

# Enhancing the Student Engagement in an Introductory Programming: A Holistic Approach in Improving the Student Grade in the Informatics Department of the University of Surabaya

Budi Hartanto

Informatics Engineering, University of Surabaya, Surabaya, Indonesia  
budi@ubaya.ac.id

**Abstract.** Student engagement has long been known can increase the student performance. However bring this concept to practice may not be as easy as it sounds. Some factors, such as the teacher, the students and their background, the course content, the academic atmosphere, the study culture, etc., influence its implementation. This paper presents the practice that is applied in the Informatics Department of the University of Surabaya to enhance the student engagement in the introductory programming course (i.e. Algorithm and Programming). This course is one of the course that is commonly known as difficult and make many Informatics students in the University of Surabaya dropped their study in the Department. The practice to enhance the student engagement in the Informatics Department of the University of Surabaya is designed to fit best with the condition in the Department. As a result, the students' performance increase two grade level compared to the students' performance in the previous years.

**Keywords:** Student engagement, introductory programming, increasing the student performance.

## 1 Introduction

Algorithm and Programming (Alpro) – an introductory to programming course in Informatics Department of University of Surabaya – has long been known as difficult and make many students in Informatics students in the University of Surabaya dropped their study in the Department [1, 2, 3, 4, 5]

Alpro is used as a prerequisite before the students can take the other programming course that is Object Oriented Programming (OOP). To take OOP, the students must pass Alpro with a minimum grade: C. In Semester I of academic year 2012/2013, 45% of the students failed to pass Alpro. In the following year – Semester I 2013/2014, the percentage of students who failed to pass Alpro is 44%.

The efforts to reduce the number of students who failed to pass Alpro were tried for several years. However these efforts tried to fix the problem from the cognitive point of view only. In this case the teachers tried to fix the problem of the student's

inability to read and understand student code [6], the difficulty in understand the syntax and semantics [7], etc. As a result, there were modification in the breadth and depth of course content, the level of difficulty of the assessment, the delivery method, and so on. Unfortunately, the number of students who failed to pass Alpro were still very high.

This research tries to find methods, approaches, or strategies that can be used to decrease the student failure in Alpro. A lot of research showed that the student engagement is a key point to improve the student performance [2, 3], [8, 9, 10]. Therefore increasing the student engagement seems as a promising method to solve the problem in Alpro. Unfortunately, the practical approaches and strategies to improve the student engagement may vary from case to case. Some factors such as the type of the course, the education culture, the student background, etc. may influence the approaches and strategies that can be used.

In this paper, some practical approaches and strategies that are used to increase the student engagement in Alpro, in the Informatics Department of the University of Surabaya are shown and discussed. These approaches and strategies are proven can effectively increase the students' engagement that in the end increase the students' performance. As a result, the numbers of students who failed to pass Alpro in Semester I, 2014/2015 is projected to become 12% only. This is 28% lower than the one in the previous year. In Addition, the average mark of the students is projected to increase two grade levels, from BC to AB.

## 2 The Historical Background of Algorithm and Programming

Alpro is offered in the Informatics Department of the University of Surabaya as an introduction to programming course. This course is designed to be administered in a classroom for four unit hours (1 unit hour = 50 minutes) per week. In addition, the students can also take two more unit hours (optional) in the computer lab. A teacher administers the meeting in the classroom, while a student assistant administers the meeting in the lab.

The meetings in classroom were intended to give a conceptual view of programming to the students. In some meetings the students were also asked to do some programming exercises. The solutions of these exercises can be written in the book. The intention to use classroom instead of lab was meant to increase the number of exercises that could be solved by the students. This was possible because the students do need to run and debug the program that they wrote. Sometimes, finding errors from even a logically correct program can spend a lot of time.

In this scenario, the students were expected to copy the solutions in the book to the development software in their computer, at home. Unfortunately, not many students performed this. As a result, it was common to find students that never use the development software. Some of them even did not have the software installed in their computer.

The existence of the lab session did not help too much. Because the students' attendance were not compulsory, the number of students in the lab session dwindle quickly week by week. The lab session that was meant to help them in the programming problem

and to support them to understand programming better, did not fascinate the majority of the students.

As described in the Introduction section, 45% of the students failed to pass Alpro in Semester I 2012/2013, and 44% in Semester I 2013/2014. The students are considered failed to pass Alpro if their grade is D or E.

### 3 Holistic Approach to Enhance the Student Engagement

Theoretical and empirical research suggests that academic engagement can improve the student performance [2, 3], [8, 9, 10]. However the practical strategies to improve the student engagement may differ from case to case. A holistic approach may be required to truly actualize the student engagement in class.

In the following subsection I will discuss some factors that influence the student engagement, as well as the approaches and strategies that I and the teaching team perform to improve the student engagement.

#### 3.1 The Course Content

Learning programming is a skill [7]. Human commonly masters a skill by performing hands on experience on the skill to be mastered. In learning how to program, the hands on experience are writing the program and learning how to find and fix the error.

That is the reason why teachers always ask students to write the program for solving the task in the exercise. Unfortunately, experience from the previous semesters showed that many students were not motivated to write the program. The engagement of the students to the class activity were very low.

One contributed factor to this condition is the content of the course that do not support them to understand well and cannot arouse their excitement. In order to increase the student engagement from the course content factor, we perform the following approaches and strategies:

- Problem based learning  
One reason that brings the students disinterest in writing programs is the absence of clear relation between the problem in the exercise to the problem that they thought they would face in real life [11]. For example, one common exercise in Alpro is asking the student to write a program to find factorial numbers. The majority of the students may thought that the program to find factorial numbers was meant as a programming exercise only. They thought that they will never face this kind of problem when they work as programmers later on. This kind of thought will certainly decrease the students' motivation to write the program.

Our strategy for this case is to rewrite all the given exercises in the previous semesters, and make them similar to the problem that the students thought they may face in real life. By this kind of exercise, we expect that the exercise would fit better to the student expectation if they work as a programmer later on.

As an example, to ask the students to write a factorial program, we create a story of a certain matchmaking company that wants to hire a programmer to solve a certain problem. In this case the company problem is to find out the number of possible combination to match the men and women that are already registered in that company. Certainly the problem of finding the number of combination can be solved by finding a factorial number. Applying the knowledge to be learnt to the real life problems can increase the student interest in learning [12].

- Building on prior knowledge

It is important to know that many students in the Informatics Department of University of Surabaya have no background in programming. Introducing programming as a new and isolated entity will make it hard to be understood or to be retained in the students' memory. Human memory works by connecting the new information to the existing information that they had in memory [13].

Because the students may not have a programming background then we need to link the concept of programming to the events that they have experienced in their life. We use analogy. For example, to introduce the concept of how a program works, we gave an analogy of cooking an instant noodle. When a person cooks an instant noodle using the exact steps as written on the noodle package, the detail of the instructions, the usage of the appropriate words, and the correct order of the instruction will influence the result that the person gets. Using this analogy, the students are expected can imagine the concept of programming.

In addition, in explaining the new topic we tries to link it to the previous topics. Learning can be effective if learners can connect the new material with the one that already exists in memory [14]. Linking the new topic to the previous topic can make the new topic become easier to be understood and also strengthen the knowledge of the previous topic.

- Learning by example

Human learn by example [15]. Therefore it is important to provide an example and let the students do the exercises by mimicking the work in the example. In previous semesters, examples were also given in the end of certain topics. Unfortunately, after further analysis we found that some examples were stressing on how to use the program constructs rather than solving a problem using the program constructs. An example that shows how to solve a problem is more beneficial than an example that shows how to use the program construct.

We checked all examples in the previous semesters and revised them to become examples that show how to solve certain problems. We ensure that the first exercise has a similar solution to the example solution but it is wrapped in different case study. Therefore, the work in the example can be used as a foundation of thought to do the first exercise. In addition, we also ensure that all topics are always ended with an example.

- Learning by doing

Because programming is a skill [7], the students should have hands on experience of it. Learning how to program can be mastered if the students often practice writing the correct program [12, 13]. Unfortunately experience from the previous semesters showed that not many students were motivated enough to write the program, especially in the development software.

In this research we applied some strategies to encourage and to push the students to write the correct program in the development software. The first strategy that we performed was asking the student to install the development software in their computer. The development software that we use is Visual Studio Express Version that can be downloaded and used for free. To motivate the students, we rewarded a 10 point mark to the students that can show that Visual Studio are already installed in their computer. Some students sent a screen capture of the software and some students were happily sent their selfie photo with Visual Studio displayed on their monitor. The good thing is that all students did this task.

The existence of the development software in the student computer is the first step required to encourage the student to write the program in the development software. At any time the student wants to write the program, the development software is already in their computer, ready to be used. On the contrary, without the development software ready, the students that originally want to write a program, must spend some time to download and install the development software. The lengthy process of downloading and installing the software may put out the students intention to write the program.

The second strategy is to encourage the students to bring their laptop to the classroom. With this laptop, the students can write the exercise solution directly on the development software. They do not need to write the solution in the book first and copy it to the development software later on. Because of this encouragement and other strategies from other factors, there are many students bring their laptop to the classroom in this semester. Writing the program on the development software enables the students to immediately see the result of their program. A program that can run and display the result correctly often give a certain level of satisfaction to the programmer.

If the students did not bring laptop, we still wanted the students to write the program in their book. Therefore we repeatedly reminded the students that learning programming is a skill and can only be mastered by practice writing programs. We told the students that we would really happy if they can show their program to us to get any feedback. Many students were happily showed their program to us for a feedback. The teacher feedback is one of the situational factors that affect student learning [17].

The third strategy is to balance the time used to deliver the knowledge and to practice writing the program. We believe that the students need to practice writing the

program, but we also believe that they need to understand the knowledge that underlie the practice. Therefore learning the knowledge and practice is equally important [18] Because of this we strive to use only 50% of the allocated time in every meeting to deliver the knowledge and to use another 50% for practice.

Although this strategy looks simple, its implementation was not easy. In the previous semesters, one topic was usually delivered as a whole (just like a book chapter) before it was continued with the exercise. Because of this strategy, we have to split a certain topic into several sub topics. We have to ensure that the sub topic can be delivered in more or less 50% of the allocated time. The application of this strategy can successfully make the students understand the topic better.

### 3.2 Academic Atmosphere

In previous semester, it was common to see a passive and apathy students in class. These students preferred to use their time for chatting with their friends instead of doing the exercises. As a result, the students that actually wanted to do the academic activity eventually got affected and joined the chat. The engagement to the class activity was low. The reasons that make the students become passive and apathy are possibly from the students' disability to do the exercise but they are too shy or afraid to ask the teacher about how to do it. This is a common academic problem in many education level in Indonesia. The following are some strategies that we do to improve the academic atmosphere.

- Encourage collaborative work

Collaborative work or peer learning is known to be good to enhance the student engagement [19, 20]. In this semester, we encourage the students to perform collaborative work. They can informally make a group and have discussion to find the solution of the exercise. From our observation we can see that in every group there are always one or two students who look smarter than the others. These students were not reluctant to share their knowledge to the less-able students when they asked for help. The students were not shy or afraid to ask questions anymore because now they could asked the questions to their friends. No more reason for the less-able students to say that they do not know how to solve the problem because they can discuss the solution of the problem with their friends.

- Encourage a good student-teacher communication

A good communication between the student and the teacher is very crucial to make the student become more open to the teacher. It is well known that Indonesian students are very shy and very reluctant to ask to the teacher. From my own experience as a student and from my observation as a teacher, this behavior was possibly formed due to the traumatic experience of asking questions in class. It is common for Indonesian students or even teachers to give funny comments about the questions or about the person that asks the question. Therefore, many students prefer not to raise questions to avoid the embarrassment.

In order to achieve a good student-teacher communication, we try to get closer to the students by going around the classroom at the exercise time. Therefore the students do not need to be afraid to be mocked by their classmates because now they can ask us personally and no need to say it loud in front of their classmates. In addition, the other strategy that we did – that encourage the students to show their work to us – align with this strategy. We performed a verbal and nonverbal immediacy when the student showed his/her work to us. Verbal and nonverbal immediacy support a better learning [21]. This strategy is proven can make a good student-teacher communication and can make the student become more open to the teacher.

### 3.3 The Teacher

Another factor that influence the student engagement is the teacher role in class. Some approaches and strategies are applied in this semester to increase the student engagement from this factor.

**Be a Motivator, not Just a Provider.** In many classes, it is very common to see teachers spend all of their teaching time to deliver and explain the knowledge to the students. This behavior come from the common thought that the teachers' job is to deliver knowledge to the student. Not many teachers realize that it is also their responsibility to motivate the students and arouse the students' interests.

With the current advance of the searching engine, explaining the full features of a topic are not critical anymore. The teacher can just explain the most common and important part of the knowledge and ask the students to explore deeper by searching from the internet search engine. By this approach the teachers can have some spare time that can be used to motivate the students or to arouse the students' interest. This is especially needed for students with a weak background in the subject [22]. A motivated student would willing to enrich themselves with a deeper knowledge.

**Give Feedback, not Only Judgment.** Assessment can be classified into two categories: formative and summative. The formative assessment is used to gather information for the specific purpose of improving learning (for learning), while the summative assessment is used to gather information for the purpose of recording and reporting (of learning) [23, 24]. In formative assessment we can get the feedback of the current teaching and learning condition and use the feedback to improve them. On the other hand, the summative intention is solely to form a judgment [25].

There are six assessments that are performed for the students in Alpro. These assessment are administered in Week 4, 6, 8, 12, 14, and 17. In the University of Surabaya, the teaching and exam weeks have a pattern of 7, 2, 7, 2. The first seven weeks are used as the teaching weeks and the next two weeks are used as the exam weeks. This pattern is repeated again after the exam weeks. The teacher can perform any assessment in the teaching weeks but cannot perform a lecture in the exam weeks.

Although all assessment marks contribute to the students final grades, five of them actually can be considered as formative assessments. This is because the result of the assessments can be used to give feedback to the teacher and the student about the



current condition of the teaching and learning. The sixth test is performed in the second exam week and is used solely to measure or judge the student performance.

The number of students in each Alpro class is around 40 to 50. If for example the total number of students in a particular semester is 300 then there will be 6 classes open for Alpro. The number of teacher is usually set to be the same as the number of classes.

In order to avoid the marking variant between teachers we usually assign one teacher to mark only one questions in a particular assessment but he/she should mark the work of the students in all classes. If the number of students is 300 for example, then one teacher will mark 300 students' works. Checking the works of 300 students in a short time is challenging and need a big commitment to finish it before the next test. Returning the students' work at or after the next test will make the formative assessment lose its essence as a feedback instrument. Unfortunately, in the previous semesters due to the tight schedule of the assessments, teachers often return the assessment result at or after the next assessment.

In this semester, we make a commitment to return the assessment result at most one week after the assessment. With this commitment, the formative assessments can now be used to give a real feedback to the teacher and the student about their performance. As described by [26], an assessment can become catalyst to the student performance, the validity of the program, and the validity of the student performance. A late returned assessment can only benefit in the validity of the student performance.

### 3.4 The Students

One last factor in our consideration to improve the students' engagement is the students themselves. Some approaches and strategies that are applied in this semester from the students' factors point of view are described below.

- Increase the student self-efficacy

Many ordinary students lack of belief that they can solve the given problem with their current capability (lack of self-efficacy). Self-efficacy itself is defined as beliefs that build a person's motivation and confidence in achieving a desired outcome [27, 28]. Without self-efficacy the students will give up too early and not use their best efforts to find the solution of the problems.

There are several strategies that we perform to increase the student self-efficacy. As described by [17], the students will have a high self-efficacy if the students can solve difficult problem or do not spend too much effort to solve a problem. On the contrary, the students that solve an easy problem or spend too much effort to solve a problem will have a low self-efficacy. Therefore our strategy to increase the students' self-efficacy is wrapping some easy problems with applied cases to make them appear like advance problems. As an example, the problem of finding a factorial number is wrapped in an applied case such as finding the number of possible combination to pair man and woman in a matchmaking company. This wrapper will make the original problem of finding a factorial number looks more advance. However, finding the solution for this problem is not hard.



In another strategy to increase the student self-efficacy, we create an exercise where its solution is similar to the example given. However we wrap this exercise in different case study to make it looks different from the case study in the example. As a beginner programmer, the students usually do not realize that the solution for the exercise is actually very similar to the one in the example. We expect the student to consider the difficulty of the exercise is at medium level (not at easy level) and we also expect the students can solve the problem without too much effort.

- Provide help as needed

Helping students in a medium or a big size class can be problematic. On one hand we want the students to ask question if they have problem, but on the other hand we know that we will not be able to answer many of the students' questions if many of them raise questions at the same time. The inability to answer the students' question may frustrate the students and demotivate them to learn.

The strategy to overcome this problem is related to the strategy in section 3.2. In that strategy we encourage the student to perform collaboration learning so they can help each other. With this collaborative learning the students only ask questions to the teacher when all students in the group do not know the answer. An effective collaborative learning, helps reducing the number of questions significantly.

Although the collaborative learning can reduce the number of questions, it is possible that there are still many questions raised to the teacher at the same time. In the programming class, the condition can be worse. The correct program for a given exercise can vary from student to student. A good teacher should be able to see the student intention in his/her program and give help according to the current student program. The process to understand the student program itself can take a considerable time before another time is needed to think how to align the student intention with the correct solution and explain the solution to the student. Only experience teacher in programming that can performed these process quickly. Based on this need, our strategy to provide help to as many students as possible is by assigning a senior and experience teacher in programming [5].

## 4 Evaluation

The approaches and strategies described in here are already applied to the Alpro class in Semester I 2014/2015. As a result, the student engagement in class is improved a lot. The students become active in every activity given and the academic atmosphere in class is very positive.

The data in table 1 shows the percentage of students that failed Alpro in Semester I 2012/2013 and Semester I 2013/2014. Based on the students' whole marks (marks in mid and final semester), 45% students failed to pass Alpro in year 2012/2013 and 44% in 2013/2014. This percentage is similar to the percentage of the students who failed to pass Alpro if we used their mid semester marks only to judge their grade (41% and 40%). The actual number of students that failed to pass Alpro is 4% worse than the number of students that failed to pass Alpro based on their mid semester marks only.

**Table 1.** The comparison of the student failure and the student mid semester marks

	Number of students	Student failure		Average marks	
		based on the mid semester marks	based on the whole marks	based on the mid semester marks	based on the whole marks
2012/2013	193	41%	45%	59.62	54.29
2013/2014	245	40%	44%	61.06	57.34
2014/2015	255	12%	-	75.29	-

Therefore based on the students' marks in the mid semester, we can estimate and project the percentage of the students that will actually failed to pass Alpro in the end semester. When this paper is written, the students are still in the end of the teaching week and have not completed their final exam. Therefore we cannot get their whole marks yet. Because of this we will compare the student performance across academic years by using the mid semester marks to calculate the students' grade.

Based on the data in table 1, we can see that the students' failure in Semester I 2014/2015 decreases significantly to the students' failure in the previous academic years. If the pattern of the students' failure in academic year 2014/2015 is the same as the one in the two previous years, then only 16% (12% + 4%) of the students will fail in the end of semester I 2014/2015. It means that the number of students' failure in this semester decreases 28% to the students' failure in the previous semesters.

The average of the students' performance based on their mid semester marks also increase significantly. The students' average marks in year 2012/2013 and 2013/2014 are 59.62 and 61.06. In year 2014/2015 the average marks increase to 75.29. If we convert this values to grades then the students' average marks have increased from BC to AB. Therefore the students' performance is projected to increase two grade level higher (the grades in the University of Surabaya are: A, AB, B, BC, C, D, and E)

## 5 Conclusion

Improving the student engagement is proven can increase the student performance. However improving the student engagement can be different from case to case and must be elaborated into practical approach and strategies. These approach and strategies must be aligned with the teaching pedagogies, the type of the course, the education culture, and the student background.

This research shows that a holistic approach and strategies to improve the student engagement work well in an introduction to programming course named Alpro, in Informatics Department of the University of Surabaya, in Semester I 2014/2015. As a result of applying the approaches and strategies described in here, the students are willing to participate in the class activity and some of them are even keen to complete their work even though the class is over already. This students' engagement are proven can increase the student performance. Based on the mid semester marks, the students' failure in the end of semester is projected to decline to 16% (12% + 4%). This projected

students' failure is 28% smaller than the students' failure in the two previous academic years. The average students' performance based on their mid semester marks also shows an improvement. The students' performance in Semester I 2014/2015 is projected to increase to two grade level higher than the ones in the previous two academic years.

## References

1. Kinnunen, P., Malmi, L.: Why students drop out CS1 course? In: Proceedings of the Second International Workshop on Computing Education Research, pp. 97–108. ACM (2006)
2. Woods, E.G.: Reducing the dropout rate. School Improvement Research Series (SIRS): Research You Can Use. Northwest Regional Educational Laboratory, Portland (1995)
3. Kushman, J.W., Sieber, C.: Hearnold - Kinney, P.: This isn't the place for me: School dropout. In: Capuzzi, D., Gross, D.R. (eds.) *Youth at Risk: A Prevention Resource for Counselors, Teachers, and Parents*, 3rd edn., pp. 471–507. American Counseling Association, Alexandria (2000)
4. Corney, M., Teague, D., Thomas, R.N.: Engaging students in programming. In: Proceedings of the Twelfth Australasian Conference on Computing Education, vol. 103, pp. 63–72. Australian Computer Society, Inc. (2010)
5. Beaubouef, T., Mason, J.: Why the high attrition rate for computer science students: some thoughts and observations. *ACM SIGCSE Bulletin* 37(2), 103–106 (2005)
6. Mannila, L.: Progress reports and novices' understanding of program code. In: Proceedings of the 6th Baltic Sea Conference on Computing Education Research: Koli Calling 2006, pp. 27–31. ACM (2006)
7. Robins, A., Rountree, J., Rountree, N.: Learning and teaching programming: a review and discussion. *Journal of Computer Science Education* 13(2), 137–172 (2003)
8. Dev, P.C.: Intrinsic motivation and academic achievement: What does their relationship imply for the classroom teacher? *Remedial and Special Education* 18(1), 12–19 (1997)
9. Martin, A.J.: Examining a multidimensional model of student motivation and engagement using a construct validation approach. *British Journal of Educational Psychology* 77(2), 413–440 (2007)
10. Afzal, H., Ali, I., Khan, M.A., Hamid, K.: A study of university students' motivation and its relationship with their academic performance. *International Journal of Business and Management* 5(4), P80 (2010)
11. Kurkovsky, S.: Mobile game development: improving student engagement and motivation in introductory computing courses. *Computer Science Education* 23(2), 138–157 (2013)
12. Hartanto, B., Reye, J.: Incorporating Anchored Learning in a C# Intelligent Tutoring System. In: ICCE 2013, p. 5 (2013)
13. Mohs, R.: How human memory works (2007), <http://science.howstuffworks.com/life/inside-the-mind/human-brain/human-memory2.htm>
14. Mayer, R.E.: The psychology of how novices learn computer programming. *ACM Computing Surveys (CSUR)* 13(1), 121–141 (1981)
15. Robinson, A.: Children learn by example (1998), <http://www.home-school.com/Articles/children-learn-by-example.php>
16. Weragama, D., Reye, J.: Analysing Student Programs in the PHP Intelligent Tutoring System. *International Journal of Artificial Intelligence in Education* 24(2), 162–188 (2014)
17. Schunk, D.H.: Self-efficacy and cognitive skill learning. *Research on Motivation in Education* 3, 13–44 (1989)

18. Eckerdal, A.: Novice programming students' learning of concepts and practise. Ph.D. dissertation, Uppsala University, Division of Scientific Computing, Numerical Analysis (2009)
19. Jacobson, N., Schaefer, S.K.: Pair programming in CS1: overcoming objections to its adoption. *ACM SIGCSE Bulletin* 40(2), 93–96 (2008)
20. McDowell, C., Hanks, B., Werner, L.: Experimenting with pair programming in the classroom. *ACM SIGCSE Bulletin* 35(3), 60–64 (2003)
21. Witt, P.L., Wheelless, L.R.: An experimental study of teachers' verbal and nonverbal immediacy and students' affective and cognitive learning. *Communication Education* 50(4), 327–342 (2001)
22. Beheshti, M., Aló, R.A., Fernandez, J., Gates, A.Q., Ranjan, D., Boadi, A., Barker, L.: Work in progress-CS0 course implementation in Computer Science. In: 38th Annual Frontiers in Education Conference, FIE 2008, pp. F4A-7. IEEE (2008)
23. Taras, M.: Assessment – summative and formative – some theoretical reflections. *British Journal of Educational Studies* 53(4), 466–478 (2005)
24. Taras, M.: Summative and formative assessment Perceptions and realities. *Active Learning in Higher Education* 9(2), 172–192 (2008)
25. Wiliam, D.: Formative assessment: Getting the focus right. *Educational Assessment* 11(3-4), 283–289 (2006)
26. Martin, J.B., Collins, R.A.: Formative and Summative Evaluation in the Assessment of Adult Learning. In: *Assessing and Evaluating Adult Learning in Career and Technical Education*, 127 (2011)
27. Collins, J.W., O'Brien, N.P. (eds.): *The Greenwood dictionary of education*. ABC-CLIO (2011)
28. Bandura, A.: *Self-efficacy: The exercise of self-control*. WH Freeman & Co., Gordonsville (1997)

Rolly Intan  
Chi-Hung Chi  
Henry N. Palit  
Leo W. Santoso (Eds.)

Communications in Computer and Information Science 516

# Intelligence in the Era of Big Data

4th International Conference on Soft Computing,  
Intelligent Systems and Information Technology, ICSIIT 2015  
Bali, Indonesia, March 11–14, 2015, Proceedings

 Springer

**Editorial Board**

Simone Diniz Junqueira Barbosa

*Pontifical Catholic University of Rio de Janeiro (PUC-Rio),  
Rio de Janeiro, Brazil*

Phoebe Chen

*La Trobe University, Melbourne, Australia*

Alfredo Cuzzocrea

*ICAR-CNR and University of Calabria, Cosenza, Italy*

Xiaoyong Du

*Renmin University of China, Beijing, China*

Joaquim Filipe

*Polytechnic Institute of Setúbal, Setúbal, Portugal*

Orhun Kara

*TÜBİTAK BİLGEM and Middle East Technical University, Ankara, Turkey*

Igor Kotenko

*St. Petersburg Institute for Informatics and Automation of the Russian  
Academy of Sciences, St. Petersburg, Russia*

Krishna M. Sivalingam

*Indian Institute of Technology Madras, Chennai, India*

Dominik Ślęzak

*University of Warsaw and Infobright, Warsaw, Poland*

Takashi Washio

*Osaka University, Osaka, Japan*

Xiaokang Yang

*Shanghai Jiao Tong University, Shanghai, China*

More information about this series at <http://www.springer.com/series/7899>



Rolly Intan · Chi-Hung Chi  
Henry N. Palit · Leo W. Santoso (Eds.)

# Intelligence in the Era of Big Data

4th International Conference  
on Soft Computing, Intelligent Systems  
and Information Technology, ICSIIT 2015  
Bali, Indonesia, March 11–14, 2015  
Proceedings

*Editors*

Rolly Intan  
Informatics  
Petra Christian University  
Surabaya  
Indonesia

Henry N. Palit  
Informatics  
Petra Christian University  
Surabaya  
Indonesia

Chi-Hung Chi  
CSIRO  
Hobart  
Tasmania  
Australia

Leo W. Santoso  
Informatics  
Petra Christian University  
Surabaya  
Indonesia

ISSN 1865-0929

ISSN 1865-0937 (electronic)

Communications in Computer and Information Science

ISBN 978-3-662-46741-1

ISBN 978-3-662-46742-8 (eBook)

DOI 10.1007/978-3-662-46742-8

Library of Congress Control Number: 2015934823

Springer Heidelberg New York Dordrecht London

© Springer-Verlag Berlin Heidelberg 2015

This work is subject to copyright. All rights are reserved by the Publisher, whether the whole or part of the material is concerned, specifically the rights of translation, reprinting, reuse of illustrations, recitation, broadcasting, reproduction on microfilms or in any other physical way, and transmission or information storage and retrieval, electronic adaptation, computer software, or by similar or dissimilar methodology now known or hereafter developed.

The use of general descriptive names, registered names, trademarks, service marks, etc. in this publication does not imply, even in the absence of a specific statement, that such names are exempt from the relevant protective laws and regulations and therefore free for general use.

The publisher, the authors and the editors are safe to assume that the advice and information in this book are believed to be true and accurate at the date of publication. Neither the publisher nor the authors or the editors give a warranty, express or implied, with respect to the material contained herein or for any errors or omissions that may have been made.

Printed on acid-free paper

Springer-Verlag GmbH Berlin Heidelberg is part of Springer Science+Business Media  
(www.springer.com)

# Preface

This proceedings volume contains papers presented at the fourth International Conference on Soft Computing, Intelligent System and Information Technology (the 4th ICSIIT) held in Bali, Indonesia, during March 11–14, 2015. The main theme of this international conference is “Intelligence in the Era of Big Data,” and it was organized and hosted by Informatics Engineering Department, Petra Christian University, Surabaya, Indonesia.

The Program Committee received 92 submissions for the conference from across Indonesia and around the world. After peer-review process by at least two reviewers per paper, 53 papers were accepted and included in the proceedings. The papers were divided into 14 groups: fuzzy logic and control system, genetic algorithm and heuristic approaches, artificial intelligence and machine learning, similarity-based models, classification and clustering techniques, intelligent data processing, feature extraction, image recognition, visualization technique, intelligent network, cloud and parallel computing, strategic planning, intelligent applications, and intelligent systems for enterprise government and society.

We would like to thank all Program Committee members for their effort in providing high-quality reviews in a timely manner. We thank all the authors of submitted papers and the authors of selected papers for their collaboration in preparation of the final copy.

Compared to the previous ICSIIT conferences, the number of participants at the 4th ICSIIT 2015 is not only higher, but also the research papers presented at the conference are improved both in quantity and quality. On behalf of the Organizing Committee, once again, we would like to thank all the participants of this conference, who contributed enormously to the success of the conference.

We hope all of you enjoy reading this volume and that you will find it inspiring and stimulating for your research and future work.

February 2015

Rolly Intan  
Chi-Hung Chi  
Henry N. Palit  
Leo W. Santoso



VIII Organization

Can Wang	CSIRO, Australia
Chi-Hung Chi	CSIRO, Australia
Dengwang Li	Shandong Normal University, China
Eduard Babulak	Maharishi University of Management in Fairfield, USA
Enrique Dominguez	University of Malaga, Spain
Erma Suryani	Sepuluh Nopember Institute of Technology, Indonesia
Felix Pasila	Petra Christian University, Indonesia
Hans Dulimarta	Grand Valley State University, USA
Henry N. Palit	Petra Christian University, Indonesia
Hong Xie	Murdoch University, Australia
Ibrahiem M. M. El Emary	King Abdulaziz University, Saudi Arabia
Ilung Pranata	The University of Newcastle, Australia
Julien Dubois	Université de Bourgogne, France
Kassim S. Mwitondi	Sheffield Hallam University, UK
Kelvin Cheng	National University of Singapore, Singapore
Marian S. Stachowicz	University of Minnesota, USA
Masashi Emoto	Meiji University, Japan
Mehmed Kantardzic	University of Louisville, USA
Moeljono Widjaja	Agency for the Assessment and Application of Technology, Indonesia
Mohd Yunus Bin Nayan	Universiti Teknologi Petronas, Malaysia
Muhammad Aamir Cheema	Monash University, Australia
Noboru Takagi	Toyama Prefectural University, Japan
Nur Iriawan	Sepuluh Nopember Institute of Technology, Indonesia
P.S. Avadhani	Andhra University, India
Pitoyo Hartono	Chukyo University, Japan
Pujianto Yugopuspito	Pelita Harapan University, Indonesia
Raymond Kosala	Binus University, Indonesia
Raymond Wong	University of New South Wales, Australia
Roberto Rojas-Cessa	New Jersey Institute of Technology, USA
Rolly Intan	Petra Christian University, Indonesia
Rudy Setiono	National University of Singapore, Singapore
S. Thabasu Kannan	Pannai College of Engineering and Technology, India
Sankar Kumar Pal	Indian Statistical Institute, India
Saurabh K. Garg	University of Tasmania, Australia
Selpi	Chalmers University of Technology, Sweden
Shafiq Alam Burki	University of Auckland, New Zealand
Shan-Ling Pan	University of New South Wales, Australia
Simon Fong	University of Macau, Macau
Smarajit Bose	Indian Statistical Institute, India

Son Kuswadi	Electronic Engineering Polytechnic Institute of Surabaya, Indonesia
Suash Deb	CV Raman College of Engineering, India
Suphamit Chittayasothorn	King Mongkut's Institute of Technology Ladkrabang, Thailand
Taweesak Kijkanjanarat	Thammasat University, Thailand
Vatcharaporn Esichaikul	Asian Institute of Technology, Thailand
Vincent Vajnovszki	Université de Bourgogne, France
Wen-June Wang	National Central University, Taiwan
Wichian Chutimaskul	King Mongkut's University of Technology Thonburi, Thailand
Xiaojun Ye	Tsinghua University, China
Yung-Chen Hung	Soochow University, Taiwan
Yunwei Zhao	Tsinghua University, China

# Contents

## Invited Paper

- On the Relation of Probability, Fuzziness, Rough and Evidence  
Theory ..... 3  
*Rolly Intan*

## Fuzzy Logic and Control System

- A Study of Laundry Tidiness: Laundry State Determination Using  
Video and 3D Sensors ..... 19  
*Daiki Hirose, Tsutomu Miyoshi, and Kazuki Maiya*

- Direction Control System on a Carrier Robot Using Fuzzy Logic  
Controller ..... 27  
*Kevin Ananta Kurniawan, Darmawan Utomo, and Saptadi Nugroho*

- Multidimensional Fuzzy Association Rules for Developing Decision  
Support System at Petra Christian University ..... 37  
*Yulia, Siget Wibisono, and Rolly Intan*

## Genetic Algorithm and Heuristic Approaches

- Genetic Algorithm for Scheduling Courses ..... 51  
*Gregorius Satia Budhi, Kartika Gunadi,  
and Denny Alexander Wibowo*

- Optimization of Auto Equip Function in Role-Playing Game Based on  
Standard Deviation of Character's Stats Using Genetic Algorithm ..... 64  
*Kristo Radion Purba*

- The Design of Net Energy Balance Optimization Model for Crude Palm  
Oil Production ..... 76  
*Jaizuluddin Mahmud, Marimin, Erliza Hambali, Yandra Arkeman,  
and Agus R. Hoetman*

- ACO-LS Algorithm for Solving No-wait Flow Shop Scheduling  
Problem ..... 89  
*Ong Andre Wahyu Riyanto and Budi Santosa*

- A New Ant-Based Approach for Optimal Service Selection with E2E  
QoS Constraints ..... 98  
*Dac-Nhuong Le and Gia Nhu Nguyen*



## Artificial Intelligence and Machine Learning

Implementation Discrete Cosine Transform and Radial Basis Function Neural Network in Facial Image Recognition . . . . .	113
<i>Marprin H. Muchri, Samuel Lukas, and David Habsara Hareva</i>	

Implementation of Artificial Intelligence with 3 Different Characters of AI Player on “Monopoly Deal” Computer Game . . . . .	119
<i>Irene A. Lazarusli, Samuel Lukas, and Patrick Widjaja</i>	

Optimizing Instruction for Learning Computer Programming – A Novel Approach . . . . .	128
<i>Muhammed Yousoof and Mohd Sapiyan</i>	

Sequential Pattern Mining Application to Support Customer Care “X” Clinic . . . . .	140
<i>Alexander Setiawan, Adi Wibowo, and Samuel Kurniawan</i>	

## Similarity-Based Models

The Comparation of Distance-Based Similarity Measure to Detection of Plagiarism in Indonesian Text . . . . .	155
<i>Tari Mardiana, Teguh Bharata Adji, and Indriana Hidayah</i>	

Document Searching Engine Using Term Similarity Vector Space Model on English and Indonesian Document . . . . .	165
<i>Andreas Handojo, Adi Wibowo, and Yovita Ria</i>	

Knowledge Representation for Image Feature Extraction . . . . .	174
<i>Nyoman Karna, Iping Suwardi, and Nur Maulidevi</i>	

Using Semantic Similarity for Identifying Relevant Page Numbers for Indexed Term of Textual Book . . . . .	183
<i>Daniel Siahaan and Sherly Christina</i>	

## Classification and Clustering Techniques

The Data Analysis of Stock Market Using a Frequency Integrated Spherical Hidden Markov Self Organizing Map . . . . .	195
<i>Gen Niina, Tatsuya Chuuto, Hiroshi Dozono, and Kazuhiro Muramatsu</i>	

Attribute Selection Based on Information Gain for Automatic Grouping Student System . . . . .	205
<i>Oktariani Nurul Pratiwi, Budi Rahardjo, and Suhono Harso Supangkat</i>	

Data Clustering through Particle Swarm Optimization Driven  
 Self-Organizing Maps ..... 212  
*Tad Gonsalves and Yasuaki Nishimoto*

**Intelligent Data Processing**

A Search Engine Development Utilizing Unsupervised Learning  
 Approach ..... 223  
*Mohd Noah Abdul Rahman, Afzaal H. Seyal, Mohd Saiful Omar,  
 and Siti Aminah Maidin*

Handling Uncertainty in Ontology Construction Based on Bayesian  
 Approaches: A Comparative Study ..... 234  
*Foni Agus Setiawan, Wahyu Catur Wibowo, and Novita Br Ginting*

Applicability of Cyclomatic Complexity on WSDL ..... 247  
*Sanjay Misra, Luis Fernandez-Sanz, Adewole Adewumi,  
 Broderick Crawford, and Ricardo Soto*

**Feature Extraction**

Multiclass Fruit Classification of RGB-D Images Using Color and  
 Texture Feature ..... 257  
*Ema Rachmawati, Iping Supriana, and Masayu Leylia Khodra*

Content-Based Image Retrieval Using Features in Spatial and  
 Frequency Domains ..... 269  
*Kazuhiro Kobayashi and Qiu Chen*

Feature Extraction for Java Character Recognition ..... 278  
*Rudy Adipranata, Liliana, Meiliana Indrawijaya,  
 and Gregorius Satia Budhi*

Fast Performance Indonesian Automated License Plate Recognition  
 Algorithm Using Interconnected Image Segmentation ..... 289  
*Samuel Mahatmaputra Tedjojuwono*

**Image Recognition**

A Study of Laundry Tidiness: Socks Pairing Using Video and  
 3D Sensors ..... 303  
*Kazuki Maïya, Tsutomu Miyoshi, and Daiki Hirose*

Design and Implementation of Skeletonization ..... 314  
*Kartika Gunadi, Liliana, and Gideon Simon*

A Computer-Aided Diagnosis System for Vitiligo Assessment: A Segmentation Algorithm .....	323
<i>Arfika Nurhudatiana</i>	
Face Recognition for Additional Security at Parking Place .....	332
<i>Semuil Tjiharjadi and William Setiadarma</i>	
Optic Disc Segmentation Based on Red Channel Retinal Fundus Images .....	348
<i>K.Z. Widhia Oktoeberza, Hanung Adi Nugroho, and Teguh Bharata Adji</i>	
<b>Visualization Techniques</b>	
Multimedia Design for Learning Media of Majapahit .....	363
<i>Silvia Rostianingsih, Michael Chang, and Liliana</i>	
Adding a Transparent Object on Image .....	372
<i>Liliana, Meliana Luwuk, and Djoni Haryadi Setiabudi</i>	
3D-Building Reconstruction Approach Using Semi-global Matching Classified .....	382
<i>Iqbal Rahmadhian Pamungkas and Iping Supriana Swardi</i>	
<b>Intelligent Network</b>	
Spanning Tree Protocol Simulation Based on Software Defined Network Using Mininet Emulator .....	395
<i>Indrarini Dyah Irawati and Mohammad Nuruzzamanirridha</i>	
Varnish Web Cache Application Evaluation .....	404
<i>Justinus Andjarwirawan, Ibnu Gunawan, and Eko Bayu Kusumo</i>	
DACK-XOR: An Opportunistic Network Coding Scheme to Address Intra-flow Contention over Ad Hoc Networks .....	411
<i>Radha Ranganathan, Kathiravan Kannan, P. Aarthi, and S. LakshmiPriya</i>	
Network Security Situation Prediction: A Review and Discussion .....	424
<i>Yu-Beng Leau and Selvakumar Manickam</i>	
<b>Cloud and Parallel Computing</b>	
Lightweight Virtualization in Cloud Computing for Research .....	439
<i>Muhamad Fitra Kacamarga, Bens Pardamean, and Hari Wijaya</i>	
A Cloud-Based Retail Management System .....	446
<i>Adewole Adewumi, Stanley Ogbuchi, and Sanjay Misra</i>	

Towards a Cloud-Based Data Storage Medium for E-learning Systems  
in Developing Countries ..... 457  
*Temitope Olokunde and Sanjay Misra*

Fast and Efficient Parallel Computations Using a Cluster of  
Workstations to Simulate Flood Flows ..... 469  
*Sudi Mungkasi and J.B. Budi Darmawan*

**Strategic Planning**

A Simulation Model for Strategic Planning in Asset Management of  
Electricity Distribution Network ..... 481  
*Erma Suryani, Rully Agus Hendrawan, Eka Adipraja Philip Faster,  
and Lily Puspa Dewi*

Enhancing the Student Engagement in an Introductory Programming:  
A Holistic Approach in Improving the Student Grade in the Informatics  
Department of the University of Surabaya ..... 493  
*Budi Hartanto*

Business Process Maturity at Agricultural Commodities Company ..... 505  
*Lily Puspa Dewi, Adi Wibowo, and Andre Leander*

Innovation Strategy Services Delivery: An Empirical Case Study of  
Academic Information Systems in Higher Education Institution ..... 514  
*John Tampil Purba and Rorim Panday*

**Intelligent Applications**

Public Transport Information System Using Android ..... 529  
*Agustinus Noertjahyana, Gregorius Satia Budhi,  
and Agustinus Darmawan Andilolo*

Lecturers and Students Technology Readiness in Implementing Services  
Delivery of Academic Information System in Higher Education  
Institution: A Case Study ..... 539  
*Rorim Panday and John Tampil Purba*

Tool Support for Cascading Style Sheets' Complexity Metrics ..... 551  
*Adevole Adewumi, Onyeka Emebo, Sanjay Misra,  
and Luis Fernandez*

**Intelligent Systems for Enterprise, Government and Society**

Generic Quantitative Assessment Model for Enterprise Resource  
Planning (ERP) System ..... 563  
*Olivia and Kridanto Surendro*

XXIV Contents

The Implementation of Customer Relationship Management: Case Study from the Indonesia Retail Industry .....	572
<i>Leo Willyanto Santoso, Yusak Kurniawan, and Ibnu Gunawan</i>	
The Implementation of Customer Relationship Management and Its Impact on Customer Satisfaction, Case Study on General Trading and Contractor Company .....	579
<i>Djoni Haryadi Setiabudi, Vennytha Lengkong, and Silvia Rostianingsih</i>	
Towards e-Healthcare Deployment in Nigeria: The Open Issues .....	588
<i>Jumoke Soyemi, Sanjay Misra, and Omoregbe Nicholas</i>	
<b>Author Index</b> .....	601



<http://www.springer.com/978-3-662-46741-1>

Intelligence in the Era of Big Data  
4th International Conference on Soft Computing,  
Intelligent Systems, and Information Technology, ICSIT  
2015, Bali, Indonesia, March 11-14, 2015. Proceedings  
Intan, R.; Chi, C.-H.; Palit, H.N.; Santoso, L.W. (Eds.)  
2015, XXIV, 602 p. 334 illus., Softcover  
ISBN: 978-3-662-46741-1



# Author details

## Hartanto, Budi

[View potential author matches](#)

<http://orcid.org/0000-0001-8408-2311>

Affiliation(s):

Universitas Surabaya, Surabaya, Indonesia [View more](#)

Subject area: [Computer Science](#) [Social Sciences](#) [Engineering](#) [Materials Science](#) [Mathematics](#)

Documents by author

4

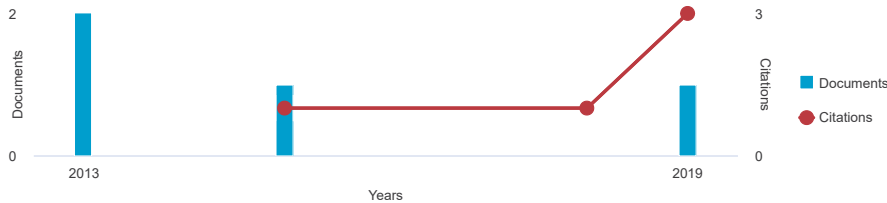
Total citations

5 by 4 documents

*h*-index:

2

Document and citation trends:



## Profile actions

- [Edit author profile](#)
- [Connect to ORCID](#)
- Alerts
  - [Set citation alert](#)
  - [Set document alert](#)
- [Learn more about S Profiles](#)

**BH** **Budi Hartan**  
 4 Documents  
[View Mendeley pr](#)

[4 Documents](#) [Cited by 4 documents](#) [1 co-author](#) [Topics](#)

Preview users can view an author's latest 10 documents. [View 50 references](#) >

[Set document alert](#)

Document title	Authors	Year	Source
Anchored instruction ITS: A novel approach to make learning programming interesting and effective <a href="#">Open Access</a> <a href="#">View abstract</a> <a href="#">Related documents</a>	Hartanto, B., Reye, J.	2019	IOP Conference Series: Materials Science and Engineering
Enhancing the student engagement in an introductory programming: A holistic approach in improving the student grade in the informatics department of the University of Surabaya <a href="#">View abstract</a> <a href="#">Related documents</a>	Hartanto, B.	2015	Communications in Computer and Information Science
Incorporating anchored learning in a C# Intelligent Tutoring System <a href="#">View abstract</a> <a href="#">Related documents</a>	Hartanto, B., Reye, J.	2013	Doctoral Student Consortia - Proceedings of the 21st International Conference on Computers in Education, ICCE 2013
Code analyser in CSTutor - A C# Intelligent Tutoring System <a href="#">View abstract</a> <a href="#">Related documents</a>	Hartanto, B., Reye, J.	2013	Proceedings of the 21st International Conference Computers in Education, ICCE 2013



Preview users can view an author's latest 10 documents.



The data displayed above is compiled exclusively from documents indexed in the Scopus database. To request corrections to any inaccuracies or provide any further feedback, please use the [Author Feedback Wizard](#).

## About Scopus

[What is Scopus](#)  
[Content coverage](#)  
[Scopus blog](#)  
[Scopus API](#)  
[Privacy matters](#)

## Language

[日本語に切り替える](#)  
[切换到简体中文](#)  
[切换到繁體中文](#)  
[Русский язык](#)

## Customer Service

[Help](#)  
[Contact us](#)

---

## ELSEVIER

[Terms and conditions ↗](#) [Privacy policy ↗](#)

Copyright © Elsevier B.V. All rights reserved. Scopus® is a registered trademark of Elsevier B.V.

We use cookies to help provide and enhance our service and tailor content. By continuing, you agree to the use of cookies.