http://www.icact.org/journal.asp

ICACT-TACT JOURNAL

Transactions on Advanced Communications Technology

Volume 4 Issue 6, Nov. 2015, ISSN: 2288-0003

Editor-in-Chief
Prof. Thomas Byeongnam YOON, PhD.

GIRI Global IT Research Institute

Journal Editorial Board

■ Editor-in-Chief

■ Editors

Dr. Youssef SAID, Tunisie Telecom, Tunisia
Dr. Noor Zaman, King Faisal University, Al Ahsa Hofuf, Saudi Arabia
Dr. Srinivas Mantha, SASTRA University, Thanjavur, India
Dr. Shahriar Mohammadi, KNTU University, Iran
Prof. Beonsku An, Hongik University, korea
Dr. Guanbo Zheng, University of Houston, USA
Prof. Sangho Choe, The Catholic University of Korea, korea
Dr. Gyanendra Prasad Joshi, Yeungnam University, korea
Dr. Tae-Gyu Lee, Korea Institue of Industrial Technology(KITECH), korea
Prof. Ilkyeun Ra, University of Colorado Denver, USA
Dr. Yong Sun, Beijing University of Posts and Telecommunications, China
Dr. Yulei Wu, Chinese Academy of Sciences, China
Mr. Anup Thapa, Chosun University, korea
Dr. Vo Nguyen Quoc Bao, Posts and Telecommunications Institute of Technology, Vietnam
Dr. Harish Kumar, Bhagwant institute of technology, India
Dr. Jin REN, North china university of technology, China
Dr. Joseph Kandath, Electronics & Commn Engg, India
Dr. Mohamed M. A. Moustafa, Arab Information Union (AIU), Egypt
Dr. Mostafa Zaman Chowdhury, Kookmin University, Korea
Prof. Francis C.M. Lau, Hong Kong Polytechnic University, Hong Kong
Prof. Ju Bin Song, Kyung Hee University, korea
Prof. KyungHi Chang, Inha University, Korea
Prof. Sherif Welsen Shaker, Kuang-Chi Institute of Advanced Technology, China
Prof. Seung-Hoon Hwang, Dongguk University, Korea
Prof. Dal-Hwan Yoon, Semyung University, korea
Prof. Chongyang ZHANG, Shanghai Jiao Tong University, China
Dr. H K Lau, The Open University of Hong Kong, Honh Kong
Prof. Ying-Ren Chien, Department of Electrical Engineering, National Ilan University, Taiwan
Prof. Mai Yi-Ting, Hsiuping University of Science and Technology, Taiwan
Dr. Sang-Hwan Ryu, Korea Railroad Research Institute, Korea
Dr. Yung-Chien Shih, MediaTek Inc., Taiwan
Dr. Kuan Hoong Poo, Multimedia University, Malaysia
Dr. Michael Leung, CEng MIET SMIEEE, Hong Kong
Dr. Abu sahman Bin mohd Supa'at, Universiti Teknologi Malaysia, Malaysia
Prof. Amit Kumar Garg, Deenbandhu Chhotu Ram University of Science & Technology, India
Dr. Jens Myrup Pedersen, Aalborg University, Denmark
Dr. Augustine Ikechi Ukaegbu, KAIST, Korea
Dr. Jamshid Sangirov, KAIST, Korea
Prof. Ahmed Dooguy KORA, Ecole Sup. Multinationale des Telecommunications, Senegal
Dr. Se-Jin Oh, Korea Astronomy & Space Science Institute, Korea
Dr. Rajendra Prasad Mahajan, RGPV Bhopal, India
Dr. Woo-Jin Byun, ETRI, Korea
Dr. Mohammed M. Kadhum, School of Computing, Goodwin Hall, Queen's University , Canada
Prof. Seong Gon Choi, Chungbuk National University, Korea
Prof. Yao-Chung Chang, National Taitung University, Taiwan
Dr. Abdallah Handoura, Engineering school of Gabes - Tunisia, Tunisia
Dr. Gopal Chandra Manna, BSNL, India

Dr. Il Kwon Cho, National Information Society Agency, Korea
Prof. Jiann-Liang Chen, National Taiwan University of Science and Technology, Taiwan
Prof. Ruay-Shiung Chang, National Dong Hwa University, Taiwan
Dr. Vasaka Visoottiviseth, Mahidol University, Thailand
Prof. Dae-Ki Kang, Dongseo University, Korea
Dr. Yong-Sik Choi, Research Institute, IDLE co., ltd, Korea
Dr. Xuena Peng, Northeastern University, China
Dr. Ming-Shen Jian, National Formosa University, Taiwan
Dr. Soobin Lee, KAIST Institute for IT Convergence, Korea
Prof. Yongpan Liu, Tsinghua University, China
Prof. Chih-Lin HU, National Central University, Taiwan
Prof. Chen-Shie Ho, Oriental Institute of Technology, Taiwan
Dr. Hyoung-Jun Kim, ETRI, Korea
Prof. Bernard Cousin, IRISA/Universite de Rennes 1, France
Prof. Eun-young Lee, Dongduk Woman s University, Korea
Dr. Porkumaran K, NGP institute of technology India, India
Dr. Feng CHENG, Hasso Plattner Institute at University of Potsdam, Germany
Prof. El-Sayed M. El-Alfy, King Fahd University of Petroleum and Minerals, Saudi Arabia
Prof. Lin You, Hangzhou Dianzi Univ, China
Mr. Nicolai Kuntze, Fraunhofer Institute for Secure Information Technology, Germany
Dr. Min-Hong Yun, ETRI, Korea
Dr. Seong Joon Lee, Korea Electrotechnology Research Institute, korea
Dr. Kwihoon Kim, ETRI, Korea
Dr. Jin Woo HONG, Electronics and Telecommunications Research Inst., Korea
Dr. Heeseok Choi, KISTI(Korea Institute of Science and Technology Information), korea
Dr. Somkiat Kitjongthawonkul, Australian Catholic University, St Patrick's Campus, Australia
Dr. Dae Won Kim, ETRI, Korea
Dr. Ho-Jin CHOI, KAIST(Univ), Korea
Dr. Su-Cheng HAW, Multimedia University, Faculty of Information Technology, Malaysia
Dr. Myoung-Jin Kim, Soongsil University, Korea
Dr. Gyu Myoung Lee, Institut Mines-Telecom, Telecom SudParis, France
Dr. Dongkyun Kim, KISTI(Korea Institute of Science and Technology Information), Korea
Prof. Yoonhee Kim, Sookmyung Women s University, Korea
Prof. Li-Der Chou, National Central University, Taiwan
Prof. Young Woong Ko, Hallym University, Korea
Prof. Dimiter G. Velev, UNWE(University of National and World Economy), Bulgaria
Dr. Tadasuke Minagawa, Meiji University, Japan
Prof. Jun-Kyun Choi, KAIST (Univ.), Korea
Dr. Brownson ObaridoaObele, Hyundai Mobis Multimedia R&D Lab , Korea
Prof. Anisha Lal, VIT university, India
Dr. kyeong kang, University of technology sydney, faculty of engineering and IT , Australia
Prof. Chwen-Yea Lin, Tatung Institute of Commerce and Technology, Taiwan
Dr. Ting Peng, Chang'an University, China
Prof. ChaeSoo Kim, Donga University in Korea, Korea
Prof. kirankumar M. joshi, m.s.uni.of baroda, India
Dr. Chin-Feng Lin, National Taiwan Ocean University, Taiwan
Dr. Chang-shin Chung, TTA(Telecommunications Technology Association), Korea

Editor Guide

■ Introduction for Editor or Reviewer

All the editor group members are to be assigned as a evaluator(editor or reviewer) to submitted journal papers at the discretion of the Editor-in-Chief. It will be informed by eMail with a Member Login ID and Password.

Once logined the Website via the Member Login menu in left as a evaluator, you can find out the paper assigned to you. You can evaluate it there. All the results of the evaluation are supposed to be shown in the Author Homepage in the real time manner. You can also enter the Author Homepage assigned to you by the Paper ID and the author's eMail address shown in your Evaluation Webpage. In the Author Homepage, you can communicate each other efficiently under the peer review policy. Please don't miss it!

All the editor group members are supposed to be candidates of a part of the editorial board, depending on their contribution which comes from history of ICACT TACT as an active evaluator. Because the main contribution comes from sincere paper reviewing role.

■ Role of the Editor

The editor's primary responsibilities are to conduct the peer review process, and check the final camera-ready manuscripts for any technical, grammatical or typographical errors.

As a member of the editorial board of the publication, the editor is responsible for ensuring that the publication maintains the highest quality while adhering to the publication policies and procedures of the ICACT TACT(Transactions on the Advanced Communications Technology).

For each paper that the editor-in-chief gets assigned, the Secretariat of ICACT Journal will send the editor an eMail requesting the review process of the paper.

The editor is responsible to make a decision on an "accept", "reject", or "revision" to the Editor-in-Chief via the Evaluation Webpage that can be shown in the Author Homepage also.

■ Deadlines for Regular Review

Editor-in-Chief will assign a evalaution group(a Editor and 2 reviewers) in a week upon receiving a completed Journal paper submission. Evaluators are given 2 weeks to review the paper. Editors are given a week to submit a recommendation to the Editor-in-Chief via the evaluation Webpage, once all or enough of the reviews have come in. In revision case, authors have a maximum of a month to submit their revised manuscripts. The deadlines for the regular review process are as follows:

Evalution Procedure	Deadline
Selection of Evaluation Group	1 week
Review processing	2 weeks
Editor's recommendation	1 week
Final Decision Noticing	1 week

■ Making Decisions on Manuscript

Editor will make a decision on the disposition of the manuscript, based on remarks of the reviewers. The editor's recommendation must be well justified and explained in detail. In cases where the revision is requested, these should be clearly indicated and explained. The editor must then promptly convey this decision to the author. The author may contact the editor if instructions regarding amendments to the manuscript are unclear. All these actions could be done via the evaluation system in this Website. The guidelines of decisions for publication are as follows:

Decision	Description
Accept	An accept decision means that an editor is accepting the paper with no further modifications. The paper will not be seen again by the editor or by the reviewers.
Reject	The manuscript is not suitable for the ICACT TACT publication.
Revision	The paper is conditionally accepted with some requirements. A revision means that the paper should go back to the original reviewers for a second round of reviews. We strongly discourage editors from making a decision based on their own review of the manuscript if a revision had been previously required.

■ Role of the Reviewer

Reviewer Webpage:

Once logined the Member Login menu in left, you can find out papers assigned to you. You can also login the Author Homepage assigned to you with the paper ID and author's eMail address. In there you can communicate each other via a Communication Channel Box.

Quick Review Required:

You are given 2 weeks for the first round of review and 1 week for the second round of review. You must agree that time is so important for the rapidly changing IT technologies and applications trend. Please respect the deadline. Authors undoubtedly appreciate your quick review.

Anonymity:

Do not identify yourself or your organization within the review text.

Review:

Reviewer will perform the paper review based on the main criteria provided below. Please provide detailed public comments for each criterion, also available to the author.

- How this manuscript advances this field of research and/or contributes something new to the literature?
- Relevance of this manuscript to the readers of TACT?
- Is the manuscript technically sound?
- Is the paper clearly written and well organized?
- Are all figures and tables appropriately provided and are their resolution good quality?
- Does the introduction state the objectives of the manuscript encouraging the reader to read on?
- Are the references relevant and complete?

Supply missing references:

Please supply any information that you think will be useful to the author in revision for enhancing quality of the paperor for convincing him/her of the mistakes.

Review Comments:

If you find any already known results related to the manuscript, please give references to earlier papers which contain these or similar results. If the reasoning is incorrect or ambiguous, please indicate specifically where and why. If you would like to suggest that the paper be rewritten, give specific suggestions regarding which parts of the paper should be deleted, added or modified, and please indicate how.

Journal Procedure

Dear Author,

- **You can see all your paper information & progress.**

- **Step 1. Journal Full Paper Submission**

Using the Submit button, submit your journal paper through ICACT Website, then you will get new paper ID of your journal, and send your journal Paper ID to the Secretariat@icact.org for the review and editorial processing. Once you got your Journal paper ID, never submit again! Journal Paper/CRF Template

- **Step 2. Full Paper Review**

Using the evaluation system in the ICACT Website, the editor, reviewer and author can communicate each other for the good quality publication. It may take about 1 month.

- **Step 3. Acceptance Notification**

It officially informs acceptance, revision, or reject of submitted full paper after the full paper review process.

Status	Action
Acceptance	Go to next Step.
Revision	Re-submit Full Paper within 1 month after Revision Notification.
Reject	Drop everything.

- **Step 4. Payment Registration**

So far it's free of charge in case of the journal promotion paper from the registered ICACT conference paper! But you have to regist it, because you need your Journal Paper Registration ID for submission of the final CRF manuscripts in the next step's process. Once you get your Registration ID, send it to Secretariat@icact.org for further process.

- **Step 5. Camera Ready Form (CRF) Manuscripts Submission**

After you have received the confirmation notice from secretariat of ICACT, and then you are allowed to submit the final CRF manuscripts in PDF file form, the full paper and the Copyright Transfer Agreement. Journal Paper Template, Copyright Form Template, BioAbstract Template,

Journal Submission Guide

All the Out-Standing ICACT conference papers have been invited to this "ICACT Transactions on the Advanced Communications Technology" Journal, and also welcome all the authors whose conference paper has been accepted by the ICACT Technical Program Committee, if you could extend new contents at least 30% more than pure content of your conference paper. Journal paper must be followed to ensure full compliance with the IEEE Journal Template Form attached on this page.

➢ How to submit your Journal paper and check the progress?

Step 1. Submit	Using the Submit button, submit your journal paper through ICACT Website, then you will get new paper ID of your journal, and send your journal Paper ID to the Secretariat@icact.org for the review and editorial processing. Once you got your Journal paper ID, never submit again! Using the Update button, you can change any information of journal paper related or upload new full journal paper.
Step 2. Confirm	Secretariat is supposed to confirm all the necessary conditions of your journal paper to make it ready to review. In case of promotion from the conference paper to Journal paper, send us all the .DOC(or Latex) files of your ICACT conference paper and journal paper to evaluate the difference of the pure contents in between at least 30% more to avoid the self replication violation under scrutiny. The pure content does not include any reference list, acknowledgement, Appendix and author biography information.
Step 3. Review	Upon completing the confirmation, it gets started the review process thru the Editor & Reviewer Guideline. Whenever you visit the Author Homepage, you can check the progress status of your paper there from start to end like this, " Confirm OK! -> Gets started the review process -> ...", in the Review Status column. Please don't miss it!

Volume. 4 Issue. 6

Method and Prototype of Utility for Partial Recovering Source Code for Low-Level and Medium-Level Vulnerability Search

Mikhail Buinevich*, Konstantin Izrailov*, Andrei Vladyko*
*The Bonch-Bruevich Saint-Petersburg State University of Telecommunications, Russian Federation, Saint-Petersburg, 22-1 Prospekt Bolshevikov
bmv1958@yandex.ru, konstantin.izrailov@mail.ru, vladyko@bk.ru

***Abstract*— The article describes a automated method for searching of low-level and medium-level vulnerabilities in machine code, which is based on its partial recovering. Vulnerability search is positioned in the field of telecommunication devices. All various and typical vulnerabilities in source code and algorithms for its search is given. The article contains examples of usage method and its utility. There is forecast to develop methods and utilities in the near future.**

***Keywords*— machine code, reverse-engineering, static analyzer, telecommunication devices, vulnerability**

I. INTRODUCTION

Accidental and intentional errors that lead to vulnerabilities in software (hereinafter referred to as the SW) are one of the top challenges of the modern worlds that have taken the informatization path of development. Though the errors that have been made by hackers are seen less, they make the SW more unsafe, as long as such errors are the ultimate goal of the attackers. With view to the fact that critical information is usually shared via telecommunication devices, with the functional of such devices implemented with the help of the SW, the task of vulnerability search is one of the principal tasks of information security. This task is sophisticated and depends on an underdeveloped search methodology, which is defined via a set of methods used for such search. In this context, the efficiency of such methods is a function of purely practical application aspects, i.e. speed and complexity, and is now estimated as extremely unsatisfactory.

Manuscript received on June 19, 2015. This work is a follow-up of the invited journal to the accepted conference paper of the 17th International Conference on Advanced Communication Technology.

M. V. Buinevich is with The Bonch-Bruevich Saint-Petersburg State University of Telecommunications, Russian Federation, Saint-Petersburg, 22-1 Prospekt Bolshevikov (e-mail: bmv1958@yandex.ru).

K. E. Izrailov is with The Bonch-Bruevich Saint-Petersburg State University of Telecommunications, Russian Federation, Saint-Petersburg, 22-1 Prospekt Bolshevikov (corresponding author to provide phone: +7(921)555-2389; fax: none; e-mail: konstantin.izrailov@mail.ru).

A. G. Vladyko is with The Bonch-Bruevich Saint-Petersburg State University of Telecommunications, Russian Federation, Saint-Petersburg, 22-1 Prospekt Bolshevikov (e-mail: vladyko@bk.ru).

Therefore, development of new and highly efficient methods for vulnerability search for telecommunication device SW is of theoretical (in terms of methodology) and, obviously, practical interest.

II. ANALYZING

In order to analyze available methods for SW vulnerability search, we would like to break such methods into the following groups by their application target.

Any methods that are applied exclusively to the SW source code fall into the first group and they are most abundant. Such methods are quite developed and are used very efficiently. A large base of typical source code vulnerabilities and methods for identification of such vulnerabilities has been collected. CppCheck, Lint and Cland may serve as implementation examples for the C/C++ code.

If the source code is not available, you will have to search for vulnerabilities using the final representation, i.e. machine code. Such methods form the second group and they usually rely on code disassembling and manual analysis by security experts (hereinafter referred to as the Expert). Individual implementations of such methods you may found in such products as Binary Static Analysis (SAST) by Veracode and Software Static Analysis Toolset by MALPAS. However, such methods have a different ultimate goal and may not be used as a comprehensive solution for the search of machine code vulnerabilities. However, a certain quantity of theoretical works, that are close to the solution of this problem still exists and is based on the use of the character programming [1].

It should be mentioned that the source code here means any code that must be compiled in a platform-dependent machine code for running such code – so called unmanaged code (e.g., C++, Pascal). A managed code (e.g. Java, C#) is an alternative and is compiled into an interim bytecode, which is then run on a virtual machine and is unambiguously converted into the source code. Availability and search for any vulnerabilities in the managed code are, obviously, a different task, which is less popular and more simple.

As long as a telecommunication device is usually supplied with the SW in the form of the machine code already installed on such device and one does not have any access to the source code, it seems so that the search for any vulnerabilities in the telecommunication device is done at the moment exclusively using manual techniques.

We should also mention a vast majority of telecommunication device models, modifications and SW versions leading to a tremendous number of various machine code images. And such machine code images may be of a rather big size – up to hundreds of megabytes. Also, it is obvious that each and every image must be analyzed "from scratch".

Thus and so, there is a critical task of search for vulnerabilities in the telecommunication device machine code in the above area of interest, and its solving efficiency is obviously not satisfactory. In this context, efficiency means a total number of vulnerabilities found in the machine code for a reasonable time, but not in the defined volume (i.e. vulnerability density). So, the manual technique would allow for identification of a larger number of vulnerabilities in small machine code volumes vs. any automated techniques. However, application of such technique for any line of telecommunication device machine code will be so time-consuming in practice, even for a medium-sized volume, that the results will just not keep up with the release of new upgrades. Highly qualified Experts must also always be available for application of the manual technique, which is not always possible.

Design of a method and means of automated search for vulnerabilities in the machine code may be an obvious solution of the task, in which case involvement of human factor, especially highly qualified, will be minimized. Operational capability and vulnerability identification for large code volumes exactly for a small time at the expense of quality and density of the vulnerabilities found is the basic requirement to such method. However, the last factor is not critical, as long as you may always add any manual search methods, with the findings of the automated search simply facilitating such work.

Comparison of manual and automatic methods for vulnerability search, using various criteria, is presented in Table I.

TABLE I
COMPARISON OF METHODS FOR VULNERABILITY SEARCH BY CRITERIA

Criteria	Manual	Automatic
Search time	Long	Short
Number of detected vulnerabilities	Many	Few
Detected vulnerability patterns	All	Template-based
Amount of code processed	Small	Large
Required qualification	Expert	Engineer
Ability to bypass anti-detection mechanisms	Yes	Sometimes
Source code information	Preferable	Not needed
Formalization of results	Possible	Always

Advantages of the automatic method are highlighted in Table I, which brings us to the following conclusion: fully automated search methods can be used efficiently to detect telecommunication devices machine code vulnerabilities.

For better review of the object domain, let us divide all vulnerabilities into 4 types, according to their layout at the software build-up levels. Such division is presented in Figure 1.

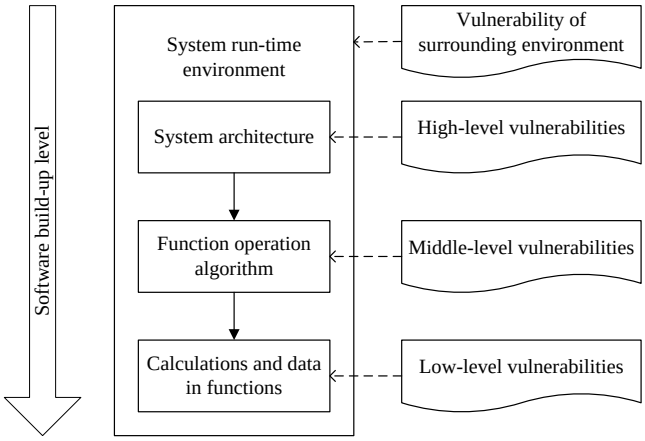


Fig. 1. Types of vulnerabilities according to SW build-up level

First of all, these include low-level vulnerabilities, such as calculation errors, data structure, data access etc. For example, dividing by zero, or incorrect structure field processing due to non-standard alignment of the members, may fall into this type. Second of all, these include medium-level vulnerabilities, such as incorrect implementation of algorithms and functions, transfer of input parameters, function returns etc. For example, occurrence of infinite cycles and recursions, appearance of an unreachable code (despite the fact that such code does not usually lead to any adverse effects), and IF-ELSE transition test errors may fall into this type. Third of all, these include high-level vulnerabilities, such as program system architecture errors [2]: violation of general principles of system functioning and security, incorrect implementation of various mechanisms and protocols etc. For example, errors in implementation of algorithm of common secret key importation, according to Diffie-Hellman protocol may fall into this type. And, forth of all, these include vulnerabilities of the surrounding environment, such as errors in run-time modules of the active program system etc. For example, DLL injection (running code within the address space of another process by forcing it to load a dynamic-link library) and operating errors of objective code loader into the system address area (ld.so for Unix-systems) may fall into this type.

The following available developments may be used as initial data for the task solution. First of all, as mentioned before, the development of methods for vulnerability search has allowed us to gather certain information about the vulnerabilities in the source code and search specifics. Second of all, the required vulnerability search may be implemented via conversion of the machine code of the device into the source code and application of the available search algorithms to such code. Therewith, any previous work of the authors [3] also touched this task directly, as long as it aimed at machine code algorithm recovery in the telecommunication device, which may be considered a

partial source code recovery. Having combined such developments and with view to our goal, we hereby offer a solution of the task, which includes two major steps. We must systemize typical source code vulnerabilities and adapt such vulnerabilities to the machine code, i.e. select the vulnerabilities that may be found and identified in the machine code first. We must then develop a method for machine code vulnerability search (hereinafter referred to as the Method) based on the original recovery method for the algorithms and adapted vulnerabilities in the machine code. The Method offered is designed for searching two types of vulnerabilities, such as: low-level vulnerabilities and partially medium-level vulnerabilities, which is the most important peculiarity of this Method. In this case, the basic method of the authors [3] is suitable for full-sized search of medium-level and high-level vulnerabilities only. Machine code of the telecommunication device is usually a detached binary image and is run completely, using special hardware. Therefore, search of vulnerabilities of the surrounding environment is not covered by this task. Let us further consider vulnerabilities that are connected to the proposed technique.

A. Adapted vulnerabilities in the machine code.

As long the vulnerability in the SW is basically an integral part of the SW contents (i.e. functional), any code presentation shape conversions may not affect vulnerability in any way, i.e. make it disappear or change significantly. Therefore, any source code vulnerability will be reflected in the machine code to some degree. However, due to any operations that reduce the program “structurization” (compilation, assembling), the final presentation of the vulnerability may become "washed out", which may make it practically impossible to identify such vulnerability, even in cases of reversed engineering (disassembling, decompilation). Let us mark out the vulnerabilities that can be found in the source code, and are still integral in the machine code, which will be indicative of the potential ability of them being identified in the code. For this purpose there is a sufficient quantities of theoretical studies and practical implementations, for example, in the works [4] and [5].

Vulnerability 1 – Dividing by 0

This vulnerability can be found in operations of dividing by 0, which should not occur in any correct programs at all. The reasons for such vulnerability may include missed check by zero value of denominators of expressions, and operational logic mistakes. Exclusion of division by zero will be called up as a result, which may lead to incorrect program exit. This vulnerability search algorithm is based on defining any possible variable value ranges, which are included in the expression with dividing and signalization of the possibility of the dominator equal to 0. Example of a code with this vulnerability is given at the following listing.

```
if(y == 0)
  z = x / y;
else
  z = x * y;
```

Vulnerability 2 – Using a non-initialized variable

This vulnerability occurs during the first use of a variable that was not assigned any initial value. The reasons for such vulnerability may include any programmer's error, which usually includes missing initial value of the variable or assumption of such value equalling to 0 by default. As a result, the variable may take random values (so called 'garbage'), which will lead to incorrect calculations. The vulnerability search algorithm is based on determining locations of the initial variable assignment/use and signalization, if such use occurred before the assignment. . An example of code, which involves this vulnerability, is presented in the following listing.

```
int x, y;
y = x * 2;
```

Vulnerability 3 – Buffer overflow

This vulnerability occurs due to no control over going out of the object stored in the buffer, which usually constitutes an array. Overwriting of the memory area, which is physically located out of the buffer, may occur and it may lead to writing off the contents of any other objects and change of the program code, or simply exclusion into the protected memory based on the record. The vulnerability search algorithm is based on detecting the buffer memory range, operation algorithms for the indexes of buffer objects, and possible values of such pointers, and signalization in case of index going out of the range [6]. Example of a code with this vulnerability is given at the following listing.

```
int arr[10];
…
arr[10] = 0;
```

Vulnerability 4 – Handling incorrect memory pointers

This vulnerability can be found, while dereferencing of pointer containing incorrect memory paths. The reasons for such vulnerability may include errors in function operation algorithm or incorrect pointer initialization. Reading and writing using the pointer may be done on the area of the executed code as a result. Dereferencing of the $0x0^{th}$ pointer is a special case. The vulnerability search algorithm is based on determining pointer values and dereference locations. Example of a code with this vulnerability is given at the following listing.

```
int *ptr_1 = 0x0;
int *ptr_2 = &funct;
*ptr1 = 0;
*ptr2 = 0;
```

Vulnerability 5 – Memory leaks

This vulnerability can be found in case of unlimited memory use. The reasons for such vulnerability may include missing required operations to free up the memory, in particular in case of cyclic execution. Memory shortage exclusion or write-off of the contents of used objects (due to incorrect stack handling) may occur as a result. Although such situation does not make the full-featured vulnerability, it may interrupt program operation in case of prolonged use, which typical just for the telecommunication device. Unfortunately, there is not any common vulnerability search algorithm available (as long as memory selection and freeing-up are phenomena that are defined exclusively within the program and program libraries). Nevertheless, manual algorithm set-up, such as obvious function and

memory operation template determination will allow for identifying vulnerabilities of such type with the help of management flow graph analysis, including call-ups of such functions in the graph. An example of code, which involves this vulnerability, is presented in the following listing ('malloc()' is a memory allocation function).

```
funct(int x){
  int *ptr;
  while(x){
    ptr = malloc(10);
    if(x % 2 == 0)
      dummy(ptr);
    --x;
  }
}
```

Vulnerability 6 – Infinite loops and recursions

This vulnerability can be found in case of looping of function management flows or calling up the function as a result of incorrect program logics. Thus and so, the program may go, under certain circumstances, into a cycle that does not have any executable conditions for completion, or call up the same function indefinitely. This vulnerability can be identified by building complete management flow graphs and call-ups with further analysis of the conditions for simultaneous execution of their paths. It should be mentioned that the software of the telecommunication device often has an architecture, which consists of the single processing cycle for the incoming network packages, and such case must be processed separately. An example of code, which involves this vulnerability, is presented in the following listing.

```
int funct(int x){
  return funct(x+1);
}
…
bool flag = false;
do{
  …
  flag = true;
}while(flag);
```

Vulnerability 7 – Unused code

This situation happens, if there are code areas in the programs with their instructions never to be run, and it corresponds to the destructed function algorithm structure. The reasons for such vulnerability may include errors in function algorithm operation, and a consequence of "rough" introduction of an alien code in the program. Although such situation does not make the full-featured vulnerability, it is indicative of an abnormality in the machine code, which is a potential vulnerability of the "bookmark" type. An example of code, which involves this vulnerability, is presented in the following listing.

```
int funct(int x){
  if(x)
    return 1;
  else
    return 2;
  x += 1;
  return x;
}
```

B. Method for vulnerability search in the machine code.

As offered, the method for vulnerability search in the machine code must contain consecutive steps for source code recovery and vulnerability search using such code with the help of the existing algorithms. Although complete source code recovery (i.e. decompilation) is not practically possible, it may be recovered partially, which is a minimum requirement to search performance. Thus, for example, it is not crucially necessary to recover names of variables and points of return from the function in order to identify buffer overload vulnerability.

As mentioned before, a number of sub-tasks (algorithm recovery to be specific) were solved partially in the original method. Therefore, some phases of this method may be used in this Method. In particular, it is reasonable to use the IDA Pro product [7], which has the following functional. First of all, this product is a full-scale machine code disassemble for a large variety of processors. And second of all, this product performs a partial machine code recovery, as long as it divides memory areas into functions and data blocks, uses debugging information, if any in the machine code, and recognizes library functions by their signatures. It should be mentioned that crucial product specifics, such as interactivity and debugging capacity, may not be used for this Method in practice.

The rest of the Method relies on implementation of adapted machine code vulnerability search algorithms using the interim representation including output of any findings.

If we sum up the above, basic phases of the Method are:

Phase 1 – Disassembling machine code

The machine code is converted to the assembler for disassembly and analysis. This phase may be fully implemented based on the IDA Pro. This phase may be implemented completely on the basis of IDA Pro, which includes an external API and supports internal scripts. A rather correct division of code and data sections, definition of function body and global variables, which may be used at further phases, are the operational specifics of this product.

Phase 2 – Recovering partially source code

The assembly code is disassembled, an internal program representation is built, necessary conversions are made, source code elements are assumed, and algorithms and other necessary information is recovered – this phase and all further phases must be carried out using a specialized software application (hereinafter referred to as the Utility). It is obvious that SW analysis for a certain processor is not possible, unless its machine code is supported in the IDA Pro and FrontEnd of Utility (i.e. input parser) is available for its assembler. The basics code recovery is made using the IDA Pro immediately after the decompilation and it is reflected in the assembler, which is generated in Phase 1. This phase may be implemented in most part using algorithms based on the original machine code algorithm recovery method, as complemented by the algorithms gathering the information that is required for searching the adapted vulnerabilities. First of all, aliases must be supported, i.e. information about common code areas that are indicated by several different pointers or objects. Second

of all, calculation of any possible value ranges, which are used in the object program, including pointers is required. Third of all, variable life time graphs must be plotted, including points of their first and last initialization / use.

Phase 3 – Vulnerability search

Search for each of adapted vulnerabilities is made using the internal code presentation obtained. It is obvious that such presentation must not be a sophisticated text (usually used for SW development), but a set of specialized graphs, tables and structures determining program run. Such presentation will allow for more effective searching. Generalized search algorithms for each vulnerability have been provided earlier.

Phase 4 – Gathering findings

Machine code analysis findings are generated, i.e. information about the code (scope, number of functions), identified vulnerabilities or suspected locations of vulnerabilities etc. This processed SW code for the telecommunication device is especially characterized by its possible large size, various variations, and contents. Thus and so, this Method must be applicable multiple times with summarization of the findings obtained. Therefore, the presentation format of the vulnerabilities found must be suitable for program processing (i.e. its syntax must be strict) and for investigation by the Expert (i.e. human friendly). YAML [8] is a suitable presentation option for this task, which is characterized by such peculiarities as formalization and readability.

Diagram of Method phases and data used for such phases can be found in Fig. 2.

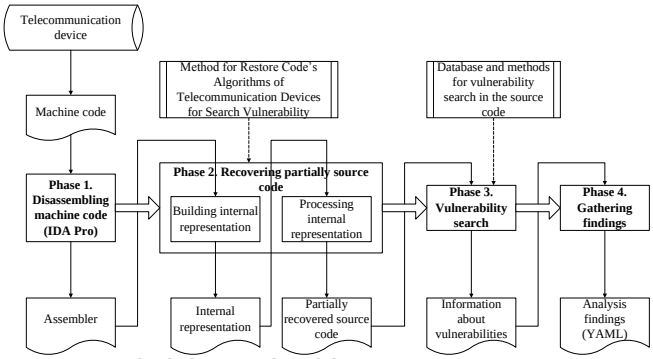


Fig. 2. Method phases and used data

Note. The process of obtaining the telecommunication device firmware and reducing such firmware to the machine code is out of scope of this Method, as long as it is a separate and purely technical task that may be solved quite effectively.

III. Evaluating the Method

An imaginary experiment of application of individual phases of the offered Method on typical abstract machine codes, containing each of vulnerabilities in the original source code was done to study the specifics of such Method. We will get an unambiguous assembly code representation after the Phase 1. This representation will also include vulnerabilities of all types, as long as IDA Pro converts binary processor instructions into text lines including operations unambiguously. The machine code does not usually include any debugging information, and, therefore, a partial code recovery in the Phase 2 will be done without any source code meta information, such as function names and arguments. This does not appear to be rather significant; however, it may still affect the accuracy of our findings. To the contrary, machine code obfuscation could worsen recovery efficiency significantly, as long as it usually destructs the algorithm structure, albeit it is used quite rarely. Machine code optimization (source code compilation involving optimization options to be specific) is not applied to all SW types; in particular, it is not used often the telecommunication device. Various steganography messages embedded in the executable file, at this phase, will be lost; nevertheless, it is likely note rather than a disadvantage [9]. Efficiency of the Phase 3 depends on the outcomes of the Phase 2 completely, i.e. on the quality of the partially recovered source code. Therefore, assigning of some degree of reliability to the found vulnerabilities may be reasonable. This may prove useful for a semi-automatic application of the Method, i.e. including further analysis done by the Expert. Phase 4 involves flow-by-flow output of vulnerability search findings in a special format and does not depend on the previous phases or analyzed machine code greatly. YAML does not have any application limitations. Therefore, any vulnerability details (if any) for each typical machine code will be gathered in one place and converted into the uniform base suitable for the manual analysis.

According to our imaginary experiment, application of the Method and its individual phases for vulnerability search in the telecommunication device machine code can be justified strictly and logically. The machine code for the processor supported by IDA Pro and Utility will be Method input, and a list of vulnerabilities in YAML will be Method output. Unsuitability for any obfuscated machine codes and average efficiency for any optimized code are among Method limitations.

IV. A hypothetical example

Let us consider a hypothetical example of operation of this technique, and pay special attention of Phase 2, which is most complicated. As it was mentioned before, phase realization must be in the form of a separate automating utility and may be taken partially from the utility of the basic technique [3]. At the moment, development of a utility prototype for the technique is in progress; however, we may already predict a scheme and operational details of such utility. Various program representations at all phases of the technique and utility (from a source code to a formalized list of vulnerabilities) are presented below.

A. Input data (source code)

Let us assume we have a program that comprises 'funct()' function, which involves a vulnerability in the form of a possible division by zero (a code of such function is presented in the following listing, using the high-level C language).

```
01:    int funct(int x, int y){
02:        int z = 0;
03:        if (y == 0) {
04:            z = x / y;
```

```
05:         } else {
06:             z = x * y;
07:         }
08:         return z;
09:     }
```

The vulnerability is located in line '04:' and it occurs, if the IF-ELSE condition is fulfilled in line '03:', which leads to dividing variable 'y' equalling to 0.

B. Input data (machine code)

Machine code of this example for the PowerPC processing has the following listing.

```
94 21 FF D0 93 E1 00 2C  7C 3F 0B 78 90 7F 00 18
90 9F 00 1C 38 00 00 00  90 1F 00 08 80 1F 00 1C
2F 80 00 00 40 9E 00 18  81 3F 00 18 80 1F 00 1C
7C 09 03 D6 90 1F 00 08  48 00 00 14 81 3F 00 18
80 1F 00 1C 7C 09 01 D6  90 1F 00 08 80 1F 00 08
7C 03 03 78 39 7F 00 30  83 EB FF FC 7D 61 5B 78
```

It is obvious that we will not be able to discover the fact of vulnerability existence by an expert manual technique in this representation. Application of automated search algorithm may be successful; nevertheless, realization of such algorithms will be highly non-trivial in this representation.

C. Phase 1 – Disassembling machine code

Application of the IDA Pro will result in an assembly representation of the program, like the one in the following listing.

```
0x00:     stwu      r1, -0x30(r1)
0x04:     stw       r31, 0x2C(r1)
0x08:     mr        r31, r1
0x0C:     stw       r3, 0x18(r31)
0x10:     stw       r4, 0x1C(r31)
0x14:     li        r0, 0
0x18:     stw       r0, 8(r31)
0x1C:     lwz       r0, 0x1C(r31)
0x20:     cmpwi     cr7, r0, 0
0x24:     bne       cr7, loc_3C
0x28:     lwz       r9, 0x18(r31)
0x2C:     lwz       r0, 0x1C(r31)
0x30:     divw      r0, r9, r0
0x34:     stw       r0, 8(r31)
0x38:     b         loc_4C
0x3C: loc_3C:
0x3C:     lwz       r9, 0x18(r31)
0x40:     lwz       r0, 0x1C(r31)
0x44:     mullw     r0, r9, r0
0x48:     stw       r0, 8(r31)
0x4C: loc_4C:
0x4C:     lwz       r0, 8(r31)
0x50:     mr        r3, r0
0x54:     addi      r11, r31, 0x30
0x58:     lwz       r31, -4(r11)
0x5C:     mr        r1, r11
0x60:     blr
```

Despite the fact that such representation may already be analyzed by experts, required efforts for such process are critically high.

D. Phase 2 – Recovering partially source code

This phase and all further phases must be implemented, using a utility that is being developed at the moment. Most of its algorithms and representations will be similar to those of the basic technique, which uses the operational utility prototype [10].

First of all, the assembly representation will be converted into an abstract syntax tree that describes the source program in a formalized and structured way. Then, it will be converted into a similar internal representation. Complete independency from the run-time processor and source assembler is the specifics of such representation, as long as all operations and variables are fully abstract. The appearance of such trees is quite similar. A text form of such trees is presented in the following listing, where left indent is indicative of the depth of an element.

```
IrList()
  IrFunct('funct')  // Function 'funct()'
    IrArgs
      IrReg('r1')   // Function arg N_1
      IrReg('r2')   // Function arg N_2
    IrLocalVars
      IrReg('r3'), value='0' // Local var N_1
    IrList()        // Function body
      IrBranch      // if (r2 == 0) goto label_1
        IrCond('beq'), kind='=='
          IrReg('r2')
          IrInteger('0')
        IrLabel('label_1')
      IrOperation('mr'), kind='='    // r3=r1*r2
        IrReg('r3')
        IrOperation('mul'), kind='*'
          IrReg('r1')
          IrReg('r2')
      IrGoto('b')                // goto label_2
        IrLabel('label_2')
      IrLabel('label_1')             // label_1:
      IrOperation('mr'), kind='='    // r3=r1/r2
        IrReg('r3')
        IrOperation('div'), kind='/'
          IrReg('r1')
          IrReg('r2')
     IrLabel('label_2'),    name='label_2'    //
label_2:
      IrReturn('blr')    // return r3
        IrReg('r3')
```

Second of all, a graph of basic blocks, which reflects the sequence of operations and conditional transfers will be constructed, based on the internal representation tree. Third of all, the internal representation tree will be analyzed for any possible values of the variables. We will obtain the following graph, as a result.

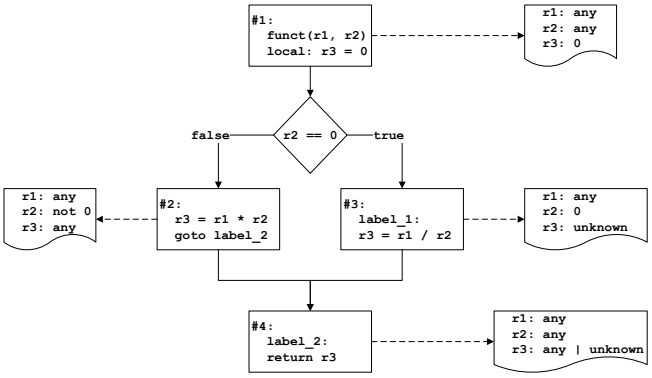


Fig. 3. Intermediate Representation Graph with variables value ranges

According to the graph in Figure 3, Block#1 is a function header, including arguments 'r1', 'r2' and local variable 'r3', which is initialized with '0' value. Conditional check 'r2 == 0' and transitions to blocks #2 and #3, depending on the results of such conditional check, will form the first

block of the function. Block #2 performs a multiplication of variables ‘r1’ and ‘r2’, and the result is entered into variable ‘r3’. Thus and so, according to the conditional transition, variable 'r2' may take any values, except for '0', and variable ‘r3’ may take any values. Block #3 performs a division of variables ‘r1’ and ‘r2’, and the result is entered into variable ‘r3’. In this case, according to the conditional transition, variable ‘r2’ equals to 0. Therefore, ‘Divide by zero’ exclusion occurs after the division in the program, and value of variable 'r3' will not be defined. Block #4 performs an exit from the function and return of the value in variable 'r3'.

E. Phase 3 – Vulnerability search

Vulnerability search algorithms are applied at this phase. According to the internal representation graph and possible variable values, one of the search algorithms may detect the fact of dividing by 0 in block #3, which leads to the exclusion. This means an incorrect program behaviour in general. This way the error in the form 'Vulnerability – Dividing by 0' is discovered. Thus and so, the algorithm will find the vulnerability in the code and will add details about such vulnerability to the resulting list.

F. Phase 4 – Gathering findings

The final phase of the technique will gather all information about the vulnerabilities and will generate such information in a formalized, but operator-friendly way, i.e. in the YAML format (see the following listing).

```
Sources:
  - Name: "test.asm"
  - Vulnerabilities:
    -
      - Type: "Dividing by 0"
      - Machine address: 0x00000030
      - Machine instruction: "divw r0, r9, r0"
      - BasicBlock: 3
      - IRSubTree: "
          IrOperation('div'), kind='/'
            IrReg('r1')
            IrReg('r2')"
```

This example justifies to some extent the accuracy of implementation of the proposed technique, as well as the automating utility that is being developed.

V. CONCLUSION

This methods aims at solving the most important task of machine code vulnerability search, which is critical specifically for the telecommunication device, as long as the SW of such device affects the security the information transferred. Application of the available developments in the area of vulnerability search by the source code as combined with the previous research of the authors [11] are the specifics of implementation of this Method. Completion of implementation of a software tool to operate the Method automatically and approbate such Method using existing telecommunication device software will be the only next logical step and will allow to prove Method feasibility, in which case the theoretical value of the Method will be the development of the methodology of the highly popular machine code security area in terms of vulnerability search. Combination of the offered and previous original methods forms a methodological basis for search of vulnerabilities of all possible types in the machine code. Practical value of the Method is priceless, as long as identification of a vast number of machine code vulnerabilities for a tremendous number of existing telecommunication device firmware is foreseen upon automated application of such Method. Some of such vulnerabilities are still in action, which reduces significantly the overall security of information transferred via telecommunication networks.

ACKNOWLEDGMENT

In the end, we would like to express our gratitude to the Bonch-Bruevich Saint-Petersburg State University of Telecommunications for the possibility and support in development of the area of scientific studies, which was described in this and other articles.

The authors are thankful to their parents and Teachers for their craving for knowledge cultivated since their childhood and scientific achievements, which have been demonstrated multiple times using their own examples. First of all, to Evgeny Izrailov for development of source beams ultracold polarized H-atoms for metrology and physical investigations. Second of all, to Vladimir Rosenberg for developing the special mathematical management support methodology.

REFERENCES

[1] Cova. M, Felmetsger V., Banks G., and Vigna G., "Static Detection of Vulnerabilities in x86 Executables," in *Proc. 22nd Annu. Computer Security Applications Conference*, Miami Beach, 2006, pp. 269-278

[2] Buinevich M.V. and Izrailov K.E., “Architectural software vulnerabilities,”theses, *6th Congressional Research Undergraduate and Graduate Students “Engecon-2013”*, 2013

[3] Buinevich M.V. and Izrailov K.E., “Method and Utility for Recovering Code Algorithms of Telecommunication Devices for Vulnerability Search,” in *Proc. IEEE 16th Int. Conf. on Advanced Communications Technology*, PyeongChang, 2014, pp. 172-176

[4] Xin L. and Wandong C., “A program vulnerabilities detection frame by static code analysis and model checking,” in *Proc. IEEE 3rd Int. Conf. on Communication Software and Networks*, Xi'an, 2011, pp. 130-134

[5] Ivannikov V. P., Belevantsev A. A., Borodin A. E., Ignatiev V. N., Zhurikhin D. M., and A. I. Avetisyan, “Static analyzer Svace for finding of defects in program source code,” in *Proc. Institute for System Programming of Russian Academy of Sciences*, vol. 26, no. 1, pp. 231-250, 2014

[6] Rawat S. and Mounier L., “Finding Buffer Overflow Inducing Loops in Binary Executables,” in *Proc. IEEE 6th Int. Conf. on Software Security and Reliability*, Gaithersburg, 2012, pp. 177-186

[7] The IDA Pro website [Online]. Available: https://www.hex-rays.com/products/ida/

[8] The official YAML website [Online]. Available: http://yaml.org

[9] Shterenberg S.I. and Krasov A.V., “Variants of embedding information in the executable file with format .Intel HEX,” *Spb.: Information Technology and Telecommunications*, no. 4, pp. 52-64, 2013

[10] Izrailov K.E., “The internal representation of a prototype utility for the algorithmization of the code,” in *Proc. 2th Int. Scientific and Practical Conf. on Fundamental and Applied Research in the Modern World*, Saint Petersburg, 2013, pp. 79–90

[11] The research area and method of the authors [Online]. Available: http://www.demono.ru

Mikhail Buinevich was born in 1958 in the USSR. He received education of the military engineer of electronic engineering.
He served in the naval fleet and government agencies for information security. He held classes at various universities. His research interests include methods of information security. He has more than 100 scientific works. His primary publications are as

follows:
1. M.V. Buinevich and others. Safety provision of high-security objects of the naval fleet in relation to damage effects in crisis and emergency situations in peacetime./ Under the editorship of the admiral V.S. Vysotskii.- Saint Petersburg: Publishing house ELMOR, 2008.- 300 p.
2. M.V. Buinevich and others. Provision of organizational and technical support of stability of function and safety of general communications network./ Under the general editorship of S.M. Dotsenko.- Saint Petersburg: Publishing house SPbSUT, 2013.- 142 p.
Dr. Prof. Buinevich, at the present time, is the professor of the Protected Communications System Chair of Saint Petersburg State University of Telecommunications (SPbSUT).

Konstantin Izrailov was born in 1979 in the city of Saint Petersburg (Russia). In 1996 he graduated from Saint Petersburg State Polytechnic University, Physical and Mechanical Department.
At the moment he is a postgraduate student of the Protected Communications System Chair of Saint Petersburg State University of Telecommunications (SPbSUT). He has about 20 published articles; he is an author of 3 scientific and research works and has a patent on the software tool. His scientific interests include information security, search of vulnerabilities in machine code, reverse engineering and telecommunication devices.
Mr. Izrailov has the title of the best postgraduate student of SPbSUT in 2012 and is the presidential scholar in 2013.

Andrei Vladyko (IEEE member (M'14)) acquired his Degree of the Candidate of Sciences at Komsomolsk-on-Amur State Technical University, Russia in 2001. At present he is a head of the Scientific Work Organization and Researchers Training Administration of Bonch-Bruevich Saint-Petersburg State University of Telecommunications, Saint-Petersburg, Russia. His major interests include control systems, soft computing, communication networks, network security management.

Rapid Detection of Stego Images Based on Identifiable Features

Weiwei Pang*,**, Xiangyang Luo*,***, Jie Ren*,**, Chunfang Yang*,**, Fenlin Liu*,**

*State Key Laboratory of Mathematical Engineering and Advanced Computing, Zhengzhou 450001, China

**Zhengzhou Science and Technology Institute, Zhengzhou 450001, China

***Science and Technology on Information Assurance Laboratory, Beijing 100072, China

pangweiwei01@126.com, luoxy_ieu@sina.com, renjie@vip.126.com, chunfangyang@126.com, liufenlin@vip.sina.com

***Abstract*—An increasing number of images in the Internet brings forward a higher requirement on the speed of steganalysis. For the problem of real-time detection of stego images, a rapid images steganalysis method based on identifiable features is proposed, where the identifiable features are specific character sequences left in stego images by steganography tools. The stego and cover images are distinguished according to whether the identifiable features are found in the detected images. Meanwhile, for the case of that multiple identifiable features appeared on the same location of an image, the AC (Aho-Corasick) multi-features matching algorithm is applied to improve the detection speed. In experiments, the detection method is used to detect eight steganography tools such as Invisible Secrets, E-Show, BMP Secrets and so on. The results show that the proposed steganalysis method can achieve a nearly perfect detection precision, and the detection speed can be improved significantly comparing with traditional methods (matching bytes one by one).**

***Keyword*—Steganalysis, Identifiable features, Steganography tools, AC(Aho-Corasick) matching algorithm, Stego image detection.**

I. INTRODUCTION

Steganography is a covert communication technique to embed confidential message into the redundancy parts of multimedia files such as digital images, audios and videos, and then transfer the obtained stego objects through public communication channels [1]. Contrarily, steganalysis includes judging detected object is stego or cover, recognizing the steganography algorithm, estimating the length or location of secret message, cracking the embedded key and extracting the secrets message. The stego objects detection is especially important because which is the first step of steganalysis. Generally, steganography has been broken if an attacker can judge the detected object whether contains secret messages with a success better than random guessing. Compare to other media forms (audios, videos, etc), images are the most commonly covers used in steganography. So this paper mainly studies image steganalysis. As the rapid popularization of telephone and camera in recent years, the number of images appearing in the Internet is increasing dramatically. Data shows that about 10 million images are uploaded to social networks each hour [2]. Therefore, the fast and accurate detection of stego images from a large number of images is one of the most urgent practical problems to be resolved.

Currently, researches on detection of stego images can be divided into three classes: sensory detection, statistical feature detection and identifiable feature detection. Sensory detection, as an early detection algorithm, has been obsoleting since it is difficult to implemente automatically. Statistical feature detection is the research hotspot of steganalysis because of that most of steganographic algorithms can be detected reliably by this methodology. For example, Pevnỳ and Fridrich [3] extended the 23 DCT features set [4] to get a 274-dimensional feature vector, then used the new feature to construct a Support Vector Machine multi-classifier capable of assigning stego images to six popular steganographic algorithms: OutGuess [5], F5 [6], MB [7], etc.; Fridrich and Kodovsky extracted the 34671-dimensional SRM (Spatial Rich Model) [8] feature and 22510-dimensional CC-JRM (Cartesian Calibrated-Jpeg Rich Model) [9] feature from spatial images and jpeg images respectively to attack some algorithms successfully, such as HUGO (Highly Undetectable steGO) [10], LSB (Least Significant Bit) [11], EA (Edge Adaptive) [12], MME (Modified Matrix Encoding) [13], nsF5 (no-shrinkage F5) [14], etc. Denemark, Fridrich and Holub [15] put forward the novel concept of content-selective residual to increase the detection precision of S-UNIWARD. These steganalysis methods based on statistical features above can attack many steganographic algorithms reliably, but the dimensions of these statistical features are high, and detection speed is low, it is difficult to meet the requirement of real-time detection for a large number of images. Besides, steganalysis based on statistical features has a high false

Manuscript received June 17, 2015. This work is a follow up of the accepted conference paper as an outstanding paper for the 17th International Conference on Advanced Communication Technology.

This work was financially supported by the National Natural Science Foundation of China (No. 61379151, 61272489, 61302159, 61401512, 61373020 and 61572052), the Excellent Youth Foundation of Henan Province of China (No. 144100510001), and the Foundation of Science and Technology on Information Assurance Laboratory (No. KJ-14-108).

W. PANG is currently a M.S candidate at the State Key Laboratory of Mathematical Engineering and Advanced Computing, Zhengzhou, China (phone: +86 13027790289; e-mail: pangweiwei01@126.com) .

X. LUO, J. REN, C. YANG (corresponding author, phone: +86 13513891391) and F. LIU are with the State Key Laboratory of Mathematical Engineering and Advanced Computing, and Zhengzhou Science and Technology Institute, China (e-mail: luoxy_ieu@sina.com, e-mail:renjie@vip.126.com, chunfangyang@126.com, liufenlin@vip.sina.com). X. LUO also is a researcher at Science and Technology on Information Assurance Laboratory, Beijing, China.

detecting rate for lower embedding rate images.

Nowadays, the kinds of steganography tools are more than one thousand and some of them will leave identifiable features in stego images. The identifiable feature detection method recognizes these stego images through checking whether detected images contain these identifiable features. The missing rate of steganalysis based on identifiable features is 0 and the false detecting rate is low for lower embedding rate images [16]. Besides compare with the steganalysis based on statistical features, this method has a significant speed advantage. Bell and Lee [17] proposed a fast and accurate automatic detection method based on the characterized regularities in output media caused by weak implementations of some steganographic algorithms. Bell and Lee [17] have used the proposed method to detect 6 kinds of steganography tools such as Steganos, Inv.Sevrets, OutGuess, JSteg, STools and MP3Stego. Pevnў and Ker [18] used the length of message as dynamical identifiable feature of OutGuess to crack the stego key. In this method, every key in the key dictionary can be used to get a message length, if the message length extracted (can be seen as a dynamic identification feature) is more than the estimated embedding capacity of the stego image embedded by OutGuess, the key must be wrong and should be dropped from the key dictionary. Because messages length extracted from different stego images (the same stego key is used) by the same wrong key are very possibly different, if the stego images with the same stego key are enough, then keys in the key dictionary can be reduced to one or a few by exhaustion attacks. Because above methods do not consider how to reasonably organize the identifiable features, with the number of steganography tools increasing, the number of identifiable features must increases, and the detection time of the traditional detection (matching bytes one by one) will increase linearly. This would not meet the requirement of detecting stego images generated by many steganography tools from a large number of images.

In [19], we have briefly given a method to rapidly detect the identifiable features in stego images, and experimented with two steganography tools (A Plus the File Protection and 007 Electronic Stego Water). In this paper, above method will be supplied with more details and tested with eight steganography tools. According to the different areas of stego images where identifiable features locate, the proposed method divides identifiable features into head features, data features and tail features. Then, head feature table is constructed to detect the head area of images. Because of the speed advantage of AC multi-pattern matching algorithm [20] in multi-feature matching, a multi-pattern fuzzy matching machine of data features is constructed to detect the data area of images by AC multi-pattern fuzzy matching algorithm, and a multi-pattern exact matching machine of data features is constructed to detect the tail area of images by AC multi-pattern extract matching algorithm. Experimental results demonstrated the effectiveness of the detection method proposed in this paper, which could significantly improve the detection speed on the condition that the missing rate is zero and the false detection rate is very low. The problem that detection time increases linearly with the number of features increasing is relieved effectively.

II. Steganalysis Based on Identifiable Features Classification

Identifiable features are constant marks left in stego images by steganography tools to protect copyright or check whether images have been embedded, they usually present as specific characters sequences appeared in specific bits of stego images. Identifiable features exist in different positions of stego images have different forms. For example, the identifiable features located in the head of images usually are represented as abnormal properties values in head of stego images, and these property values often are less than two characters, so the image which will be detected is a stego or cover will be judged through comparing properties of images which will be detected with "abnormal properties" of stego images. The identifiable features located in data area of images are represented as LSB sequences consisted of pixels' LSBs of stego images usually, the detection speed will be fast if the method of detecting on data area of images based on multi-pattern fuzzy exact matching algorithm is used to detect data area of images. Similarly, the identifiable features located in tail area of images are represented as hexadecimal character sequences consisted of pixels of stego images usually, the detection speed will be fast if the method of detecting on tail area of images based on multi-pattern exact exact matching algorithm is used to detect data tail of images. For this reason, identifiable features are divided into head features, data features and tail features. Different identifiable features recognizing algorithms are used to recognize the three classes of features. Detection methods are described as follows.

A. Identifiable features classification

Head Features: some image properties (such as width, height, resolution, palette, etc.) may be falsified by some steganography tools when embedding. For example, the length of the file head of a BMP image will be increased 1 if the image is embedded by Imagehide [16]. The first reserved value of a BMP image will be changed to the message length if the image is embedded by E-Show. The resolution of a BMP image will be changed to 73*73 if the image is embedded by BMPSecret [16]. The resolution of a BMP image will be changed to 0 if the image is embedded by Invisible Secrets. In addition, there is a characters sequence consisted of 256 characters (1, 2, 3, ⋯ ,255 , 0) will be added in the palette redundancy of a BMP image if the image is embedded by Stegomagic1.0 [16].

TABLE I
BMPSECRET & E-SHOW HEAD FEATURE TABLE ITEMS

Tools	**Format**	**Size**	**Offset**	**Reserved_1**
BMPScerets	*BMP*	*-1*	*-1*	*-1*
E-Show	*BMP*	*-1*	*-1*	*Msg length*
DataSize	**Resolution**	**Width**	**Height**	**……**
-1	*73*73*	*-1*	*-1*	*-1*
-1	*-1*	*-1*	*-1*	*-1*

The head features are often expressed as abnormal properties values in the heads of stego images, and these properties values can be achieved easily through analyzing of image format. So head feature table is constructed for every head feature which existed in feature library. Then whether

the image head contains head features will be judged by comparing head feature table with file head of the detected images. Head feature table items of BMPSecret and E-Show are shown by TABLE I, where the normal properties of image are represented of *-1*.

Data features: data features are identifiable features left in the data area of image by a part of steganography tools, they often represent as specific character sequences existed in LSB or 2LSB, etc. Because images may be distorted if pixels or DCT coefficients are tampered to specific characters sequence, these sequences often located in LSB or MLSB (commonly under 4LSB). LSBs are the least bits of image's pixels, so a pixel is an odd number if LSB of the pixel is 0 and a pixel is an even number if LSB of the pixel is 1. That means LSBs of image's pixels is equivalent to the odd-even sequence of image's pixels. LSBs represent as characters sequences consisted of 0 and 1 in this paper. Similarly, 2LSBs represent as characters sequences consisted of {0, 1, 2, 3}. Data features are listed as follows, the sequence "0100 0011 0100 100" appear in the LSBs from 37 to 45 pixels in BMP stego images which have been embedded by A Plus File Protection [16], the sequence "0100 0010 0110 1001 0111 0010 0110 0100" appears in the LSBs from 36 to 55 pixels in BMP stego images which have been embedded by 007 Electronic Stego Water [16]. The sequence "1000 1110 1000 0111 1001 1111 1001 0001" appears in the LSBs and the sequence"1010 1000 1111 0100 0001 1010 1001 0000" appears in the 2LSBs from 0x36 to 0x56 pixels in BMP stego images which have been embedded by Inthepicture [16]. The probability that different features have the same prefix will be increasing as the number of identifiable features increasing. Character types of these sequences is no more than four (0 1 2 3), so it is more prone to have the same prefix.

Multiple features detection actually is an issue of multi-pattern matching. As a typical multi-pattern matching algorithm, AC multi-pattern matching algorithm has obvious speed advantage than other algorithms in multi-pattern matching. Therefore, a multi-pattern fuzzy matching algorithm based on AC multi-pattern matching algorithm is proposed and adopted to detect data area of images. The multi-pattern fuzzy matching algorithm is described in section III.

Tail features: some particular characters sequences consisted of pixels will be appended to the tail redundancy area in images by some tools. These features are tail features in this paper. For example, the character sequence "0x07 0x00 0x00 0x00" is appended to the last bit of a BMP or JPEG image if the image is embedded by Bulletproof vest [16]. Character "FF" is appended to the last bit of a BMP or JPEG image if the image is embedded by E-Show. The character sequence "0xCC 0x99 0xFF 0x66" is appended to the last bit of a BMP image if the image is embedded by safe & quick hide file 2002. Besides, the characters sequence "0x5B 0x3B 0x31 0x53 0x00" is appended to the last bit of a JPEG image if the image embedded by Jpegx [16].

Tail features will be represented as specific sequences consisted of hexadecimal character. Hexadecimal character has only 16 kinds of characters. So as similar to data features, a multi-pattern exact matching algorithm based on AC multi-pattern matching algorithm is proposed and adopted to detect the tail redundancy area of images. The algorithm is described in section III.

B. Stego images recognition based on identifiable features classification

As shown in Fig.1 and Fig.2, detection method proposed in this paper is divided into two phases: pre-processing phase and detection phase. Every identifiable feature in the feature library should to be pre-processed before detection. Different processing rules are set to different classes of identifiable features above. Three results of pre-processing (head features table, data features fuzzy matching machine and tail features exact matching machine) are achieved after pre-processing phase. Then the three pre-processing results are used to detect three areas (head area, data area and tail area) of image to get three detection results (R_1, R_2, R_3). The three detection results are used to get the final detection result R by RGUD (Results Judging based on United-Decision) algorithm. The final detection result R will tell you that the detected image is a stego image or cover image, besides, name of the steganography tool will be known by R if the detected image is a stego image.

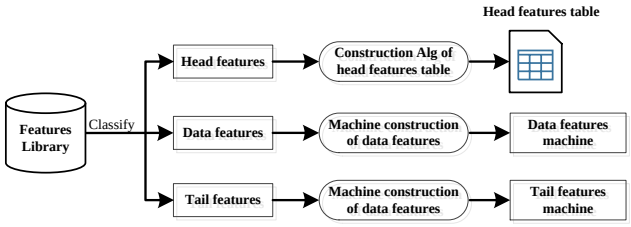


Fig. 1. Identifiable features pre-processing

As shown in Fig. 1, head feature table is constructed for every head feature existed in features library. If other properties (except steganography tools name and image format) have head features, then these head features will be added to head feature tables. The head feature table consisted of identifiable features of Invisible Secrets and E-Show is shown in TABLE I. Head detection is described as follows: properties' values of the detected image are achieved through analyzing the image format first, then decide whether abnormal values (head features) exist in these properties according to checking head features tables. If they exist, the image is a stego image. R_1 is true and tools name is extracted from the head feature table. For example, if the resolution of an image is 73*73, the image may be a stego image and which is embedded by BMPSecrets, if the first reserved value of an image is not 0, the image may be a stego image and which is embedded by E-Show. If all properties of the image have no any abnormal value, head data of the image is normal and R_1 is false. In view of the situation that a property may have more than one features (such as the resolution of BMP images is 0*0 or 73*73 if the image is embedded by Invisible Secrets or BMPSecrets) and the length of head features is short (1 or 2 bit), binary search algorithm is adopted to detect these properties to improve detection efficiency. For data area and tail redundancy area, these areas of images were detected by using matching machines constructed by data features or tail features in the identifiable features pre-processing stage. Then detection results R_2 and R_3 have been achieved. In the end,

united-decision algorithm was adopted to get the final detection result R. Details are introduced as follows.

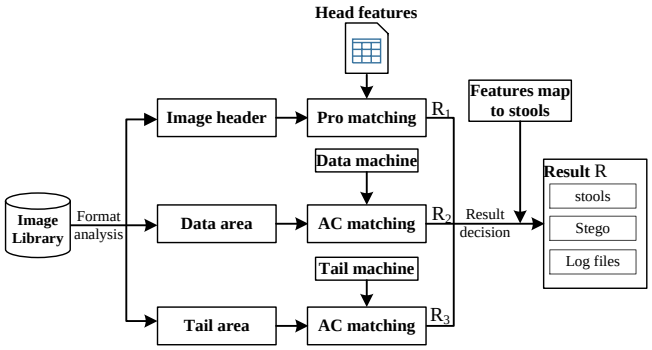


Fig. 2. Identifiable features rapid recognition

If different data features appear in the same location of LSBs or 2LSBs, a multi-pattern fuzzy matching machine consisted of parity sequences should be constructed. Similarly, if different tail features appear in the same location of image, a multi-pattern exact matching machine consisted of characters (0, 1, 2, 3) should be constructed. The algorithm of matching machine construction is described in section III.

Detection method is shown in Fig. 2. Through format analyzing of images, the detected image is divided into three sections: head, data and tail. Different detection algorithms are used to detect three sections of images by pre-processing results (head features tables, fuzzy matching machine and exact matching machine) for three classes of features above. Three detection results R_1, R_2, R_3 would be achieved after detection. United-decision algorithm is proposed to analyze these results, the final detection result R is achieved after decision.

C. Results judging based on united-decision

As shown in Fig. 2, the result set R_1, R_2, R_3 has been achieved from detection of image head area, data area and tail area above, then the three detection results will be judged to get the final result R by RGUD (Results Judging based on United-Decision) algorithm. The pseudo-code of RGUD is shown in TABLE II where T_1, T_2 and T_3 are names of steganography tools. If two or three results from the result set are true and steganography tools which results refers to are different, it is needed to judge the result set. The process is: when the three results are all false, the image detected can be judged as a cover; when three results are true, the image detected can be judged as a stego image, then extracting the name of steganography tool which the true result points to; if two of the results is true, the image is a stego image, if the two true results points to the same steganography tool, then the image is a stego image embedded by the steganography tool, If the two software name are inconsistent , the two types of tools are suspicious, extracting the two tools name and recording into the final result R, respectively; in a similar way, if the three results are true, the image is stego image, when the three results point to the same tool, judging that the image embedded by the tool. If there are two results in three results point to the same tool, the two software are suspicious tool, and should be included in the final detection result, the tool which the two results points to has a higher priority, If there is no any same name of tools which the three results points to, the three types of tools are suspicious and write into the final result R, there is no priority order. At this point, the detection method proposed in this paper is completed and the final result R is gotten.

TABLE II
PSEUDO-CODE OF RGUD ALGORITHM

Name: RGUD
Input: R_1, R_2, R_3
Output: R is stego or cover, stego tool names and probability

```
If R1, R2, R3 all are False
    R is a cover image
End
If one of R1, R2, R3 is True
    R is a stego image
    T1 is the tool with probability 100%
End
If two of R1, R2, R3 are True
    R is a stego image
End
    If two tools are the same
        T1 is the tool with probability 100%
    Else T1, T2 are the two tools with probability 50%, respectively.
    End
If R1, R2, R3 are all True
    R is a stego image
    If three tools are the same
        T1 is the same tool with probability 100%
    End
    If two tools of the three are the same
        T1 is the same tool with probability 67%, T2 is the other tool with
          probability 33%
    End
    If three tools all are different
        T1, T2, T3 are those tools with probability 33%, respectively
    End
End
```

III. IDENTIFIABLE FEATURES DETECTION BASED ON MULTI-PATTERN MATCHING

Multi-pattern matching is an algorithm which can finish detecting in a matching process [20]. AC, WM (Wu-Manber) [21] and SBOM (Set Backward Oracle Match) [22] are typical multi-pattern matching algorithms in the field of intrusion detection. The character type of identifiable feature is less than 16 and there are no bad characters (characters which present in image data but not exist in features) in image data, so the advantage of WM that increasing jump step by bad characters is not reflected. Besides, Chen Xiao-jun argues that memory access time of WM and SBOM is longer than AC in paper [23], and the advantage of memory access time of AC is especially obvious when the number of patterns is large. Taken together, AC algorithm has higher detection efficiency than WM for detecting data area and tail redundancy area, and the efficiency is higher with the number of patterns increasing. In addition, considering the position of identifiable features is relatively fixed, the concept of position verification is added into AC algorithm in order to further reduce the false detecting rate in this paper.

The algorithm includes two parts. The former is

pre-processing phase, a finite state feature matching machine should be constructed of all identifiable features which existed in data area or tail redundancy area; the latter is detection phase, matching machines constructed above will be used to detect image data area or tail redundancy area.

Pre-processing phase: Matching machine constructed process is shown in Fig. 1. There are three functions Goto function, failure function and output function should be constructed. Goto function stands for turning to the next state, and continuing to match until to the situation that input characters and feature's characters matching successfully, which was indicated by solid arrows in Fig. 3. Node numbers in Fig. 3 were set in the order that feature's ID smaller first between different features and left character is first in the same feature. Failure function stands for which state should be jumped to when input character is not equals to features' character, which was indicated by dotted arrows in Fig. 3. Failure function is a backtracking process and which reduce access times of the same prefix characters from n (the number of features) to one when n features have the same prefix characters. For example, the three features {1011, 1110, 1100} have the same prefix "1", then the first character "1" just need be accessed one time during the whole searching process. Output function stands for outputting an identifiable feature when the feature and image data detected matching successfully, which was indicated by states {4, 7, 9} in Fig. 3. It means the feature exist in image data, then go to the position verification stage. If the position verify successfully, the image is judged as a stego image. Using identifiable features in BMP image data area as an example to illustrate the process, if features exist in least significant bit of BMP image, then they are represented as the sequence consisted of zeros and ones. For example, there are three data features "odd, even, odd, odd", "odd, odd, odd, even", "odd, odd, even, even" in the same pixels, where the "odd" and "even" are mean that the pixel is an odd or even. For the convenience of expressing, "odd" and "even" are expressed with "1" and "0", respectively. That means that the three data features can be expressed with {1011, 1110, 1100}, and LSBs of pixels where these features located is {00101110010}, the process of matching machine construction is shown as step 1) to 3).

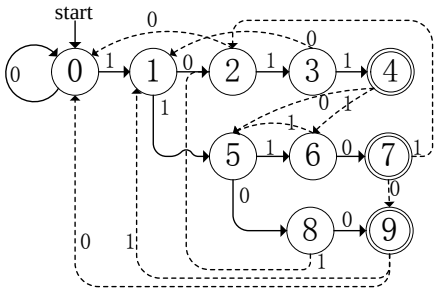


Fig. 3. The fuzzy matching machine constructed of data features

1). Goto function is consisted of characters which in the features set {1011, 1110, 1100}. Prefix relationships between the three features were described by the function consisted of ten directed edges and ten nodes (can be called states). Prefix relationships between the three features has decreased the number of states from 12 to 10. So matching times would be reduced and detection efficiency would be improved. Which are shown as solid arrows in Fig. 3.

2). Failure function maps a state into another [18]. The function's main aim is to look for which state jump to whenever Goto function reports fail. The disadvantage of backtracking in BF (Brute Force) algorithm is eliminated in the failure function. Construction process of failure function was described as follows: first, failure values of all states s which depth is 1 were initialized to 0. Then failure values of other states were computed in order of depth-first. It means failure values of states which depth is d should be concluded by failure values of states r which depth is $d-1$. As shown in formula (1):

$$f(s)=\begin{cases}0 & d=1\\ f(s') & d>1\end{cases} \tag{1}$$

where $f(s')$ is an iterative process of $f(s)$, detailed steps are described as follows: ①Set $state=f(r)$, where r is the direct precursor of s; ②according to the Goto function to calculate the value of $g(state,a)$, where a is an input character, if the value is null, then this step is executed iteratively many times until the value is not null, then the non-null value is the value $f(s)$ maps to. The non-null value must exist in Goto function because $g(0,0)=0$, $g(0,1)=1$ and input character is only 0 or 1; ③certain states are designated as output states which indicate that a set of features.

Example:

Initializing state which depth is 1, set $f(1)=0$;

Calculating the failure values of states {2, 5} which depth are 2 utilizing the failure values of states which depth are 1:

set $state=f(1)=0$, set $f(2)=0$ as $g(0,0)=0$;

set $state=f(1)=0$, set $f(5)=1$ as $g(0,1)=1$;

Calculating the failure values of states {3, 6, 8} which depth are 3 utilizing the failure values of states which depth are 2:

set $state=f(2)=0$, set $f(3)=1$ as $g(0,1)=1$;

set $state=f(5)=1$, set $f(6)=5$ as $g(1,1)=5$;

set $state=f(5)=1$, set $f(8)=2$ as $g(1,0)=2$;

As above, calculating the failure values of states {4, 7, 9} which depth are 4 utilizing the failure values of states which depth are 3;

set $state=f(3)=1$, set $f(4)=5$ as $g(1,1)=5$;

set $state=f(6)=5$, set $f(7)=8$ as $g(5,0)=8$;

set $state=f(8)=2$, set $f(9)=0$ as $g(2,0)=0$,

$state=f(2)=0$ and $f(0,0)=0$.

i	1	2	3	4	5	6	7	8	9
$f(i)$	0	0	1	5	1	5	8	2	0

(2)

3). To improve the detection accuracy, the property *firstCharIndex* (the index of feature's first character in stego images) is added to struct of output features to check whether features is located in the specific location of stego image. The struct of node is shown in TABLE III. Output states are shown as double loop nodes in Fig. 3.

TABLE III
STRUCT OF MATCHING MACHINE OUTPUT NODES

Struct **outpatstruct**; // struct name

Char ***opat****[PATLEN]*; // patterns content
long ***firstCharIndex***; // bit number in stego images where the first character of patterns located.
int ***patternIndex***; // patterns' ID (unique)
struct outpatstruct ****next***; // point to the next pattern

The matching machine consisted of features set {1011, 1110, 1100} has been constructed now, detection phase is beginning. The fuzzy matching machine is constructed to detect the data area of image.

Similarly, the exact matching machine is constructed to detect the tail area of images. The process of exact matching machine constructing is similar to the process of fuzzy matching machine. For example, there are three tail features in the last some bits of tail area and they are {0x00 0x00 0xFF, 0x07 0xEF 0x0A, 0x00 0x00 0x3C}, the last some bits of tail area are {0x0E 0x00 0x00 0x3C 0xEF 0xFF}, then the exact matching machine is show in Fig.4 where Ω is any hexadecimal character.

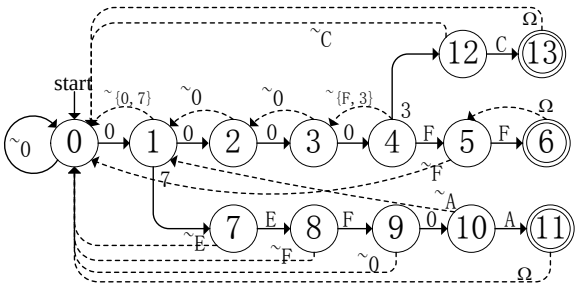


Fig. 4. The exact matching machine constructed of tail features

Detection phase: According to a window sliding on a characters sequence consisted of image data bytes to searching for features by the matching machine constructed above. An identifiable feature will be outputted when the feature is found in images. Then check whether the value of *firstCharIndex* is the index of location where the feature located. If so, detection results R_2 or R_3 is true and the feature code will be outputted. If not, then R_2 or R_3 is false. Detection of data area and tail area completed now. The detection result set R_1, R_2, R_3 are achieved when detection of the whole image has been completed, then united-decision algorithm is adopted to get the final detection result R.

IV. EXPERIMENTS

Detection precision includes two sides: undetected rate and false rate. As AC, WM and BF are precise matching algorithms, so the undetected rate is determined by identifiable features are accurate or not. If identifiable feature is correct, then undetected rates of three algorithms are zero. If identifiable features are can't distinguish stego images from cover images, then undetected rates are not reliable yet. Besides, the false rate is determined by identifiable features are integrity or not. The false rate will be increase if identifiable features are not integrity. Besides, the false rate is affected by the length of identifiable feature. For example, characters "0xFF" are appended to the last bit of BMP or JPEG images by E-Show, and the probability of that the cover images have the same last bytes as stego images is $1/32$. So the false rate of E-Show is $1/32$ under the condition that the detected image has tail redundancy bytes. For the steganography tool "007 Electronic Stego Water", the length of identifiable feature is 32, and the probability that images have the same LSBs of the identifiable feature is $1/2^{32}$, so the false rate is $1/2^{32}$.

To validate the accuracy and rapidity of the detection method proposed in this paper, experiments are designed as follows. They include three parts: detection of image head, detection of image data area and detection of image tail redundancy.

Hardware environment: the experimental environment is 64-bit Windows 7 Operating System, Pentium(R) P6200 (2.13 GHz) CPU, 3.67 GB RAM, and development environment is Microsoft Visual Studio 2010. Note that the level of hardware performance and the busy degree of CPU all might make the detection time float on a small range, therefore, in order to ensure the precision of the detection time, the same PC is used to test AC, WM and BF algorithms at the same time in detection.

Construction of images library: 10000 PGM gray-scale images of BOSSBase-1.01 database [24] are transformed into BMP images, the resolution and size of these images are 512*512 and 257 KB uniformly. 80 BMP images selected randomly are used to generate stego images. First, they are divided into 8 groups randomly, every group include ten images. Then eight groups of images are embedded by 8 steganography tools separately. Eight steganography tools are Invisible Secrets, E-Show, BMPSecret, A Plus File Protection, 007 Electronic Stego Water, Encryption Excellent Soldier, Small Encryption Lock and Bulletproof Vest. Secret message is a random length (less than embedding capacity of cover images) of text document (txt). These 80 stego images were put into other 9920 cover images. Then the test images library has been constructed.

TABLE IV
TABLE OF STEGANOGRAPHY TOOLS INDENTIFIABLE FEATURES

Tools	Area	Location	Features
Invisible Secrets	Head	Resolution	0*0
E-Show	Head & Tail	Reserved _1 Last character	Msg length "0xFF"
BMPSecret	Head	Resolution	73*73
A Plus File Protection	Data	0x36~0x45 LSBs	0100 0011 0100 100
007 Electronic Stego Water	Data	0x36~0x55 LSBs	0100 ... 0100
Encry Excellent Soldier	Tail	First 24 characters of tail redundancy	0x21 0x3F ... 0x3F 0x21
Small Encryption Lock	Tail	Last of tail redundancy	0x3C 0x3C ... 0x3C 0x3C
Bulletproof Vest	Tail	Last 4 characters of tail redundancy	0x07 0x00 0x00 0x00

A. Detection of image head area

As shown in TABLE IV, three steganography tools which identifiable features in image head are Invisible Secrets, E-Show and BMPSecret. If the detected image is a BMP image, then the resolution and reserved values are extracted by format analysis. When the resolution value is 0*0 or 73*73,

the image may be a stego image and is embedded by Invisible Secrets or BMPSeret. When the resolution value is not zero and less than embedding capacity (because steganographic algorithm of E-Show is LSB replacement, the embedding capacity can be replaced of image size/8) of the image, the image may be a stego image and is embedded by E-Show. Experimental result is described as TABLE V.

TABLE V
TABLE OF HEAD DETECTION RESULT

	Invisible Secrets	E-Show	BMPSecret
Stego number	9990	10	10
Undetected rate	0%	0%	0%
False rate	99.8%	0%	0%

Detection precision includes two sides: undetected rate and false rate. From detection result above, it can be seen that undetected rates of three tools are zero. But the false rate of Invisible Secrets is 99.8%, because of identifiable feature of the tool is can't distinguish stego images from cover images, then undetected rates are not reliable yet. No correlation with the detection method proposed in this paper. It can be proven that the false rate of BMPSecret is 0%. The time of image head image detection is about 496ms.

B. Detection of image data area

Construction of features set: characters sequences "0100 0011 0100 100" and "0100 0010 0110 1001 0100 0010 0110 0100" which generated by A Plus the File Protection and 007 Electronic Stego Water, respectively. Besides, since the current identification features which we have are deficiency, to test the influence on detection speed with the number of features increasing, we have constructed a generator to generate virtual identification features code. About 500 virtual identification features consisted of random characters {0, 1} were constructed by the generator, the length of these virtual identification features is 17 bits.

Detection objects: the LSBs of pixels which before the 56 (hex) bytes in BMP images, a total of 87 bits characters sequences consisted of zeros or ones.

Detection algorithms: AC, WM and BF.

Accuracy verification: experimental results show that 20 stego images can be accurately identified by three detection algorithms, undetected rate and false rate are 0. The recognition algorithm of identifiable features proposed in this paper is a precise matching algorithm. There is no undetected case if identifiable features are correct and complete. So undetected probability can keep zero for any embedding rate. For pixels' LSB in data area, the probability of cover images and stego images having the same characters is $1/2^n$, where n is the number of features bits. So false rate is less than $1/2^n$. In the experiment, n is more than 15, so the false rate is less than 0.003% and close to 0. Besides, the results show that the undetected probability and the false rate of three detection algorithms (AC, WM and BF) are 0. On the contrary, steganalysis based on statistical features is difficult to get the higher detection precision when the embedding rate is less than 1%. So steganalysis based on identifiable features is reliable for lower embedding rate images.

Rapidity verification: the algorithm of identifiable features recognized based on AC proposed in this paper has a speed advantage to other algorithms, and result is shown as Fig. 5 and TABLE VI.

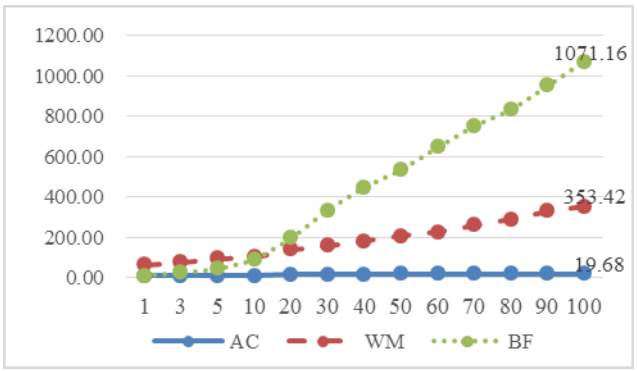


Fig. 5. Net detection time for three algorithms

Fig. 5 abscissa is the number of identifiable features and which ordinate is net detection time (ms). The net detection time only refers to the matching time, does not include the time of reading image and outputting result. Namely, it is just refers to the time of matching stage, does not include pre-processing for AC and WM. The time of pre-processing can be ignored when a large of detected images. In a word, the detection time is only impacted by the number of identifiable features and the size of detected images.

The first column of TABLE VI is the number of features. Detection result of three algorithms (AC, WM, BF) shows that the undetected probability and the false rate are 0, the detection accuracy of the detection method proposed in this paper is still keep 100% for the lower embedding rate images. As can be seen from the Fig. 5 and TABLE VI, detection time of the detection algorithm based on AC keeps steady with identifiable features increasing. While detection time of the other two algorithms is increasing linearly. So detection algorithm based on AC proposed in this paper has a higher detection speed.

TABLE VI
TABLE OF NET DETECTION TIME

Features number	Detection net time (ms)			Time rate	
	AC	WM	BF	AC/WM	AC/BF
100	19.68	353.42	1071.16	5.57%	1.84%
200	21.40	680.37	2075.72	3.15%	1.03%
300	23.72	1032.57	3199.38	2.30%	0.74%
400	24.15	1389.35	4289.33	1.74%	0.56%
500	24.44	1692.50	5488.26	0.91%	0.45%

C. Detection of image tail area

TABLE VII
TABLE OF FINAL DETECTION RESULT

	E-Show	Soldier	Lock	Vest
Stego number	10	10	10	10
Undetected rate	0%	0%	0%	0%
False rate	0%	0%	0%	0%

As shown in TABLE IV, four steganography tools which identifiable features in image tail redundancy are E-Show, Encryption Excellent Soldier, Small Encryption Lock and Bulletproof Vest. If the detected image has tail redundancy bytes, then detection of image tail redundancy is started. If tail redundancy bytes of images have one identifiable feature,

then the image may be stego image. Then combine the detection result with head and data area detection results, the final detection result is achieved by decision of detection result. The final detection result is shown as TABLE VII.

Detection results show that there are forty images have tail redundancy bytes. It means that cover images and stego images can be divided just by checking images have tail redundancy bytes or not. Possibly because cover images used are transformed from BOSSBase-1.01 database and they are unified. Which steganography tool used for every stego image can be obtained by recognizing identifiable features.

V. Conclusions

A method of steganalysis of image based on identifiable features left by steganography tools is proposed in this paper. The method can detect reliably the lower embedding rate images. To solve the problem that many identifiable features appearing in the same place, an algorithm of identifiable features recognized rapidly based on AC is proposed. Experimental results show that the algorithm improve effectively the detection speed. However, this algorithm applies to some steganography tools which identifiable features have been achieved and can't work well in the case of tools which identifiable features have not been achieved. So, extraction of identifiable features of steganography tools will be studied in further research.

References

[1] J. Lu, F. Liu, and X. Luo, "Recognizing F5-like stego images from multi-class JPEG stego images," *KSII Transactions on Internet and Information Systems*, vol. 8, no. 11, pp. 153-169, 2014.

[2] http:// www.computer.org/portal/web/tetc/

[3] T. Pevný and J. Fridrich, "Merging Markov and DCT features for multi-class JPEG steganalysis." in *Proceedings of SPIE Electronic Imaging*, vol. 6505, pp. 3 1-3 14, 2007.

[4] J Fridrich. "Feature-based steganalysis for JPEG images and its implications for future design of steganographic schemes". in *Proceedings of 6th International Workshop on Information Hiding*, 2005: 67-81.

[5] N. Provos, "Defending Against Statistical Steganalysis." *Usenix Security Symposium*, vol. 10, pp. 323-336, 2001.

[6] A. Westfeld, "High capacity despite better steganalysis (F5–a steganographic algorithm)." in *Proceedings of 4th International Workshop on Information Hiding*, vol. 2137, pp. 289-302, 2001.

[7] P. Sallee, "Model-based steganography." in *Proceedings of 2nd International workshop on Digital water-marking*, vol. 2939, pp.154-167, 2004.

[8] J. Fridrich, and J. Kodovsky, "Rich models for steganalysis of digital images." *IEEE Transactions on Information Forensics and Security*, vol. 7, no. 3, pp. 868-882, 2012.

[9] J. Kodovský and J. Fridrich, "Steganalysis of JPEG images using rich models." in *Proceedings of SPIE, Electronic Imaging, Media Watermarking, Security, and Forensics*, vol. 8303, pp. 0A 1-13, 2012.

[10] T. Pevný and P. Bas, "Using high-dimensional image models to perform highly undetectable steganography." in *Proceedings of 12th International Workshop on Information Hiding*, vol. 6387, pp. 161-177, 2010.

[11] C. Kurak and J. McHugh, "A cautionary note on image downgrading." in *Proceedings of the Computer Security Applications*, pp. 153-159, 1992.

[12] W. Luo, F. Huang, and J. Huang, "Edge adaptive image steganography based on LSB matching revisited." *IEEE Transactions on Information Forensics and Security*, vol. 5, no. 2, pp. 201-214, 2010.

[13] Y. Kim, Z. Duric, and D. Richards, "Modified matrix encoding technique for minimal distortion steganography." in *Proceedings of 8th International Workshop on Information Hiding*, vol. 4437, pp. 314-327, 2007.

[14] J. Fridrich, T. Pevný, and J. Kodovský. "Statistically undetectable jpeg steganography: dead ends challenges, and opportunities." in *Proceedings of the ACM 9th workshop on Multimedia & security*, pp. 3-14, 2007.

[15] T. Denemark, J. Fridrich, and V. Holub, "Further study on the security of S-UNIWARD." in *Proceedings of SPIE, Electronic Imaging, Media watermarking, Security, and Forensics*, vol. 9028, pp. 04 1-12, 2014.

[16] G. Ren, "Analysis & Attack of the popular network steganography software." *Zhengzhou Information Science and Technology Institute in Chinese*, 2009.

[17] G. Bell and Y. Lee, "A method for automatic identification of signatures of steganography software." *IEEE Transactions on Information Forensics and Security*, vol. 5, no. 2, pp.354-358, 2010.

[18] T. Pevný, A. Ker. "Steganographic key leakage through payload metadata." in *Proceedings of the 2nd ACM workshop on Information hiding and multimedia security*, pp.109-114, 2014.

[19] W. Pang, X. Luo, J. Ren, C. Yang, & F. Liu. Rapid detection of stego images based on identifiable features. in *Proceeding of the IEEE 17th International Conference on Advanced Communication Technology, ICACT 2015*, pp. 472-477, 2015.

[20] N. P. Tran and M. Lee, "High performance string matching for security applications." in *Proceedings of the International Conference on ICT for Smart Society (ICISS)*, pp. 1-5, 2013.

[21] D. Agrawal and A. EI Abbadi, "An efficient and fault-tolerant solution for distributed mutual exclusion." *ACM Transactions on Computer Systems (TOCS)*, vol. 9, no. 1, pp. 1-20, 1991.

[22] V. Rana, G. Singh, "MBSOM: An agent based semantic ontology matching technique." in *Proceedings of Futuristic Trends on Computational Analysis and Knowledge Management (ABLAZE)*, pp.267-271, 2015.

[23] X. Chen, Z. Zhang, and Y. Liu, "Charactering memory access behavior of large scale multi-string matching algorithms." *Computer Engineering and Applications*, vol. 43, no. 26, pp. 106-109, 2007.

[24] http://boss.gipsa-lab.grenobleinp.fr/

Weiwei Pang was born in Henan Province, China, in 1989. Pang got the B.S. degree from Zhengzhou University, Zhengzhou, China, in 2013. He is currently a M.S candidate in the State Key Laboratory of Mathematical Engineering and Advanced Computing at Zhengzhou Science and Technology Institute. His current research interest is in image steganography and steganalysis technique.

Xiangyang Luo was born in Hubei Province, China, 1978. Luo received the B.S. degree, the M.S. degree and the Ph.D. degree from Zhengzhou Science and Technology Institute, Zhengzhou, China, in 2001, 2004 and 2010, respectively. He is now a researcher at Science and Technology on Information Assurance Laboratory. He is the author or co-author of more than 70 refereed international journal and conference papers. He is also a guest editor for "International Journal of Internet" and "Multimedia Tools and Applications". His current research interests include Networking and Information Security.

Rana V, Singh G. MBSOM: An agent based semantic ontology matching technique[C]//Futuristic Trends on Computational Analysis and Knowledge Management (ABLAZE), 2015 International Conference on. IEEE, 2015: 267-271.

Jie Ren was born in Anhui Province, China, 1977. He received the B.S.and M.S degrees from the Zhengzhou Science and Technology Institute in 1999 and 2007, respectively. Currently, he is now a researcher at Science and Technology on Information Assurance Laboratory. His current research interest is Information Security.

Chunfang Yang was born in Fujian Province, China, 1983. He received the B.S., M.S., and Ph.D. degrees from the Zhengzhou Science and Technology Institute in 2005, 2008, and 2012, respectively. Currently, he is now a researcher at Science and Technology on Information Assurance Laboratory. His current research interests include image steganography and steganalysis technique.

Fenlin Liu was born in in Jiangsu Province, China, 1964. He received his B.S. from Zhengzhou Institute of Science and Technology in 1986, M.S. from Harbin Institute of Technology in 1992, and Ph.D. from the Eastnorth University in 1998. Now, he is a professor of Zhengzhou Institute of Science and Technology. His current research interests include Networking and Information Security.

An Innovative Tour Recommendation System for Tourists in Japan

Quang Thai LE*, Davar PISHVA**

*Faculty of International Management, Ritsumeikan Asia Pacific University, Beppu, Japan

** Faculty of Asia Pacific Studies, Ritsumeikan Asia Pacific University, Beppu, Japan

quanle11@apu.ac.jp, dpishva@apu.ac.jp

***Abstract*[1]— The paper demonstrates prototype of system that is capable of suggesting optimal touring plans which are composed of various points of interest (POI) and take travelers' preferences and context into account. It systematically collects and analyzes information on thousands of tourists attraction areas and geographical nodes of Japan Railway (JR) train stations together with concurrent weather information, estimated travel time, associated expenses, and lists of multiple cultural events in order to demonstrate practicality as well as reliability of the system. A programmatic approach based on the heuristic greedy search is employed for transforming the obtained data into informative routes. It demonstrates the feasibility of the approach through its mobile prototype on web platform and tests it under various scenarios in eight different places in Japan which includes Tokyo, Osaka, Kyoto, Kobe, Yokohama, Nagoya, Fukuoka and Sapporo. Its result and the performance can be considered as a stepping stone towards a more localized and practical recommendation system in the field of tourism in the near future.**

Keywords— e-tourism, travel planning system, web scraping, modeling, and data mining.

I. INTRODUCTION

Olympic Games usually provide economic stimulus and since Japan won the bid to host the 2020 Summer Olympic Games in September 2014, various think-tank groups have announced their estimates for its economic impacts. The Olympic 2020 will, therefore, be imposing both challenges and opportunities to the current stream of globalization in the country [1]. According to Japan National Tourist Organization (JNTO), 12 million tourists visited Japan in the year 2014, but the number is expected to increase to 20 million by 2020 and as many as 8.5 million foreign tourists are projected to visit the country just during the Olympic [2]. Hence, the next 5 years will be a good opportunity for Japan to leverage its "world class" brand in tourism and hopefully become a tourism-oriented country in the near future [3]. Similarly, in April 2014, Japanese Ministry of Education decided to provide 37 selected universities in Japan with an annual subsidy of about $3.6 million each for the next 10 years so as to support them to become the so-called top global universities [4]. This is just one of the many efforts used for attracting international professors, researchers, students and cooperating with other prestigious institutions around the world, welcoming about 300,000 talented foreign students to visit and study in Japan until 2020 [5]. Despite its academic nature, possible tourism dimension of such huge number of foreign consumers is undeniably an open issue for the tourism industry.

Another challenge facing Japan is the sledding down of its world's economic ranking. The government believes that through artificial inflation (i.e., devaluation of its currency), not only Japanese products could become more completive in the world market, but also the country as a whole could become a more attractive tourist destination. This idea seems to be working since a 40% depreciation of Japanese currency during the past three years has increased both Japanese products export as well as the number of tourists visited the country. In fact for the first time during the past 45 years, the number tourists that visited Japan during the six months period surpassed the number Japanese tourists who visited other countries (OBS News, July 2015). Hence, the 2020 Summer Olympic is not just a precious opportunity for the country to leverage the domestic tourism to recover its economy, but also a momentum for the nation to strengthen its image of a globalizing country.

As an effort to bridge the gap between academia and empirical application in the field of tourism, our paper proposes a practical recommendation system for foreign tourists in Japan. It initially shares some background information regarding characteristics of Japan tourism and previous related works and then addresses the need of an innovative touristic recommendation system for Japan. Section 3 provides a detailed explanation on the methodologies employed in the system, including a brief introduction to the system's architecture, its data retrieval technique, the employed heuristic cluster-first-route-second approach, and demonstration of a web-based prototype application which has been developed to test our approach. Section 4 shows the results obtained on various traveling scenarios across Japan and compares them with results of other approaches. Section 5 points out current limitations and many promising future directions along this research. Section 6 subsequently summarizes and concludes the paper.

II. LITERATURE REVIEWS

A. Characteristics of Tourism Activities in Japan

Japan tourism has seasonal flavor and different marketing strategies are used during each of the four seasons. For example, Japanese Railway (JR) Company issues

Manuscript received on May 8, 2015. This work is a follow-up of the invited journal to the accepted conference paper of the 17th International Conference on Advanced Communication Technology.

Thai Quang LE is a student at Ritsumeikan Asia Pacific University (APU) in the field of Business Administration focusing on Strategic Management and candidate for graduation.

Davar Pishva is a professor in ICT at Ritsumeikan Asia Pacific University (APU) Japan (corresponding author phone: +81-977-78-1261, fax: +81-977-78- 1261, e-mail:dpishva@apu.ac.jp).

promotional free train tickets during a certain period in spring, summer or winter (e.g., Seishun 18 ticket) [6]. The country is also famous for its numerous annual cultural events. No wonder Japan was chosen as one of the World's top tourist destinations in 2011 by the CNN [7], and as the world's ninth most tourist-friendly country [8]. Its transportation infrastructure consists of modern buses and highly efficient train system equipped with rapid-transit railway network that link thousands of stations located throughout the country [6]. Tokyo is by far the most popular destination in Japan, accounting for 57.4% of foreign visitors [9].

Even though touring Japan seems to connote economic affluence, the recent depreciation of the Japanese Yen has made it possible for people from emerging countries to come and experience the Japanese life [9]. Despite continuous efforts to cope with the increasing flow of foreign tourists, there are still some major problems from the foreign tourists' perspectives in Japan. Specifically, in 2011, it is reported that public transport related issues (route details, usage and fees), lack of public LAN services, difficulties in communication are the top problems, accounting for about 45%, 37%, and around 25% respectively among all of the foreign correspondents [10]. Furthermore, the task of selecting a tourism destination in Japan is very important [11], and it requires "comprehensive information" so as to be "comfortable" [12]. This is because; there is still a lack of communication means in order to effectively bridge the tourists with the domestic tourism assets. Hence, the exploration of a collective delivery of tourism related knowledge to the right target audience under a particular context is an important practical research topic. Therefore, not only infrastructure metrics such as international airport capacity, wireless network or, foreign language signs availability, etc., but also intangible services such as multilingual promotion websites, tour-guide profession, etc. have to be further enhanced [13].

B. Previous Works on Tourism Recommendation System and Research Objectives

1) A Brief Literature Review

Previous approaches in implementing digital recommender system (RS) for the tourism industry mainly focused on providing more specialized products to customers. For example, in 2014, Damianos et al. did an intensive review on this topic with 19 different touristic recommendation systems, discovering that more and more researches take into account tourists' contextual information in the recommendation algorithm such as "attracts already visited", "user mobility pattern", "transportation mode", etc. [14]. The review also suggested consideration of weather conditions, more practical route and flexible usage of transportation as prospect future research directions [14]. Moreover, as clearly stated in the paper, most of the recommendation in tourism provides functions mainly either for suggesting tourism attractions or tourism services (e.g., restaurants, hotels, etc.), and there is a lack of systems that can combine these two [14]. In the same year, Ricardo, Lino, Ana, and Paulo introduced a system called PSiS. PSiS Mobile is a sightseeing tours recommendation system also takes the concurrent weather conditions as a real-time constraint [15]. The system also recognizes an interesting implementation problem when certain places are filtered out according to their operation time schedule, even though sometimes the architectonic outer appearance is sometimes the selling point [15]. Moreover, the system also emphasized device context (e.g., battery, connectivity) and dynamic context (e.g., real-time adaptation) [15]. In another research, Chiang and Huang (2015) made use of the user interface to continuously receive feedback from the users, as well as facilitate them to adjust any unsatisfied recommended results [16].

Regarding recommendation algorithms, most of the existing studies in the field has employed similarity or collaborative filtering (CF) to achieve the task. This probably results from the ability to match the user's preferences from previous cases, with the assumption that users who are similar are likely to have the same POIs [17]. To illustrate, Gulcin and Buse (2011) introduced a generic intelligent system applying similarity algorithm to find and match customers' preferences to previous cases that shared common attributes the most [18]. The paper also summarized a list of factors that are influential to decision-making (e.g., travel budget, local knowledge, hobbies, gender, age, etc.), of which "origins of customers" plays an important role [18]. Even though the CF algorithm was proven to draw good results on the POIs ranking tasks for new users, the fact that it requires historical data of actual cases with travelers' profile in details, disprove its practicality in the early stage (e.g., "cold start" problem). Kai, Huagang, Peng, Nenghai proposed the use of geo-tagged textual information of photos of various attractions retrieved from Panoramio with CF to recommend POIs to new users. Even though the research claims that its approach can generally tackle the "cold start" problem of CF, it does not consider factors such as users' sex, nationality, gender, etc., which are claimed to be very important in the recommendation process [18]. More importantly, since CF algorithm usually ends up only with a list of places that are highly attractive to a certain user's profile, the nature of the algorithm does not consider the geographical influence, or a whole route planning with logistics consideration, which is claimed to be very essential for a realistic system [17]. Hence, some previous researches with solely CF implementation failed to demonstrate their system's practicality.

Data sources are indispensable in any recommendation research. Even though there have been a number of previous researches investigating recommendation systems in tourism, there is yet any commonplace regarding metadata (e.g., geographical coordinates), sources, and type of input data. This might result from not only whether a list of POIs or whole itinerary is being achieved by the system, but also which algorithms are being employed. Particularly, some researches make use of collaborative filtering technique on previous tourism cases collected from a local tourism agency of a specific city to suggest a touristic planning for the user, while others make use of available data sources such as public photos from Flicker [17], etc., to heuristically picture different cases of sightseeing trips at some famous destinations around the world, which are then can also be fed into the CF algorithm. As for travelers' preferences, they are usually extracted from social network platforms such as foursquare or OpenSocial API [14].

System architecture of previous system mainly employs formal mobile application system, comprising of user preferences or queries input module, core algorithmic recommendation engine module, customized recommendations module, and a presentation module (visualized maps, GIS, etc.) [14]. There are just a few pieces of research which emphasize detail technical concerns regarding the performance of the implementation platform

such as network connectivity (e.g., Wi-Fi network), or battery consumption. Real-time feature, further enhancing the recommendation system with the flexibility to alter the suggested tour trip according to changes in user's context or the environment (e.g., user's current location, unexpected breaks, weather changes, attraction closure, etc.), seems to be quite unique and realistic module.

2) A Need of Localization of Tourism Recommendation System

Recommendation system in tourism is very different from other fields such as movies, or music recommendations. Besides the fact that user's preferences in the case of tourism recommendation are much harder to capture because of a big gap in the activity frequency, context is also another perspective that distinguish the two cases. Particularly, recommended objects such as films, music, news, etc. are very generic to any user's situation or preferences; hence the recommendation process can work well with solely user's preferences and object's contents. In contrast, the practicality of recommendation system depends much on the context, or the environment such as the country destination, regional regularity, public transportation, or seasonal factors. The variety of input data spaces of previous researches can further support this statement. Many researchers, even though tend to build a generic recommendation system, exploit data sources that seem to be only available from certain resources (e.g., historical cases from local tourism agency) or for a certain regions (e.g., public sightseeing photos are only sufficient to famous places). The disjoint in the data input or data sources is possibly one of the reasons for the current lack of evaluation tests for this kind of system, which usually requires formal field studies [14]. Nevertheless, most of the previous system concentrates more on the theoretical approach of the recommending engine, making the applicable scope become too general that unexpectedly reduce the practicality of actual implementation. Thus, we call for a demand of touristic recommendation systems that are more localized, or tailored to a specific regional area such as city, or country. Only by shifting the focus to this direction, we can ensure that future research on recommendation system would be able to not only share and reuse different resources (user's preferences, data input space, test cases, etc.) but also better bridge tourists with regional tourism promotional activities and policies.

3) An Innovative Recommendation System for Foreign Tourists in Japan

Our research investigates four out of seven new prospects in tourist touristic planning service, which are clearly stated in a thorough survey research in the field [14].

a) Most of the related work proposes more of a generic program rather than systems that could tackle specific situations in a region or country. However, countries such as Japan impose unique difficulties, such as language barriers, complex train network, and unique culture to the tourists. Moreover, by targeting a specific tourism region in Japan one can focus on leveraging local advantages such as well-developed subway networks, bus, a variety of cultural events, etc. in building a better recommendation system. Since to the best of our knowledge, there are not many tours planning recommendation system designed specifically for Japan targeting foreign tourists, our research may be considered a pioneer along this direction.

b) Our system integrates logistic planning during the recommendation process. Particularly, aside from user's context (e.g., current location, weather condition), we consider train public transportation as a foundation for generating recommendation result. Such perspective is emphasized as a necessity in this line of research [19], which is different from public transportation advisory services (e.g., [20],[21]) that can be requested by the user after a list of attractions or services suggestion. Furthermore, because our suggested attractions are clustered and selected simultaneously with geographical influence, our approach ensures the practicality of generated solution while improving user's satisfaction [17].

c) We examine a "unified attractions/tourist services recommendation" [14] with time and budget constraints. Because of the availability and diversity of input data sources, we are able to suggest not only attractions or point of interests, but also services such as restaurants, cultural events, foreign tourist support spots, etc. Since services cost much more than sightseeing scenarios, which usually do not put much financial concerns on the users, the involvement of services recommendation would further raise a concern about budget constraints. This matter is also addressed and resolved by our system, which generates results that satisfy user's constraints including budget and time window.

d) Since we consider the practicality of the system as our priority in the research process, we thrive to make the solution as realistic as possible. As clearly stated by Damianos at el.: a realistic tour had better provide travelers with breaks, either for a meal or resting in a nearby parks [14]. This function is also equipped in the prototype of the system in a way that the user can interrupt the touring anytime for a break, and the system would immediately suggest parks, coffee shops, restrooms, etc. that are adjacent to the user's current location.

III. METHODOLOGY

This research proposes an empirical and practical approach to the described context above so as to support individual tourists, tourism policy makers, as well as tourism managers and provide more fruitful experiences to foreign tourists. The rest of the paper is as follows.

A. Data Collection

1) Data Source and Description:

A collection of appropriate data is crucial for any successful projects, and this is especially a conundrum in the field of tourism where relevant data is scattered across wide networks rather than being aggregated at a fixed location. Not to mention that tourism activities usually involves other domains such as public infrastructures, policy and promotional activities, weather forecasting, etc. which picture tourism related data as much prevalent. Particularly, the result shown in this paper is based on the data that were obtained from four relevant websites and services; namely, Navitime (http://www.navitime.co.jp), TripAdvisor (http://tripadvisor.com), and Jalan (*http://www.jalan.net*). The Navitime.*com* website contains a list of relevant information on all of the JR train stations in Japan (e.g., their names, addresses, geographical coordinates), which have already been categorized into 47 different prefectures and operation companies. The website has also an extensive list of tourism guidance offices and regional specialty shops located

throughout Japan [22]. The TripAdvisor.com is a worldwide famous entity for its rich database on tourism products. Its portal provides an extensive list of the most famous attractions in Japan, which are ranked by actual travelers, and are neatly classified into various categories such as "sight & landmarks", "natural & parks", or "museums" [23]. The Jalan.net is a domestic tourism portal that provides insight information on regularly updated cultural events and intangible attributes such as average visiting time, restaurants' price and rating, best time to visit, etc., that have been collected from travelers [24]. Lastly, Yahoo API is used as a reliable concurrent weather forecasting.

2) Collecting Data Using Web Scraping Technique

To equip the system with the capability of suggesting a unified combination of both tangible and intangible tourism products, and incorporating public transportation during the process, the work of collecting relevant data from the three stated online sources is indispensable. However, the fact that the three sites do not have a unified structure and their relevant data records (e.g., details regarding a specific place) are displayed on a separate web page; makes the task quite cumbersome. Fortunately, this problem can be resolved using web-scraping technique, a programmatic approach that enables generating numerous virtual web agents that are capable of interacting websites and extracting their data in a systematic manner. As such, this research employs Ruby programming language to build a program that utilizes "Watir" and "Nokogiri" library to facilitate the task. Moreover, "Parallel" library is also used to shorten the execution time by processing multiple URLs simultaneously. All of the retrieved data is subsequently stored in a local MySQL database, which acts as an intermediary data library for the recommendation system.

B. Overall System Architecture

This section provides an illustrative example of situations in which our system could be beneficial.

Instead of focusing on a total tour recommendation package, which usually suggest a fix accommodation facility (e.g., hotel, hostel, etc.) for the whole trip and lasts for couple of days, our approach centers on a practical scenario that involves two separate module. While the first module gets executed once, the second module operates dynamically and continuously adjust the recommendation to real-time needs:

a) The first module involves generation of short-term touristic trip (e.g., within a day) which passes by various physical attractions, places, sceneries, or cultural events in accordance with public transportation starting from an origin (e.g., user's current location), and satisfying other preferences (e.g., favorite attractions categories, budget and time constraints, concurrent weather condition, etc.).

b) Second process more or less operates in real-time and recommends the so-called "on-demand services" such as restaurants (e.g., lunch, dinner, etc.), resting places (e.g., nearby coffee shops, parks, public restrooms, etc.), and attend to immediate inquiries and demands of the users (e.g., passing by tourism office in the vicinity). These are called "on-demand services" since their occurrences are not fixed and depend on different contexts. This is because, different people might have different desires for meal-time, break-time or a nearby vicinity maybe unpredictable, etc. Furthermore, since during the first process, the system also incorporates user's preferences that may dynamically change the trip (e.g., weather condition, etc.), the real-time adjustment capability of the second module could become quite handy.

Figure 1 shows the system flow and the next subsections explains the methodology used in its implementation.

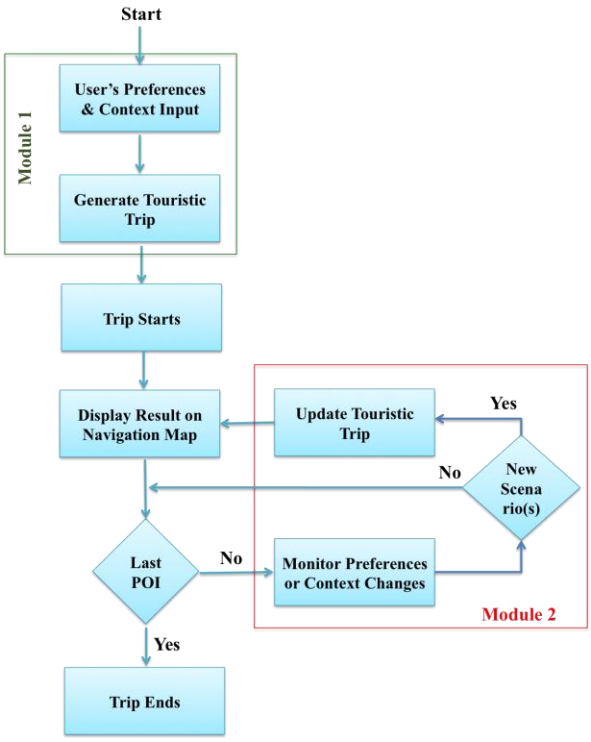


Fig. 1. System flow and its two modules (shown in green and red color).

C. Module 1: Touristic Trip Recommendation Using Cluter-first-route-second Heuristics.

Our system employs cluster-first-route-second which is inspired by the nature of touristic trip in Japan. Public transportations such as trains, subways, and buses are highly developed and utterly integrated into daily lives of local people. This has consequently shaped many aspects of the tourism industry, notably those in the touristic planning process. Particularly, as an effort to leverage the well-established network of more than 10,000 train stations in Japan with the 1-day free train pass ticket to reduce huge amount of transportation cost, a typical touring trip usually involves commuting through train stations, stopping at some of them and walking to surrounding areas, nearby sceneries and attraction sites. In fact, this observation aligns with many of researches on power-law distribution which predicts the user's likely favor nearby attractions rather than those that far away [25][26]. This is also further supported by the argument that travelers, after touring one attraction, have a tendency to move to an adjacent point of interest [17]. Therefore, in the linear programming context, looking for a sequence of point of interests out of a selection pool of thousands together with an optimal travelling route among train stations can simply be interpreted as searching for a route that passes by a sequence of train stations which satisfies user's touristic preferences and context. The following procedures explain details of this process.

1) Category and Weather Based POIs Filtering

From a pool of all of available attractions, we first filter them based on the user's selected categories and current weather information (if requested) so as to get a list those

POIs that meet the user's preferences. Depending on the weather conditions (e.g., raining, snowing), outdoor POIs such as natural parks or bridges may not get included in the result list. When this category option is unselected, the algorithm outputs a list of the most popular places in the examining area.

2) Heuristic POIs Clustering Based on Nearest JR Train Station and Walking Preferences:

In this step, our approach subsequently clusters all of the filters POIs into different geographical groups associated with a nearby JR train station, hence the distance among the station and attractions located inside each of the clusters become the walking distance travelled by the user. Walking preference is one of the interesting factors that the paper takes into consideration. Different people have different walking preferences when traveling which can be traced to differences in demographic and cultural preferences [27]. So, the most likely problem that has to be addressed is how to search for POIs that are located within a favorable walking distance from a nearby JR station. Therefore, the measurement of walking distances among thousands of the examining stations and attractions is indipensable, which is yet very expensive in terms of computation. Instead of using available public API for finding the nearest station to/from a place which would require long processing time to go through thousands of places, this paper suggests the Euclidean measurement to heuristically facilitate the categorization. Undoubtedly the Euclidean measurement is not a perfect approach for calculating the walking distance among places; the trade-off between its results and performance is acceptable, however.

As shown in Figure 2, on a 2-dimensional geographical map, from each of the JR station, a circle, having the relative walking distance as its radius, can be drawn to indicate which POIs can be reached from the station. As a result, one can easily compare relative distances when traveling from places to places quite efficiently and obtain lists of nearby stations for each of the POIs for later access. However, since walking distance preferences vary among users, an addition requirement to the user's input is necessary to retrieve the suitable walking distances used in the clustering process. So, we categorize users' walking distance preferences into three levels: "Not really", "Fairly", and "Definitely". Because there

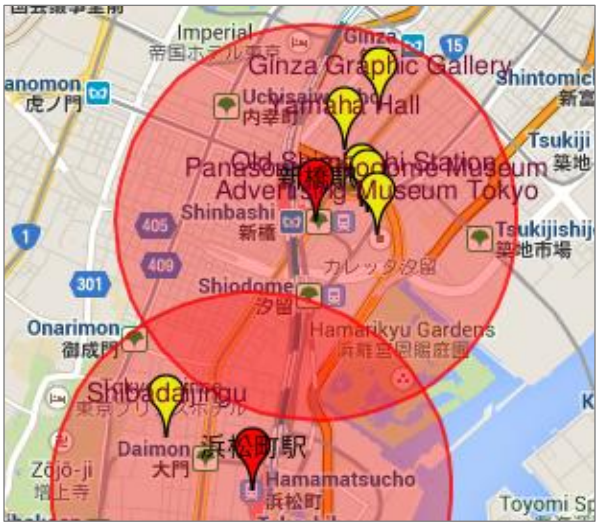


Fig. 2. Red circles with 800-meter radius are drawn to estimate the Euclidean distance from each of station (red markers) to nearby POIs (yellow markers).

are trade-off between preferable walking distance and number of POIs can be covered during the clustering process, walking distances associating with each of three levels are set depending on different touristic areas to ensure that an appropriate portion of POIs can be covered. To illustrate,

Table 1 describes the result of POIs coverage in percentage in accordance with seven different radius distance used during the clustering process within eight different major cities and downtown areas as recommended by the JNTO. The finding shows that three different levels of walking preferences (shown in bold) are selected.

TABLE I.
CLUSTERING POIS BY NEARBY JR STATION

Distance (meter)		300	500	800	1000	1500	2000	3000
Tokyo	Cover	18	32	**53**	61	**78**	**88**	95
	Top 20	11	21	**35**	41	**53**	**60**	64
	Top 50	14	25	**42**	49	**63**	**72**	77
	Top 100	15	28	**46**	53	**69**	**77**	83
Yokohama	Cover	6	20	**41**	50	**57**	**63**	88
	Top 20	4	11	**25**	30	**36**	**41**	63
	Top 50	5	16	**33**	39	**46**	**51**	73
	Top 100	5	18	**37**	44	**50**	**56**	79
Kyoto	Cover	1	4	**11**	**16**	**24**	63	81
	Top 20	1	3	**7**	**11**	**15**	49	60
	Top 50	1	3	**9**	**13**	**18**	54	68
	Top 100	1	3	**10**	**14**	**20**	58	73
Osaka	Cover	19	33	**63**	**72**	**94**	96	99
	Top 20	13	22	**42**	**51**	**69**	71	72
	Top 50	16	27	**51**	**60**	**79**	81	84
	Top 100	17	29	**55**	**64**	**85**	87	90
Kobe	Cover	11	**42**	**64**	**70**	75	77	83
	Top 20	8	**32**	**47**	**50**	55	57	60
	Top 50	9	**36**	**54**	**58**	64	65	70
	Top 100	10	**38**	**58**	**63**	69	70	76
Fukuoka	Cover	5	9	**17**	22	**33**	**44**	83
	Top 20	1	4	**10**	12	**19**	**26**	55
	Top 50	3	6	**12**	17	**25**	**36**	70
	Top 100	4	7	**14**	18	**28**	**38**	75
Nagoya	Cover	5	9	16	**26**	**46**	**72**	87
	Top 20	3	6	11	**18**	**34**	**53**	64
	Top 50	3	6	13	**21**	**38**	**60**	74
	Top 100	4	8	14	**22**	**41**	**65**	79
Sapporo	Cover	7	26	34	38	50	60	72
	Top 20	3	19	21	23	31	39	48
	Top 50	6	22	26	28	39	48	59
	Top 100	7	25	30	33	43	52	63

Furthermore, by clustering POIs, each of the clusters is identified not only by the central station, but also by a ranking score. As implied by its name, the ranking score represents the relative popularity of the cluster and can be calculated by summing up the rankings of all the included POIs. However, the lower the score, the better its cluster's ranking. Even though our approach is not focusing on the most well-known attractions, ranking is yet an assistive metric in the following searching process.

3) Heuristics Based Clustering and POIs Searching

By using the greedy algorithm, we iteratively search for adjacent stations surrounded by POIs that posses the best ranking positions. As illustrated in Figure 3, this searching process can be examined by modeling it on a directed graph. Each vertex of the graph describes location of a train station, which can either be the one nearest to the user's current position, or to the desired destination, or the ones to pass through on the trip. Each of those nodes will have a weighted ranking score. The lower the score, the more satisfying it is. Edges connecting vertexes represent the travel duration between them. The overall goal in solving this problem is to accumulatively select the best available cluster and included POIs that are located as close to the current position as possible, and satisfy all of other input constraints such as available budget and time window.

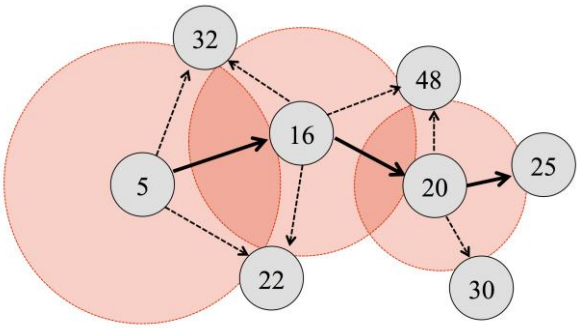


Fig. 3. Multiple circles with gradually increasing radius is drawn from each of vertex (small gray circles) to recognize its nearest next best station.

a) Searching for next best cluster: Through the greedy algorithm, we try to obtain a good tour by searching locally from one vertex to another in order to choose the next best vertex to visit. The next best vertex (or station), must be as close to the current vertex (or station) as possible, and posses the most satisfied weighted ranking score (e.g., Figure 2). In order to find nearby vertex from a specific location, a similar approach being used during the POIs clustering process that employs Euclidean measurement can be done. This method to replace extensive distance calculation between all of the vertexes hence reduce the computational burden on the system.

b) Searching for the next best POI: At each of the vertex, greedy algorithm with local search technique is subsequently used to select the best POIs located within a specified preferred walking distance from central station. So as to pick the POIs that both has high rankings, and satisfy the constraints variables, all of the POIs is ascendingly sorted by their rankings, and the selection process is done from the top in accordance with the remaining budget and time. Following each of the POI selection, those constraints variables are updated. The search would stop once either all of the constraints are met or a balanced number of POIs from each of the desired categories is obtained. A sample result is illustrated in Figure 4 as follows.

Fig. 4. Search result of a suggested trip of about 4 hours in cloudy weather condition, starting from the Sofitel Hotel Tokyo (blue marker), visit a nearby Shinjuku eki (red marker), and subsequently walks through each POIs (yellow marker) which include "Isetan department store", "Omoide Yokocho", "Hanazono Jinja", "Museum of Haiku", and "Shinjuku Takashimaya", elements of specified categories "Landmarks", "Specialty Museums", "Religious Sites", and "Shopping".

4) Route Construction & Optimization

After obtaining a list of suggested POIs and its associated intermediary stations, an optimal sequence of traveling route that passes through all the places is investigated. Basically, the overall route will comprise of the traveling order through stations, and the walking order within a group of POIs located at a single station. Despite the fact that many genetic algorithms such as tabu search, etc., [28][29] facilitate the route optimization procedure, we used Google Maps API because of its proven success and the practical nature of our research approach. Apart from the optimal route passing by multiple stations and POIs that can be easily calculated with the support of the API, in order to present the most appropriate touristic planning to the user, the system undergoes two main routing and optimization procedures as follows.

a) Taking traveling time among stations, and POIs into account: Because this factor was not considered during the local search to enhance the performance, a more thorough time duration needed to finish the whole trip is calculated. Since travelling time between two station might vary during the day, depending on the type of train that the user gets on, it is can be requested by sending parameters including estimated departure time, the origin and as well as the destination station to Google API service. Similarly, the walking time inside each of the clusters is also obtainable. By doing this, we can reconsider the initial solution by eliminating any redundant POIs, if necessary, backward from the trip to better match with the original time constraint.

b) Re-matching POIs: Through observations, after the searching process, there might be cases where POIs are more accessible from nearby POIs or a station belonging to another geographical cluster (e.g., Figure 5). In such cases, an optimization process is designed to iteratively examine each of the selected POIs and move them to a more appropriate central station, and remove any of the redundant stations from the initial route. Moreover, the optimization process also considers POIs that initially eliminated in the clustering

process due to further distance compared with the user's preferences as possible replacements.

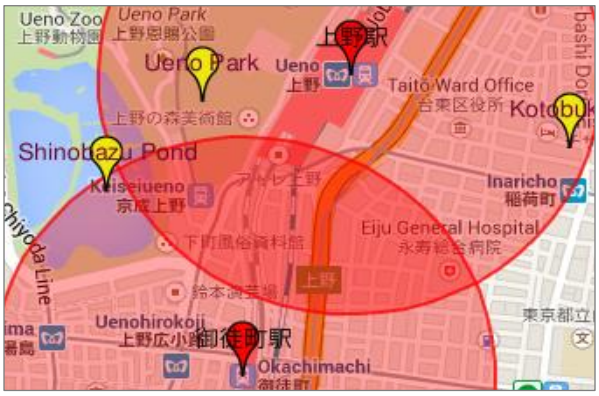


Fig. 5. A better solution can be achieved by matching the "Shinobazu Pond" POI (yellow markers) to the upper station to reduce the commuting time between the two stations (red markers).

The illustrated POIs clustering and filtering, heuristic local search, post-optimization steps of the first module are further elaborated by pseudo-codes in Figures 6, 7 and 8 respectively.

D. Module 2: Real-time Preferences, Context Monitor and Recommendation

Regardless of how good an initially generated result may be, a good recommendation system has to consider the possibility of incorporating dynamic changes in both user's preferences and the environmental impacts. This is especially applicable in the field of tourism. For instance, the user might want to add a new POI during the trip, take a rest, have a meal at nearby restaurants, look for restrooms; or suddenly it might start raining. The fact that our problem deals not with a single user but travelers from different countries with varied cultural preferences and habits, there is definitely a need for a certain level of flexibility in the system, which can make it more compatible to such capriciousness.

We categorized all of the possible internal (e.g., from user's perspective) and external changes as follows:

1) POIs related changes: These happen when the user wants to change the type of attractions during the trip, add, remove, or change an attraction on the list. This category also considers effects of sudden weather changes (e.g., raining, sunny, etc.).

2) Additional services demands: On an on-going trip, the user might request the system to recommend surrounding services including culinary services (e.g., restaurants, coffee shops, etc.), resting places (e.g., parks, rest room), tourism information office, free wifi-spots and so on.

Such data can be retrieved by web scraping technique from some of well-known local websites such as Navitime.net (for public rest rooms, tourism information office), FreeSpot.com (for public free Wi-Fi spots), Jalan.net (for culinary services). For its achievement, a background module is set-up in the system to continuously capture and response such changes at a specific interval of 5 seconds. By re-using part of the 1st module to generate touristic result basing on the newly captured input, this module equips the system with more flexibility and responsiveness. The details of practical implementation architecture, and as well as some technologies can be employed to resolve some of recognized connectivity and energy consumption of portable devices bill be addressed in the next section of the paper.

```
Result: Array filteredPOIs
foreach POI in POIs do
    if POI match User.category then
        if POI match currentWeather then
            ADD POI to filteredPOIs
        end
    end
end
```

Algorithm 1: POIs Filtering on Category and Weather

Fig. 7. Pseudo-code for step 1: Filtering POIs basing on user's preferences on choices of category and concurrent weather condition.

```
Result: Array finalPOIs, finalStations
nextStation = User.currentLocation;
while checkTripSatisfied(finalPOIs) == false do
    POIS = findPOIsNearStation(filteredPOIs, nextStation);
    POIS = sortByRankAscending(POIS);
    foreach POI in POIS do
        if checkTripSatisfied(finalPOIs) == false then
            if POI.timeNeeded < User.timeConstraint AND
            POI.budgetNeeded < User.budgetConstraint then
                User.timeConstraint -= POI.timeNeeded;
                User.budgetConstraint -= POI.budgetNeeded;
                POI.nearByStation = nextStation;
                ADD POI to finalPOIs;
                if nextStation not in finalStations then
                    ADD nextStation to finalStations;
                end
            end
        end
        else
            Exit For;
        end
    end
    nearStations = findNearStationsFromLocation(nextStation);
    if nearStations is not null then
        nextStation = findStationHighestScore(nearStations);
    end
    else
        Exit While;
    end
end
```

Algorithm 2: POIs and Station Searching

Fig. 6. Pseudo-code for step 2: Local searching for POIs and Station with user's constraints using the Greedy algorithm.

E. Prototype Implementation

A prototype of the proposed system on web-based mobile platform is built and tested so as to further explain some empirical perspectives on its practical implementation, namely system architecture, GUI, connectivity and energy consumption.

1) System Architecture

```
Result: Array finalPOIs, finalStations
foreach POI in finalPOIS do
    bestNearByStation = findBestNearbyStation(POI, finalStations);
    if POI.nearByStation != bestNearByStation then
        POI.nearByStation = bestNearByStation;
    end
end
foreach POI in allPOIsInArea do
    if POI not in finalPOIs then
        replacablePOI = findReplacablePOI(POI, finalPOIs);
        if replacablePOI then
            replacePOI(POI, replacablePOI, finalPOIs);
        end
    end
end
removeRedundantStations(finalPOIs, finalStations);
```

Algorithm 3: Post Optimization

Fig. 8. Pseudo-code for step 3: Optimizing the list of stations and POIs by re-matching each of POIs to their most optimal station, searching for better POI in the area, and removing any redundant stations.

Overall, the system architecture comprises of a central server with an installed database such as MySQL, Mongo DB, etc., and several clients connected. The proposed recommendation algorithm can be programmed to compile under native mobile systems (e.g., iOS, Android, Windows phone) and be easily loaded into tourist portable devices (e.g., smart phones, tablet, etc.), together with its language preference (e.g., English, Vietnamese, Chinese, Korean, etc.) and other resource files. This software acts as a client, connects to the server and downloads or updates data packages on user's request, or on a regular basis. Different touristic zones and regions (e.g., Tokyo, Kyoto, Osaka, etc.), will have a distinguished data package stored in the database and selectable upon the user's preference. For demonstration purpose, we have implemented the proposed system as a web-based mobile application. Its GUI and various functions are introduced in the next section.

2) Prototype GUI & Integrated Functions

This section describes the implemented web-based prototype of the proposed trip recommendation application for foreign tourists in Japan. Default names of all stations and POIs are shown in English, but the language is intended to be configurable so as to provide a much friendlier user interface.

As shown in Figure 9, when activated, the software asks its user for initial input preferences including trip origin (e.g., current location, specific hotel, train station, etc.), available budget (in Japanese Yen), time window (in hours), level of walking preference (e.g., Definitely, Not Really, or Not At All), whether or not to consider weather forecast, and preferable types of destinations (e.g., those shown in Figure 10). When every parameter is specified, the user can execute the program by clicking the button "Experience Japan" (top button in Figure 9) and receive a recommended touristic trip result, a sample of which is shown in Figure 11.

Since system comprehensively utilizes Google Maps library to visualize the mapping layers above the generated result, the program can also be used as a real-time navigation

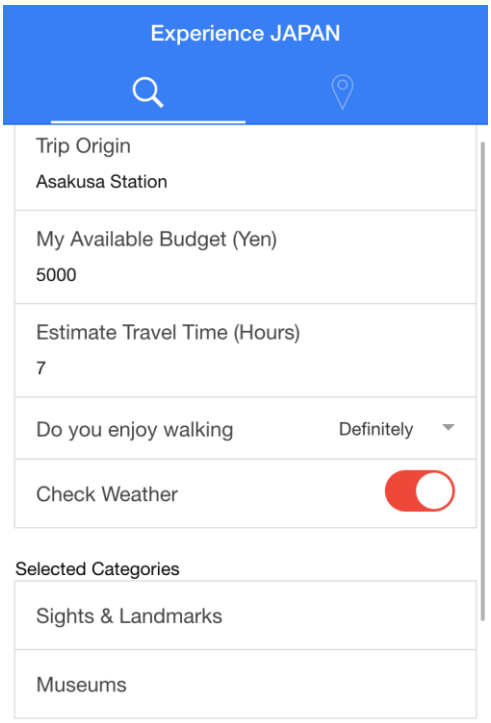


Fig. 9. GUI for preferences input from the user. In this case, the trip starts from user's current location, which is "Asakusa Station" in Tokyo area. A budget of 5000 Yen and 7-hour availability are also specified. "Definitely" is selected as walking preference. The trip would be generated basing on the current weather condition

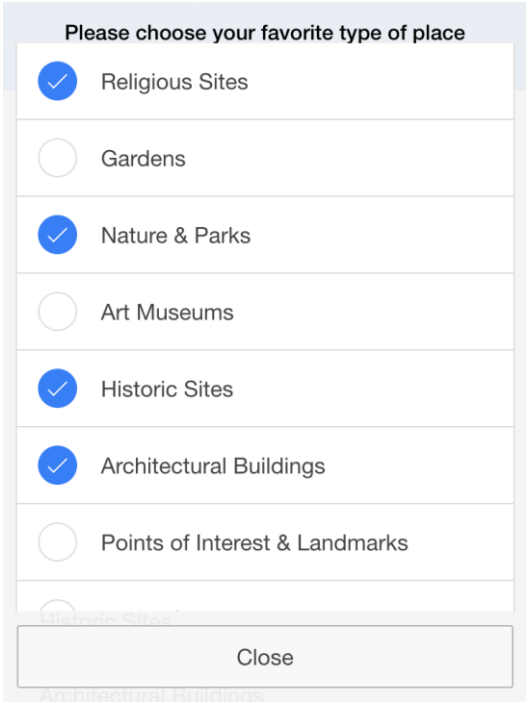


Fig. 10. An example of modal box showing different options for favorite type of destinations.

system during the trip. An array of functional buttons, placed at the bottom of the result screen, enable the user to get the trip's information (Info), look for nearby services or places such as restaurants, tourism office (Search), center the current trip on the navigation map (Center), and tell the system to start navigating the trip from the current location (Start).

Figure 12 illustrates the info board displaying all the information and available actions relevant to the trip.

Fig. 11. GUI of trip recommendation being integrated with visual navigation map. Blue marker represents user's current location. Red markers show the location of train stations, and yellow one locates POIs during the trip.

Through this interface, its user can check all of the POIs and their traveling order, etc., and as well modify the trip by adding or removing a specific attraction, or re-order them. Other input preferences such as the available budget, time, and favorite categories can be altered on the input screen. All of the changes will be updated on the concurrent trip by considering its user's current context (e.g., new location, remaining budget, etc.).

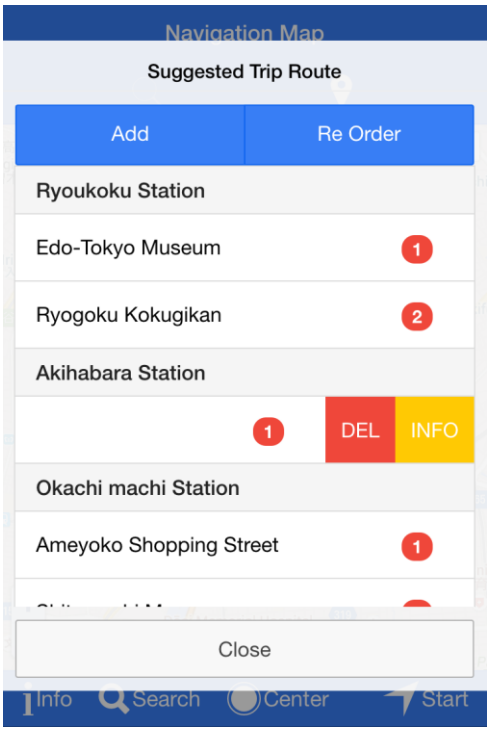


Fig. 12. GUI of modal box displaying the information of the recommended trip. Each of POI is categorized into list heading by the name of the central station. Various tasks can be easily done by the user such as swiping left on an item to delete or search info of a specific attraction, adding a specific place, or re-order the route.

During the trip, the user can activate "Search" function on the main navigation screen to discover the surroundings of the current location. Through a relevant popup GUI shown in Figure 13, its user can find nearby restaurants or coffee shops for lunch or dinner; look for a nearby tourism guidance office for additional help, search for accessible Wi-Fi spots, and seek out for a short resting place (e.g., Natural Park). As shown in Figure 14, each of the newly discovered service location will be indicated on the main map with distinct markers for easier navigation.

3) Connectivity & Energy Consumption

As it is evident from our presented GUIs, the proposed system not only plays the role of trip planning, but also functions as a navigation system throughout the trip. Therefore, the tasks of processing location services (e.g., GPS), database update, and weather forecast impose concerns about the connectivity and energy consumption of the user's device. Battery consumption is a paramount aspect in practical mobile applications [30], especially in real-time or near real-time implementation. There is a strong correlation between the communication protocol and energy consumption of mobile devices, which is actually dependent on the type of communication method (e.g., Wi-Fi, 3G) [30]. Because our system targets foreign tourists in Japan, who most likely visit the country for the first time, there is no guarantee that a sustainable Internet connection is always available, regardless some of the recent efforts being made by

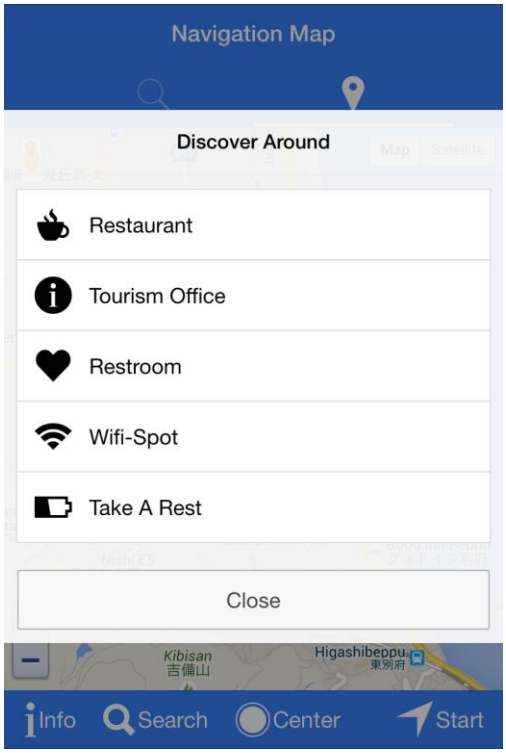


Fig. 13. GUI of "Search" or local discovery function, including the searching for nearby restaurants, tourism office, restroom, Wi-Fi spot, or a surrounding park for taking a rest.

Fig. 13. As a result of the search function, the map shows a green marker representing a tourism guidance office located nearby user's current location (e.g., Blue marker)

the government and domestic service providers. This hence raises our concern about which communication protocol should be employed for the touristic recommendation software, and whether all of the necessary processing should be done on the client or at the server site.

With the recent prevalence of HTML 5 and Web Socket communication technology, we propose the use of Web Socket as the main data transmission protocol (e.g., Table 2). Even though XMPP protocol consumes less energy than Web Socket in the case of Wi-Fi connection, considering its overall high performance and low energy level consumption (e.g., rank 1st in 3G and 2nd in Wi-Fi connection [30]), Web Socket is selected as the single protocol method for a consistent implementation throughout the system.

Furthermore, in order to minimize extensive processing tasks for the mobile devices, the searching and trip generation procedure (module 1), particularly when an Internet connection is unavailable, as well as the regular web-scraping process to update the database, should be done on the central server. This can make the system more efficient, since we could both take advantage of a well-performing server

infrastructure and as well optimize the communication process.

TABLE II.
PROPOSED COMMUNICATION METHODS

Tasks	Wi-Fi	3G	Internet Unavailable
Searching & Generating Trip	Server (Web Socket)	Server (Web Socket)	Client
Database Update	Web Socket	Web Socket	Not available
Location Service	GPS	GPS	GPS

IV. DISCUSSION

A. Analytical Result

This subsection examines the result of our various testing attempts in order to demonstrate the practicality and performance of the proposed tourism recommendation system.

We tested our approach on eight different major cities and their downtown areas as recommended by the JNTO which are Tokyo, Osaka, Kyoto, Kobe, Nagoya, Sapporo, Fukuoka, and Yokohama. The results are shown in Tables 2 – 9 and for each of the area, the following three practical scenarios and parameters are presented.

1) General Usage: Travel from a main station to up-to 10 different POIs categorized as "Museum", "Religious Site", or "Historical Site". "Definitely" or "Not Really" is used as walking distance preference.

2) General Usage / Short-term business with long-time window: Travel from a main station or famous hotel (e.g., top 50 regional hotel according to TripAdvisor) to up to 10 different well-known POIs. "Definitely" or "Not Really" is used as walking distance preference.

3) Short-term business with short-time window: Travel from a famous hotel (e.g., top 50 regional hotel according to TripAdvisor) to up-to 4 different popular POIs in the nearby area. "Not at all" is used as walking distance preference.

Even though the suggested trip depends much on the trip origin, the consideration of some main stations in the scenarios partly strengthens the analysis. In each of the test case, percentage of POIs coverage in the top 50, 100, as well as the number of stations for the suggested trip are used as a metric for examination. Performance wise, the system, in general, comes up with the result in less than 10 seconds, but it is much faster for the case of short-term trip with a short-time window. In our view, this scenario is pretty useful in a practical sense, since people attending to short-term business trips, conferences, or transit passengers, etc., would like to tour the country as much as possible, something which has not been recognized and implemented in many commercial tourist packages.

As can be noticed, except for Kyoto city, 60% to 80% of the suggested POIs belong to top 50 and 100 regional rankings, respectively. This demonstrates the balance between the popularity of POIs and distance needed to travel, which turns out to be 3 train stations. Although this paper focuses on JR train system as the main transpiration system, for the case of Kyoto city bicycles and buses turns out to be more convenient, something which should be considered in future work for a much more comprehensive approach.

TABLE III.
TESTING RESULT FOR TOKYO AREA

Criteria	*Result*		
Origin	Tokyo Station	Harajuku Station	Park Hyatt Tokyo Hotel
POIs	Tokyo Central Railway Station, Edo Castle Ruin, Bridgestone Museum of Art, Mitsubishi Ichigokan Museum, Mitsui Memorial Museum, Nihonbashi, Suitengu Shrine, Fukutoku Shrine, Takarada Ebisu Shrine, The National Museum of Modern Art	Shinjuku Gyoen National Garden, Meiji Jingu, Yoyogi Park, Shibuya Pedestrian Scramble, Shibuya Center-gai, Omotesando, Aoyama Cemetery, Shibuya Fureai Botanical Center, Harajuku Takeshita-dori, Meijijingu Gaien	Tokyo Metropolitan Government Office, Shinjuku Golden Gai, Omoide Yokocho, Shinjuku Camera Town
Top 50	20%	**60%**	50%
Top 100	40%	**80%**	50%
Stations	1	3	1
Cluster Radius	1000 meters	2000 meters	800 meters

TABLE IV.
TESTING RESULT FOR YOKOHAMA AREA

Criteria	*Result*		
Origin	Minatomirai Station	Hotel Monterey Yokohama	BEST WESTERN Yokohama
POIs	Cupnoodles Museum, Mitsubishi Minatomirai Industrial Museum, Japan Coast Guard Museum, Iseyama Kotai Shrine, Nipponmaru Memorial Park, Kanagawa Prefectural Museum, Kuan Ti Miao Temple, Yokohama Port Opening Hall, Ma Zhu Miao, Yokohama Museum of Art	Osanbashi Pier, Yokohama Chinatown, Yokohama Stadium, Iseyama Kotai Shrine, Kanagawa Prefectural Museum, Diplomats House, Bashamichi Shopping Destrict, NYK Maritime Museum, Yokohamabashi Shopping District, Yokohama Customs Headqurters	Yokohama Chinatown, Yokohama Stadium, Kanagawa Prefectural Museum, Diplomats House
Top 50	70%	**60%**	50%
Top 100	90%	**90%**	100%
Stations	3	3	2
Cluster Radius	1500 meters	1000 meters	800 meters

TABLE V.
TESTING RESULT FOR FUKUOKA AREA

Criteria	*Result*		
Origin	Hakata Station	Hakata Station	Hotel Nikko Fukuoka
POIs	Kushida Shrine, Tochoji Temple,	Kushida Shrine, Tochoji Temple,	Tochoji Temple, Jotenji Temple,

Criteria	*Result*		
	Jotenji Temple, Hakata Machiya Folk Museum, Hakata Traditional Craft Center, Genko Historical Museum	Jotenji Temple, Fukuoka Asian Art Museum, Shofukuji Temple, Yatai, Hakata Machiya Folk Museum, Rakusuien, Seiryu Park, Itazuke Iseki	Hakata Sennen no Mon, Kakueiji Temple
Top 50	50%	**60%**	50%
Top 100	100%	**80%**	50%
Stations	6	1	1
Cluster Radius	1000 meters	1500 meters	800 meters

TABLE VI.
TESTING RESULT FOR OSAKA AREA

Criteria	*Result*		
Origin	Osaka Station	Osaka Station	Swissotel Nankai Osaka Hotel
POIs	Museum of Oriental Ceramics, Osaka City Central Hall, Midosuji Street, Tsuyunoten Shrine, Osaka Prefecture Library, Namban Culture Center, Dojima Yakushido, Taiyuji Temple, Osaka Science Museum, Bank of Japan Osaka Branch Old Building	Floating Garden, Hep Five Ferris Wheel, Museum of Oriental Ceramics, Osaka City Central Hall, Tsuyunoten Shrine, Nakanoshima, Shin Umeda City, Midosuji Street, Shin-Umeda Shokudogai, K itashinchi	Dotonbori, Shinsaibashi, Hozenji Yokocho, Sennichimae Doguyasuji Shopping Street
Top 50	20%	**60%**	100%
Top 100	70%	**100%**	100%
Stations	2	2	0
Cluster Radius	1000 meters	1000 meters	800 meters

TABLE VII.
TESTING RESULT FOR KYOTO AREA

Criteria	*Result*		
Origin	Inari Station	Kyoto Station	Hotel Granvia Kyoto
POIs	Fushimi Inari Shrine, Tofukuji Temple, Unryuin, Sennyuji Temple, Komyoi, Kyoto Municipal Science Center For Youth, Sekihoji Temple, Sanjusangendo Hall, Kyoto National Museum, Toyokuni Shrine Karamon	Sanjusangendo Hall, Tofukuji Temple, Umekoji Steam Locomotive Museum, Nishi Honganji, Sagano, Kyoto National Museum, Sumiya Motenashi Cultural and Art Museum, Toyokuni Shrine Karamon, Kozan Temple, Higashi Honganji	Shinsengumi Mibu Tonjo Kyuseki, Mibudera, Old Maekowa Residence, Chishaku-in
Top 50	30%	**50%**	0%
Top 100	50%	**70%**	50%
Stations	2	2	2
Cluster Radius	1500 meters	1500 meters	800 meters

TABLE VIII.
TESTING RESULT FOR KOBE AREA

Criteria	*Result*		
Origin	Kobe Station	Kobe Station	Kobe Portopia Hotel
POIs	Takenaka Carpentry Tools Museum, Kawasaki Good Times World, Kobe Kitano Museum, Minatogawa Shrine, Kazamidori no Yakata, Kobe Anpanman Children's Museum & Mall, Kobe Kitano Tenman Shrine, Matsuo Inari Shrine, Jodoshianyoji Temple, Puraton Ornament Museum	Kobe Harborland, Kitano-cho, Kobe Kitano Museum, Chinatown, Kobe Port Tower, Kazamidori no Yakata, Former Drewell Mansion, Urokono-ie Uroko Museum of Art, Moegi no Yakata, Moegi no Yakata	Minatogawa Shrine, Shinkobe Ropeway, Urokono-ie Uroko Museum of Art, New Port Fifth Jetty Old Signal Station
Top 50	70%	**60%**	25%
Top 100	80%	**90%**	75%
Stations	2	2	2
Cluster Radius	800 meters	1000 meters	500 meters

TABLE IX.
TESTING RESULT FOR NAGOYA AREA

Criteria	*Result*		
Origin	Nagoya Station	Nagoya Station	Nagoya Marriott Associa Hotel
POIs	Toyota Commemorative Museum of Industry and Technology, Shikemichi, Lucent Avenue, Former Kato Shokai Building, Asama Shrine, Keihoin, Nagoya/Boston Museum of Fine Arts, Hioki Shrine, Osukannon	Nagoya Castle, Yamazaki River, Osu Shopping Street, Hoshoin, Nana-chan Mannequin, Hitsuji Shrine, Sky Promenade, Shikemichi, Great Beckoning Cat Statue, Aichiken Gokoku Shrine	Nunoike Catholic Church, Takamu Shrine, Art Salon Wasabi, Mt. Hachiman Tomb
Top 50	33%	**60%**	0%
Top 100	56%	**90%**	50%
Stations	2	1	2
Cluster Radius	1500 meters	2000 meters	800 meters

TABLE X.
TESTING RESULT FOR SAPPORO AREA

Criteria	*Result*		
Origin	Sapporo Station	Sapporo Station	JR Tower Hotel Nikko Sapporo
POIs	Former Hokkaido Government Office Building, Sapporo Beer Museum, The Hokkaido University Museum, Botanic Garden Hokkaido University, Clock Tower, Sapporo Science Center, Kotoni Shrine, Shin Kotoni Shrine, Teine Shrine, Seikatei	Hokkaido University Sapporo Campus, Sapporo JR Tower Observatory, Former Hokkaido Government Office Building, Sapporo TV Tower, Botanic Garden Hokkaido University, Clock Tower, Ganso Sapporo Ramen Street, Seikatei, Sapporo Ekimaedori, Izumi Statue	Sapporo JR Tower Observatory, Ganso Sapporo Ramen Street, Sapporo Ekimaedori, Izumi Statue
Top 50	50%	**60%**	25%
Top 100	80%	**80%**	50%
Stations	4	0	1
Cluster Radius	1500 meters	1000 meters	500 meters

B. Comparison with Other Tourism Recommendation Systems

This subsection briefly compares our approach with other recommendation systems in order to highlight its unique features.

1) Nature of Recommendation System

While many recommendation systems such as SigTur/E-Destination [31] mainly focuses on the act of POIs filtering, the final result of which is a list of POIs that match with user's preferences, our approach concentrates more on the touristic planning process which embraces logistic consideration and geographical context of the user. Furthermore, our approach introduces more flexibility by considering user's dynamic needs during the trip.

2) Scope of the Recommendation System

Even though Tokyo metropolitan area is chosen as the target vicinity, availability of diverse data sources, as well as examination of Japan's unique traits (e.g., complex train system, free train pass ticket) make the proposed methodologies applicable to other regions as well. Considering that other recommendation systems rarely take the localization aspect into consideration (e.g., [17], [31]), our approach can be considered as a pioneer tourism recommendation system for Japan at a national level.

3) Input Data Space for POIs Filtering

We consider the open accessibility of input data source for POIs filtering quite important. Algorithms such as collaborative filtering require an initial database or previous actual cases, which seems collectible solely through tours agencies. Geo-tagged sightseeing photos collectible by members from various sharing platforms such as Flickr, Panoramio, etc. are also being used to rank tourism attractions [17]. In our case, however, the system makes use of data openly available on the Internet. Because two out of the three main data sources are very active Japanese online services relating to the field of tourism, the input data can be considered more reliable and openly accessible.

V. LIMITS AND FUTURE RESEARCH

The following limitations can provide space for improvement and direction for future research.

1) A systematic approach to update the current database on a regular basis is necessary for providing the most uptodate information (e.g., new events) to the user.

2) The heuristic greedy algorithm finds the local optimal solution and the hope to meet the global optimum has proven its successful in many practical pieces of research and applications [31]. Despite that, there is a need for specific metrics to define the measurement of how good a tour recommendation is according to a specific user's profile.

3) Presently, our proposed system concentrates on short-term touristic plan and without considering hotel accommodation. This ability, however, is available in its architecture and can easily accommodate multi-days trip with hotel recommendation feature. Not only both TripAdvisor.com and Jalan.net contains information on thousands of hotels, the same programming approach used in the system can also extract hotel accommodate information from other well-maintained online sources. In fact, by incorporating hotel accommodation as part of the tour recommendation package, the system can suggest a hotel which complements its user's other input preferences and context.

4) The long-term goal of the research is to launch a system that not only able to draw a good tour plan in accordance with user's preferences, but also to recognize distinguished traveling interests of tourists from different countries while they visit and experience Japan. This differences in cultural preferences often lead to latent conflicts in communications, etc., which is a crucial issue in tourism hospitability. In order to resolve the conflicts, the work of recognizing of patterns, or implicit knowledge, of different users from different cultures needed to be done. However, the work should be not only precise, but also dynamic to cope with social changes. This can hopefully be resolved by applying machine learning and statistical learning approach to collected empirical data from surveys or preliminary mobile application and picture out the repeated pattern of a variety of cultural preferences. For example, Thai tourists tend to visit lots of temples while European travelers enjoy long-distance walk during the tour.

VI. CONCLUSION

This paper shares the results of an on-going research in recommending tour planning to tourists in Japan under a practical perspective. It has leveraged web scraping technique to collect a huge amount of supporting data, has built a prototype of a system that uses the greedy algorithm for POIs searching, and demonstrated its practicality. Despite its current limitations, potential benefits of such system that is generic to various tourism regions in Japan and customizable to numerous users from different nations is shown to be high. This paper also opens a discussion about the necessity of more localization and customization in building recommendation system, not only in the field of tourism, but in other disciplines as well.

REFERENCES

[1] JLL's Hotels & Hospitality Group, "Tokyo 2020 Olympics: expectations for the hotel industry, November 2014.

[2] "Japan outlines plan to achieve 20m tourists by 2020 Olympics". (n.d.), retrieved May 3, 2015, from http://www.travelweekly.co.uk/articles/2014/10/03/49577/japan outlines plan to achieve 20m tourists by 2020 olympics.html.e

[3] Japan Tourism Agency, "Efforts to promote a tourism-oriented nation", November, 2009.

[4] "Not just international but'Super Global Universities'", University World News. (n.d.), Retrieved May 3, 2015, from http://www.universityworldnews.com/article.php?story=20141120233337379.

[5] "Promoting two-way student exchange". (n.d.), retrieved May 3, 2015, from http://www.mext.go.jp/english/highered/1303572.htm.

[6] "Transportation - the official guide", Japan National Tourism Organization. (n.d.).

[7] Japan National Tourism Organization, "Japan Rankings", retrieved May 3, 2015, from http://us.jnto.go.jp/survey/japan_rankings.php.

[8] Japan ranked ninth most tourist-friendly nation. (n.d.). Japan Times. Retrieved July 22, 2015.

[9] Han Jiho, Yotsumoto Yukio, "The trends regarding foreign tourists to Beppu, Oita prefecture in Japan", Journal of Ritsumeikan Social Sciences and Humanities, vol. 2, pp. 61-72.

[10] Yuka Mera, Yoshiyuki Kurachi, and Naoko Ozaki, "Recent Increase in Foreign Visitors and Impact on Japan's Economy", Bank of Japan Review, December 2013.

[11] João Romão, Bart Neuts, Peter Nijkamp, Asami Shikida, "Determinants of trip choice, satisfaction and loyalty in an eco-tourism destination: a modelling study on the Shiretoko Peninsula, Japan", Ecological Economics, vol. 107, November 2014, pp. 195-205.

[12] Iponics Japan, "Marketing to Japan - appealing to the Japanese tourist". (n.d.). Retrieved May 3, 2015.

[13] "Attracting more tourists to Japan", The Japan Times, January, 2014. retrieved May 4, 2015.

[14] Damianos Gavalas, Charalampos Konstantopoulos, Konstantinos Mastakas, Grammati Pantziou, "Mobile recommender systems in tourism", Journal of Network and Computer Applications, vol. 39, March 2014, pp. 319-333.

[15] Ricardo Anacleto, Lino Figueiredo, Ana Almeida, Paulo Novais, Mobile application to provide personalized sightseeing tours, Journal of Network and Computer Applications, Volume 41, May 2014, Pages 56-64, ISSN 1084-8045

[16] Hsiu-Sen Chiang, Tien-Chi Huang, "User-adapted travel planning system for personalized schedule recommendation", Information Fusion, vol. 21, January 2015, pp. 3-17.

[17] Kai Jiang, Huagang Yin, Peng Wang, Nenghai Yu, Learning from contextual information of geo-tagged web photos to rank personalized tourism attractions, Neurocomputing, Volume 119, 7 November 2013, Pages 17-25, ISSN 0925-2312

[18] Gülçin Büyüközkan, Buse Ergün, "Intelligent system applications in electronic tourism", Expert Systems with Applications, vol. 38(6), June 2011, pp. 6586-6598.

[19] G. Tumas and F. Ricci, "Personalized Mobile City Transport Advisory System", Proc of ENTER'2009, pp. 173-183, 2009.

[20] Beatriz Rodríguez, Julián Molina, Fátima Pérez, Rafael Caballero, Interactive design of personalised tourism routes, Tourism Management, Volume 33, Issue 4, August 2012, Pages 926-940, ISSN 0261-5177.

[21] A. Garcia, P. Vansteenwegen, O., Arbelaitz, W. Souffriau and M.T. Linaza, "Integrating Public Transportation in Personalised Electronic Tourist Guides", Comput & Oper Res, 40(3), pp. 758-774, 2013

[22] NAVITIME. (n.d.). Retrieved July 22, 2015, from http://www.navitime.co.jp

[23] TripAdvisor: Read Reviews, Compare Prices & Book. (n.d.). Retrieved July 22, 2015, from http://tripadvisor.com

[24] 宿・ホテル予約 - 旅行ならじゃらん net. (n.d.). Retrieved July 22, 2015, from http://jalan.net

[25] M. Ye, P. Yin, W.-C. Lee, D.-L. Lee, Exploiting geographical influence for collaborative point-of-interest recommendation, in: Proceedings of the 34th International ACM SIGIR Conference on Research and Development in Information, SIGIR '11, ACM, New York, NY, USA, 2011, pp. 325–334.

[26] L. Backstrom, E. Sun, C. Marlow, Find me if you can: improving geographical prediction with social and spatial proximity, in: Proceedings of the 19th International Conference on World Wide Web, WWW '10, ACM, New York, NY, USA, 2010, pp. 61–70.

[27] Wallace, Floyd, Associates Inc, "Massachusetts pedestrian transportation plan", 1998.

[28] Kylie Bryant, 1., & Arthur Benjamin. (n.d.). "Genetic Algorithms and the Traveling Salesman Problem".

[29] Sylejmani, K., & Dika, A. "Solving touristic trip planning problem by using taboo search approach". International Journal of Computer Science Issues (IJCSI), vol. 8(5). 2011.

[30] da Silva, V. C. O., Oliveira, D. M., de Araujo, J. C. T., & Maciel, P. R. M. (2014). Energy Consumption in Mobile Devices Considering Communication Protocols. Advances in Information Sciences & Service Sciences, 6(5).

[31] Antonio Moreno, Aida Valls, David Isern, Lucas Marin, Joan Borràs, SigTur/E-Destination: Ontology-based personalized recommendation of Tourism and Leisure Activities, Engineering Applications of Artificial Intelligence, vol. 26(1), January 2013, pp. 633-651.

Thai Quang LE was born in Khanh Hoa Province, Vietnam in 1993. He received his university degree from Ritsumeikan Asia Pacific University (APU) in the field of Business Administration focusing on Strategic Management. He has received numerous scholarships including, APU Tuition Reduction Scholarship from 2011-2015, JASSO (Japan Student Services Organization) Scholarship from 2011-2012, and Oita Prefecture Scholarship in 2012-2013, 2014-2015. His research interests include applied programming in business, modeling for decision-making and machine learning, on which he has carried out a few presentations. The theme of his thesis is retail activity optimization and he has become an IEEE member since 2014.

Davar Pishva is a professor in ICT at the College of Asia Pacific Studies, Ritsumeikan Asia Pacific University (APU) Japan. In teaching, he has been focusing on information security, technology management, VBA for modelers, structured decision making and carries out his lectures in an applied manner. In research, his current interests include biometrics; e-learning, environmentally sound and ICT enhanced technologies. Dr. Pishva received his PhD degree in System Engineering from Mie University, Japan. He is Secretary General of IAAPS (International Association for Asia Pacific Studies), Senior Member of IEEE, and a member of IEICE (Institute of Electronics Information & Communication Engineers), IAAPS and University & College Management Association.

A WSN-Based Prediction Model of Microclimate in a Greenhouse Using Extreme Learning Approaches

Qi Liu*, Dandan Jin*, Jian Shen*, Zhangjie Fu**, Nigel Linge***

* College of Computer and Software, Nanjing University of Information Science and Technology, Nanjing, Jiangsu, China

** Jiangsu Engineering Centre of Network Monitoring, Nanjing University of Information Science and Technology, Nanjing, Jiangsu, China

*** The University of Salford, Salford, Greater Manchester, UK

qi.liu@nuist.edu.cn, 1005949332@qq.com, s_shenjian@126.com, wwwfzj@126.com, n.linge@salford.ac.uk

***Abstract*—Monitoring and controlling microclimate in a greenhouse becomes one of the research hotspots in the field of agrometeorology, where the application of Wireless Sensor Networks (WSN) recently attracts more attentions due to its features of self-adaption, resilience and cost-effectiveness. Present microclimate monitoring and control systems achieve their prediction by manipulating captured environmental factors and traditional neural network algorithms; however, these systems suffer the challenges of quick prediction (e.g. hourly and even minutely) when a WSN network is deployed. In this paper, a novel prediction method based on an Extreme Learning Machine (ELM) algorithm and KELM (Kernel based ELM) is proposed to predict the temperature and humidity in a practical greenhouse environment in Nanjing, China. Indoor temperature and humidity are measured as data samples via WSN nodes. According to the results, our approach (0.0222s) has shown significant improvement on the training speed than Back Propagation (BP) (0.7469s), Elman (11.3307s) and Support Vector Machine (SVM) (19.2232s) models, the accuracy rate of our model is higher than those models. In the future, research on faster learning speed of the ELM and KELM based neural network model will be conducted.**

***Keyword*—Wireless Sensor Networks; Kernel based Extreme Learning Machine; Greenhouse Microclimate; Prediction Model**

I. INTRODUCTION

MODERN greenhouses provide a suitable indoor microclimate meeting the requirements of plant growth. A prediction model of the microclimate in a greenhouse therefore becomes critical for the establishment of control strategies and consequent evaluation [1]. The design of such a model becomes challenging due to the features of the greenhouse microclimate, i.e. nonlinear, multiple input multiple output, and its strong coupling between relevant factors. It is also affected by the indoor and outdoor climate environment, crops grown inside and movements of control facilities [2]. All reasons above make it difficult to establish a precise mathematical model to achieve fast and accurate prediction [3]. With rapid development of short-range wireless communication, e.g. Wireless sensor networks (WSN), real-time (or nearly real-time) collection of relevant environmental data in a greenhouse turns out to be convenient, but also raises new challenges on microclimate prediction [4].

Mechanism modelling method based on energy balance because of its multiple parameters and low accuracy, it is difficult to meet the need of practical application [1]. Another system identification method based on the data input and output because it needs less parameters, good adaptive ability and higher simulation accuracy [1-3]. And it has been widely applied. The traditional neural network algorithm, Back Propagation (BP) was applied to build the prediction model of microclimate in a greenhouse. Support Vector Learning was also used for the prediction [5-6]. However, the existence of slow training speed, easy to fall into local minima and the choice of learning rate sensitive etc. inherent shortcomings in the neural network makes its application in the greenhouse prediction model is not ideal.

Extreme Learning Machine (ELM) algorithm [7] is a new algorithm for these mentioned shortcomings of the neural networks, and it has the advantages of faster training speed, the global optimal solution and good generalization. In addition, Kernel Extreme Learning Machine (KELM) has applied kernel function algorithm to the ELM [8]. In this paper, the ELM and KELM algorithms are used to establish

Manuscript received October 1, 2015. This work is supported by the NSFC (61300238, 61300237, 61232016, U1405254, 61373133), Basic Research Programs (Natural Science Foundation) of Jiangsu Province (BK20131004), Scientific Support Program of Jiangsu Province (BE2012473) and the PAPD fund.

Q. Liu is with the College of Computer and Software, Nanjing University of Information Science and Technology, Nanjing, 210044, China (e-mail: qi.liu@nuist.edu.cn).

D. Jin is with the College of Computer and Software, Nanjing University of Information Science and Technology, Nanjing, 210044, China (e-mail: 1005949332@qq.com).

S. Shen is with the College of Computer and Software, Nanjing University of Information Science and Technology, Nanjing, 210044, China (e-mail: s_shenjian@126.com).

Z. Fu is with the Jiangsu Engineering Centre of Network Monitoring, Nanjing University of Information Science and Technology, Nanjing, 210044, China (corresponding author; e-mail: wwwfzj@126.com).

N. Linge is with the School of Computing, Science and Engineering, University of Salford, Salford, M5 4WT, UK (E-mail: n.linge@salford.ac.uk).

prediction models of microclimate in a greenhouse.

This paper presents the WSN-based prediction model of microclimate in a greenhouse using both ELM and KELM algorithms is structured as follows: Section II introduces the related work. Section III presents the principles of ELM algorithm and KELM algorithms and the principles of modelling the greenhouse environment using the ELM and KELM algorithms. Section IV presents the experimental results from real-world data and discussion, and Section V presents conclusions.

II. Related Work

The system identification method based on input-output data needs less parameter and obtains high simulation precision. Patril et al. [9] used the auto-regression and neural network to build a temperature model of tropical greenhouse. A greenhouse microclimate model was built based on neural network in [10], where it was found that outdoor wind and temperature is not vital input factors for the greenhouse microclimate model in summer. Indoor temperature was gathered in a greenhouse in [11], where relative humidity, the intensity of solar radiation and wind speed was collected as input items. In [12], an environment factor model was established in a greenhouse to predict its microclimate based on a fuzzy neural network. Wang et al. [13] used BP algorithm to establish the rainy season greenhouse microclimate model in Jianghuai area, and results show that the model has higher precision and is a beneficial supplement to the physical model. Ferreira [14] used Radical Basis Function (RBF) neural networks to establish fitting model of a hydroponic greenhouse, and obtained the very good fitting results. Fourati [15] used an Elman neural network to emulate the direct dynamics of a greenhouse, the Elman model was used to train control model. Another identification method of nonlinear systems, support vector machine regression (SVMR) was applied to the modelling of greenhouse microclimate system areas, such as online modelling method of weighted least squares support vector machine based on [16]. In [17], indoor data, such as inner temperature, humidity, wind speed, solar radiation intensity, etc. were gathered, so that a greenhouse microclimate humidity model was designed for the prediction in the north of china in winter by using a BP neural network improved by a genetic algorithm.

III. Prediction Model

A. Principles of ELM

For a given set of random samples, $S_Q = \{(\boldsymbol{x}_i, t_i)\}_{i=1}^{Q}$, where $\boldsymbol{x}_i = [x_i, x_{i+1}, \ldots, x_{i+n-1}]$, $t_i = x_{i+n}$, x_i is the input vector, $\boldsymbol{t_i}$ is the output corresponding to x_i, n is the embedded dimension. An ELM regression model containing the L hidden layer neurons can be expressed as in [18]:

$$\sum_{i=1}^{L} \beta_i f(\boldsymbol{\omega}_i \boldsymbol{x}_j + b_i) = t_j, \qquad j = 1,2,\ldots,Q \tag{1}$$

where Q is the number of samples in the training set, ω_i is the input weights between the i[th] neuron and the input layer, and $\omega_i = [\omega_{i1}, \omega_{i2}, \ldots, \omega_{in}]$, β_i is the output weights between the i[th] neuron and the output layer, b_i is the threshold values of the i[th] neuron. Eq. (1) can be written in a matrix form as:

$$\boldsymbol{H\beta} = \boldsymbol{T}' \tag{2}$$

where $\boldsymbol{H}$ is an output matrix of the hidden layer, can be written specifically as:

$$\mathrm{H} = \begin{bmatrix} f(\boldsymbol{\omega}_1 \boldsymbol{x}_1 + b_1) & \ldots & f(\boldsymbol{\omega}_L \boldsymbol{x}_1 + b_L) \\ \vdots & \vdots & \vdots \\ f(\boldsymbol{\omega}_1 \boldsymbol{x}_Q + b_1) & \ldots & f(\boldsymbol{\omega}_L \boldsymbol{x}_Q + b_L) \end{bmatrix}_{Q \times L} \tag{3}$$

$\boldsymbol{\beta}$ is an output weight, which can be written specifically as: $\beta = [\beta_1, \beta_2, \cdots, \beta_L]^T$.

$\boldsymbol{T}$ is an output weight, which can be written specifically as: $\mathrm{T} = [t_1, t_2, \cdots, t_L]^T$.

In most cases of Eq. (3), $\boldsymbol{Q}$ is much greater than $\boldsymbol{L}$. By solving Eq. (2), the output weights $\boldsymbol{\beta}$ can be calculated as in [19]:

$$\hat{\boldsymbol{\beta}} = \boldsymbol{H}^{+}\boldsymbol{T} \tag{4}$$

where H^{+} is the Moore-Penrose generalized inverse of the hidden layer output matrix $\boldsymbol{H}$, and it can be calculated as: $\mathbf{H}^{+} = (\mathbf{H}^{\mathrm{T}}\mathbf{H})^{-1}\mathbf{H}^{\mathrm{T}}$.

Therefore, after training the final prediction model using ELM can be written as:

$$\mathrm{t} = \sum_{i=1}^{L} \beta_i f(\boldsymbol{\omega}_i \boldsymbol{x} + b_i) \tag{5}$$

where $\boldsymbol{x}$ is the input vector of prediction model using ELM, $\boldsymbol{t}$ is the output vector of prediction model using ELM.

B. Principles of KELM

Due to the fact that two thresholds, w and b in the ELM model are generated randomly, the performance of an ELM based prediction model suffers from its poor stability. KELM is therefore designed in order to introduce a stable kernel for the ELM algorithm for constant fitness [8].

The KELM algorithm can be summarized as follows:

Input: the training set $(\boldsymbol{x}_i, \boldsymbol{t}_i)$, the kernel function $\boldsymbol{K}$, the ridge regression parameter $\boldsymbol{C}$.

Output: prediction of test set $\boldsymbol{f(x)}$.

Step 1. Calculate kernel matrix $\boldsymbol{K}$:

$$\boldsymbol{\Omega}_{ELM} = \boldsymbol{H}\boldsymbol{H}^{T}: \ \Omega_{ELM_{ij}} = h(\boldsymbol{x}_i) \cdot h(\boldsymbol{x}_j) = K(\boldsymbol{x}_i, \boldsymbol{x}_j)$$

Step 2. Calculate matrix inverse:

$$(\frac{\boldsymbol{I}}{\boldsymbol{C}} + \Omega_{ELM})^{-1}$$

Step 3. Calculate the mapping of test data set:

$$\boldsymbol{K}_x = \begin{bmatrix} K(\boldsymbol{x}, \boldsymbol{x}_1) \\ \vdots \\ K(\boldsymbol{x}, \boldsymbol{x}_N) \end{bmatrix}$$

Step 4. Calculate the predicted results:

$$\boldsymbol{f}(\boldsymbol{x}) = \boldsymbol{K}_x (\frac{\boldsymbol{I}}{\boldsymbol{C}} + \Omega_{ELM})^{-1} \boldsymbol{T}$$

C. Prediction Model of Microclimate in a Greenhouse Using ELM and KELM

Greenhouse microclimate is an extremely complicated system, influenced by outdoor environmental factors, the structures of greenhouse and the operation status of

environmental control equipment, etc. So the historical sample of indoor temperature and humidity imply the above information. Hence in order to predict temperature and humidity in greenhouse that can only use the historical sample of indoor temperature and humidity to establish prediction models. While establishing prediction model To predict the next sample prediction model uses the three past samples: $s_{t-3}, s_{t-2}, s_{t-1}$, and the current sample s_t. Hence, the t[th] input-output instance is:

$$x_t = [s_{t-3}, s_{t-2}, s_{t-1}, s_t], \; t_t = s_{t+1}$$

where **s** can be temperature or humidity value.

The ELM algorithm can be represented as follows:

Step 1. Initialization. Set the number of the neurons and activation function in the hidden layer, randomly generate input weights **ω** and the threshold value **b**;

Step 2. Use the parameters obtained in Step 1 and the input matrix of the training set to calculate the output matrix H of the hidden layer;

Step 3. Use H and T to calculate the output weights $\hat{\beta}$: $\hat{\beta} = H^{+}T$;

Step 4. Use Eq. (5) to calculate the predicted results.

IV. Results and Discussion

A. WSN

A weather observation system is implemented based on a ZigBee enabled wireless sensor network. Customised sensor nodes are designed providing four open communication interfaces for wide compatibility of sensors. The nodes also support multiple wireless data communication methods including free short-range ISM radio bands at 2.4GHz and 915MHz, as well as enhanced mobile telecommunication technologies, e.g. EDGE and HSDPA. The prototype design, hardware block diagram and PCBs of the sensor node are shown in Fig. 1.

A closed polycarbonate casing design makes the dust particle and liquid ingress protection of our nodes reach IP6x and IPx5 respectively. Four generic communication interfaces are offered to accept third-party analogue and digital sensors. Adapters/converters, shown in the perception part of Fig. 1(c) have been organised and fitted into the corresponding pins of the interfaces, as depicted in Fig. 2.

B. Results of the predicting using the ELM model

To establish the temperature model, the training parameters of ELM were as follows: the number of neurons in the hidden layer is 26 and the activation function of neurons in the hidden layer is sin. In order to meet the requirements of the prediction model, all data have been normalized to range [0, 1] according to Eq. (6):

$$s_t = (s_t - s_{min})/(s_{max} - s_{min}) \quad (6)$$

where s_{min} is the minimum number of sample series, s_{max} is the maximum number of sample series.

Establishing humidity model used the same method with temperature model, the difference is the sample set, so establishing humidity model was not described in detail.

The fitting results of using the ELM algorithm to predict the temperature and humidity have been done through simulation tests using Matlab. The results are shown in Fig. 3 and Fig. 4, where the red line indicates the actual value, and the blue line indicates the predicted value. As can be seen from the figures, using ELM algorithm to predict the greenhouse environmental factors basically reached the expected results. The models have higher accuracy, with a good fit between the predictive values and the actual values. This shows that the simulation of greenhouse environment factors using ELM algorithm is effective and can meet the needs of agricultural production. And, the effect in the temperature simulation is more superior.

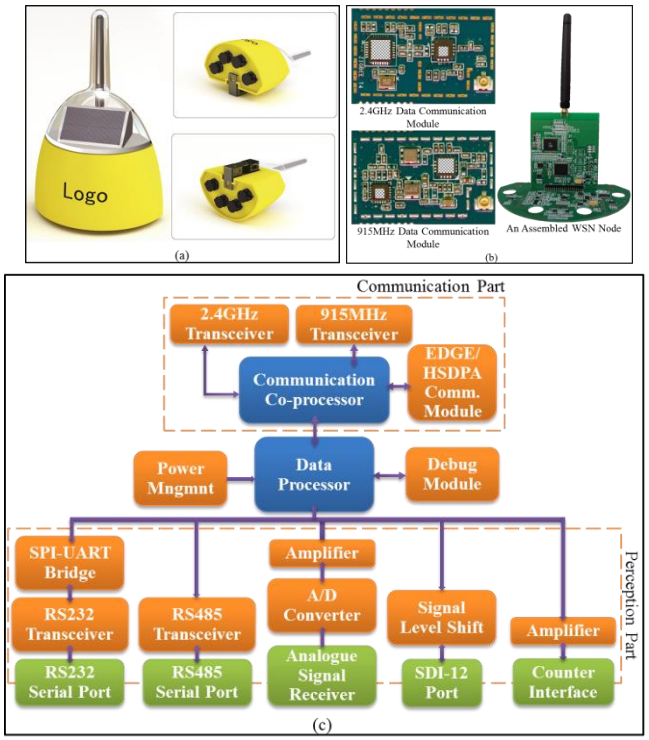
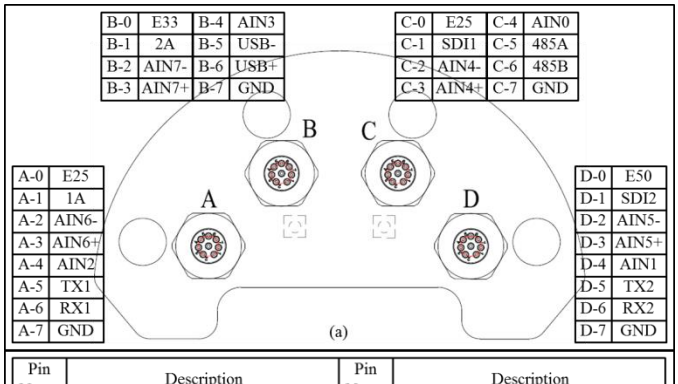


Fig. 1. The hardware implementation of a customised WSN node. (a) The prototype design; (b) the PCB design of communication module and the entire assembled design of a node; (c) the block diagram.



Pin Name	Description	Pin Name	Description
Exx	Power Output*	TXx	UART Tx**
xA	Counter Input**	RXx	UART Rx **
AINx-	Negative Differential Analog Input**	USB-	USB Negative Signal
AINx+	Positive Differential Analog Input**	USB+	USB Positive Signal
AINx	Analog Input**	SDIx	SDI-12 data signal **
GND	Ground	485	RS485 signal

*: "xx" specifies the output voltage level
**: 'x' specifies the sequence number

(b)

Fig. 2. The pin arrangement of four interfaces. (a) Pin numbers and their names specified in four pin tables; (b) the pin names and their corresponding description.

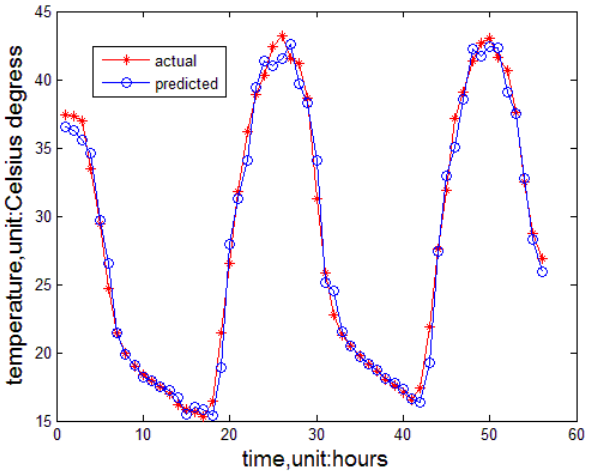


Fig. 3. The curve fitting of predicted and actual data for the greenhouse temperature.

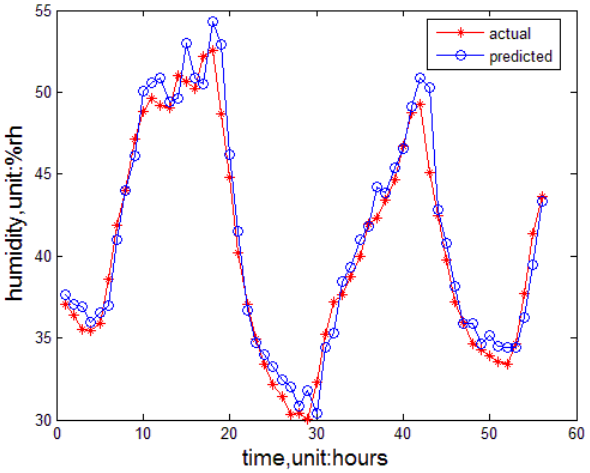


Fig. 4. The curve fitting of predicted and actual data for the greenhouse humidity.

C. *Results of the predicting using the KELM model*

Same history samples as from the ELM were used for the KELM model. Related parameters for Genetic Algorithms (GA) configuration to Optimize KELM learning parameters were set as well, as shown in Table I.

TABLE I
GA SETTINGS

GA Parameters	Value
Max. Population No.	20
Max. Evolution Generation	200
Gap Rate	0.9
Crossover Probability	0.7
Mutation Probability	0.07

Fig. 5 and 6 depict that when using genetic algorithm to optimize KELM learning parameters, the optimal fitness value of GA can quickly reached extremes. It can also be used to find the optimal learning combination (C, σ), which makes the fitting performance of KELM prediction model optimal.

D. *Discussion*

In order to better evaluate the performance of the ELM models, three algorithms are used to compare with ELM. Three algorithms were selected, i.e. BP, Elman, and SVM. These models have same training and testing sets with ELM models, the input and output data also take the same way with ELM models.

BP neural network has 5 hidden neurons of temperature and humidity model, the maximum number of iterations is 1000, the learning rate is 0.2, and the mean squared error goal is 0.0001, other parameters default. Elman neural network has 13 hidden neurons of temperature model and 18 hidden neurons of humidity model, the hidden layer transfer function is the tansig, the output layer transfer function is the purelin, and the mean squared error goal is 0.0001, other parameters default. Kernel function of SVM model is RBF. In temperature Model penalty factor parameter c = 16, variance g= 0.125, in humidity model c = 11.3137, g = 0.353553, other parameters are set as default values.

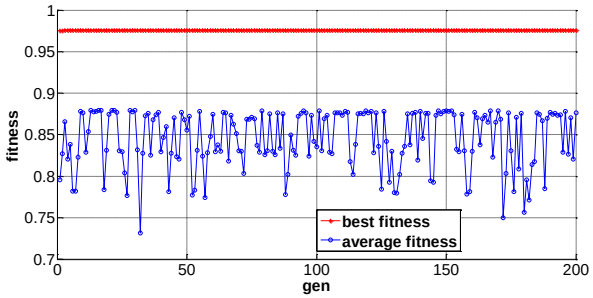


Fig. 5. The Fitness of temperature using GA to optimize KELM parameters.

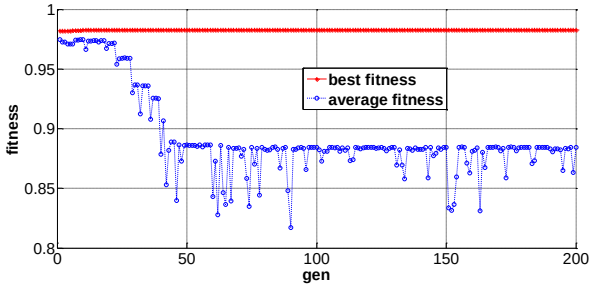


Fig. 6. The Fitness of Humidity using GA to optimize KELM parameters.

Table II shows the results of predicting the environment factors using three algorithms just mentioned. Since the input weights and thresholds are generated randomly, for without loss of generality, performance evaluation parameters for all algorithms were taken average of 50 times as the results. The performance evaluation parameters are training time, root mean square error (RMSE) and the coefficient of determination (R^2). And the smaller training time, the smaller RMSE and the higher R2 indicated a better performance of the model. Specific calculation formulas are shown as Eq. (7) and (8):

$$\mathrm{RMSE} = \sqrt{\frac{1}{l}\sum_{i=1}^{l}(\hat{t}_i - t_i)^2} \tag{7}$$

$$R^2 = \frac{\left(l\sum_{i=1}^{l}\hat{t}_i t_i - \sum_{i=1}^{l}\hat{t}_i \sum_{i=1}^{l} t_i\right)^2}{\left(l\sum_{i=1}^{l}\hat{t}_i{}^2 - (\sum_{i=1}^{l}\hat{t}_i)^2\right)\left(l\sum_{i=1}^{l} t_i{}^2 - (\sum_{i=1}^{l} t_i)^2\right)} \tag{8}$$

where, l is the number of testing samples, $t_i (i = 1,2,\ldots,l)$ is the actual value for the i[th] sample, $\hat{t}_i (i = 1,2,\ldots,l)$ is the predicted value for the i[th] sample.

Seen from Table II, the ELM algorithm ran around 33 times faster than BP, 510 times faster than Elman, and 865 times faster than SVM for predicting the temperature in the greenhouse. So the ELM model showed great superiority in training speed. At the same time, the RMSE of ELM model

are lower than and R^2 was higher than BP, Elman and SVM model, the ELM model showed the high accuracy and fitting ability. The ELM model of humidity also exhibited the same advantages. It can also been that ELM models can quickly predict the greenhouse environmental factors in the context of maintaining the accuracy. Therefore, ELM model is more suitable to predict the greenhouse environmental factors.

From Table III, it can be seen the KELM model outperforms the training speed and generalization ability. First, the train speed of the KELM model is 2.9 times faster than the ELM model. In addition, the standard deviation of the KELM model is 0, showing its stability. The results on humidity have shown similar trends. Compared to BP, Elman and SVR, the KELM has also depicted better performance, as shown in Table IV.

TABLE II
PERFORMANCE COMPARISON OF THE ELM, BP, ELMAN AND SVM ALGORITHMS

Environment factors	Algorithms	Training Time(seconds)	RMSE	R2	Nodes of Neurons
temperature	ELM	0.0222	1.0586	0.9883	26
	BP	0.7469	1.1841	0.9858	5
	Elman	11.3307	1.1840	0.9853	13
	SVM	19.2232	1.1537	0.9865	-
humidity	ELM	0.0187	1.4177	0.9648	18
	BP	0.6833	1.6398	0.9586	5
	Elman	11.5784	1.4191	0.9641	18
	SVM	20.0633	1.4635	0.9605	-

TABLE III
PERFORMANCE COMPARISON OF THE ELM AND KELM ALGORTHAMS

Environment Factors	Algorithms	Parameter	Value	Training Time (seconds)	$R^2 \pm SD$
temperature	KELM	(C，σ)	($2^{16.1170}$，$2^{-6.1797}$)	0.0023	0.9762±0
	ELM	L	13	0.0068	0.9749±0.0019
humidity	KELM	(C，σ)	($2^{999.9870}$，$2^{1.2519}$)	0.0026	0.9829±0
	ELM	L	27	0.0069	0.9828±0.0003

TABLE IV
PERFORMANCE COMPARISON OF THE ELM, BP, ELMAN AND SVR ALGORITHMS

Environment Factors	Algorithms	Parameter	Value	Training Time (seconds)	$R^2 \pm SD$
temperature	KELM	(C，σ)	($2^{16.1170}$，$2^{-6.1797}$)	0.0023	0.9762±0
	BP	L	5	0.9126	0.9727±0.0033
	Elman	L	10	15.8528	0.9748±0.0009
	SVR	(C，σ)	($2^{1.4142}$，$2^{1.4142}$)	0.0318	0.9731±0
humidity	KELM	(C，σ)	($2^{999.9870}$，$2^{1.2519}$)	0.0026	0.9829±0
	BP	L	5	0.9999	0.9809±0.0026
	Elman	L	12	15.0532	0.9807±0.0006
	SVR	(C，σ)	(2^{16}，$2^{1.4142}$)	0.0330	0.9827±0

V. CONCLUSION

This paper applied ELM algorithm and KELM algorithm to predict the greenhouse environmental factors. Different from the traditional learning algorithms, ELM algorithm randomly generated input weights and thresholds, and simply set the number of hidden layer neurons, we can obtain the unique global optimal solution. The algorithm is simple, fast and high simulation precision. Comparison of BP, Elman and SVM algorithms in environmental factors prediction, ELM showed better performance. It proved that it is feasible to use ELM algorithm to predict environmental factors, and which can provide support for the intelligent control of greenhouse. Compared with the prediction model based on the ELM, the refined model based on the KELM depicts better results on computing speed and accuracy, with the stability of the model being maintained. Compared with the prediction models based on BP, Elman and SVR, the KELM model requires less training time, but shows stronger fitness and more stable performance.

Because of the influence of environment factors in greenhouse by different types of greenhouse structure and material, the types of crops and planting mode, the weather changes, human disturbance and run state of control equipment and many other factors, the steady model to predict the environmental factor is inappropriate. In future research, the model will be improved further. Building an online model of greenhouse environment factors is very necessary. Moreover, if shortening time interval (e.g. 5min, 10min), whether or not to make model more perfect, there is need for further study.

REFERENCES

[1] L. Qin and G. Wu, "The present situation and prospect of modeling and control of the greenhouse microclimate", *Automation Panorama*, vol. 2010, no. 2, pp. 58 - 64, 2010. (Chinese)

[2] N. Bennis, J. Duplaixb, G. Enéab, M. Halouac and H. Youlal, "Greenhouse climate modelling and robust control", *Computers and electronics in agriculture*, vol. 61, no. 2, pp. 96-107, 2008.

[3] J. P. Coelho, P. B. de Moura Oliveira and J. B. Cunha, "Greenhouse air temperature predictive control using the particle swarm optimisation algorithm", *Computers and Electronics in Agriculture*, vol. 49, no. 3, pp. 330-344, 2005.

[4] R. Pahuja, H. K. Verma and M. Uddin, "A Wireless Sensor Network for Greenhouse Climate Control", *IEEE Pervasive Computing*, vol. 12, no. 2, pp. 49-58, 2013.

[5] B. Gu, V. S. Sheng, K. Y. Tay, W. Romano and S. Li, "Incremental Support Vector Learning for Ordinal Regression", *IEEE Transactions on Neural Networks and Learning Systems*, vol. 26, no.7, pp. 1403-1416, 2015.
[6] B. Gu, V. S. Sheng, Z. Wang, D. Ho, S. Osman and S. Li, "Incremental learning for ν-Support Vector Regression", *Neural Networks*, vol. 67, pp. 140-150, 2015.
[7] G.-B. Huang, Q.-Y. Zhu and C.-K. Siew, "Extreme learning machine: a new learning scheme of feedforward neural networks", *IEEE Proceedings of International Joint Conference on Neural Networks*, pp.985-990, 2004.
[8] G.-B. Huang and C.-K. Siew, "Extreme learning machine with randomly assigned RBF kernels", *International Journal of Information Technology*, vol. 11, no. 1, pp. 16-24, 2005.
[9] S. L. Patil, H. J. Tantau and V. M. Salokhe, "Modelling of tropical greenhouse temperature by auto regressive and neural network models", *Biosystems Engineering*, vol. 99, no. 3, pp. 423-431, 2008.
[10] I. Seginer, T. Boulard and B. J.Bailey, "Neural network models of the greenhouse climate", *Journal of Agricultural Engineering Research*, vol. 59, no. 3, pp. 203-216, 1994.
[11] J. B. Cunha, C. Couto and A. E. Ruano, "Real-time parameter estimation of dynamic temperature models for greenhouse environmental control", *Control Engineering Practice*, vol. 5, no. 10, pp. 1473-1481, 1997.
[12] B. T. Tien and G. Van Straten, "A Neuro-Fuzzy approach to identify lettuce growth and greenhouse climate", *Artificial Intelligence Review*, vol. 12, no. 1, pp. 71-93, 1998.
[13] X. Wang, W. Ding, W. Luo and J. Dai, "Simulation and analysis of micro-climate of gutter connected Venlo greenhouse during rainy season in Jianghuai region of China using BP neural network", *Transactions of the Chinese Society of Agricultural Engineering*, vol. 20, no. 2, pp. 235-238, 2004. (Chinese)
[14] P. M. Ferreira, E. A. Faria and A. E. Ruano, "Neural network models in greenhouse air temperature prediction", *Neurocomputing*, vol. 43, no. 1, pp. 51-75, 2002.
[15] F. Fourati, "Multiple neural control of a greenhouse", *Neurocomputing*, vol. 139, pp. 138-144, 2014.
[16] D. Wang, M. Wang and X Qiao, "Support vector machines regression and modeling of greenhouse environment", *Computers and electronics in agriculture*, vol. 66, no. 1, pp. 46-52, 2009.
[17] F. He and C. Ma, "Genetic algorithm to optimize neural network model in the application of solar greenhouse humidity forecast", *China agriculture bulletin*, vol. 24, no. 1, pp. 492-495, 2008. (Chinese)
[18] G.-B. Huang, H. Zhou, X. Ding and R. Zhang, "Extreme learning machine for regression and multiclass classification", *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics*, vol. 42, no. 2, pp. 513-529, 2012.
[19] G.-B. Huang, D.-H. Wang and Y. Lan, "Extreme learning machines: a survey", *International Journal of Machine Learning and Cybernetics*, vol. 2, no. 2, pp. 107-122, 2011.

Qi Liu (M'11) received his BSc degree in Computer Science and Technology from Zhuzhou Institute of Technology, China in 2003, and his MSc and PhD in Data Telecommunications and Networks from the University of Salford, UK in 2006 and 2010. His research interests include context awareness, data communication in MANET and WSN, and smart grid. His recent research work focuses on intelligent agriculture and meteorological observation systems based on WSN.

Dandan Jin received her bachelor's degree in Software Engineering from Nanjing University of Information, Science and Technology in 2015, and she will continue to pursue a master's degree in computer science and technology at the Nanjing University of Information Science and Technology. Her research interests include smart grid and greenhouse microclimate modeling.

Jian Shen received his bachelor's degree in Electronic Science and Technology Specialty from Nanjing University of Information, Science and Technology in 2007, and he received his masters and PhD in Information and communication from CHOSUN University, South Korean in 2009 and 2012. His research interests includes Computer network security, information security, mobile computing and network, wireless ad hoc network.

Zhangjie Fu received his BS in education technology from Xinyang Normal University, China, in 2006; received his MS in education technology from the College of Physics and Microelectronics Science, Hunan University, China, in 2008; obtained his PhD in computer science from the College of Computer, Hunan University, China, in 2012. Currently, he works as an assistant professor in College of Computer and Software, Nanjing University of Information Science and Technology, China. His research interests include cloud computing, digital forensics, network and information security.

Nigel Linge received his BSc degree in Electronics from the University of Salford, UK in 1983, and his PhD in Computer Networks from the University of Salford, UK, in 1987. He was promoted to Professor of Telecommunications at the University of Salford, UK in 1997. His research interests include location based and context aware information systems, protocols, mobile systems and applications of networking technology in areas such as energy and building monitoring.

Security Middleware Infrastructure for Medical Imaging System Integration and Monitoring

Weina Ma, Kamran Sartipi

Department of Electrical, Computer and Software Engineering, University of Ontario Institute of Technology, 2000 Simcoe St N, Oshawa, Ontario, Canada

{Weina.Ma, Kamran.Sartipi}@uoit.ca

***Abstract*— With the increasing demand for electronic medical records sharing, it is a challenge for medical imaging service providers to protect the patient privacy and IT infrastructure security in an integrated environment. In this paper, we present a novel security middleware infrastructure for seamlessly and securely linking legacy medical imaging systems, diagnostic imaging web applications as well as mobile applications. In this infrastructure, software agents such as user agent and security agent have been integrated into medical imaging domains that can be trained to perform their tasks. The proposed security middleware utilizes both online security technologies such as authentication, authorization and accounting, as well as post security operations to discover system security vulnerability. By integrating with the proposed security middleware, both legacy system users and Internet users can be uniformly identified and authenticated; access to patient diagnostic images can be controlled based on patient's consent directives and other access control polices defined at a central point; relevant user access activities can be audited at a central repository; user access behavior patterns are studied by utilizing data mining techniques; the explored behavior patterns provide system administrators valuable knowledge to refine existing security policies; behavior-based access control is enforced by capturing user's dynamic behavior and determining their access rights through comparing with the discovered knowledge of common behaviors. A case study is presented based on the proposed infrastructure.**

***Keyword*—Behaviour Pattern, Data Mining, Medical Diagnostic Imaging, Middleware, Security**

I. INTRODUCTION

MORDEN Diagnostic Imaging (DI) solutions maintain and manage patient radiology images (e.g., CT scans, X-ray, MRI, ultrasound), and corresponding diagnostic reports in digital formats, for the purpose of diagnosis, treatment improvement and medical science research. Over the past decades, Picture Archiving and Communication Systems (PACS) have taken a dominant role in the workflow of DI solutions in a single hospital or radiology department. A federated DI domain allows for a centralized capture, long-term archiving and non-proprietary sharing of radiology information across a large distributed network. A central diagnostic imaging repository (DI-r) provides common services to the participating hospitals. According to the status of DI-r projects across Canada [1], 19 provincial DI-r's have been developed or being developed to reliably maintain, deliver and share DI information to consumers within the electronic health record (EHR) systems. Meanwhile, mobile health information technology (mHealth) is increasingly important in telemedicine, but traditional security infrastructure deployed in PACS and DI-r systems is not ready for accessing DI records through mobile devices.

Integrating the Healthcare Enterprise (IHE) has developed a number of integration profiles [2], [3] that address security requirements to improve the way computer systems in healthcare share information. These security control requirements are achieved through a trusted model where each local medical imaging system is responsible for ensuring that the personal health information is adequately protected. A key challenge with this trusted model is the lack of federated capabilities: i) access control rules are local to each system, which means consistency of access rules across all systems has to be managed manually; ii) patient consent directives and their impact on access control are not communicated automatically to each system; iii) user authentication is local to each system that imposes a significant administrative burden to ensure that individuals are uniformly identified in each system; iv) access to data is audited in each local system which also imposes a significant burden to investigate inappropriate access or monitor security breaches.

Middleware is a software layer that lies between service providers and consumers in a distributed computer network. Our proposed security middleware enables secure radiology image sharing among different provincial DI-r's, heterogenous PACS systems in distributed hospitals, as well as web clients and mobile clients. The main objective of this study is to propose an infrastructure for development of security middleware that provides: online security mechanism

Manuscript received October 12, 2015. This work is sponsored by an Ontario Research Fund (ORF) grant, and a follow-up of the invited journal to the accepted out-standing conference paper of the 17th International Conference on Advanced Communication Technology (ICACT2015). Grant ID: RE-05-073. Project: Secure Intelligent Content Delivery System for Timely Delivery of Large Data Sets in a Regional/National Electronic Health Record. This research was conducted with collaboration of Dr. David Koff and Dr. Peter Bak at MIIRCAM Centre of McMaster University.

W. M. is with Department of Electrical, Computer and Software Engineering, University of Ontario Institute of Technology, 2000 Simcoe St N, Oshawa, ON L1H 7K4, Canada (corresponding author, phone: +1-905-721-8668, fax: 905-721-3178, e-mail: Weina.Ma@uoit.ca).

K. S. is with Department of Electrical, Computer and Software Engineering, University of Ontario Institute of Technology, 2000 Simcoe St N, Oshawa, ON L1H 7K4, Canada (e-mail: Kamran.Sartipi@uoit.ca).

such as common authentication and authorization methods; post security mechanism that assists system administrators in exploring user access behavior patterns by mining audit logs; and applying behavior based access control by capturing user's dynamic behavior, and determining access rights through comparing with the discovered common behaviors. In this context, the main contributions of this paper include: i) designing middleware architecture for seamlessly and securely integrating legacy medical imaging systems; ii) proposing a behavior-based technique which allows to detect outlier behaviors and enhance the system's access control policies; iii) presenting a new method to measure behavior similarity and outlier degree; and iv) introducing generic software agents which can be customized and trained to perform the assigned tasks (e.g., access control, or auditing).

The remaining of this paper is organized as follows: Related work is discussed in Section II. Section III presents the proposed infrastructure of security middleware, and user behavior monitoring. Section IV is allocated to a case study, and finally conclusion is presented in Section V.

II. RELATED WORK

IHE is an initiative by healthcare professionals and industry which aims at setting up consolidated healthcare information sharing through standards based approaches [4]. It guides enterprises in using established standards to achieve interoperability based on existing IT infrastructure. However, the IHE suggested trust model in cross-enterprise domains lacks federated capabilities. Also, the small and medium scale medical service providers lack the proper skills and technology to make reliable and accurate authorization decision independently, especially in cloud and mobile computing environments. In such context, we introduce a security middleware that provides one common method for integrating a broad range of medical service providers.

A software agent is a program that acts on behalf of an agency for different users or other programs. The notion of generic and lightweight agent that resides at client side to be utilized by different service providers is introduced in [5]. The agents can be customized and trained based on the service provider generated role description and knowledge to perform the assigned tasks. This technology is an extension of the service-oriented architecture (SOA) model that allows for providing personalized services and maintaining client privacy through processing client's data locally. In our proposed architecture, we use cooperative-agents that reside at both client side and service provider side to interact with the security middleware and perform the assigned tasks.

In an earlier work [6] and [7], we proposed a general and secure infrastructure for sharing medical images between PACS and EHR systems. The proposed environment in that work was based on federated authentication and authorization techniques (OpenID and OAuth) [8], and cooperative agents with dedicated tasks to provide both action-based and behaviour-pattern based access control. As for legacy PACS systems, an agent-based approach [9] is proposed allowing for capturing PACS communication messages, identifying PACS users and extracting user actions to feed into an action-based access control mechanism.

Most of the existing access control models deal only with static systems. Behaviour-based access control for distributed healthcare systems is initially introduced in [10]. The proposed access control model captures the dynamic behavior of the user, and determines access rights through comparing with the expected behavior. Ideally, the distance between observed behavior and expected behavior is significant if the user acts abnormally. This model is also applied in security sharing of medical images [6]. In our proposed architecture, we define a behavior pattern as: *consistent observations of a sequence of actions that a user or a group of users conducted in a common context during a specific time interval (e.g., a session, a day, a week)*. Our work enhanced the behaviour-based access control by proposing a new behaviour similarity metric to determine the closeness between the observed dynamic behaviour and discovered common behaviour, and introducing an outlier degree to detect outliers.

Despite the placement of security mechanisms such as authentication, authorization and secure communication in most systems, authorized users, intended or carelessly, exhibit risky behaviours that may cause data leakage or damage to protected resources. Examining human behaviour among authorized users is helpful in assisting security professionals to make access control decisions. Our proposed security middleware provides: online security services to identity and authorize user access; and post security services to monitor and analyse the authorized user's access behaviour patterns. Such an acquired knowledge can lead administrators to security policy enhancements.

Acquiring decent user access behaviour patterns is crucially important in our approach. We analysed the audit logs of distributed PACS systems, and extracted sequencing, association and timing constraints to represent a behavior pattern: sequencing requires that a series of steps occur in a certain order; timing limits the occurrence frequency of certain values; and association identifies the cases where two or more system values occur at the same time. We employ data mining techniques in user access behaviour discovery. Association rules mining was originally introduced by Agrawal [11], aiming at analyzing customer purchase habits by finding association relations between items in the customer shopping baskets. Sequential pattern mining was also proposed by Agrawal [12], detecting frequently occurring ordered events or subsequence as frequent patterns. There are many applications involving sequenced data, such as customer shopping sequences, web click streams, and biological sequences. Clustering is a method of grouping objects in a way that objects in one cluster are very similar to each other but they are dissimilar to the objects in other clusters [13]. Similarity-based clustering methods define and utilize similarity metrics to determine the closeness between the pairs of objects [14]. We proposed a behavior model based on association, sequencing and time constraints, which utilizes association mining, sequential pattern mining and similarity-based clustering techniques to explore user behaviors from audit logs.

An obvious measure of the closeness of two sequences is to find the maximum number of identical items in those two sequences (preserving the symbol order), which is defined as Longest Common Subsequence (LCS) of the sequences [15]. Formally, let $X=(x_1, x_2, \ldots, x_m)$ and $Y=(y_1, y_2, \ldots, y_n)$ be two sequences of lengths m and n, respectively. A common subsequence cs of X and Y represented by $cs(X, Y)$ is a subsequence that occurs in both sequences. The longest common subsequence lcs of sequence X and Y, $lcs\ (X, Y)$ is a common subsequence of both sequences with maximum length. The length of $lcs(X, Y)$ is denoted by $R(X, Y)$. Solving $R(X, Y)$ is to determine the longest common subsequence for all possible prefix combinations of the two sequences X and Y. Let $r(i, j)$ be the length of the lcs of x_i and y_i, where $x_i = (x_1, x_2, \ldots, x_i)$ and $y_i = (y_1, y_2, \ldots, y_j)$. Then $R(X, Y)$ can be defined recursively as following [14]:

$$r(i,j) = \begin{cases} 0 & if\ i = 0\ or\ j = 0 \\ r(i-1, j-1) + 1 & if\ x_i = y_j \\ \max\{r(i-1, j), r(i, j-1)\} & if\ x_i \neq y_j \end{cases} \quad (1)$$

III. PROPOSED INFRASTRUCTURE

The overall architecture of the proposed security middleware infrastructure for medical imaging system integration and monitoring is shown in Figure 1 and its detailed workflow is shown in Figure 2.

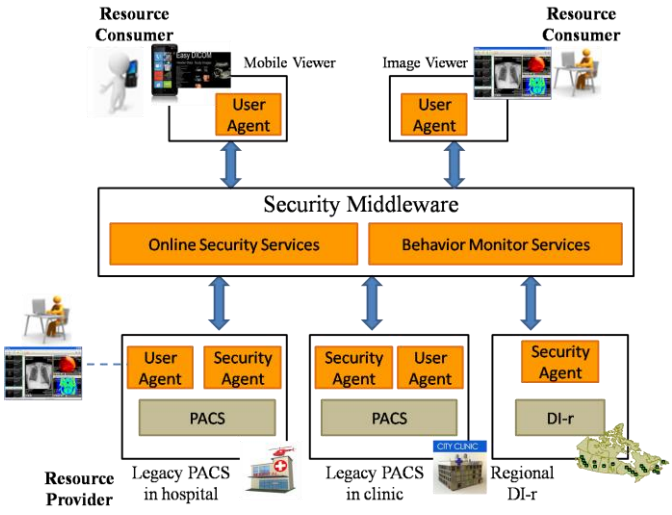


Fig. 1. Architecture for security middleware integration with legacy PACS, DI-r's and client applications

A. Architecture

Figure 1 illustrates the proposed architecture, where the client's access requests can be authorized under different access control models in legacy PACS and DI-r domains, but they are ruled according to the unified access control policies. The Security Middleware monitors and analyses user access behaviour patterns and assists the system administrators in consolidating existing access control policies based on the acquired knowledge from the extracted behaviour patterns. The components of the architecture are as follows.

Resource Consumer, is a medical imaging viewer (including mobile image viewer) that provides quality diagnostic images to the end users. According to the definition of SOA, both provider and consumer are roles that are played by software agents on behalf of their owners.

Resource Provider, is a medical imaging system that provides electronic image storage and convenient access to images from multiple resource consumers.

User Agent, is a software agent that is deployed at the client side to perform authentication request on behalf of the client application (e.g., image viewer) against the Security Middleware.

Security Agent, is a generic agent that is deployed at the server provider side for making access control decisions and collecting information about the user activities. Security Agent is customizable and trainable for different authorization models. The security middleware sends control information (access control polices), training data (authorization model) and assigned tasks (collecting user activity events) to customize and train a Security Agent. Based on the acquired training, assigned tasks, and user's data, Security Agent acts as a local access control mechanism. It also performs some filtering operations on the collected local user activities to allow for the behavior monitoring services at the Security Middleware.

Security Middleware, is an infrastructure that utilizes both online security technologies such as authentication, authorization and accounting, and post security procedures such as association and sequential pattern mining and pattern extraction to monitor users' behaviors.

Online Security Services, supports a set of centralized user directories and provides a common service that handles all user authentication requests. It also provides centralized access control policy management and a set of authorization models. The existing IT infrastructure in legacy domains is operating based on different technologies, procedures and models. It is not necessary to employ exactly the same access control mechanism across these domains, but it is necessary that they agree at the policy level.

Behaviour Monitor Services, provides the mechanism for monitoring the activities within the resource consumer and medical imaging systems. Data mining engines are employed to assist the system administrators obtain deep insight into the user access behavior patterns. With the system administrator's agreement, the discovered behavior pattern knowledge (common behavior) is sent to Security Agent as training data. At the same time, a behavior based access control task is assigned to Security Agent. Security Agent monitors the users' dynamic behaviors and compares with the common behaviours. Security Agent notifies the system administrator if any user behaves significantly different from the identified common behaviours.

A typical PACS system contains: image acquisition devices namely modalities (e.g., CT scan, MRI system); image archives where the acquired images are stored; and workstations where radiologists view the images. Both User Agent (serving workstations and modalities) and Security Agent (serving image archives) are deployed at each PACS system. The DI-r provides registry services for querying patient's medical images from legacy systems, and repository

service for storing and retrieving medical images. Security Agent is deployed to each DI-r system serving such services.

B. Workflow Model

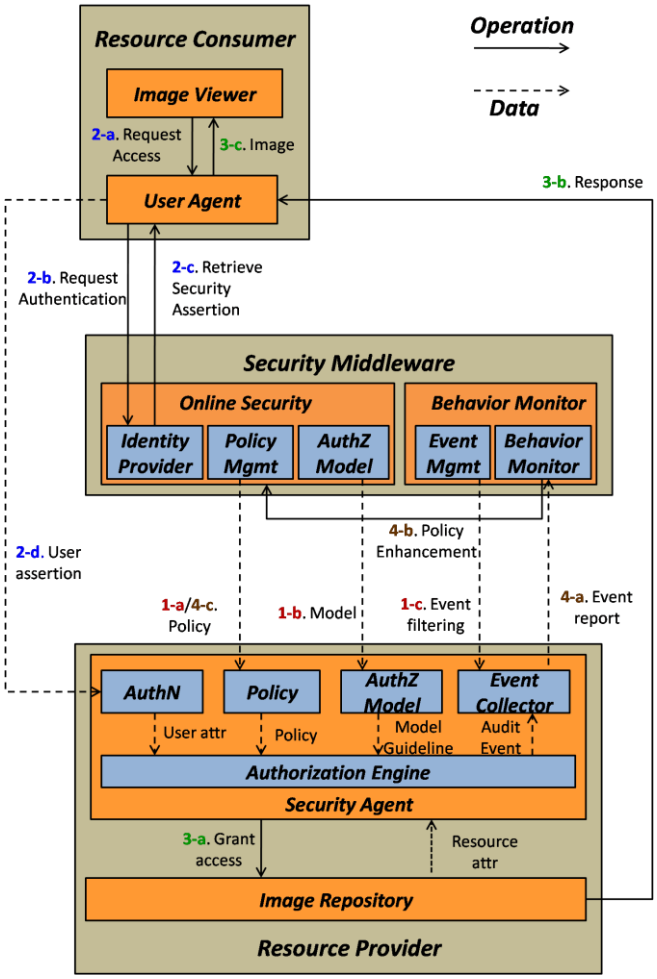


Fig. 2. Authentication, authorization and user behavior monitoring workflow for the proposed security middleware

The overall workflow model is shown in Figure 2. The steps of the model's operations are as follows.

Step 1) ***Security Agent customization*** (1-a to 1-c, red colour). Security Middleware generates the required training knowledge to train the generic Security Agent. The training knowledge is defined as a set of: 1-a) role based *access control polices* that are applicable to the protected resources; 1-b) *authorization model* that defines the access control procedure, and information-provider servers such as user attribute provider and resource-attribute provider; 1-c) *event filtering criteria* to be used for collecting user's access to resources. Security Agent receives the provided knowledge as well as the relevant Resource Provider's context, and then modifies the general authorization process and event collection task for the purpose of behavior analysis.

Step 2) ***Authentication*** (2-a to 2-d, blue colour). User Agent is a software agent deployed at the Resource Consumer. Image Viewer employs User Agent to fulfil the authentication flow (2-a). *Identity Provider* is an identity authentication server that is capable of authenticating the end users (2-b) and provides "security assertions" containing authentication statement and user attribute statement (2-c). A user assertion is communicated between User Agent and Security Agent for exchanging authentication and authorization data (2-d). Authentication statement confirms that the user has been identified and approved by the authentication server; the attribute statement asserts that the user is associated with certain attributes. These asserted attributes feed Security Agent to make access control decisions.

Step 3) ***Authorization*** (3-a to 3-c, green colour). Resource Provider sends instructions to Security Agent to perform authorization. Security Agent constitutes the following components: Authorization Engine that evaluates applicable policies and renders an access control decision; AuthN that provides the user's associated attributes; Policy that contains security middleware assigned policies and sends relevant policies to Authorization Engine for a specified target; AuthZ Model that guides Authorization Engine to fulfil the agreed-on authorization procedure; Event Collector records the authorization decisions. If this access request is granted, Security Agent sends an access request to Image Repository (3-a). Image Repository serves the request and returns its response (e.g., requested image) to User Agent (3-b). User Agent forwards the requested resource (image) to Image Viewer (3-c).

Step 4) ***Behaviour pattern mining and policy enhancement*** (4-a to 4-c, brown colour). User behaviour pattern is defined as consistent observations of a sequence of actions performed by the same user, under certain environment and during a specific time interval. Event Collector sends the collected data (i.e., event-log data) to Behavior Monitor component after filtering out the uninterested events (4-a). A knowledge driven behavior pattern discovery process is applied to orchestrate user's common behaviour patterns. Finally, the system administrators explore the opportunities to refine existing security policies by means of analysing salient features and characteristics of the discovered behaviour patterns (4-b). Finally, the consolidated polices are dispatched to the corresponding Security Agent to take effect (4-c), which closes an access control policy loop.

C. Behavior Anomaly Definition

Behavior anomaly is widely classified into the following three categories: i) *point anomaly*: where an individual data instance is considered as anomalous with respect to the rest of dataset; ii) *contextual anomaly*: where an individual data instance is considered as anomalous in a specific context, but might be considered as normal in a different context; and iii) *collective anomaly*: where a collection of related data instances is considered as anomalous with respect to the rest of dataset; however, the individual data instances in the collection may not be anomalous [15]. We propose a new method to detect contextual collective anomalies. In the followings, we define the data instance as an event that constitute a set of attributes; and define contextual collective anomalies as outlier behaviours that are dissimilar with the common behaviours in a specific context.

Event

An *event* records a single user-system interaction (i.e., any communication with the system such as storing and retrieving

a diagnostic image). An event is composed of a set of domain specific attributes. Whenever an attribute value changes, a new event is recorded. For example, an event of PACS system is represented by a tuple of attributes, as follows: *Event = <User, Role, Location, Action, Resource, Patient, Emergency>*. The attributes can be classified into three categorizes:

1) *Actor attributes.* The actor attributes are used to explain the subject of events. For example, *User* is an actor attribute, which identifies an individual who performed the action; *Role* is also an actor attribute which determines a group of people having similar privileges and responsibilities.

2) *Contextual attributes.* The contextual attributes determine the context (or neighbourhood) of events. For example, *Location* can be a contextual attribute which limits the neighbour events happened at the same location or nearby; *Time* can be considered as a contextual attribute which determines the neighbour events happened within a short period of time; *Patient* could be a contextual attribute which explains the neighbor events should be accessing the health records of a specific patient.

3) *Behavioral attributes.* The behavioral attributes define the non-actor and non-contextual characteristics of the events. For example, *Action* is a behavioural attribute, which describes one step of the workflow under a specific scenario; *Location* can also be a behavioral attribute which indicates one location of ward-round by nurses. Behavioral attributes in a dataset may be contextual attributes in another dataset, such as location that is a behavioral attribute in robot moving dataset but a contextual attribute in service accessing dataset.

Behavior

User behavior is extracted from a collection of user-system interactions (i.e., events). We propose a user behavior pattern representation based on association, sequencing and timing rules. *Association* indicates the concurrence of a set of attribute values together. *Sequencing* requires that a series of steps occur in a certain order. *Timing* allows sequencing the events; limits the events' occurrence frequency; and assigns the gaps between successive events.

In our approach, behavior is represented as a quadruple:

Behavior=<Actor, Sequence, Context, Time Interval>

Where *Actor* issues a behavior; *Sequence* is the sequence of steps performed by the Actor; *Context* is the circumstances in which the behavior takes place; and *Time Interval* is the time duration within which the behavior is recovered.

Common Behavior

Intuitively, frequently occurring user behaviors that are discovered from a large event dataset are reasonable to be regarded as user common behaviors. In other words, if a specific behavior is repeatedly performed by a group of people, most probably it is a common behavior. Also, given a large dataset of events, we can expect to discover a collection of common behaviors. The actor of a behavior is extracted from the actor attributes of events to categorize the behaviors. The context of a behavior is extracted from contextual attributes of the events to determine the neighborhood. The sequence of a behavior is extracted from behavioral attributes to explain user's behavioral characteristics. The time interval of a behavior is extracted from the time constraints.

Outlier Behavior

As discussed in subsection *Behavior*, an actor of behavior can be an individual or a group of people that have the same behaviors. We are interested in exploring the common behaviors of individuals or among a group of people. If an individual performs quite differently from his previous behavior, his current behavior is an outlier. If a person is categorized by role, he is supposed to perform similarly with the people who are assigned the same role. If a person has a collection of neighbors who are sharing the same context, he is expected to behave similarly with these neighborhoods. Compared with the anomaly categories discussed at the beginning of section C, the outlier behaviors explored by our approach are contextual collective anomalies.

Dynamic Behavior

The knowledge of extracted common behavior is sent to Security Agent. Security Agent monitors user's dynamic behavior (runtime event traces) and compares it with this user's previous behavior, and with common behaviors of similar actors in specific contexts. Given an outlier degree threshold, the dynamic behavior that is dissimilar to the actor's previous behavior or dissimilar to any common behavior is defined as outlier. Outlier behavior may be abnormal behavior, or maybe not, which requires system administrator's final determination.

D. Common Behavior Mining

Discovering common behavior patterns in a large event dataset (in the range of several hundreds of thousands or millions of events) is a hard problem and sometimes infeasible. To tackle this problem, we partition the search space (event dataset) into clusters of similar events based on their shared attributes using association mining operation. We operate an association mining engine on the event dataset to extract the shared attributes among events. Such shared attributes constitute the contexts of different common behaviours. The association mining engine receives a threshold value that we refer to as "*minsup-assoc"* (i.e., minimum support for association mining, with a value between 0% and 100%). Typically, such a search engine discovers many attribute-sets that occur frequently in the event dataset. A *frequent attribute set* is a collection of attribute values that appears in at least *minsup-assoc* events. Suppose 10% of events of the entire dataset occur at location "L-1" around 12:00 pm (represented as "T-12"). Given a threshold *minsup-assoc* 5%, association mining engine is capable of discovering the frequent attribute set <L-1, T-12> and a collection of events that contain the attribute set. A combination of the number of shared attributes and the number of sharing events measures the similarity between those events. Such an association-based similarity is used for clustering highly related events under a certain context, where each cluster becomes a smaller search space for the next phase.

After a clustering phase, sequential pattern mining is applied on each cluster to extract the frequent behavior

sequences. The input to the sequential pattern mining engine is an event sequence dataset and a user-specified threshold "*minsup-seq"* (i.e., minimum support of sequential pattern mining, with a value between 0% and 100%), and the output is a list of frequent sequence patterns that occur in at least *minsup-seq* sequences within the sequence dataset. To perform the sequential pattern mining, we should convert the event dataset to sequence dataset where each sequence is a set of ordered events performed by the same user within one day. Therefore, the discovered frequent sequence patterns can be viewed as user's daily behavior. In the same way, we could explore user's hourly behavior, weekly behavior, and monthly behavior.

As the events within one cluster share rather similar association patterns, the extracted behaviors from one cluster present the common behaviors under similar contexts. The association patterns may: i) include actor attribute *User*; ii) include actor attribute *Role*; or iii) include no actor attribute. If the association pattern includes actor attribute *User*, all behaviors extracted from this cluster belongs to a specific user; other attribute values of the association pattern contribute to the context of the common behaviors. For example, a cluster collects highly related events that share association pattern <U-1, L-1>, so that all behaviors explored from this cluster are common behavior of user U-1 at location L-1(context). If the association pattern includes actor attribute *Role*, the behaviors extracted from this cluster are common behaviors shared among a group of people with the same role. For example, a cluster collects highly related events that share association pattern <R-1, L-1>, so that all behaviors explored from this cluster are common behavior of a group of people that are assigned role R-1 at location L-1(context). If the association pattern does not include any actor attribute, all the attribute values in the association pattern contribute to the context of the common behaviors. The actor of these behaviors can be anyone. For example, a cluster collects highly related events that share association pattern <T-1, L-1>, so that all behaviors explored from this cluster are common behavior at location L-1 around time T-1. Such behaviors have common characteristics under certain context, which are not determined by the privileges and responsibilities of the actors.

E. Formal Representation of Outlier Behavior Detection

First, we formally define the knowledge of common behaviors that are sent from Behavior Monitor to Security Agent. Let $B = \{B_1, B_2, \ldots, B_n\}$ be a set of discovered common behaviors. Let $B_i = <B_ia, B_ic, B_is, B_it>$ be a common behavior, where B_ia is actor, B_ic is context, B_is is sequence, and B_it is time constraint. User's dynamic behavior is a trace of events of a specific user. Let $E = \{e_1, e_2, \ldots, e_m\}$ be an ordered event sequence of a single system user, where *Eu* represents the user of the event sequence. If the common behaviors *B* are daily behaviors, Security Agent performs the outlier detection operation once a day. *E* presents the collected events of user *Eu* within one day. A subsequence of *E* is represented as $E_{jk} = \{e_j, \ldots, e_k\}$, where $E_{jk} \subseteq$ E if there exists integers $1 \le j \le k \le m$.

The problem of finding outlier behaviors is defined as follows. Given a collection of common behaviors *B* and user's dynamic behavior E (observed user's event sequence during B_it), an outlier detector is designed based on the dissimilarity between *E* and *B*. Outlier behaviors are three types of observations: i) behave distinct different from his previous behavior; ii) behave quite different from people who have the same privileges and responsibilities; iii) behave quite different from others under certain context. Accordingly, the common behaviors are divided into three categories as shown in formula (2): *Bu* presents a collection of common behaviors of the same user *Eu*; *Br* presents a collection of common behaviors of people who are assigned the same role as *Eu*; *Bc* presents a collection of common behaviors under the same context shared by events in *E*.

$$\begin{aligned} &B = Bu \cup Br \cup Bc \\ &Bu = \{B_i \mid B_i \in B, B_ia = Eu\} \\ &Br = \{B_i \mid B_i \in B, Eu \in B_ia\} \\ &Bc = \{B_i \mid B_i \in B, B_ic \subseteq \textit{Shared Contexts in } E\} \end{aligned} \quad (2)$$

If the observed dynamic behavior *E* is dissimilar to any category of the common behaviors, it is considered as an outlier behavior. The outlier degree of *E* is defined in (3):

$$outlier(B, E) = \max(outlier(Bu, E), outlier(Br, E), outlier(Bc, E)) \quad (3)$$

The outlier degree is defined based on the behavior similarity. Ideally, the dynamic behavior is expected to be exactly the same as one of the common behaviors. If the dynamic behavior *E* is quite similar to any common behavior in *B*, it is unlikely to be an outlier. Formula (4) presents the outlier degree of E, compared with each of the user's previous behavior. If the dynamic behavior is dissimilar to all of his previous behaviors, its outlier degree increases. The behavior similarity $sim(B_i, E)$ is normalized with values between 0 and 1. The outlier degree calculation method is the same for all common behavior categories, hence we can calculate $outlier(Br, E)$ and $outlier(Bc, E)$ using the same formula (4).

$$outlier(Bu, E) = 1 - \max_{B_i \in Bu}(sim(B_i, E)) \quad (4)$$

To compare dynamic behavior with each common behavior, a new behavior similarity metric is defined as (5):

$$sim(B_i, E) = \begin{cases} \dfrac{|LCS(B_is, E)|}{|B_is|} & \text{if } |B_ic|=0 \\ \max\limits_{\substack{\{E_{jk} | E_{jk} \subseteq E\} \\ \forall e_p \in E_{jk},\, e_{p|a} \supset B_ic}} \left(\dfrac{|LCS(B_is, E_{jk})|}{|B_is|}\right) & \text{if } |B_ic| \neq 0 \end{cases} \quad (5)$$

where the similarity between common behavior B_i and observed dynamic behavior *E* is determined by the Longest Common Subsequences (LCS) [16] length under certain context B_ic. There are two cases for common behavior context: i) no context defined in common behavior ($|B_ic|=0$): in this case behavior similarity is determined by LCS between dynamic behavior *E* and common behavior sequence B_is; and

ii) context is not empty in common behavior ($|B_ic|\neq 0$): in this case behavior similarity is determined by the maximum LCS between the subsequences of dynamic behavior $\{E_{jk} | E_{jk} \subseteq E\}$ and B_is; E_{jk} is a subsequence of E with each event e_p in E_{jk} shares the same context with common behavior B_ic ($e_{p|a}$ means comparing attributes of e_p with context of B_ic). E is considered as similar to common behavior B_i if they share longer subsequences under the same context. If the common behavior is defined under certain context, but the dynamic behavior does not occur at such context, comparing the similarity between them is unreasonable and meaningless.

The length of LCS is considered as a measure of the closeness of two sequences, which finds the maximum number of identical items in these two sequences (preserving the event order). Each element of the sequence may be an itemset, but the formula of LCS as (1) can only compare simple items rather than itemset. For example, a sequence of behavior about actions and accessed objects looks like <<*A-1, O-1*> <*A-2, O-1*> <*A-3, O-2*>>. The itemsets <*A-1, O-1*> and <*A-1, O-2*> are partially identical. We enhanced the LCS formula as (6), which allows comparing itemsets in sequence. Let $X=(x_1, x_2, \ldots, x_m)$ and $Y=(y_1, y_2, \ldots, y_n)$ be two sequences of lengths m and n, respectively. An element of the sequence, $x_i \in X$ and $y_j \in Y$, can be an itemset. Suppose the attribute values of an itemset (x_i and y_j) are ordered, such as all elements in sequence X and Y follows the order of <*Action, Resource, Location*>. For example, x_i = <*A-1, O-1, None*> and y_j = <*A-1, None, L-2*>. The problem of comparing two itemset x_i and y_j can be converted to the problem of $lcs(x_i, y_j)$. A common subsequence *cs* of x_i and y_j represented by $cs(x_i, y_j)$ is a subsequence that occurs in both sequences. $lcs\ (x_i, y_j)$ is a common subsequence of both sequences with maximum length. The length of $lcs(x_i, y_j)$ is denoted by $R(x_i, y_j)$. Solving $R(X, Y)$ is to determine the longest common subsequence for all possible prefix combinations of the two sequences X and Y. Let $r(i, j)$ be the length of the *lcs* of $(x_1, x_2, \ldots, x_i)$ and $(y_1, y_2, \ldots, y_j)$. Then $R(X, Y)$ can be defined recursively as following:

$$r(i,j) = \begin{cases} 0 & \text{if } i = 0 \text{ or } j = 0 \\ r(i-1, j-1) + \dfrac{R(x_i, y_j)}{\max(|x_i|, |y_j|)} & \text{if } R(x_i, y_j) > 0 \\ \max\{r(i-1, j), r(i, j-1)\} & \text{if } R(x_i, y_j) = 0 \end{cases} \quad (6)$$

Finally the outlier will be detected by comparing the outlier degree $outlier(B, E)$ in formula (3) with an outlier degree threshold δ. If the outlier degree is greater than a threshold δ, E is identified as an outlier and will be notified to system administrators. The system administrator makes the final decision to grant or deny the outlier behavior. Based on intensive training, Security Agent may acquire enough trust from the system administrators about the outlier detection, and then Security Agent can be configured to make the final decision without manual involvement.

IV. CASE STUDY

In this section, we present an end-to-end case study to examine our proposed approach.

A. Implementation

We developed a prototype implementation of the proposed approach and applied on a simulated legacy PACS system and DI-r. ClearCanvas [17] is an open source implementation of a PACS viewer. A User Agent is deployed on the workstation to assist the ClearCanvas viewer to render the authentication flow. Health information exchange open source (HIEOS) [18] is an open source implementation that is used to simulate a set of DI-r web service interfaces to retrieve images. A generic Security Agent is deployed in front of HIEOS to perform authorization flow. Security middleware and DI-r make an agreement about applicable authorization policies, authorization model, and event filtering criteria. Security Agent is trained based on the security middleware generated training knowledge to perform its tasks.

B. Online security services

Let us consider a scenario where a user intends to use a PACS viewer application to display a patient's diagnostic report that is stored at the DI-r. One applicable authorization policy in this case is "*Only physicians are allowed to view and change a patient's diagnostic reports; other healthcare staffs only have the privilege of viewing the patient's diagnostic reports.*" Identity Provider issues an assertion including the statement of user's role "physician" after authenticating the end user. Resource Provider supplies the resource type as "diagnostic report" and the resource owner as "patient". Authorization engine grants this access request after evaluating the applicable policies with attribute values.

C. Post security services

The system kept running over one month and the Behaviour Monitor component totally collected 3000 user access events from the DI-r. These events are parsed and converted into attributed events. Each event is described by the following attributes: "User(U), Role(R), Location(L), Operation(O), Resource owner(W), Resource(E), Date(D), Time(T)". Each attribute value is represented by a quantitative value (e.g., L-1 means location "Oshawa"; L-2 means location "Toronto"; R-1 means role "physician"; R-2 means role "nurse").

The Apriori algorithm [11] is applied on the attributed events for discovering highly associated groups of events, where all events in one group share the same set of attribute values. We refer to the group of events as *basketset* and the shared set of attribute values as *itemset*. We define an association-based similarity metric between two events, which encode both the size of basketset and the length of itemset. Figure 3 is a visualization of the relationship among events. This graph is generated by Gephi [19], an open source network analysis and visualization software package. The undirected graph edges illustrate the associations between events according to our defined similarity metric. Each node represents an event, and each weighted edge represents the similarity value between two events. The events are grouped into a few of clusters. Our approach allows an event being assigned to multiple clusters.

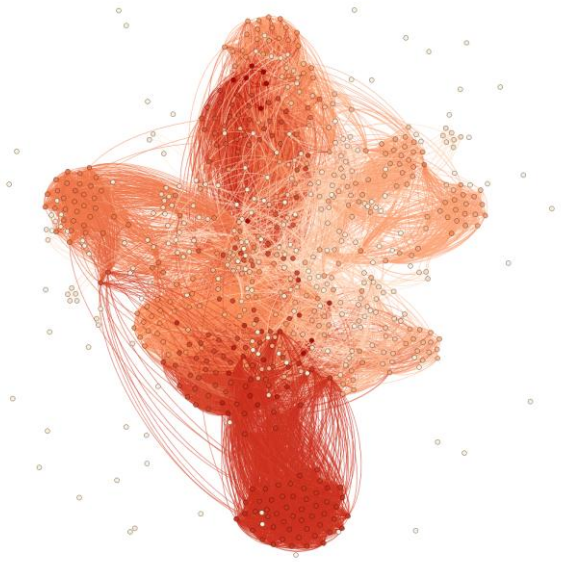


Fig. 3 Visualization of association between events

Sequential pattern mining algorithm CloSpan [20] is employed to discover user's daily behaviour in each cluster. First, we convert the event database into sequence dataset where each sequence is a set of ordered events performed by the same user within one day. Therefore, the discovered frequent sequence patterns can be viewed as the user's daily behaviour. In a post-analysis phase, we investigate the characteristics of the discovered sequence patterns in each cluster. For example: What is common among the users who accessed the system around the rush hour? What is the frequent behaviour pattern of a specific user in the system? Through analysing the common attribute values in each item of sequence patterns, context attributes are extracted to describe the circumstances of the complete sequence. The followings are some discovered behaviour patterns in the experiment:

- 50% of users have access requests at most 6 times during rush hour "10:00am".
- 80% of access requests from user "U-22" at location "L-6" are at time "1:00pm".

We can see the busiest time of user "U-22" is different from other users: "U-22" has more access request at 1:00pm but the normal rush hour is 10:00am. The system administrators may limit the maximum access request number during rush hour with differentiated policies. For example, an observed dynamic behaviour of user "U-22" is considered as outlier behavior if most access requests of user "U-22" is around "10:00am", because the dynamic behavior is changed from his pervious behaviour. In contrast, an observed dynamic behavior of user "U-23" is considered as outlier if most access requests of user "U-23" is around "3:00pm", because the dynamic behavior of user "U-23" is quite different from other users.

V. CONCLUSION

This paper contributes to the security and access control literature by proposing a common method for secure sharing medical images among legacy PACS systems and DI-r's. We have proposed a novel security middleware that replaces the existing trusted model for cross-PACS domains integration. Customizable and trainable software agents are deployed at the legacy systems to fulfil the authentication flow, to make authorization decisions as well as to collect user activities. In addition to the online security services, the security middleware provides post security services to recover user's access behavior patterns. We introduced a behavior model to represent behavior patterns. A variety of data mining techniques (i.e., association mining, sequence mining, and clustering) are applied to explore the user's common behavior. Furthermore, this research work proposed a new behavior similarity metric to measure the closeness between observed dynamic behavior and common user behaviors, and an outlier degree measurement to determine whether an observed dynamic behavior is outlier or not.

We plan to extend our work to provide step-by-step guidance throughout the whole policy enhancement process such as: i) investigating the characteristics of the extracted behavior patterns and committing recommendations to identify common behavior and abnormal behavior; and ii) detecting system security policy vulnerabilities and providing reasonable advice on policy consolidation.

REFERENCES

[1] Dossier Santé du Québec, "Diagnostic imaging group, Status of Diagnostic Imaging Repository (DI-r) projects across Canada". Available: http://www.camrt.ca/

[2] IHE IT Infrastructure Technical Framework Integration Profiles Volume 1. Available: http://www.ihe.net/, 2012

[3] IHE IT Infrastructure White Paper for Access Control, Available: http://www.ihe.net/, 2009

[4] Integration the Healthcare Enterprise website. Available: http://www.ihe.net

[5] N. Mehran, and K. Sartipi, "Modeling service representatives in enterprise systems using generic agents," *Service Oriented Computing and Applications (SOCA)*, vol. 5, pp. 245-264, Dec. 2011.

[6] K. Sartipi, K. Kuriakose, and W. Ma, "An Infrastructure for Secure Sharing of Medical Images between PACS and EHR Systems," *International Conference on Computer Science and Software Engineering (CASCON)*, pp. 245-259, 2013

[7] W. Ma, & K. Sartipi. Cloud-based Identity and Access Control for Diagnostic Imaging Systems. *In Proceedings of the International Conference on Security and Management (SAM) (p. 320). The Steering Committee of The World Congress in Computer Science, Computer Engineering and Applied Computing (WorldComp)*, 2015

[8] W. Ma, K. Sartipi, H. Sharghi, D. Koff, and P. Bak. "OpenID connect as a security service in Cloud-based diagnostic imaging systems." *In SPIE Medical Imaging,. International Society for Optics and Photonics*, pp. 94180J-94180J, 2015.

[9] W. Ma, and K. Sartipi, "An Agent-Based Infrastructure for Secure Medical Imaging System Integration," *in Computer-Based Medical Systems (CBMS), 2014 IEEE 27th International Symposium on*, pp. 72-77. IEEE, 2014.

[10] M. H. Yarmand, and K. Sartipi, and D. G. Down, "Behavior-based access control for distributed healthcare systems," *Journal of Computer Security*, 21.1, pp. 1-39, 2013

[11] A. Rakesh, T. Imieliński, and A. Swami, "Mining association rules between sets of items in large databases," *in ACM SIGMOD Record*, vol. 22, no. 2, pp. 207-216. ACM, 1993.

[12] A. Rakesh, and R. Srikant, "Mining sequential patterns," *in Proceedings of the Eleventh International Conference on. IEEE*, pp. 3-14. IEEE, 1995.

[13] Aggarwal, C. C., and Reddy, C. K. (Eds.). *Data clustering: algorithms and applications*. CRC Press, 2013.

[14] Xu, R., and Wunsch, D. 2005. Survey of clustering algorithms. *Neural Networks, IEEE Transactions on*, 16(3), 645-678.

[15] Chandola, V., Banerjee, A., & Kumar, V. Anomaly detection: A survey. *ACM computing surveys (CSUR)*, 41(3), 15, 2009

[16] Bergroth, L., Hakonen, H., and Raita, T. 2000. A survey of longest common subsequence algorithms. *In String Processing and*

Information Retrieval, 2000. SPIRE 2000. Proceedings. Seventh International Symposium on (pp. 39-48). IEEE.

[17] Open Source ClearCanvas PACS Website. [online]. Available: http://www.clearcanvas.ca/

[18] Open Source HIEOS Website. [Online]. Available: http://sourceforge.net/projects/hieos/

[19] Gephi - The Open Graph Viz Platform Website. [Online]. Available: http://gephi.github.io/

[20] X. Yuan, J. Han, and R. Afshar, "CloSpan: Mining closed sequential patterns in large datasets,"i*In Proceedings of SIAM International Conferen*Figures

Weina Ma was born in Baoding/China, in 1982. She obtained her B.Sc and M.Sc both in Computer and Software Engineering from Northwestern Polytechnic University in Xi'an/China, in 2005 and 2008, respectively. She started Ph.D study in Software Engineering in University of Ontario Institute of Technology in Oshwa/Canada from 2013. Her major research interests are knowledge engineering and data mining, eHealth services, and high performance computing and cloud computing.

Karmran Sartipi received B.Sc and M.Sc in Electrical Engineering from University of Tehran, and MMath and Ph.D in Computer Science (Software Engineering) from University of Waterloo. Dr. Sartipi has over 70 publications in Computer Science with focus on Information Security, Software and Knowledge Engineering, Data Analytics, and Medical Informatics. He has supervised more than 30 graduate students in inter-disciplinary fields, and developed several software tools in different scientific areas. He has collaborated with researchers in computer science, health science, business, and entrepreneurship fields for several years.

Volume 4 Issue 6, Nov. 2015, ISSN: 2288-0003

ICACT-TACT JOURNAL

Global IT Research Institute

1713 Obelisk, 216 Seohyunno, Bundang-gu, Sungnam Kyunggi-do, Republic of Korea 463-824

Business Licence Number : 220-82-07506, Contact: secretariat@icact.org Tel: +82-70-4146-4991