ISSN : 2165-4069(Online)
ISSN : 2165-4050(Print)

IJARAI

International Journal of Advanced Research in Artificial Intelligence

Volume 1 Issue 4

www.ijarai.thesai.org

A Publication of
The Science and Information Organization

IJARAI

INTERNATIONAL JOURNAL OF ADVANCED RESEARCH IN ARTIFICIAL INTELLIGENCE

SAI

THE SCIENCE AND INFORMATION ORGANIZATION

www.thesai.org | info@thesai.org

Editorial Preface

From the Desk of Managing Editor…

“The question of whether computers can think is like the question of whether submarines can swim.” ― Edsger W. Dijkstra, the quote explains the power of Artificial Intelligence in computers with the changing landscape. The renaissance stimulated by the field of Artificial Intelligence is generating multiple formats and channels of creativity and innovation. This journal is a special track on Artificial Intelligence by The Science and Information Organization and aims to be a leading forum for engineers, researchers and practitioners throughout the world.

The journal reports results achieved; proposals for new ways of looking at AI problems and include demonstrations of effectiveness. Papers describing existing technologies or algorithms integrating multiple systems are welcomed. IJARAI also invites papers on real life applications, which should describe the current scenarios, proposed solution, emphasize its novelty, and present an in-depth evaluation of the AI techniques being exploited. IJARAI focusses on quality and relevance in its publications. In addition, IJARAI recognizes the importance of international influences on Artificial Intelligence and seeks international input in all aspects of the journal, including content, authorship of papers, readership, paper reviewers, and Editorial Board membership.

In this issue we have contributions on security assessment of software design using neural network; the need for a new data processing interface for digital forensic examination; intelligent agent based flight search and booking system; temperature control system using fuzzy logic technique; imputation and classification of missing data using least square support vector machines – a new approach in dementia diagnosis; and also a proposed hybrid technique for recognizing Arabic characters

The success of authors and the journal is interdependent. While the Journal is in its initial phase, it is not only the Editor whose work is crucial to producing the journal. The editorial board members , the peer reviewers, scholars around the world who assess submissions, students, and institutions who generously give their expertise in factors small and large— their constant encouragement has helped a lot in the progress of the journal and shall help in future to earn credibility amongst all the reader members. I add a personal thanks to the whole team that has catalysed so much, and I wish everyone who has been connected with the Journal the very best for the future.

Thank you for Sharing Wisdom!

Managing Editor
IJARAI

Volume 1 Issue 4 July 2012
ISSN: 2165-4069(Online)
ISSN: 2165-4050(Print)

Editorial Board

Reviewer Board Members

CONTENTS

Security Assessment of Software Design using Neural Network

A. Adebiyi, Johnnes Arreymbi and Chris Imafidon
School of Architecture, Computing and Engineering
University of East London,
London, UK

Abstract— **Security flaws in software applications today has been attributed mostly to design flaws. With limited budget and time to release software into the market, many developers often consider security as an afterthought. Previous research shows that integrating security into software applications at a later stage of software development lifecycle (SDLC) has been found to be more costly than when it is integrated during the early stages. To assist in the integration of security early in the SDLC stages, a new approach for assessing security during the design phase by neural network is investigated in this paper. Our findings show that by training a back propagation neural network to identify attack patterns, possible attacks can be identified from design scenarios presented to it. The result of performance of the neural network is presented in this paper.**

Keywords- Neural Networks; Software security; Attack Patterns.

I. Introduction

The role software applications play in today's hostile computer environment is very important. It is not uncommon to find software applications running our transportation systems, communication systems, medical equipment, banking systems, domestic appliances and other technologies that we depend on. Since many of these software applications are missions critical, the need to ensure the security of their data and other resources cannot be overlooked. The increase of attacks aimed directly at software applications in the past decades calls for software applications to be able to defend itself and continue functioning. However, when software applications are developed without security in mind, attackers take advantage of the security flaws in them to mount multiple attacks when they are deployed. To address this problem a new research field called software security emerged in the last decade with the aim of building security into software application during development. This approach views security as an emergent property of the software and much effort is dedicated into weaving security into the software all through SDLC

One of the critical areas in this approach is the area of software design and security which proactively deals with attacking security problems at the design phase of SDLC. Reportedly, 50% of security problems in software products today have been found to be design flaws [17]. Design-level vulnerability has been described as the hardest category of software defect to contend with. Moreover, it requires great expertise to ascertain whether or not a software application has design-level flaws which makes it difficult to find and automate [9]. Many authors also argue that it is much better to find and fix flaws during the early phase of software development because it is more costly to fix the problem at a late stage of development and much more costly when the software has been deployed [6][29][30]. Therefore, taking security into consideration at the design phase of SDLC will help greatly in producing secured software applications.

There are different approaches and tools currently used for integrating security during the phases of SDLC. However, software design security tools and technologies for automated security analysis at the design phase have been slow in coming. This is still an area where many researches are currently being undertaken. Neural Networks has been one of the technologies used during software implementation and testing phase of SDLC for software defect detection in order to intensify software reliability and it has also been used in area of application security and network security in technologies such as authentication system, cryptography, virus detection system, misuse detection system and intrusion detection systems (IDS) [2] [4] [14] [20] [31][32]. This research takes a further step by using neural networks as a tool for assessing security of software design at the design phase of SDLC.

II. Related Works On Security Assessment Of Software Design

In order to design software more securely many approaches have been adopted for assessing the security in software designs during the design phase of SDLC. Some of these approaches are discussed below.

Threat modeling is an important activity carried out at the design phase to describe threats to the software application in order to provide a more accurate sense of its security [1]. Threat modeling is a technique that can be used to identify vulnerabilities, threats, attacks and countermeasures which could influence a software system [18]. This allows for the anticipation of attacks by understanding how a malicious attacker chooses targets, locates entry points and conducts attacks [24]. Threat modeling addresses threats that have the ability to cause maximum damage to a software application.

Architectural risk analysis is also used to identify vulnerabilities and threats at the design phase of SDLC which may be malicious or non-malicious in nature due to a software system. It examines the preconditions that must be present for the vulnerabilities to be exploited by various threats and assess the states the system may enter after a successful attack on the system. One of the advantages of architectural risk analysis is that it enables developers to analysis software system from its component level to its environmental level in order to evaluate the vulnerabilities, threats and impacts at each level [17].

Attack trees is another approach used to characterize system security by modeling the decision making process of attackers. In this technique, attack against a system is represented in a tree structure in which the root of the tree represents the goal of an attacker. The nodes in the tree represent the different types of actions the attacker can take to accomplish his goal on the software system or outside the software system which may be in the form of bribe or threat [6],[23]. "Attack trees are used for risk analysis, to answer questions about the system's security, to capture security knowledge in a reusable way, and to design, implement, and test countermeasures to attacks" [24].

Attack nets is a similar approach which include "places" analogous to the nodes in an attack tree to indicate the state of an attack. Events required to move from one place to the other are captured in the transitions and arcs connecting places and transitions indicate the path an attacker takes. Therefore just as attack trees, attack nets also show possible attack scenarios to a software system and they are used for vulnerability assessment in software designs [6].

Another related approach is the vulnerability tree which is a hierarchy tree constructed based on how one vulnerability relates to another and the steps an attacker has to take to reach the top of the tree [23]. Vulnerability trees also help in the analysis of different possible attack scenarios that an attacker can undertake to exploit a vulnerability.

Gegick and Williams [6] also proposed a regular expression-based attack patterns which helps in indicating the sequential events that occur during an attack. The attack patterns are based on the software components involved in an attack and are used for identifying vulnerabilities in software designs. It comprises of attack library of abstraction which can be used by software engineers conducting Security Analysis For Existing Threats (SAFE-T) to match their system design. An occurrence of a match indicates that the vulnerability may exist in the system being analyzed and therefore helps in integrating effective countermeasures before coding starts. Another advantage about this approach is that it can be easily adapted by developers who are novices on security.

Mouratidis and Giorgini [19] also propose a scenario based approach called Security Attack Testing (SAT) for testing the security of a software system during design time. To achieve this, two sets of scenarios (dependency and security attack) are identified and constructed. Security test cases are then defined from the scenarios to test the software design against potential attacks to the software system.

Essentially SAT is used to identify the goals and intention of possible attackers based on possible attack scenarios to the system. Software engineers are able to evaluate their software design when the attack scenarios identified are applied to investigate how the system developed will behave when under such attacks. From this, software engineers better understand how the system can be attacked and also why an attacker may want to attack the system. Armed with this knowledge, necessary steps can be taken to secure the software with capabilities that will help in mitigating such attacks

For most of the approaches discussed above, the need to involve security experts is required in order to help in identifying the threats to the software technology, review the software for any security issues, investigate how easy it is to compromise the software's security, analysis the impact on assets and business goals should the security of the software be compromised and recommend mitigating measures to either eliminate the risk identified or reduce it to a minimum. The need for security experts arises because there is an existing gap between security professionals and software developers. The disconnection between this two has led to software development efforts lacking critical understanding of current technical security risks [22].

In a different approach, Kim T. et.al [12] introduced the notion of dynamic software architecture slicing (DSAS) through which software architecture can be analyzed. "A dynamic software architecture slice represents the run-time behavior of those parts of the software architecture that are selected according to a particular slicing criterion such as a set of resources and events" [12] DSAS is used to decompose software architecture based on a slicing criterion. "A slicing criterion provides the basic information such as the initial values and conditions for the ADL (Architecture description language) executable, an event to be observed, and occurrence counter of the event" [12] While software engineers are able to examine the behavior of parts of their software architecture during run time using the DSAS approach, the trade-off is that it requires the software to be implemented first. The events examined to compute the architecture slice dynamically are generated when the Forward Dynamic Slicer executes the ADL executable. This is a drawback because fixing the vulnerability after implementation can be more costly [6].

Howe [10] also argues that the industry needs to invest in solutions that apply formal methods in analyzing software specification and design in order to reduce the number of defects before implementation starts. "Formal methods are mathematically based techniques for the specification development and verification of software and hardware systems" [7] Recent advances in formal methods have also made verification of memory safety of concurrent systems possible [7].

As a result, formal methods are being used to detect design errors relating to concurrency [10]. A software development process incorporating formal methods into the overall process of early verification and defects removal through all SDLC is Correct by Construction (CbyC) [24]. CbyC has proved to be very cost effective in developing software because errors are eliminated early during SDLC or not introduced in the first place. This subsequently reduces the amount of rework that would be needed later during software development. However, many software development organizations have been reluctant in using formal methods because they are not used to its rigorous mathematical approach in resolving security issues in software design. Model checkers also come with their own modeling language which makes no provision for automatically translating informal requirements to this language. Therefore, the translation has to be done manually and it may be difficult to check whether the model represent the target system [21]

III. The Nueral Network Approach

Our proposed Neural Network approach in analysing software design for security flaws is based on the abstract and match technique through which software flaws in a software design can be identified when an attack pattern is matched to the design. Using the regularly expressed attack patterns proposed by Williams and Gegick [6], the actors and software components in each attack pattern are identified. To generate the attack scenarios linking the software components and actors identified in the attack pattern, online vulnerability databases were used to identify attack scenarios corresponding to the attack pattern. Data of attack scenarios from online vulnerability databases such as CVE Details, Security Tracker, Secunia, Security Focus and The Open Source Vulnerability Database were used.

A. *The Neural Network Architecture*

A three-layered feed-forward back-propagation was chosen for the architecture of neural network in this research. The back-propagation neural network is a well-known type of neural network commonly used in pattern recognition problems [25]. A back-propagation network has been used because of its simplicity and reasonable speed.

The architecture of the neural network consists of the input layer, the hidden layer and the output layer. Each of the hidden nodes and output nodes apply a tan-sigmoid transfer function (2/(1+exp(-2*n))-1) to the various connection weights.

The weights and parameters are computed by calculating the error between the actual and expected output data of the neural network when the training data is presented to it. The error is then used to modify the weights and parameters to enable the neural network to a have better chance of giving a correct output when it is next presented with same input

B. *Data Collection*

From the online vulnerability databases mentioned above, a total of 715 attack scenarios relating to 51 regularly expressed attack patterns by Williams and Gegick's were analysed. This consisted of 260 attack scenarios which were unique in terms of their impact, mode of attack, software component and actors involved in the attack and 455 attack scenarios which are repetition of the same type of exploit in different applications they have been reported in the vulnerability databases. The attacks were analysed to identify the actors, goals and resources under attack.

Once these were identified the attack attributes below were used to abstract the data capturing the attack scenario for training the neural network. The attack attributes includes the following.

1. The Attacker: This attribute captures the capability of the attacker. It examines what level of access possessed when carrying out the attack.
2. Source of attack: This attributes captures the location of the attack during the attack.
3. Target of the attack: This captures the system component that is targeted by the attacker
4. Attack vector: This attributes captures the mechanism (i.e. software component) adopted by the attacker to carry out the attack
5. Attack type: The security property of the application being attacked is captured under this attribute. This could be confidentiality, integrity or availability.
6. Input Validation: This attributes examines whether any validation is done on the input passed to the targeted software application before it is processed
7. Dependencies: The interaction of the targeted software application with the users and other systems is analysed under this attributes.
8. Output encoding to external applications/services: Software design scenarios are examined under this attributes to identify attacks associated with flaws due to failure of the targeted software application in properly verifying and encoding its outputs to other software systems
9. Authentication: This attribute checks for failure of the targeted software application to properly handle account credentials safely or when the authentication is not enforced in the software design scenarios.
10. Access Control: Failure in enforcing access control by the targeted software application is examined in the design scenarios with this attribute.
11. HTTP Security: Attack Scenarios are examined for security flaws related to HTTP requests, headers, responses, cookies, logging and sessions with this attribute
12. Error handling and logging: Attack scenarios are examined under this attributes for failure of the targeted application in safely handling error and security flaws in log management.

C. *Data Encoding*

The training data samples each consist of 12 input units for the neural network. This corresponds to the values of the attributes abstracted from the attack scenarios.

The training data was generated from the attack scenarios using the attributes. For instance training data for the attack on webmail (CVE 2003-1192) was generated by looking at the online vulnerability databases to get its details on the attributes we are interested in.

This attack corresponds to regularly expressed attack pattern 3. Williams and Gegick [6] describe the attack scenario in this attack pattern as a user submitting an excessively long HTTP GET request to a web server, thereby causing a buffer. This attack pattern is represented as:

(User)(HTTPServer)(GetMethod)
(GetMethodBufferWrite)(Buffer)

TABLE I. SAMPLE OF PRE-PROCESSED TRAINING DATA FROM ATTACK SCENARIO

S\N	Attribute	Observed data
1	Attacker	No Access
2	Source	External
3	Target	Buffer
4	Attack Vector	Long Get Request
5	Attack Type	Availability
6	Input Validation	Partial Validation
7	Dependencies	Authentication & Input Validation
8	Output Encoding	None
9	Authentication	None
10	Access Control	URL Access
11	HTTP Security	Input Validation
12	Error	None

In this example, the data generated from the attack scenario using the attribute list is shown in Table I. Using the corresponding values for the attributes; the data is then encoded as shown in the

TABLE II. SAMPLE OF TRAINING DATA AFTER ENCODING

S\N	Attribute	Value
1	Attacker	0
2	Source	1
3	Target	9
4	Attack Vector	39
5	Attack Type	5
6	Input Validation	2
7	Dependencies	6
8	Output Encoding	0
9	Authentication	0
10	Access Control	2
11	HTTP Security	3
12	Error	0

The second stage of the data processing involves converting the value of the attributes in Table II into ASCII comma delimited format before it is used in training the neural network. For the expected output from the neural network, the data used in training network is derived from the attack pattern which has been identified in each of the attack scenarios. Each attack pattern is given a unique ID which the neural network is expected to produce as an output for each of the input data samples. The output data sample consists of output units corresponding to the attack pattern IDs. For instance, the above sample data on Webmail attack which corresponds to regularly expressed attack pattern 3, the neural network is trained to identify the expected attack pattern as 3.

D. The Neural Network Training

To train the neural network the training data set is divided into two sets. The first set of data is the training data sets (260 samples) that were presented to the neural network during training.

TABLE III. TRAINING AND TEST DATA SETS

Number of Samples	**Training Data**	**Test Data**
Data Set 1	143	26
Data set 2	117	25
Total	260	51

The second set (51 Samples) is the data that were used to test the performance of the neural network after it had been trained. At the initial stage of the training, it was discovered that the neural network had too many categories to classify the input data into (i.e. 51 categories) because the neural network was not able to converge. To overcome the problem, the training data was further divided into two sets. The first set contained 143 samples and the second set contained 117 samples. These were then used for training two neural networks. Mat lab Neural Network tool box is used to perform the training. The training performance is measured by Mean Squared Error (MSE) and the training stops when the generalization stops improving or when the 1000th iteration is reached.

E. Result and Discussion

It took the system about one minute to complete the training for each the back-propagation neural network. For the first neural network, the training stopped when the MSE of 0.0016138 was reached at the 26th iteration. The training of the second neural network stopped when the MSE of 0.00012841 was reached at the 435th iteration.

TABLE IV. COMPARISION OF ACTUAL AND EXPECTED OUTPUT FROM NEURAL NETWORK

s\n	Attack Pattern Investigated	Actual Output	Expected Output
Results from Neural Network 1			
1	Attack Pattern 1	1.0000	1
2	Attack Pattern 2	2.0000	2
3	Attack Pattern 3	2.9761	3
4	Attack Pattern 4	4.0000	4
5	Attack Pattern 5	4.9997	5
6	Attack Pattern 6	5.9998	6
7	Attack Pattern 7	7.0000	7
8	Attack Pattern 8	8.0000	8
9	Attack Pattern 9	9.0000	9

10	Attack Pattern 10	7.0000	10
11	Attack Pattern 11	11.0000	11
12	Attack Pattern 12	12.0000	12
13	Attack Pattern 13	12.9974	13
14	Attack Pattern 14	13.772	14
15	Attack Pattern 15	15.0000	15
16	Attack Pattern 16	16.0000	16
17	Attack Pattern 17	16.9999	17
18	Attack Pattern 20	19.9984	20
19	Attack Pattern 21	21.0000	21
20	Attack Pattern 22	22.0000	22
21	Attack Pattern 23	23.0000	23
22	Attack Pattern 24	23.9907	24
23	Attack Pattern 25	25.0000	25
24	Attack Pattern 26	26.0000	26
25	Attack Pattern 27	27.0000	27
26	Attack Pattern 28	28.0000	28
Results from Network 2			
27	Attack Pattern 29	28.999	29
28	Attack Pattern 30	29.9983	30
29	Attack Pattern 31	31.0000	31
30	Attack Pattern 32	31.998	32
31	Attack Pattern 33	32.8828	33
32	Attack Pattern 34	33.9984	34
33	Attack Pattern 35	32.8828	35
34	Attack Pattern 36	35.9945	36
35	Attack Pattern 37	36.6393	37
36	Attack Pattern 38	37.9999	38
37	Attack Pattern 39	37.9951	39
38	Attack Pattern 40	39.1652	40
39	Attack Pattern 41	40.9669	41
40	Attack Pattern 42	41.9998	42
41	Attack Pattern 43	42.998	43
42	Attack Pattern 44	43.9979	44
43	Attack Pattern 45	44.9991	45
44	Attack Pattern 46	45.8992	46
45	Attack Pattern 47	46.9956	47
46	Attack Pattern 48	47.9997	48
47	Attack Pattern 49	48.9999	49
48	Attack Pattern 50	49.8649	50
49	Attack Pattern 51	50.9629	51
50	Attack Pattern 52	50.6745	52
51	Attack Pattern 53	52.7173	53

To test the performance of the network, the second data sets were used to test the neural network. It was observed that the trained neural network gave an output as close as possible to the anticipated output. The actual and anticipated outputs are compared in the Table IV. The test samples in which the neural network gave a different output from the predicted output when testing the network includes tests for attack patterns 10, 35, 39, 40 and 52. While looking into the reason behind this, it was seen that the data observed for these attack patterns were not much. With more information on these attack patterns for training the neural network, it is predicted that the network will give a better performance. During the study of the results from the neural networks, it was found that the first neural network had 96.51% correct results while the second neural network had 92% accuracy. The accuracy for both neural networks had an average of 94.1%. Given the accuracy of the neural networks, it shows that neural networks can be used to assess the security in software designs.

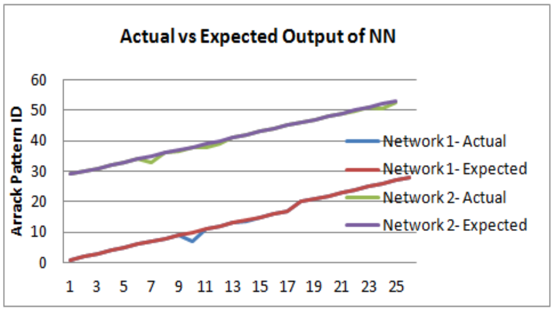


Figure 1. Actual vs. Expected output of Neural Network

IV. Future Work

To further improve the performance of the neural network system as a tool for assessing security in software designs, we are currently looking into the possibility of the system suggesting solutions that can help to prevent the identified attacks. Current research on solutions to software design security flaws gives a good insight in this area. Suggested solutions such as the use security patterns [11] and introduction of security capabilities into design in the SAT approach [19] are currently investigated. Furthermore, the performance of the neural network tool would be compared to current approaches used in assessing security in software designs in a case study on the design of an online shopping portal.

The regularly expressed attack pattern used in training the neural network is a generic classification of attack patterns Therefore; any unknown attack introduced to the neural network will be classified to the nearest regularly expressed attack pattern. Nevertheless the success of the neural network in analysing software design for security flaws is largely dependent upon the input capturing the features of the software design presented to it. As this requires a human endeavour, further work is required in this area to ensure that correct input data is retrieved for analysis. In addition, the neural network needs to be thoroughly tested before it can gain acceptance as a tool for assessing software design for security flaws.

V. Conclusion

Previous research works have shown that the cost of fixing security flaws in software applications when they are deployed is 4–8 times more than when they are discovered early in the SDLC and fixed. For instance, it is cheaper and less disruptive to discover design-level vulnerabilities in the design, than during implementation or testing, forcing a pricey redesign of pieces of the application. Therefore, integrating security into a software design will help tremendously in saving time and money during software development

Therefore, by using the proposed neural networks approach in this paper to analyse software design for security flaws the efforts of software designers in identifying areas of security weakness in their software design will be reinforced. Subsequently, this will enhance the development of secured software applications in the software industry especially as software designers often lack the required security expertise. Thus, neural networks given the right information for its training will also contribute in equipping software developers to develop software more securely especially in the area of software design.

REFERENCES

[1] Agarwal, A. 2006), "How to integrate security into your SDLC", Available at: http://searchsoftwarequality.techtarget.com/tip/0,289483,sid92_gci1174897,00.html, (Accessed 24/10/2010)

[2] Ahmad, I., Swati, S.U. and Mohsin, S. (2007) "Intrusion detection mechanism by resilient bpck Propagation (RPROP)", European Journal of Scientific Research, Vol. 17(4), pp523-530

[3] Arkin B, (2006), "Build security into the SDLC and Keep the bad guys out", Available at, http://searchsoftwarequality.techtarget.com/qna/0,289202,sid92_gci1160406,00.html,(Accessed 24/10/2010)

[4] Liu, G., Hu, F. and Chen, W.(2010), "A neural network emsemble based method for detecting computer virus", In proceedings of 2010 International conference on computer, mechatronics, control and electronic engineering, Vol. 1, pp391-393

[5] Croxford, M. (2005), "The challenge of low defect, secure software- too difficult and too expensive", Secure Software Engineering, Available at: http://journal.thedacs.com/issue/2/33 (Accessed 25/02/2012)

[6] Gegick, M. and Williams, L. (2006), "On the design of more secure software-intensive systems by use of attack patterns", Information and Software Technology, Vol. 49, pp 381-397

[7] Hinchey, M et al, (2008), "Software engineering and formal methods", Communications of the ACM, Vol.51(9), pp54-59

[8] Ho, S. L.; Xie, M. and Goh, T. N. (2003), "A Study of the connectionist model for software reliability prediction", Computer and Mathematics with Applications, Vol. 46, pp1037 -1045

[9] Hoglung, G and McGraw G. (2004), "Exploiting software: The Achilles' heel of cyberDefense", Citigal, Available at: http://citigal.com/papers/download/cd-Exploiting_Software.pdf (Accessed 02/12/2011)

[10] Howe (2005), "Crisis, What Crisis?" IEEE Review, Vol. 51(2), p39

[11] Kienzle, D. M and Elder, M. C. (2002) "Final Technical Report: Security Patterns for Web Application Development", Available at http://www.scrypt.net/~celer/securitypatterns/final%20report.pdf, (Accessed 26/01/2012)

[12] Kim, T., Song, Y. Chung, L and Huynh, D.T (2007) "Software architecture analysis: A dynamic slicing approach, ACIS International Journal of Computer & Information Science, Vol. 1 (2), p91-p103

[13] Lindqvist, U, Cheung, S. and Valdez, R (2003) "Correlated attack Modelling (CAM)", Air Force Research Laboratory, New York, AFRL-IF-RS-TR-2003-249

[14] Lyu, M. R, (2006), "Software reliability engineering: A roadmap", Available at: http://csse.usc.edu/classes/cs589_2007/Reliability.pdf (Accessed 21/09/2011)

[15] Mohan, K. K. Verma, A. K. and Srividya, A. (2009) "Early software reliability prediction using ANN process oriented development at prototype level", In proceedings of 20th International symposium on software reliablity engineering (ISSRE), India, Available at: http://www.issre2009.org/papers/issre2009_181.pdf (Accessed 12/05/2012)

[16] McAvinney, A. and Turner, B. (2005), "Building a neural network for misuse detection", Proceedings of the Class of 2006 Senior Conference, pp27-33

[17] McGraw, G. (2006), "Software security: building security in", Addison-Wesley, Boston, MA

[18] Meier, J. D., Mackman, A. And Wastell, B. (2005), "Threat modelling web applications", Available at: http://msdn.microsoft.com/en-us/library/ms978516.aspx (Accessed 24/10/2010)

[19] Mouratidis, H. and Giorgini, P (2007), "Security attack testing (SAT)- testing the security of information systems at design time", Information Systems, Vol. 32, p1166- p1183

[20] Pan, Z, Chen, S., Hu, G. and Zhang, D. (2003), "Hybrid neural network and c4.5 for misuse detection", In proceedings of 2003 International conference on machine learning and cybernetics, Vol.4, pp2463-2467

[21] Palshikar, G. K. (2004), "An Introduction to model checking", Embbedd.com, Available at http://www.embedded.com/columns/technicalinsights/17603352?_requestid=12219,(Accessed 20/02/2012)

[22] Pemmaraju, K., Lord, E. and McGraw, G.(2000), "Software risk management. The importance of building quality and reliability into the full development lifecycle", Available at: http://www.cigital.com/whitepapers/dl/wp-qandr.pdf, (Accessed 07/06/2011)

[23] Ralston, P.A.S; Graham, J.H and Hieb, J. L. (2007), "Cyber security risk assessment for SCADA and DCS networks", ISA Transaction, Vol.46(4), pp583-594

[24] Redwine, S. T. Jr and Davis, N.; et al, (2004), "Process to produce secure software: Towards more secure software", National Cyber Security Summit, Vol. 1

[25] Srinivasa, K.D. and Sattipalli, A. R, (2009), "Hand written character recognition using back propagation network", Journal of Theoretical and Applied Information Technology, Vol. 5(3), pp 257-269

[26] Tamura, Y.; Yamada, S. and Kimura, M. (2003), "A software Reliability assessment method based on neural networks for distributed development environment", Electronics & Communications in Japan, Part 3: Fundamental Electronic Science, Vol. 86(11), pp13-20.

[27] Telang, R. and Wattal, S.(2004), "Impact of software vulnerability announcement on market value of software vendors- an empirical investigation", The Third Workshop, University of Minnesota, 13-14 May, Minnesota.

[28] Threat Risk Modelling (2010) Available at: http://www.owasp.org/index.php/Threat_Risk_Modeling, (Accessed 24/10/201)

[29] Mockel C and Abdallah, A.E (2011) 'Threat Modelling Approaches and Tools for Securing Architectural Designs of E-Banking Application', Journal of Information Assurance and Security', Vol. **6**(5), pp 346-356

[30] Spampinato, D. G. (2008), 'SeaMonster: Providing Tool Support for Security Modelling', NISK Conference, Available at: http://www.shieldsproject.eu/files/docs/seamonster_nisk2008.pdf (Last Accessed: November 2011)

[31] Joseph, A., Bong, D.B.L. and Mat, D.A.A (2009) 'Application of Neural Network in User Authentication for Smart Home Systems' World Academy of Science, Engineering and Technology, Vol. 53, pp1293-1300.

[32] Zhang B.J.Y, and Wang, J.H.S.(2007), 'Computer Viruses Detection Based on Ensemble Neural Network', Computer Engineering and Applications, Vol. 43(13), pp 26-29.

AUTHORS PROFILE

Adetunji Adebiyi Doctoral student with the University of East London UK. His research focuses on integrating security into software design during SDLC. His research has led him to give talks and presentations in conferences and seminars he has attended.

Johnnes Arreymbi is a Senior Lecturer at the School of Computing, Information Technology and Engineering, University of East London. He has also taught Computing at London South Bank University, and University of Greenwich, London. He leads as Executive Director and co-founder of eGLobalSOFT, USA; an innovative Patented Software (ProTrack™) Company.

Chris Imafidon is a Senior Lecturer at the School of Computing, Information Technology and Engineering, University of East London. Chris is a multi-award winning researcher and scientific pioneer. He is a member of the

Information Age Executive Round-table forum. He is a consultant to the government and industry leaders worldwide.

The Need for a New Data Processing Interface for Digital Forensic Examination

Inikpi O. ADEMU
School of Architecture,
Computing and Engineering
University of East London
London, United Kingdom

Dr Chris O. IMAFIDON
Formerly, Head of Management of Technology Unit,
Queen Mary University of London,
Currently Senior Academic, School of Architecture,
University of East London,
London, United Kingdom

Abstract— **Digital forensic science provides tools, techniques and scientifically proven methods that can be used to acquire and analyze digital evidence. There is a need for law enforcement agencies, government and private organisations to invest in the advancement and development of digital forensic technologies. Such an investment could potentially allow new forensic techniques to be developed more frequently. This research identifies techniques that can facilitates the process of digital forensic investigation, therefore allowing digital investigators to utilize less time and fewer resources. In this paper, we identify the Visual Basic Integrated Development Environment as an environment that provides set of rich features which are likely to be required for developing tools that can assist digital investigators during digital forensic investigation. Establishing a user friendly interface and identifying structures and consistent processes for digital forensic investigation has been a major component of this research.**

Keywords-autonomous coding; intellisense; visual sudio; integrated development environment; relational reconstruction; data processing.

I. INTRODUCTION

Digital forensics plays an important part in the investigation of crimes involving digital devices. Digital forensic techniques are used primarily by private organisations and law enforcement agencies to capture, preserve and analyze evidence on digital devices. Digital evidence collected at a crime scene has to be analyzed and connections between the recovered information need to be made and proven. The search for digital evidence is thus a tedious task that consumes time. An extremely large amount of evidence needs to be processed in a very limited time frame which leads to delay in processing schedules. Digital forensic science provides tools, techniques and scientifically proven methods that can be used to acquire and analyze digital evidence (Ademu et al, 2012). Digital forensic investigators interact with digital evidence and digital forensic tools. The digital evidence can be used to backtrack or reconstruct illegal event.

Digital forensic investigators are constantly trying to find better and more efficient ways of uncovering evidence from digital sources. The problem here is that the research process itself is usually time intensive and consumes a lot of resources. Considering tools in forensic investigation also consumes time. The application of forensic technique is an extremely complex task, requiring an in depth understanding of the chosen digital devices. The development of a forensic application would typically require a researcher to develop hugely complex code that would perform tasks similar to those of the operating system or application in question in an attempt to discover stored data.

The law enforcement and corporate security professionals require tools for effective digital evidence acquisition. These tools already exist to capture, preserve and analyze data from hard drives, memory and network streams. These tools are available through open source licenses with their own unique user interface and feature and run on a variety of operating systems. This presents a challenge to digital forensic investigator since they must become acquainted with the operational characteristics and user interface of each tool they want to use. Also most of these tools are not online and are complicated to use, so it requires each investigative agency to set up their own suite of forensic tools.

The goal of this research is to provide an approach that supports digital forensic investigaors by identifying activities that facilitates and improves digital forensic investigation process. This research identifies the Visual Basic Integrated Development Environment (VBIDE) as a set of rich features that are likely to be required for developing tools that can assist digital investigators during digital forensic investigation.

II. RELATED WORK

Digital evidence is defined by (Carrier and Spafford, 2006) as a digital data that supports or refutes a hypothesis about digital events or the state of digital data. This definition includes evidence that may not be capable of being entered into a court of law, but may have investigative value, this definition is in agreement to (Nikkel, 2006) definition that states, digital evidence as a data that support theory about digital events. Evidence can be gathered from theft of or destruction of intellectual property, fraud or anything else criminally related to the use of a digital device. Evidence, which is also referred to as digital evidence is any data that can provide a significant link between the cause of the crime and the victim (Perumal, 2009). Digital forensics is the science of digital crime investigation. The main purpose of digital investigation is to collect digital evidences, which could be unaltered (Palmer, 2001). The use of computer system and other electronic devices has been widely used in the last two decades. The large amount of information is produced, accumulated, and distributed via electronic means. The majority of organizations interact with electronic devices every day. For this purpose, there is a need for finding digital evidences in computer systems and other electronic devices. However, when looking for digital evidences, there are some problems faced by an investigator. There is usually a large amount of files stored in computer system devices, and only few of them may comprise the valid evidences, but if the investigators don't know the location, it could consume a lot of time. According to (Jones et al 2006) the information is not only stored in working devices, there is need of recovering data from a broken device. Another important issue is the need for digital evidence being unaltered. If it cannot be proven that evidence has not been altered, it cannot be used as a valid digital evidence for persecuting a crime. A particular case is different from another. Techniques that should be used and actions that have to be taken as well as a type of digital evidence needed are mutable factors. And also the computer environment such as the operating system, type of storage devices used and the authentication method is another issue.

The investigative process is structured to encourage a complete, accurate investigation, ensure proper evidence handling and reduce the chance of mistakes created by preconceived theories and other potential pitfalls (Ademu et al, 2011). This process applies to criminal investigations as well as public and private inquiries dealing with policy violations or system intrusion. Investigators and Examiners work hand in hand in a systematic and determined manner in an effort to present an accurate and relaible evidence in the court. While in the court evidence are handed over to the prosecutors who scrutinize the findings and decide whether to continue or discontinue the case. In this research the digital forensic investigation processes are mentioned below:

- Planning / Preparation
- Identification /Interaction
- Documentation
- Collection and Preservation
- Examination
- Exploratory testing
- Relational Reconstruction
- Analysis
- Result Reporting
- Presentation

The size of data problem can be lessened by using automated tools. The manual analysis of hard drive images due to available sizes in gigabytes is really not realistic. Therefore, it is important to provide a tool to perform parts of the analysis automatically while shielding investigators from unnecessary details. According to (Moore, 2006) a vital problem of digital forensics is economics. This is the employment and training of investigators, this places a financial weight on the agencies that carries out investigations. These agencies can only employ limited amount of investigators therefore leading to backlogs in digital forensics. In an attempt to solve these problems, some form of automated processing must be introduces to lessen the problem faced by digital investigators.

The data processing methodology involves several steps to reduce the number of files that require analysis and translate unreadable data into a readable form. One approach is using command line utilities. Command line remains a powerful tool for digital forensic examiners. Command line tools enable examiners to perform very specific, auditable tasks, also by scripting a series of commands together, examiners can create very powerful set of files to automate a significant portion of evidentiary processing, thereby increasing productivity while reducing the chances of human errors during routine tasks. Graphical User Interface (GUI) tools such as Encase and FTK are another approaches used for filtering data but theses tools are very complicated for most users.

The New Technologies Inc. (NTI) developed an intelligent Filter program known as the Filter_1 which has the ability to make binary data printable and to extract potentially useful data from a large volume of binary data (Middleton, 2004). The intelligent filter program or Filter_1 tool help to reduce the size of the bitstream files without sacrificing useful information. IP Filter is possibly the most interesting and useful of the Forensic Utilities. It was developed by NTI to help law enforcement track down and investigate child pornography cases. It has a simple DOS user interface and is used in almost the same way as the Filter_1 (Stephenson, 2002). The difference is that it searches for instances of email addresses, Web URLs, and graphic or Zip file names. TextSearch Plus is a utility for searching a disk for text strings. It can search both allocated space and unallocated space (slack space). When used to search the physical disk, it can be used against any file system. TextSearch Plus makes an excellent tool for parsing very large logs in an internet backtracing investigation. It uses fuzzy logic and is designed to process a large amount of data in a relatively short time.

In considering setting up a working build environment on a Windows system can be abit complex, an out-of-the-box Windows system does not have a complier or interpreters and a native capability to mount or examine image files, it only supports a handful of file systems. According to (Altheide and Carvey, 2011) compiling native Windows code will usually require the use of Microsoft's Visual Studio. Although the complete versions of Visual Studio is commercial software, Microsoft releases the Visual Studio Express versions targeted towards specific langauge at no cost.

III. The Digital Forensic Examination Platform

The forensic examination involves preparing digital evidence to support the analysis phase. The nature and extent of a digital evidence examination depends on the known circunstances of the crime and the constraints placed on the digital investigator (Casey, 2004). With the reduction in the cost of data storage amd increasing volume of commercial files in operating system and application software, forensic digital forensic examiners can be devastated easily by huge number of files on even one hard drive ot backup disk.

Therefore, digital forensic examiners need procedures, to focus in on potentially useful data.

Different digital forensic tools have unique system requirements and usually have certain requirements as to the type of operating system they can run on. According to (Casey, 2002) computers may be running completely different operating systems and file systems in the future, it is therefore important that digital investigators should not become excessively dependent on tools and must develop a solid understanding of the underlying technology and related forensic investiagation techniques.

Visual Studio is a professional tool that provides a fully Integrated Development Environment (IDE) for visual C++, Visual C#, Visual J# and Visual Basic . IDE integrated all kinds of different codes written in C++, C#, J# or the Visual Basic programming language to create Windows application. The IDE also provides a wide range of productivity enhancements, such as intelligence, code validation, an assortment of wizard that writes code and element to create and manage databases (Schneider, 2004).

Digital forensic investigators must also keep pace with new developments in area such as .NET Framework. The .NET Framework can be considered of as operating system within an operating system. It is an execution environment similar in concept to java, that is designed to run on Windows 98/ME/NT/2000/XP etc. operating systems and to provide a common environment for programs. This enables programmers to write applications in their preferred language such as visual Basic, C++, Perl etc and compile them for the .NET environment, provding greater flexibility and functionality. There are variety of operating system and applications, it is not possible to describe or even identify every possible source of information that might be useful in an investigation. Also, each case in forensic investigation is different, requiring digital investigators to explore and research components. This Chapter provides examples of important aspect of an integrated development environment, providing greater flexibility and functionality enabling programmers to write application in their preferred languaues and compile with ease and less time consumption. The integrated development environment can be a digital forensic suite for future development.

IV. SETUP OF THE EXAMINATION SYSTEM

The setup required to perform examination with the Visual Basic Integrated Development Environment will go through the following steps.

1) Building Application

There will be one or more working build environment on the system. In this Chapter of this research there will be a generic developemnt environment that can be used to build open source applications written in the C and C++ langauges.

2) Installing Interpreters

In this resesrch some of the applications that will be used are written in interpreted languages such as Visual Basic. To run this program the appropriate interpreter and a way to install the prerequisite modules that the application rely upon is needed.

3) Working with image files

One main part of forensic examination is working with image files, which is the forensic copies of media. This is easier on some platforms than on others. It is important to be able to open a container inorder to get the content. An important part of setting up an examination system is ensuring that you can access image files directly.

4) Working with file system

This is the ability to interact with the file systems contained in image files using system functionslity.

a) Visual Basic Integrated Development Environment

The Visual Basic's Integrated Development Enviroment (IDE) enables to create, run and debug Windows programs without the need to open additional program.

Visual Basic was designed to make user-friendly programs easier to develop. Previously, programmers use language such as C or C++ etc. requiring hundred of lines of codes to get a window to appear on the screen. Recently, the same program can be created with much less time and fewer instructions.

Once Visual Basic (VB) is started the Microsoft Visual Page appears, this page tells that VB is starting. Next, the new project window appears on the screen. From this window a choce can be made to choose to create a new project or work on an existing project that is listed under the Recent tabs. A new project can be Standard executable files, ActiveX dynamic link libraries (DLLs), ActiveX controls, and Add-Ins are a few of the most commom project types. In this research standard executable file is used.

The Visual Basic Integrated Development Environment is a collection of menus. Toolbars and windows that make up a programming environment. The menu bar allows general management of programming, it is a typical drop down menu that is activated by clicking on a selected menu heading. The toolbar contains icons that provide a shortcut way of performing various tasks found on the menu bar, toolbar enables the accessing of the menu bar functionality through various toolbar buttons. Forms are the main building block of Visual Basic programs. ToolBox is used to add controls to the forms and the project explorer displays the projects, properties windows are customized within the Properties Window. The project explorer window displays a hierarchical list of open projects and the items contained in these projects. Each project is a different Visual Basic program. Therefore, Visual Basic allows a programmer to simultaneously have multiple programs open in the IDE. A form is a default container for Visual Basic controls. It involves the user interface of the program, forms is viewed on screen within the Form Layout Window. All controls in Visual Basic are objects, therefore, Visual Basic is an Object Oriented Language. All objects have different properties associated with them.

b) Code Overview

In order to write the source code, the command button on the form was double clicked. A code window opened and VB automatically places the code headings in the window. As the code was typed in, there appears a pop-up menu. This pop-up menu lists the available methods and properties of the object. Desired method or property can be selected from this pop-up menu or desired method or property name can be typed and VB will automatically scroll in the pop-up menu to method or property name. This feature is the auto code-completion feature (intellisense) of VB. This Saves a lot of time from doing numerous and rigorous coding.

The implementation code was divided into different Visual Studio programming suite, according to the functional tasks of the classes. These programming languages provided an easy way of structuring the code into clear and distinct logical groups of classes. The programming languages used in the implementation is Visual Basic.

The code view in this research is shown below:

```
Public Class Form1
    Private Sub Button1_Click(ByVal sender As
System.Object, ByVal e As System.EventArgs)
Handles Button1.Click
        'Analyze the first character of a
string
        Dim anyString As String
        anyString = Button1.Text.ToUpper
        Select Case anyString.Substring(0, 1)
            Case "S", "z"
                Button1.Text = "The string
begins with a sibilant."
            Case "A" To "z"
                Button1.Text = "The string
begins with a nonsibilant."
            Case "0" To "9"
                Button1.Text = "The string
begins with a digit."
            Case Is < "0"
                Button1.Text = "The string
begins with a character of " & -"ANSI value
less than 48."
            Case Else
                Button1.Text = "The string
begins with : ; < = > " & -" ? @ [ \ ] ^ _ or
' . "
        End Select
    End Sub
End Class
```

c) Installing Interpreters

Through the installation of Microsoft Visual Studio, Visual Basic has already been installed.

d) Working with image files

The application of Visual Basic allows display of object. It can also allow the display of animations, play sounds, music and videos etc.

e) Working with file system

The VB project is composed of two main file types known as project files and form files. A VB project can contain multiple forms, but in this thesis one form is used. The project file tells VB which forms are associated with a specific project and the form file lists the objects on the form, the object property settings and the source code associated with the form.

In this research, below is a simple scenario of acquisition of data by the help of introducing simple programs in the Visual Basic environment during digital forensic investigation.

The following program has the string selector ‘anyString.Substring (0, 1)’.

TABLE 1: OBJECT AND PREFIX

OBJECT	PREFIX	EXAMPLE
FORM	Frm	frmAnalyze
LABEL	Lbl	lblEnter
TEXT BOX	txt	txtString
BUTTON	btn	btnAnalyze

Object applied and their three letter Prefixes

In this scenario, the form is to analyze the first character of a string where once a string is entered in the text box and the button cliked result will be read only if the string begins with a particular selector. Below are the steps in designing the graphical user interface.

After clicking on Microsoft Visual Basic Express Edition, the first page that appears is known as Start Page.

The next step is to click on File, and then click on New Project to produce dialog box as shown in figure 4.3 and then renamed.

By using the Common Controls in the Toolbox a form is created. This is as shown in Figure 1

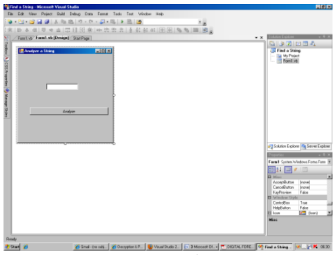
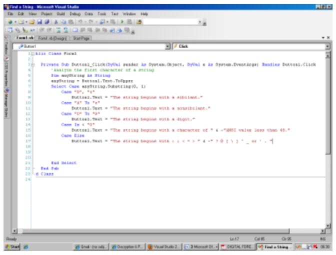


Figure 1: Text property for entering string

The next step is designing the Code Editor by clicking the right mouse button anywhere on the Main area and click on View Code. The Form Designer IDE is replaced by the Code Editor also called Code View or Code Window as shown in Figure 2



Find a String - Microsoft Visual Studio
File Edit View Project Build Debug Data Tools Test Window Help
Form1.vb
Form1.vb [Design]
Start Page
Button1
Click
Ready

```
Public Class Form1

    Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Button1.Click
        'Analyze the first character of a string
        Dim anyString As String
        anyString = Button1.Text.ToUpper
        Select Case anyString.Substring(0, 1)
            Case "S", "z"
                Button1.Text = "The string begins with a sibilant."
            Case "A" To "z"
                Button1.Text = "The string begins with a nonsibilant."
            Case "0" To "9"
                Button1.Text = "The string begins with a digit."
            Case Is < "0"
                Button1.Text = "The string begins with a character of " & -"ANSI value less than 48."
            Case Else
                Button1.Text = "The string begins with : ; < = > " & -" ? @ [ \ ] ^ _ or ' . "



        End Select
    End Sub
End Class
```

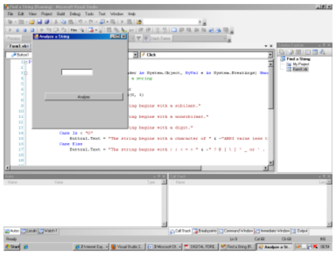
Figure 2: Visual Basic Integrated Developer Environment Code Editor mode

In the Code Editor, lines of code will be written for the event procedure. In the case of this thesis, the first line is the header for the event procedure named `Button1_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Button1.Click.`

```
This procedure is triggered by the event,
Button1.Text = "The string begins with a
sibilant."
```

That means, whenever there is a sting according to the condition entered in the text box and once button is clicked the code between the two lines will appear

The program is run by pressing F5 and a dialog box is shown in Figure 3 as follows:



```
    Case Is < "0"
        Button1.Text = "The string begins with a character of " & -"ANSI value less t
    Case Else
        Button1.Text = "The string begins with : ; < = > " & -" ? @ [ \ ] ^ _ or ' .
```

Figure 3: The code view

In the next step a string according to the second condition is entered into the text box e.g. Saturday is entered in the box and the code between the first line of condition is displayed as shown in Figure 4 below:

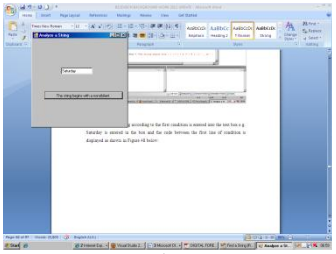


Figure 4: Text result from the string entered

In order to make a decision the investigator needs to specify a condition that determines the course of action. A condition can be said to be an expression involving relational operators such as < AND =) that is either true or false. The Visual Basic Integrated Develoment Environment facilitates a structured and disciplined approach to computer program design.

V. CONCLUSION

Digital evidence must be precise, authenticated and accurate in order to be accepted in the court. Digital evidence is fragile in nature and they must be handled properly and carefully. Detailed digital forensic investigative processes provide important assistance to forensic investigators in establishing digital evidence admissible in the court of law.

It is good practice to begin a new investigation by preparing an organised working environment. In digital forensic analysis, this involves preparing adequate and safe media on which to copy the data to be processed. This research describes the application of a techniques that can facilitates the process of digital forensic investigation, therefore allowing digital investigators to utilize less time and fewer resources.

The research introduces a structured and consistent approach for digital forensic investigation. The research provides an investigative process that helps improve digital forensic investigation identifying Visual Basic Integrated Development Environment as an environment that provides a basis for the development of techniques and especially tools to support the work of investigators. The purpose of identifying the Visual Basic Integrated Development Environment is to provide easy to use and a set of rich features which are likely to be required for developing tools that can assist digital investigators during digital forensic investigation. This technique provides a basis for future work in the development of techniques and especially tools to support the work of investigators.

ACKNOWLEDGEMENT

The authors would like to thank Dr David Preston, University of East London, and School of Architecture, Computing and Engineering and the University of Cambridge Computer laboratory for providing support during this research.

REFERENCES

[1] Ademu, I. Imafidon, C. Preston, D., (2012) Intelligent Software Agent applied to Digital Forensic and its Usefulness Vol. 2 (1) Available (online): http://interscience.in/IJCSI_Vol2Iss1/IJCSI_Paper_21.pdf Accessed on 10th April 2012

[2] Ademu, I. Imafidon, C. I. Preston, D. (2011) A New Approach of Digital Forensic Model for Digital Forensic Investigation Vol. 2, (12) Available (online): http://thesai.org/Downloads/Volume2No12/Paper%2026-A%20New%20Approach%20of%20Digital%20Forensic%20Model%20for%20Digital%20Forensic%20Investigation.pdf Accessed 28th April 2012

[3] Altheide, C. Carvey, H (2011) Digital forensics with open source tools Pp 26 – 27 Elsevier-Waltham

[4] Carrier, B. Spafford, H. (2006), Categories of digital investigation analysis techniques based on the computer history model. Available (Online): http://dfrws.org/2006/proceedings/16-carrier.pdf Accessed on the 12th April 2012

[5] Casey, E (2004) Digital evidence and computer crime forensic science, computers and the internet 2nd Edition Pg 101 Academic Press – London

[6] Casey, E. (2002) Handbook of computer crime and investigation Pg 116 Academic Press - London

[7] Jones, K. Bejtlich, R. Rose, C. (2006) Real digital forensics: Computer security and incident response Pg 172

[8] Middleton, B. (2004) Cyber Crime Investigator's Field Guide 2nd Edition Pp 53-54 Auerbach – Florida

[9] Moore, T. (2006) The Economics of Digital Forensic Available (online): http://people.seas.harvard.edu/~tmoore/weis06-moore.pdf Accessed on 30th April 2012

[10] Nikkel, B. (2006) the role of digital forensic with a corporate organisation Available (online): www.digitalforensics.ch/nikkel/06a.pdf Accessed on 25th February 2012

[11] Palmer, G. (2001) a road map to digital forensic research Available (online): http://www.dfrws.org/2001/dfrws-rm-final.pdf Accessed on 25th April 2012

[12] Panda labs Annual Report (2009) Available (online):

[13] http://www.pandasecurity.com/img/enc/Annual_Report_Pandalabs2009.pdf Accessed on 5th May 2012

[14] Perumal, S. (2009) Digital forensic model based on Malaysian investigation process Vol. 9 (8) Available (online): http://paper.ijcsns.org/07_book/200908/20080805.pdf Accessed on 7th April 2012

[15] Schneider, D (2004) An introduction to programming using Visual Basic 6.0 4th Edition Pp 32 Prentice Hall – New Jersey

[16] Stephenson, P. (2000) Investigating Computer-Related Crime Florida: CRC pg 32.

Intelligent Agent based Flight Search and Booking System

Floyd Garvey
Mona Institute of Applied Science
University of West Indies, Kingston, Jamaica

Suresh Sankaranarayanan [1, 2]
[1]Computing & Information System,
[1]Institut Teknologi Brunei, Brunei
[2]Department of Computing
[2]University of West Indies, Kingston, Jamaica

Abstract— **The world globalization is widely used, and there are several definitions that may fit this one word. However the reality remains that globalization has impacted and is impacting each individual on this planet. It is defined to be greater movement of people, goods, capital and ideas due to increased economic integration, which in turn is propelled, by increased trade and investment. It is like moving towards living in a borderless world. With the reality of globalization, the travel industry has benefited significantly. It could be said that globalization is benefiting from the flight industry. Regardless of the way one looks at it, more persons are traveling each day and are exploring several places that were distant places on a map. Equally, technology has been growing at an increasingly rapid pace and is being utilized by several persons all over the world. With the combination of globalization and the increase in technology and the frequency in travel there is a need to provide an intelligent application that is capable to meeting the needs of travelers that utilize mobile phones all over. It is a solution that fits in perfectly to a user's busy lifestyle, offers ease of use and enough intelligence that makes a user's experience worthwhile. Having recognized this need, the Agent based Mobile Airline Search and Booking System is been developed that is built to work on the Android to perform Airline Search and booking using Biometric. The system also possess agent learning capability to perform the search of Airlines based on some previous search pattern .The development been carried out using JADE-LEAP Agent development kit on Android.**

Keywords- Agents; Biometric; JADE-LEAP; Android.

I. INTRODUCTION

The Airline industry controls the world of travel and this industry alone has managed to reduce the distance between places that are geographically miles apart to merely in hours and minutes. According to investopedia, "*Few inventions have changed how people live and experience the world as much as the invention of the airplane".* There are thousands of airlines worldwide that cover thousands of miles daily and travel has become an acceptable part of our routine. Therefore, to ensure that we get to where we need on time, individuals have to book flights in advance or have someone book the flights on their behalf. In some situations unless a flight is booked well in advance, then one may have to miss such a flight. As the world progresses in these areas, it has become apparent that technology has to play a key role and hence many individuals use the internet to assist in making world of travel a little easier. We find many persons booking flights, cancelling flights and accessing general information about flights via internet. The technological advancements that we have made over the last ten years have tried its best to make the world of travel a lot easier [1]. Various technologies have been employed over the years to address the varying concerns of the travel industry [1]. Still we see yearly in each winter airports in Europe, England and even North America getting jammed with persons, because of cancelled flights and consequently individuals sleep at airports. All these are normally caused by bad weather. However, a lot of this could have been aborted if these travelers had the technological means to manage their flight experiences in a better way. When we look closer home within the Caribbean, we might not suffer from snowstorms that leave our airports inundated but we experience lengthy delays and cancelled flights. With these as background, we here have developed an Intelligent Agent based Mobile system that can provide users the capability to search and book flights and additionally avail enough information so that users of this system will not have to sleep in airports. This system also provides an additional component to users with the capability to see the reviews of airlines and the services so that they might not have to make a mistake that probably was made by someone else and already noted. The system possesses unique feature of booking flights using mobile handset with Biometrics to avoid frauds in credit card payment. However, before going into the details of the system developed, we would first review in brief about some existing Airline Reservation systems in vogue in section 2. Section 3 provides some introduction to Intelligent Agents followed by Agent learning, AI in Flight reservation and biometrics. Section 4 gives the details on the proposed Intelligent Agent based Flight search and booking architecture followed by flowchart and algorithm. Section 5 gives the implementation and validation details using JADE-LEAP and Android 2.2 with Google Maps API. Section 6 is the conclusion and future work.

II. REVIEW OF AIRLINE RESERVATION SYSTEMS

The history of the Computer Reservation Systems (CRS) in Airline industry dates back to 1970s when airlines began modifying and enhancing their internal reservation systems to make the sale of airline tickets through travel agents more efficient. The CRS gave travel agents access to information about flight schedules, fares, and seat availability. It also enabled them to make reservations and issue of tickets automatic. Although the computer reservation systems are owned and operated by particular airlines, travel agent can use one to get information and make reservations on virtually any scheduled carrier [2]. Since the system, make both airlines and travel agents more productive, CRS owners charge both of them for the use of their systems. Travel agents rent the equipment, while airlines pay a booking fee for each flight reservation. American Airlines introduced the first computer reservation system; United, TransWorld, Eastern, and Delta each followed with systems of their own. American and United, however, dominate the CRS industry; in 1986, they accounted for 41 percent and 33 percent, respectively, of the flight segments booked through computer reservation system [2].

A great majority of these airlines have online web based system as most persons such as business travelers and persons technically inclined to facilitate their travelling process by booking flights online. It is because of the large increase in the amount of persons that travel gave rise to the Online Reservation System. The modern airline reservation system is a comprehensive suite of products to provide system that assists with variety of airline management tasks and service customer needs from the time of initial reservation through completion of the flight [3].Now with the advent of tremendous development in mobile technology, we see many people searching and booking flights using their mobile handset. There is also talk about receiving tickets and boarding pass on mobile to make airline ticket booking paperless. There has been some application developed in terms of Flight booking on the latest Android handset such as; Kayak, CheckMyTrip Mobile Companion, Pageonce Travel, Flight Trip Planner and TripIt. But many of these applications are mobile versions to websites only that provide services like flight booking, location information, weather information and information of the destinations like hotels, restaurants, gas stations, cinemas and so on similar to any online web based Airline search system

The booking of flights using a mobile phone has become extremely popular over the last half decade. As smart phones became popular, reserving flights via mobile phone was introduced. However, there were some challenges in booking flights via mobile devices, which has proved a logistical challenge for technology providers in the managed travel space. Though search is good in terms of returning results, the search is more brute force as opposed to intelligence. The search results are returned by querying the database built on the user criteria. The search most often times is not refined by the software and it is left to the user to refine the search to retrieve the most appropriate results by sitting in front of the computer connected to internet or from the mobile. After results returned, the user is left with the task to mine through all this information after which the choice is then made or if the search is not sufficient perform other searches until the result is satisfactory. This puts lots of work on the user though the search operation is carried out by the search algorithm.

Now to book a flight users have to use their credit cards by inputting the details online with all their related information. According to Scamwatch, victims have reported losses of more than a $1000 for fake international flight bookings and instances of identity theft [4]. Though many online airline web systems take extreme care in securing the financial details for booking, still there are some discrepancies in the system that lead to fraud and identity theft. With all this in mind one has to be careful of how booking is done as there are several sites that exist that basically mimic real sites so the concept of booking flights online by entering all your information in view of the security challenges that currently exist is not ideally safe. So with all these in mind, we here have developed Intelligent Agent based Flight Search and booking system [5] which searches the Airline based on user criteria and makes intelligent decision rather than leaving to the user to make decision. Also booking flight been carried out using Biometrics to avoid credit card fraud. However, before going into those details, we will review in brief about Intelligent Agent technologies followed by AI in flight Reservation system and biometrics.

III. Intelligent Agents

Agent technology has emerged as formidable IT area. Agents can be defined to be autonomous, problem-solving computational entities capable of effective operation in dynamic and open environments [6-10]. An agent is something that acts in an environment. For agents to be classified as intelligent they not only must exhibit intelligent behavior but they must have the ability to learn and follow similar patterns of learning. Learning is defined to be the acquisition of knowledge or skills through experience, practice, or study, or by being taught [11][12]. Learning is done by humans, animals and some machines. In order for an agent to learn, they must be able to act intelligently. The concept of Agent Learning and the consequent artificial intelligence is not new [13][14]. Since that time, hundreds if not thousands, of articles have been published on the topic, and at least two books [15 - 20]

A. Artificial Intelligence in Flight Reservation Systems

Artificial Intelligence (AI) is the key technology in many of today's novel applications, ranging from banking systems that detect attempted credit card fraud, to telephone systems that understand speech, to software systems that notice when you are having problems and offer appropriate advice. These technologies would not exist today without the sustained federal support of fundamental AI research over the past three decades [21]. The area of flight reservation systems is no exception to the existence of artificial intelligence. Many airlines have opted to divest most of their holdings to Global Distribution Systems (GDS) due to which many systems are now accessible to consumers through Internet gateways for hotels, car rental agencies, and other services as well as airline tickets. A traveler or a travel agent can chalk out an itinerary using a GDS which is a global system interconnecting airlines, hotels, travel agents, car rental companies, cruise liners etc. [22].

There are four major Global Distribution Systems, and they are AMADEUS, GALILEO, SABRE and WORLDSPAN. The SABRE reservation system is used by American Airlines and boasts an intelligent interface named PEGASUS, which is a spoken language interface, connected to SABRE which allows subscribers to obtain flight information and make flight reservations via a large, on-line dynamic database accessed through their personal computer over the telephone.

As the technology advances and more persons are becoming smart phone users, the need exists to give internet users from desktops, laptops and smart phones the ability to search for flights and to book flights online. Therefore, we have several applications that have given users the ability to work on smart phones such as blackberries, iphone and android. One of the major concerns for Smartphone users is the actual booking of the flight because this includes the use of credit cards and with the many incidents of identity theft and fraud over the internet, this raises a red flag. However, we offer in this paper, to prospective users a secure environment to do these transactions without worry or concerns. However, before going into those details, we look in brief about Biometrics.

B. Biometrics

Biometrics is the science and technology of measuring and analyzing biological data. In information technology, biometrics refers to technologies that measure and analyze human body characteristics, such as DNA, fingerprints, eye retinas and irises, voice patterns, facial patterns and hand measurements, for authentication purposes [23]. In this research area of biometrics, we will focus on fingerprint capture, verification and encryption [24-26]. Biometric is a standard now that all laptops come with biometric security options that give users the ability to store their passwords as biometric imprints and log onto their devices using their fingers as opposed to typing in passwords in a traditional way.

We will now present the details of how biometric data is captured and verified. To convert the biometric input, a software application is used to identify specific points of data as match points. The match points in the database are processed using an algorithm that translates that information into a numeric value. The database value is compared with the biometric input from the end user who has entered into the scanner and authentication is either approved or denied [23]. In order to enroll a fingerprint several steps are performed [27] as shown in Fig 1. Therefore, we see the use of biometrics as a very secure way of implementing security in a system that users' private and sensitive data are being accessed and want to keep out of unauthorized personnel to prevent identity theft. It has been seen from the literature that work has been done in the use of AI in flight reservation systems and technologies been used to avoid identity fraud in payment. But in all the above system AI search algorithms, being used to perform the search of airlines with some intelligence and also security has been used to avoid credit card theft in payment. But still the system lacks intelligence and smartness in searching of airlines where again the burden falls on the user towards refining the search, making decision based on retrieved results. The system also possesses no facility of searching based on past experience or so. Also the system gives no information on the rating of airlines and so. In addition to search, there still exist challenges to facilitate a secure platform that users can trust to carry out their transaction in a technical space free from interference. The applications that exist are good but in many of the instances, they provide real time flight information to prospective clients and facilitate payment with the use of third party intervention. All these systems been developed as web based only which can be accessed from desktop or mobile and not for mobile handset as such. These drawbacks that exist can be accomplished by means of intelligent agent which is however seen that no research exists or been carried out towards airline reservation and booking system.

Our proposed system [5] so allows the users to search for airline based on their preference using intelligent agent to make intelligent decision and display on mobile handset by applying fuzzy preferences. Also, extend the Intelligent Search Agent with learning capability that may be searching for a flight with minimal individual preferences based on previous search experience of the agent. System also aims to protect users from identity theft and fraud by providing a platform to validate airline based on flight selection by user and facilitate booking and cancellation of flights by customers using their own credit cards using biometric and encryption technology to ensure a secure platform.

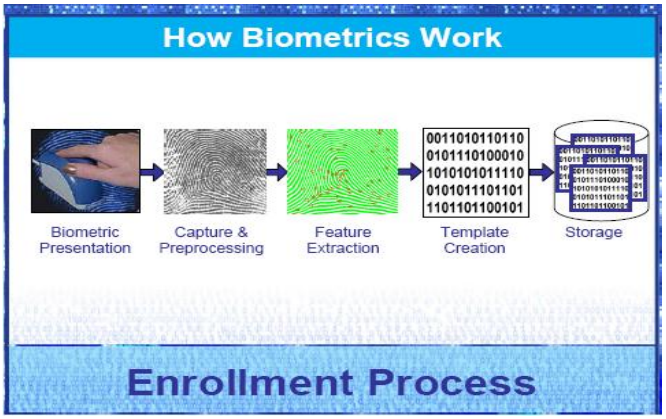


Figure 1 Biometric Enrollment Process

IV. INTELLIGENT AGENT BASED FLIGHT SEARCH AND SECURED BOOKING ARCHITECTURE

The Intelligent Agent based Search and secured Booking system i.e. IABFS-SBS [5] developed surrounds two important facets of the travel industry, which includes search for flights and booking the flights, which includes payment part. When a search is done by means of agent, which is replication of travel agent, it is important that the optimum results are returned to the user. Another factor to be considered is that the search must be efficient, quick and it must return only what is asked for which will prevent the user from spending a lot of time skimming through unnecessary information as it exists in the current system. Fig 2 depicts the search process flow in our system developed.

The booking of flights via a mobile device is similar to booking flights via a website but what is different is that in this situation we are not booking flights via the mobile device by accessing a website directly and doing the work just as we would if we were using our desktop or laptop computers. In order to successfully do this we have several factors to keep in mind such as; Security, Third Party Connectivity Channels, Encryption Mechanism, Biometric Implementation which forms a crucial part of the security platform. In addition, the booking of flight involves the usage of a customer's credit card, which includes confidential information. With this in mind, care has to be taken on how a customer transaction is processed via the mobile device as shown in Fig.3. We will now explain the roles and responsibilities of each component in the architectures shown in Figs 2 and 3.

1) Human Agent –

The human agent is the end user that is interested in searching for flights using this platform. This agent is the individual operating on the android mobile handset based on different criteria input in the system by this agent. Different queries are constructed and search is executed on the central database and the required results are returned to the human agent for viewing for proceeding with further search and executing the action of booking a flight.

2) Mobile Device –

The mobile device is the android handset that the application is installed on, and the human agent is using to perform search operations and booking operations. Additionally the device could be utilized to just view daily arrivals and departures.

3) Search Agent –

The Search Agent is assigned the roles and responsibilities such as: Construct Queries based on the user input., Query the central database for user information based on the query that was constructed., Return results of search to the device for the user's viewing., Sort the information so that they are returned to the user in an organized matter., Send user choice to Airline Agent if user is interested in booking a flight so that the airline can be validated.

4) Airline Agent –

The Airline Agent is assigned the roles and responsibilities such as: The airline agent validates the airline website as a first security measure before the system proceeds with the booking process., Transmits information to user when the airline has been validated., Receives booking and cancellation information from other agents., Updates airline database with booking or cancellation references., Ensures that when flight is booked user's flight information is reserved with airline.

5) Security Agent –

The Security Agent is assigned the roles and responsibilities such as: The security agent facilitates the reading of fingerprints and the encryption of fingerprint information. Communicates with trusted third party to transmit secure information such a biometric data or user encrypted payment information to airline agent., Establishes HTTPS SSL connection to ensure secure transfer of biometric information., Establishes HTTPS SSL connection with card agent to ensure secure transfer of user encrypted information to facilitate payment., Establish secure https SSL connection with airline agent to obtain certificate for trust verification., Receives passenger booking status verification results and securely transfers over https connection results to the search agent for presentation to the passenger.

6) Card Agent –

The Card Agent is assigned the roles and responsibilities such as : Receives payment or cancellation request accompanied with user encrypted information for processing of payment or cancellation., Establish secure https SSL connection with card companies to facilitate secure payment or cancellation of payment., Transmits status of payment or cancellation via https SSL connection to security agent.

7) Learner Agent –

The Leaner Agent is assigned the roles and responsibilities such as: The agent studies the search patterns of various users and constructs a pattern based on user's travel pattern. The agent is able to perform search based on learnt behavior and return results that the user would have chosen., The agent communicates the learnt behavior with the search agent to construct queries to perform on the central database.

8) Trusted Third Party –

The external trusted third party is assigned the roles and responsibilities such as: Securely liaise via https with Security Agent and Airline Agent to validate airline., Securely liaise with the security agent to receive biometric and encrypted information for validation purposes., Securely liaise with card agent to facilitate secure payment for booking of flight, Securely liaise with card agent to facilitate secure cancellation of flight, Securely verify applicant identity using encrypted biometric data.

9) Directory facilitator –

This is an integral part of the agent platform that operates by providing yellow page services to other agents. The DF maintains an accurate, complete and timely list of agents and must provide the most current information about agents in its directory to all authorized agents [29]

B. IABS-SBS Algorithm

The process flow of Intelligent Agent Based Search and Secured Booking Flight System has been shown as Flow Charts in Figs 4 &.5. We will expand this by providing an explanation of how this works. Initially the User accesses the application from the mobile android handset. The user has the option to view arrivals, view departures, search for flights and perform booking. Other sessions are as under.

1) Arrivals –

User selects Arrivals to view the latest information on arrivals available in the system**.** The information is displayed to the user organized according to dates**.,** When user selects one of the flights being displayed under any date displayed**.,** Details of the selected flight are displayed.

2) Departures –

User selects Departures to view the latest information on departures available in the system**.** The information is displayed to the user organized according to dates**.,** When user selects one of the flights being displayed under any date displayed**.,** Details of the selected flight are displayed.

3) Search and Booking-

User enters search criteria from a GUI on mobile device (Intelligent Agent) such as Departure and Arrival city using GMAP, price, percentage markup, rating, facilities etc.**,** Upon submission the search criteria is passed to the Search agent that starts the search process sending the request to the central database to look up a list of suitable airlines matching the search criteria as :

(i) *If airline is available for lower price range with exact or closest departure and arrival city matching the amenities, (ii) If airline is available for the price range specified with exact or closest departure and arrival city matching amenities, (iii)If no airlines available within the price range or so for exact or closest departure and arrival city, it finds an airline with the facilities for any price range.*

*Now t*he search agent interacts with a database that in turn forwards the results to the user along with the rating and popularity information., The search agent returns the result and displays it on the GUI of the mobile devices.**,** From the detail screen user selects to book the flight.**,** Search agent contacts the security agent for validating the airline agent.**,** After verification, details of flight including price and other pertinent information is displayed to user for confirmation.**,** After user confirms the information, user is asked to enter fingerprint information to begin the payment process.**,** User swipes finger, after which the fingerprint is encrypted and transmitted to security agent who verifies the user over HTTPS SSL connection by contacting the card agent.**,** After verification data is transmitted to card agent over HTTPS SSL connection to process payment.**,** Card agent facilitates payment via a HTTPS SSL connection and then uses a similar connection to send status to security agent.**,** Security agent communicates with airline agent the status of payment who in turn updates airline database with reservation details and Information is sent to the device and user is presented with an update and reference details.

4) Cancellation -

User selects status from device which presents option to view flight status or cancel flight.**,** User selects flight cancellation; this presents the option for user to enter booking reference and biometric signature which is user’s fingerprint.**,** Booking reference is verified against fingerprint for validation.**,** Once verification is complete, the fingerprint is encrypted and the airline is verified through the airline agent**.** Once airline is identified data is sent to security agent for verification**.,** once verification is done security agent connects airline agent with card agent**.,** User verifies transaction and accepts cancellation policy**.,** Ticket is cancelled and a status is sent to the mobile handset for the user’s viewing

C. Ratings and Popularity Index

The system is built with a ratings and popularity index feature to assist clients and prospective clients to be able to make informed decisions about their choice of flight and also after they have used the services of any airline they are able to provide feedback as to the services that were offered and provide ratings as to how good were these services.

1) Ratings:

The ratings of an airline are divided into a few categories and they are; Infrastructure, cleanliness, security, facilities, price, snacks and ground staff. All these areas are rated out of five stars with one star being the lowest and five stars being the highest meaning the service is excellent. The results from these different areas when collected are then averaged to arrive at a final rating value. So when there are ratings of a particular airline for example four stars, this result is based on the average as calculated from the areas mentioned earlier.

2) Popularity Index:

The popularity index of any airline is based on the average of the amount of times persons have booked and travelled the airline. The value for the PI ranges from 1 to 5 and follows the following interpretation where percentage indicates the number of persons opting for the airline in a particular sector.

- 1 is Poor : 0% and above to less than 15%
- 2 is Satisfactory: 15% and above to less than 30%
- 3 is Average: 30% and above to less than 50%
- 4 is Good : 50% and above to less than 80%
- 5 is Excellent: 80% and above

V. IMPLEMENTATION USING JADE-LEAP

The main purpose of IABFS-SBS is to enable travelers to search for flights according to user specification and also book online using biometric. The system is also equipped to view the Flight departure and arrival information and also how well the services are viewed by their customers. The system is created using Android 2.2 with Google API 8 and Java Agent Development Toolkit (JADE) with Lightweight Extensible Authentication Protocol (LEAP) [28-31]. The list of all agents implemented in this system is shown below in Fig.6. The abbreviated names that were introduced when giving details of the agents’ functionality is used.

A. Flight Search Implementation

The user normally would access the systems interface and select the Flight search option to initiate a flight search. The search capabilities of the system give the user the ability to perform search for departure and arrival city selection in two ways. The first method allows the user to use the Google map API which loads the Google map within the android environment for selecting the departure and arrival cities as shown in Figs 7 and 8. The other option to select the departure and arrival city is called the wild card search. Users here do not have to use the Google Map option; once they are aware of the city they are travelling to. They just start typing the name of the city. Once the users start typing the name of the city in the textbox field, the search agent immediately starts querying the database for cities that begins with the letters as they are typed. The results immediately appear as a drop down below the text box that the user is typing in. Since there may be several cities with the same name below, each option in the drop down box is a description telling where the city is so that the choice made by user can be more informative as shown in Figs 9 and 10. In this scenario, once the arrival and departure city is selected from the Google map or wild card search, the departure and arrival cities are loaded into the text boxes.

After that the user is presented with several other criteria such as price, price mark up, facilities like in-flight snacks, Ground service, wheel chair access etc to customize their search as shown in Fig 11a and b. So here once the user selects the criteria with a price of say $300.00 with 5% price markup and other facilities included such as Ground service, In-flight snacks, Airport Shuttle etc and feedback of 3 months, the search agent possess the intelligence to search the database and does not return a list of all flights but tunes the results to match exactly what is being searched for by the user i.e $300.00 from Miami to Kingston with facilities included as shown in Fig. 12a. Fig 12band c shows the complete details of flight returned with the average rating of 4 which is Good and popularity index of 5 which means more than 80% of travelers have opted this airline for this sector Miami to Kingston.

Let us consider another scenario where the user selects the criteria to fly from Miami i.e. Departure city to Portmore i.e Arrival city with a price of say $500.00 with price markup of 5% and all facilities included. Once the search is initiated, the Search agent will look for flights and retrieve the results for Miami to Ocho Rios or Kingston only as there exist no Airports in Portmore and obviously no flights from Miami to Portmore with all other criteria remaining the same.

The search agent here possess the intelligence to return the search results with the price of $525 by applying 5% price markup as per user selection and additionally chooses an airport closest to the city that the user selected which is Kingston and a message is displayed on screen that there is no airport in that city and hence another city is selected. The city selected is the closest one to what placed as a search criteria by the user. Fig 13a and b shows the complete Flight details with the rating of 4 which is Good and popularity index of 5 which shows again 80% and more travelers have opted this airline for this sector.

B. Intelligent Agent Learning

Till now we have seen as how the user selects the criteria which includes price, markup and also facilities. The facilities if left blank the agent would take some default value and return the results. So we here bring the learning capability of search agent where it is enough the user gives the basic criteria for search i.e Departure city, Arrival city, Price , Price markup, ratings and other parameters such as Facilities are left as blank. In here the search agent does not search flights taking the default value for facilities but possess the learning capability to search for flights for a particular price with facilities which is most commonly been used by users based on popularity information. This shows the past search experience and learning capability of search agent similar to what we would experience with the human travel agent.

Let us consider a scenario where user is searching flight for price of $300.00 with markup of 5% from Kingston to Toronto by leaving the facilities as blank with feedback of 3 months. The search agent here initiates the learner agent which retrieves the flights from Kingston to Toronto for price of $300.00 with facilities based on past search experience or agent learning capability having Excellent rating of 5 and popularity index which means more than 80% of users have opted this flight for this route and price with facilities as shown in Fig.15.

Let us consider another scenario where the user is searching for flight from Kingston to London with a price of $500.00 with some facilities. The Search Agent was unable to retrieve results that match the criteria requested by user and therefore the Learner Agent uses the past search experience and retrieved a flight with the exact price match of $500.00 with any facilities for the route as shown in Fig 16a which is been opted by most user as displayed by the popularity index which is 4 i.e 50% and more travelers have opted this airline and rating is 4. Also the Learner agent retrieved the flights for any price with matching facilities for the route as shown in Fig 16b which is opted by most users as displayed by popularity index i.e 4 and rating of 4.

Let us consider last scenario in Agent learning where user is looking for a flight with a price of $700.00 from Kingston to Chicago with facilities left as blank. The search agent was unable to retrieve the results and therefore the learner agent was initialized which again could not retrieve flights for the price quoted by the user for that route with facilities with past search experience. So the learner agent uses more intelligence and based on past experience have retrieved flights on that route with any price and any facilities as shown in 17a and b having excellent popularity index and rating 4. The breakdown of rating is shown in Fig.18 comprising of infrastructure, Cleanliness, Security, Facilities, Price, Snacks and Ground Staff

C. Flight Booking and Payment

Now that the search is completed, the customer would go about booking the flight of his choice. In here let us take one search agent results as shown in Fig.15 where there is option to book the flight which the customer selects to book on the android mobile device as shown in Fig.19 The security Agent collects the information from the user and transmits encrypted information to a third party to verify if the Airline website can be trusted or not as shown in Fig.20 by contacting the Airline Agent. After the Airline is successfully verified and validated by the security agent, the Flight information based on user preference is returned to the user from the Airline Agent for confirmation as shown in Fig.21.

Also the Airline cancellation policy also present to the user to accept for payment processing as shown in Fig.22. Once the user accepts this information and proceeds to book flight, the user is presented with the screen to input details of credit card and present biometric which in our case is the fingerprint. The novelity in our research is that the user need not remember card number or so. The only thing the user needs to know is the credit card type and give his biometric information for transaction as shown in Fig .23.

After the fingerprint is accepted, the system checks to see if the fingerprint is found in the database by contacting the card agent as shown in Fig 24. For that the fingerprint is encrypted and the encryption keys are generated using 256 bit encryption as shown in Fig 25. Once encrypted, the information is passed onto the security Agent which contacts the card agent to see if the Fingerprint is found in the database and the confirmation is sent to user's mobile handset. If the user fingerprint is not available in the database, then it is intimated to user mobile phone as record not found. This could be due to many reasons such as Fingerprint not given properly or user does not possess credit card etc. Once record found based on fingerprint data, the user credit card information is retrieved and displayed along with the final billing amount for the user to confirm and do final payment as shown in Fig 26. Fig 27 shows the payment being processed. Once payment processed successfully, the transaction information is sent to user's mobile handset by the card agent and also updated in database too as shown in Fig .28.

D. Flight Cancellation

For cancellation, the user selects the option to search for the flight using the ticket information that they have. After the passenger inputs the booking information and scans their fingerprint, the fingerprint is taken and validated towards refund of money by the Airline to the authenticated passenger, which depends on Airline cancellation policy. For verifying the fingerprint and successfully matching against the booking reference, the fingerprint is encrypted. This is done as the security agent will have to pass this information to the card agent to match against the records that exist for payment because the transaction must be posted against the correct credit card account as shown in Fig 29. The agent from the mobile handset sends request towards verifying the airline agent that is to be used to perform the cancellation as shown in Fig.30. The information is passed to the Security agent as shown in Fig.31 once the airline agent has been identified.

Even though the airline and booking reference is showed, the encrypted finger print is also passed to the Security agent as part of the process to do further checks before forwarded to the appropriate card agent as shown in Fig .32. The information is received by the Security Agent and the SA verifies the AA. After which the AA is connected with the Card Agent (CA) to facilitate the processing of the cancellation. The Card Agent would process the cancellation based on what type of card is used whether MasterCard or Visa. After the Airline Agent has submitted the information to the Card Agent and the card type has been verified there is a two step user verification process where the flight information is displayed to the user to verify to ensure that the correct flight is being cancelled and the correct amount be posted to the account.

The second step of the user verification process is the display of the cancellation policy so that user is reminded of the consequences to cancelling a flight and accepts the policies. If user accepts with the policies then user moves accepting and continuing with the process as shown in Fig.33 After the user accepts the terms and conditions for cancelling the flight the Card Agent proceeds to cancelling the flight by reversing the transaction a shown in Fig.34 The Card Agent cancels the transaction and credits the customer's account and transmits the information to the Airline Agent so that the information can be updated with the airline and in the central database. After the flight is successfully cancelled the status information is sent to the user after which the Airline Agent database and the central database are updated with the flight cancellation information as shown in Fig.35.

There are situations where popularity index is shown as Average that shows that less than 50% of travelers have only used this airline for a particular sector as shown in Fig .36 which is indicated by means of warning message on user's mobile handset. There is another situation which the user can experience in agent learning results where the popularity is good i.e. 50% of traveler has opted but rating is below 3 as shown in Fig .37 which is indicated by warning message. Airline Agent interface provide information of ratings and popularity for airlines across various continents. Ratings and Popularity can be viewed for all airlines by continent and based on selected periods, which range from three month periods to up to a year. Database agent sends warning message because some airlines contain below normal values for ratings and popularity and removed from database after period of 3 months.

VI. CONCLUSION & FUTURE WORK

With the advancement of technology you find many persons with Smartphone that operates at level and capability of a desktop PC or laptop. Most large corporations today provide Smartphone to managers and supervisors. With all this in mind users are now capable of accessing various applications and services from their mobile devices. Therefore IABFS-SBS seeks to address gaps that exist where persons can continue with their busy life styles and yet book flights, cancel flights, view airline popularity and ratings based on frequency of travel by passengers, view real time arrivals and departures. There are several applications in existence that offers travelers the capability to book flights, and hotel rooms and outline a travel schedule as was previously discussed. However, IABFS-SBS offers to the Android platform specifically the ability to enjoy Intelligent Agent based flight search that has the added facility of agent learning that studies patterns and returns user specific information to the potential passenger. Additionally the system is replete with security, offering biometric encrypted features used when booking or cancelling a flight, which offers potential users enough comfort that even though the operation is being done from a mobile device.

The system provides real time viewing of flight arrivals and departures but this is just for the users viewing as future work. Users could be allowed to select any flight they see in the departure dashboard and book it. Additionally the system is built to provide Intelligent Agent based flight search and secured booking capabilities but as we look at trends with similar applications and similar facilities we may find the need in the future to expand the work and offer potential users a more holistic service in terms of not just flight search and booking but assistance with booking of hotels using agent technology and the popularity and rating module built within the current application. User could also get assistance with taxi reservations. As the android platform increases its capabilities and Smartphone technology get more advanced in the future, the system could look at real time communication with its users by sending text messages to advise them of delays, snow storms and airline details being offered. Last but not the least, the system could also facilitate Agent Based mobile check in, so that when users get to the airport things run a little smother.

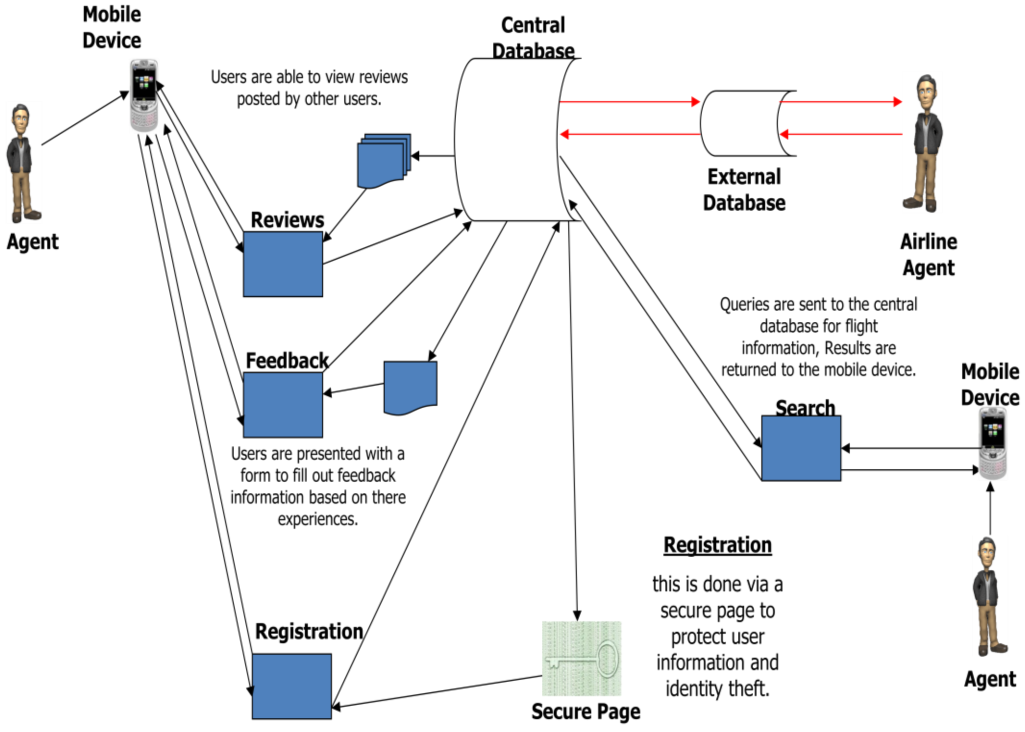

Mobile Device
Agent
Users are able to view reviews posted by other users.
Central Database
External Database
Airline Agent
Reviews
Feedback
Users are presented with a form to fill out feedback information based on there experiences.
Queries are sent to the central database for flight information, Results are returned to the mobile device.
Search
Mobile Device
Agent
Registration
this is done via a secure page to protect user information and identity theft.
Registration
Secure Page

Fig 2 Intelligent Agent based Flight Search

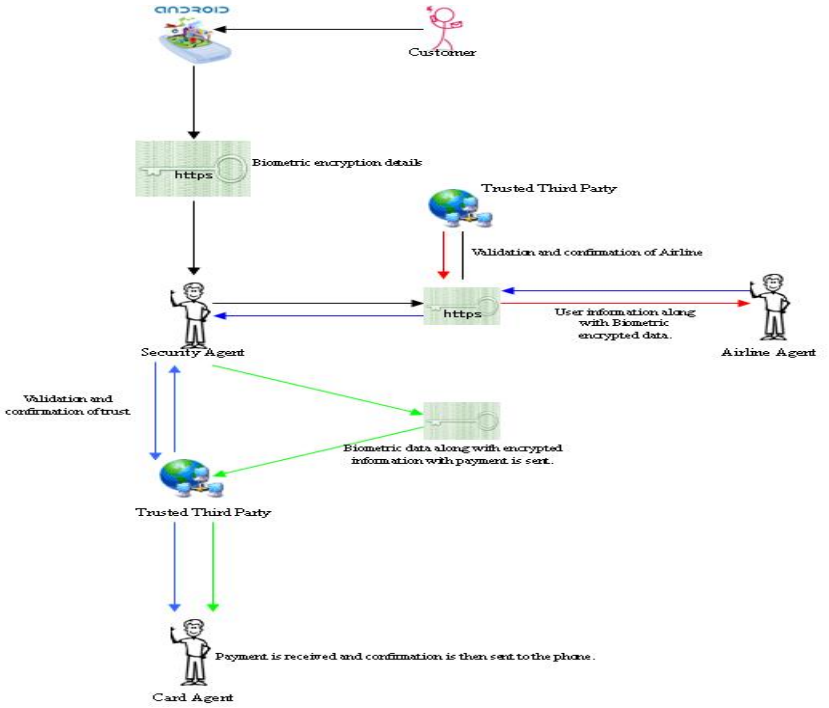

android
Customer
https
Biometric encryption details
Trusted Third Party
Validation and confirmation of Airline
Security Agent
https
User information along with Biometric encrypted data.
Airline Agent
Validation and confirmation of trust
Biometric data along with encrypted information with payment is sent.
Trusted Third Party
Payment is received and confirmation is then sent to the phone.
Card Agent

Figure 3 Intelligent Agent based Booking and Payment Architecture

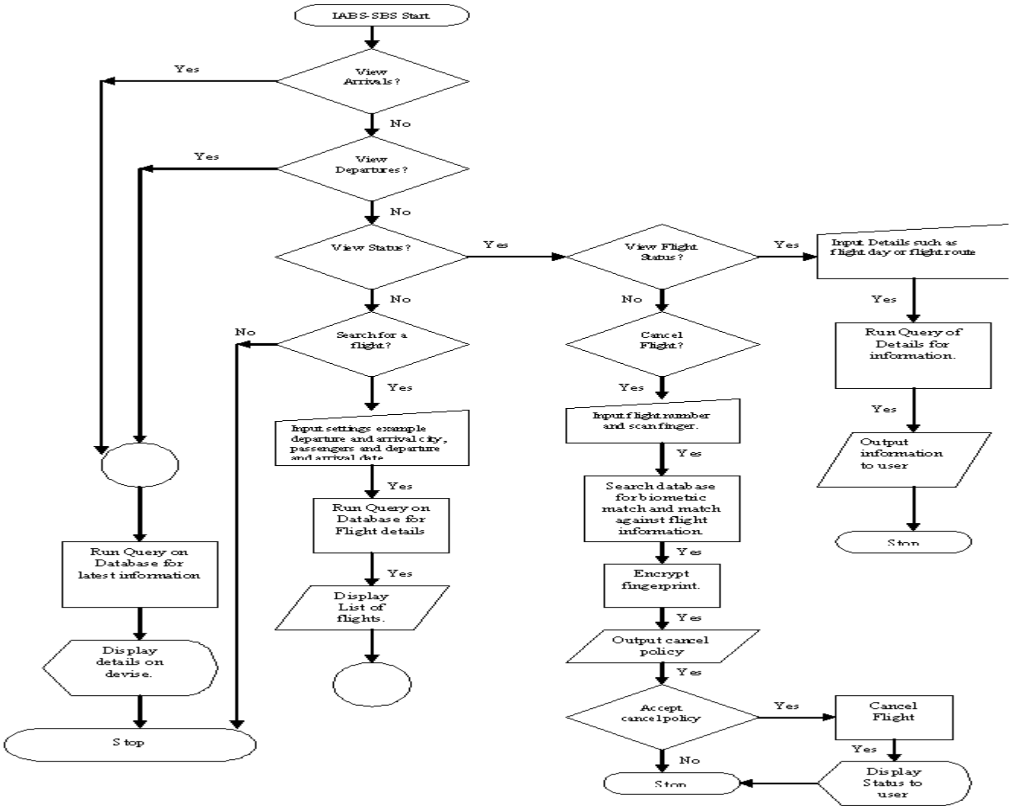

IABS-SBS Start
View Arrivals?
View Departures?
View Status?
View Flight Status?
Input Details such as flight day or flight route
Run Query of Details for information.
Output information to user
Stop
Search for a flight?
Cancel Flight?
Input flight number and scan finger.
Search database for biometric match and match against flight information.
Encrypt fingerprint.
Output cancel policy
Accept cancel policy
Cancel Flight
Display Status to user
Stop
Input settings example departure and arrival city, passengers and departure and arrival date
Run Query on Database for Flight details
Display List of flights.
Run Query on Database for latest information
Display details on devise.
Stop
Yes
No

Figure 4 IABS-SBS Process Flow

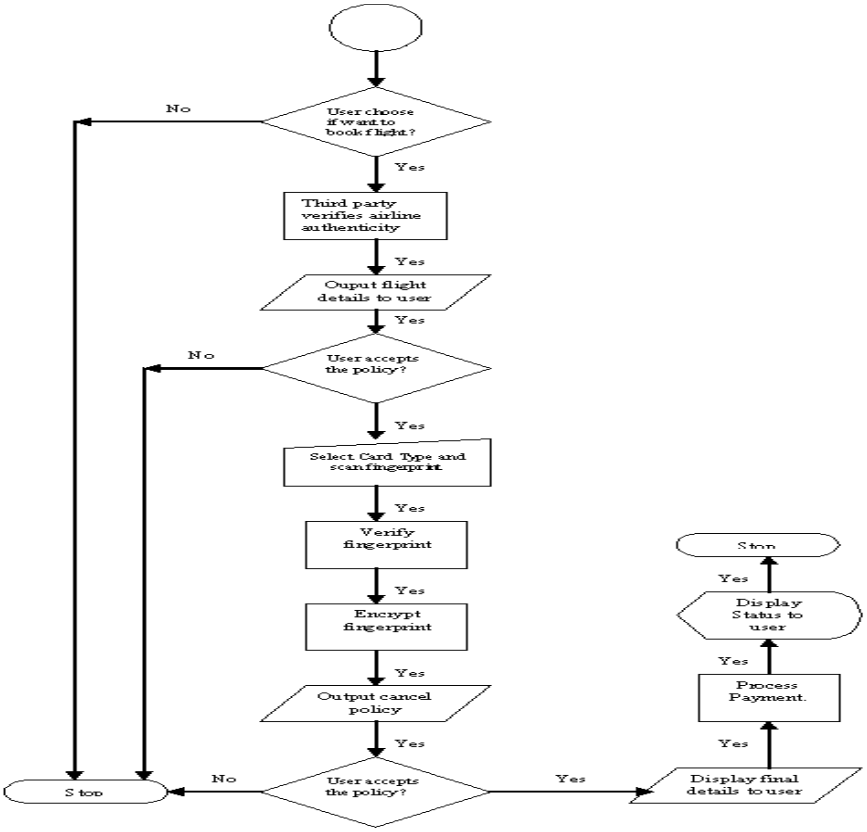
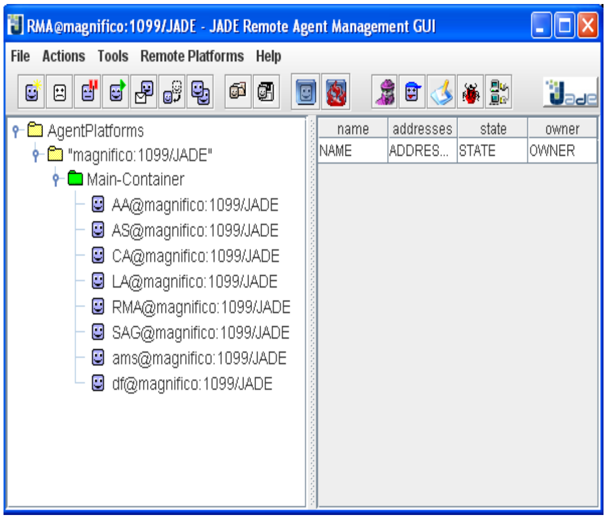

User choose if want to book flight?
Third party verifies airline authenticity
Ouput flight details to user
User accepts the policy?
Select Card Type and scan fingerprint
Verify fingerprint
Encrypt fingerprint
Output cancel policy
User accepts the policy?
Display final details to user
Process Payment.
Display Status to user
Stop
Stop
Yes
No

Figure 5 IABS-SBS Process Flow (contd)



AgentPlatforms
"magnifico:1099/JADE"
Main-Container
AA@magnifico:1099/JADE
AS@magnifico:1099/JADE
CA@magnifico:1099/JADE
LA@magnifico:1099/JADE
RMA@magnifico:1099/JADE
SAG@magnifico:1099/JADE
ams@magnifico:1099/JADE
df@magnifico:1099/JADE

name	addresses	state	owner
NAME	ADDRES...	STATE	OWNER

Figure 6 Agents in the JADE Container

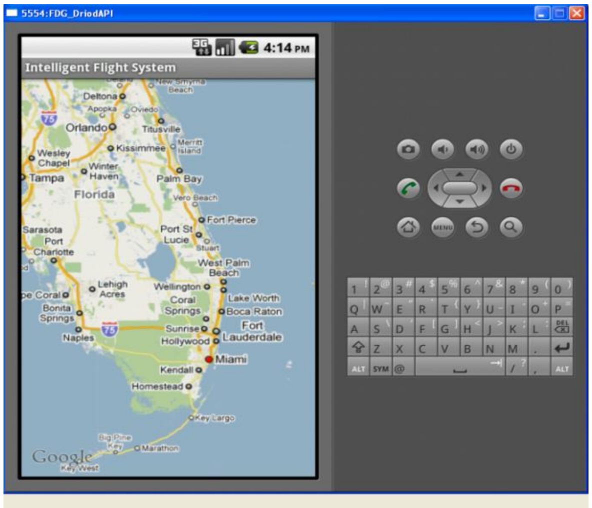


Figure. 7 Arrival City Selection

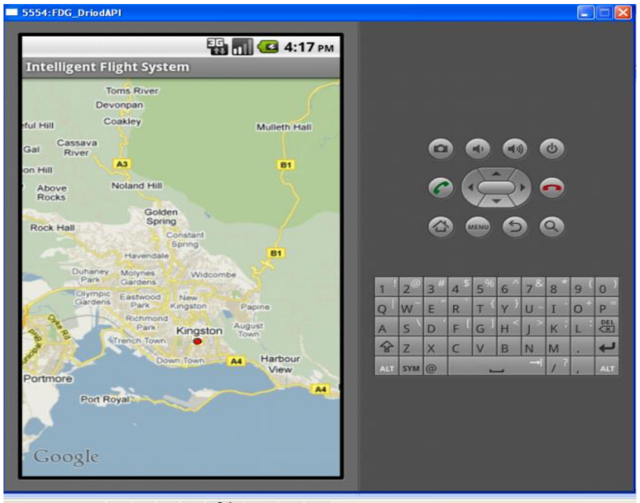


Figure. 8 Departure City Selection

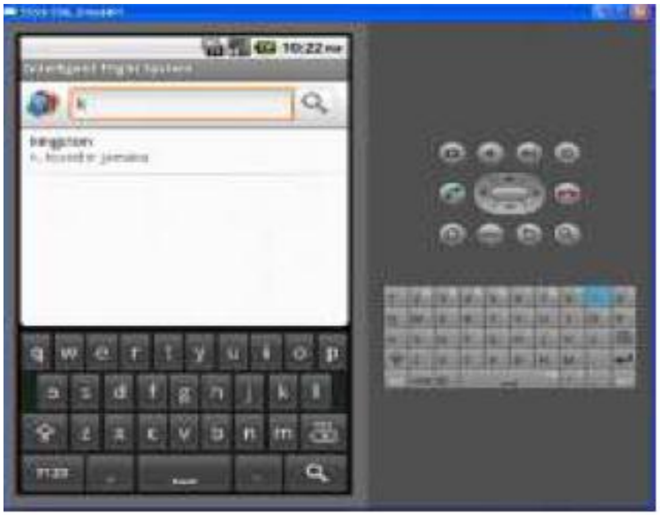


Figure.10 Departure City Wild Card Search

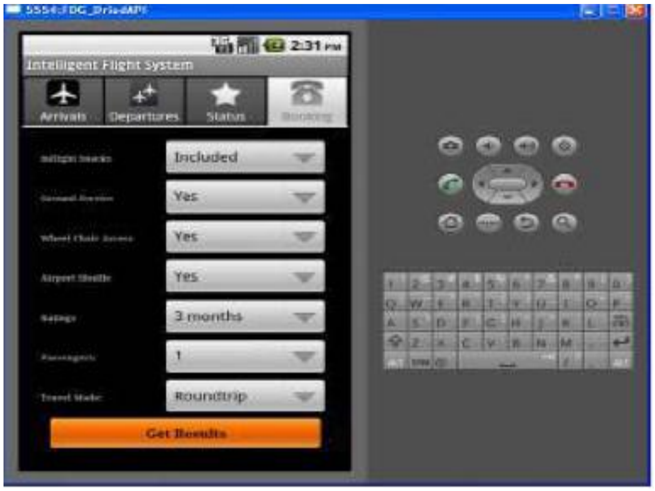


Figure. 11b Search Screen-1

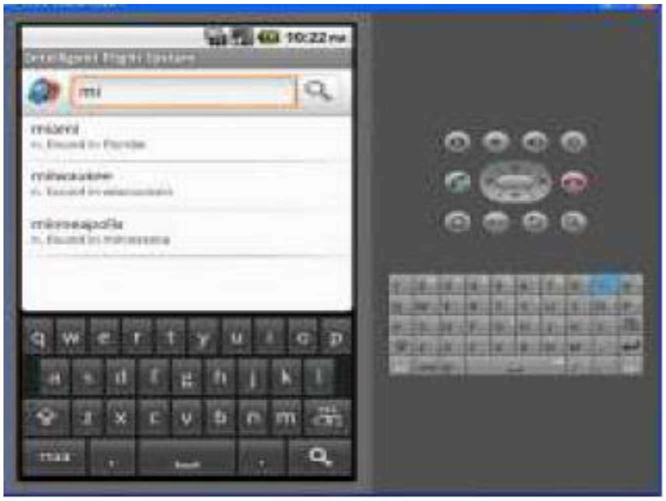
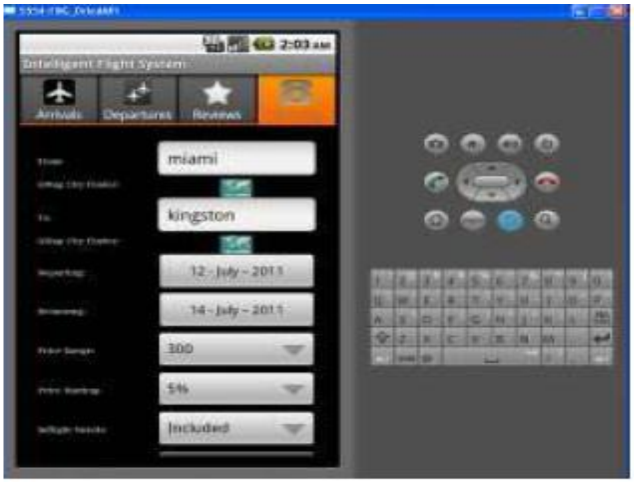
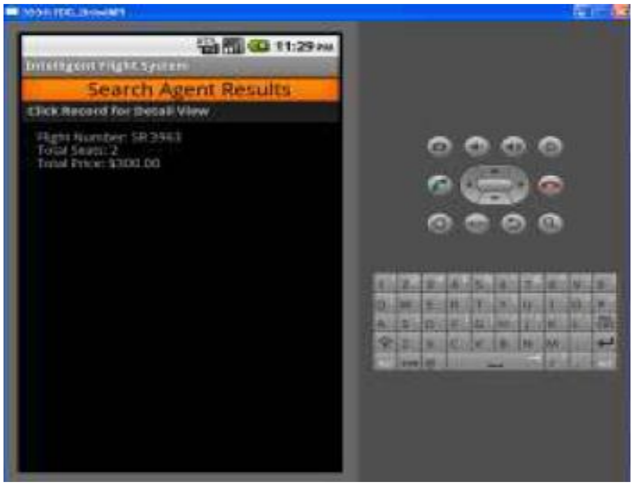


Figure.9 Arrival City Selection



miami
kingston
12 - July - 2011
14 - July - 2011
300
5%
Included

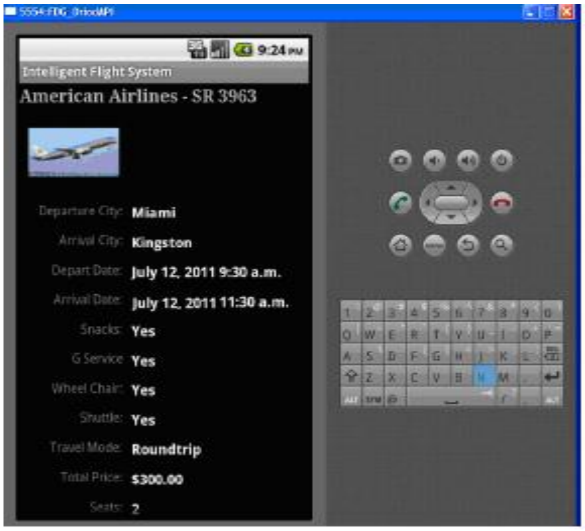
Figure. 11a Search Screen-1



Search Agent Results
Click Record for Detail View

Flight Number: SR 3963
Total Seats: 2
Total Price: $300.00

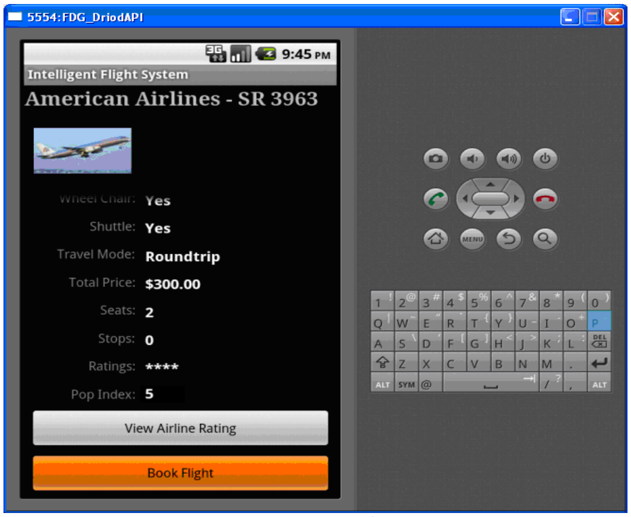
Figure. 12a Search Results-1



American Airlines - SR 3963

Departure City:	Miami
Arrival City:	Kingston
Depart Date:	July 12, 2011 9:30 a.m.
Arrival Date:	July 12, 2011 11:30 a.m.
Snacks:	Yes
G Service:	Yes
Wheel Chair:	Yes
Shuttle:	Yes
Travel Mode:	Roundtrip
Total Price:	$300.00
Seats:	2

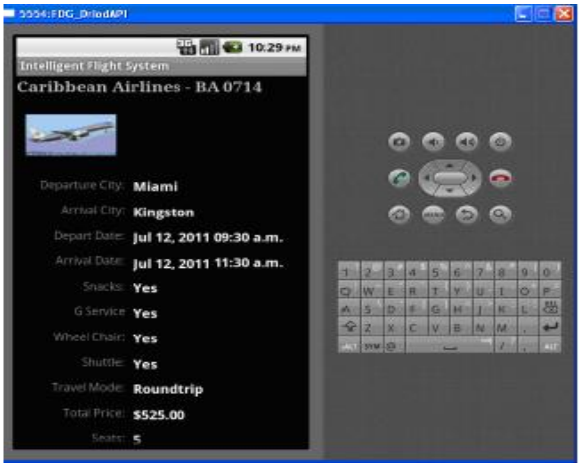
Figure.12b Detailed Results-1



American Airlines - SR 3963

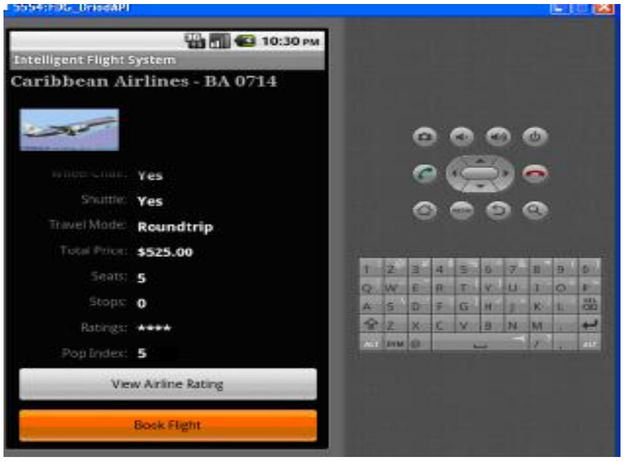
Wheel Chair:	Yes
Shuttle:	Yes
Travel Mode:	Roundtrip
Total Price:	$300.00
Seats:	2
Stops:	0
Ratings:	****
Pop Index:	5

Figure.12c Detailed View of Results-2



Caribbean Airlines - BA 0714

Departure City:	Miami
Arrival City:	Kingston
Depart Date:	Jul 12, 2011 09:30 a.m.
Arrival Date:	Jul 12, 2011 11:30 a.m.
Snacks:	Yes
G Service:	Yes
Wheel Chair:	Yes
Shuttle:	Yes
Travel Mode:	Roundtrip
Total Price:	$525.00
Seats:	5

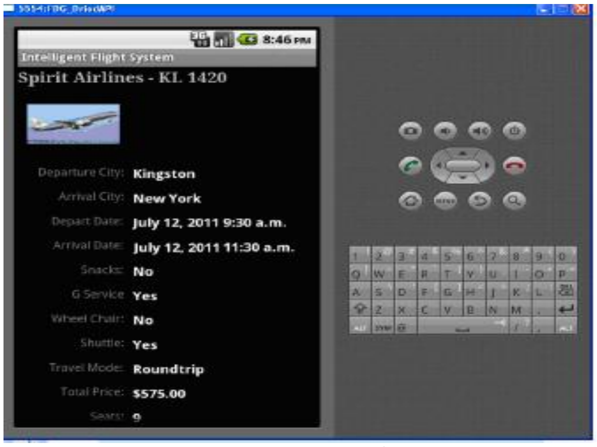


Caribbean Airlines - BA 0714

Wheel Chair:	Yes
Shuttle:	Yes
Travel Mode:	Roundtrip
Total Price:	$525.00
Seats:	5
Stops:	0
Ratings:	****
Pop Index:	5

Figure.13a Detailed View of Results-3

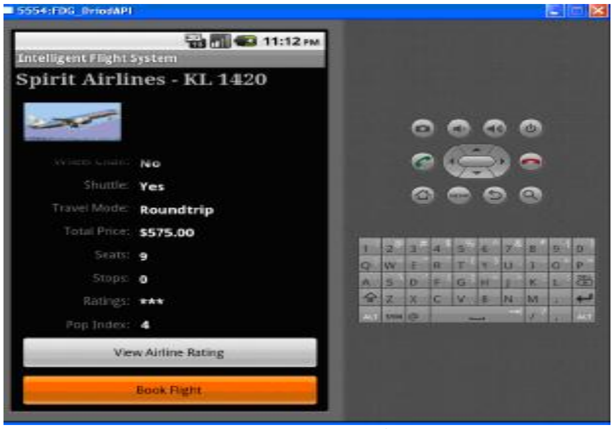
Figure.13b Detailed View of Results-3



Spirit Airlines - KL 1420

Departure City:	Kingston
Arrival City:	New York
Depart Date:	July 12, 2011 9:30 a.m.
Arrival Date:	July 12, 2011 11:30 a.m.
Snacks:	No
G Service:	Yes
Wheel Chair:	No
Shuttle:	Yes
Travel Mode:	Roundtrip
Total Price:	$575.00
Seats:	9

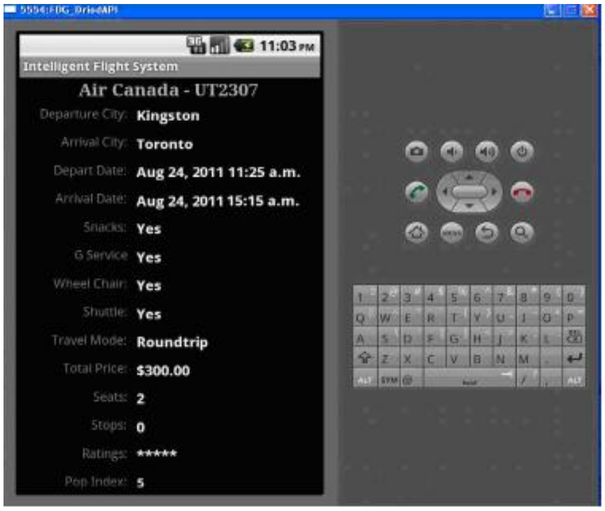
Figure.14a Detailed View of Results-5



Spirit Airlines - KL 1420

Wheel Chair:	No
Shuttle:	Yes
Travel Mode:	Roundtrip
Total Price:	$575.00
Seats:	9
Stops:	0
Ratings:	***
Pop Index:	4

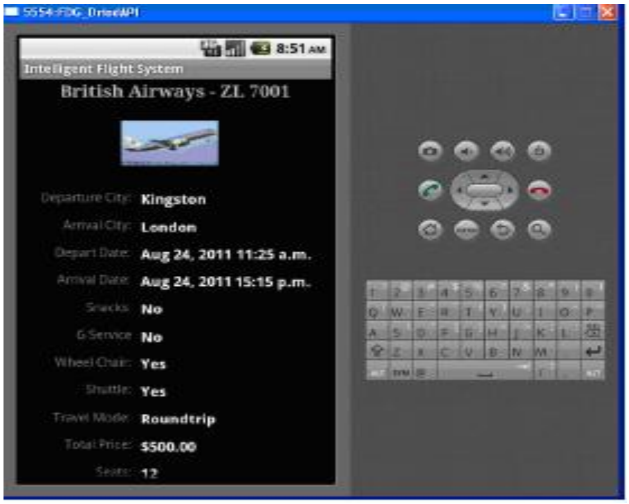
Figure.14b Detailed View of Results-6



Air Canada - UT2307

Departure City:	Kingston
Arrival City:	Toronto
Depart Date:	Aug 24, 2011 11:25 a.m.
Arrival Date:	Aug 24, 2011 15:15 a.m.
Snacks:	Yes
G Service:	Yes
Wheel Chair:	Yes
Shuttle:	Yes
Travel Mode:	Roundtrip
Total Price:	$300.00
Seats:	2
Stops:	0
Ratings:	*****
Pop Index:	5

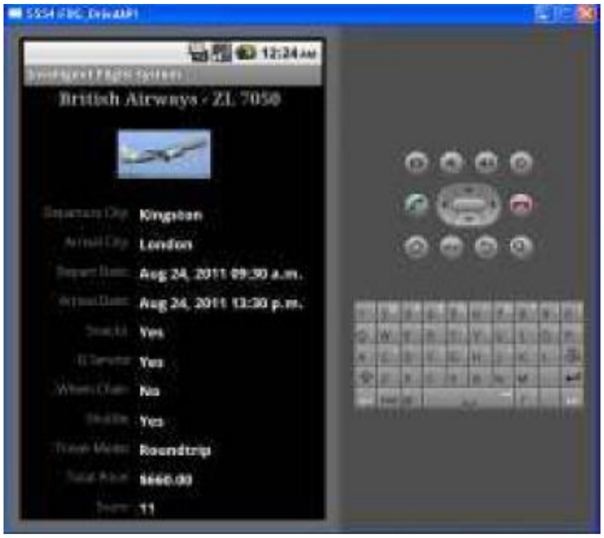
Figure.15 Detailed View of Results-7



British Airways - ZL 7001

Departure City:	Kingston
Arrival City:	London
Depart Date:	Aug 24, 2011 11:25 a.m.
Arrival Date:	Aug 24, 2011 15:15 p.m.
Snacks:	No
G Service:	No
Wheel Chair:	Yes
Shuttle:	Yes
Travel Mode:	Roundtrip
Total Price:	$500.00
Seats:	12

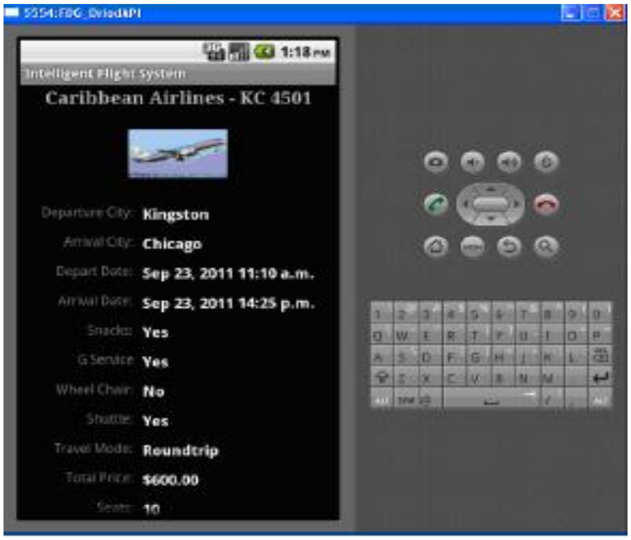
Figure.16a Detailed View of Results-8



British Airways - ZL 7050

Departure City:	Kingston
Arrival City:	London
Depart Date:	Aug 24, 2011 09:30 a.m.
Arrival Date:	Aug 24, 2011 13:30 p.m.
Snacks:	Yes
G Service:	Yes
Wheel Chair:	No
Shuttle:	Yes
Travel Mode:	Roundtrip
Total Price:	$660.00
Seats:	11

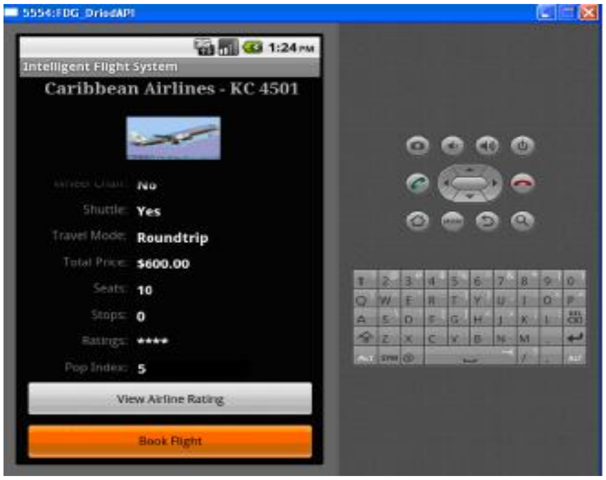
Figure.16b Detailed view of Results-9



Caribbean Airlines - KC 4501

Departure City:	Kingston
Arrival City:	Chicago
Depart Date:	Sep 23, 2011 11:10 a.m.
Arrival Date:	Sep 23, 2011 14:25 p.m.
Snacks:	Yes
G Service:	Yes
Wheel Chair:	No
Shuttle:	Yes
Travel Mode:	Roundtrip
Total Price:	$600.00
Seats:	10

Figure.17a Detailed view of Results-10

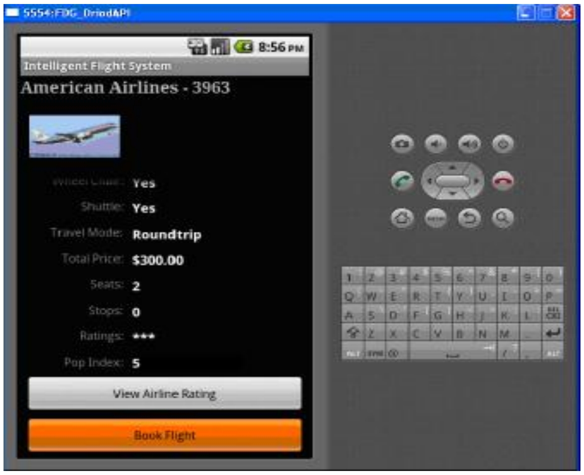


Intelligent Flight System

Caribbean Airlines - KC 4501

Wheel Chair: No
Shuttle: Yes
Travel Mode: Roundtrip
Total Price: $600.00
Seats: 10
Stops: 0
Ratings: ****
Pop Index: 5

Figure.18 Breakdown of Rating

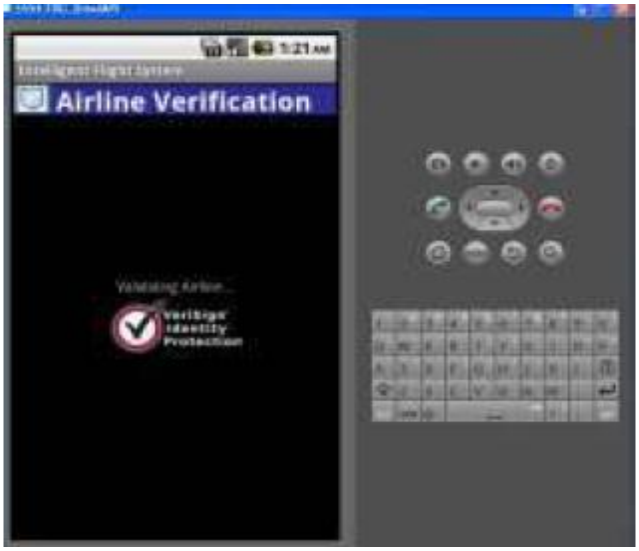


Intelligent Flight System

American Airlines - 3963

Wheel Chair: Yes
Shuttle: Yes
Travel Mode: Roundtrip
Total Price: $300.00
Seats: 2
Stops: 0
Ratings: ***
Pop Index: 5

Figure.19 Book Flight Initiated

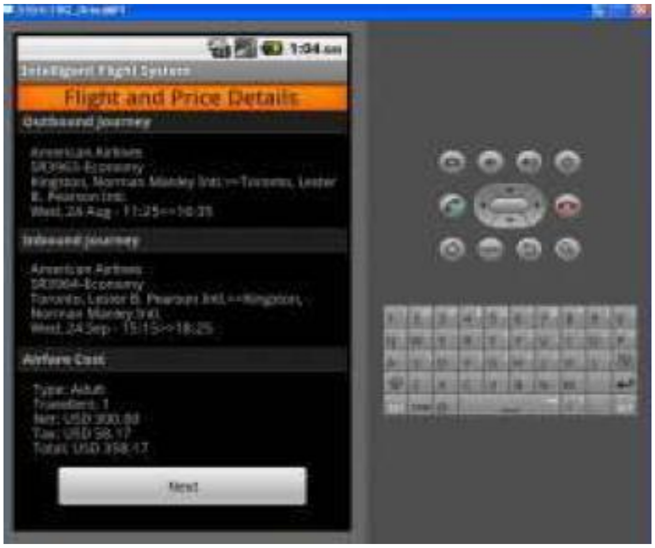


Intelligent Flight System

Airline Verification

Validating Airline ...

Figure.20 Airline Verification



Intelligent Flight System

Flight and Price Details

Outbound Journey

American Airlines
SR3963-Economy
Kingston, Norman Manley Intl.-->Toronto, Lester B. Pearson Intl.
Wed, 24 Aug - 11:25-->16:35

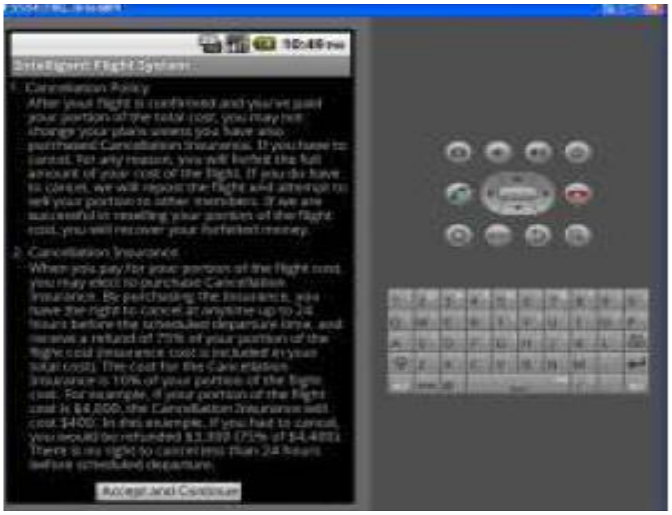
Inbound Journey

American Airlines
SR3964-Economy
Toronto, Lester B. Pearson Intl.-->Kingston, Norman Manley Intl.
Wed, 24 Sep - 15:15-->18:25

Airfare Cost

Type: Adult
Travellers: 1
Net: USD 300.00
Tax: USD 58.17
Total: USD 358.17

Figure.21 Flight Details for Verification

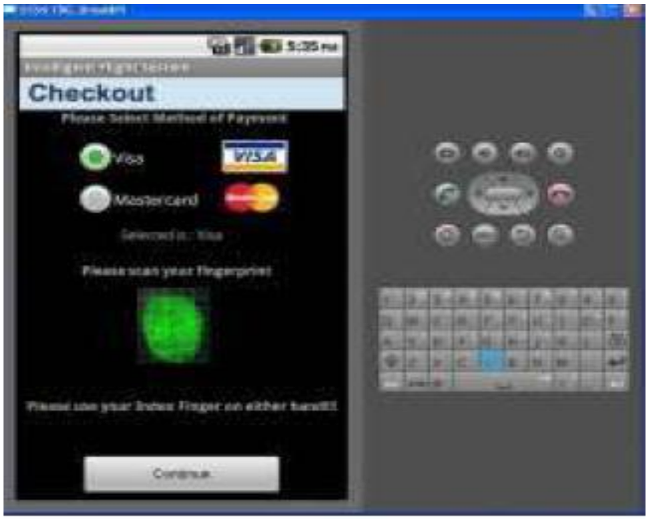


Intelligent Flight System

1. Cancellation Policy
After your flight is confirmed and you've paid your portion of the total cost, you may not change your plans unless you have also purchased Cancellation Insurance. If you have to cancel, for any reason, you will forfeit the full amount of your cost of the flight. If you do have to cancel, we will repost the flight and attempt to sell your portion to other members. If we are successful in reselling your portion of the flight cost, you will recover your forfeited money.

2. Cancellation Insurance
When you pay for your portion of the flight cost you may elect to purchase Cancellation Insurance. By purchasing the insurance, you have the right to cancel at anytime up to 24 hours before the scheduled departure time, and receive a refund of 75% of your portion of the flight cost (insurance cost is included in your total cost). The cost for the Cancellation Insurance is 10% of your portion of the flight cost. For example, if your portion of the flight cost is $4,000, the Cancellation Insurance will cost $400. In this example, if you had to cancel, you would be refunded $3,300 (75% of $4,400). There is no right to cancel less than 24 hours before scheduled departure.

Figure.22 Flight Cancellation Policy



Intelligent Flight System

Checkout

Please Select Method of Payment

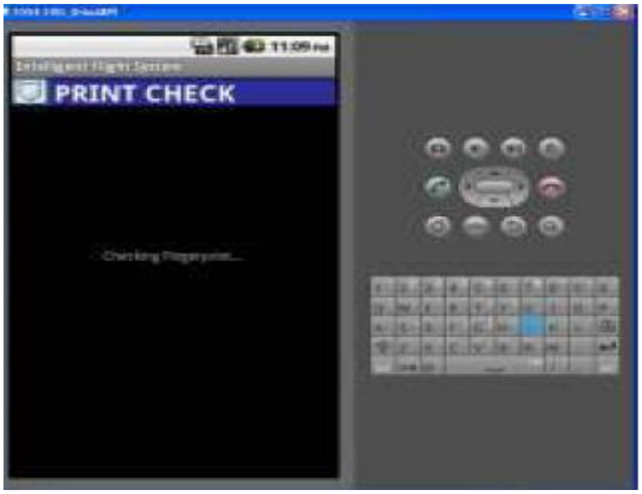
● Visa
○ Mastercard

Selected is : Visa

Please scan your fingerprint

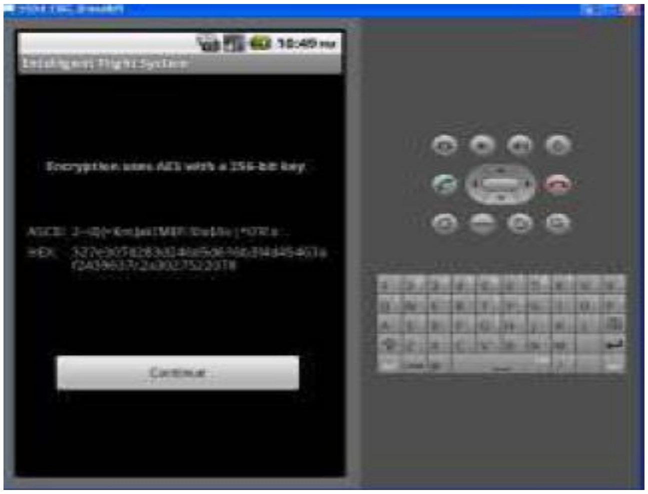
Please use your Index Finger on either hand!!!

Figure.23 Financial Information



Checking Fingerprint...

Figure.24 Fingerprint Checking

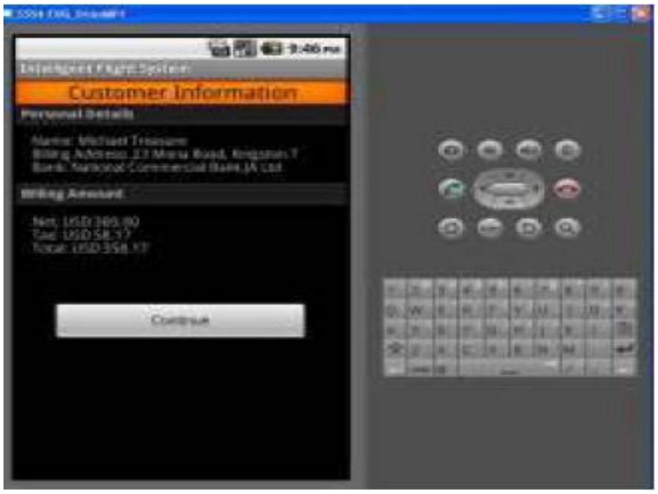


Encryption uses AES with a 256-bit key

ASCII: [illegible]

HEX: [illegible]

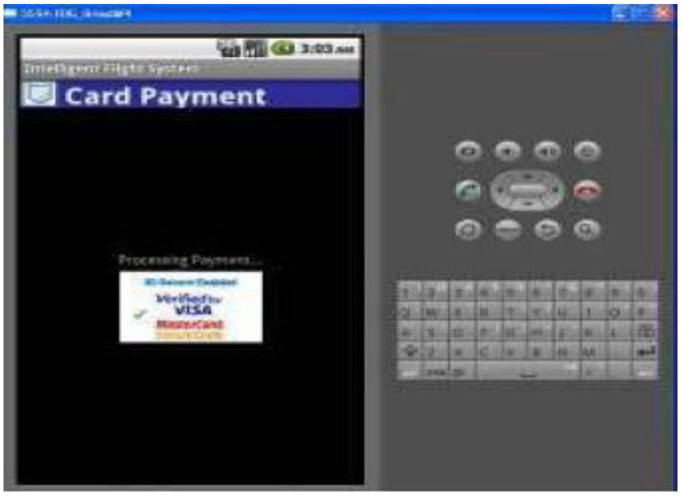
Figure.25 Fingerprint Encrypted



Name: Michael Treasure
Billing Address: 23 Mona Road, Kingston 7
Bank: National Commercial Bank JA Ltd

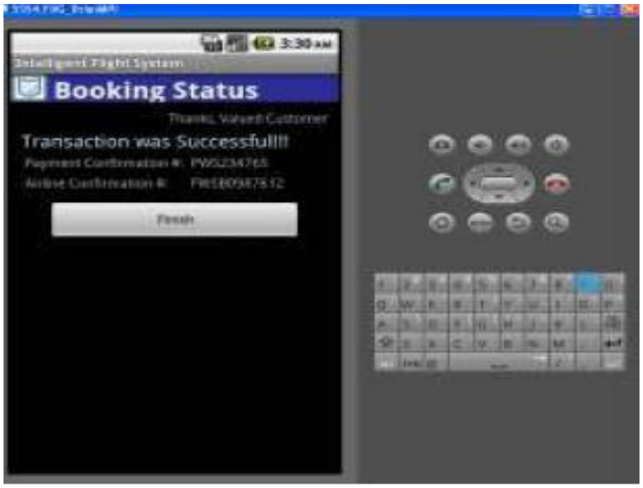
Net: USD 300.00
Tax: USD 58.17
Total: USD 358.17

Figure.26 Payment Confirmation



Processing Payment...

Figure.27 Payment Processing

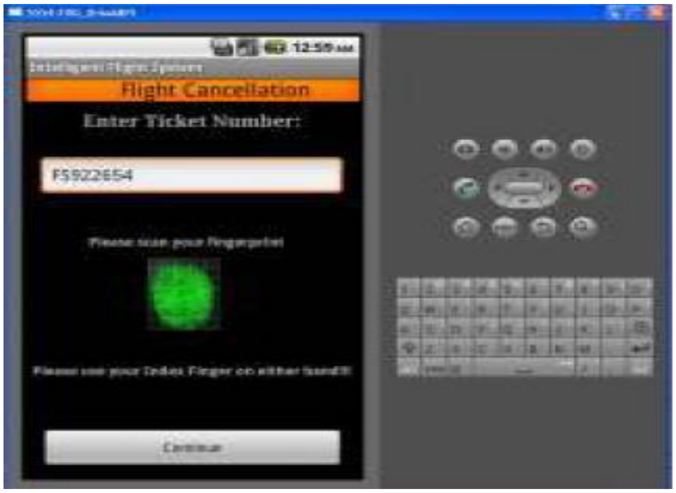


Thanks, Valued Customer

Transaction was Successfull!!

Payment Confirmation #: PW52234765
Airline Confirmation #: FW580987612

Figure.28 Payment Transaction



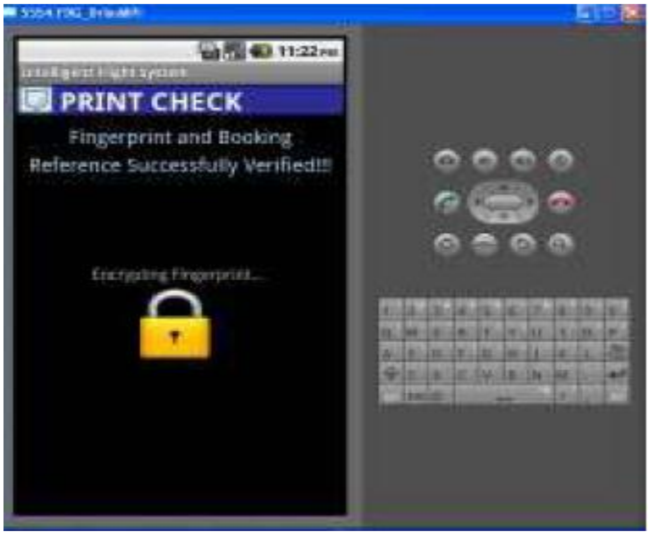
Enter Ticket Number:

FS922654

Please scan your fingerprint

Please use your Index Finger on either hand!!!

Figure.29 Ticket Cancellation



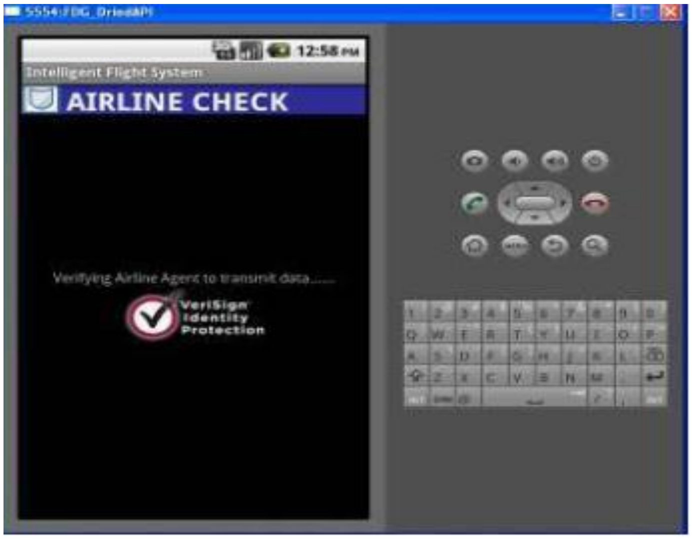
Intelligent Flight System

PRINT CHECK

Fingerprint and Booking Reference Successfully Verified!!!

Encrypting Fingerprint...

Figure.30 Fingerprint and Booking Validation

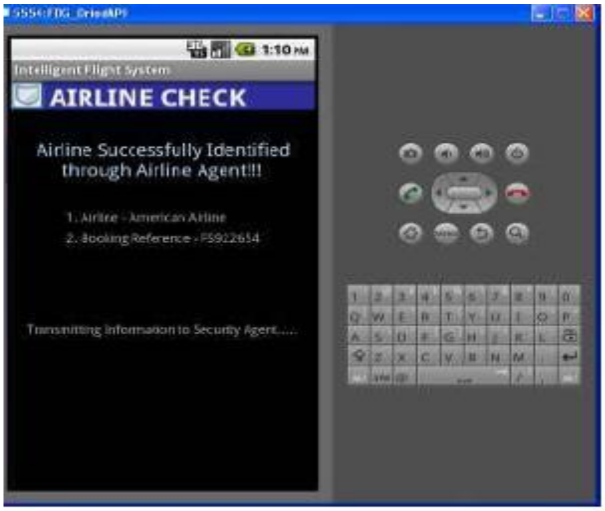


Intelligent Flight System

AIRLINE CHECK

Verifying Airline Agent to transmit data......

Figure.31 Airline Agent Verification



Intelligent Flight System

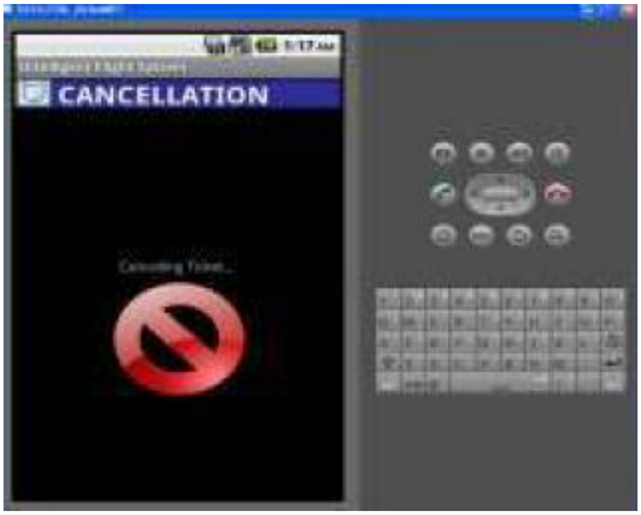
AIRLINE CHECK

Airline Successfully Identified through Airline Agent!!!

1. Airline - American Airline
2. Booking Reference - FS922654

Transmitting Information to Security Agent.....

Figure.32 Security Agent Transmission

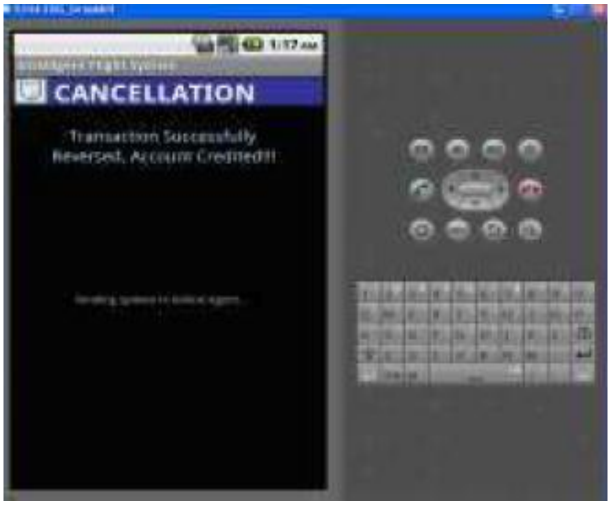


Intelligent Flight System

CANCELLATION

Canceling Ticket...

Figure.33 Flight Cancellation



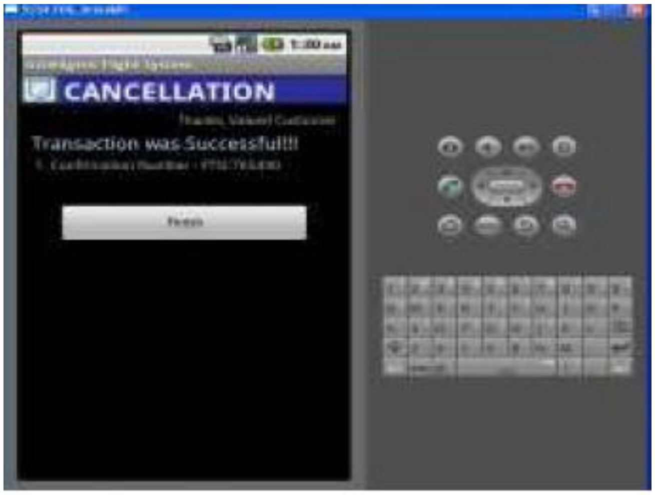
Intelligent Flight System

CANCELLATION

Transaction Successfully Reversed, Account Credited!!!

[illegible]

Figure. 34 Flight Cancelled



Intelligent Flight System

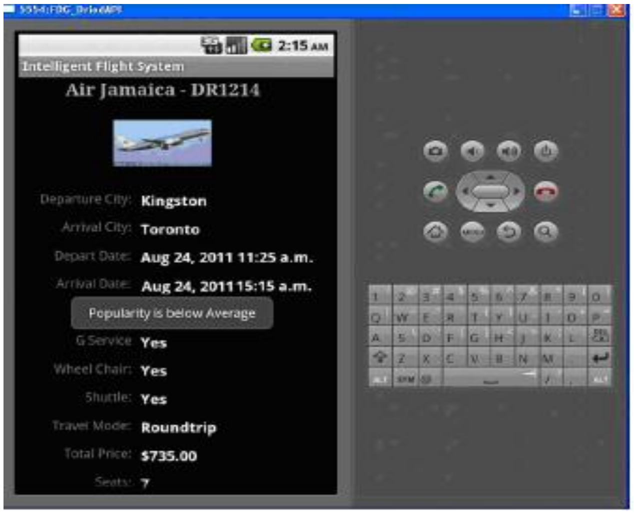
CANCELLATION

Thanks, Valued Customer

Transaction was Successful!!!

1. Confirmation Number - [illegible]

Figure.35 Flight Cancellation Status



2:15 AM

Intelligent Flight System

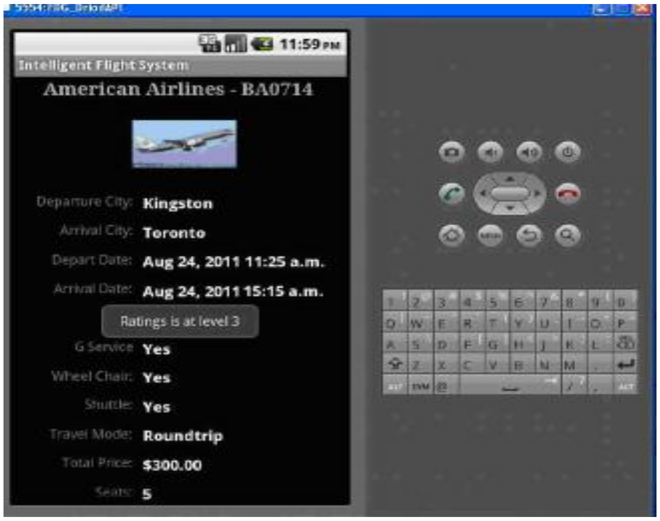
Air Jamaica - DR1214

Departure City: Kingston
Arrival City: Toronto
Depart Date: Aug 24, 2011 11:25 a.m.
Arrival Date: Aug 24, 2011 15:15 a.m.

Popularity is below Average

G Service: Yes
Wheel Chair: Yes
Shuttle: Yes
Travel Mode: Roundtrip
Total Price: $735.00
Seats: 7

Figure.36 Results with Low Popularity



11:59 PM

Intelligent Flight System

American Airlines - BA0714

Departure City: Kingston
Arrival City: Toronto
Depart Date: Aug 24, 2011 11:25 a.m.
Arrival Date: Aug 24, 2011 15:15 a.m.

Ratings is at level 3

G Service: Yes
Wheel Chair: Yes
Shuttle: Yes
Travel Mode: Roundtrip
Total Price: $300.00
Seats: 5

Figure.37 Results with low Rating

REFERENCES

[1] Information, Technology and Tourism. (2009). Retrieved from http://EzineArticles.com/3292939

[2] Computer Researvation System. (2004). Retrieved from http://www.cbo.gov/ftpdocs/55xx/doc5541/doc02b-part_4.pdf

[3] Videcom Airline Reservation System. (2004). Retrieved from http://www.videcom.com/general_overview.htm

[4] Travellers Warning. (2011). Retrieved,from http://www.computerworld.com.au/article/405074/scamwatchj_warns_travellers_fake_flight_booking_sites

[5] Garvey (2011). "Secured Agent based Mobile Airline Search and Secured Booking system", Unpublished M.Sc Dissertation, Department of computing, University of WestIndies, Jamaica

[6] Michaelides, A., & Moraitakis, N. (1997). Intelligent Software Agents in Travel Reservation Systems

[7] Franklin, S., & Graesser, A. (1996). Is it an Agent, or just a Program?: A Taxonomy for Autonomous Agents. Third International Workshop on Agent Theories Architectures and Languages. Springer-Verlag

[8] Pincemaille, C. (2008). Intelligent Agent Technology Cork Institute of Technology, Ireland, Bishopstown, 2008

[9] Russell S, Norvig P (1995) Artificial Intelligence: A Modern Approach, Prentice Hall Series in Artificial Intelligence. Englewood Cliffs, New Jersey

[10] Wooldridge & Jennings. (1995). Intelligent Agents: Theory And Practice. The Knowledge Engineering Review, Vol. 10:2, Pp 115-152

[11] Machine Learning. (2011). Retrieved from http://en.wikipedia.org/wiki/Machine_learning

[12] Mitchell, T. (1997). Machine Learning. McGraw Hill

[13] Hannan, J. F. (1957). Approximation to Bayes risk in repeated plays. Contributions to the Theory of Games, 3:97–139.

[14] Blackwell, D. (1956). Controlled random walks. In Proceedings of the International Congress of Mathematicians, volume 3, pages 336–338. North-Holland

[15] Fudenberg, D. and Levine, D. (1995). Universal consistency and cautious fictitious play. Journal of Economic Dynamics and Control, 19:1065–1089

[16] Young, H. P. (2004). Strategic Learning and Its Limits. Oxford University Press.

[17] Kaelbling, L. P., et al (1996). Reinforcement learning: A survey. Journal of Artificial Intelligence Research, 4:237–285.

[18] Shoham, Y., Powers, R., and Grenager, T. (2004). On the agenda(s) of research on multi-agent learning. In AAAI 2004 Symposium on Artificial Multi-Agent Learning [FS-04-02]. AAAI Press.

[19] Sutton, R. S. and Barto, A. G. (1998). Reinforcement Learning: An Introduction. MIT Press

[20] Alonso, E., & Inverno, M., D., (2001). Learning in multi-agent systems, UK:Cambridge University Press.

[21] Artificial Intelligence. (1996). Retrieved from http://www.cs.washington.edu/homes/lazowska/cra/ai.html

[22] Travel Agents. (2001). Retrieved from http://www.asta.org/about/content.cfm?ItemNumber=1985&navItemNumber=515

[23] Biometrics. (1998). Retrieved from http://searchsecurity.techtarget.com/definition/biometrics

[24] Pfitzmann, B., & Sadeghi, A.-r. (1996). Anonymous fingerprinting. Berlin: Springer-Verlag

[25] Jain, A., et al. (1997). On-Line Fingerprint Verification. IEEE Transactions on Pattern Analysis and Machine Intelligence VOL. 19, No. 4, 302-305.

[26] O'Gorman, L. (1999). Fingerprint Verification. New Jersey: Verdicom Inc

[27] .William, S. (2005). Cryptography and Network Security Principles and Practices, Fourth Edition. Prentice Hall

[28] Bellifemine, F., et al. (2007). Developing multi-agent systems with Jade. John Wiley & Sons, Ltd

[29] Android. (2011). Retrieved from http://developer.android.com/guide/basics/what-is-android.html

[30] .Moreno et al. (2003). Using JADE-LEAP to implement agents in mobile devices. Retrieved from http://jade.tilab.com/papers/Exp/02Moreno.pdf.

[31] Morris, J., (2011). Android User Interface Development. 32 Lincoln Road, Olton, Birmingham, UK: Packt Publishing Ltd.

AUTHORS PROFILE

Floyd Garvey is a final year M.Sc. Computer Science student in the Department of Computing at the University of the West Indies, Jamaica since 2009 and also a Bachelor's degree in Management Information system. His e also a Microsoft certified professional. He started his professional career during 2003 as System/Network Administrator in Smith & Steward Distributions Ltd. Later during 2005-2008 he was working as IT administrator in Jamaica in the Theological Seminary & Caribbean Graduate School of Theology. And the during 2008-2011 he was working as System Support office in Bank of Novo Scotia Jamaica and presently working as

Senior System Support Analyst in Scotia Bank, Ontario, Canada. In his Master's Programme he did Master's thesis on Secure Agent based Airline Search and Booking system which focused on using intelligent agents to search, the airlines based on user requirements and also booking the flights online securely suing biometrics. His research interest includes Intelligent Agents, Mobile commerce, Robotics, Nanotechnology and Neural Networks.

Dr. Suresh Sankaranarayanan holds a PhD degree (2006) in Electrical in Networking from the University of South Australia. Later he has worked as a Postdoctoral Research Fellow and then as a Lecturer in the University of Technology, Sydney and at the University of Sydney, respectively during 2006-08. He is a Senior Member of IEEE computer Society and Computer Society of India. He was working as a Lecturer (Asst. Prof. Status) in the Department of Computing and lead the Intelligent Networking Research Group, in the University of West Indies, Kingston, Jamaica, during 2008-11.He has also worked as a Professor, School of Computer Science and Engineering, Vellore Institute of Technology (VIT University), Chennai Campus, India, for a short period during 2011. He is now working as Associate Professor, Department of Computer & Information Systems, Institute of Technology, Brunei (ITB – A technological university). Currently he is also functioning as a Visiting Professor, Department of computing, Faculty of Pure & applied Science**,** University of West Indies, Mona Campus**,** Kingston-7, Jamaica, West Indies. He has supervised more than 25 research students leading to M.Sc, ME, M.Phil and M.S degrees and currently supervising 6 students leading to M.Sc, M.Phil and PhD respectively. He has got to his credit, as on date, about 50 fully refereed research papers published in the Proceedings of major IEEE international conferences, as Book Chapters and in International Journals.

Imputation And Classification Of Missing Data Using Least Square Support Vector Machines – A New Approach In Dementia Diagnosis

T.R.Sivapriya
Dept. of Computer Science Lady Doak College
Madurai , India

A.R.Nadira Banu Kamal
Dept. of MCA
TBAK College
Kilakarai, India

V.Thavavel
Dept. of MCA
Karunya University
Coimbatore, India

***Abstract*— This paper presents a comparison of different data imputation approaches used in filling missing data and proposes a combined approach to estimate accurately missing attribute values in a patient database. The present study suggests a more robust technique that is likely to supply a value closer to the one that is missing for effective classification and diagnosis. Initially data is clustered and z-score method is used to select possible values of an instance with missing attribute values. Then multiple imputation method using LSSVM (Least Squares Support Vector Machine) is applied to select the most appropriate values for the missing attributes. Five imputed datasets have been used to demonstrate the performance of the proposed method. Experimental results show that our method outperforms conventional methods of multiple imputation and mean substitution. Moreover, the proposed method CZLSSVM (Clustered Z-score Least Square Support Vector Machine) has been evaluated in two classification problems for incomplete data. The efficacy of the imputation methods have been evaluated using LSSVM classifier. Experimental results indicate that accuracy of the classification is increases with CZLSSVM in the case of missing attribute value estimation. It is found that CZLSSVM outperforms other data imputation approaches like decision tree, rough sets and artificial neural networks, K-NN (K-Nearest Neighbour) and SVM. Further it is observed that CZLSSVM yields 95 per cent accuracy and prediction capability than other methods included and tested in the study.**

Keywords- ***Lease Square Support Vector Machine; z-score; Classification; KNN; Support Vector Machine.***

I. Introduction

Knowledge mining in databases especially medical databases of patient details consists of several steps like understanding the disease domain, forming the correct data set and cleaning the data, extracting of disease regularities hidden in the data thus formulating knowledge in the form of patterns or models, evaluation of the correctness and usefulness of results. Availability of large collections of medical data provides a valuable resource from which potentially new and useful knowledge can be discovered through data mining. Data Mining is increasingly popular as it holds to gain insight into the relationships and patterns hidden in the data. Patient records collected for diagnosis and prognosis typically encompass values of clinical and laboratory parameters and results of particular investigations specific to the disease domain. Such data are not usually complete and inadequate due to inappropriate selection of parameters for the given task.

Development of Data Mining tools for medical diagnosis and prediction is an utmost of the hour. Patient database often has measurements of a set of parameters at different times, requesting temporal component to be taken into account in data analysis. In this study, patients have been under a longitudinal and cross-sectional monitoring to record data through various modalities like neuropsychological testing and Magnetic Resonance Imaging.

Researchers usually address missing data by including in analysis only complete cases i.e. those individuals who have no missing data in any of the variables required for that analysis. However, results of such analyses could be biased. Furthermore, cumulative effect of missing data in several variables often leads to exclusion of a substantial proportion of the original sample, which in turn causes a substantial loss of precision and power leading to wrong diagnosis and treatment.

The risk of biased inclusion due to missing data depends on the reasons why data are missing. Reasons for missing data are commonly classified as: missing completely at random (MCAR), missing at random (MAR), and missing not at random (MNAR). If it is plausible that data are missing at random, and not completely at random, analyses based on complete cases could be biased and such biases could be overcome using multiple imputation methods that allow individuals with incomplete data to be included in the analyses. Unfortunately, often it is not possible to distinguish between missing values at random and missing not at random in observed data. Therefore, biases caused by data set that are missing not at random can be addressed only by sensitivity analyses to examine the effect of different assumptions on missing data mechanism.

II. Related Work

Several Statistical and data mining methods have been used to analyse diagnosis of dementia. There are two traditional missing value imputation techniques. They are parametric and non-parametric imputation strategies. Parametric method is applied when relationship between conditional attributes is known. Non- parametric method is applied when the relationship between the conditional attributes is unknown. Parametric methods like Nearest Neighbour [4][10][25] have been used for the prediction of missing attribute(s). Non-parametric technique such as empirical likelihood [32], clustering [26], Semi-parametric techniques [21][33] have also been applied for missing data imputation. Techniques like mixture model clustering [9], machine learning [12] have been used for imputing missing data. Multiple imputations [22] provide another way of finding missing values of attribute(s). In case of regression models, parametric regression imputation performs better if a dataset could be adequately and accurately modeled parametrically, or if users could correctly specify parametric forms for the dataset. Non-parametric imputation algorithm is found to be very effective when the user is unaware of the distribution of the dataset. Neural network method is regarded as one of non-parametric techniques used to compensate for missing values in sample surveys [24].A non-parametric algorithm is useful only when form of relationship between conditional attributes and target attribute is not known apriori.

For imputation in medical databases, Jose et.al [11] have concluded that the methods based on machine learning techniques have been found to be suited for imputation of missing values and led to a significant enhancement of prognosis accuracy compared to imputation methods based on statistical estimation. In another approach, STATA v.10 [1] is used to impute missing data in patient database, lowest scores[8] of MMSE were used to fill missing values in diagnosis of dementia.

Several algorithms have been proposed as a solution for diagnosis of dementia. Kloppel et al. developed a supervised method using a support vector machine (SVM) in a high dimensional space [14], Trosset et al. proposed another semi-supervised learning method, which used multidimensional scaling (MDS) [30], Ceyhan et al. analyze the shape and size of hippocampus, where prominent neuropathological markers are shown to be present in AD [3].In our previous study [27][28], we have investigated classification of dementia patients using SVM and an automatic supervised classification approach based on image texture analysis with Gabor wavelets as input to SVM, LS-SVM for distinguishing demented and non-demented patients.

This paper also evaluates approaches used to fill missing values and proposes a new and better approach to handle missing value situation and thereby enabling to feed correct input to the LSSVM classifier to get better prediction, diagnosis and treatment of the given data. The present study also examines the multiple biomarkers that contribute to dementia rather than concentrating on a single volume factor as described in the above studies through LS-SVM-PSO.

III. MISSING DATA HANDLING MECHANISMS

Several methods have been applied in data mining to handle missing values in database. Data with missing values could be ignored, or a global constant could be used to fill missing values (unknown, not applicable, infinity), such as attribute mean, attribute mean of the same class, or an algorithm could be applied to find missing values[34]. Missing data imputation technique means a strategy to fill missing values of a data set in order to apply standard methods which require completed data set for analysis. These techniques retain data in incomplete cases, as well as impute values of correlated variables.

Missing data imputation techniques are classified as ignorable missing data imputation methods, which include single imputation methods and multiple imputation methods, and non-ignorable missing data imputation methods which include likelihood based methods and the non-likelihood based methods. A single imputation method could fill one value for each missing value and it is more commonly used at present than multiple imputations which replace each missing value with several plausible values and better reflects sampling variability about actual value.

IV. DATA SETS

OASIS provides brain imaging data that are freely available and used for distribution and data analysis [17]. This data set consists of a cross-sectional collection of 416 subjects covering adults in the age group 18 to 96 with early-stage Alzheimer’s Disease (AD) . For each subject, 3 or 4 individual T1-weighted MRI scans taken during a single imaging session are available.

The basic data source for the present studies is obtained from Alzheimer's Disease Neuroimaging Initiative (ADNI), a clinic-based, multicenter, which provides longitudinal study with blood, CSF, PET, and MRI scans repeatedly measured in 229 participants with normal cognition (NC), 397 with mild cognitive impairment (MCI), and 193 with mild AD during 2005-2007.

V. IMPUTATION STRATEGIES

A. K-Nearest Neighbors (KNN) Imputation

If a training example contains one or more missing values, the *distance* between the example with missing values and all other examples is measured. Distance metric is a modified version of the *Manhattan distance* – distance between two examples is sum of the distances between the corresponding attribute values in each example. For discrete attributes, this distance is 0 if the values are the same, and 1 otherwise. In order to combine distances for discrete and continuous attributes, we perform a similar distance measurement for continuous attributes is performed– if the absolute difference between the two values is less than half of standard deviation, the distance is treated as 0; otherwise, 1.

The K complete examples closest to the example with missing values are used to choose a value. For a discrete attribute, the most frequently occurring value is used. For a continuous attribute, the average of the values from the K neighbors is used. In this study K value is determined by MMSE (Mini Mental State Examination) attribute distribution and set as 4 and 5 for demented and Non-Demented sets respectively.

B. Decision Tree

Decision tree is a classifier expressed as a recursive partition of the instance space. Decision trees are self-explanatory. They can handle both nominal and numeric input attributes and can handle datasets that may have errors and missing values. C4.5 is an evolution of ID3[20]. It uses gain ratio as splitting criteria. Splitting ceases when the number of instances to be split is below a certain threshold. Error-based pruning is performed after the growing phase. C4.5 can handle numeric attributes. C4.5's distribution-based imputation (DBI)[19], is used in this study. MMSE score is the splitting criterion based on which patient details are classified. Further CDR(Clinical Dementia Rating)[17] is an essential attribute in dementia diagnosis.

C. Back propagation algorithm

In our study a multilayered back-propagation neural network has been used (10 inputs from each of the 150 adolescents of the longitudinal and cross-sectional data set, comprising input patterns, and two binary outputs). The network was exposed to data, and parameters (weights and biases) have been adjusted to minimize error, using back-propagation training algorithm. The Input layer has 7 neurons, where each neuron represents reduced patient group. The number of neurons in the hidden layer is calculated based on the following equation :

$$N3 = ((2/3)*(N1))+N2$$

N1 represents number of nodes in the input layer; N2 represents number of nodes in the output layer; N3 represents number of nodes in the hidden layer.

D. Support Vector Machines

SVM is a classification technique originated from statistical learning theory [5][31] . Depending on the chosen kernel, SVM selects a set of data examples (support vectors) that define the decision boundary between classes. SVM is known for excellent classification performance, though it is arguable whether support vectors could be effectively used in communication of medical knowledge to domain experts.

Standard formulation of support vector machines (SVMs) fails if data has missing values for any of the attributes. The present study examines methods by which data sets containing missing values can be processed using an SVM. This is typically accomplished by one of the two means namely ignoring missing data (either by discarding examples with a missing attribute value or discarding an attribute that has missing values), or using a process generally referred to as imputation through, by which a value is generated for the attribute. These techniques are typically carried out on data set prior to its being supplied to learning algorithm.

First SVM is trained to use all training examples that have no missing values[16]. Then ignoring original classification value from the data set, value of the attribute imputed is utilised as target value. It is to be noted that any other attribute that has missing value is ignored while generating this new training data set.

E. LS-SVM

Least squares support vector machine (LS-SVM) [29] is a least squares version of support vector machine (SVM). In this technique estimated value of the missing value is obtained by solving a convex quadratic programming (QP) for classical SVMs. Least Squares SVMs (LS-SVMs) classifiers, in Suykens and Vandewalle. LS-SVM is a class of kernel based learning methods. Primary goals of the LS-SVM models are regression and classification.

If the attribute has continuous values, LSSVM in regression mode is applied to study the data. If the attribute is discrete with only two values, standard LSSVM in classification mode is used. For discrete attribute with more than two values, special handling is required with the standard LSSVM technique of one-against- all. After an LSSVM is trained on each data set[18], then that model is ut9lised to classify or perform regression on examples of that attribute with missing values. If more than one LSSVM model generates a positive classification, selection is made on the basis of accuracy and sensitivity of the classifier.

F. CZLSSVM imputation

In this study of automatic classification of dementia, filling missing values[12] is done through a combined approach to overcome overfitting of data.Several methods have reported in literature along with their own advantages and disadvantages. Our proposed method is a trial to give the best fit mechanism for filling in missing values in a patient database especially when data is collected over a period of time of several years along with several visits(a pool of cross section and time series data).

Data is clustered in two groups namely AD[15] and CN(Cognitively Normal). Z-score of the attribute MMSE is computed for each cluster in AD and CN. K-means clustering is an efficient algorithm applied in processing very large databases[6]. In a k-means cluster [2][7] constructed using similarity measure of MMSE, a missing value could be imputed based on (a) mean value of the corresponding attribute in other items contained in this cluster, or (b) similarity to nearest instance with a non-missing value (c) z-score of values in the cluster.

Steps :

1. Cluster the data sets based on MMSE in AD and CN groups using k-means algorithm
2. Find the mean and standard deviation for each cluster
3. Compute z-score for each cluster in each group

$$v' = \frac{v - \mu_A}{\sigma_A}$$

Where v' is the estimate of the missing value to be computed , v is observed value, μ is mean and σ is the standard deviation of the cluster respectively.

4. Generate datasets with multiple imputation
5. Train LS-SVM with imputed values and check for classification accuracy
6. Evaluate the imputation strategy based on accuracy and sensitivity yielded by the classifier

Muliple imputation is done by LSSVM which is trained with various z-score values computed for each value of MMSE belonging to the demented group. Similarly , same procedure is repeated to find multiple values for missing attirbute in non-demented group.

VI. LSSVM – PSO CLASSIFIER

In standard SVMs and its reformulations, LS-SVM, regularization parameter and kernel parameters are called hyper-parameters, which play a crucial role to the performance of the SVMs. There exist different techniques for tuning the hyper-parameters related to regularization constant and parameter of kernel function. PSO (Particle Swarm Optimisation) is an evolutionary computation technique based on swarm intelligence [13]. It has many advantages over other heuristic techniques. This technique has an edge over distributed and parallel computing capabilities, escapes local optima and enables quick convergence.

LSSVM-PSO is trained and tested with multiple imputation values [23] in 5 different data sets A, B, C, D, E constructed from the existing data from ADNI and OASIS database. Four different models of the classifier are designed by varying the number of particles in PSO search that improves the quick convergence and classification. Models I, II , III and IV are evaluated based on their Sensitivity, Specificity and Accuracy to find the best-fit for the diagnosis of dementia.

A. Optimization of LSSVM Parameters

In the case of LS-SVM with radial kernel function , optimized parameters are: γ, which is the weight at which testing errors are treated in relation to separation margin and parameter σ, which corresponds to width of the kernel function. It is unknown in advance what combination of these two parameters will achieve the best result of classification. In order to find the best values several techniques like Grid-Search, K-fold Cross-Validation, Particle Swarm Optimization have been in use. PSO provides better optimization than Grid-Search and K-fold method.

VII. RESULTS

All classification results could have an error rate and on occasion will either fail to identify dementia or misclassify a normal patient as demented. It is common to describe this error rate by the terms true positive and false positive and true negative and false negative as follows:

True Positive (TP): the classification result is positive in the presence of the clinical abnormality.

True Negative (TN): the classification result is negative in the absence of the clinical abnormality.

False Positive (FP): the classification result is positive in the absence of the clinical abnormality.

False Negative (FN): the classification result is negative in the presence of the clinical abnormality.

Sensitivity = TP/ (TP+FN) *100%

Specificity = TN/ (TN+FP) *100%

Accuracy = (TP+TN)/ (TP+TN+FP+FN)*100 %

TP, TN, FP, FN, Sensitivity, Specificity and Accuracy are used to measure the performance of the classifiers. Experiments were carried out in MATLAB.

Imputation methods based on CZLSSVM-PSO method outperformed other imputation methods in the prediction of Dementia. Sensitivity and sensitivity analysis revealed a significant difference in percentage, error rate evaluation showed that the rate of error detected for CZLSSVM is significantly lower than KNN, BPN, C4.5 and SVM methods. Table 1 indicates the average error rate of imputation methods. Table 2 and 3 illustrate the accuracy of LSSVM-PSO classifier yielded by various imputation strategies in OASIS and ADNI databases respectively.

Table 4 and 5 depict that the overall performance of LSSVM-PSO classifier is high with the input of data imputed by the proposed CZLSSVM method compared to other methods. Out of the 4 models tested for classification as illustrated in Figure 1., Model 3 of LSSVM-PSO classifier is found to be very effective when combined with CZLSSVM method.

Validation

A neural network model with 10 X 7 X 1 structure has been used in the present study to perform classification by setting aside 20% of the patterns (or observations) as validation (or testing) data. In this cross-validation approach, training is done repeatedly exposing the network to the remaining 80% of the patterns (training data) for several epochs, where an epoch is one complete cycle through the network for all cases. Data has been normalized before training. A network trained in this manner is considered generalizable, in the sense that it can be used to make estimate.

TABLE I. COMPARISON OF ERROR RATE OF IMPUTATION METHODS

Imputation Methods	**Average Error rate interval (OASIS)**	**Average Error rate interval (ADNI)**
k-nn	2.7±0.22	2.2±0.22
BPN	1.5±0.03	1.2±0.13
C5.0	2.5±0.15	1.5±0.25
SVM	0.5±0.04	0.9±0.03
CZLSSVM	0.03±0.01	0.23±0.11

SOURCE : COMPUTED USING OASIS AND ADNI DATA

NOTE : POOL OF CROSS-SECTION AND TIME SERIES DATA IS USED

TABLE II. CLASSIFICATION ACCURACY OF LSSVM-PSO FOR MULTIPLE IMPUTATION IN 5 DATASETS A, B, C ,D, E SELECTED FROM OASIS DATABASE

IMPUTATION METHODS	DATA SETS				
	A	B	C	D	E
K-NN	75±0.4	78±0.7	80±0.2	79±0.33	76±0.08
BPN	89±0.3	85±0.5	87±0.32	86±0.56	88±0.04
C5.0	80±0.2	79±0.04	81±0.22	85±0.43	83±0.03
SVM	89±0.7	90±0.25	91±0.4	89±0.28	85±0.06
CZLSSVM	90±0.6	97±0.34	96±0.23	98±0.48	96±0.33

TABLE III. CLASSIFICATION ACCURACY OF LSSVM-PSO FOR MULTIPLE IMPUTATION IN 5 DATASETS A, B, C ,D, E SELECTED FROM ADNI DATABASE

IMPUTATION METHODS	DATA SETS				
	A	B	C	D	E
K-NN	77±0.03	80±0.54	73±0.06	74±0.01	76±0.03
BPN	85±0.21	87±0.43	83±0.04	88±0.03	89±0.01
C4.5	80±0.11	81±0.29	83±0.4	84±0.03	84±0.05
SVM	90±0.24	91±0.46	89±0.02	88±0.12	86±0.04
CZLSSVM	95±0.2	97±0.02	98±0.31	95±0.04	98±0.14

Table IV. Comparison of Efficiency of LSSVM-pso classifier for time series data set with multiple imputation strategies.

MEASURE	**k-means**	**SVM**	**CZLSSVM**	**BPN**	**C4.5**
Sensitivity %	89	92	95	89	88
Accuracy %	90	90	96	90	89
Specificity %	89	9 3	97	90	85

TABLE V. COMPARISON OF EFFICIENCY OF LSSVM-PSO CLASSIFIER FOR CROSS-SECTION DATA SET WITH MULTIPLE IMPUTATION STRATEGIES

MEASURE OF PERFORMANCE	**IMPUTATION METHODS**				
	k-means	**SVM**	**CZLSSVM**	**BPN**	**C4.5**
Sensitivity %	78	94	94	89	85
Accuracy %	79	94	96	88	86
Specificity %	80	97	99	90	77

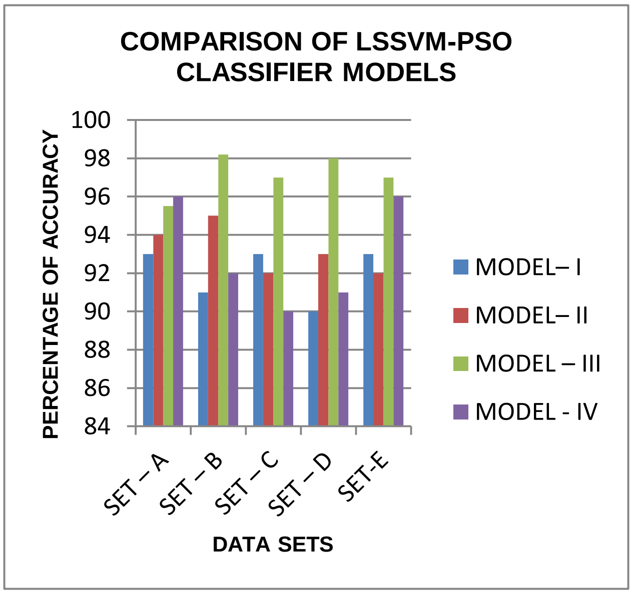


Fig. I. PERFORMANCE EVALUATION OF LSSVM-PSO CLASSIFIER MODELS

VIII. CONCLUSION

Methods based on multiple imputation coupled with z-score and support vector machine classifier is found to be the most suited technique for imputation of missing values and led to a significant enhancement of prognosis accuracy compared to imputation methods based on k-NN, BPN, C4.5 and SVM procedures. Classification accuracy of LSSVM-PSO is very high when the missing values imputed by CZLSSVM are given as input as opposed to other methods in the diagnosis of dementia.

ACKNOWLEDGEMENT

I would like to acknowledge the guidance, and expert opinion given by Dr. Sabesan, former Prof. and Head, Govt. Rajaji Hospital, Madurai and Dr.S.Vijayalakshmi M.Sc. Ph.D., former Professor and Head, School of Economics, Madurai Kamaraj University.

REFERENCES

[1] Anstey et al(2010). Estimates of probable dementia prevalence from population-based surveys compared with dementia prevalence estimates based on meta-analyses, BMC Neurology, 10:62.

[2] Bankat M. Patil et al (2010) Missing value Imputation based on K-Means with Weighted Distance, Part I, CCIS 94, 600-609, Springer-Verlag Berlin Heidelberg (2010)

[3] Ceyhan, E., Ceritoglu, C. et al. (2008)Analysis of metric distances and volumes of hippocampi indicates different morphometric changes over time in dementia of alzheimer type and nondemented subjects. *Technical Report*, Department of Mathematics, Koc University, Istanbull, Turkey,.

[4] Chi-Chun, H. and Hahn-Ming, L (2004) A Grey-Based Nearest Neighbor Approach for Missing Attribute Value Prediction. Journal of Artificial Intelligence **20**,239-252.

[5] Cristianini, N., Shawe-Taylor, J. (2000) An introduction to Support Vector Machines, Cambridge University Press, Cambridge.

[6] Dan Li, Jitender Deogun, William Spaulding and Bill Shuart (2004) Towards Missing Data Imputation: A Study of Fuzzy K-means Clustering Method, ROUGH SETS AND CURRENT TRENDS IN COMPUTING , Lecture Notes in Computer Science, **3066/2004**, , 573-579, DOI: 10.1007/978-3-540-25929-9_70.

[7] Fujikawa, Y., Ho, T.B. (2002) Cluster-based algorithms for dealing with missing values. In: Knowledge Discovery and Data Mining Conference, pp. 549--554. Springer, Berlin.

[8] Harriet M M Smeding, Inge de Koning (2000), Frontotemporal dementia and neuropsychology:the value of missing values, Journal of Neurol Neurosurg Psychiatry **68**, 726–730.

[9] Hunt, L., Jorgensen, M. (2003) Mixture model clustering for mixed data with missing information. Comput. Statist.Data Anal. 41, 193–210.

[10] ItoWasito, Boris Mirkin (2006) Nearest neighbours in least-squares data imputation algorithms with different missing patterns, Computational Statistics & Data Analysis 50, 926 – 949, doi:10.1016/j.csda.2004.11.009

[11]Jose M. Jerez, Ignacio Molina, Pedro J. García-Laencina, Emilio Alba, Nuria Ribelles, Miguel Martín, Leonardo Franco (2010) Missing data imputation using statistical and machine learning methods in a real breast cancer problem , Artificial Intelligence in Medicine 50.

[12]Kamakashi, L., Harp, S.A., Samad, T., Goldman, R.P. (1996) Imputation of missing data using machine learning techniques. In: Simoudis, E., Han, J., Fayyad, U. (Eds.), Second International Conference on Knowledge, Discovery and Data Mining. Oregon, 140–145.

[13]Keneddy J. and Eberhart R. C. (1995) Particle swarm optimization, in Proc. IEEE Int. Conf. Neural Networks,, 1942–1948.

[14]Kloppel, S., et al. (2008) Accuracy of dementia diagnosis -a direct comparison between radiologists and a computerized method, Brain, 131, 2969 -2974.

[15]Little RJA.. (1995) Modeling the drop-out mechanism in longitudinal studies. J. Am. Statist. Assoc. 90: 1112–21

[16]Mallinson, H. & Gammerman, A. (2003) Imputation Using Support Vector Machines http://www.cs.york.ac.uk/euredit/

[17]Marcus, DS., Wang, TH., Parker J.M., Csernansky JG., Morris, JC., Buckner, RL. (2007) Open Access Series of Imaging Studies (OASIS): Cross-Sectional MRI Data in Young, Middle Aged, Nondemented and Demented Older Adults. *Journal of Cognitive Neuroscience*, **19** 1498-1507.

[18] Maytal Saar, Foster Provost (2007) Handling Missing Values when Applying Classification Models Journal of Machine Learning Research 8 1625-1657

[19] Qinbao Song ,Martin Shepperd, Xiangru Chen, Jun Liu (2008) Can k-NN imputation improve the performance of C4.5 with small software project data sets? A comparative evaluation The Journal of Systems and Software 81 2361–2370.

[20] Quinlan, J. R.: C4.5(1993) Programs for machine learning, Morgan Kaufmann, San Mateo, CA.

[21][21] Robins, J. M., Rotnizky, A. & Zhao, L. P. (1995) Analysis of semiparametric regression models for repeated outcomes in the presence of missing data. *Journal of the American Statistical Association*, 90,,106–121.

[22] Rubin D.B (1987) Multiple imputation for nonresponse in surveys.. John Wiley and Sons.

[23]Saar-Tsechansky M. and Provost F. (2007) Handling missing values when applying classification models. Journal of Machine Learning Research, 8:, 1625–1657.

[24] Setiawan N.A, Venkatachalam P.A., Hani A.F.M. (2008) Missing Attribute Value Prediction Based on Artificial Neural Network and Rough Set Theory, ISBN: 978-0-7695-3118-2, International Conference on BioMedical Engineering and Informatics, DOI Bookmark: http://doi.ieeecomputersociety.org/10.1109/BMEI.2008.322.

[25] Shichao Zhang (2011) Shell-neighbor method and its application in missing data imputation Journal of Artificial Intelligence 35, 123-133.

[26] Schichao Zhang , Jilian Zhang, et al (2008) Missing value imputation based on data clustering, Transactions on Computer Science 128-138.

[27] Sivapriya T.R., Saravanan,V. and Ranjit Jeba Thangaiah, P. (2011) Texture Analysis of Brain MRI and Classification with BPN for the Diagnosis of Dementia, Communications in Computer and Information Science, Volume 204, Trends in Computer Science, Engineering and Information Technology, Part 1, 553-563.

[28] Sivapriya, T.R., Saravanan, V. (2011) Dementia Diagnosis relying on Texture based features and SVM classification, ICGST-AIML Journal, 11, 9-19.

[29]Suykens, J. A. K., Van Gestel, T., De Brabanter, J., De Moor, B. , Vandewalle, J. (2002) Least Squares Support Vector Machines, World Scientific Publishing Company,.

[30]Trosset,M., C. Priebe, Y. Park, and M. Miller. (2007) Semisupervised learning from dissimilarity data. *Technical Report* Department of Statistics, Indiana University, Bloomington, IN4705.

[31]Vapnik, V. N. (1995) The Nature of Statistical Learning Theory. Springer-Verlag, New York,

[32] Wang, Q. and Rao, J. N. K. (2002) Empirical likelihood-based inference under imputation for missing response data. Ann. Statist., 30 896-924.

[33] Yongsong Qin, Shichao Zhang, Xiaofeng Zhu, Jilian Zhang and Chengqi Zhang (2007) Semi-parametric optimization for missing data imputation, Journal of Artificial Intelligence 27, 79-88.

[34] Zhang, C.Q., et al. (2007) An Imputation Method for Missing Values. PAKDD, LNAI, 4426, 1080-1087.

A Proposed Hybrid Technique for Recognizing Arabic Characters

S.F. Bahgat, S.Ghomiemy
Computer Eng. Dept.
College of Computers and Information Technology
Taif University
Taif, Saudi Arabia

S. Aljahdali, M. Alotaibi
Computer Science Dept.
College of Computers and Information Technology
Taif University
Taif, Saudi Arabia

***Abstract*— Optical character recognition systems improve human-machine interaction and are urgently required for many governmental and commercial departments. A considerable progress in the recognition techniques of Latin and Chinese characters has been achieved. By contrast, Arabic Optical Character Recognition (AOCR) is still lagging although the interest and research in this area is becoming more intensive than before. This is because the Arabic is a cursive language, written from right to left, each character has two to four different forms according to its position in the word, and most characters are associated with complementary parts above, below, or inside the character. The process of Arabic character recognition passes through several stages; the most serious and error-prone of which are segmentation, and feature extraction & classification. This research focuses on the feature extraction and classification stage, being as important as the segmentation stage. Features can be classified into two categories; Local features, which are usually geometric, and Global features, which are either topological or statistical. Four approaches related to the statistical category are to be investigated, namely: Moment Invariants, Gray Level Co-occurrence Matrix, Run Length Matrix, and Statistical Properties of Intensity Histogram. The paper aims at fusing the features of these methods to get the most representative feature vector that maximizes the recognition rate.**

***Keywords-** Optical Character Recognition; Feature Extraction; Dimensionality Reduction; Principal Component Analysis; Feature Fusion.*

I. INTRODUCTION

OCR is the process of converting a raster image representation of a document into a format that a computer can process. Thus, it may involve many sub-disciplines of computer science including image processing, pattern recognition, artificial intelligence, and database systems. Despite intensive investigation, the ultimate goal of developing an optical character recognition (OCR) system with the same reading capabilities as humans still remains unachieved and more so in the case of Arabic language. Most commercially available OCR products are for typed English text because English text characters do not have all the extra complexities associated with Arabic letters.

Arabic is a popular script. It is estimated that there are more than one billion Arabic script users in the world. If OCR systems are available for Arabic characters, they will have a great commercial value. However, due to the cursive nature of Arabic script, the development of Arabic OCR systems involves many technical problems, especially in the segmentation and feature extraction & classification stages. Most characters have dot(s), zigzag(s), madda, etc, associated with the character and this can be above, below, or inside the character. Many characters have a similar shape, the position or number of secondary strokes and dots makes the only difference. Although many researchers are investigating solutions to solve the problems, little progress has been made.

Feature extraction is one of the important basic steps of pattern recognition. Features should contain information required to distinguish between classes, be insensitive to irrelevant variability in the input, and also be limited in number to permit efficient computation of discriminant functions and to limit the amount of training data required. In fact, this step involves measuring those features of the input character that are relevant to classification. After feature extraction, the character is represented by the set of extracted features.

Features can be classified into two categories: Local features, which are usually geometric (e.g. concave/convex parts, number of endpoints, branches, joints, etc), and Global features, which are either topological (connectivity, projection profiles, number of holes, etc) or statistical.

The objective of this paper is to examine the performance of four of these global statistical features; namely: Moments Invariants (MIs), Gray Level Co-occurrence Matrix (GLCM), Run Length Matrix (RLM), and Statistical Properties of Intensity Histogram (SFIH), and to study the effect of fusing two or more of these features on the recognition rate.

The rest of the paper is organized as follows. Section II summarizes the related work. Section III introduces the proposed approach. Results and discussion are presented in Section IV. The paper is terminated by concluding remarks and proposals for future work.

II. RELATED WORK

The features extraction stage, playing the main role in the recognition process, controls the accuracy of recognition by the information passed from this stage to the classifier (recognizer). These information can be structural features such as loops, branch-points, endpoints, and dots; or statistical which includes, but is not limited to, pixel densities, histograms of chain code directions, moments, and Fourier descriptors. Because of the importance of this stage many approaches and techniques have been proposed.

In [1], two methods for script identification based on texture analysis have been implemented: Gabor filters and GLCMs. In tests conducted on exactly the same sets of data, the Gabor filters proved to be far more accurate than the GLCMs, producing results which are over 95% accurate.

[2] presented a new technique for feature extraction based on hybrid spectral-statistical measures (SSMs) of texture. They studied its effectiveness compared with multiple-channel (Gabor) filters and GLCM, which are well-known techniques yielding a high performance in writer identification in Roman handwriting. Texture features were extracted for wide range of frequency and orientation because of the nature of the spread of Arabic handwriting compared with Roman handwriting. The most discriminant features were selected with a model for feature selection using hybrid support vector machine-genetic algorithm techniques. Experiments were performed using Arabic handwriting samples from 20 different people and very promising results of 90.0% correct identification were achieved.

In [3], a novel feature extraction approach of handwritten Arabic letters is proposed. Pre-segmented letters were first partitioned into main body and secondary components. Then moment features were extracted from the whole letter as well as from the main body and the secondary components. Using multi-objective genetic algorithm, efficient feature subsets were selected. Finally, various feature subsets were evaluated according to their classification error using an SVM classifier. The proposed approach improved the classification error in all cases studied. For example, the improvements of 20-feature subsets of normalized central moments and Zernike moments were 15 and 10%, respectively. This approach can be combined with other feature extraction techniques to achieve high recognition accuracy.

In [4], a new set of run-length texture features that significantly improve image classification accuracy over traditional run-length features were extracted. By directly using part or all of the run-length matrix as a feature vector, much of the texture information is preserved. This approach is made possible by the utilization of the multilevel dominant eigenvector estimation method, which reduces the computation complexity of KLT by several orders of magnitude. Combined with the Bhattacharyya measure [5], they form an efficient feature selection algorithm. The advantage of this approach is demonstrated experimentally by the classification of two independent texture data sets. Experimentally, they observed that most texture information is stored in the first few columns of the RLM, especially in the first column. This observation justifies development of a new, fast, parallel RLM computation scheme. Comparisons of this new approach with the co-occurrence and wavelet features demonstrate that the RLMs possess as much discriminatory information as these successful conventional texture features and that a good method of extracting such information is key to the success of the classification.

In [6], Zernike and Legendre Moments for Arabic letter recognition have been investigated. Experiments demonstrated both methods' effectiveness in extracting and preserving Arabic letter characteristics. ZM is used due to its ability to compute the complex orthogonal moments precisely. The system has achieved satisfactory performance when compared with other OCR systems. The translational and scaling invariant, on the other hand, had struggled in LM to detect rotational invariant forms in the experiments. The objective for maximising the correct matching and retrieval from the Arabic database while minimising the false positive rate has been achieved.

[7] explores a design-based method to fuse Gabor filter features and co-occurrence probability features for improved texture recognition. The fused feature set utilizes both the Gabor filter's capability of accurately capturing lower frequency texture information and the co-occurrence probability's capability in texture information relevant to higher frequency components. Fisher linear discriminant analysis indicates that the fused features have much higher feature space separation than the pure features. Overall, the fused features are a definite improvement over non-fused features and are advocated in texture analysis applications.

III. Proposed Approach

Substantial research efforts have been devoted during last years to AOCR and many approaches have been developed (structural, geometric, statistics, stochastic…). However, certain problems remain open and deserve more attention in order to achieve results equivalent to those obtained for other scripts such as Latin. Besides, other methods must be explored and various sources of information have also to be used [8].

The process of isolated Arabic optical character recognition comprises three main stages: Preprocessing, Feature extraction, and Classification. The structure of the proposed approach is shown in Figure 1.

The training dataset includes the 28 (100 x 100) *jpg* images of the isolated Arabic characters shown below:

خ	ح	ج	ث	ت	ب	أ
ق	ف	غ	ع	ظ	ط	ض
ص	ش	س	ز	ر	ذ	د
ي	و	هـ	ن	م	ل	ك

The test datasets include:

1) 3 datasets composed of the clean set corrupted by salt and pepper noise of intensity 1 %, 3 %, and 5 % respectively.

2) 3 datasets composed of the clean set corrupted by impulse noise of intensity 1 %, 3 %, and 5 % respectively.

3) 3 datasets composed of the clean set corrupted by Gaussian noise of intensity 1 %, 3%, and 5% respectively.

Figure 2 displays the letter " ش " as an example of the 10 datasets.

The procedure proceeds as follows:

1) *In the preprocessing phase, the noise removal is carried out using median filter, and the binarization is done with histogram thresholding.*

2) *In the feature extraction phase, four feature sets are calculated using MIs, GLCM, RLM, and SFIH, respectively. These initial feature vectors are used for evaluating the maximum possible recognition rate for the corrupted datasets, using each set of features. The relations used for calculating the features of the different techniques are discussed below.*

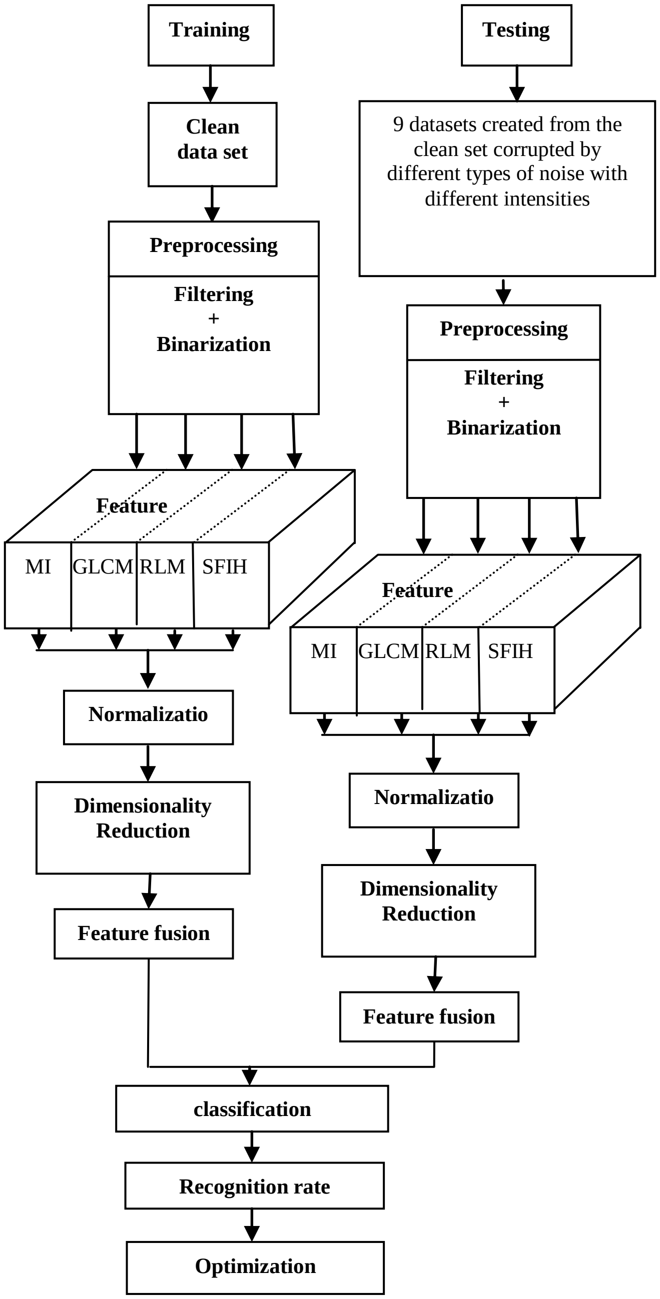


Figure 1. Proposed approach flow diagram

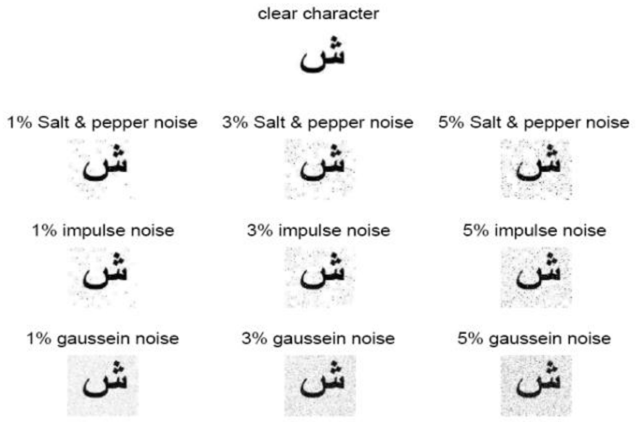


Figure 2. Sample Images of different datasets

A. *MIs Features:*

The regular moment of a shape in an M by N binary image is defined as:

$$u_{pq} = \sum_{j=0}^{N-1}\sum_{i=0}^{M-1} i^p j^q f(i,j) \qquad (1)$$

where *f(i,j)* is the intensity o

f the pixel (either 0 or 1) at the coordinate (i,*j*) and *p+q* is said to be the order of the moment. The coordinates of the centroid are determined using the relations:

$$i' = \frac{u_{10}}{u_{00}} \quad and \quad j' = \frac{u_{01}}{u_{00}} \qquad (2)$$

Relative moments are then calculated using the equation for central moments defined as:

$$u_{pq} = \sum_{j=0}^{N-1}\sum_{i=0}^{M-1} (i-i')^p (j-j')^q f(i,j) \qquad (3)$$

A set of seven rotational invariant moment functions which form a suitable shape representation were derived by Hu [9, 10, 11]. These equations, used throughout this work are shown in Appendix A_i .

B. *GLCM Features*

The GLCM is a tabulation of how often different combinations of pixel brightness values (grey levels) occur in an image [13]. The GLCM is used for a series of "second order" texture calculations. GLCM texture considers the relationship between *groups of two (usually neighboring) pixels* in the original image. at a time, called the reference and the neighbour pixel. The neighbour pixel is chosen to be the one to the east (right) of each reference pixel. This can also be expressed as a (1,0) relation: 1 pixel in the x direction, 0 pixels in the y direction. Each pixel within the window becomes the reference pixel in turn, starting in the upper left corner and proceeding to the lower right. Pixels along the right edge have no right hand neighbour, so they are not used for this count.

To create a GLCM, use the graycomatrix function. The graycomatrix function creates a gray-level co-occurrence matrix (GLCM) by calculating how often a pixel with the intensity (gray-level) value *i* occurs in a specific spatial relationship to a pixel with the value *j*. By default, the spatial relationship is defined as the pixel of interest and the pixel to its immediate right (horizontally adjacent), but you can specify other spatial relationships between the two pixels. Each element (*i*,*j*) in the resultant GLCM is simply the sum of the number of times that the pixel with value *i* occurred in the specified spatial relationship to a pixel with value *j* in the input image.

Because the processing required to calculate a GLCM for the full dynamic range of an image is prohibitive, graycomatrix scales the input image. By default, graycomatrix uses scaling to reduce the number of intensity values in grayscale image from 256 to eight. The number of gray levels determines the size of the GLCM. To control the number of gray levels in the GLCM and the scaling of intensity values, using the NumLevels and the Gray Limits parameters of the graycomatrix function. The GLCM can reveal certain properties about the spatial distribution of the gray levels in the texture image. For example, if most of the entries in the GLCM are concentrated along the diagonal, the texture is coarse with respect to the specified offset. You can also derive several statistical measures from the GLCM. The set of features extracted from the GLCM matrix [14] is shown in Appendix A_{ii}.

C. RLM Features

Run-length statistics capture the coarseness of a texture in specified directions. A run is defined as a string of consecutive pixels which have the same gray level intensity along a specific linear orientation. Fine textures tend to contain more short runs with similar gray level intensities, while coarse textures have more long runs with significantly different gray level intensities [15].

A run-length matrix *P* is defined as follows: each element *P(i, j)* represents the number of runs with pixels of gray level intensity equal to *i* and length of run equal to *j* along a specific orientation. The size of the matrix *P* is *n by k*, where *n* is the maximum gray level in the image and *k* is equal to the possible maximum run length in the corresponding image. An orientation is defined using a displacement vector *d(x, y)*, where *x* and *y* are the displacements for the *x-axis* and *y-axis*, respectively. The typical orientations are 0°, *45°*, *90°*, and *135°*, and calculating the run-length encoding for each direction will produce four run-length matrices.

Once the run-length matrices are calculated along each direction, several texture descriptors are calculated to capture the texture properties and differentiate among different textures [15]. The set of RLM features is shown in Appendix A_{iii}.

D. SFIH Features

A frequently used approach for texture analysis is based on statistical properties of intensity histogram. One such measures is based on statistical moments. The expression for the *n*th order moments about the mean is given by:

$$\mu_n = \sum_{i=0}^{L-1} (z_i - m)^n p(z_i) \qquad (4)$$

Where z_i is a random variable indicating intensity, $p(z_i)$ is the histogram of the intensity levels in the image, L is the number of possible intensity levels and

$$m = \sum_{i=0}^{L-1} z_i p(z_i) \qquad (5)$$

is the mean (average) intensity. The set of features following this approach is shown in Appendix A_{iv}.

Feature selection helps to reduce the feature space which improves the prediction accuracy and minimizes the computation time. This is achieved by removing irrelevant, redundant and noisy features, i.e., it selects the subset of features that can achieve the best performance in terms of accuracy and computation time. It performs the Dimensionality reduction. Principal Components Analysis (PCA) is a very popular technique for dimensionality reduction. Given a set of data on *n* dimensions, PCA aims to find a linear subspace of dimension *d* lower than *n* such that the data points lie mainly on this linear subspace. Such a reduced subspace attempts to maintain most of the variability of the data. Applying PCA for dimensionality reduction, we get the minimum number of features giving the maximum possible recognition rate obtained earlier using the full feature vector), for each procedure. Analyzing the effect of feature fusion by fusing the features of each two of the four procedures, and evaluating the resultant recognition rate.

Classification is the main decision stage of the OCR system in general. In this stage the features extracted from the primitive is compared to those of the model set. As the classification is generally implemented according to the criterion of minimizing the Euclidian distance between feature vectors, it is necessary to normalize the fused features. The normalization should comply with a rule that each feature component should be treated equally for its contribution to the distance. The rationale usually given for this rule is that it prevents certain features from dominating distance calculations merely because they have large numerical values. A linear stretch method can be used to normalize each feature component over the entire data set to be between zero and one. A feature selection procedure can be used after the feature vectors are fused. A weighting method called feature contrast, is employed to perform an unsupervised feature selection.

Denote the ith n-D fused feature vector as $F_i = \{f_{i,1}, f_{i,2}, \cdots, f_{i,n}\}$. The feature contrast of the j^{th} component of the feature vector is defined as:

$$\xi_j = \frac{\max_i(f_{i,j}) - \text{mean}_i(f_{i,j})}{\max_i(f_{i,j}) + \text{mean}_i(f_{i,j})} \qquad (6)$$

Then each feature component is weighted by its feature contrast divided by the maximum feature contrast of all feature components, that is,

$$F_i^* = \frac{1}{\max_j(\xi_j)} \{\xi_1 f_{i,1}, \{\xi_2 f_{i,2}, \ldots., \{\xi_n f_{i,n}\} \qquad (7)$$

A common strategy of feature fusion is first to combine various features and then perform feature selection to choose an optimal feature subset according to the feature data set itself, such as by principal component analysis (PCA).

As we are interested mainly in feature extraction, no great emphasis is paid for the classier. We will implement only the basic classifier; namely: Nearest-Neighbor Classifier, based on the Euclidean distance between a test sample and the specified training samples. Let xi be an input sample with p features (xi1,xi2,…,xip) , n be the total number of input samples (i=1,2,…,n) and p the total number of features (j= 1,2,…,*p*) . The Euclidean distance between sample $\mathbf{x}_i$ and $\mathbf{x}_l$ (l =1,2,…,n) is defined as:

$$d(x_i, x_l) = \sqrt{((x_{i1} - x_{l1})^2 + (x_{i2} - x_{l2})^2 + \cdots + (x_{ip} - x_{lp})^2)} \qquad (8)$$

IV. Results & Discussion

In the training phase, four sets of features are calculated for the clean dataset using the four methods under consideration (MIs, GLCM, RLM, and SFIH). The PCA algorithm is also applied in each case, and the corresponding feature vectors are stored for further processing.

In the testing phase, the same approach is followed for the data in the nine corrupted datasets. Using the full feature vectors, the recognition rate is determined for each method, and is labeled as the maximum possible recognition rate that can be achieved in this situation. As the feature vectors are sorted in a descending order as a result of applying PCA, we searched for the minimum number of features giving the maximum possible recognition rate for each method. According to Table 1, the maximum recognition rate was achieved using MIs, followed by GLCM, and RLM. The SFIH gave the least recognition rate. Figure 3, clarifies these results. The minimum number of features satisfying maximum recognition rate was found to be 2, 3, 4, and 1 for IMs, GLCM, RLM, and SFIH, respectively.

TABLE 1. Maximum possible recognition rate for the corrupted datasets,

Noise Type	MIs	GLCM	RLM	SFIH
Salt & Pepper	99.107	96.429	95.536	87.5
Gaussian Noise	98.214	95.536	91.071	80.357
Impulse Noise	99.107	96.429	91.071	87.5
Average	98.813	96.13	92.559	85.119

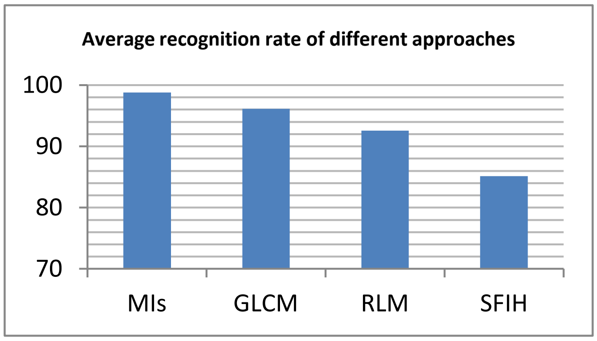


Figure 3. Average recognition rate of different approaches

The effect of the number of features of MIs on the recognition rate is shown in Table 2 for the different types of noise. On the average, the effect of the number of features of MIs on the recognition rate is shown in Figure 4.

TABLE 2. The relation between the number of features and the obtained recognition rate for MIs

Noise Type	2 Features	1 Feature
Salt & Pepper	99.107	73.214
Gaussian Noise	98.214	67.857
Impulse Noise	99.107	75.893
Average	98.8093333	72.321333

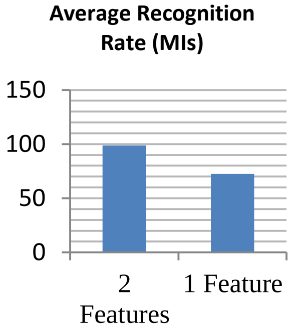


Figure 4. Average recognition rates of MIs as a function of the number of features

The effect of the number of features of GLCM on the recognition rate is shown in Table 3 for the different types of noise. On the average, the effect of the number of features of MIs on the recognition rate is shown in Figure 5.

TABLE 3. The relation between the number of features and the obtained recognition rate for GLCM

Noise Type	3 Features	2 Features	1 Feature
Salt & Pepper	96.429	96.429	91.071
Gaussian Noise	95.538	94.643	85.714
Impulse Noise	96.429	96.429	85.714
Average	96.132	95.833667	87.499667

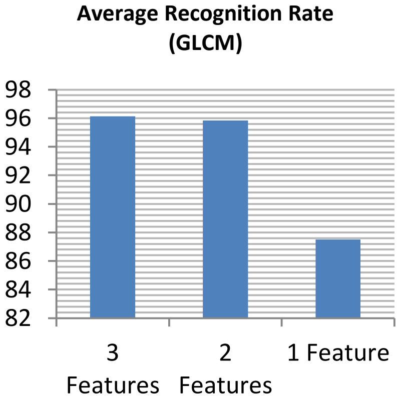


Figure 5. Average recognition rate of GLCM as a function of the number of features

The effect of the number of features of RLM on the recognition rate is shown in Table 4 for the different types of noise. On the average, the effect of the number of features of MIs on the recognition rate is shown in Figure 6.

TABLE 4. The relation between the number of features and the obtained recognition rate for RLM

Noise Type	4 Features	3 Features	2 Features	1 Feature
Salt & Pepper	95.536	91.071	75	75
Gaussian Noise	91.071	85.712	69.643	65.179
Impulse Noise	91.071	88.393	77.679	73.214
Average	92.55933	88.392	74.107333	71.131

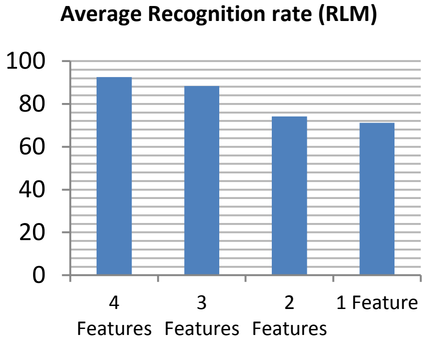


Figure 6. Average recognition rate of RLM as a function of the number of features

The effect of the number of features of SFIH on the recognition rate is shown in Table 5 for the different types of noise. On the average, the effect of the number of features of MIs on the recognition rate is shown in Figure 7.

TABLE 5. The relation between the number of features and the obtained recognition rate for SFIH

Noise Type	2 Features	1 Feature
Salt & Pepper Noise	87.5	87.5
Gaussian Noise	80.357	80.357
Impulse Noise	88.393	87.5
Average	85.4166667	85.119

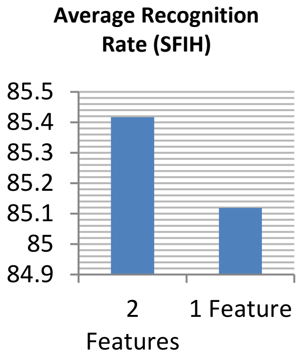


Figure 7. Average recognition rate of SFIH as a function of the number of features

As the main objective is to emphasize the effect of hybridization (feature fusion) on the enhancement of recognition rate, Tables 6, 7, and 8 illustrate the resultant recognition rate due to fusing features GLCM with RLM , MIs with GLCM, and GLCM with SFIH.

TABLE 6. The GLCM features with the RLM features

Recognition rate	Gaussian	impulse noise	salt & pepper	Average
GLCM with one feature (G_1)	85.714	85.714	91.071	87.5
RLM with 4 features (R_4)	91.0714	91.0714	95.535	92.559
GLCM and RLM (G_R)	95.5357	94.6428	97.321	95.833
GLCM with 2 features (G_2)	94.642	96.428	96.428	95.833
RLM with 4 features (R_4)	91.071	91.071	95.535	92.559
GLCM and RLM (G_R)	95.535	95.535	97.321	96.130
GLCM with one feature (G_1)	85.714	85.714	91.071	87.5
RLM with 3 features (R_3)	85.714	88.392	91.071	88.392
GLCM and RLM (G_R)	95.535	94.642	97.321	95.833

According to Figure 8, fusing 2 features of GLCM with 4 features of RLM, gives the best recognition rate (96.13 %).

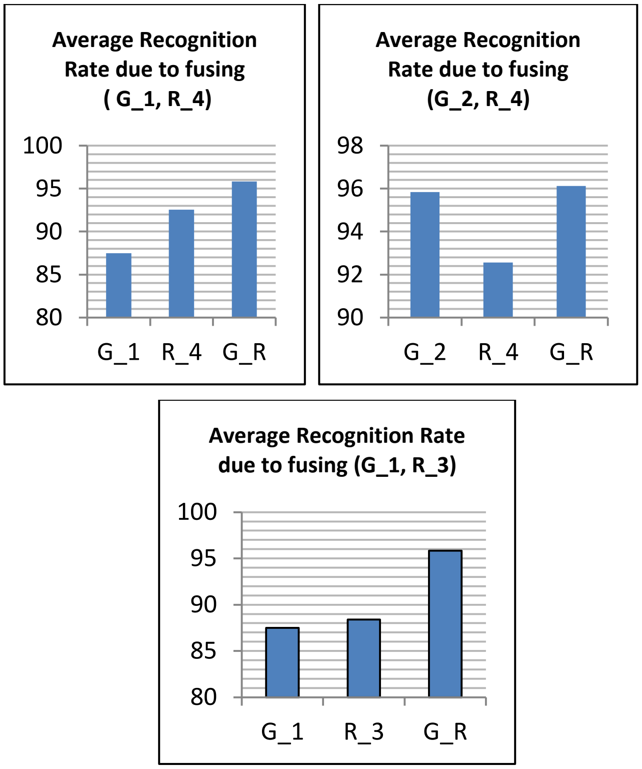


Figure 8. Average recognition rate of Fusing GLCM features and RLM features

On the other hand, fusing two MIs features with two GLCM features leads to the best recognition rate (99.4%), as shown in Figure 9.

TABLE 7. The moment features with the GLCM features

Recognition rate	Gaussian	impulse noise	salt & pepper	Average
moments with 2 features (M_2)	98.214	99.107	99.107	98.809
GLCM with 2 features (G_2)	94.642	96.428	96.428	95.833
moments and GLCM (M_G)	98.214	100	100	99.404
moments with one feature (M_1)	67.857	75.892	73.214	72.321
GLCM with one feature (G_1)	85.714	85.714	91.071	87.5
moments and GLCM (M_G)	88.392	89.285	94.642	90.773
moments with 2 features (M_2)	98.214	99.107	99.107	98.809
GLCM with 1 feature (G_1)	85.714	85.714	91.071	87.5
moments and GLCM (M_G)	88.392	89.285	94.642	90.773
moments with 1 feature (M_1)	67.857	75.892	73.214	72.321
GLCM with 2 features (G_2)	94.642	96.428	96.428	95.833
moments and GLCM (M_G)	98.214	100	100	99.404

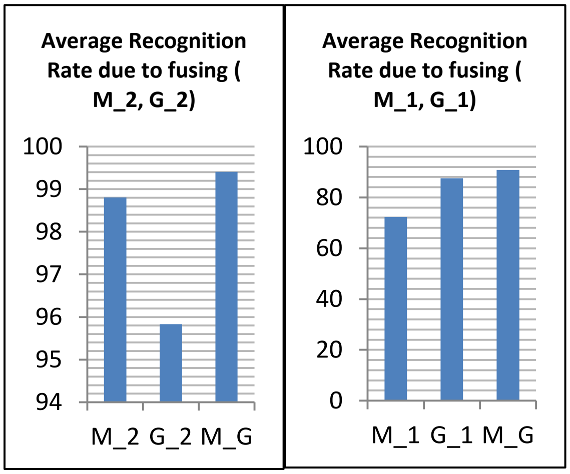


Figure 9. Average recognition rate of Fusing IMs and GLCM features

TABLE 8. GLCM features with the Statistical features

Recognition rate	Gaussian	impulse noise	salt & pepper	Average
statistical with 2 features (S_2)	80.357	88.392	87.5	85.416
GLCM with 2 features (G_2)	94.642	96.42	9	95.833
statistical and GLCM (S_G)	95.535	97.321	97.321	96.726
statistical with one feature (S_1)	80.357	87.5	87.5	85.119
GLCM with one feature (G_1)	85.714	85.714	91.071	89.285
statistical and GLCM (S_G)	85.7142	86.607	91.964	89.880

However, fusing features of GLCM with features of SFIH, gives very small enhancement in the recognition rate as shown in Figure 10.

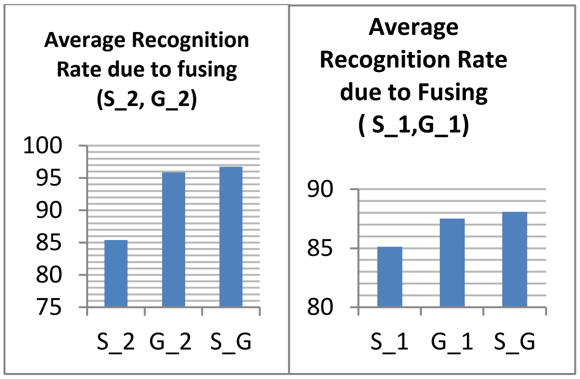


Figure 10. Average recognition rate of Fusing GLCM features and SFIH features

V. CONCLUSIONS & FUTURE WORK

This paper investigates the performance of approaches for the statistical feature extraction techniques, namely; Moment Invariants, Gray Level Co-occurrence Matrix, Run Length Matrix, and Statistical Features of Intensity Histogram, and proposes a hybrid technique fusing features from the four methods for enhancing the Arabic characters recognition rate. Three types of noise, with different intensity levels were used for estimating the gained enhancement, namely; salt & pepper, impulse, and Gaussian noise, with intensity levels of 1%, 3%, and 5% for each type of noise. It was found that the fusion of the moment features with those of GLCM leads to about 100% recognition rate for all noise intensity levels used. Further investigation is needed for fusing more than two types of features and using higher intensity levels to generalize the obtained results.

REFERENCES

[1] G.S.Peake and T.N.Tan, "Script and Language Identification from Document Images", Proceeding ACCV '98 Proceedings of the Third Asian Conference on Computer Vision-Volume II, Springer-Verlag London, UK ©1997, ISBN:3-540-63931-4.

[2] Al-Dmour, Ayman; Zitar, Raed Abu, "Arabic writer identification based on hybrid spectral-statistical measures", http://en.zl50.com/1201106198616921.html, Volume 19, Number 4, December 2007, pp. 307.

[3] Gheith Abandah and Nasser Anssari, "Novel Moment Features Extraction for Recognizing Handwritten Arabic Letters", Journal of Computer Science 5 (3): 226-232, 2009, ISSN 1549-3636.

[4] Xiaoou Tang, "Texture information in run-length matrices", IEEE Transactions on Image Processing, Vol.7, Issue 11, pp 1602-1609, 1998.

[5] Frank J. Aherne; Neil A. Thacker; Peter I Rockett, “The Bhattacharyya metric as an absolute similarity measure for frequency coded data”, Kybernetika, Vol. 34 (1998), No. 4, [363]—368.

[6] H. Aboaisha1, Zhijie Xu1, I. El-Feghi, “An Investigation On Efficient Feature Extraction Approaches For Arabic Letter Recognition”, Proceedings of The Queen’s Diamond Jubilee Computing and Engineering Annual Researchers’ Conference 2012: CEARC’12. University of Huddersfield, pp. 80-85. ISBN 978-1-86218-106-9.

[7] David A. Clausi, Huawu Deng, “Design-based texture feature fusion using Gabor filters and co-occurrence probabilities”, IEEE Transactions on Image Processing, 2005,Vol. 14, issue 7.pages 925—936.

[8] Øivind Due Trier, Anil K. Jain, Torfinn Taxt, “Feature extraction methods for character recognition-A survey” ELSEVIER, Pattern Recognition, Volume 29, Issue 4, April 1996, Pages 641–662.

[9] Qing Chen, “Evaluation of OCR Algorithms for Images with Different Spatial Resolutions and Noises”, A Master of Applied Science thesis submitted to the School of Graduate Studies and Research, Ottawa-Carleton Institute for Electrical Engineering, School of Information Technology and Engineering, Faculty of Engineering, University of Ottawa, Canada, 2003.

[10] R.Muralidharan1, C.Chandrasekar, “Object Recognition using SVM-KNN based on Geometric Moment Invariant”, International Journal of Computer Trends and Technology- July to Aug Issue 2011, ISSN: 2231-2803, Page 215-220.

[11] R. J. Ramteke, “Invariant Moments Based Feature Extraction for Handwritten Devanagari Vowels Recognition”, International Journal of Computer Applications (0975 - 8887), Volume 1 – No. 18, 2010.

[12] Fritz Albregtsen, “Statistical Texture Measures Computed from Gray Level Co-occurrence Matrices, Image Processing Laboratory, Department of Informatics, University of Oslo, November 5, 2008.

[13] Zidouri, A., "PCA-Based Arabic Character Feature Extraction" IEEE, 9th International Symposium on Signal Processing and its Applications, Vol S1-3; pp: 652-655;King Fahd University of Petroleum & Minerals, (2007) .http://www.kfupm.edu.sa.

[14] http://www.fp.ucalgary.ca/mhallbey/GLCM_as_probability.htm

[15] Dong-Hui Xu, Arati S. Kurani, Jacob D. Furst, Daniela S. Raicu, “Run-Length Encoding For Volumetric Texture”,

[16] X. Tang, Texture information in run-length matrices, IEEE Transactions on Image Processing, 7(11), 1998, 1602-1609.

APPENDICES

APPENDIX A$_{\mathrm{I}}$ (MOMENT FEATURES)

$$M_1 = (u_{20} + u_{02})$$

$$M_2 = (u_{20} - u_{02})^2 + 4u_{11}^2$$

$$M_3 = (u_{30} - 3u_{21})^2 + (3u_{21} - u_{30})^2$$

$$M_4 = (u_{30} + u_{12})^2 + (u_{21} + u_{03})^2$$

$$M_5 = (u_{30} - 3u_{12})(u_{30} + u_{12})((u_{30} + u_{12})^2 - 3(u_{21} + u_{03})^2) + (3u_{12} - u_{30})(u_{21} + u_{03})(3(u_{30} + u_{12})^2 - (u_{21} + u_{03})^2)$$

$$M_6 = (u_{20} - u_{02})((u_{30} + u_{12})^2 - (u_{21} + u_{03})^2) + 4u_{11}(u_{30} + 3u_{12})(u_{21} + u_{03})$$

$$M_7 = (3u_{12} - u_{30})(u_{30} + u_{12})((u_{30} + u_{12})^2 - 3(u_{21} + u_{03})^2) - (u_{30} - 3u_{12})(u_{21} + u_{03})(3(u_{30} + u_{12})^2 - (u_{21} + u_{03})^2)$$

APPENDIX A$_{\mathrm{II}}$ (GLCM FEATURES)

$$Contrast = \sum_{i,j=0}^{N-1} P_{i,j}\,(i-j)^2 \quad ,$$

$$Homogenity = \sum_{i,j=0}^{N-1} \frac{P_{i,j}}{1+(i-j)^2}$$

$$Dissimilarity = \textstyle\sum_{i,j=0}^{N-1} P_{i,j}\,|i-j|,$$

$$Similarity = \sum_{i,j=0}^{N-1} \frac{P_{i,j}}{1+|i-j|}$$

$$Homogenity = \sum_{i,j=0}^{N-1} \frac{P_{i,j}}{1+(i-j)^2}$$

$$Angular\ Second\ Moment(ASM) = \sum_{i,j=0}^{N-1} {P_{i,j}}^2$$

$$GLCM\ Mean{:}\ \mu_i = \sum_{i,j=0}^{N-1} i(P_{i,j}), \qquad \mu_j = \sum_{i,j=0}^{N-1} j(P_{i,j})$$

$$GLCM\ Variance{:}\ {\sigma_i}^2 = \sum_{i,j=0}^{N-1} P_{i,j}\,(i-\mu_i)^2,$$

$${\sigma_j}^2 = \sum_{i,j=0}^{N-1} P_{i,j}\,(j-\mu_j)^2$$

$$Standard\ Deviation{:}\ \sigma_i = \sqrt{{\sigma_i}^2},\ \ \sigma_j = \sqrt{{\sigma_j}^2}$$

$$GLCM\ Correlation{:}\ \sum_{i,j=0}^{N-1} P_{i,j}\Big[\frac{(i-\mu_i)(j-\mu_j)}{\sqrt{{\sigma_i}^2}\sqrt{{\sigma_i}^2}}\Big]$$

APPENDIX$_{\mathrm{III}}$ (RUN LENGTH FEATURES)

$$SRE = \frac{1}{n_r}\sum_{i=1}^{M}\sum_{j=1}^{N}\frac{p(i,j)}{j^2}$$

$$LRE = \frac{1}{n_r}\sum_{i=1}^{M}\sum_{j=1}^{N} p(i,j) * j^2$$

$$HGRE = \frac{1}{n_r}\sum_{i=1}^{M}\sum_{j=1}^{N} p(i,j) * i^2$$

$$LGRE = \frac{1}{n_r}\sum_{i=1}^{M}\sum_{j=1}^{N}\frac{p(i,j)}{i^2}$$

$$SRLGE = \frac{1}{n_r}\sum_{i=1}^{M}\sum_{j=1}^{N}\frac{p(i,j)}{i^2 * j^2}$$

$$SRHGE = \frac{1}{n_r}\sum_{i=1}^{M}\sum_{j=1}^{N}\frac{p(i,j) * i^2}{j^2}$$

$$LRLGE = \frac{1}{n_r}\sum_{i=1}^{M}\sum_{j=1}^{N}\frac{p(i,j) * j^2}{i^2}$$

$$LRHGE = \frac{1}{n_r}\sum_{i=1}^{M}\sum_{j=1}^{N} p(i,j) * i^2 * j^2$$

APPENDIX A$_{\mathrm{IV}}$ (STATISTICAL FEATURES)

$$\mu = \bar{X} = \frac{1}{N}\sum_{i} X_i$$

$$\sigma^2 = \frac{1}{N-1}\Big(\sum (X_i - \bar{X}\Big)^2)$$

$$\sigma = \sqrt{\tfrac{1}{N-1}\,(\textstyle\sum (X_i - \bar{X})^2)}$$

$$R = 1 - \frac{1}{(1+\sigma^2)}$$

$$\mu_3 = \frac{\sum (X-\mu)^3}{N\sigma^4}$$

$$\mu_4 = \frac{\sum (X-\mu)^4}{N\sigma^4}\text{-}3$$

$$U = \textstyle\sum_{i=0}^{L-1} p^2(z_i)$$

$$e = -\textstyle\sum_{i=0}^{L-1} p(z_i)\,log_2\,p(z_i)$$

$$RLNU = \frac{1}{n_r}\sum_{i=1}^{M}(\sum_{j=1}^{N} p(i,j))^2$$

$$RPC = \frac{n_r}{P(i,j) * j}$$

Automatic Melakarta Raaga Identification Syste: Carnatic Music

B. Tarakeswara Rao[1], Sivakoteswararao Chinnam[2], P Lakshmi Kanth[3], M.Gargi[4]
School of Computing Vignan University,[1,2]
Department of IT Vignan Lara Institute of Technology [3,4]

Abstract— **It is through experience one could as certain that the classifier in the arsenal or machine learning technique is the Nearest Neighbour Classifier. Automatic melakarta raaga identification system is achieved by identifying the nearest neighbours to a query example and using those neighbours to determine the class of the query. This approach to classification is of particular importance today because issues of poor run-time performance are not such a problem these days with the computational power that is available. This paper presents an overview of techniques for Nearest Neighbour classification focusing on; mechanisms for finding distance between neighbours using Cosine Distance, Earth Movers Distance and formulas are used to identify nearest neighbours, algorithm for classification in training and testing for identifying Melakarta raagas in Carnatic music. From the derived results it is concluded that Earth Movers Distance is producing better results than Cosine Distance measure.**

Keywords- Music; Melakarta Raaga; Cosine Distance; Earth Movers Distance; K-NN;

I. Introduction

Performance in Indian classical music is always within a Melakarta raaga, except for solo percussion. Melakarta raaga is a system within which performers improvise and compose. Melakarta raagas are often summarized by the notes they use, though many Melakarta raagas in fact share the same notes. Melakarta raaga recognition is a difficult task even for humans. A Melakarta raaga is popularly defined as a specified combination, decorated with embellishments and graceful consonances of notes within a mode which has the power of evoking a unique feeling distinct from all other joys and sorrows. It possesses something of a transcendental element.

A Melakarta raaga is characterized by several attributes, like its Vaadi-Samvaadi, Aarohana-Avrohana and Pakad [17], besides the sequence of notes. It is of utmost importance to note here that no two performances of the same Melakarta raaga, even two performances by the same artist, will be identical. A certain music piece is considered a certain Melakarta raaga, as long as the attributes associated with it are satisfied. This concept of Indian classical music, in that way, is very open.

Based on this work the following major contributions to the study of musical raagas and KNN with CD and EMD are made. In first place, our solutions based primarily on techniques from speech processing and pattern matching, which shows that techniques from other domains can be purposefully extended to solve problems in computational musical raagas, Secondly, the two note transcription methods presented are novel ways to extract notes from sample raagas of Indian classical music. This approach has given very encouraging results.

The rest of the paper is organized as follows. Section 2 highlights some of the useful and related previous research work in the area. The solution strategy is discussed in detail in Section 3. The test procedures and experimental results are presented in Section 4, Finally, Section 5 lists out the conclusions.

II. Pevious Work

The earlier work in Carnatic music retrieval is on a slow pace compared to western music. Some work is being done in Swara identification [1] and Singer identification [2] of Carnatic music. In Hindustani music work has been done in identifying the Melakarta raaga of Hindustani music [3]. In [3][19] the authors have created a HMM based on which they have identified two raagas of Hindustani music. The fundamental difference between Hindustani Raaga pattern and Carnatic Raaga pattern is that in Hindustani R1, R2 are present as against R1, R2, R3 in Carnatic. Similarly G, D, N all has three distinct frequencies in Carnatic music as compared to two frequencies in Hindustani [8]. This reduces the confusion in identifying the distinct frequencies in Hindustani music as compared to Carnatic music. The authors have not used polyphonic music signal and have assumed that the input music signal is a voice only signal.

The fundamental frequency of the signal was also assumed and based on these features the Melakarta raaga identification process was done for two Hindustani raagas. On the western music aspect, melody retrieval is being performed by researchers. The one proposed by [9] is based on identifying the change in frequency in the given query.

The query is received in the form a humming tune and based on the rise and fall in the pitch of the received query, the melody pattern that matches with the query's rise and fall of pitch is retrieved. The melody retrieval based on features like distance measures and gestalt principles. The approach is based on low level signal features and the Melakarta raaga is identified by considering different instrument signal as input to our system. In the present work Melakarta raaga identification is done using KNN with two different distance metrics one CD and the other EMD.

III. PRESENT WORK

K Nearest Neighbour has rapidly become one of the booming technologies in today's world for developing convoluted control systems. Melakarta raaga Recognition is the fascinating applications of KNN – which is basically used in Melakarta raaga identification for many cases, Melakarta raaga detection is considered as a rudimentary nearest neighbour problem. The problem becomes more fascinating because the content is an audio – given an audio find the audio closest to the query from the trained database.

The intuition underlying Nearest Neighbour Classification is quite straight forward, classified based on the class of their nearest neighbours. It is often useful to take more than one neighbour into account so the technique is more commonly referred to as k-Nearest Neighbour (k-NN) Classification where k nearest neighbours are used in determining the class. Since the training examples are needed at run-time, i.e. they need to be in memory at run-time, it is sometimes also called Memory-Based Classification. Because induction is delayed to run time, it is considered a Lazy Learning technique. Because classification is based directly on the training examples it is also called Example-Based Classification or Case-Based Classification.

The basic idea is as shown in Figure 1 which depicts a 3-Nearest Neighbour Classifier on a two-class problem in a two-dimensional feature space. In this example the decision for q_1 is straightforward – all three of its nearest neighbours are of class O so it is classified as an O. The situation for q_2 is a bit more complicated at it has two neighbours of class X and one of class O. This can be resolved by simple majority voting or by distance weighted voting (see below). So k−NN classification has two stages; the first is the determination of the nearest neighbours and the second is the determination of the class using those neighbours. The following section describes the techniques CD and EMD which is used to raaga classification.

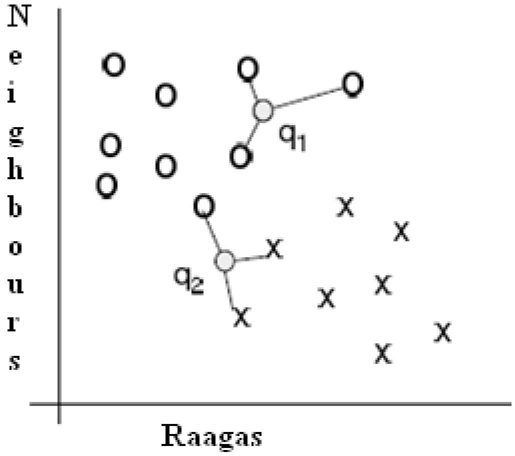


Fig. 1 A simple example of 3-Nearest Neighbour Classification

A. COSINE DISTANCE

Cosine similarity (CD) between two vectors x and y is defined as:

CD(x; y) = $(x^T * y)/(||x||*||y||)$ ------- (1)

Cosine similarity has a special property that makes it suitable for metric learning: the resulting similarity measure is always within the range of -1 and +1. This property allows the objective function to be simple and effective.

B. EARTH MOVER DISTANCE (EMD)

The Earth Mover Distance is based on the solution to a discrete optimal mass transportation problem. EMD represents the minimum cost of moving earth from some source locations to fill up holes at some sink locations. In other words, given any two mass (or probability) distributions, one of them can be viewed as a distribution of earth and the other a distribution of holes, then the EMD between the two distributions is the minimum cost of rearranging the mass in one distribution to obtain the other. In the continuous setting, this problem is known as the Monge-Kantorovich optimal mass transfer problem and has been well studied over the past 100 years the importance here is that EMD can be used to measure the discrepancy between two multidimensional distributions.

C. METHODOLOGY/ALGORITHM FOR MELAKARTA RAAGA RECOGNITION SYSTEM

Following is the methodology is used for the Melakarta raaga Recognition for training and testing. Initially first k-Nearest Neighbour Classifier is determined on a two-class problem in a two-dimensional feature space which is shown in the following diagram raagas in horizontal axis and neighbours of raaga on the vertical axis. In this proposed approach the decision for raaga is straightforward – one of its nearest neighbours is of class O and one of class X.

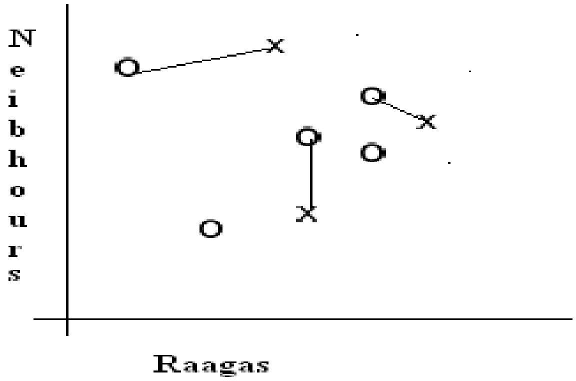


Fig. 2 1-Nearest Neighbour classification of Raagas

A training dataset D is made up of (xi), I Є[1,|D|] training samples where xi is the raaga. The raaga is divided in to 15 samples by eliminating unwanted frequencies (disturbances, accompanied instruments) by using low level filter-Fourier Transform of a Signal (Spft). The same process is repeated for each raaga in database D. Then these samples are trained by using Self- Organizing and Learning Vector Quantization Nets. The grouping process is carried by us. Each training example is labeled with a class label y_j Є Y. Our objective is to classify an unknown example raaga q. Now training process is completed. Next the testing phase is performed by using KNN classification.

The KNN approach carried in two phases

1 Determination of Nearest Neighbours

2 Determination of the class using those neighbours

DETERMINATION OF NEAREST NEIGHBOURS :

For each xi Є D the distance between q and xi is calculated as follows:

$$d(q, x_i) = \sum_{f \in F} \varpi_f \delta(q_f, x_{if})$$ ------(2)

Where x_i = trained raaga ,

q = testing raaga,

f = feature(flow pattern)

w_f = weighted feature of raaga

There are huge ranges of possibilities for this distance metric; a basic version for continuous and discrete attributes would be:

$$\delta(q_f, x_{if}) = \begin{cases} 0 & f \text{ discrete and } q_f = x_{if} \\ 1 & f \text{ discrete and } q_f \neq x_{if} \\ |q_f - x_{if}| & f \text{ continuous} \end{cases}$$

----(3)

The k nearest neighbours is selected based on this distance metric. In order to determine the class of q the majority class among the nearest neighbours is assigned to the query. It will often make sense to assign more weight to the nearer neighbours in deciding the class of the query.

DETERMINATION OF THE CLASS USING THOSE NEIGHBOURS:

If more than one of the neighbours is identified then it can be resolved by simple majority voting or by distance weighted voting. A fairly general technique to achieve this is distance weighted voting where the neighbours get to vote on the class of the query case with votes weighted by the inverse of their distance to the query.

$$Vote(y_i) = \sum_{c=1}^{k} \frac{1}{d(q, x_c)^n} 1(y_j, y_c)$$ ------ (4)

Thus the vote assigned to class y_j by neighbour x_c is 1 divided by the distance to that neighbour, i.e. $1(y_j , y_c)$ returns 1 if the class labels match and 0 otherwise. From the above equation would normally be 1 but values greater than 1 can be used to further reduce the influence of more distant neighbours. Now the distance measures Cosine and EMD measures applied to our KNN process is discussed.

1) *COSINE DISTANCE MEASURE*

The cosine similarity measure is the cosine of the angle between these two vectors, suppose d_i and d_j are the paths between a_i and a_j in instance x_i and instance x_j, respectively. d_i and d_j are represented as vectors of term frequencies in the vector-space model. The cosine is calculated by using the following formula

$$\cos(d_i, d_j) = \frac{\sum_k a_{i,k} . a_{j,k}}{\sqrt{\sum_k a_{i,k}^2} . \sqrt{\sum_k a_{j,k}^2}}$$ ----- (5)

2) *EARTH MOVER DISTANCE*

The Earth Mover Distance (EMD) is a distance measure that overcomes many of problems that arise from the arbitrariness of binning. As the name implies, the distance is based on the notion of the amount of effort required to convert one instrumental music to another based on the analogy of transporting mass from one distribution to another. If two instrumental music are viewed as distributions and view one distribution as a mass of earth in space and the other distribution as a hole (or set of holes) in the same space then the EMD is the minimum amount of work involved in filling the holes with the earth. Some researchers analysis of the EMD argue that a measure based on the notion of a signature is better than one based on a histogram. A signature {s_j = m_j ,wm_j } is a set of j clusters where m_j is a vector describing the mode of cluster j and wm_j is the fraction of features falling into that cluster.

Thus, a signature is a generalization of the notion of a histogram where boundaries and the number of partitions are not set in advance; instead j should be 'appropriate' to the complexity of the instrumental music. The example in Figure 3 illustrates this idea. The clustering can be thought as a quantization of the instrumental music in some frequency space so that the instrumental music is represented by a set of cluster modes and their weights. In the figure the source instrumental music is represented in a 2D space as two points of weights 0.6 and 0.4; the target instrumental music is represented by three points with weights 0.5, 0.3 and 0.2. In this example the EMD is calculated to be the sum of the amounts moved (0.2, 0.2, 0.1 and 0.5) multiplied by the distances they are moved. Calculating the EMD involves discovering an assignment that minimizes this amount.

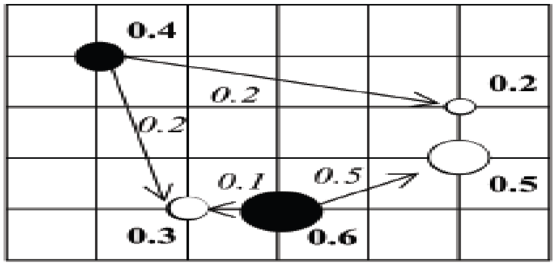


Fig. 3. An example of the EMD between two 2D signatures with two points (clusters) in one signature and three in the other.

For two instrumental music described by signatures S = {m_j ,wm_j }n_j =1 and Q = {p_k,wp_k}r k=1 . The work required to transfer from one to the other for a given flow pattern F:

$$WORK(S, Q, F) = \sum_{j=1}^{n} \sum_{k=1}^{r} d_{jk} f_{ik}$$ ---- (6)

where d_{jk} is the distance between clusters m_j and p_k and f_{jk} is the flow between m_j and p_k that minimizes overall cost. Once the transportation problem of identifying the flow that minimizes effort is solved by using dynamic programming. The EMD is defined as:

$$EMD(S,Q) = \frac{\sum_{j=1}^{n} \sum_{k=1}^{r} d_{jk} \; f_{jk}}{\sum_{j=1}^{n} \sum_{k}^{r} f_{jk}} \qquad \text{-----(7)}$$

EMD is expensive to compute with cost increasing more than linearly with the number of clusters. Nevertheless it is an effective measure for capturing similarity between instrumental music. It is identified that the EMD approach is giving better results than Cosine measure.

IV. Results And Discussion

The input signal is sampled at 44.1 KHz. The identification of different Raagams for the purpose of evaluating this algorithm is considered. For the purpose of Melakarta raaga identification seven different instruments are considered. The signal is made to pass through the signal separation algorithm, and segmentation algorithm.

The result showing the segmentation points for one input is shown in below Figures. This is the first level of segmentation where the protruding lines indicate the points of segmentation. After identifying the segmentation points the frequency components are determined using the HPS algorithm and tabulated the frequency values which have the dominant energy. Using the raaga identification system, the confusion matrix is determined.

The following figure shows the plot graphs and edge detection graphs:

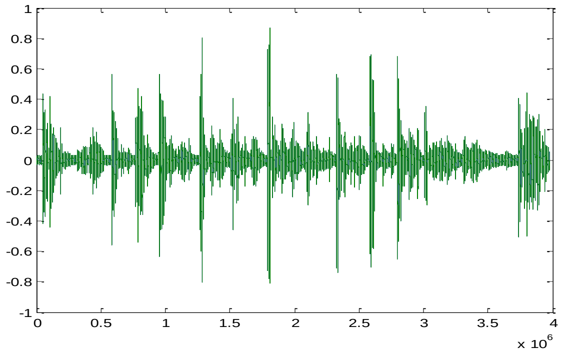


Fig.4 Plot Graph for Begada Raaga

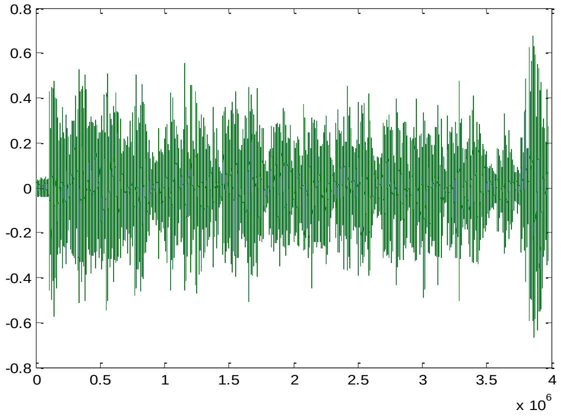


Fig. 5 Plot Graph for Kharaharapriya Raaga

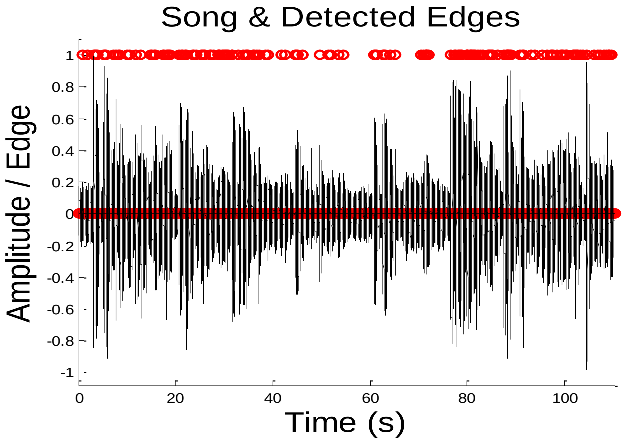


Fig. 6 Bhiravi raaga for Edge detection

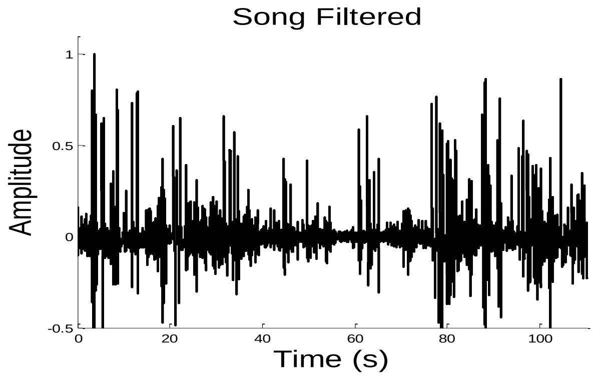


Fig. 7 Bhiravi raaga for Song Filtered

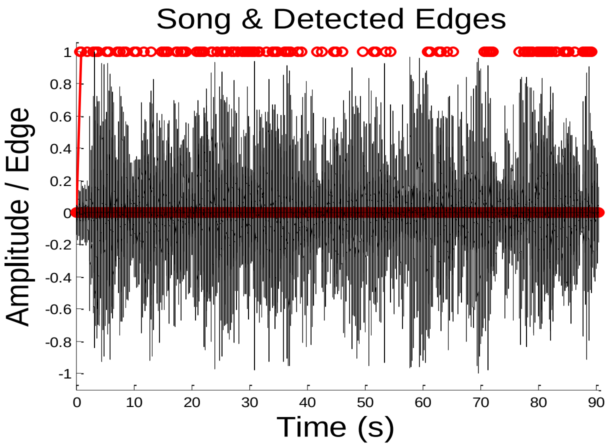


Fig. 8 Malahari raaga for Edge detection

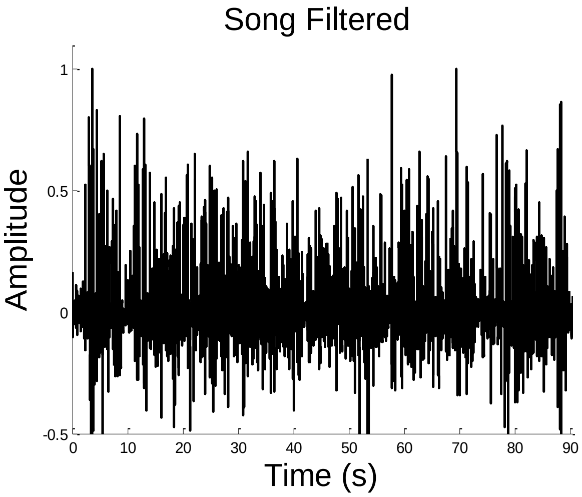


Fig. 9 Malahari raaga for Song Filtered

The following are the results obtained by applying Cosine Distance measure

Cosine Distance: The Data is same for Train and Sample

Table 1 Confusion Matrix: Same data for Train and Sample

Name of the Raaga	Recognized Raagas (%)				
	Begada	Vanasapathi	sundavinodini	Desh	Hindilom
Begada	92	56	56	60	65
Vanasapathi	58	92	63	70	72
sundavinodini	58	68	92	68	70
Desh	62	70	76	92	75
Hindilom	65	72	70	85	92

Cosine Distance: The Data is different for Train and Sample

Table 2 Confusion Matrix: Different data for Train and Sample

Name of the Raaga	Recognized Raagas (%)				
	Begada	Vanasapathi	sundavinodini	Desh	Hindilom
Sri	90	58	58	62	65
Bhiravi	58	88	63	70	72
Abheri	58	68	78	68	70
Malahari	62	70	76	84	75
Sahana	65	72	70	85	86

The following are the results obtained by applying EMD Distance measure

EMD: The Data is same for both Train and Sample

Table 3 Confusion Matrix: Same data for Train and Sample

Name of the Raaga	Recognized Raagas (%)				
	Begada	Vanasapathi	sundavinodini	Desh	Hindilom
Begada	**96**	66	57	74	68
Vanasapathi	78	**96**	65	82	80
sundavinodini	72	78	**96**	88	70
Desh	72	70	76	**96**	85
Hindilom	66	74	72	86	**96**

EMD: The Data is different for Train and Sample

Table 4 Confusion Matrix: Different data for Train and Sample

Name of the Raaga	Recognized Raagas (%)				
	Begada	Vanasapathi	sundavinodini	Desh	Hindilom
Sahana	89	68	78	62	65

Todi	68	88	63	70	72
Sri	58	68	78	68	70
Bhiravi	72	70	76	84	75
Abheri	75	72	70	85	86
Malahari	70	75	68	78	80

V. Conclusion

K-NN is very simple to understand and easy to implement. So it should be considered in seeking a solution to any classification problem. In some circumstances where an explanation of the output of the classifier is useful, K-NN can be very effective if an analysis of the neighbours is useful as explanation. In order to improve classification process an EMD approach is used for fast convergence. K-NN is very sensitive to irrelevant or redundant features because all features contribute to the similarity and thus to the classification. This can be ameliorated by EMD approach and feature selection or feature weighted voting. The EMD results are compared with Cosine distance measure and observed that EMD gives better results.

References

[1] Rajeswari Sridhar, Geetha T. V, "Swara Indentification for South Indian Classical Music", ICIT '06 Proceedings of the 9th International Conference on Information Technology, IEEE Computer Society, ISBN:0-7695-2635-7.

[2] Youngmoo E. Kim, Brian Whitman" Singer Identification in Popular Music Recordings Using Voice Coding Features", citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.115.

[3] Paul G., Corine G.M., Christophe C., Vincent F. "automatic classification of environmental noise events by hidden Markov models", citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.52

[4] Berger. "Some factors in the recognition of timbre". *J. Audio. Eng. Soc*. 30, pp. 396-406.

[5] Clark, Milner. "Dependence of timbre on the tonal loudness produced by musical instruments". *J. Audio. Eng. Soc*. 12, pp. 28-31.

[6] Eagleson, H. W., Eagleson, O. W. "Identification of musical instruments when heard directly and over a public-address system". *J. Acoust. Soc. Am*. 19, pp. 338-342.

[7] Strong, Clark. "Perturbations of synthetic orchestral wind instrument tones". *J. Acoust. Soc. Am*., Vol. 41, pp. 277-285.

[8] Bhatkande.V (1934), Hindusthani Sangeet Paddhati.Sangeet Karyalaya, 1934.

[9] Schroeder.M.R (1968), " Period histogram and product spectrum: New methods for fundamental-frequency measurement", Journal of the Acoustical Society of America, , vol. 43, no. 4, 1968.

[10]. A. Ghias, J. Logan, D. Chamberlin and B. C. Smith: "Query by Humming – Musical Information Retrieval in an Audio Database": Proc. ACM Multimedia, pp. 231-236: 1995.

[11]. H. Deshpande, U. Nam and R. Singh: "MUGEC: Automatic Music Genre Classi- fication": Technical Report, Stanford University: June 2001.

[12]. S. Dixon: "Multiphonic Note Identification": Proc. 19th Australasian Computer Science Conference: Jan-Feb 2003.

[13]. W. Chai and B. Vercoe: "Folk Music Classification Using Hidden Markov Models": Proc. Internation Conference on Artificial Intelligence: June 2001.

[14]. A. J. Viterbi: "Error bounds for convolutional codes and an asymptotically optimal decoding algorithm": IEEE Transactions on Information Theory, Volume IT-13, pp.260-269: April 1967.

[15]. L. E. Baum: "An inequality and associated maximization technique in statistical estimation for probabilistic functions of Markov processes": Inequalities, Volume 3, pp. 1-8: 1972.

[16]. L. E. Baum and T. Petrie: "Statistical inference for probabilistic functions of finite state Markov chains": Ann.Math.Stat., Volume 37, pp. 1554-1563: 1966.

[17]. Gaurav Pandey, Chaitanya Mishra, and Paul Ipe "TANSEN: a system for automatic raga identification"

[18]. A. Prasad et al. "Gender Based Emotion Recognition System for Telugu Rural Dialects using Hidden Markov Models" Journal of Computing: An International Journal, Volume2 ,Number 6 June 2010 NY, USA, ISSN: 2151-9617

[19]. Tarakeswara Rao B et. All "A Novel Process for Melakartha Raaga Recognitionusing Hidden Markov Models (HMM)", International Journal of Research and Reviews in Computer Science (IJRRCS), Volume 2, Number 2,April 2011, ISSN: 2079-2557.