

Received January 14, 2020, accepted February 24, 2020, date of publication March 2, 2020, date of current version March 16, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.2977679

# Design and Evaluation of a Deep Learning Recommendation Based Augmented Reality System for Teaching Programming and Computational Thinking

PEI-HSUAN LIN<sup>1</sup> AND SHIH-YEH CHEN<sup>2</sup>

<sup>1</sup>Department of Engineering Science, National Cheng Kung University, Tainan 701, Taiwan

<sup>2</sup>Department of Computer Science and Information Engineering, National Taitung University, Taitung 950, Taiwan

Corresponding author: Shih-Yeh Chen (sychen@nttu.edu.tw)

This work was supported in part by the Ministry of Science and Technology of the Republic of China, Taiwan, under Contract MOST 108-2511-H-143-002.

**ABSTRACT** Programming is considered a skill to arouse and inspire learner's potential. Learning to program is a complex process that requires students to write grammar and instructions. The structure of a programming language does not cause impose problems to students, the real obstacle is how to apply these learned grammars and present them in a complete and correct program code for problem solving. In this study, a deep learning recommendation system was developed, which includes augmented reality (AR) technology, and learning theory, and is provided for study by students in non-major and also from different learning backgrounds. Those students divided into two groups, the students participating in the experimental group were using the AR system with deep learning recommendation and the students participating in the control group were using the AR system without deep learning recommendation. The results show that students in experimental group perform better than the control group with regards to learning achievement. On the other hand, in the part of computational thinking ability, students using a deep learning recommendation based AR system is significantly better than those using non-deep learning recommendation based AR system. Among the various dimensions of computational thinking, creativity, logical computing, critical thinking, and problem-solving skills are significantly different among the two groups of students. The students in experimental group perform better than the control group with regards to the dimensions of computational thinking, creativity, logical computing, critical thinking, and problem-solving skills.

**INDEX TERMS** Deep learning recommendation, computational thinking, AR technology.

## I. INTRODUCTION

Due to the widespread applications of computers, people are paying more attention to computer science education, and the skills and knowledge related to programming have attracted attention and become an indispensable part of education. Most teachers teach unilaterally, and even when they teach programming. A programming class is usually taught by lecturing and demonstrations of fixed and static program codes. Most students think that program compilation is a difficult area to understand, they cannot understand the actual situation of the programming compilation during the learning process, and do not know how to apply implementation,

The associate editor coordinating the review of this manuscript and approving it for publication was Mu-Yen Chen<sup>3</sup>.

which leads to challenges and obstacles for students to solve real problems in programming in the process of programming compilation [2], [3]. The learning process of programming is a complex process that requires students to write grammar and instructions. The structure of the programming language does not cause learning difficulties for students, the real problem is how to apply these learned grammars and present them in a complete and correct program code [4], [5]. In order to enable students to write appropriate and correct program code during the compilation process, this study uses augmented reality (AR) features to dynamically overlap digital material with the real environment, which provides students with a context-aware learning environment for writing and compiling program codes in a problem-solving manner. AR technology is now widely used in learning environments

and is considered to be an educational tool with positive impact [6], [7]. AR can provide students with a context-aware environment between reality and virtual, and can enhance students' immersive experience in the learning process [8]. Students can interact with the virtual world through practical operations to achieve learning results [9]. However, in an AR learning environment, which provides a large amount of knowledge content, it is still necessary to effectively help students master the tools of learning focus [10], [11].

How to predict the difficulties students encounter in the process of learning, as well as after they learn, is a concern of teachers and parents. In the past, students' academic performance or classroom assessments were used to judge students' understanding of learning content. There are also many different data mining techniques applied to predict student performance, such as Naive Bayes, Decision tree, Support Vector Machine, K-Nearest Neighbor, etc [12]. The use of artificial intelligence and deep learning has increased in recent years. Deep learning, which is an emerging technology in educational data mining, is a type of neural network, and is regarded as one of the best predictors of student learning behavior, as it that can detect correlations with predictive variables. Even if there is a complex nonlinear relationship between dependent variables and independent variables, deep learning can still make predictions [13], [14]. Fok *et al.* [15] used the Google Tensorflow Deep Learning analytic engine to predict the development of students after study according to their academic performance, such as the results of the examinations of various subjects: Chinese, English, history, and physics, and their non-study performance, such as learning behavior, sports, arts, learning services, etc. Therefore, this study records students' operational processes and learning behaviors for analysis, in order to recommend the grammar and unit where the students need remedial teaching, and provide students with a personalized learning way. Moreover, ability assessment after learning is an important step in the process of promoting computer science education [16], [17] Chen *et al.* [17] mentioned that programming courses are fast and diverse learning areas, thus, if students are only required to complete a single instruction, it is not a completely successful study. The focus of truly successful learning is to allow students to gain different skills and mindsets to facilitate their learning and problem solving in the future. Computational thinking is considered to be a very important core competence to help students learning and thinking in the future [18]–[20]. How to help those students of different learning areas and ages inspire computational thinking skills is also a topic that is often discussed [21].

Based on the above, this study proposes an image-based programming learning system, which includes AR technology, deep learning with recommendations, and learning theory, and is provided for study by students in different learning areas, in order that they can overcome the learning difficulties of being non-major undergraduates, and then, explores the differences of computational thinking abilities. The research questions are as follows:

- Are there differences in learning achievement between students who learn by AR system with deep learning recommendation and AR system without deep learning recommendation?
- Are there differences in computational thinking between students who learn by AR system with deep learning recommendation and AR system without deep learning recommendation?

## II. RELATED RESEARCH

### A. AR IN EDUCATION

AR is a technology that superimposes dynamic digital content in a real-world environment, thus, providing users with a realistic and immersive perspective [11], [22]–[24], while also practicing the visualization of “invisible” concepts or objects, meaning that in a 3D virtual environment, AR technology can combine real and virtual objects to provide users with immersive and interactive experiences [6], [25]. In recent years, many educational researchers have applied AR technology to training and education, and their findings indicate that AR technology can allow students to engage in authentic learning activities and explore real environments [26]–[28]. Through AR technology, students can interact with virtual objects from different perspectives and enhance their visual perception, which can help students to improve their learning and understanding, and also attract students to explore and investigate problems in the real world [25], [29] Dunleavy *et al.* [8] pointed out that if an AR learning environment is carefully designed, it can increase students' situational awareness and experience in an immersive learning environment. There have been many studies exploring AR learning systems related to learning activity design, such as: Di Serio *et al.* [24] applied AR technology to the visual arts curriculum of middle school student. Wang *et al.* [30] used AR simulation systems for collaborative inquiry learning activities, and then, for comparison with traditional teaching methods; Huang and Lin [28] applied AR in the primary school astrology curriculum to present the movements of the stars through virtual objects, and then, to explore the learning outcomes and flow experience of students in the learning activities; Lin *et al.* [31] use the digital picture books combined with AR technology to allow students to understand the four stages of insect life cycle; and explore students' imagination and learning motivation. Therefore, this study applies AR in a programming course, in order that students can see the objectives of the project to be completed in the real environment, and complete the learning content through the application and editing of the learned program syntax.

### B. DEEP LEARNING

Artificial intelligence and big data are the trend of research topics in various/different fields, these technologies support predicting, problem-solving and decision-making [32]. Deep learning, which is a process where multi-level neural networks perform specific tasks, is a branch of machine

learning and a part of AI [33]. Through a series of logics from a large number of examples and data, multiple processes have been nonlinearly transformed to provide results that are sufficient to represent data characteristics and features without specific rules [34]. Deep learning is known as one of illustrious technology for analysis, classification and predictions [35]–[38]. Deep learning is a kind of machine learning technology and it could be semi-supervised, unsupervised or supervised [35]–[37]. In the other words, deep learning model can be built from learning experience and with minimal external interference [39]. Peters [40] presented the trends of deep learning: (1) deep learning has a long and rich history, and has been discussed more and more in recent years. There are also many different opinions reflected in the definitions of nouns; (2) deep learning is considered useful and usable due to its extensive use; (3) with the advancement of software and hardware, the scope of deep learning model application is expanding; and (4) with the evolution of time, deep learning can address more and more complex applications. Deep learning can be said to be a collection of many emerging technologies, and there are many breakthrough developments and applications in different fields, as compared to other existing machine learning algorithms. For example, Baldi *et al.* [41] used deep learning in the field of high-energy physics to solve the problem of signal-versus-background classification; LeCun *et al.* [36] proposed deep learning to identify abstract data processing, and achieved flexibility and high accuracy for speech and image recognition; Gulshan *et al.* [34] applied deep learning to medicine for automatic detection of retinopathy; Day and Lin [42] employed deep learning to emotional analysis to evaluate smartphone user reviews and find user opinions by using emotional dictionaries; Goh *et al.* [43] considered deep learning as the most valuable tool in the application of computational chemistry; Esteva *et al.* [44] used deep learning to analyze image data of skin disorders; Wu *et al.* [45] translated handwritten text content through deep learning. Huang *et al.* [32] apply deep learning to massive open online courses (MOOCs), due to the main method of MOOCs is watching video, they analyze student's learning log to predict student are able to respond specific difficultly questions and recognition the degree of question correctly. Xing and Du [46] use deep learning mechanism to predict individual student who discontinue their studies, provide the method to avoid the high risk of student dropout. Most such researches applied deep learning to the analysis of data results. This study combines the characteristics of deep learning with an AR system for real-time application at an education site, offers students immediate learning feedback, and provides relevant learning tasks to help clarify the difficulties and doubts of the students in learning the program language.

### C. COMPUTATIONAL THINKING

Computational thinking was proposed by Wing [47], which aims to solve problems, design systems, and understand human behavior through the basic concepts of calculator

science. Wing [48] also proposed four main dimensions of computational thinking ability: (1) Decomposition; (2) Pattern Recognition; (3) Pattern Generalization and Abstraction; (4) Algorithm Design; and with training of those four abilities, students can effectively develop their thinking skills in a planned manner. In the past, many scholars and experts in related fields applied a wide range of computational thinking; therefore, there are different definitions in this important process [49]–[53]. For example, Cuny *et al.* [49] proposed that computational thinking is a thinking process that involves computation and problem solving, and can present effective solutions by means of information; Barr and Stephenson [52] put forward a structured model of the core concepts and capabilities of computational thinking, such as data collection, data analysis, data presentation, problem decomposition, etc.; Aho [50] simplified the definition of computational thinking into a method to solve problems through computational steps and algorithms; Sysło and Kwiatkowska [53] emphasized that computational thinking is a series of thinking skills, and not just the result of computer compilation; García-Peñalvo [51] suggested that computational thinking is a high-level abstract and computational approach to solving problems.

In simple terms, computational thinking is the way of thinking and practicing computing, as well as a way to positively solve problems; however, it is not mandatory to use technology to solve problems, instead, it guides students to solve problems with the concept of technology [54]. The topic of computational thinking is considered in the field of research and in the application of education [55]. Many experts in the education field emphasized that computational thinking is an important skill in the field of education and technology in the 21st century [56]. Computational thinking has been applied to education, such as Chen *et al.* [17], who explored the changes in students' inspirations regarding challenges that focused on potential and computational thinking in a robot coding course; Tsai and Tsai [57] explored externally-facilitated regulated learning in a quasi-experimental manner, and created a blended learning environment of computational thinking for the improvement of students' computer skills; Wu *et al.* [58] analyzed and discussed computational thinking ability according to the co-compiling learning method.

The abovementioned shows the importance of computational thinking ability. This study aims to develop a program compiling learning system for different learning fields and non-major undergraduates, where the purpose is to inspire students' computational thinking ability and explore the differences between those non-major undergraduates.

### III. METHOD

In order to evaluate the impact of innovative learning methods on students, this study conducted an experiment to explore the differences between students' learning outcomes and computational thinking according to different learning strategies. The course is named the "Program Logic Thinking

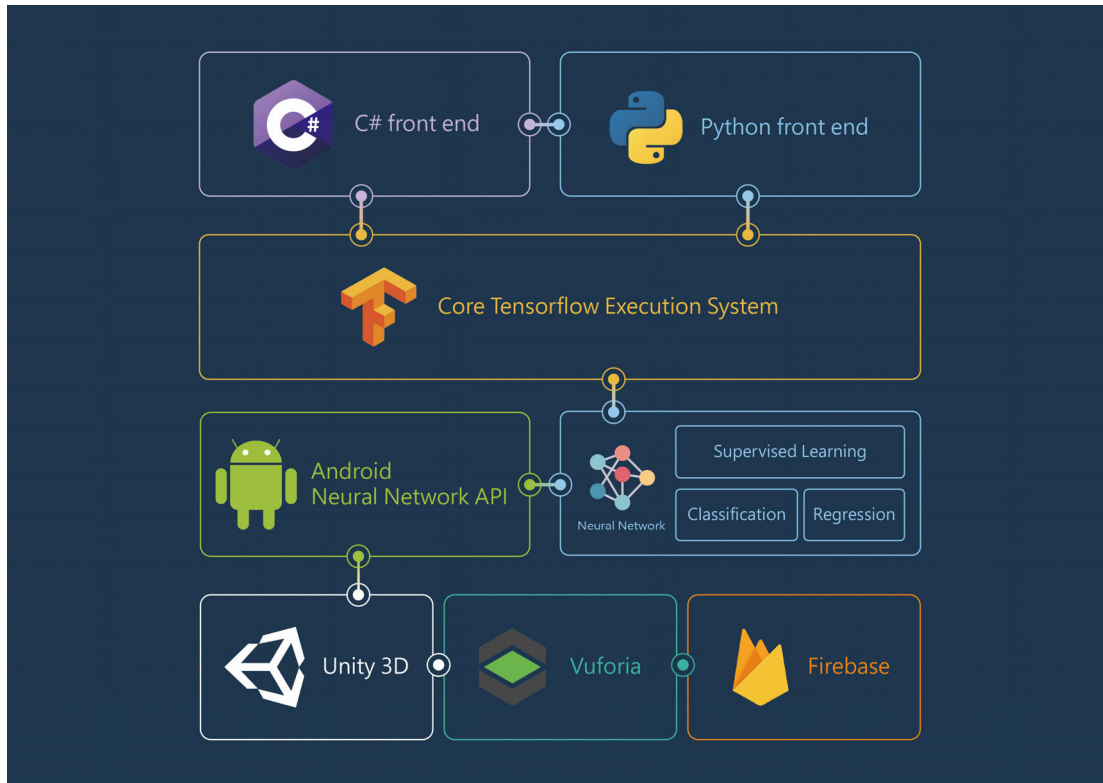


FIGURE 1. The structure of the deep learning recommendation based AR system.

Education” course in the university general education curriculum, and the purpose of this course is to provide students with basic programming concepts and to develop students’ computational thinking skills in the process of program compilation.

#### A. PARTICIPANTS

The participants of this study are 97 students from a university in eastern Taiwan, who were assigned to an experimental group and a control group. The students participating in this experiment were not students in the information technology related department. There were 48 college students in the experimental group, including 23 students from the Department of Arts and 25 students from the Department of Music; the control group consisted of 49 college students, with 24 students from the Department of Chinese and 25 students from the Department of Public Administration. This study used the convenience sampling method, the average age of students was 20 years old, and one teacher taught the same course content to all students.

#### B. DEVELOPMENT OF DEEP LEARNING RECOMMENDATION BASED AR SYSTEM

In this study, Unity 3d is a cross-platform game engine that is used for developing the AR function. Regarding the recommendation function, it built on the Unity Machine Learning Agents based on tensorflow framework to allow Unity scripts to receive data from python scripts. The recurrent neural network (RNN) is adopted to model dynamic and sequence data,

which can more accurately learn the feature of users and items to achieve supervised learning path recommendations. As for the back-end development, Firebase is selected, which is an app development platform that supports Android, iOS and website, to help app developers for quickly building back-end services in the cloud (Show as Figure 1).

The AR system combined AI with deep learning technology to determine its relevance through the compilation and operation process of students, and recommended different learning tasks for students who were confused or found the learning process incomprehensible, in order to improve students’ logic and application of related programming languages. Figure 2 shows the AR system architecture, which is mainly composed of an AR learning system module and a personalized learning module. The AR learning system module includes: (1) Deep Learning recommendation function, where the system recommendation strengthens the program syntax and logic related learning tasks, and provides more practice opportunities for students according to their learning process; (2) AR object control, which includes learning tasks and learning materials to present material management; (3) learning mission and materials, which provide different AR learning materials and tasks for students to query and practice according to the compiled learning units of different programming languages. The personalized learning module includes: (1) learning process, which records the student’s learning and operation process; (2) learning achievement, where the teacher gives the student’s grade according to the completion degree of the student’s learning task; and (3) code



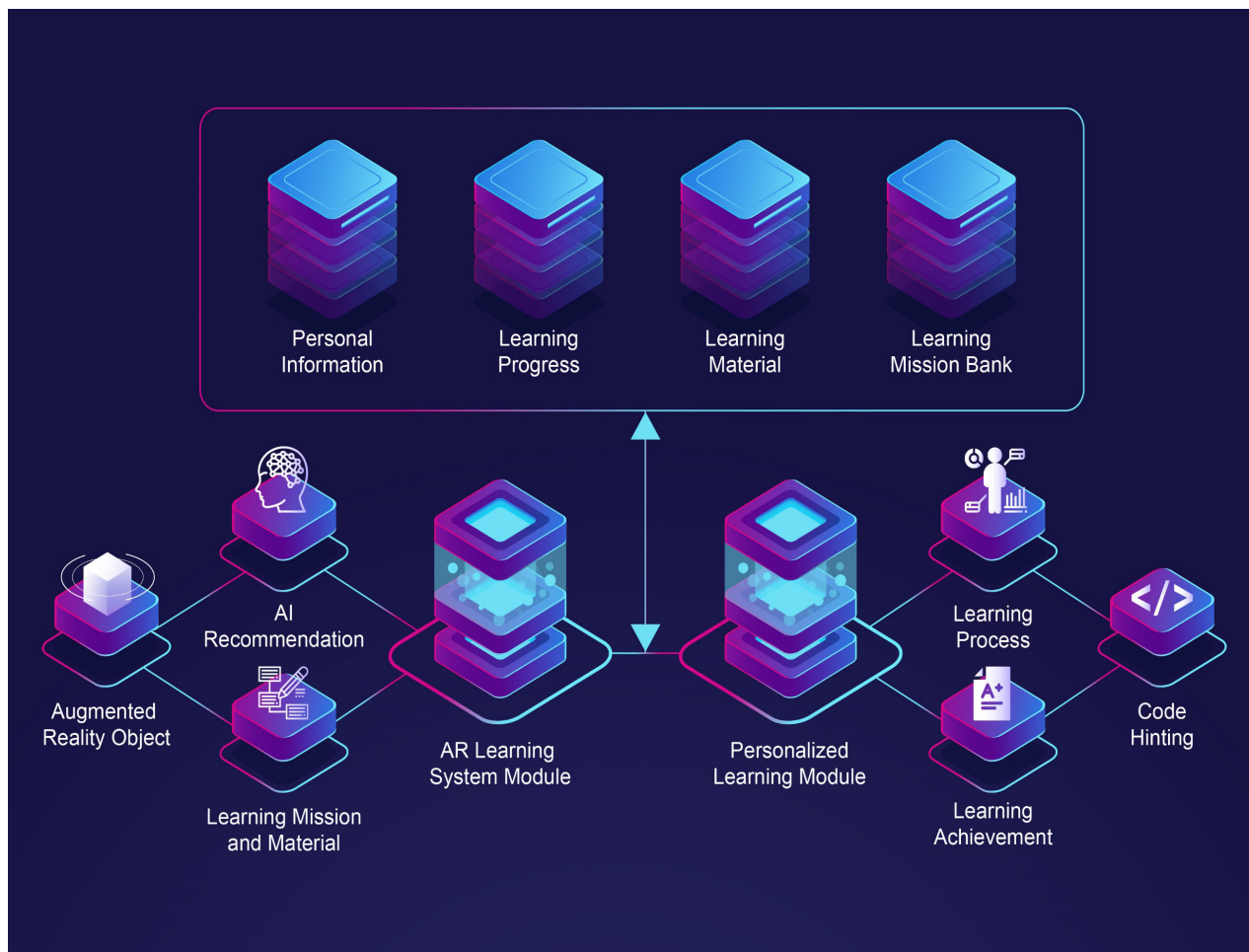


FIGURE 2. The structure of the deep learning recommendation based AR system.

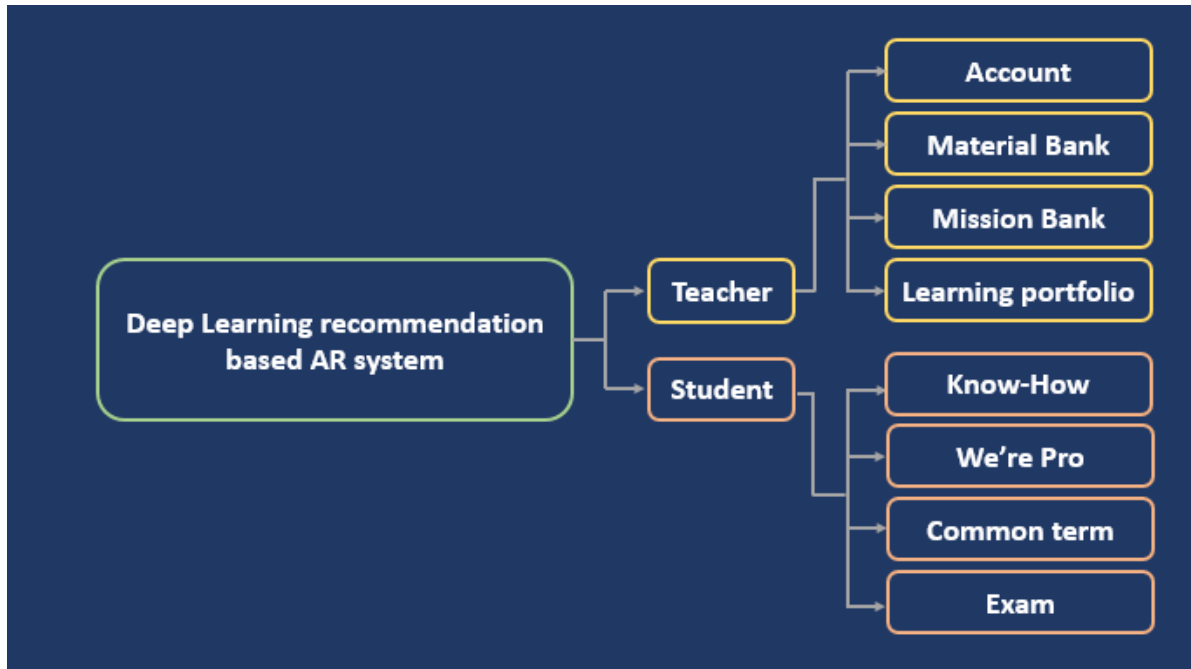
hints, where relevant prompt content is provided to the students according to the task.

The system is divided into two parts, the teacher part and the student part, each part with four main functions. Figure 3 shows the functions of AR system. The teacher part includes (1) Account: Teacher can add, modify and delete student account authority for participating in the programming language course which also access student’s major, gender and other related information; (2) Material Bank: Teacher can add, modify and delete learning materials based on student’s learning progress and chapter content; (3) Mission Bank: Teacher can add, modify and delete learning mission based on student’s learning progress and chapter content; (4) Learning Portfolio: Teacher can grasp and control the learning status of students and make adjustments in teaching method or progress at any time. The student part includes (1) Know-How: Student can choose to read different chapters depend on their own learning progress; (2) We’re Pro: Provide AR based programming language learning mission to student; (3) Common term: Provide student with commonly used programming language grammar; (4) Exam: Provide related questions for each chapter.

All participating students used their mobile devices in their learning activities for AR presentation of program language learning content and learning tasks. Figure 4 shows the operation screen of students in the learning task. Students must be individually logged into the learning system, which provides the related learning tasks for different learning units, as presented by AR technology. The 3D dynamic objects were combined with the real environment, where students could disassemble, compile, and apply the learned knowledge content, and then, complete the learning tasks and achieve learning goals.

C. LEARNING ACTIVITY DESIGN

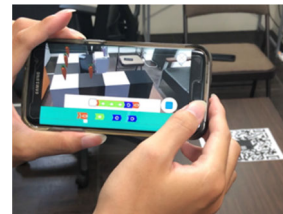
The experimental activities were carried out in the “Program Logic Thinking Education” university general education curriculum. The learning system was designed with an image program, which combines deep learning recommendation with AR technology to explore the impact on the learning effectiveness and computational thinking of non-major undergraduates. Figure 5 shows the learning activity process. First, this experiment divided four non-major undergraduate students into the experimental group into the control group. In the first stage of the experiment, the same teacher taught



**FIGURE 3.** The functions of the deep learning recommendation based AR system.

the basic concepts of programming language, and their logic applications, in order to confirm that all students have similar prior knowledge, the pre-test was applied. In the second stage, the teacher explained the operating procedures of the AR system to all students on their mobile devices, where the learning content and 3D materials in learning activities were all the same, and the students were divided into the experimental group and the control group for a total of 5 hours of learning activities. The experimental group students used the AR system with deep learning recommendation to perform learning tasks in a ubiquitous learning environment, where different 3D objects were presented in the real environment depending on the learning tasks, students judged and chose the programming syntax that could achieve the goal according to the task instructions, and the deep learning recommendation mechanism suggested the learning task to be strengthened according to the student's logical judgment and the application of the procedural grammar. The control group students used the AR system without deep learning recommendation to perform learning tasks in a ubiquitous learning environment, where different 3D objects were presented in the real environment depending on the learning tasks, and students judged and chose the programming syntax that could achieve the goal according to the task instructions. In the third stage, after the learning activities, all students conducted post-test and questionnaires to evaluate the impact of the learning activities on the students' computational thinking. In addition, students in the experimental group were interviewed to understand their willingness to use the system.

In addition, figure 6 shows the student-operated learning scenario of this study, student entered their own account and



**FIGURE 4.** The snapshot of learning mission.

password to confirm their identity and permissions. After verify, student can login to the system for the programming language course. Student can choose different chapters of programming language course to read according to their personal learning progress. After reading, student can practice and apply programming language through the Mission function. The learning mission provide different 3D route mission to allow student coding from the starting point to the end point through programming language. During the programming language learning process, not only examine student's ability of programming language but also train student's ability of logical thinking and judgment in solving mission. The results of the compiled programming code will be recorded in the database for analysis. The deep learning recommendation mechanism will decompose and analysis student's programming language learning process and recommends the related learning mission that the student who lack of programming language understanding. Finally, the assessment of student's learning achievement is carried out by exam. If the learning achievement isn't as expected, student can restart the learning scenario of programming language until the apply it well. All the learning process will be recorded in the database,

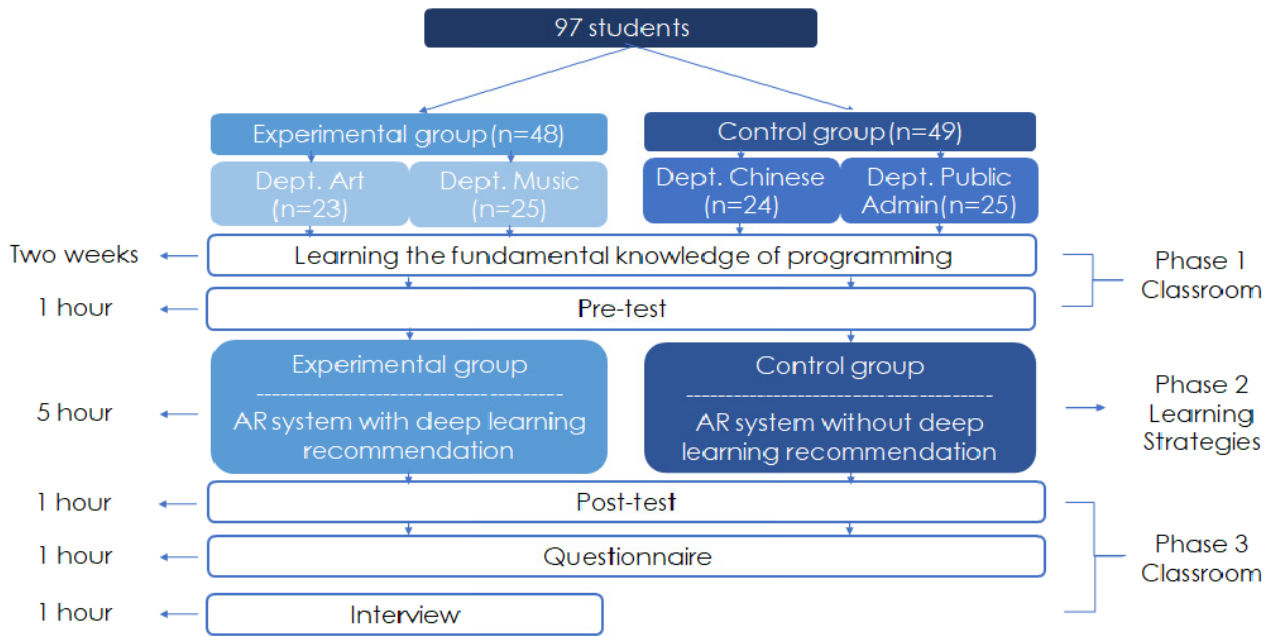


FIGURE 5. Learning activity scenario.

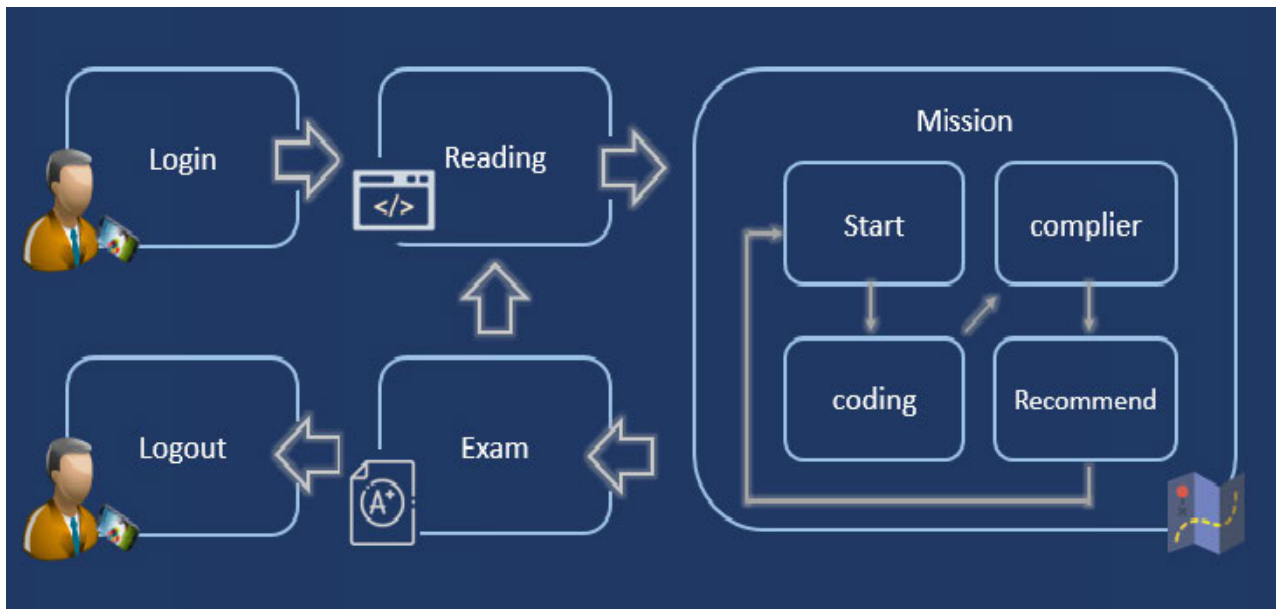


FIGURE 6. Student-operated learning scenario.

including login and logout time, learning material read by student, time of reading, time of learning mission, the steps of student solving mission.

**D. MEASURING TOOL**

In this study, the pre-test, post-test, and questionnaire were designed to assess the impact of learning activities on students’ learning achievement and computational thinking skills. The pre-test was used to confirm there was no significant difference in the basic concepts of the programming

language before the learning activities of the students of the experimental group and control group, which includes 10 multiple-choice questions and 10 fill-in-the-blank questions. Moreover, the post-test was used to evaluate students’ understanding and application of the programming courses, including five yes-no questions, five multiple-choice questions, and ten fill-in-the-blank questions. The pre-test and post-test questions, with a total score of 100, were designed by experts with years of programming experience. The computational thinking ability questionnaire was adapted from

**TABLE 1.** The Cronbach's  $\alpha$  value of computational thinking dimension.

Dimensions	Cronbach's $\alpha$	Developer
Creativity	0.953	Korkmaz, Çakir and Özden [1]
Algorithmic Thinking	0.951	
Cooperativity	0.912	
Critical thinking	0.964	
Problem solving	0.965	

**TABLE 2.** T-test result for learning achievement.

Group	<i>N</i>	<i>M</i>	<i>SD</i>	<i>t</i>	<i>F</i>
<b>Pre-test</b>					
Experimental	48	52.5	17.894	-0.693	1.486
Control	49	54.9	16.153		
<b>Post-test</b>					
Experimental	48	84.06	13.033	4.780	4.778*
Control	49	68.88	17.920		

Notes: \* $p < 0.05$

**TABLE 3.** T-test result for computational thinking.

Group	<i>N</i>	<i>M</i>	<i>SD</i>	<i>t</i>
Experimental	48	4.46	0.504	8.045*
Control	49	3.51	0.649	

Notes: \* $p < 0.05$

Korkmaz *et al.* [1], which has a total of 29 questions scored according to a five-point Likert scale, where the range is 1 (very disagree) to 5 (very agree). Table 1 shows the Cronbach's  $\alpha$  value of computational thinking dimension. The questionnaire is divided into 5 dimensions: Creativity, Algorithmic Thinking, Cooperativity, Critical Thinking, and Problem Solving. There are eight questions in creativity, such as "I believe I can solve problems when I face new tasks", "My dream is an important factor when I perform many tasks", and "In the course of performing tasks, I believe in my intuition and feelings about solving problems", and its Cronbach's  $\alpha$  is 0.953. There are six questions for logical thinking, such as "I am particularly interested in the mathematical process", "I can solve problems with mathematics in my daily life", and "I think I can quickly discover the relationship between numbers", and its Cronbach's  $\alpha$  is 0.951. There are four questions for cooperativity, such as "I like to experience cooperative learning with my classmates", "More ideas will be generated in the cooperative learning process", and "I think that through cooperative learning, we can achieve/get better results", and its Cronbach's  $\alpha$  is 0.912. There are a total of 5 questions for critical thinking, such as "I am very good at formulating steps to solve complex problems", "I think it is very interesting to try to solve complex problems", and "I am willing to learn something that is challenging for myself", and its Cronbach's  $\alpha$  is 0.964. There are 6 questions for problem solving, such as "I can put forward many ideas when considering ways to solve problems", "I can gradually apply my ideas to solve problems", and "I can consider

the existence of possible variables in the process of solving problems", and its Cronbach's  $\alpha$  is 0.965. In addition, in the qualitative interview part, the experimental group students (with the use of AR system with deep learning recommendation) were taken as the interview subjects, and the interview content is the Technology Acceptance Model (TAM), as proposed by Davis [59].

#### IV. DATA ANALYSIS AND RESULTS

This study used the SPSS 25 Windows version to analyze the study data, where the *P*-value must be less than 0.05 to be considered significant. This study used independent sample *t*-test to determine the difference between the data of the students in the experimental group and the control group. Table 2 shows the *t*-test used to examine the difference in learning outcomes between the students before and after the learning activity. The results show that, in the pre-test part, the average of the experimental group is 52.5, and the standard deviation is 17.894; the average of the control group is 54.9, and the standard deviation is 16.153. The *t*-test result ( $t = -0.693$ ) shows no significant difference between the two groups. In the post-test part, the experimental group average is 84.06, and the standard deviation is 13.033; the control group average is 68.88, and the standard deviation is 17.920. The *t*-test result ( $t = 4.780$ ,  $p < 0.05$ ) shows a significant difference between the two groups. It can be seen that the two groups of students had the same cognition level of the programming language course before the learning activities, and after the learning activities, the experimental



**TABLE 4. T-test result for the five dimensions of computational thinking.**

Group	<i>N</i>	<i>M</i>	SD	<i>t</i>
<b><i>Creativity</i></b>				
Experimental	48	4.53	0.407	13.848*
Control	49	3.02	0.643	
<b><i>Algorithmic Thinking</i></b>				
Experimental	48	4.47	0.579	8.537*
Control	49	3.21	0.853	
<b><i>Cooperativity</i></b>				
Experimental	48	4.01	0.839	0.851
Control	49	3.85	0.985	
<b><i>Critical thinking</i></b>				
Experimental	48	4.33	0.622	2.566*
Control	49	3.86	1.111	
<b><i>Problem solving</i></b>				
Experimental	48	4.69	0.427	6.284***
Control	49	3.68	1.044	

Notes: \* $p < 0.05$ , \*\*\* $p < 0.001$

group students have better learning outcomes than the control group students.

Table 3 shows the computational thinking of the two groups of students after the learning activities. The average of the experimental group is 4.46, and the standard deviation is 0.504; the average of the control group is 3.51, and the standard deviation is 0.649. The  $t$ -test result ( $t = 8.045$ ,  $p < 0.05$ ) shows a significant difference between the two groups, meaning that the experimental group students significantly improved their computational thinking ability after the learning activities. Table 4 shows the influence of each group of students in computational thinking. In the part of creativity, the experimental group average is 4.53, and the standard deviation is 0.407; the control group average is 3.02, and the standard deviation is 0.643. The  $t$ -test result ( $t = 13.848$ ,  $p < 0.05$ ) shows a significant difference between the two groups. In the logical thinking part, the experimental group average is 4.47, and the standard deviation is 0.579; the control group average is 3.21, and the standard deviation is 0.853. The  $t$ -test result ( $t = 8.537$ ,  $p < 0.05$ ) shows a significant difference between the two groups. In the part of cooperativity, the experimental group average is 4.01, and the standard deviation is 0.839; the control group average is 3.85, and the standard deviation is 0.985. The  $t$ -test result ( $t = 0.851$ ) shows no significant difference between the two groups. In the part of critical thinking, the experimental group average is 4.33, and the standard deviation is 0.622; the control group average is 3.86, and the standard deviation is 1.111; The  $t$ -test result ( $t = 2.566$ ,  $p < 0.05$ ) shows a significant difference between the two groups. Finally, in the problem-solving part, the experimental group average is 4.69, and the standard deviation is 0.427; the control group average is 3.68, and the standard deviation is 1.044. The  $t$ -test result ( $t = 6.284$ ,  $p < 0.001$ ) shows a significant difference between the two groups.

The qualitative interview content design is based on the TAM model, and the students of the experimental group

were interviewed after participating in the learning activities. There are four dimensions in the TAM model, perceived usefulness, perceived ease of use, attitude toward using, and behavioral intention to use. In terms of perceived usefulness, the experimental group students believe that the deep learning recommendation method helped “improve their learning scores” and “organize knowledge and content faster”. Students mentioned that the task which provide from deep learning recommended learning allowed them to enhance and apply those program syntaxes that they are not familiar with or do not understand, rather than just memorizing the syntax. In terms of perceived ease of use, the experimental group students believe that the AR system is “easy to operate” and “the interface is clear and easy to use”. Students mentioned that system operation can be directly used intuitively, without much thinking. In terms of attitude toward using, the experimental group students believe that the AR system “made learning interesting and improved their confidence in learning”. Students mentioned that the presentation of the AR learning tasks made the tasks feel interesting and changeable, and they felt satisfied with their ability to actually compile the programming language to solve the problem. In terms of behavioral intention to use, the experimental group students expressed “I hope to use this system to learn in the future.” Students mentioned that courses like programming languages are usually rather boring, but the guidance and practice of these learning tasks were a great help when constructing their programming language knowledge content.

## V. CONCLUSION

This research provides an image-based programming learning system, including deep learning based learning recommendations, AR technology, and learning theory to students in different learning areas to overcome the learning difficulties of non-major undergraduates. Deep learning recommendation method can enable students facing different

learning tasks to apply and practice the programming language and logic.

In order to evaluate the effect of the AI learning recommendation methods, this study designed an experiment to compare the effects of deep learning recommendation based AR system with non-deep learning recommendation based AR learning system on student learning achievement, as well as inspire students' computational thinking abilities. The experimental results show that, through deep learning recommendation method, the students' learning achievement is significantly better than that of non-deep learning recommendation method. This means the deep learning recommendation method is quite helpful for non-major undergraduates to learn through a programming language. In addition, in the part of computational thinking ability, students using deep learning recommendation based AR system are significantly better than using non-deep learning recommendation based AR system. Among the various dimensions of computational thinking, creativity, logical computing, critical thinking, and problem-solving skills are significantly different among the two groups of students; there was no significant difference in cooperativity. Learning through deep learning recommendation system not only improves students' learning outcomes, it also helps students to develop their computational thinking skills. As the cooperation between students was only limited to discussions of the learning activities, there was no significant difference in cooperativity.

Qualitative interviews show that most students accepted and were willing to learn the programming language with the deep learning recommendation system. Students said that this way of learning improved their learning effectiveness and made them more confident in the subject of programming language, which is very helpful for non-major undergraduate students to overcome learning gaps.

## VI. DISCUSSION AND FUTURE STUDIES

In this study, students learned a programming language through a deep learning recommendation based AR system, and the students' learning performance and computational thinking ability after the learning activities were explored. This way of learning can effectively solve the programming language difficulties of students, and abstract syntax and logic application can be improved in the learning activities. Through the recommendation of deep learning, which can reflect on the myths and misunderstandings in learning, and allow students to change from passive absorption of knowledge to actively understanding problems, thinking about problems, and solving problems.

This research was mainly designed for college students as the research subjects. In the learning activities, all students were non-major undergraduates, thus, their learning behaviors were slightly different from information technology based students. For example, possibly due to their sufficient knowledge of visual arts, students from the Department of Arts were quicker in the progress of the AR learning activities, thus, future research can discuss the different

learning behaviors of different majors. In addition, this research was designed for a programming language, thus, related research in the future can focus on different subjects.

## REFERENCES

- [1] Ö. Korkmaz, R. Çakir, and M. Y. Özden, "A validity and reliability study of the computational thinking scales (CTS)," *Comput. Hum. Behav.*, vol. 72, pp. 558–569, Jul. 2017.
- [2] L. Ma, J. Ferguson, M. Roper, and M. Wood, "Investigating and improving the models of programming concepts held by novice programmers," *Comput. Sci. Edu.*, vol. 21, no. 1, pp. 57–80, Mar. 2011.
- [3] M. J. Rubin, "The effectiveness of live-coding to teach introductory programming," in *Proc. 44th ACM Tech. Symp. Comput. Sci. Edu. (SIGCSE)*, 2013, pp. 651–656.
- [4] K. M. Cauley, "Studying knowledge acquisition: Distinctions among procedural, conceptual and logical knowledge," Virginia Commonwealth Univ., Richmond, VA, USA, Tech. Rep. ED 278 682, 1986.
- [5] E. Soloway, "Learning to program = learning to construct mechanisms and explanations," *Commun. ACM*, vol. 29, no. 9, pp. 850–858, Sep. 1986.
- [6] K.-Y. Chin, C.-S. Wang, and Y.-L. Chen, "Effects of an augmented reality-based mobile system on students' learning achievements and motivation for a liberal arts course," *Interact. Learn. Environ.*, vol. 27, no. 7, pp. 927–941, Jul. 2018.
- [7] T.-C. Huang, C.-C. Chen, and Y.-W. Chou, "Animating eco-education: To see, feel, and discover in an augmented reality-based experiential learning environment," *Comput. Edu.*, vol. 96, pp. 72–82, May 2016.
- [8] M. Dunleavy, C. Dede, and R. Mitchell, "Affordances and limitations of immersive participatory augmented reality simulations for teaching and learning," *J. Sci. Edu. Technol.*, vol. 18, no. 1, pp. 7–22, 2009.
- [9] K. R. Bujak, I. Radu, R. Catrambone, B. MacIntyre, R. Zheng, and G. Golubski, "A psychological perspective on augmented reality in the mathematics classroom," *Comput. Edu.*, vol. 68, pp. 536–544, Oct. 2013.
- [10] F. Ke and Y.-C. Hsu, "Mobile augmented-reality artifact creation as a component of mobile computer-supported collaborative learning," *Internet Higher Edu.*, vol. 26, pp. 33–41, Jul. 2015.
- [11] P.-H. Wu, G.-J. Hwang, M.-L. Yang, and C.-H. Chen, "Impacts of integrating the repertory grid into an augmented reality-based learning design on students' learning achievements, cognitive load and degree of satisfaction," *Interact. Learn. Environ.*, vol. 26, no. 2, pp. 221–234, 2018.
- [12] A. M. Shahiri, W. Husain, and N. A. Rashid, "A review on predicting student's performance using data mining techniques," *Procedia Comput. Sci.*, vol. 72, pp. 414–422, 2015.
- [13] D. Liu and C. Yang, "A deep reinforcement learning approach to proactive content pushing and recommendation for mobile users," *IEEE Access*, vol. 7, pp. 83120–83136, 2019.
- [14] Z. Huang, C. Yu, J. Ni, H. Liu, C. Zeng, and Y. Tang, "An efficient hybrid recommendation model with deep neural networks," *IEEE Access*, vol. 7, pp. 137900–137912, 2019.
- [15] W. W. T. Fok, Y. S. He, H. H. A. Yeung, K. Y. Law, K. Cheung, Y. Ai, and P. Ho, "Prediction model for students' future development by deep learning and tensorflow artificial intelligence engine," in *Proc. 4th Int. Conf. Inf. Manage. (ICIM)*, May 2018, pp. 103–106.
- [16] D. Weintrop, E. Beheshti, M. S. Horn, K. Orton, L. Trouille, K. Jona, and U. Wilensky, "Interactive assessment tools for computational thinking in high school STEM classrooms," in *Proc. Int. Conf. Intell. Technol. Interact. Entertainment*, 2014, pp. 22–25.
- [17] G. Chen, J. Shen, L. Barth-Cohen, S. Jiang, X. Huang, and M. Eltouky, "Assessing elementary students' computational thinking in everyday reasoning and robotics programming," *Comput. Edu.*, vol. 109, pp. 162–175, Jun. 2017.
- [18] A. A. de Souza, T. S. Barcelos, R. Munoz, R. Villarroel, and L. A. Silva, "Data mining framework to analyze the evolution of computational thinking skills in game building workshops," *IEEE Access*, vol. 7, pp. 82848–82866, 2019.
- [19] R. Munoz, R. Villarroel, T. S. Barcelos, F. Riquelme, A. Quezada, and P. Bustos-Valenzuela, "Developing computational thinking skills in adolescents with autism spectrum disorder through digital game programming," *IEEE Access*, vol. 6, pp. 63880–63889, 2018.
- [20] N. Da Cruz Alves, C. Gresse Von Wangenheim, and J. C. R. Hauck, "Approaches to assess computational thinking competences based on code analysis in K-12 education: A systematic mapping study," *Inform. Edu.*, vol. 18, no. 1, pp. 17–39, Apr. 2019.

- [21] T.-C. Hsu, S.-C. Chang, and Y.-T. Hung, "How to learn and how to teach computational thinking: Suggestions based on a review of the literature," *Comput. Edu.*, vol. 126, pp. 296–310, Nov. 2018.
- [22] A. Mejías Borrero and J. M. A. Márquez, "A pilot study of the effectiveness of augmented reality to enhance the use of remote labs in electrical engineering education," *J. Sci. Edu. Technol.*, vol. 21, no. 5, pp. 540–557, 2012.
- [23] R. T. Azuma, "A survey of augmented reality," *Presence, Teleoperators Virtual Environ.*, vol. 6, no. 4, pp. 355–385, 1997.
- [24] Á. Di Serio, M. B. Ibáñez, and C. D. Kloos, "Impact of an augmented reality system on students' motivation for a visual art course," *Comput. Edu.*, vol. 68, pp. 586–596, Oct. 2013.
- [25] H.-K. Wu, S. W.-Y. Lee, H.-Y. Chang, and J.-C. Liang, "Current status, opportunities and challenges of augmented reality in education," *Comput. Edu.*, vol. 62, pp. 41–49, Mar. 2013.
- [26] T. H. C. Chiang, S. J. H. Yang, and G.-J. Hwang, "Students' online interactive patterns in augmented reality-based inquiry activities," *Comput. Edu.*, vol. 78, pp. 97–108, Sep. 2014.
- [27] G.-J. Hwang, P.-H. Wu, C.-C. Chen, and N.-T. Tu, "Effects of an augmented reality-based educational game on students' learning achievements and attitudes in real-world observations," *Interact. Learn. Environ.*, vol. 24, no. 8, pp. 1895–1906, 2016.
- [28] Y.-M. Huang and P. H. Lin, "Evaluating students' learning achievement and flow experience with tablet PCs based on AR and tangible technology in u-learning," *Library Hi Tech*, vol. 35, no. 4, pp. 602–614, Nov. 2017.
- [29] Y.-C. Chen, H.-L. Chi, W.-H. Hung, and S.-C. Kang, "Use of tangible and augmented reality models in engineering graphics courses," *J. Prof. Issues Eng. Edu. Pract.*, vol. 137, no. 4, pp. 267–276, Oct. 2011.
- [30] H.-Y. Wang, H. B.-L. Duh, N. Li, T.-J. Lin, and C.-C. Tsai, "An investigation of university students' collaborative inquiry learning behaviors in an augmented reality simulation and a traditional simulation," *J. Sci. Edu. Technol.*, vol. 23, no. 5, pp. 682–691, May 2014.
- [31] P.-H. Lin, Y.-M. Huang, and C.-C. Chen, "Exploring imaginative capability and learning motivation difference through picture E-Book," *IEEE Access*, vol. 6, pp. 63416–63425, 2018.
- [32] N.-F. Huang, C.-C. Chen, J.-W. Tzeng, T.-T. Fang, and C.-A. Lee, "Concept assessment system integrated with a knowledge map using deep learning," in *Proc. Learn. MOOCs (LWMOOCs)*, Sep. 2018, pp. 113–116.
- [33] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning: Mit Press Essential Knowledge Series*. Cambridge, MA, USA: MIT Press, 2019.
- [34] V. Gulshan, L. Peng, M. Coram, M. C. Stumpe, D. Wu, A. Narayanaswamy, S. Venugopalan, K. Widner, T. Madams, J. Cuadros, R. Kim, R. Raman, P. C. Nelson, J. L. Mega, and D. R. Webster, "Development and validation of a deep learning algorithm for detection of diabetic retinopathy in retinal fundus photographs," *J. Amer. Med. Assoc.*, vol. 316, no. 22, p. 2402, Dec. 2016.
- [35] K.-L. Du, and M. N. Swamy, *Neural Networks and Statistical Learning*. New York, NY, USA: Springer, 2013.
- [36] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, no. 7553, p. 436, 2015.
- [37] O. Chapelle, B. Scholkopf, and Eds. A. Zien, "Semi-supervised learning (Chapelle, O. et al., Eds.; 2006) [book reviews]," *IEEE Trans. Neural Netw.*, vol. 20, no. 3, p. 542, Mar. 2009.
- [38] X. Zhu and A. B. Goldberg, "Introduction to semi-supervised learning," *Synth. Lectures Artif. Intell. Mach. Learn.*, vol. 3, no. 1, pp. 1–130, 2009.
- [39] M. A. Ahad, G. Tripathi, and P. Agarwal, "Learning analytics for IoE based educational model using deep learning techniques: Architecture, challenges and applications," *Smart Learn. Environ.*, vol. 5, no. 1, p. 7, Aug. 2018.
- [40] M. A. Peters, *Deep Learning, Education and the Final Stage of Automation*. New York, NY, USA: Taylor & Francis, 2018.
- [41] P. Baldi, P. Sadowski, and D. Whiteson, "Searching for exotic particles in high-energy physics with deep learning," *Nature Commun.*, vol. 5, no. 1, p. 4308, Jul. 2014.
- [42] M.-Y. Day and Y.-D. Lin, "Deep learning for sentiment analysis on Google play consumer review," in *Proc. IEEE Int. Conf. Inf. Reuse Integr. (IRI)*, Aug. 2017, pp. 382–388.
- [43] G. B. Goh, N. O. Hodas, and A. Vishnu, "Deep learning for computational chemistry," *J. Comput. Chem.*, vol. 38, no. 16, pp. 1291–1307, Mar. 2017.
- [44] A. Esteva, B. Kuprel, R. A. Novoa, J. Ko, S. M. Swetter, H. M. Blau, and S. Thrum, "Dermatologist-level classification of skin cancer with deep neural networks," *Nature*, vol. 542, no. 7639, pp. 115–118, Jan. 2017.
- [45] Y. Wu *et al.*, "Google's neural machine translation system: Bridging the gap between human and machine translation," 2016, *arXiv:1609.08144*. [Online]. Available: <http://arxiv.org/abs/1609.08144>
- [46] W. Xing and D. Du, "Dropout prediction in MOOCs: Using deep learning for personalized intervention," *J. Educ. Comput. Res.*, vol. 57, no. 3, pp. 547–570, 2019.
- [47] J. M. Wing, "Computational thinking," *Commun. ACM*, vol. 49, no. 3, pp. 33–35, 2006.
- [48] J. M. Wing, "Computational thinking and thinking about computing," *Philos. Trans. Roy. Soc. A, Math., Phys. Eng. Sci.*, vol. 366, no. 1881, pp. 3717–3725, 2008.
- [49] J. Cuny, L. Snyder, and J. M. Wing, "Demystifying computational thinking for non-computer scientists," unpublished, 2010. [Online]. Available: <http://www.cs.cmu.edu/~CompThink/resources/TheLinkWing.pdf>
- [50] A. V. Aho, "Computation and computational thinking," *Comput. J.*, vol. 55, no. 7, pp. 832–835, 2012.
- [51] F. J. García-Penalvo, "Editorial computational thinking," *IEEE Revista Iberoamericana de Tecnologías del Aprendizaje*, vol. 13, no. 1, pp. 17–19, Feb. 2018.
- [52] V. Barr and C. Stephenson, "Bringing computational thinking to K-12: What is involved and what is the role of the computer science education community?" *ACM Inroads*, vol. 2, no. 1, p. 48, Feb. 2011.
- [53] M. M. Sysło and A. B. Kwiatkowska, "Informatics for all high school students," in *Proc. Int. Conf. Inform. Schools, Situation, Evol., Perspect.*, 2013, pp. 43–56.
- [54] F. J. García-Penalvo and A. J. Mendes, "Exploring the computational thinking effects in pre-university education," *Comput. Hum. Behav.*, vol. 80, pp. 407–411, Mar. 2018.
- [55] L. Mannila, V. Dagiene, B. Demo, N. Grgurina, C. Mirolo, L. Rolandsson, and A. Settle, "Computational thinking in K-9 education," in *Proc. Innov. Technol. Comput. Sci. Educ. Conf.*, 2014, pp. 1–29.
- [56] J. Voogt, P. Fisser, J. Good, P. Mishra, and A. Yadav, "Computational thinking in compulsory education: Towards an agenda for research and practice," *Edu. Inf. Technol.*, vol. 20, no. 4, pp. 715–728, Jun. 2015.
- [57] M.-C. Tsai and C.-W. Tsai, "Applying online externally-facilitated regulated learning and computational thinking to improve students' learning," *Universal Access Inf. Soc.*, vol. 17, no. 4, pp. 811–820, 2018.
- [58] B. Wu, Y. Hu, A. R. Ruis, and M. Wang, "Analysing computational thinking in collaborative programming: A quantitative ethnography approach," *J. Comput. Assist. Learn.*, vol. 35, no. 3, pp. 421–434, Feb. 2019.
- [59] F. D. Davis, "Perceived usefulness, perceived ease of use, and user acceptance of information technology," *MIS Quart.*, vol. 13, no. 3, pp. 319–340, Sep. 1989.



**PEI-HSUAN LIN** is currently pursuing the Ph.D. degree with the Department of Engineering Science, National Cheng Kung University, Taiwan. Her research interests include education technology, ubiquitous learning, and augmented reality.



**SHIH-YEH CHEN** received the Ph.D. degree in mobile agent from the Department of Engineering Science, National Cheng Kung University, Tainan, Taiwan, in 2016.

Since 2017, he has been an Assistant Professor with the Department of Computer Science and Information Engineering, National Taitung University, Taitung, Taiwan. His main research interests include e-learning education, social interactions, and mobile networking.

• • •