



Research article

Large-scale experiment in STEM education for high school students using artificial intelligence kit based on computer vision and Python

Meechai Lohakan^{*}, Choochat Seetao*Department of Teacher Training in Electrical Engineering, Faculty of Technical Education, King Mongkut's University of Technology North Bangkok, Bangkok, Thailand*

ARTICLE INFO

Keywords:

Computer vision
Artificial intelligence
Electrical engineering education
STEM education
Interdisciplinary

ABSTRACT

This study proposes an artificial intelligence (AI) kit for high school students in science, technology, engineering, and mathematics (STEM). The AI kit includes an edge AI machine and electronic components. A compact, purpose-built kit resembling a laptop was designed for ease of replication and portability. Utilizing pre-trained convolutional neural network models and computer vision algorithms, five Thai schools participated in on-site instructions. A quasi-experimental study assessed the students' learning outcomes using a paired sample *t*-test. Results revealed improved knowledge and reduced score variation. Additionally, gender analysis confirmed that both male and female students met the learning criteria. The students expressed satisfaction with the distinctive hardware and learning method employed during the class activities. Notably, the test results demonstrated that the AI kit enhanced students' enthusiasm and facilitated comprehension.

1. Introduction

Science, technology, engineering, and mathematics (STEM) education, as a meta-disciplinary framework, integrates independent knowledge to assist students in solving real-world problems. It equips students with essential technical expertise while emphasizing crucial soft skills vital for leadership and effective project management. Educators seek novel approaches to boost student motivation and foster learner engagement [1]. A study investigated female STEM instructors' representation in secondary education in Columbia from 2010 to 2013. Researchers analyzed the correlation between female STEM teachers' exposure during secondary education and female enrolment in tertiary STEM program [2].

A quantitative study in Vietnam explored STEM education. The findings correlate teachers' attitudes with the tension between their beliefs and teaching goals, which are directly linked to local cultural values and expectations [3].

A meta-analysis of computer-supported collaborative learning (CSCL) effects in STEM education, based on studies conducted between 2005 and 2014, revealed insight regarding the way factors interact with overall CSCL effectiveness [4].

Regarding early childhood education, a curriculum design for artificial intelligence was proposed in China [5]. This study established a framework encompassing essential components: aim, subject matter, method, and evaluation. The authors advocated for artificial intelligence (AI) literacy, emphasizing three core competencies: AI knowledge, AI skill, and AI attitude.

^{*} Corresponding author.

E-mail addresses: meechailo@kmutnb.ac.th (M. Lohakan), choochats@kmutnb.ac.th (C. Seetao).

A study developed a compact Arduino-based platform for monitoring student cardiac functions in educational settings [6]. This integrated solution bridges the gap between education and technology, fostering proper engineering approaches and attitudes among future science, technology, and engineering students.

AI is critical to advancing computer vision. Its applications span diverse domains including, gaming [7–9], medical treatments [10–12], and autonomous driving [13–15]. An embedded system utilizing real-time AI was developed for apple detection in orchards. The study investigated edge AI boards and employed transfer learning with the YOLOv3-tiny framework to detect, count, and measure apple sizes. The results demonstrate significant potential for real-time power consumption monitoring [16].

An article on virtual reality (VR)-supported instructional design for primary and secondary STEM education is reviewed. The authors recommend that instructional designers either develop innovative VR applications or integrate existing approaches into their teaching methods [17].

Animal repelling using embedded edge AI was designed to safeguard crops. This study employs AI hardware for precision agriculture, safeguarding crops against threats, and minimizing production losses. Various edge-computing devices have undergone rigorous evaluation for cost-effectiveness and performance optimization [18].

Educators aim to employ AI for engineering student instruction. The AI curriculum targets students in Grades 7 to 9 at the Chinese University of Hong Kong [19]. This study introduces a co-creation process for assessing junior secondary curricula. It analyses the experimental results using descriptive statistics, thereby significantly contributing to the existing literature by presenting an effective curriculum evaluation method.

Scratch, a low-code programming language, has been recommended for high school students. The authors introduced an academic course covering AI systems and robotic fundamentals [20,21]. Their findings demonstrate that this course significantly improved students' comprehension of AI algorithms. Furthermore, students could analyze AI algorithms and grasp computational concepts related to the AI process.

The proposed cutting-edge technologies enable interdisciplinary STEM education. An electronic board facilitated on-site structural experiments for civil engineering students [22]. These components addressed digital-era challenges in engineering education. The research team advocates implementing this approach across STEM disciplines.

The Digital Economy Promotion Agency (DEPA) in Thailand initiated a campaign to encourage high school students to acquire coding competencies. DEPA emphasizes that adolescents in the digital age should possess problem-solving skills, critical thinking abilities, and innovative idea development [23]. Academic and classroom environments must provide coding skill-building opportunities for student engagement.

This study introduced innovative AI kits for computer-based AI systems in STEM education. These kits utilize electronic components to create an all-in-one experimental platform. Additionally, the learning process incorporates the motivation information application progress (MIAP) model [24]. Students receive instruction in Python programming and technological hardware.

1.1. Research objectives

- To design and construct AI prototypes tailored for high school students.
- To monitor student engagement with an AI kit during instruction.
- To assess high school students' learning outcomes using an AI kit and an active learning model.
- To investigate AI-related learning outcomes in high school students with a focus on gender.

1.2. Research question

The active learning model and AI kits were evaluated using the following research questions:

RQ1: How do students engage with the AI kit?

RQ2: Can MIAP model and AI kits enhance student learning achievements?

RQ3: Is gender associated with learning achievement in AI systems?

1.3. Research hypothesis

The study posits the following hypotheses.

- Students satisfy 80 % of the learning criteria through their proficiency in software and hardware.
- Learner satisfaction through the MIAP model is significantly high.
- No significant gender difference exists in AI learning achievement.

2. Hardware design

The AI kit design incorporates portable, durable, and stackable features. A custom-designed printed circuit board (PCB) precisely accommodates the Jetson Nano [25–28], sensors, and protoboard. Additionally, the edge device seamlessly interfaces with peripherals such as keyboards, electronic components, and cameras.

2.1. Structure design

Fig. 1 depicts an experimental box created using SOLIDWORKS, a computer-aided design (CAD) program. The laptop-like frame and base customized from colored plastic were precisely laser-cut to achieve a perfect fit. The form factor allows easy replication for mass production with a portable monitor connected via a standard high-definition multimedia interface (HDMI) port on the lid.

Fig. 2 depicts the top view of the box, with a hinged lid on both sides. The lid's laptop-friendly design allows for space-saving functionality. The material covering the box is black gloss plastic, laser-engraved with the "STEM Education" logo. The experimental box measures $28.5 \times 21.5 \times 8.5$ cm, as illustrated in Fig. 3.

2.2. PCB design and electronic components

The PCB layout features an L-shaped, as depicted in Fig. 4. The front and bottom sides of the mainboard are uniquely fabricated from an FR-4 copper-clad PCB material. The device legend on the front side is labelled to ensure accurate placement.

The Jetson Nano can be inserted into the top-left corner, aligning with the basement, as depicted in Fig. 5. The Jetson Nano's general-purpose input/output (GPIO) port is connected to external pins on the board. In the PCB design, a central protoboard facilitated wiring. Power was delivered via a single direct current (DC) adapter connected to the top-right corner. This approach eliminates the necessity for an alternating current (AC) outlet, enhancing convenience during teaching.

In the experimental box, electronic components were included. Each part had a jumper pin-head connector for convenient wiring during experiments. Additionally, to safeguard against damage, each part is equipped with a small switch for circuit control. Fig. 6 illustrates the positions of the light-emitting diode (LED) and buzzer. The first LED circuit consists of eight bits (D1 to D8). The second features an RGB LED (D9) capable of emitting red, green, or blue light. Additionally, the board includes both active and passive buzzers [29]. Fig. 7 depicts the 7-Segment, while Fig. 8 illustrates the tactile switches, relays, and servo motors.

This study utilized the MCP3008 Analog-to-digital converter (ADC), which employs a serial peripheral interface. This chip effectively converts analog signals to digital signals and features eight channels, as depicted in Fig. 9. This section incorporates a light-dependent resistor (LDR) and a potentiometer.

3. Edge-AI device

The Jetson Nano B01 from NVIDIA, an AI computer, was chosen as an edge device for computer vision and AI tasks. Table 1 provides detailed specifications and features of the Jetson Nano. Regarding AI applications, the Jetson Nano, with its 128-core Maxwell graphical processing unit (GPU), outperforms the Raspberry Pi 4. This efficiency advantage makes the Jetson Nano a preferred choice for AI workloads, leveraging its compute unified device architecture (CUDA) technology.

In this study, the Jetson Nano was positioned at the base of the experimental box and shielded with plastic to prevent unforeseen incidents during teaching. The GPIO ports were wired using a 40-pin cable on the PCB board to facilitate convenient connection and maintenance, as depicted in Fig. 10.

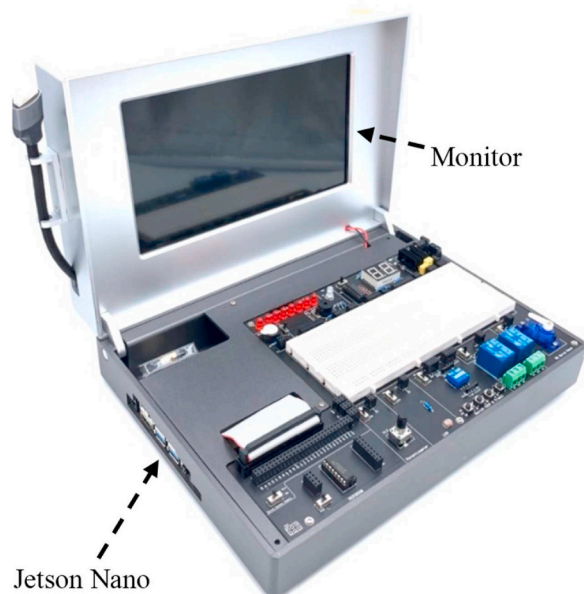


Fig. 1. Front view of an experimental box.

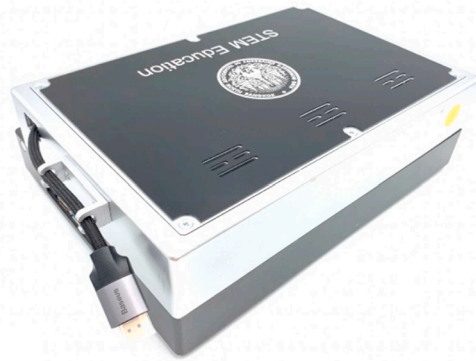


Fig. 2. Top view and lid.



Fig. 3. Dimension of experimental box.

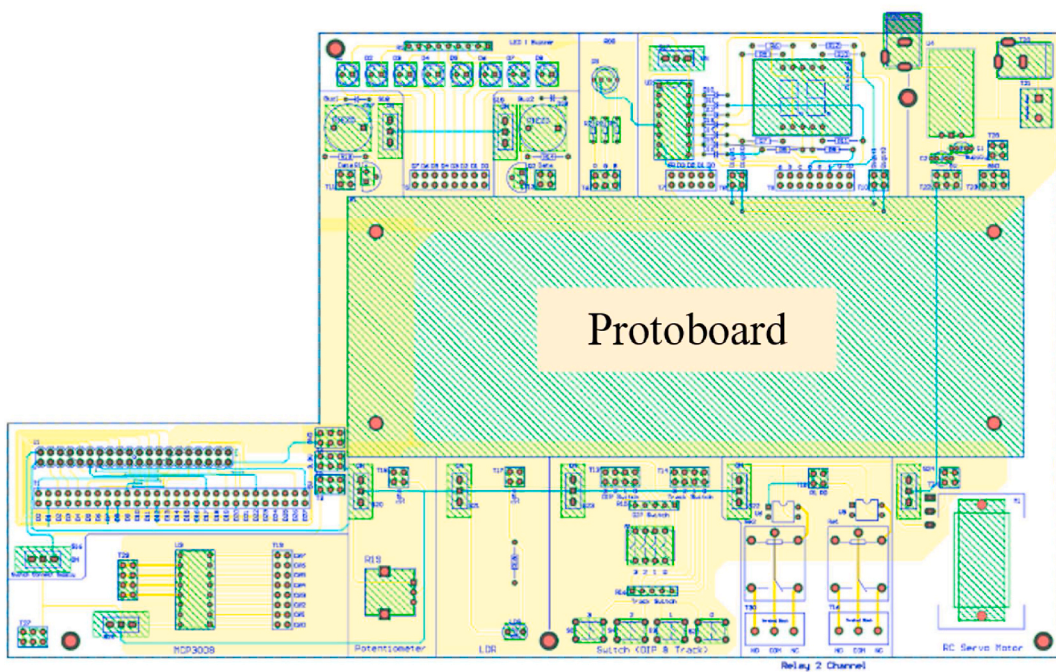


Fig. 4. PCB layout.

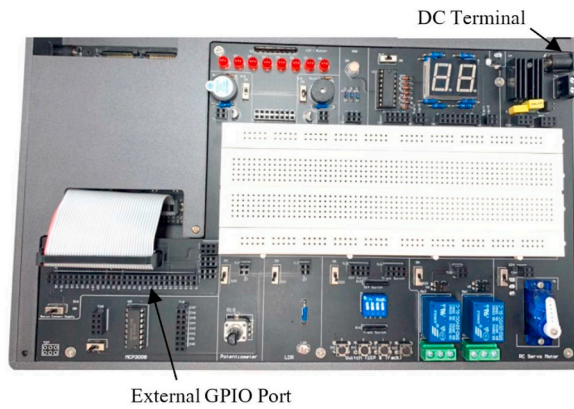


Fig. 5. Component installation.



Fig. 6. LED and buzzer.

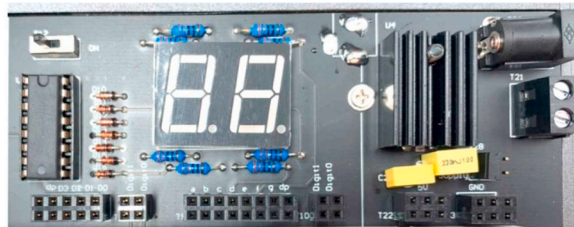


Fig. 7. Seven-segment installation.

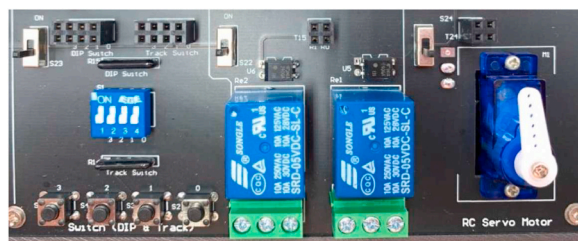


Fig. 8. Tactile switches, relays, and servo motors.

4. Software for computer vision and AI

The selected software concept, JetPack, is used under a public educational license and is thus not a copyright violation. JetPack, developed by NVIDIA, serves as a development environment for the Jetson Nano. It includes a Linux kernel, device driver, and pre-

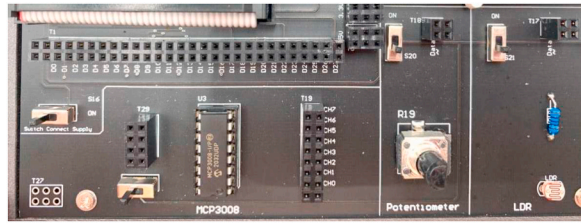


Fig. 9. ADC, LDR, and potentiometer.

Table 1
Jetson Nano features.

CPU	64-bit Quad-Core Arm 1.43 GHz
GPU	128 CUDA core
Memory	4 GB 64-bit LPDDR4 25.6 GB/s
Video Port	HDMI
Camera	2 × CSI
Networking	Gigabit Ethernet (RJ45)
USB	4 USB 3.0

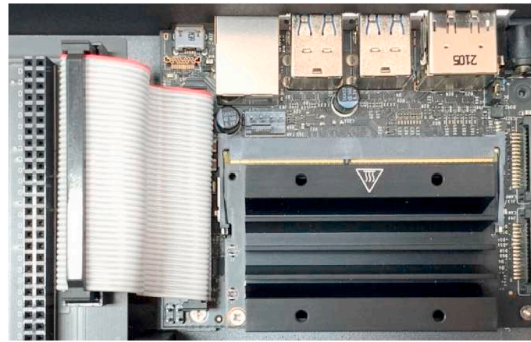


Fig. 10. Jetson Nano installed at the base of the experimental box.

bundled components such as the open computer vision library (OpenCV) [30–33] and Python [34–36]. These components are tailored specifically for the Jetson Nano platform.

This study employed OpenCV, a powerful open-source library for computer vision, to handle real-time video data (VDO) in AI and machine learning applications using Python. Table 2 provides a succinct summary of essential OpenCV statements in Python, including library import, camera configuration, frame display, and geometric visualization.

This study focused on leveraging pre-trained AI models for computer vision, simplifying the learning process for beginners, as depicted in Fig. 11. The output of the pre-trained model is transmitted to IO devices, including servo motors and LEDs.

The Python code for the AI model is presented in Table 3. The algorithm initializes the model using cv2.dnn.readNetFromCaffe() based on a pre-trained model. The captured frame underwent preprocessing using the cv2.dnn.blobFromImage function, involving

Table 2
Python statements for computer vision using OpenCV.

Python Statement	Description
import cv2	Import OpenCV library
cap = cv2.VideoCapture()	Set camera ID for cap
ret, frame = cap.read()	Capture a single frame
cv2.imshow('Input', frame)	Show a captured frame
cap.release()	Close camera
img = cv2.imread('lena.jpg')	Read image from file
cv2.rectangle()	Draw a rectangle in VDO or image
cv2.putText()	Add text in VDO or image
ed = cv2.Canny()	Detect edge using Canny Method
blur = cv2.GaussianBlur()	Blur image
cw = imutils.rotate(img,45)	Rotate image CCW 45°

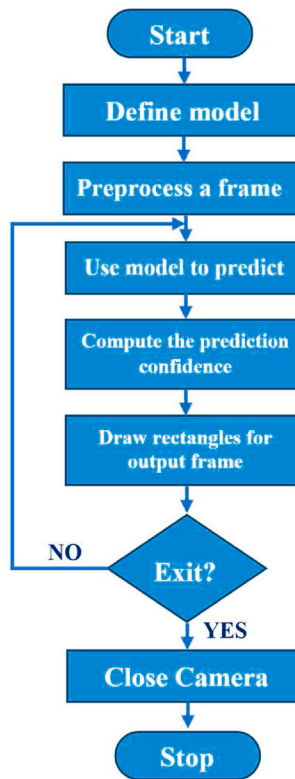


Fig. 11. Pre-trained model algorithm.

Table 3

Python statements for prediction from pre-train model.

Python Statement	Description
<code>net = cv2.dnn.readNetFromCaffe()</code>	Set a pre-trained model
<code>blob = cv2.dnn.blobFromImage()</code>	Preprocessing input image
<code>net.setInput(blob)</code>	Pass blob to CNN
<code>detections = net.forward()</code>	Perform prediction process
<code>confidence = detections[0, 0]</code>	Draw the confidence

mean subtraction and image scaling. In preparation for subsequent computation, an image must undergo suitable preprocessing. An OpenCV deep learning module (DNN) is employed to detect and predict objects within the image frame. The results of this processing are fed into the `net.forward` statement. Rectangles are drawn when confidence values surpass the specified threshold.

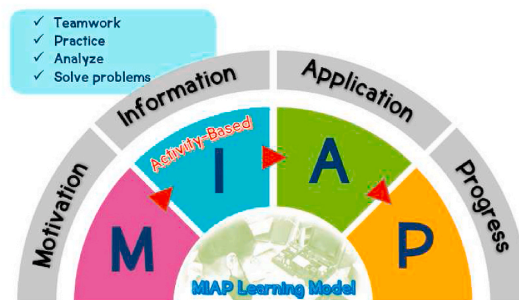


Fig. 12. MIAP learning model.

5. Active learning model

This study employed the MIAP learning model as an active learning approach. The model focuses on fostering learner growth, transitioning “Cannot Do It” to “Able to Do It”. Expected student behaviors encompass practical skills, positive attitudes, and knowledge. MIAP learning consists of four distinct components, as depicted in Fig. 12. The model can be described as follows.

- Motivation (M) facilitates student learning by inspiring them to comprehend new material and tackle challenges. Consequently, learners exhibit a heightened enthusiasm for studying.
- Information (I) imparts knowledge to students within the course objectives.
- Application (A) involves applying acquired knowledge and skills to identify and solve problems.
- Progress (P) serves as a metric for evaluating student learning performance. Learning achievements are assessed via problems and exercises.

The MIAP learning model commences with an introductory lesson (M) on real-world AI system applications. Subsequently, Python programming and hardware content (I) are imparted through an activity and an AI kit. Instructors mandated students to complete the exercise (A) and submit results via the mobile application. During the final step (P), teachers assess student knowledge and provide guidance. Upon completion of the MIAP procedure for each module, results are returned to the instructor. If learning outcomes are suboptimal, the teacher should revisit the information, application, and progress steps to enhance student comprehension, as depicted in Fig. 13.

6. Methodology

6.1. Participants

For this study, all participants consented to image publication, and parental consent was obtained before data collection. The investigation procedure was comprehensively explained. The study received authorization from the school director and was approved by the Human Research Ethics Committee of the STEM Education Center, King Mongkut’s University of Technology North Bangkok, Thailand (Reference Number: STEM-01-2566). This study involved a large-scale experiment with high school students (Grade 10) from five schools in Thailand. A total of 149 participants (65 males and 84 females) took part. This study employed purposive sampling to investigate students in science and mathematics programs. The average participant age was 16 years. The experiment took place between June 2023 and August 2023. Table 4 presents detailed information for each school.

6.2. Procedure

This study conducted a one-day course from 08:30 to 16:30. The Line application was utilized for students, and students were mandated to provide feedback via the mobile application. A quasi-experiment [37,38] with a one-group pretest and posttest was performed to measure learning achievement in the course. The experimental procedure comprised nine steps, as depicted in Fig. 14. The pretest and posttest questions comprised 20 multiple-choice questions, assessing knowledge related to four key AI concepts with scores ranging from 0 to 20. The study employed questions to evaluate learning progress in four domains: Python programming (four questions), computer vision (four questions), hardware interface (four questions), and AI application using the experimental box (eight

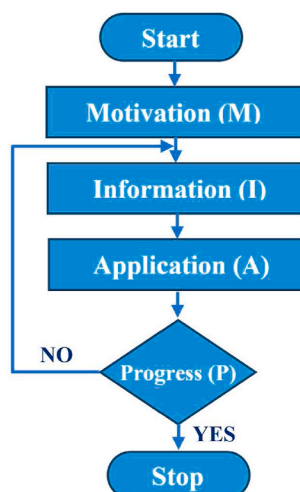


Fig. 13. Feedback process of MIAP model.

Table 4
Participant detail.

School	Grade	Male	Female	Total
Satrinonthaburi	10	0	30	30
Anurat Prasit	10	17	15	32
Samsenwittayalai	10	9	6	15
Wat Khemampirataram	10	21	17	38
Satriwitthaya 2	10	18	16	34
Total		65	84	149

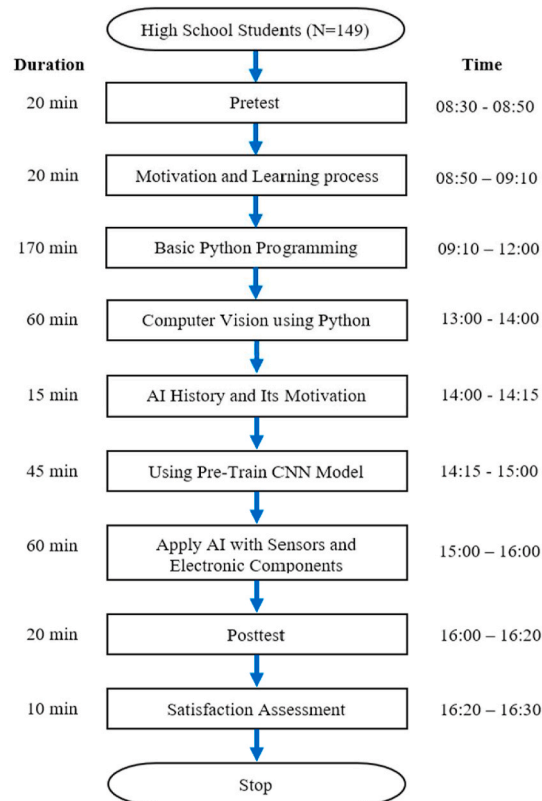


Fig. 14. Experimental procedure.

questions). The test assessed participants' ability to evaluate students' learning achievements following instructional guidance.

In the initial phase, students completed a pretest within a 20-min timeframe. This step, which encouraged the use of the experimental box and learning resources, fostered student motivation and inspiration. Students received instruction in fundamental Python programming and were required to submit assignments via the Line application. The teacher provided guidance on Python scripting based on the e-book, assisting students in algorithm implementation. The entire process lasted 170 min. In the subsequent phase, a 1-h session on Python-based computer vision was presented to the students. Within 15 min, the historical context and motivation behind AI were introduced. Subsequently, the students were instructed on Python-based training for pre-trained AI models. Finally, the operation of sensors and electronic components was conducted using Python for 60 min. This stage integrates the AI process with hardware devices. In the eighth phase, students completed a 20-min posttest and 10-min satisfaction assessment, maintaining consistency by using identical questions from the pretest in the first phase [37].

6.3. Educational statistic and measurement tools

A quasi-experimental design was used to evaluate the effectiveness of the proposed learning approach. The educational statistics used in this study were *t*-test, data frequency, average value, and standard deviation (SD). A *t*-test with paired differences was performed to determine learning achievement in each school. The IBM SPSS statistical software was used to process the data. The measurement tools used were an AI kit, pretest, posttest, and questionnaire. Pretest and posttest were used to assess students' learning achievements. Both tests included 20 multiple-choice questions. Questionnaire items were measured using a 5-point Likert scale

ranging from 1 (strongly disagree) to 5 (strongly agree) [39] for satisfaction evaluation.

7. Research results

7.1. On-site teaching

In five schools, on-site teaching involved 15 student groups, each assigned to an experimental box equipped with a keyboard, mouse, and USB camera. Two teaching assistants provided guidance to students. Fig. 15 illustrates student activities during Python programming using Visual Studio Code (VS Code).

The students received instructions on utilizing OpenCV for computer vision tasks and AI functions. The VDO frame is generated by an AI algorithm that captures a single frame using a camera. One of the tasks involves adding rectangles and text to the frame at the detected face position using a face-detection model. Subsequently, students' exercise outcomes were transmitted via a Line application, as depicted in Fig. 16. Furthermore, they exhibited proficiency in coding AI concepts using the following algorithms.

- Face detection using Haar cascade [40–42],
- Vehicle detection,
- Face detection using deep learning models and
- Real-time object detection.

7.2. Learning achievement

In this study, paired-sample tests were utilized to assess learning achievement by comparing pretest and posttest scores. The learning achievements of all schools according to the large-scale experiment are listed in Tables 5–9. The results for each school revealed significant differences in learning performance ($p < 0.05$).

The overall learning achievement of all schools ($N = 149$) is presented in Table 10. The results showed significant differences in learning performance, with $t = 23.81$ ($p < 0.05$). After the procedure, students' average scores increased from 11.15 to 18.67, with the SD decreasing from 3.76 to 2.92. This aspect implies that students can enhance their knowledge and have less variation in mean scores.



Fig. 15. Student activities at schools.



Fig. 16. Assignments to handle the face detection algorithm.

Table 5
Learning achievement at Satrinonthaburi school.

Pretest (n = 30)		Posttest (n = 30)		Paired Differences			df	t
M	SD	M	SD	M	SD	Std. Err Mean		
10.53	3.44	19.13	1.25	8.60	3.37	0.62	29	13.98

* $p < 0.05$.

Table 6
Learning achievement at Anurat Prasit school.

Pretest (n = 32)		Posttest (n = 32)		Paired Differences			df	t
M	SD	M	SD	M	SD	Std. Err Mean		
11.44	3.42	18.75	1.74	7.31	3.32	0.59	31	12.47

* $p < 0.05$.

Table 7
Learning achievement at Samsenwittayalai school.

Pretest (n = 15)		Posttest (n = 15)		Paired Differences			df	t
M	SD	M	SD	M	SD	Std. Err Mean		
13.87	3.96	18.93	1.83	5.07	3.01	0.78	14	6.51

* $p < 0.05$.

Table 8
Learning achievement at Wat Khemampirataram school.

Pretest (n = 38)		Posttest (n = 38)		Paired Differences			df	t
M	SD	M	SD	M	SD	Std. Err Mean		
10.68	3.64	18.26	2.72	7.58	4.11	0.67	37	11.36

* $p < 0.05$.

Table 9
Learning achievement at Satriwitthaya 2 school.

Pretest (n = 34)		Posttest (n = 34)		Paired Differences			df	t
M	SD	M	SD	M	SD	Std. Err Mean		
10.76	4.03	18.53	1.99	7.76	4.42	0.67	33	10.23

* $p < 0.05$.

Table 10
Learning achievement of all schools.

Pretest (n = 149)		Posttest (n = 149)		Paired Differences			df	t
M	SD	M	SD	M	SD	Std. Err Mean		
11.15	3.76	18.67	2.92	7.52	3.85	0.76	148	23.81

* $p < 0.05$.

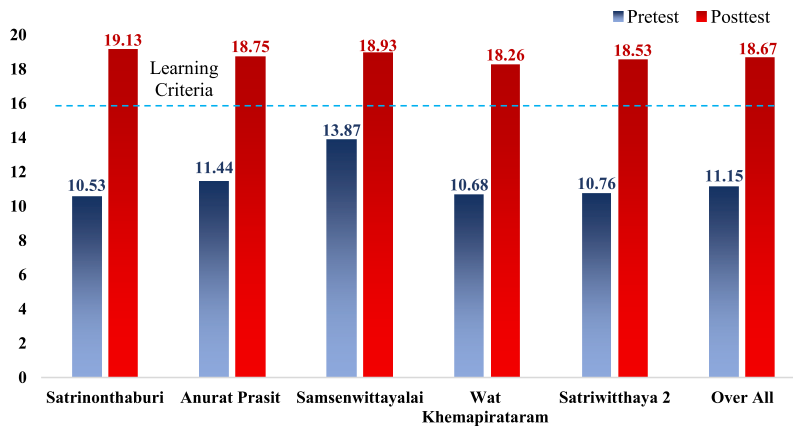


Fig. 17. Learning criteria in each school.

Fig. 17 presents bar graphs depicting the alignment of learning achievements with the established learning criteria. The learning criterion set for this study was 80 % [43] (equivalent to 16 points). Pre-procedure, all students scored below 80 %. Post-procedure, their learning achievement exceeded the criteria by 80 %, confirming the effectiveness of a practical teaching technique using Python programming for AI topics in STEM education.

Fig. 18 depicts data scattering characteristics for the entire student cohort (N = 149). The pretest group (blue dot) was significantly lower than the posttest group (red dot). Despite satisfying the posttest learning criteria, certain students need additional intervention to enhance their knowledge and skills beyond 80 %. Subsequent learning revisions were necessary following in-class teaching via online content.

Tables 11 and 12 present learning achievement results by student gender, with $t = 12.47$ for male and $t = 21.01$ for female ($p < 0.05$). Notably, both male and female students achieved the 80 % learning criteria, as depicted in **Fig. 19**.

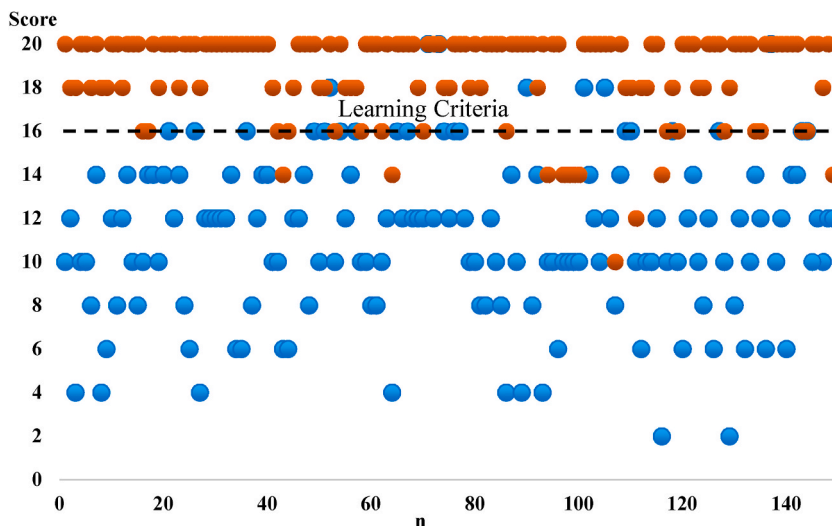


Fig. 18. Data scattering of pretest (blue dot) and posttest (red dot). (For interpretation of the references to color in this figure legend, the reader is referred to the Web version of this article.)

Table 11
Learning achievement for male students.

Pretest (n = 65)		Posttest (n = 65)		Paired Differences			df	t
M	SD	M	SD	M	SD	Std. Err Mean		
11.44	3.42	18.75	1.74	7.31	3.32	0.59	31	12.47

*p < 0.05.

Table 12
Learning achievement for female students.

Pretest (n = 84)		Posttest (n = 84)		Paired Differences			df	t
M	SD	M	SD	M	SD	Std. Err Mean		
10.69	3.31	18.83	1.87	8.14	3.55	0.39	83	21.01

*p < 0.05.

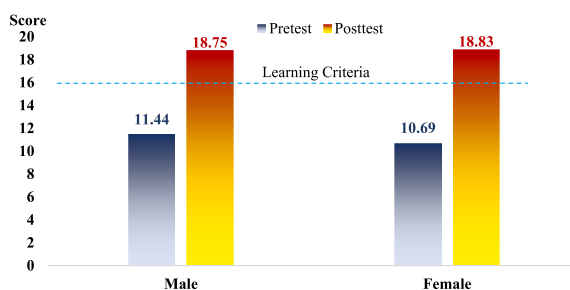


Fig. 19. Learning criteria compared between male and female.

7.3. Satisfaction evaluation of students

The conductive class environment fostered student engagement, curiosity, and energy, contributing to effective teaching and seamless class management. As presented in Table 13, the students were impressed with their studies. Subsequently, they reflected on their satisfaction assessments for each topic. All topics were assessed by the students (N = 149) as very high. Moreover, the SD was below 1, indicating satisfactory results.

8. Discussion

The significance of this study was the development of AI kits for teaching modern AI topics, including software and hardware systems. An active learning model was used in the learning process to achieve the learning outcomes.

8.1. RQ1: students' engagement behaviors

The on-site teaching results from five schools demonstrated significant student engagement in active activities using the MIAP model. This engagement was primarily focused on Python programming and an AI experimental box due to the practical-based approach. In the initial phase, students enhanced their Python programming skills for computer vision. Subsequently, they demonstrated increased engagement in AI programming related to electronic hardware, as depicted in Figs. 15 and 16. In alignment with the AI curriculum in early childhood education in China [5], students actively engage in questioning and collaborative experimentation during the class, rather than adopting passive approaches. In the STEM class, the teacher facilitated student learning by assigning a

Table 13
Satisfaction evaluation of students with N = 149.

Topics	M	SD	Interpretation
An experimental box	4.93	0.26	Very High
Learning facility	4.87	0.52	Very High
Knowledge for future study	4.61	0.64	Very High
Teaching procedure	4.73	0.46	Very High
Teaching assistant	4.71	0.61	Very High
Service	4.65	0.60	Very High

spectrum of exercises. Active learning, particularly effective in group settings, enabled students to acquire both software [20] and hardware skills for AI applications.

8.2. RQ2: students' learning achievement

This study developed and designed new AI kits and learning methods for STEM education, including software and hardware operations, learning activities, and educational evaluations. Pretest and posttest [37] were employed to assess students' AI competency. Learning achievements demonstrated that students improved their coding and AI knowledge skills. They boosted their critical thinking, creativity, communication, and cooperation skills after completing the group assignments, as indicated in Table 10. In addition, the students' average scores consistently exceeded the 80 % learning criteria [43], as depicted in Fig. 17.

8.3. RQ3: students' gender related to learning achievement

In this study, the student's gender was investigated. The findings from Tables 11 and 12 indicate no statistically significant disparity in learning achievement between male and female students studying AI, whether using Python programming or AI kits. As depicted in Fig. 19, the male student scores averaged 18.75, while the female scores averaged 18.83. Both genders demonstrated score improvement after completing the AI kit-based learning procedure. Ultimately, the teaching method and AI experimental box enhance learning outcomes for all genders.

9. Limitations

This study acknowledges certain limitations. Specifically, teachers are required to assemble various hardware components for AI kits, including wiring cables, external cameras, and power adaptors, before conducting their teaching activities. The optimal student-to-AI kit ratio is two. In addition, the Jetson Nano requires installation on an NVIDIA-specific operating system (OS). Prior to any instructional activities, related software, including VS Code, Python library, and AI dependencies, should be installed. Following software installations, an image file can be restored to replicate other AI kits. The classroom setup time before teaching was 20 min. Teaching assistants are necessary for large classrooms (over 30 students).

10. Practical implications

The findings of this study indicate that the AI kit can be used as an experimental device for various subjects, such as computer programming and the Internet of Things (IoT) for high school students. The AI kit is a computer-based box with low power consumption. No additional computers are required to operate. Because of its compact shape, the AI kit is suitable for both on-site training and classrooms and can be scaled to support large classrooms.

An interdisciplinary approach using an AI kit with MIAP for STEM classrooms has broad appeal. Project-based learning and problem-based learning can be applied to teaching using an AI kit. These findings suggest that the AI kit enhances student engagement in active learning.

11. Conclusion

Regarding sustainable learning and education, this study developed AI kits and instructional procedures to impart AI concepts for STEM education on a broad scale. The experimental box was tailored to facilitate diverse applications. Within this box, a specifically designed PCB accommodated the Jetson Nano as an edge AI device. On the assembled PCB, sensors and electronic components are meticulously soldered onto the panel. The enclosure houses a portable monitor and sensors within a single box, making it suitable for both on-site and online teaching. A laptop-style plastic case underwent precise laser shaping. Subsequently, students received pre-installed software and training to operate the hardware. OpenCV with Python programming was used to perform the real-time VDO for the AI task.

In this study, experiments were conducted across five high schools with Grade 10 students, representing a mix of male and female participants. Following the teaching process, the posttest scores exhibited a statistically significant improvement compared to the pretest scores. In addition, learning achievements exceeded the learning criteria. Students' satisfaction was also assessed. Students exhibit engagement behaviors in AI programming with electronic devices. They were satisfied with the hardware and learning processes during enjoyable class activities. No significant gender disparities were observed in learning achievements related to AI topics when utilizing AI kits. The study's outcome suggests that AI kits offer a promising avenue for enhancing both hardware and software competencies among high school students and those at higher education levels.

In future research, learning topics focusing on AI graphical programming with electronic components for industrial applications will be developed using low-cost devices. Specifically, a human machine interface (HMI) board with a Node-RED will be employed as an embedded device. A small size and built-in monitor are advantageous for this platform. Furthermore, custom training techniques for various frameworks will be included in the learning kits for STEM or STEAM education for high school and vocational students in Thailand.

Data availability statement

The data that support the findings of this study are openly available on the Open Science Framework at <https://osf.io/f3jvq/>, DOI 10.17605/OSF.IO/F3JVQ, and <https://bit.ly/3VGpd5Y>.

CRedit authorship contribution statement

Meechai Lohakan: Writing – review & editing, Writing – original draft, Software, Methodology, Data curation. **Choochat Seetao:** Resources, Project administration, Conceptualization.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

- [1] D. Zhao, et al., An innovative multi-layer gamification framework for improved STEM learning experience, *IEEE Access* 10 (2022) 3879–3889, <https://doi.org/10.1109/ACCESS.2021.3139729>.
- [2] O.V. Dulce-Salcedo, D. Maldonado, F. Sánchez, Is the proportion of female STEM teachers in secondary education related to women's enrollment in tertiary education STEM programs? *Int. J. Educ. Dev.* 91 (May 2022) 102591 <https://doi.org/10.1016/J.IJEDUDEV.2022.102591>.
- [3] L.T.B. Le, T.T. Tran, N.H. Tran, Challenges to STEM education in Vietnamese high school contexts, *Heliyon* 7 (12) (Dec. 2021) e08649, <https://doi.org/10.1016/J.HELIYON.2021.E08649>.
- [4] H. Jeong, C.E. Hmelo-Silver, K. Jo, Ten years of computer-supported collaborative learning: a meta-analysis of CSCL in STEM education during 2005–2014, *Educ. Res. Rev.* 28 (Nov. 2019) 100284, <https://doi.org/10.1016/J.EDUREV.2019.100284>.
- [5] J. Su, Y. Zhong, Artificial Intelligence (AI) in early childhood education: curriculum design and future directions, *Comput. Educ. Artif. Intell.* 3 (Jan. 2022) 100072, <https://doi.org/10.1016/J.CAEAI.2022.100072>.
- [6] Z. Gingl, G. Makan, J. Mellar, G. Vadai, R. Mingesz, Phonocardiography and photoplethysmography with simple Arduino setups to support interdisciplinary STEM education, *IEEE Access* 7 (2019) 88970–88985, <https://doi.org/10.1109/ACCESS.2019.2926519>.
- [7] T. Rath, N. Preethi, Application of AI in video games to improve game building, in: 2021 10th IEEE International Conference on Communication Systems and Network Technologies (CSNT), 2021, pp. 821–824, <https://doi.org/10.1109/CSNT51715.2021.9509685>.
- [8] J. Tao, et al., XAI-driven explainable multi-view game cheating detection, in: 2020 IEEE Conference on Games (CoG), 2020, pp. 144–151, <https://doi.org/10.1109/CoG47356.2020.9231843>.
- [9] I. Oh, S. Rho, S. Moon, S. Son, H. Lee, J. Chung, Creating pro-level AI for a real-time fighting game using deep reinforcement learning, *IEEE Trans. Games* (2021) 1, <https://doi.org/10.1109/TG.2021.3049539>.
- [10] L. Chang, J. Wu, N. Moustafa, A.K. Bashir, K. Yu, AI-driven synthetic biology for non-small cell lung cancer drug effectiveness-cost analysis in intelligent assisted medical systems, *IEEE J. Biomed. Heal. Informatics* (2021) 1, <https://doi.org/10.1109/JBHI.2021.3133455>.
- [11] Y. Fu, et al., Artificial intelligence in radiation therapy, *IEEE Trans. Radiat. Plasma Med. Sci.* 1 (2021), <https://doi.org/10.1109/TRPMS.2021.3107454>.
- [12] M. Jamshidi, et al., Artificial intelligence and COVID-19: deep learning approaches for diagnosis and treatment, *IEEE Access* 8 (2020) 109581–109595, <https://doi.org/10.1109/ACCESS.2020.3001973>.
- [13] Y. Zhang, T. You, J. Chen, C. Du, Z. Ai, X. Qu, Safe and energy-saving vehicle-following driving decision-making framework of autonomous vehicles, *IEEE Trans. Ind. Electron.* (2021) 1, <https://doi.org/10.1109/TIE.2021.3125562>.
- [14] J. Chen, C. Zhang, J. Luo, J. Xie, Y. Wan, Driving maneuvers prediction based autonomous driving control by deep Monte Carlo tree search, *IEEE Trans. Veh. Technol.* 69 (7) (2020) 7146–7158, <https://doi.org/10.1109/TVT.2020.2991584>.
- [15] H. Ning, R. Yin, A. Ullah, F. Shi, A survey on hybrid human-artificial intelligence for autonomous driving, *IEEE Trans. Intell. Transp. Syst.* (2021) 1–16, <https://doi.org/10.1109/ITITS.2021.3074695>.
- [16] V. Mazza, A. Khaliq, F. Salvetti, M. Chiaberge, Real-time apple detection system using embedded systems with hardware accelerators: an edge AI application, *IEEE Access* 8 (2020) 9102–9114, <https://doi.org/10.1109/ACCESS.2020.2964608>.
- [17] N. Pellas, A. Dengel, A. Christopoulos, A scoping review of immersive virtual reality in STEM education, *IEEE Trans. Learn. Technol.* 13 (4) (2020) 748–761, <https://doi.org/10.1109/TLT.2020.3019405>.
- [18] D. Adami, M.O. Ojo, S. Giordano, Design, development and evaluation of an intelligent animal repelling system for crop protection based on embedded edge-AI, *IEEE Access* 9 (2021) 132125–132139, <https://doi.org/10.1109/ACCESS.2021.3114503>.
- [19] T.K.F. Chiu, H. Meng, C.-S. Chai, I. King, S. Wong, Y. Yam, Creation and evaluation of a pretertiary artificial intelligence (AI) curriculum, *IEEE Trans. Educ.* 65 (1) (2022) 30–39, <https://doi.org/10.1109/TE.2021.3085878>.
- [20] J. Estevez, G. Garate, M. Graña, Gentle introduction to artificial intelligence for high-school students using scratch, *IEEE Access* 7 (2019) 179027–179036, <https://doi.org/10.1109/ACCESS.2019.2956136>.
- [21] P. Plaza, et al., Scratch as driver to foster interests for STEM and educational robotics, *IEEE Rev. Iberoam. Tecnol. del Aprendiz.* 14 (4) (2019) 117–126, <https://doi.org/10.1109/RITA.2019.2950130>.
- [22] Z. Huang, E. Kougiannos, X. Ge, S. Wang, P.D. Chen, L. Cai, A systematic interdisciplinary engineering and technology model using cutting-edge technologies for STEM education, *IEEE Trans. Educ.* 64 (4) (2021) 390–397, <https://doi.org/10.1109/TE.2021.3062153>.
- [23] A. Nanthamornphong, J. Holmes, P. Asawateera, A case study: phuket city data platform, 17th Int. Conf. Electr. Eng. Comput. Telecommun. Inf. Technol. ECTI-CON (2020) 717–722, <https://doi.org/10.1109/ECTI-CON49241.2020.9158101>, 2020.
- [24] J. Benjamaha, P. Uantrai, Active learning management based on MIAP learning model to enhance electronic technician competence, in: 2021 6th International STEM Education Conference (iSTEM-Ed), 2021, pp. 1–4, <https://doi.org/10.1109/iSTEM-Ed52129.2021.9625152>.
- [25] Y. Zhao, P. Deng, J. Liu, M. Wang, J. Wan, LCArNet: lightweight context-aware attention networks for earthquake detection and phase-picking on IoT edge devices, *IEEE Syst. J.* (2021) 1–12, <https://doi.org/10.1109/JSYST.2021.3114689>.
- [26] W. Pang, X. Jiang, M. Lv, T. Gao, D. Liu, W. Yi, Towards the predictability of dynamic real-time DNN inference, *IEEE Trans. Comput. Des. Integr. Circuits Syst.* (2021) 1, <https://doi.org/10.1109/TCAD.2021.3120329>.
- [27] A. Gutierrez-Torre, et al., Automatic distributed deep learning using resource-constrained edge devices, *IEEE Internet Things J.* (2021) 1, <https://doi.org/10.1109/JIOT.2021.3098973>.
- [28] H. Xu, M. Guo, N. Nedjah, J. Zhang, P. Li, Vehicle and pedestrian detection algorithm based on lightweight YOLOv3-promote and semi-precision acceleration, *IEEE Trans. Intell. Transp. Syst.* (2022) 1–12, <https://doi.org/10.1109/TITS.2021.3137253>.
- [29] P. Abichandani, V. Sivakumar, D. Lobo, C. Iaboni, P. Shekhar, Internet-of-Things curriculum, pedagogy, and assessment for STEM education: a review of literature, *IEEE Access* 10 (2022) 38351–38369, <https://doi.org/10.1109/ACCESS.2022.3164709>.

- [30] V. Jain, R.M. Patrikar, A low-cost portable dynamic droplet sensing system for digital microfluidics applications, *IEEE Trans. Instrum. Meas.* 69 (6) (2020) 3623–3630, <https://doi.org/10.1109/TIM.2019.2932526>.
- [31] J. Sigut, M. Castro, R. Arnay, M. Sigut, OpenCV basics: a mobile application to support the teaching of computer vision concepts, *IEEE Trans. Educ.* 63 (4) (2020) 328–335, <https://doi.org/10.1109/TE.2020.2993013>.
- [32] M. V. Kubrikov, M. V. Saramud, M. V. Karaseva, Method for the optimal positioning of the cutter at the honeycomb block cutting applying computer vision, *IEEE Access* 9 (2021) 15548–15560, <https://doi.org/10.1109/ACCESS.2021.3052964>.
- [33] P. Musil, R. Juránek, M. Musil, P. Zemčík, Cascaded stripe memory engines for multi-scale object detection in FPGA, *IEEE Trans. Circuits Syst. Video Technol.* 30 (1) (2020) 267–280, <https://doi.org/10.1109/TCSVT.2018.2886476>.
- [34] I. Kovalev, N. Chervonnova, R. Nezhmetdinova, Development of a module for analyzing milling defects using computer vision defects using computer vision, in: 2021 International Russian Automation Conference (RusAutoCon), 2021, pp. 986–990, <https://doi.org/10.1109/RusAutoCon52004.2021.9537548>.
- [35] K. Parkinson, A. Minaie, R. Sanati-Mehrziy, Recognizing fractal behavior in Jackson Pollock artwork through computer vision, in: 2020 International Engineering, Technology and Computing (IETC), 2020, pp. 1–6, <https://doi.org/10.1109/IETC47856.2020.9249100>.
- [36] H.E. Ascencio, C.F. Peña, K.R. Vásquez, M. Cardona, S. Gutiérrez, Automatic multiple choice test grader using computer vision, in: 2021 IEEE Mexican Humanitarian Technology Conference (MHTC), 2021, pp. 65–72, <https://doi.org/10.1109/MHTC52069.2021.9419920>.
- [37] V.G. Al Hakim, S.-H. Yang, M. Liyanawatta, J.-H. Wang, G.-D. Chen, Robots in situated learning classrooms with immediate feedback mechanisms to improve students' learning performance, *Comput. Educ.* 182 (Jun. 2022) 104483, <https://doi.org/10.1016/J.COMPEDU.2022.104483>.
- [38] Y.M. Huang, L.M. Silitonga, T.T. Wu, Applying a business simulation game in a flipped classroom to enhance engagement, learning achievement, and higher-order thinking skills, *Comput. Educ.* 183 (Jul. 2022) 104494, <https://doi.org/10.1016/J.COMPEDU.2022.104494>.
- [39] C.Y. Heo, B. Kim, K. Park, R.M. Back, A comparison of Best-Worst Scaling and Likert Scale methods on peer-to-peer accommodation attributes, *J. Bus. Res.* 148 (Sep. 2022) 368–377, <https://doi.org/10.1016/J.JBUSRES.2022.04.064>.
- [40] R.F. Rahmat, Y. Azzakiro, T.Z. Lini, Tree identification to calculate the amount of palm trees using haar-cascade classifier algorithm, in: 2019 3rd International Conference on Electrical, Telecommunication and Computer Engineering (ELTICOM), 2019, pp. 36–39, <https://doi.org/10.1109/ELTICOM47379.2019.8943897>.
- [41] T.Q. Vinh, N.T.N. Anh, Real-time face mask detector using YOLOv3 algorithm and haar cascade classifier, in: 2020 International Conference on Advanced Computing and Applications (ACOMP), 2020, pp. 146–149, <https://doi.org/10.1109/ACOMP50827.2020.00029>.
- [42] M.P. Anggadhit, Y. Widiastwi, Breaches detection in Zebra cross traffic light using haar cascade classifier, in: 2020 International Conference on Informatics, Multimedia, Cyber and Information System (ICIMCIS), 2020, pp. 272–277, <https://doi.org/10.1109/ICIMCIS51567.2020.9354275>.
- [43] K. Klinbumrung, Engineering education management using project-based and MIAP learning model for microcontroller applications, in: 2020 7th International Conference on Technical Education (ICTechEd7), 2020, pp. 33–36, <https://doi.org/10.1109/ICTechEd749582.2020.9101246>.