vs. Classic Moodle Course

Following the guidelines proposed in [80] regarding the evaluation of area-specific effects of gamification which suggest knowledge improvement as an indicator relevant for gamification applications aimed at supporting learning, we have used the opportunity that the Lithuanian group had a parallel group learning programming without the use of the FGPE toolset, to compare the two educational approaches.

A knowledge evaluation survey was conducted comparing two groups of learners: the first group used the PWA-based FGPE exercise selection model of the Python pro- gramming course, while the second one used the typical Moodle course format of Python programming (non-gamified, lecturer assigned, and ordered programming tasks). Table 1 demonstrates the perceived knowledge evaluation of groups using the FGPE+ model and the Moodle course, demonstrating the efficacy of the model. The 'N' column refers to the sample size for each group. 'M (SD)' represents the mean and standard deviation of the perceived knowledge scores. 't' and 'p' are the t-statistic and p-value, respectively, from a t-test comparing the game and Moodle course group scores. The FGPE+ course group had a higher average perceived knowledge score (M = 4.11, SD = 0.51) than the Moodle group (M = 3.67, SD = 0.56). The t-value indicates that the Moodle course group's score was higher, and the small p-value ($p < 0.05$) suggests this difference was statistically significant. This could imply that the FGPE+ model was effective in increasing the perceived knowledge.

4.4. Effectiveness of Using Sharable Content Object Reference Model

The Sharable Content Object Reference Model (SCORM) is a collection of standards and specifications for web-based educational content. It provides a standardized approach to creating and delivering online learning content, ensuring interoperability, accessibility, and reusability. SCORM has been widely adopted by e-learning providers as it ensures that learning content and Learning Management Systems (LMS) can work seamlessly together, regardless of the developer or platform. The results of evaluation according to SCORM criteria are presented in Figure 7). Data were collected using a descriptive survey following the practice established by [81], and assessment was conducted using their suggested quasi-experimental approach. ANCOVA results (FT = 3.76; FC = 8.11; p = 0.04) revealed a substantial difference in the impacts of SCORM-conformant e-content and conventional material on academic achievement.

Seventeen people (normalized) responded from the original groups of FPGE+ and Moodle users (see Table 1).

5. Discussion

We believe that PWAs have not yet gained full momentum in the education sector, and their use remains limited [82]. There is still somewhat of a scarcity of research and different uses of PWAs in education, demonstrating that the approach is not commonly recognized or used in the programming education arena [83]. Our Pareto-optimized approach enabled a personalized and adapted exercise selection to meet the specific needs and skill levels of individual learners, demonstrating that adaptive learning algorithms and techniques can dynamically adjust the difficulty, content, and feedback of gamified programming exercises based on learners' progress and performance. To address these issues, the Framework for Gamified Programming Education (FGPE) was developed. The FGPE framework includes requirement formulation, a collection of gamified activities, and the creation of supporting software.

We would like to bring up a few critical topics for reader consideration. The first step is to examine the long-term consequences. Longitudinal studies are needed to assess the long-term influence of gamified exercises on students' programming abilities and infor- mation retention. We investigate whether incorporating gamification into programming instruction results in better long-term learning outcomes than standard techniques [84]. Fu- ture study should look into how various gamification tactics and mechanics affect students' motivation, perseverance, and pleasure in programming tasks. Longer research would also aid in assessing social interaction and cooperation, since including social components such as competition, collaboration, and peer evaluation might improve students' learning experiences and outcomes in the long run.

5.1. Importance of FGPE+ for STEM Education

The FGPE+ model's potential extends far beyond the realm of programming education and has significant implications for the broader STEM (Science, Technology, Engineering, and Mathematics) education landscape. In an era characterized by rapid advancements in technology and the increasing relevance of digital literacy, effective STEM education is crucial in order to allow for the break-out of current rigid education schemes [85]. The FGPE+ model, with its innovative blend of personalization, gamification, and mobile learning, is a robust tool for enhancing STEM learning experiences.

One of the central tenets of the FGPE+ model is its ability to tailor learning experiences to individual learners. This approach aligns well with the diverse nature of STEM education, where learners often come with varying levels of background knowledge and abilities. The FGPE+ model can accommodate these differences, making learning more effective and enjoyable. The adaptive algorithm can be extended beyond programming tasks to include other STEM-related exercises, such as solving mathematical problems or designing engineering solutions.

The incorporation of gamification techniques in the FGPE+ model is a significant asset for STEM education [86]. Gamification elements such as badges, points, leaderboards, and levels can make complex STEM concepts more engaging and accessible, thereby fostering a positive attitude towards these subjects. These elements can also promote healthy competition and motivation, encouraging students to continually improve their understanding and mastery of STEM subjects.

With the prevalence of mobile devices, the mobile-compatible nature of the FGPE+ model offers vast potential for remote and flexible learning. This aspect can break down barriers to STEM education, allowing learners to access resources and engage with STEM concepts anytime, anywhere. Mobile learning also aligns with the digital habits of the current generation, thereby increasing its effectiveness and appeal.

The FGPE+ model's use of real-world tasks mirrors the application-based learning that is critical in STEM fields. By engaging with tasks that reflect real-world challenges, students can gain a deeper understanding of the practical applications of their learning, making the learning process more meaningful and relevant.

In conclusion, the FGPE+ model is a significant tool that can transform STEM educa- tion. By making learning more personalized, engaging, flexible, and application-focused, it aligns with the goals of modern STEM education, promoting increased participation and achievement in these crucial fields. Future research and development efforts should consider how the principles of FGPE+ can be effectively integrated and implemented across various STEM disciplines.

5.2. Limitations

Despite the potential contributions, this study has certain limitations that should be considered:

- This study used a subset of students from Poland and Lithuania, which may not be typical of the whole population. We believe that extending the study's size and altering the demographics of the participants would potentially increase generalizability.
- This research relied heavily on self-report measures, which are subjective and prone to bias. Incorporating objective measures, such as performance-based assessments or tracking system data, could provide a more comprehensive evaluation of the platform's effectiveness.

- The study aimed to assess the FGPE PLE platform in various educational settings and learning scenarios. The findings may not fully represent the intricacies and complexity of various educational settings. Future study might look at the platform's efficacy in other educational institutions, student backgrounds, and instructional environments.
- The PWA version of the FGPE PLE platform was assessed particularly for program- ming instruction in the research. The findings may not be applicable in other domains or topic areas. Replicating the study in additional educational fields would be advan- tageous in determining the platform's generalizability.
- The investigation focused on the initial user experience and perceived knowledge. Understanding the long-term impact of the FGPE PLE platform on learners' pro- gramming skill development and information retention would necessitate additional research outside the scope of this study.

5.3. Potential Lines of Research

The study on the evaluation of the PWA version of the FGPE PLE platform opened up several potential lines of research:

- We concentrated on evaluating the platform's initial user experience and perceived knowledge. More study might be conducted to investigate the long-term consequences of utilizing the FGPE PLE platform on mobile devices. Longitudinal studies might look into the long-term influence on learning outcomes, programming skill development, and knowledge retention.
- In terms of perceived knowledge, we compared the FGPE+ model to a traditional Moodle course. Future study might compare the efficacy of other educational method- ologies, such as gamified platforms such as FGPE+ vs. traditional teaching methods. Comparative research might assist to uncover the advantages and disadvantages of each strategy and provide ideas into how to improve learning experiences.
- The platform was evaluated mostly using self-report measures in this study. Tracking user interactions, completion rates,

and performance statistics, for example, might give a more objective evaluation of learners' progress and engagement with the platform if learning analytics approaches are used. Analyzing such data might aid in the discovery of trends, the identification of areas for development, and the implementation of individualized learning interventions.

- Further study might look into the efficacy of certain instructional tactics used inside the FGPE PLE platform. Investigating how various gamification aspects, adaptive learning algorithms, or social interaction features influence motivation, engagement, and learning results might provide useful insights for building and enhancing educa- tional systems.
- Our investigation discovered some cross-national disparities in user experience per- ceptions. More thorough cross-cultural research might provide insight on how cultural backgrounds and educational environments impact FGPE PLE platform acceptability and efficacy. Understanding these cultural differences may help with the customiza- tion and localization of educational systems for varied learner groups.

6. Conclusions

This study explored the use of the FGPE+ model, a Pareto-optimized gamified pro- gramming exercise selection system, in a mobile learning context. The FGPE+ system, by integrating principles of Pareto optimization and gamification in a mobile-compatible platform, offered a unique, personalized, and engaging learning environment for pro- gramming students. The FGPE+ model's clean and user-friendly interface was another aspect that stood out, enabling learners to navigate and interact with the platform easily. The PWA-based system allowed learners to carry their learning environment with them, enhancing accessibility and convenience. The students appreciated the tailored exercise rec- ommendations, adaptive difficulty level adjustments, and gamification elements that kept them motivated to learn continually. The study also shed light on the effectiveness of the model in catering to diverse learners, including those who preferred different programming languages and had varying levels of skills and interests.

The overwhelmingly positive response to the FGPE+ model in our study is a promis- ing step towards transforming programming education, paving the way for an array of exciting future research opportunities. One immediate avenue for future research is to expand the scope of this investigation beyond the initial Polish and Lithuanian samples. Conducting comparative studies across different countries and cultures will provide a more comprehensive understanding of FGPE+'s cross-cultural efficacy and adaptability. This global approach can also reveal unique regional requirements or preferences, which can be integrated into the FGPE+ model to create a more universally effective learning platform. The current study focused on the learner's perspective. However, insights from educators who utilize FGPE+ could offer a different perspective, providing additional ways to improve and optimize the system. They could share valuable input on what works well in a classroom setting, potential areas of difficulty, or suggestions to improve learner engagement.

Additionally, the development of more sophisticated adaptive algorithms that lever- age artificial intelligence (AI) and machine learning (ML) techniques could augment the FGPE+'s capabilities. These techniques could further personalize the learning experience, making it more responsive to individual learner's needs. For instance, the system could predict what a student might struggle with based on historical data and preemptively provide resources to mitigate these challenges [87]. Incorporating Virtual Reality (VR) or Augmented Reality (AR) could also enhance the FGPE+ model's immersive learning experiences. VR/AR could be used to simulate real-world programming scenarios, making abstract programming concepts more tangible and engaging for learners.

The FGPE+ system, although robust, currently supports only a limited set of pro- gramming languages. Extending its support to encompass a broader range of languages, including emerging ones, will make it more versatile and valuable to a wider audience. Finally, we observe potential in exploring the impact of different gamification elements on learning outcomes. While FGPE+ currently uses a set of gamification techniques, under- standing which elements are most effective can help refine the system to maximize learner engagement and achievement.

In conclusion, the FGPE+ system is a promising approach to modernize programming education. By employing principles of gamification and Pareto optimization in a PWA platform, it provides an engaging, adaptive, and personalized learning experience. We believe that the positive results obtained highlight the system's potential for broader application. We anticipate that FGPE+ can be adapted to diverse educational contexts, playing a significant role in programming education across various age groups, skill levels, and cultural settings. However, more extensive and diverse studies are needed to validate these findings and refine the model to cater to a broader range of learners. By continuing to innovate and push boundaries, we believe that FGPE+ has the potential to revolutionize programming education. Through future research, the model could be further refined to make learning programming more accessible, engaging, and effective for all.

References

1. Mishra, L.; Gupta, T.; Shree, A. Online teaching-learning in higher education during lockdown period of COVID-19 pandemic.
Int. J. Educ. Res. Open 2020, 1, 100012. [CrossRef]
2. Breiki, M.A.; Yahaya, W.A.J.W. Using Gamification to Promote Students’ Engagement While Teaching Online During COVID-19. In Teaching in the Post COVID-19 Era; Springer International Publishing: Cham, Switzerland, 2021; pp. 443–453. [CrossRef]
3. Pedro, L.; Santos, C. Has Covid-19 emergency instruction killed the PLE? In Proceedings of the Ninth International Conference on Technological Ecosystems for Enhancing Multiculturality (TEEM'21), Barcelona, Spain, 26–29 October 2021; ACM: New York, NY, USA, 2021. [CrossRef]
4. Redondo, R.P.D.; Rodríguez, M.C.; Escobar, J.J.L.; Vilas, A.F. Integrating micro-learning content in traditional e-learning platforms.
Multimed. Tools Appl. 2020, 80, 3121–3151. [CrossRef]
5. Jayalath, J.; Esichaikul, V. Gamification to Enhance Motivation and Engagement in Blended eLearning for Technical and Vocational Education and Training. Technol. Knowl. Learn. 2020, 27, 91–118. [CrossRef]
6. da Silva, J.P.; Silveira, I.F. A systematic review on open educational games for programming learning and teaching. Int. J. Emerg. Technol. Learn. 2020, 15, 156–172. [CrossRef]
7. Paiva, J.C.; Leal, J.P.; Queirós, R. Fostering programming practice through games. Information 2020, 11, 498. [CrossRef]
8. Maryono, D.; Budiyono, S.; Akhyar, M. Implementation of Gamification in Programming Learning: Literature Review. Int. J. Inf. Educ. Technol. 2022, 12, 1448–1457. [CrossRef]
9. Maskeliu¯ nas, R.; Kulikajevas, A.; Blažauskas, T.; Damaševicˇius, R.; Swacha, J. An interactive serious mobile game for supporting the learning of programming in javascript in the context of eco-friendly city management. Computers 2020, 9, 102. [CrossRef]
10. Cuervo-Cely, K.D.; Restrepo-Calle, F.; Ramírez-Echeverry, J.J. Effect Of Gamification On The Motivation Of Computer Program- ming Students. J. Inf. Technol. Educ. Res. 2022, 21, 001–023. [CrossRef]
11. Chinchua, S.; Kantathanawat, T.; Tuntiwongwanich, S. Increasing Programming Self-Efficacy (PSE) Through a Problem-Based Gamification Digital Learning Ecosystem (DLE) Model. J. High. Educ. Theory Pract. 2022, 22, 131–143.
12. Ašeriškis, D.; Damaševicˇius, R. Gamification Patterns for Gamification Applications. Procedia Comput. Sci. 2014, 39, 83–90. [CrossRef]
13. Panskyi, T.; Rowin´ska, Z. A Holistic Digital Game-Based Learning Approach to Out-of-School Primary Programming Education.
Inform. Educ. 2021, 20, 1–22. [CrossRef]
14. Swacha, J.; Maskeliu¯ nas, R.; Damaševicˇius, R.; Kulikajevas, A.; Blažauskas, T.; Muszyn´ska, K.; Miluniec, A.; Kowalska, M. Introducing sustainable development topics into computer science education: Design and evaluation of the eco jsity game. Sustainability 2021, 13, 4244. [CrossRef]
15. Damaševicˇius, R.; Maskeliu¯ nas, R.; Blažauskas, T. Serious Games and Gamification in Healthcare: A Meta-Review. Information
2023, 14, 105.
16. Francillette, Y.; Boucher, E.; Bouchard, B.; Bouchard, K.; Gaboury, S. Serious games for people with mental disorders: State of the art of practices to maintain engagement and accessibility. Entertain. Comput. 2021, 37, 100396. [CrossRef]
17. Zhao, D.; Muntean, C.H.; Chis, A.E.; Rozinaj, G.; Muntean, G. Game-Based Learning: Enhancing Student Experience, Knowledge Gain, and Usability in Higher Education Programming Courses. IEEE Trans. Educ. 2022, 65, 502–513. [CrossRef]
18. Mohanarajah, S.; Sritharan, T. Shoot2learn: Fix-And-Play Educational Game For Learning Programming; Enhancing Student Engagement By Mixing Game Playing And Game Programming. J. Inf. Technol. Educ. Res. 2022, 21, 639–661. [CrossRef] [PubMed]
19. Xinogalos, S.; Satratzemi, M. The Use of Educational Games in Programming Assignments: SQL Island as a Case Study. Appl. Sci.
2022, 12, 6563. [CrossRef]
20. Barmpakas, A.; Xinogalos, S. Designing and Evaluating a Serious Game for Learning Artificial Intelligence Algorithms: SpAI War as a Case Study. Appl. Sci. 2023, 13, 5828. [CrossRef]
21. Costa, J.M. Using game concepts to improve programming learning: A multi-level meta-analysis. Comput. Appl. Eng. Educ. 2023,

31, 1098–1110. [CrossRef]
22. Soboleva, E.V.; Suvorova, T.N.; Grinshkun, A.V.; Bocharov, M.I. Applying Gamification in Learning the Basics of Algorithmization and Programming to Improve the Quality of Students' Educational Results. Eur. J. Contemp. Educ. 2021, 10, 987–1002.
23. Toda, A.M.; Valle, P.H.D.; Isotani, S. The Dark Side of Gamification: An Overview of Negative Effects of Gamification in Education. In Communications in Computer and Information Science; Springer International Publishing: Cham, Switzerland, 2018; pp. 143–156. [CrossRef]
24. Imran, H. An Empirical Investigation of the Different Levels of Gamification in an Introductory Programming Course. J. Educ. Comput. Res. 2022, 61, 847–874. [CrossRef]

25. Chatterjee, S.; Majumdar, D.; Misra, S.; Damaševičius, R. Adoption of mobile applications for teaching-learning process in rural girls' schools in India: An empirical study. Educ. Inf. Technol. 2020, 25, 4057–4076. [CrossRef]
26. Tuparov, G.; Keremedchiev, D.; Tuparova, D.; Stoyanova, M. Gamification and educational computer games in open source learning management systems as a part of assessment. In Proceedings of the 2018 17th International Conference on Information Technology Based Higher Education and Training (ITHET), Olhao, Portugal, 26–28 April 2018; pp. 1–5. [CrossRef]
27. Pérez-Berenguer, D.; García-Molina, J. A standard-based architecture to support learning interoperability: A practical experience in gamification. Software Pract. Exp. 2018, 48, 1238–1268. [CrossRef]
28. Calle-Archila, C.R.; Drews, O.M. Student-Based Gamification Framework for Online Courses. In Communications in Computer and Information Science; Springer International Publishing: Cham, Switzerland, 2017; pp. 401–414. [CrossRef]
29. Sheppard, D. Introduction to Progressive Web Apps. In Beginning Progressive Web App Development; Apress: New York, NY, USA, 2017; pp. 3–10. [CrossRef]
30. Hajian, M. PWA with Angular and Workbox. In Progressive Web Apps with Angular; Apress: New York, NY, USA, 2019; pp. 331–345. [CrossRef]
31. Devine, J.; Finney, J.; de Halleux, P.; Moskal, M.; Ball, T.; Hodges, S. MakeCode and CODAL: Intuitive and efficient embedded systems programming for education. J. Syst. Archit. 2019, 98, 468–483. [CrossRef]
32. Lee, J.; Kim, H.; Park, J.; Shin, I.; Son, S. Pride and Prejudice in Progressive Web Apps. In Proceedings of the 2018 ACM SIGSAC Conference on Computer and Communications Security, Toronto, ON, Canada, 15–19 October 2018; ACM: New York, NY, USA, 2018. [CrossRef]
33. FGPE PLE Environment. Available online: https://github.com/FGPE-Erasmus/fgpe-ple-v2 (accessed on 10 June 2023).
34. Sutadji, E.; Hidayat, W.N.; Patmanthara, S.; Sulton, S.; Jabari, N.A.M.; Irsyad, M. Measuring user experience on SIPEJAR as e-learning of Universitas Negeri Malang. IOP Conf. Ser. Mater. Sci. Eng. 2020, 732, 012116. [CrossRef]
35. Nah, F.F.H.; Zeng, Q.; Telaprolu, V.R.; Ayyappa, A.P.; Eschenbrenner, B. Gamification of Education: A Review of Literature. In
Lecture Notes in Computer Science; Springer International Publishing: Cham, Switzerland, 2014; pp. 401–409. [CrossRef]
36. Barik, T.; Murphy-Hill, E.; Zimmermann, T. A perspective on blending programming environments and games: Beyond points, badges, and leaderboards. In Proceedings of the 2016 IEEE Symposium on Visual Languages and Human-Centric Computing (VL/HCC), Cambridge, UK, 4–8 September 2016; pp. 134–142. [CrossRef]
37. Prokhorov, A.V.; Lisovichenko, V.O.; Mazorchuk, M.S.; Kuzminska, O.H. Developing a 3D quest game for career guidance to estimate students' digital competences. CEUR Workshop Proc. 2020, 2731, 312–327. . [CrossRef]
38. Padilla-Zea, N.; Gutiérrez, F.L.; López-Arcos, J.R.; Abad-Arranz, A.; Paderewski, P. Modeling storytelling to be used in educational video games. Comput. Hum. Behav. 2014, 31, 461–474. [CrossRef]
39. Hadzigeorgiou, Y. Narrative Thinking and Storytelling in Science Education. In Imaginative Science Education; Springer Interna- tional Publishing: Cham, Switzerland, 2016; pp. 83–119. [CrossRef]
40. Kusuma, G.P.; Wigati, E.K.; Utomo, Y.; Suryapranata, L.K.P. Analysis of Gamification Models in Education Using MDA Framework. Procedia Comput. Sci. 2018, 135, 385–392. [CrossRef]
41. Wu, M.; Liao, C.C.; Chen, Z.H.; Chan, T.W. Designing a Competitive Game for Promoting Students' Effort-Making Behavior by Virtual Pets. In Proceedings of the 2010 Third IEEE International Conference on Digital Game and Intelligent Toy Enhanced Learning, Kaohsiung, Taiwan, 12–16 April 2010; pp. 234–236. [CrossRef]
42. Chen, Z.H.; Liao, C.; Chien, T.C.; Chan, T.W. Animal companions: Fostering children's effort-making by nurturing virtual pets.
Br. J. Educ. Technol. 2009, 42, 166–180. [CrossRef]
43. Slavin, R.E. Cooperative Learning: Applying Contact Theory in Desegregated Schools. J. Soc. Issues 1985, 41, 45–62. [CrossRef]
44. Zakaria, E.; Iksan, Z. Promoting Cooperative Learning in Science and Mathematics Education: A Malaysian Perspective.
EURASIA J. Math. Sci. Technol. Educ. 2007, 3, 35–39. [CrossRef]

45. Correia, A.; Fonseca, B.; Paredes, H.; Martins, P.; Morgado, L. Computer-Simulated 3D Virtual Environments in Collaborative Learning and Training: Meta-Review, Refinement, and Roadmap. In Progress in IS; Springer International Publishing: Cham, Switzerland, 2016; pp. 403–440. [CrossRef]