

The Extraction of Knowledge Factors of Teachers for Physical Computing Education

Hye Seong Park^{#1}, Aoki Hiroyuki^{*}, Ja Mee Kim^{#2}

[#]Major of Computer Science Education, Graduate School of Education, Korea University,
Anam-ro, Seongbuk-gu, Seoul, 02841, Korea
E-mail: ¹phsghost@korea.ac.kr; ²celine@korea.ac.kr

^{*}Shinkoshuppansha Keirinkan Co., Ltd. Content Creation Division Chief Engineer
E-mail: aokih@exext.jp

Abstract — In informatics, physical computing focuses on interactions to realize the real world as a computing system. From 2018, how to teach the physical computing in informatics as a mandatory subject is important. The purpose of this study is to analyze the problems in the physical computing education recognized by secondary school informatics teachers and to provide implications for effective programming education. First, we extracted related keywords of physical computing in the 2015 revised informatics curriculum and science curriculum. Second, extracted keywords are classified into hardware and programming. Third, we developed a questionnaire item suitable for classification keywords. Finally, web surveys were conducted and analyzed for in-service and pre-service secondary school informatics teachers. As a result of the research, it was confirmed that the informatics teachers recognized that physical computing education was helpful for programming education. However, a large proportion of the member's lack of training time and receive appropriate education and training programs, hardware, reduced the level of knowledge about the physical computing element content.

Keywords — physical computing; informatics; education

I. INTRODUCTION

Since the breakdown of Moore's Law in the speed of hardware development, the importance of software has become more important. The importance of software can be seen through the reinforcement of informatics in each country. Middle school informatics in the UK, China, India, Israel, and Japan are required as independent or integrated subjects. In the case of Germany, it is mandatory for the students of this department. In the case of Germany, informatics is mandatory for the students of natural sciences [1], [2], [3], [4].

According to the 2015 revised curriculum of Korea, 17 hours of software education was provided in elementary school, the middle school 'informatics' course was changed to mandatory, and the high school 'informatics' course was changed to general selective subject [5], [6].

The core of the informatics curriculum is to equip students with computing thinking skills that can effectively solve various problems in real-world based on basic concepts and principles of computing.

Through the informatics, learners analyse various problems in a form that can be solved by computer and

construct algorithms based on abstract data. The programming education required to automate this can help creative problem solving and thinking skills based on computing. In addition, through programming training, learners can increase their knowledge of computer operation principles and hardware [7].

Despite the educational effects of programming education, there is a disadvantage that it is easy to lose interest and motivation to learn the grammar of the programming language at the initial stage of programming. It takes a lot of thought processes to solve the problem, but it takes a lot of time to learn the grammar rather than the process of thinking. This can lead to a vicious cycle in which programming problems become dominant in the formal problem-solving process, and the algorithm creation and implementation process become uniform [8], [9], [10]. In other words, programming education can be difficult to motivate because it cannot be seeing and touching. Even if a high-quality curriculum cannot be expected to have a high learning effect if learners cannot be interested [11], [12]. Various programming languages cause cognitive load about learning and teaching new languages. As an alternative to overcoming the disadvantages of programming education,

new physical computing has been added through the 2015 revision curriculum.

Physical computing can visualize real programming through physical computing tools, helping learners to become more interested and motivated. However, various tool selection environments still seem to be a difficult factor in teaching physical computing. This includes a selection of hardware tools, programming environment selection, circuit configuration, and input/output device knowledge. The purpose of this study is to analyse the difficulty factors of the in-service teachers for the successful settlement of the physical computing education implemented from 2018.

II. MATERIAL AND METHOD

A. Physical Computing

Physical computing is the interaction between the real world and the virtual world. That is, the phenomenon

occurring in the real world is detected through various devices, and the sensed values are controlled by the computing device [13], [14].

B. Elements of Physical Computing Contents

The content elements of physical computing can be classified into hardware elements and programming elements.

1) Hardware Elements:

Hardware elements can be seen from choosing a physical computing tool to suit the case you want to implement, to configuring the circuit between the input and output devices. The physical computing tools are set forth in Table 1, which identifies classification, the scope of implementation, and the programming type.

TABLE I
PHYSICAL COMPUTING TOOL

	Arduino [15]	Raspberrypi [16]	Pico Board [17]	Makey Makey [18]	Lego Mindstorm [19]
Classifi-cation	Circuit Type	Mini Computer	Built-in Sensor	Sensor Module Type	Module Type
Scope of Implementation	Sensor and Module Control	Multitasking	only Built-in Sensors can be used	only Input Devices can be implemented	Multitasking
Programming type	block, text	block	block	block, text	block, text

In terms of implementation, sensor board is limited because it can only use sensor built-in board, and MakeyMakey is limited to the input device. Lego Mindstorm has the disadvantage that the price is higher than other boards.

The configuration of the circuit is when all the circuits are to be composed of lead wire, resistance, input / output device when a specific sensor is embedded in the board, and when each sensor is composed of modules, the required sensor module is connected and used. However, in order to construct all the circuits, it can add to the cognitive load of

knowing the whole electronic circuit knowledge [11], [12], [20], [21].

2) Programming Elements:

The programming element refers to the selection of the programming environment supported by the physical computing tool, which is a hardware element, to the program implementation and debugging suitable for the hardware configuration. The physical computing Programming Environments are set forth in Table 2.

TABLE II
PROGRAMMING ENVIRONMENT

	S4A [22]	Entry [23]	Arduino IDE[15]	Python [24]
Support OS	Windows/Mac/ Linux/ Raspbian	Windows/Mac	Windows/Mac/ Linux	Windows/Mac/ various OS
Monitoring	partial support	support	support	support
Cording Type	block	block	text	text
extensibility	-	-	C	C
input/output	partial support	support	support	support
Hardware connectivity	to upload firmware in the Arduino IDE environment	connection with various boards in entry environment	USB driver install and port setting	run python on a raspberry pie
feedback	not support debugging	not support debugging	Support debugging	Support debugging

C. Physical Computing as Informatics

In the middle school informatics of the 2015 revised curriculum, the achievement standard of physical computing is the processing of data using sensor and implementation of motion control program. You can actually see and touch the data and information unit of the informatics content system, the problem solving and programming unit using the physical computing tool.

1) In Informatics:

In the 2015 revised Informatics curriculum, physical computing education follows the process of input, processing, and output. Input is the process of detecting the light, sound, and movement of the real world through the sensor. Processing is a process of creating a program that operates according to an input value. The output can show the processing result on the screen or by the operation of a device such as a servo motor. The process of input, processing, and output includes both data and information units and the core concepts of problem-solving and programming. The core of physical computing is interaction, which results in changes in the real world. This is consistent with the improvement of problem-solving ability in the real world which is the core of software education [5], [13], [14].

Through software training, we want to solve real-world problems. The solution to the problem is the recognition of the problem, the creation of the algorithm, the implementation through programming, and the repetition of the process of debugging. Despite the research findings that programming education is effective in improving creativity, problem-solving ability, and logical thinking ability [11], [12], the existence of many programming languages comes at the burden of learning a new language each time. Focusing on the programming language takes up most of the given time and reduces the amount of time for self-reflection. In other words, they do not develop the ability to think which is the core of actual software education [8], [25], [26], [27]. Physical computing education has the following advantages as it can be implemented directly on the phenomena commonly seen in the surroundings. Because each person sees and feels that the problem is perceived differently, various creative works can be realized. It is possible to visually confirm that the output result changes depending on the environment change so that the debugging ability for errors can be raised. Sharing of various works can lead to an expansion of thinking ability.

TABLE III
CONTENTS OF SCIENCE CURRICULUM

2015 Revision Science Curriculum: Common Curriculum Electrical and Magnetic Domains	Core Concept	Content Element	
		Electric	Elementary school (grades 5 ~ 6)
	Electric circuit		Electrical Circuit, Voltage, Current, Resistance

The basic circuit knowledge of physical computing education is dealing with in the electrical and magnetic domains of the 2015 revised science curriculum [28]. In order to educate physical computing in the informatics curriculum, it is necessary to consider the connection between science and curriculum.

TABLE IV
KEYWORD EXTRACTION RESULT

Generalized Knowledge	Informatics	Microcontroller, I/O Device, System Configuration, Programming, Control, Sensor-based Programming, Implication
	Science	Electric Charge, Electric Power, Electromotive Force, Electric Circuit, Voltage, Current, Resistance, Electric Safety
Achievement Criteria	Informatics	I/O Device, Sensor, Data Processing, Operation Control Program
	Science	Batteries, Light Bulbs, Wires, Serial / Parallel Connection
Teaching Method	Informatics	Tool Selection, Programming Language Selection, Operation Design, Programming process, Efficient Program
	Science	Electric Safety, Electric Circuit
Evaluation Method	Informatics	Hardware Configuration, Program Design, Accuracy, Creativity, Efficiency
	Science	Electric Circuit

Table 4 shows the physical computing-related keywords extracted from the Informatics curriculum and Science curriculum. Keywords were extracted from 'curriculum contents and generalized knowledge', 'achievement standard', 'teaching · learning method' and 'evaluation method' of each curriculum. The result of keyword extraction is as follows. Informatics curriculum focuses on the selection and programming of the tool. Science curriculum concentrates on the electric circuit knowledge.

Physical computing education should be preceded by hardware configuration. Circuit knowledge is required for hardware configuration, but physical computing in the informatics curriculum does not deal with circuit knowledge. Circuit knowledge is dealing with in the electrical and magnetic domains of the science curriculum.

Despite the many advantages mentioned above, the physical computing education has the disadvantage that the circuit knowledge required for hardware configuration increases the cognitive load for both learners and the instructors [10], [15], [20], [21]. There is a limit to training both hardware configuration and programming in a limited number of hours. Thus, selecting a tool that minimizes the perception burden of circuit knowledge will facilitate programming focus.

D. Object of Study

In the 2015 revised informatics curriculum, middle school informatics is mandatory, and high school informatics is

optional. Therefore, this study was conducted on 8 of secondary school in-service informatics teachers, and 16 of secondary school pre-service informatics teachers.

E. Research Procedure

The procedure and method of the whole study are set forth in Fig. 1, and the detailed procedures and methods are as follows.

First, key concepts of computing System in the 2015 revised informatics curriculum keywords were extracted from the curriculum content elements and generalized knowledge, achievement standards, teaching and learning methods and evaluation methods of physical computing. Second, contents related to electric circuits in the core concepts of electricity and a magnetic domain of common science curriculum of the 2015 revised science curriculum were extracted from curriculum content elements, generalized knowledge, achievement standards, teaching and learning methods and evaluation methods. Third, the keywords extracted from the first and second steps are classified into hardware and programming. Physical computing consists of hardware device configuration and programming for hardware device control. That's why it's important to choose the right hardware and programming

environment for your physical computing device. The result of the third step is set forth in Table 5. As a result of Table 5, it includes the selection of the boards to be used for physical computing, the input / output devices to be used after selection, and the system configuration between them. The programming part is from the selection of the programming language required to control the operation of the devices constituted in the hardware part to the implementation of the control program. Common factors include accuracy, efficiency, and creativity.

TABLE V
KEYWORD CLASSIFICATION

Hardware	Microcontroller, I/O Device, Sensor, System/Hardware Configuration, Tool Selection, Electric Circuit, Electric Safety, Accuracy, Efficiency, Creativity
Programming	Operation Control Program, Data Processing, Programming Language Selection, Program Design, Accuracy, Efficiency, Creativity

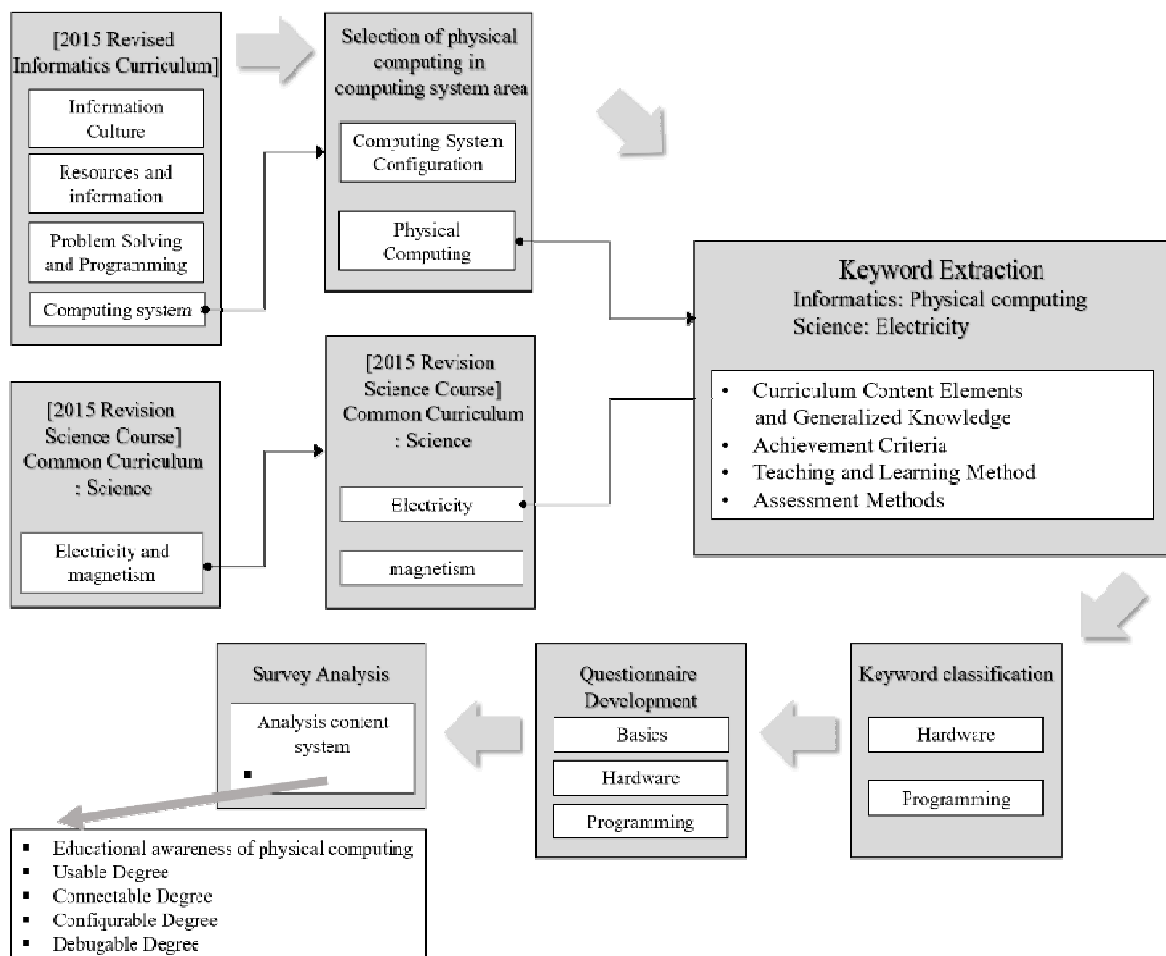


Fig. 1 Procedures and methods of research

Fourth, we developed a questionnaire to analyse the problems in the physical computing education that informatics teachers perceive. Finally, the results of the questionnaire were analysed in terms of educational perception of physical computing, the degree of usability of physical computing tools, the degree of connectivity of physical computing tools, configurability of circuits and lessons, and debuggability.

F. Development of Survey Tools

Based on the results of the keyword classification in Table 5, the development of the questionnaire tool was conducted

by developing items corresponding to the keywords. We developed the items that correspond to keywords such as tool selection, input / output device, circuit configuration, system configuration, language selection, and control programming. The developed questionnaire consisted of 25 items in terms of basic information, recognition, hardware, and programming. To analyze the questionnaire contents, we divided the hardware and programming related items into four parts: usable degree, connectable degree, configurable degree, and debuggable degree. The composition of the reclassified questionnaire is set forth in Table 6.

TABLE VI
DEVELOPMENT OF SURVEY TOOLS

	Contents	Contents Number	
		in-service teacher	pre-service teacher
Basic Information	gender	1	1
	school type	2	-
	ages	3	2
	final education	4	3
	education/training experience	7	5
	work/teaching experience	5	8, 9
Recognition	the degree of knowing	6	4
	why do you want to be educated?	8	6
	why do you not want to be educated?	9	7
	priority of education	23	
	the need for education	24	
	effect of education	25	
Usable Degree	physical computing board usability	10	
	classification and usability of physical computing devices	12	
	breadboard availability	14, 15	
	the degree to which the physical computing programming environment can be used	17	
	arduino IDE operation control function usability degree	21	
Connectable Degree	the degree to which you can connect with the Arduino board from the Arduino IDE	13	
	the degree to which the S4A and the physical computing board can be connected	19	
	entry and physical computing board connectivity	20	
Configurable Degree	choosing the board, you want to use in your class and why	11	
	ability to read, write, and modify schematics	16	
	choosing the programming language, you want to use in your class and why	18	
Debuggable Degree	error checking and fixing	22	

III. RESULTS AND DISCUSSION

A. Educational Recognition of Physical Computing

The educational recognition of physical computing was analysed by the questionnaire about the degree of knowing about physical computing, the reason for wanting to be educated, the reason for not being educated, priority, necessity, and effect.

The results are as follows. First, 87.5% of the in-service and pre-service informatics teachers said they know about

physical computing. 87.5% of in-service teachers and 57.1% of the pre-service teachers responded that they had to attend the physical computing education. 83.3% of in-service teachers responded that they did not have time because of lack of physical computing education, and 50% of pre-service teachers answered that they could not find appropriate courses. Second, 100% of the in-service teachers and 93.8% of the pre-service teachers answered that they need physical computing education. For reasons of necessity, 62.5% of the in-service teachers and 25% of the pre-service

teachers answered that it is helpful to the program education. 37.5% of the in-service teachers and 27.5% of the pre-service teachers were found to help improve creativity.

B. Problems in Physical Computing Education

The problems in the education of physical computing were analysed by classifying the questionnaires conducted

by the in-service and pre-service teachers of secondary school into the usability, connectivity, configurability, and debuggability. The overall results of the problem analysis are set forth in Table 7

TABLE VII
PROBLEMS OF PHYSICAL COMPUTING EDUCATION

UNIT: M(SD)

	Contents	In-service teacher	Pre-service teacher
Usable Degree	Physical Computing Board	2.59(0.61)	2.32(0.67)
	Input/Output Device	3.38(0.21)	3.77(0.32)
	Breadboard	2.78(0.73)	3.00(0.89)
	Programming Language	3.13(0.23)	3.25(0.29)
	Arduino IDE Control Fuction	2.91(0.29)	3.25(0.10)
Connectable Degree	Arduino	3.40(0.19)	3.98(0.20)
	S4A	2.38(1.06)	2.94(1.44)
	Entry	2.62(1.19)	2.19(1.33)
Configurable Degree	Circuit Design	3.09(0.32)	3.23(0.28)
	Board Selection	Arduino Board	
	Programming Language Selection	Block-based	Block-based, Text-based
Debuggable Degree	Debugging	3.13(1.25)	3.88(0.81)

The usable degree is the usability of board, I/O device, breadboard, programming language, and Arduino IDE operation control function. The connectable degree is enough to connect the Arduino board to the PC, the degree to which the S4A and the Arduino board can be connected, and the connection between the entry board and the board. The configurable degree is the degree of configurability of the circuit diagram creation, modification, circuit diagram implementation, and the choice of the tools used in the class. The debuggable degree is about whether a programming error can be detected and corrected. The responses to the questionnaires were as follows: 'not at all (1)', 'not (2)', 'normal (3)', 'can do (4)', Likert 5 point scale and Likert 4 point scale of 'not at all (1)', 'not (2)', 'can do (3)' and 'can do well (4)' And the results of the analysis were unified to the Likert 5 point scale.

IV. CONCLUSIONS

The concept of 'physical computing' has been added to the 2015 revised informatics curriculum, and physical computing education will be conducted in 'informatics' of secondary schools from 2018. The purpose of this study is to present the implications for effective programming education by analysing the problems in the physical computing education that the in-service and pre-service informatics teachers are aware.

The results of the study are summarized as follows.

- First, in recognition, the necessity and effect of physical computing education are recognized, but the

problem is the time to receive education and the lack of proper education program.

- Second, in terms of usability level, the degree of usability of hardware tools and programming environment was asked with Likert 5 point scale. As a result, it was analysed that the average of the in-service and pre-service informatics teachers was insufficient to carry out the class with 3 point scale.
- Third, in terms of connectability, we asked the degree of connection between the programming environment and the physical computing board with a Likert 5 point scale. As a result, it was analysed that the average of the in-service and pre-service informatics teachers was not enough to carry out the less than 3 or 3 point scale.
- Fourth, in terms of configurability, we asked the degree of likelihood of creating and modifying a schematic to implement physical computing with a Likert scale of 5 points. As a result, it was analysed that the average of the in-service and pre-service informatics teachers was insufficient to carry out the class at the early stage of 3 points. For both in-service and pre-service informatics, teachers prefer the Arduino board for the choice of boards, while in-service teachers prefer block-based programming languages, while pre-service teachers prefer both block-based and text-based programming languages Respectively.
- Fifth, the average of pre-service teachers was higher than that of in-service teachers in terms of debugging ability. Because the block-based programming

environment does not show error messages, the debugging average of in-service teachers preferring block-based is lower than the pre-service teacher. However, it was analysed that the average of the pre-service teachers is 4 points or less, which is insufficient to carry out the class.

Based on the results of this study, we will reconsider the education of physical computing in the following four aspects.

- First, various training programs for teachers should be organized. In order to facilitate the education of physical computing, teachers should be able to recognize the contents of physical computing contents and communicate them to students.
- Second, an instructional design that can improve thinking ability should be supported. It is necessary to be able to apply real-life examples to lessons by simply avoiding the lessons of the following formula. Students should be able to select topics that can interest and motivate them, and be able to choose appropriate physical computing tools for the topic.
- Third, we need to consider the linkage and sequence of physical computing education. Elementary, middle, and high schools should have a spiral curriculum to ensure that the elementary, middle, and high school education is linked.
- Fourth, debugging ability should be improved. Informatics brings creativity through various thinking. Because there are various solutions according to recognition even with the same problem, the verification and implementation of the algorithm is a process of constant thinking. Physical computing can visualize actual programming results and enhance debugging capabilities. There is a need for research on physical computing education in a text-based programming environment that outputs error messages.

This study has implications for the effective programming education by analysing the problems in the physical computing education recognized by in-service and pre-service informatics teachers. More research is needed to find out more informatics teachers are involved in.

ACKNOWLEDGMENTS

This work was supported by the National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIP) (No. 2015R1C1A1A02036950).

REFERENCES

- [1] H. J. Yang, W. G. Lee, and J. M. Kim, "Analysis of Informatics Curriculum in Secondary Schools in Overseas," in Proc. ICICPE'17, 2017, paper vol. 1, pp. 109.
- [2] G. S. Kim, "Implications of Computer Education in Korea from the U.S., U.K. and Germany Computer Curriculum," Journal of the Korean Association of Information Education, vol. 20, pp. 421-432, Aug. 2016.
- [3] H. J. Choe, T. O. Song and T. W. Lee, "Comparative Study of Informatics Subject Curriculums and Textbooks in Middle School Between Korea and England," Journal of the Korea Society of Computer and Information, vol. 21, pp. 145-152, Feb. 2016.
- [4] Lehrplan PLUS. (n.d.). Retrieved February 10, 2017, from http://www.lehrplanplus.bayern.de/fachlehrplan/gymnasium/7/nt_gym
- [5] Informatics Curriculum, Ministry of Education, Republic of Korea. 2015-74, 2015.
- [6] S. J. Lee, J. M. Kim, and W.G. Lee, "Analysis of Factors Affecting Achievement in Maker Programming Education in the Age of Wireless Communication", Wireless Personal Communications, vol. 93, pp. 187-209, 2016
- [7] J. S. You, M. H. Lee, "Effects of a Programming Class Using Dolittle on Enhancing Creativity, Problem Solving Ability, and Interest in Programming," Journal of the Korean Association of Information Education, vol. 13, pp.443-450, 2009.
- [8] Y. C. Kim, J. M. Kim, and W. G. Lee, "A Case Study on Reflection Using Worksheets for Elementary School Students in Programming Learning," Journal of the Korean Association of Information Education, vol. 16, pp.21-31, 2012.
- [9] L. Murphy, G. Lewandowski, R. Mccauley, B. Simon, L. Thomas, and C. Zznder, "Debugging: the good, the bad, and the quirky - a qualitative analysis of novices' strategies," in Proc. SIGCSE'08, 2008, p.163.
- [10] S. Fitzgerald, R. Mccauley, B. Hanks, L. Murphy, B. Simon, and C. Zander, "Debugging from the Student Perspective, Education," IEEE Transactions on, vol. 53, pp.390-396, 2010.
- [11] Y. J. Jang, W. G. Lee, and J. M. Kim, "The Changes of Middle School Student's Perception and Achievement based on the Teaching Method in Physical Computing Education," Indian Journal of Science and Technology, vol. 9, Jun. 2016.
- [12] M. Przybylia, R. Romeike, "Physical Computing and its Scope – Towards a Constructionist Computer Science Curriculum with Physical Computing," Informatics in Education, vol. 13, pp. 241-254. 2014.
- [13] (2017) Physical Computing Homepage on Wikipedia. [Online]. Available: https://en.wikipedia.org/wiki/Physical_computing
- [14] D. O'Sullivan, T. Igoe, Physical Computing: Sensing and Controlling the Physical World with Computers. 1st ed., Thomson, Boston, 2004.
- [15] What is Arduino? (n.d.). Retrieved January 7, 2017, from <https://www.arduino.cc/>
- [16] Teach, Learn, and Make with Raspberry Pi. (n.d.). Retrieved January 2, 2017, from <https://www.raspberrypi.org/>
- [17] PicoCricket , Invention kit that integrates art and technology. (n.d.). Retrieved December 13, 2016, from <http://www.picocricket.com/>
- [18] Makey Makey | Buy Direct (Official Site). (n.d.). Retrieved January 1, 2017, from <http://makeymakey.com/>
- [19] Lego Mindstorms. (2018, February 10). Retrieved February 13, 2018, from <https://www.lego.com/ko-kr/mindstorms>
- [20] H. K. Jeon, Y. S. Kim, "Design of Physical Computing Teaching-tool Based on Knowledge Structuralization in Software Education at Elementary and Secondary school," in Proc. The Korean Association of Computer Education, 2016, paper.20.2, pp. 39.
- [21] S. J. Kim, Y.J. Jeon, and T.Y. Kim, "A Practical Approach to Arduino Programming for the Physical Computing Section of the Informatics Curriculum in Korean Middle School," in Proc. The Korean Association of Computer Education, 2016, paper.20.2, pp. 29.
- [22] About S4A. (n.d.). Retrieved November 5, 2016, from <http://s4a.cat/>
- [23] The Entry website. (n.d.). Retrieved May 13, 2017, from <https://playentry.org/>
- [24] Welcome to Python.org. (n.d.). Retrieved April 2, 2017, from <https://www.python.org/>
- [25] Y. J. Jang, W. G. Lee, "Analysis of the Physical Computing Education Objectives in 2015 Revised Middle School Informatics," in Proc. The Korean Association of Computer Education, paper.20.1, p. 101, 2016.
- [26] B. G. Yu, J. M. Kim, and W. G. Lee, "Implication for Construction Computing System Unit of the 2015 Revised Curriculum," The Journal of Korean Association of computer education, vol. 19, pp.31-40, 2016.
- [27] A. Gomes, A. J. Mendes, "Learning to program – Difficulties and solutions," in Proc. ICEE'08, 2008, P. 283.
- [28] Science Curriculum, Ministry of Education, Republic of Korea. 2015-74, 2015.