Introducing MATLAB to Electronic Engineering Undergraduates through Three Weeks Laboratory Sessions

http://dx.doi.org/10.3991/ijoe.v10i2.3170

Haidi Ibrahim and Theam Foo Ng
Universiti Sains Malaysia, Penang, Malaysia.

***Abstract*—Previously, our undergraduates in electronic engineering program had to explore and learn MATLAB from the beginning by their own, without direct guidance. Based on this, we decided to formally introduce MATLAB to them as a part of the syllabus in our Advance Laboratory course, in order to continuously improve our electronic engineering undergraduates. As we want to track the significance of the course, two surveys have been given to the students. One survey has been executed at the beginning of the laboratory, and another one has been carried out at the end of the laboratory session. The outcomes from these two surveys show that the designed syllabus successfully increases both skill and confident level of our students in solving complex engineering problem using MATLAB programming.**

***Index Terms*—Computer science education, electronics engineering education, MATLAB, programming.**

I. INTRODUCTION

MATLAB has becoming one of the major computer programming languages used in various engineering fields these days [1]. To be more specific, in electronic engineering field, MATLAB together with its appropriate supporting toolboxes has been used in aiding engineers to solve complex engineering problems. MATLAB has been used as a tool in digital signal processing [2], digital image processing [3], artificial intelligent [4], control and robotic [5], pattern recognition [6], data visualization, simulations, and statistics. Furthermore, the user can use external libraries with MATLAB, which as a consequence makes the usage of this high level programming language broader [7]. As there are many benefits that can be obtained from the usage of MATLAB, in our opinion, it is much better if our students have a formal introduction to MATLAB programming.

Academic year	1		2		3		4	
Semester	1	2	1	2	1	2	1	2
Subject	EEE 123			EEE 226	EEE 320			

Figure 1. The courses related to computer programming

Our undergraduate electronic engineering program is structured into a four years program, with two semesters per academic session. Following the common formation for electronic engineering program, similar to other higher learning institutions in this world, we also provide our student with computer programming knowledge. The components of computer programming lessons in our program structure are shown in Fig. 1.

Our students have been taught computer programming during their first year, first semester, through a course known as EEE123: Programming for Engineers. This course trains our students the basic skills that are required in computer programming. This computer programming course concentrates on the usage of C++ programming language. The syllabus includes declaring variables, using control flow, creating functions and classes, and storing data in files.

In second semester of their second year, our students took EEE226: Microprocessor 1. Various aspects of microprocessor are covered by this course. In order for our students to program the microprocessor, assembly language has been taught to our students. As a continuation to this course, our students also use assembly language in their third year, first semester, for EEE320: Microprocessor 2.

With these three courses, we believe that our students are already competent in computer programming. However, in order to make our students to be more confident and competent in computer programming, we are actually still thinking that a formal introductory to MATLAB programming is required to benefit our students, especially for solving engineering problem [8]. By introducing MATLAB to our students, we hope that our students will have a broader knowledge in computer programming.

This paper is actually an extended and updated version of our conference paper, presented in 2013 IEEE International Conference on Teaching, Assessment and Learning for Engineering (TALE), which was held in Bali Dynasty Resort, Kuta, Bali, Indonesia, on 26-29 August 2013 [9]. This paper is divided into six sections. Section II gives the course objectives of our Advance Laboratory. Course contents of this MATLAB introductory laboratory are presented in Section III. Evaluation of the course is briefly presented in Section IV. We carried two surveys during this course, and the outcome from these surveys is presented in Section V. Section VI gives the conclusion from this case study.

II. COURSE OBJECTIVES

Based on a few informal feedbacks, we found out that some of our undergraduates have used MATLAB for their assignments. Unfortunately, as MATLAB is usually being considered as a high level programming language, and easy to be learnt, previously, our students have to explore this programming language by their own, without a proper guidance. As a consequence, although our students completed their assignments successfully, they still feel not very confident with their current MATLAB programming skill.

The computer programming skills might help our students to accomplish their assigned tasks during their industrial training, which normally taken place in semester break, between their third and fourth year. In addition to industrial training, some of our students will probably use MATLAB for their research and development in their final year project. MATLAB also could help our students to have a better understanding in their courses during their fourth year.

Therefore, we have decided to incorporate MATLAB in our course, EEE351: Advance Laboratory, which is a course offered in the second semester, for third year students. This is a fully coursework based course, and have been divided into four portions, accordingly to its course objectives (COs). One of the COs is to be able to solve complex engineering task by the aid of computer programming. Because one semester has 14 weeks, three weeks have been allocated to fulfill this CO (i.e., 14 weeks ÷ 4 COs ≈ 3 weeks per CO).

The objectives of this MATLAB introductory laboratory are:

1. To familiarize students with MATLAB environment.
2. To increase the interest of the students towards computer programming.
3. To increase the confident of our students in computer programming.

The first objective was been set as we assume that most of our students do not have any experience on using MATLAB before.

III. COURSE CONTENTS

During this three weeks course, the course had been divided into three sections. Each section occupies one week and requires two laboratory sessions. Each computer laboratory session is a three hours session, and therefore, at least eighteen hours are needed to completely cover this introductory course. The content and objectives of each section are explained in the following subsections:

A. Week 1: Relating MATLAB with C++

Due to limited time allocated for this MATLAB introductory laboratory, and we treat our students as adult learners, it is better to allow our students to relate MATLAB, which is a new programming language for them, with the programming language that they are already familiar with, which is C++. Therefore, the contents for this first week were designed so that they look similar to the syntaxes of C++ programming language. The difference between MATLAB and C++ code have been emphasized, such as the use of '{' and '}' in C++ as compared to "end" in MATLAB.

As most of the students do not have experience on using MATLAB before, steps on using MATLAB integrated development environment (IDE) have been shown in details by the aid of figures. The students have been guided on how to execute the codes in MATLAB, to force break a process, to use command window, and to use M-file. Figures and examples have been provided in the student manual, to further assist the students.

This laboratory section introduces students with variables in MATLAB. Similar to C++, we introduce the students with double and integer variables, so that the student can appreciate the code in order to reduce the program size, or to obtain a more precise calculation result. We also introduce students with character and string manipulation. Students also learn logical variable in MATLAB. To fully utilize the advantage of MATLAB as compared to C++, we introduce students with matrices, arrays, vectors, and complex numbers.

In the module also, we introduce the students to matrix operations and array operations in MATLAB. The differences between them have been highlighted. Furthermore, we also teach students regarding to logical expressions, and binary operations. In order to help the students to relate MATLAB with C++, control flow such as for loop, while, and if-else, have been included into the syllabus. We also teach our students on how to display the results to the computer monitor, and to get the input from the user, dynamically, during the running time using the keyboard. The students also learn how to safe their output in files in several formats, including Microsoft Excel file.

Examples for the abovementioned contents have been given in the manual. The students are required to try the examples and solve five computer programming exercises. The exercises involve in solving circuit analysis, creating multi-dimensional array, estimate the derivative for a function, creating a calendar in Microsoft Excel file, and determine the day of birth. These exercises require the students to develop and find the relevant mathematical equations, and implement these equations as MATLAB codes. The codes and corresponding outputs, together with a short report, were submitted at the end of the week 1.

B. Week 2: Presenting and Visualizing Data

One of the popular MATLAB usages is to plot the data. Therefore, the content for this week was specifically deals with this matter. The students have been introduced with plotting continuous data, and also plotting discrete data. The students are also been taught on how to plot multiple functions on the same window.

The students have been taught on how to label the plot, and how to plot multiple functions. Several examples have been given to students, and they need to solve five exercises. These exercises are mostly related to digital signal processing, and involve in modulation, sampling, finding the median and quartiles values, Fourier transform, and convolution. Students need to submit their codes and corresponding plots, together with a short report, at the end of the week 2.

C. Week 3: Mini Project

During this last week, we gave our students one computer programming problem in order to challenge them and strengthen their understanding in MATLAB. In order to fit to a broad background of our students, the problem need to be solved is a general task. In our case, we asked our student to solve a Sudoku puzzle using MATLAB.

The Sudoku that need to be solved is not so hard and complex. This is shown in Fig. 2. As shown in this figure, only 35 from 81 cells are unknown and need to be found. Because this is an easy Sudoku, we expect the students can solve this Sudoku without much trouble by their own, and thus can design the flow of the program appropriately. Therefore, the students can compare the outputs from MATLAB with their calculated values, and this can be use to verify their codes.

At the end of the week 3, the students need to present their work through a short *viva-voce* session. Each group also needs to submit the code and provide a short report for the evaluation.

IV. EVALUATION OF THE COURSE

This laboratory session was handled by one lecturer, and three tutors. It was held on March and April 2013. There are 51 students in total, with age between 22 to 26 years old. The distribution of our students, based on age is tabulated in Table I.

Our students come from different races. In order to encourage the students to work together and help each other, we divided them into 25 groups, where each group has two members (except one group has three members). The combinations of the group is presented in Fig. 3.

During the laboratory sessions, we encouraged the students to ask questions if they have any problem with their tasks. We did not provide them the solution, but guided them to solve the problem successfully. We also allowed the students to do discussions between groups.

The performances of the students were evaluated based on the submitted materials (i.e., codes, outputs, and short reports). The performances were also further evaluated during the short presentation sessions. Some of the aspects that we have considered during the evaluation are:

1. Do the codes produce the expected outcomes?
2. Are the code syntaxes written correctly?
3. Are the codes written neatly (e.g., based on the name of variables or comments)?

In order to initiate the learning process to the students, we provided all the modules (i.e. for three weeks) at the beginning of the week one. Therefore, the students can do their revision accordingly, based on their own time.

Furthermore, the students also have given opportunity to do self-assessment. In this laboratory session, each student is required to judge their own performance, their

5	3			7		9		
6			1	9	5	3	4	8
	9	8				1	6	7
8	5	9	7	6				3
4	2	6	8		3			1
7	1	3		2				6
	6					2	8	
	8		4	1	9			5
3		5		8			7	9

Figure 2. The Sudoku given as the mini project

TABLE I.
DISTRIBUTION OF STUDENTS BASED ON AGE

Age	Number of Students
22 years old	33 students
23 years old	17 students
26 years old	1 student

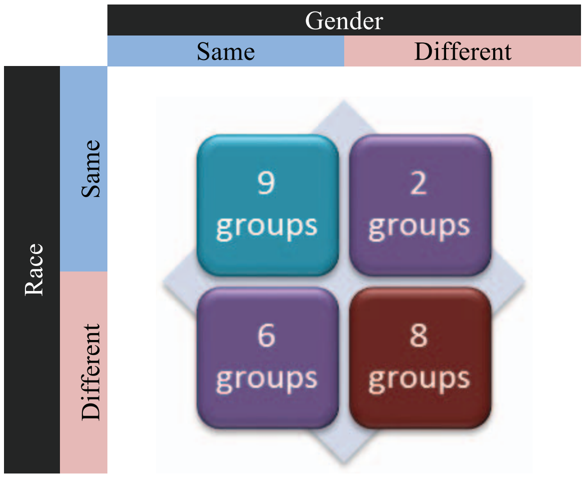

Figure 3. The combinations of EEE351 laboratory groups.

group performance, and also their partner's performance. The forms for this self-assessments were given during week one, and submitted at the end of week three. Guided by these forms, the students are able to know which aspect that are important in this laboratory sessions.

V. OUTCOME FROM THE SURVEYS

Two set of the surveys have been given to the students. One survey was submitted at the beginning of week one, while another survey was submitted at the end of week three. The questions for Survey 1 are:

1. Do you own a computer? (Yes/No).
2. What grade did you obtained for EEE123: Computer Programming for Engineers? Is it grade B- or above? (Yes/No)
3. Do you like computer programming? (Yes/No).
4. Are you confident to do the computer coding? (Yes/No)
5. Do you think computer programming is fun? (Yes/No).
6. Do you think computer programming is useful for an electronic engineer? (Yes/No)
7. Have you use MATLAB in any course before? (Yes/No).
8. Are you proficient in MATLAB programming? (Yes/No).
9. Are you interested to learn MATLAB? (Yes/No)
10. Can you solve complex engineering problem by computer programming? (Yes/No)

The questions for Survey 2 are:

11. Do you like computer programming? (Yes/No)
12. Are you confident to do the computer coding? (Yes/No).
13. Do you think computer programming is fun? (Yes/No)
14. Do you think computer programming is useful for an electronic engineer? (Yes/No)
15. Are you proficient in MATLAB programming? (Yes/No)
16. Do you think programming in MATLAB is interesting? (Yes/No)
17. Can you solve complex engineering problem by computer programming? (Yes/No)

18. Do you think this syllabus is relevant to this course? (Yes/No)
19. Do you think this syllabus is useful for you in future? (Yes/No)

We use Survey 1 to indicate the computer programming background of our student. Survey 2, which has six same questions with Survey 1, will be used in order to track the benefits from our course. All students have responded to Survey 1, and therefore we have 51 responses. However, two students did not return their form for Survey 2. Thus, we only have 49 responses for Survey 2.

A. *Outcomes from Survey 1*

The results from Survey 1 are shown as bar graph in Fig. 4. From this figure, for question 1 (Q1), all respondents admit that they own computers. Therefore, we can assume that all the students are familiar with computer, which hopefully will make the computer programming lessons easier for them.

From Fig. 4, question 2 (Q2) shows that 30 from 51 of our students have obtained grade B- or above for their EEE123 course that they took during their first year. Therefore, we can assume that more than 50% of our students have good computer programming background.

Question 3 (Q3) from Survey 1 asks the student whether they are interested in computer programming or not. As shown in Fig. 4, almost 50% of the students, which is 23 out of 50 students, show their interest in computer programming. If we inspect the relationship between Q2 with Q3, as presented in Fig. 5, we can see that the students interest towards computer programming might be contributed by the grade they obtained from EEE123.

Interestingly, based on question 4 (Q4), we can observe that not many students are confident to do their own code. It is only 37.25% of the students are confident to do their own computer coding at the beginning of our laboratory sessions. Fig. 6 shows the relationship between Q2 with Q4. As shown in this figure, even the students have good grade for their EEE123, some of them actually still do not have confident to do their own code. Fig. 7 shows the relationship between Q3 with Q4. As shown in this figure, almost half of the students who are interested in computer programming still do not have confident to do their own code.

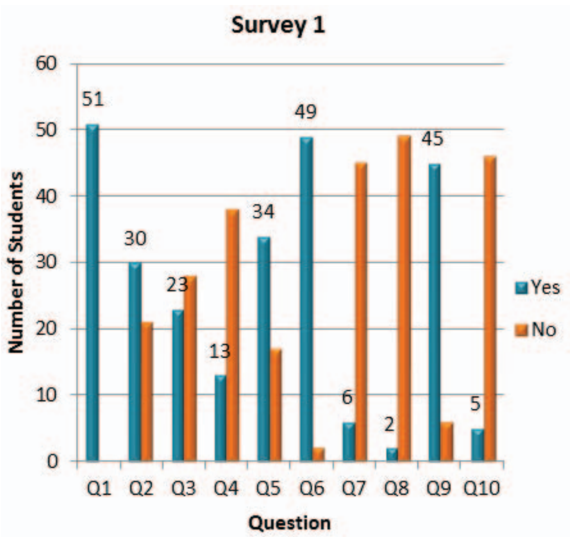

Figure 4. Responses from Survey 1

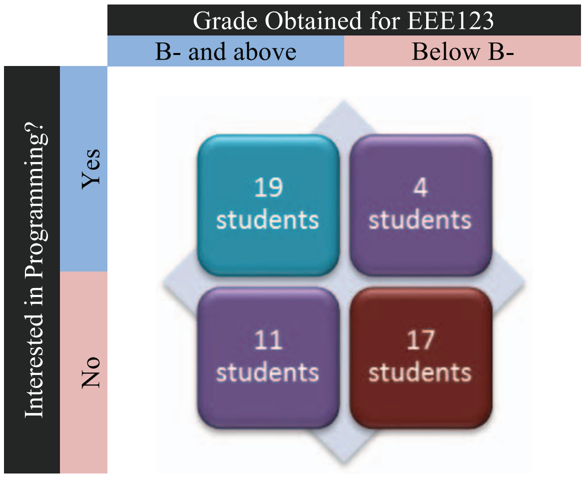

Figure 5. Relationship between student's interest in programming (Q3) with the grade obtained for EEE123 (Q2)

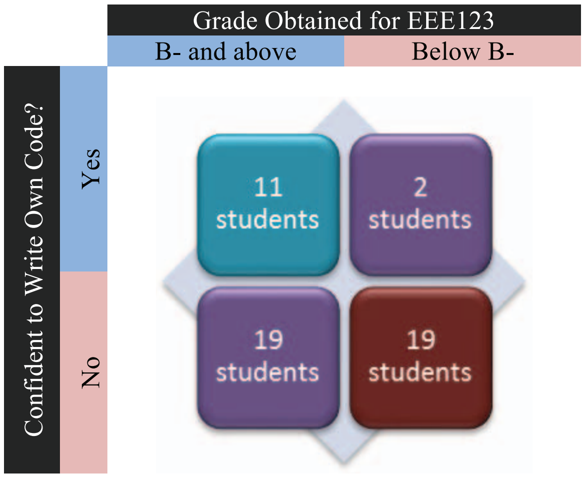

Figure 6. Relationship between student's confidence in computer programming (Q4) with the grade obtained for EEE123 (Q2)

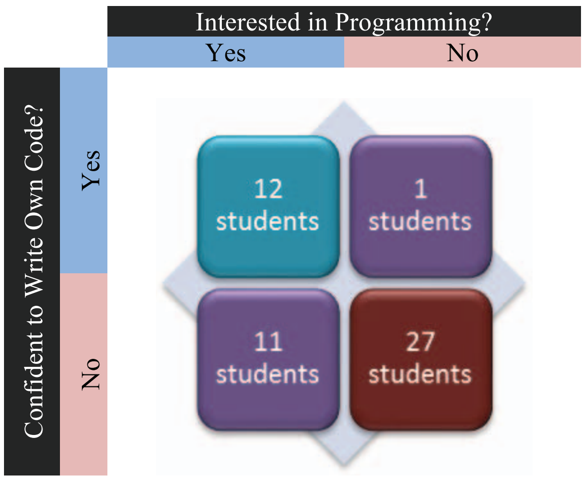

Figure 7. Relationship between student's confidence in computer programming (Q4) with their interest level (Q3)

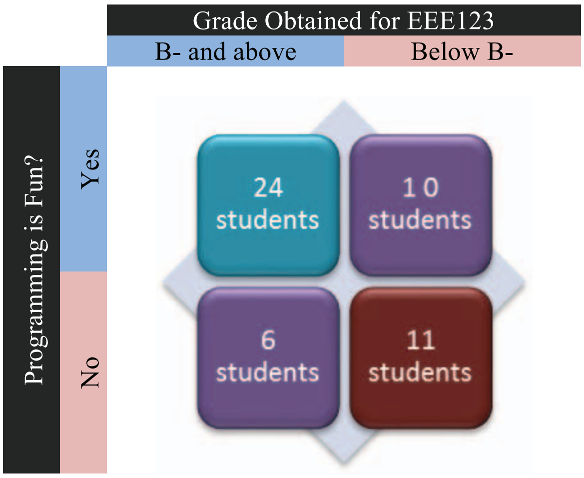

Figure 8. Relationship between student's grade in EEE123 (Q2) with their opinion about the fun of computer programming (Q5)

As shown in Fig. 4 and Fig. 8, the responses to question 5 (Q5) is almost similar to Q2, where most of the students think that computer programming is fun. Besides, from the result of question 6 (Q6), we can see that almost all the students admit that computer programming is important for electronic engineers.

Therefore, although most of the students are not exposed to MATLAB before (indicated by questions 7 (Q7) and 8 (Q8)), they are interested to learn MATLAB. This can be observed from question 9 (Q9) in Fig. 4. Yet, similar to Q4, the outcomes from question 10 (Q10) indicate that almost all of the students are not confident to solve complex engineering problem by using computer programming.

B. *Outcomes from Survey 2*

The results from Survey 2 are summarized by the bar graph shown in Fig. 9. In this survey, questions 1, 2, 3, 4, 5, and 7 (i.e. Q1, Q2, Q3, Q4, Q5, and Q7) are repeated from Survey 1. The questions are designed in this form in order to track the improvements obtained from this course.

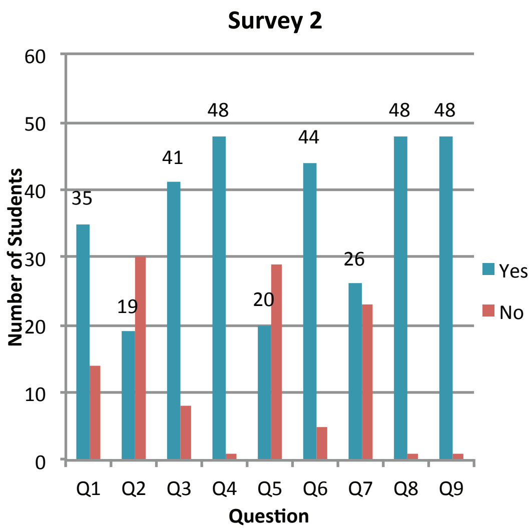

Figure 9. Responses from Survey 2

For Q6 in Survey 2, we can see that majority of the students, which is 44 out of 49 students (89.80%) think that MATLAB programming is interesting. This is tally with Q9 in Survey 1, where 88.24% of students are interested to learn MATLAB. Furthermore, from Q8 and Q9 of Survey 2, most of the students, which is 97.96% from 49 students, think that this MATLAB introductory laboratory is relevant to this course, i.e., EEE351, and might be useful for them in future.

When we compare Q1 from Survey 2, with Q3 from survey 1, we can see that there is a significant increment in the number of students who like computer programming. Similarly, when we compare Q2 in Fig. 9 with Q4 in Fig. 4, more students are becoming confident to do their own code. However, the number of students who not confident is still relatively higher.

Similarly, we can see improvements (in terms of percentage) when we compare Q1, Q2, Q3, Q4, Q5, and Q7, in Survey 2, with Q3, Q4, Q5, Q6, Q8 and Q10 in Survey 1, respectively. This is tabulated in Table II. Therefore we can say that the introductory laboratory successfully improve the confident level and programming skills of our students. Furthermore, most of the students find MATLAB programming is interesting, and this course is useful and relevant.

TABLE II.
INCREMENT OF 'YES' ANSWER IN SURVEY 2 FOR THE RESPECTIVE QUESTIONS AS COMPARED TO SURVEY 1

Survey 1, S1	(Q3) 45.1%	(Q4) 25.5%	(Q5) 66.7%	(Q6) 96.1%	(Q8) 3.9%	(Q10) 9.8%
Survey 2, S2	(Q1) 71.4%	(Q2) 38.8%	(Q3) 83.7%	(Q4) 98.0%	(Q5) 40.8%	(Q7) 53.1%
Increment, S2-S1	26.3%	13.3%	17.0%	1.9%	36.9%	43.3%
Increase?	Yes	Yes	Yes	Yes	Yes	Yes

VI. CONCLUSION

A MATLAB introductory laboratory has been executed successfully. Although it is a three weeks laboratory, the results from two surveys show that the content of this laboratory sessions successfully increase students' programming skill and they becoming more confident to solve complex engineering problem using computer programming.

ACKNOWLEDGMENT

Thanks to EEE351 students for academic session 2012/2013 for giving their feedback for these two surveys. Thanks to the anonymous reviewers for their constructive comments. Thanks also for the attendees of 2013 IEEE International Conference on Teaching, Assessment and Learning for Engineering (TALE 2013) who give their constructive comments and suggestions towards this work. Thanks also to Dr. Norlaili Mohd Noh for her suggestions and her helps to proofread this article.

REFERENCES

[1] A. Haubold, "Matlab for first-year college engineers," 37th ASEE/IEEE Frontiers in Education Conference, pp. F1H-7-F1H-12, 2007.

[2] Z. Yuxi, K. Li, W. Jun, S. Jinping, and W. Zulin, "Methods and experience of using Matlab and FPGA for teaching practice in dig-

ital signal processing," International Conference on Education and Management Technology (ICEMT), pp. 414-417, 2010.

[3] Y. G. Wang, and A. Li, "On teaching image processing course," 2011 International Symposium on IT in Medicine and Education (ITME), vol. 2, pp. 560-562, 2011.

[4] F. Margrave, N. R. Babu, A. Bradshaw, and I. Collins, "MATLAB-neural networks toolbox hardware post-processor," IEEE Colloquim on Applied Control Techniques Using MATLAB, pp. 6/1-6/3, 1995.

[5] H. Suining, and L. Wei, "Advanced robotic digital actuators controlling and communication methods based on MATLAB," 2011 International Conference on Electric Information and Control Engineering (ICEICE), pp. 1138-1140, 2011. http://dx.doi.org/10.1109/ICEICE.2011.5777759

[6] S. Theodonidis, A. Pikrakis, K. Koutroumbas, and D. Cavouras, "Introduction to Pattern Recognition: A MATLAB Approach," Academic Press, United States, 2010.

[7] J. Nehrbass, S. Samsi, J. C. Chaves, J. Unpingco, B. Guilfoos, S. Ahalt, A. Krishnamurthy, A. Chalker, and J. Gardiner, "Interfacing PC-based MATLAB directly to HPC resources," HPCMP Users Group Conference 2006, pp. 440-444, June 2006.

[8] D. M. Smith, "Engineering Computation with MATLAB," 3rd edition, Pearson Education, New York, United States, 2012.

[9] H. Ibrahim, "Short MATLAB introductory laboratory for electronic engineering undergraduate students: A three weeks hand-on syllabus: A case study," 2013 IEEE International Conference on Teaching, Assessment and Learning for Engineering (TALE 2013), pp. 400-403, August 2013. http://dx.doi.org/10.1109/TALE.2013.6654470

AUTHORS

Haidi Ibrahim is with the School of Electrical & Electronic Engineering, Engineering Campus, Universiti Sains Malaysia, 14300 Nibong Tebal, Penang, Malaysia (e-mail: haidi@eng.usm.my).

Theam Foo Ng, was with the School of Electrical & Electronic Engineering, Engineering Campus, Universiti Sains Malaysia, 14300 Nibong Tebal, Penang, Malaysia. He is now with Centre for Global Sustainability Studies, 5th Floor, Hamzah Sendut Library, Universiti Sains Malaysia, 11800 Penang, Malaysia. (e-mail: tfng@usm.my, theam.ng@gmail.com).

This article is an extended and modified version of a paper presented at the 2013 IEEE International Conference on Teaching, Assessment and Learning for Engineering (TALE2013), held 26-29 August 2013, Bali Dynasty Resort, Kuta, Indonesia.Submitted, September, 9, 2013. Published as resubmitted by the authors on March, 9, 2014.