Studies in Computational Intelligence 542

Liming Chen
Supriya Kapoor
Rahul Bhatia *Editors*

Intelligent Systems for Science and Information

Extended and Selected Results from the Science and Information Conference 2013

Springer

Studies in Computational Intelligence

Volume 542

Series editor

Janusz Kacprzyk, Polish Academy of Sciences, Warsaw, Poland
e-mail: kacprzyk@ibspan.waw.pl

For further volumes:
http://www.springer.com/series/7092

About this Series

The series "Studies in Computational Intelligence" (SCI) publishes new developments and advances in the various areas of computational intelligence—quickly and with a high quality. The intent is to cover the theory, applications, and design methods of computational intelligence, as embedded in the fields of engineering, computer science, physics and life sciences, as well as the methodologies behind them. The series contains monographs, lecture notes and edited volumes in computational intelligence spanning the areas of neural networks, connectionist systems, genetic algorithms, evolutionary computation, artificial intelligence, cellular automata, self-organizing systems, soft computing, fuzzy systems, and hybrid intelligent systems. Of particular value to both the contributors and the readership are the short publication timeframe and the worldwide distribution, which enable both wide and rapid dissemination of research output.

Liming Chen · Supriya Kapoor
Rahul Bhatia
Editors

Intelligent Systems for Science and Information

Extended and Selected Results from the Science and Information Conference 2013

Springer

Editors
Liming Chen
School of Computer Science
and Informatics
De Montfort University
The Gateway, Leicester, LE1 9BH
United Kingdom

Rahul Bhatia
The Science and Information Organization
New York
USA

Supriya Kapoor
The Science and Information Organization
New York
USA

ISSN 1860-949X ISSN 1860-9503 (electronic)
ISBN 978-3-319-04701-0 ISBN 978-3-319-04702-7 (eBook)
DOI 10.1007/978-3-319-04702-7
Springer Cham Heidelberg New York Dordrecht London

Library of Congress Control Number: 2013958350

Editor's Preface

The Science and Information (SAI) Organization (www.thesai.org) is an international professional organisation dedicated to promoting research, technology and development by providing multiple platforms for collaboration of professionals and researchers to share existing and generate new knowledge. It has predominately focused on the general areas of computer science and information technologies including emerging technology trends such as cloud computing, big data and ambient intelligence, communication systems such as 3G/4G network evolution and mobile ad-hoc networks, electronics such as novel sensing and sensor networks, security such as secure transactions, cryptography and cyber law, machine vision such as virtual reality and video analysis, intelligent data management such as artificial intelligence, neural networks, data mining and knowledge management and e-learning and e-business. Science and Information (SAI) Conference (www.conference.thesai.org) is a premier annual event organised by SAI for researchers and industry practitioners to disseminate and share their ideas, original research results and practical development experiences from all of the aforementioned areas.

SAI Conference 2013 has attracted huge attention from researchers and practitioners around the world. The three-day event in October 2013 witnessed 160 scientists, technology developers, young researcher (i.e. PhD students) and industrial practitioners from more than 55 countries engaging intensively in presentations, demonstrations and informal discussions. The inspiring keynote speeches and the state-of-the-art lectures have deeply motivated attendees and envisioned future research directions. The conference has greatly facilitated knowledge transfer and synergy, bridged gaps between different research communities/groups, laid down foundation for common purposes, and helped identify opportunities and challenges for interested researchers and technology and system developers.

To further the dissemination of high quality research and novel technologies presented in SAI Conference 2013, twenty-four chapters in the field of Intelligent Systems, which received highly recommended feedback during SAI Conference

2013 review, have been selected for this special edition of Springer book "Intelligent Systems 2013".

All chapters have gone through substantial extension and consolidation and are subject to another round of rigorous review and additional modification. We believe that these chapters represent the state of the art of the cutting-edge research and technologies in related areas, and can help inform relevant research communities and individuals of the future development in SAI.

The success of The Science and Information (SAI) Organization in general, and the SAI Conference 2013 in particular, is attributed to the strong support of many people: authors, presenters, participants, keynote speakers, session chairs, organizing committee members, student volunteers, program committee members, steering committee members, and people in other various roles. We would like to take this opportunity to express our gratitude for their valuable contributions.

De Montfort University
United Kingdom Prof. Liming Chen

Contents

A Smart Monitoring System for Assisted Living Support Using Adaptive Lifestyle Pattern Analysis

A Vision for the Future

Besim Mustafa[1], Peter Matthew[1], and Farrukh Naveed[2]

[1] Department of Computing, Edge Hill University, Ormskirk, UK
[2] Securecom Ltd., Rochdale, UK
{mustafab, peter.matthew}@edgehill.ac.uk,
f.naveed@securecom.uk.com

Abstract. In recent years there has been a rapidly increasing intensity of work going into investigating various methods of facilitating assisted living for the benefit of the elderly and those with difficulties in mobility. This chapter describes one such effort which distinguishes itself from the rest by considering and describing a system with true commercial potential and thus significant social impact. Promising efforts in investigating and identifying the requirements for a system of smart monitoring and adaptive lifestyle pattern detection and analysis are described. An initial proposal for a system relying on remote monitoring using persistent communications technology and a centralized data gathering, analysis and decision making is presented. During the initial development stage requirements for sensor placements, efficient sensor data formats and transmission protocols became apparent; unit testing and system validation demanded generation of large amounts of suitable sensor data. Here we also describe a simulator we developed in order to support these requirements; the rationale behind the simulator, its main functions and usage and the positive contribution it has made during the initial stages and the prototyping phases of the above system are explained. Finally a prototype developed in facilitating initial investigations is described and the vision for future developments is articulated.

Keywords: assisted living, pattern recognition, remote sensors, communications protocols, simulation, rule-based inference.

1 Introduction

Advancements in technology have significantly changed the way we think and act. Banking, travel, science, education and health are some of the areas that have seen huge improvements as a result of recent innovations in technology. This is especially true in the case of the health services. Leveraging modern computing power, communications technology and advancements in software developments for the benefit of those who suffer disabilities and health problems and are in need of regular supervision and personal caring have become essential for providing the best support in affordable, enhanced and caring social services. This support is aimed at providing two

L. Chen, S. Kapoor, and R.Bhatia (eds.), *Intelligent Systems for Science and Information*, Studies in Computational Intelligence 542,
DOI: 10.1007/978-3-319-04702-7_1,

main benefits: enhancing independent living of the individuals affected and making significant savings in the cost of support to individuals provided by the social services and the local authorities.

Our motivation for pursuing this research project initially stems from several important considerations often associated with ageing: rapidly increasing numbers of aged people, correspondingly increasing old-age related chronic illnesses and disabilities and increasing rate of life-threatening injuries. Fig 1(left) illustrates that in UK the number of people aged sixty and over is projected to increase from 12 million in 2001 to 18.6 million in 2031 [1]. At the same time according to Fig 1(right) chronic diseases and disabilities are projected to increase twofold to threefold [1] for people aged 65 or over. Mental illnesses like feeling alone, fear of falling, feeling weak and depression often lead to excessive intake of alcohol that further increases the risk of falling, poor mental and physical health. The following statistics on falls in UK represent some sobering facts. About a third of all people aged over 65 fall each year (which is equivalent to over 3 million); falls represent over half of hospital admissions for accidental injury and the combined cost of hospitalisation and social care for hip fractures (most of which are due to falls) is £2 billion a year or £6 million a day [2].

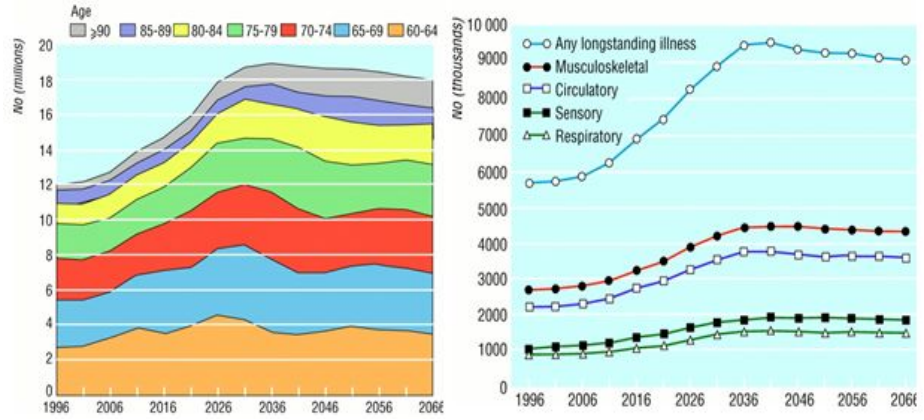


Fig. 1. (Left) Projected numbers of people aged 60 years and over in the UK; (Right) Projected numbers of people aged 65 or over with chronic illness in UK [2]

Most humans naturally prefer independent living but with advancing age or disabilities this becomes risky when living alone; elderly or people with disabilities often end up living in sheltered housing where nursing and care facilities are available at a price. This imposes increasing pressure on the resources of social services and local authorities to provide efficient and cost effective carer services especially during the current economic climate in which the stringent austerity measures are likely to become more severe.

It is in these settings that we at the Department of Computing got together with a local high-tech commercial security systems company Securecom Ltd. based in Rochdale, UK. Securecom design and develop specialist remote monitoring systems for the security industry and work closely with local authorities to provide secure

housing for a large number of local authority tenants. The collaborative work is on the initial definition of the requirements for and the subsequent development of a scalable assisted living support system based on the extensive experiences of Securecom on remote monitoring and the application of this to software based adaptive lifestyle pattern recognition technology. The requirements have resulted in a number of challenges for us to face and find solutions for. We elaborate on these challenges later on.

2 Remote 'Smart Monitoring' for Assisted-Living

Fig 2 depicts the general setup for remote home monitoring system designed to provide assisted-living support. Sensors placed in a property are used to track various aspects of the monitored person's daily activities. Sensor data are gathered by the transmitter installed in the property that is responsible for sending the sensor data to a central station via a permanent Internet connection available in the same property. This transmitter also functions as a security alarm panel protecting the occupants from hazards like intrusion or fire.

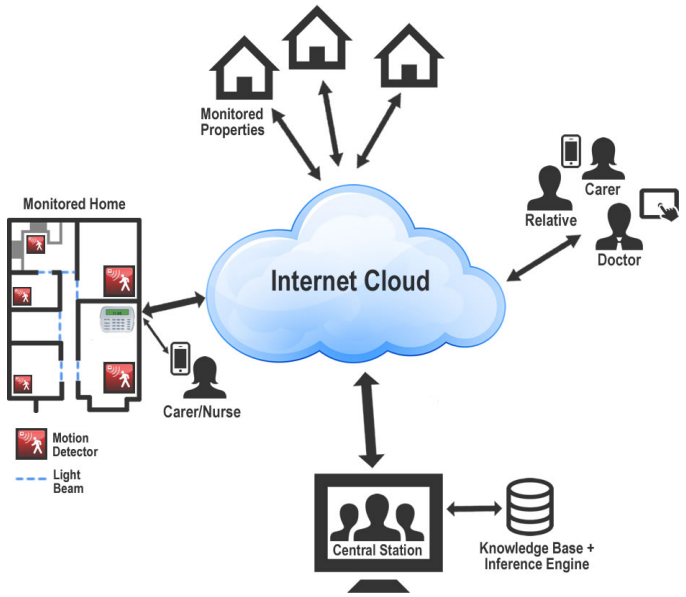


Fig. 2. Outline of remote 'smart' monitoring system

The central station constantly monitors and evaluates the incoming sensor and security alarm data and stores the information in customer database while at the same time analyzing the sensor data in order to detect any anomalies in a person's daily activities and life-style patterns; on detection of such anomalies the central station can alert the nominated person(s). The system is highly scalable and is designed to handle monitoring a large number of properties.

The information gathered can be presented in many different forms that can represent a 'picture' of a person's life-style and highlight any significant anomalies or deviations from the 'norm'. It can also identify any gradual and subtle changes in behavioural patterns that may be indicative of possible onset of an illness, e.g. loss of weight (if measured), increasing visits to toilet, increasing instability in movements, etc. This information can then be made available to interested relatives, carers and doctors via their mobile equipment such as mobile phones and tablet computers.

3 Related Work

Research in assisted living has been an area of rapidly increasing interest involving both the academics and the health industry for some time under various names such as Ambient Assisted Living (AAL), Tele-Health Systems, Smart Home Environments, E-Health Services, etc. and covers a very wide area generally involving assistance to enhance the quality of life of elderly and the disabled. The common theme is based on the acknowledgement of the power of modern computing and communications technologies that can drive and facilitate these systems and schemes. The published number of research is too numerous to review here. However, we can classify the work done under some general groups that distinguish their contributions:

- Sensor networks and activity detection devices: The research work in this group describe sensor networks, sensor devices, use of Internet for sensor data transmissions, wireless sensors and devices that can be remotely controlled for robotic assistance [3, 4, 5].
- Pattern recognition and analysis: Behavioural pattern recognition is an area that is attracting much attention. Advancements in ways of analysing patterns of daily activities are central to assisted living support [6, 7].
- Complete monitoring systems: These are systems that include sensor networks, communications methods and pattern recognition algorithms and form total systems for wide range of assisted living support [8-11].
- Simulation support: Tools to help create innovative products by usability engineers, to model living spaces, movements and locations of monitored persons, to design supporting hardware and software products [12-17].

Our work described in this paper has one important theme in common with the above related work: assisting in the definition and development of safe, efficient, affordable and meaningful assistive technology using targeted modeling and testing of different aspects of this technology. Beyond this our work significantly differs from the work of others and covers areas not covered by other similar work as far as we are aware. For example, we have not come across any simulation models that study and explore specific areas such as sensor data formats and protocols, sensor device selection and placements, local sensor data buffering and efficient transmission to remote central monitoring centers. Most of the work carried out has been theoretical and experimental with several systems proposed and frameworks developed. However, to our

knowledge, there has not been any significant system developed and demonstrated at a commercial level and scale. Therefore we claim in this paper that what sets our work apart from most of the others is that our work has primarily been driven by the desire to realize the results of our research in commercial settings.

4 The Challenges

Our research work was prompted by a need for a new comprehensive and tightly integrated technology within security industry involving 'intelligent' remote monitoring for assisted living support. The challenges faced are multifaceted:

- Collection of multitude of data using a wide range of remote sensors.
- Transmission of sensor data to a remote central station
- Storage, analysis and visual presentation of the collected sensor data.
- Determination and prediction of state-of-health of monitored persons.

We investigated each of the above challenges through preliminary test cases which involved software prototyping and simulations of sensor data collection, transmission and analysis. Below we consider our initial responses to each of the above challenges.

4.1 Collection of Multitude of Data Using a Wide Range of Remote Sensors

In order to be able to provide support for assisted living regular collection of data on location is a necessity. The challenge in this case is to determine what sensor data to gather, what the optimal placements of the sensors are and how frequently the sensor data should be gathered. In order to be able to investigate this challenge a unique software simulator was designed and developed. This simulator served three purposes: a) to investigate sensor requirements and placements, b) to assist in the development of new transmission protocols and c) to serve as a test data generator for the analysis software based at a central monitoring station.

4.2 Transmission of Sensor Data to a Remote Central Station

The challenges here are a) the economy of the format of sensor data packets, i.e. the minimum content required that can be efficiently (in terms of frequency, priority and size) transmitted and stored and b) the storage and pre-processing of data by the local data-logger and transmitter. The sensor data is sent wirelessly across the property to the data logging device which will transmit blocks of sensor data to the central monitoring station at suitable intervals. The transmission method is persistent and will rely on the transmitter's ability to use any one of the progressively lower quality transmission routes: Internet using TCP/IP connection, GPRS, telephone line using DTMF signaling and SMS text messaging.

4.3 Storage, Analysis and Visual Presentation of the Collected Sensor Data

The sensor data received at the remote central station will be stored in the most suitable and efficient manner which should enhance the accuracy and speed up the analysis phase. The central station should be able to receive and store sensor data from many remote locations often at the same time. The main challenges here are a) formatting of sensor data which may be stored in compressed form, b) the real-time analysis of sensor data and c) the presentation of the results of the analysis. The digital receivers at central station will be capable of communicating with many remote digital transmitters using proprietary protocol that will be independent of the chosen transmission route. This protocol will rely on high-level handshaking and error detection/correction methods for maximum reliability. The biggest challenge is the analysis phase. The project proposed and investigated the use of 'Adaptive Life-style Pattern Recognition and Analysis' (ALsPRA) process as the linchpin of the project and the system being investigated. The life-style pattern of a person being monitored is developed over a period of time and is central to the detection of any significant or trending deviations from a baseline, i.e. the most recent 'norm'.

4.4 Determination and Prediction of State-of-Health of Monitored Persons

This part of the investigation looks into the future. It is expected that ALsPRA method will have the potential for short-term predictive capability. This is most appropriate in the case of health and behavioural monitoring. Any deviation from norm over certain period of time may be indicative of impending health problem(s) which if treated on time may save lives or improve quality of life.

5 Simulating Assisted-Living

Our collaborative effort first needed to develop a prototype system in order to a) assist us in our initial feasibility study in exploring the potential of the proposed system including its commercial viability, b) identify any unique supporting features needed in the new generation of digital transmitters currently being developed by our collaborating company partner and c) serve as a demonstrator for the company to attract interest from its existing and future customer base. We therefore resorted to developing our own unique purpose built simulator to support our initial effort.

In order to be able to study the requirements for sensor selection and placements as well as the requirements for the essential communications parameters such as higher level protocols, data formats and frequency of transmissions we designed and implemented a simulator with two-dimensional graphical interface. Fig 3 shows the main screen of the simulator. Using colour-coded design objects the floor plan of a property can be interactively defined. This allows the placement of the walls and the internal doors guided by the grids. Different rooms are identified and furniture such as the chairs and the beds are positioned. Next various sensors are placed in selected locations within the property. For example, there are sensors for detecting entry and exit through the doors using light beams or pressure pads at doorways; there are sensors that can sense pressure exerted in beds, in chairs and on toilet seats. Each sensor is

designated a unique identity number that is used to identify the type of the sensor. The monitored person is identified as a colour-coded solid circle and can move within available spaces along the grids simulating day-to-day activities of the occupant of the property. As the person enters and exits rooms, climbs into and out of the bed and occupies and vacates the toilet seat relevant sensor data are generated and displayed at bottom right of the screen.

The simulated movements of the person being monitored can be manually captured as a series of scenarios in a library of activity scenarios. Various sequences of activities can then be constructed from this library and played back at selectable speeds. This method can be used to generate a large series of sensor data in a relatively short period of time reflecting a person's life-style al be it in a much accelerated manner thus simulating a time period that represents a much longer time in reality. Fig 4 shows the screen used to capture and play back the scenarios.

5.1 The Implementation Details

In this section we describe the design goals and the implementation behind the simulations we developed in order to assist us in the development of the proposed assisted living system based on remote monitoring of the daily activities of persons.

The 2-D graphical simulator is developed in order to mimic the daily living activities of persons in their homes. As a person's activities are simulated virtual sensor data is generated in a manner dictated by the currently defined configuration. In order to define a configuration the necessary steps are the design of the layout of floor plan across the floor grid, the placement of the furniture and the sensors and finally the activation of sensors within this layout. The configuration is manually constructed by using pre-defined color-coded design objects such as wall object, door object, chair object, etc.

The first action is the design of the floor plan using the wall and door design objects; the floor plan defines the configuration of the rooms. Next the furniture is placed in rooms using the design objects such as the bed object and the chair object. Once the floor plan is completed various sensor objects can be 'installed' and 'activated'. The 'installation' of sensor objects implicitly associates them with objects such as beds, chairs, doors, etc. For example, a sensor object associated with a wall object is able to monitor motion, a sensor object associated with a bed or a chair object is able to detect changes in pressure and a sensor associated with a door is able to detect entry and exit through the door. The intention here is to simulate actions such as movements within rooms, persons going to bed, sitting in chairs and entering and exiting rooms. Each sensor object is given a unique id number so that they can be easily differentiated. A color-coded person object is used to simulate movements and other common human activities such as lying in bed, sitting in a chair or on the toilet and going in and out of rooms. For example as the person object enters the floor grid occupied by the bed object this is regarded as the person going to bed and as the person exits this grid this is regarded as the person getting out of bed. Each of such actions will generate unique sensor data making identification of actions possible. This way the simulator enabled us to study the requirements for sensor type selection and placement options as well as the requirements for the essential communications parameters such as higher level protocols and data formats.

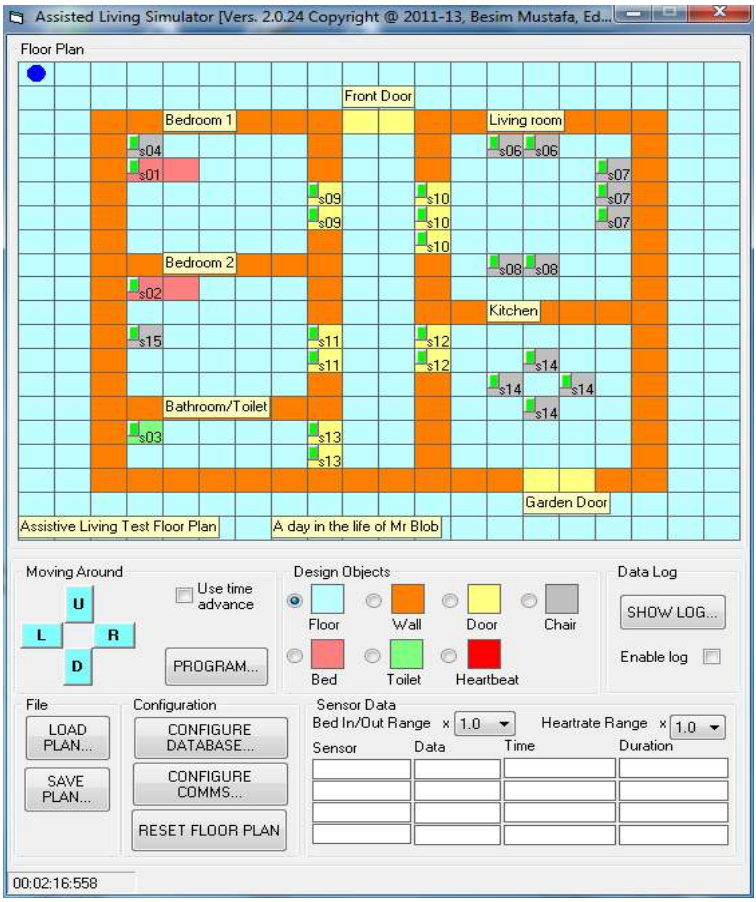


Fig. 3. Main simulator window

A feature of our simulator is its ability to capture and store the simulations of a person's activities in a library of 'activity scenarios' and to play back these scenarios at different speeds, frequency, order and combinations. Fig 4 shows the library containing sample list of scenarios. On the left is the list of captured activity scenarios and on the right is the list of scenarios selected to be played back in the order listed. The sensor data are re-generated while the captured actions are replayed. The sensor data is used to establish a person's lifestyle pattern over time as the 'norm'. The simulator can then be instructed to generate sensor data with statistically variable degrees of deviations from the established 'norm' in order to simulate exception conditions that should be detected by the sensor data analysis software thus testing and validating detection and analysis algorithms without the availability of real person subjects and physical locations.

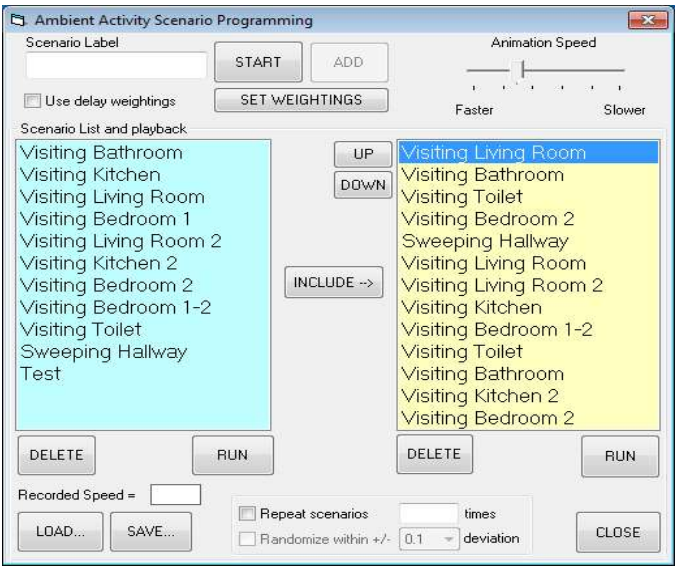


Fig. 4. Life-style scenario capture and playback showing list of scenarios

Fig 5 shows simulator's log of sample sensor data. The log records include sensor identities, activity events, activity start and end times as well as the durations of the activities thus capturing the three important parameters on which the lifestyle patterns are established. The simulator's behaviour is in fact indistinguishable from the physical transmitter/data-logger device as far as the pattern analysis software is concerned; the pattern analysis software simply receives the sensor data and is not concerned with the source of the data. Therefore the simulator can play the important and central role of helping identify any new features required in transmitter hardware and firmware in order to provide unique specialist support for assisted living at the same time as providing large amounts of realistic sensor data for the pattern analysis software testing thus making possible parallel developments of these two key areas thus speeding up the development process. The captured data can also be used as off-line test data for the analysis software. The simulator enabled us to experiment with the format of the sensor data, the storage requirements at the local transmitter and the manner in which this data is pre-processed and transmitted to the central station (frequency, size, etc.).

As the sensors are potentially capable of generating a large number of data, e.g. motion sensor data and door entry/exit data, we envisage a mechanism whereby a local buffer memory will be reserved to temporarily store blocks of non-urgent, i.e. low priority, sensor data. In order to minimise communications bandwidth requirement the buffered sensor data will be sent to the central monitoring station under certain conditions, e.g. when buffer full or at pre-configured intervals. The simulator is designed to include features to enable us to study the requirements for the size of sensor data buffer responsible for temporarily holding sensor data and to explore conditions under which the buffered data is to be sent to the central station. The simulator

therefore enabled us to experiment with the optimum buffer size requirements and to explore different configurations for transmission intervals, e.g. after every N number of sensor events or after every T minutes, etc. Provisions for these facilities can be seen in Fig 5 at the bottom of the window.

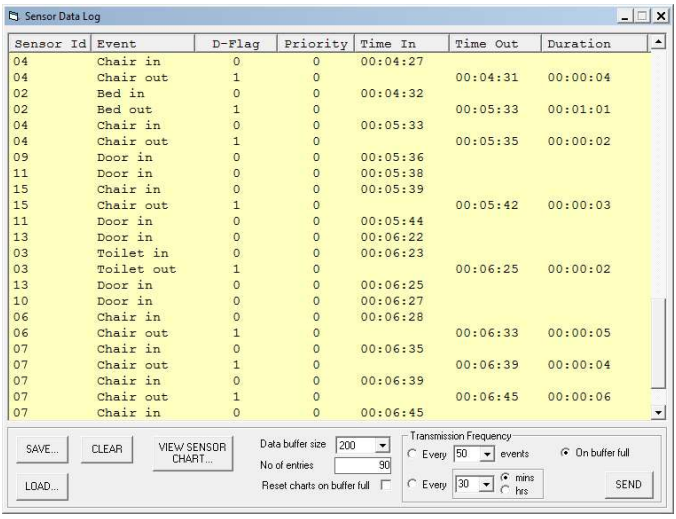


Fig. 5. Sensor data log showing a list of sensor detections

The sensor detected data can provide basic information used to construct a 'view' of a person's way of life that is characterized by three specific attributes making pattern recognition and analysis possible: frequency of events, time of events and duration of events. For example, the frequency of events can be attributed to the number of times a person visits the toilet; the time of an event can be attributed to the time a person goes to bed; the duration of an event can be attributed to the time a person spends in bed. All this information forms part of a person's lifestyle pattern in which there may be some embedded activity data that is indicative of possible underlying health related problems that have been gradually developing over the time. The simulator can generate this kind of data relatively easily and rapidly with suitably modulated temporal information for more realistic data. Fig 6 shows the simulated sensor data patterns represented by the two charts where the frequency of events detected (y-axis) is plotted as a line graph against each of the 'installed' sensors (x-axis) and the duration of certain events (y-axis) is also plotted as a bar chart against each of the installed sensors (x-axis) relevant to those events. For example, the events to do with going to bed and getting out of bed will be associated with the durations in bed, i.e. how long a person spends in bed presumably sleeping or resting; associating events such as entering and exiting rooms with durations can be less meaningful in certain cases depending on the context, i.e. time, location and any associations with other events, of the activity that is taking place. The area under the activity frequency line graph represents one aspect or 'dimension' of a person's daily activity pattern; the

duration bar chart representing another; a third 'dimension' is the representation of the activity start times not shown here.

The buffering of the sensor data prior to transmission gives us the possibility to do initial pre-processing of the sensor data. The data logger incorporates moderately powerful processing capability as described in section 8 making pre-processing possible. For example, motion detectors are likely to send their data much more frequently than any other type of sensors. This will therefore generate large amounts of sensor data all of which may neither be sensible nor necessary to transmit to the central station. It may then be possible for the data logger software to identify transitions from room to room, i.e. when a person leaves one room and enters another where there is a motion detector is fitted. This will make possible the detection of 'in and out of room' events. It can also make possible estimation of the speed of movements of the monitored persons as they move about in their homes giving some indication of their degree of mobility and may even, together with some other relevant data, contribute to the prediction of their state of health in some cases. The simulator is designed to pre-process this data giving us the ability to experiment with various algorithms for pre-processing the sensor data prior to transmission. These algorithms can then be implemented in the real data logger's firmware.

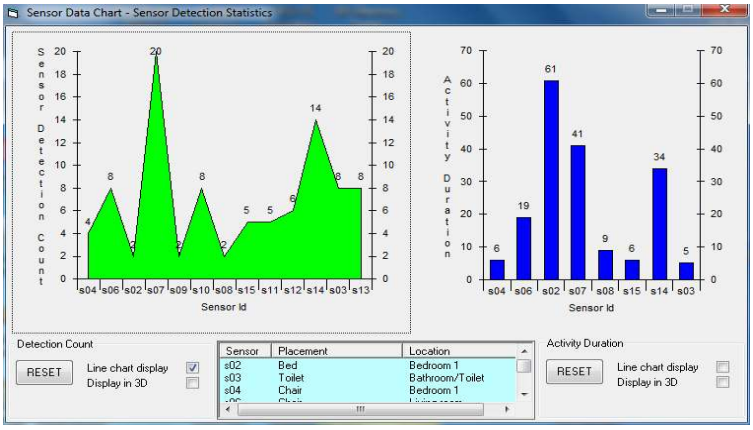


Fig. 6. Sensor data statistics: detection counts and detection durations.

The sensor detection count, the event start time and the event duration together constitute the monitored person's 3D lifestyle pattern. It is this 3D information that the inference engine (described in section 9) processes. Looking at Fig 6 we can get glimpses of the monitored person's lifestyle. For example, from the sensor detection frequency graph that the event detected by sensor 2 (going to bed) does not occur frequently; from the corresponding activity duration chart we can see that this activity is of the longest duration which is expected. On the other hand the event detected by sensor 7 (sitting in chair) does occur frequently and the duration of this occurrence is moderately high. From this information we can deduce that this person appears to sleep well and that spends a lot of time sitting which are indications of little activity.

6 Initial Results and Findings

Using the simulator we have been able to experiment and conduct our initial investigations into the various aspects of sensor data generation and transmission that included

- Type and placement of sensors
- Definition of sensor data formats
- Specification of sensor data transmission protocol
- Determination of optimum frequency of sensor data
- We elaborate further on the above in the following sub-sections.

6.1 Definition of Sensor Data Formats

Each sensor data gathered will be formatted in such a way that it conveys all the essential information to the analysis software in the most economical manner as potentially large number of sensor data can be generated in a relatively short period of time. We investigated properties of sensor data and the following represent some of the essential components of sensor data

- Sensor id: used to identify individual sensors.
- Time stamp(s): can be multiple time stamps, e.g. bed-in time and bed-out time or toilet-in time and toilet-out time.
- Priority indicator: determines the urgency of sensor data, e.g. fall detected or raised blood pressure, etc.

The sensor id and time-stamps are needed in order to be able to track the matching activities of the individual sensors. For example, a sensor can be responsible for detecting the person going to bed and the same sensor detecting the person coming out of the bed. These sensor data can be used to determine three important pieces of information: the fact that the person went to bed (presumably for sleeping), the time the person spent in bed, the fact that the person came out of the bed (presumably woken up from sleep). From these several inferences can be made: if the person sleeps; if the person sleeps long enough; if the person is able to come out of bed. Still further inferences can be made from this information such as the person is or is not sleeping, or the person is or is not sleeping well or the person is unable or unwilling to come out of bed which in turn can be further connected to the ambient temperature of the bedroom or to the count of visits to the kitchen and the toilet; it may be that an elderly person is unable to come out of bed due to lack of heating in the bedroom or it may be that the person is possibly too weak to climb out of the bed and walk. The sensor data can be used to determine the frequency of activities such as how often a person visits the toilet or the kitchen that can be used to infer any potential health problems (i.e. not eating well or regularly or changing toilet habit indicated).

The sensor data priority is required to determine if the sensor data needs to be immediately transmitted to the central monitoring station or that it can be delayed in order to minimise communications traffic when several sensor data can be

collectively transmitted at pre-determined intervals as a block. For example, if an elderly or a disabled person has fallen the data sent by the accelerometer sensor needs to be urgently transmitted so that help can be provided for the fallen person as fast as possible; the vibrations detected in a child's cot by a sensor possibly due to an epileptic episode needs to be acted upon as soon as possible. On the other hand sensor data on going to bed and coming out of bed do not need to be sent immediately but can be stored for later transmission. The priority is implied by the type of sensor as identified by its id or channel number and is facilitated by a look-up table maintained for this purpose.

6.2 Specification of Sensor Data Transmission Protocol

One of the aims of this investigation has been about identifying any new supporting features that are needed to be designed into Securecom's new breed of advanced commercial digital transmitters in order to provide efficient support for the assisted living system we are proposing. These advanced transmitters are designed specifically for remote monitoring of properties for security purposes. As a result of our collaborative work we have been able to identify and specify an extension to the existing protocol used for providing assisted living support at an early stage of product development. We describe the digital transmitter further later on.

The transmission protocol the digital transmitter uses is based on the standard used by the security industry. In order to be more flexible Securecom decided to define and use an extended proprietary version which is not publicly made available. Due to non-disclosure agreement we are unable to reveal this information in detail. However, it suffices to say that the basic elements include a means of identifying the property, the identification of the alarm trigger condition(s) and the device status. Depending on the mode of transmission each data transmitted may require acknowledgement along the same path of connection, e.g. acknowledgement packets on Internet connection (TCP/IP) or audio frequency signalling on land-line and mobile audio (GSM) connections.

6.3 Determination of Optimum Frequency of Sensor Data Generation

Monitoring persons' life styles in terms of their daily activities will require handling of a relatively large number of sensor data and the number generated in a single day can be in hundreds in the case of an active person. The sensor data generated can provide three types of information: 1) frequency of activities, 2) times of activities and 3) durations of activities. These can be used to make inferences such as that an activity is taking place (e.g. moving, sleeping, eating, going to toilet, etc.), that an activity is taking place at expected or during normal times (e.g. having breakfast in the morning, taking the pill in the morning, going to toilet at night, etc.) and that an activity is taking as long as expected or normal (e.g. sleeping around six hours, spending no more than ten minutes in the toilet, staying in kitchen for at least half an hour, etc.). The words expected and normal are used in relative sense and can be interpreted

differently for different people; what is expected and what is normal will be established on a per person basis after a period of monitoring.

The challenge here is the determination of the number of activities to monitor and how often to monitor. It is quite possible that not all activities will be necessary to monitor and monitoring only the targeted activities that reflect a person's particular nature of disability or age will be meaningful. Nevertheless monitoring of the daily activities of thousands of individuals can put a large demand on the communications devices and networks.

7 Sensors, Sensors and Biometrics

The proposed system heavily relies on activity sensors in order to be able to gather information on persons' life styles. Therefore it is important that the correct sensor technology is used and that the sensors are optimally positioned and configured.

The system proposed can be configured to use any number of sensors although this will be restricted in reality to a few strategically placed sensors. The sensors used can be off-the-shelf inexpensive devices. However, for practical reasons we propose to use sensors with wireless capabilities. For this reason the digital transmitter we use is able to handle wireless sensors as we are adopting a system that already accepts wireless detection devices for security alarms, e.g. fire, intrusion, etc. Table I lists types of typical low cost sensors that can be used in order to help establish persons' life style patterns over time. The table includes information on types of data the sensors can provide and whether the frequency and durations of data are relevant or not.

The sensor triggers will be detected by the digital transmitter using the standard wireless communications protocol. This is likely to be similar to or same as the ZigBee communications protocol offering low power requirements, good range, low price and low susceptibility to interference [21]. Although ZigBee has low data rate (20 to 250 Kbps) this is nevertheless more than adequate for short bursts of sensor data transmissions. It is decided to use the lower frequency band of 868 MHz for sensor transmissions. This band offers better penetration characteristics through brick walls as most of the older properties tend to be brick built inside. This frequency band is restricted to a single channel necessitating the transmission of sensor id numbers; each sensor is given a unique id number for this purpose. The id will then be transmitted to the local digital transmitter/data logger. The low priority events will be logged by the transmitter in its buffer in 5-byte blocks of information: 7- bit sensor id, 1-bit sequence number and 4-byte time stamp. This format will allow a maximum of 128 sensors; the sequence bit will be used to identify two-state matching events from the same sensor (e.g. in and out, on and off, etc.) and time stamp will be used to indicate the time of occurrence of the event. If the event is of high priority then this will be immediately transmitted to the central monitoring station in the same format as above without being buffered first. The buffer size will be determined by the frequency with which the buffered data will be transmitted and can be configurable depending on the circumstances of the monitored subject.

Table 1. Attributes of some common sensors

Sensor	Priority	Placement	Information	Duration	Frequency
Motion detector	Low	Rooms, other spaces in property	Motion, paths of movements	Not relevant	Not relevant
Pressure pad	Low	Doorways, beds, chairs, toilet seats	In and out of rooms, on and off chairs and toilet seats, in and out of beds	May be relevant	May be relevant
Light beam	Low	Doorways (may require two to detect direction)	In and out of rooms or property	May be relevant	May be relevant
Accelerometer	High	Worn on person	Orientation, degree of instability in movements	Not relevant	Relevant
Thermometer	Low	Rooms, outside the property	Temperature (above or below threshold)	Not relevant	May be relevant

Although the sensors in Table 1 are capable of providing data from which current state of health can be indirectly inferred to a certain extent they are nevertheless not able to directly provide biometric data that can be further relied upon for real-time supportive evidence of state of health. For example blood-pressure sensor data can be used in conjunction with the data from accelerometer sensor in order to reasonably conclude that a fainting episode is a strong possibility and together with the spatial and the temporal information gathered over time the subject's future state of health can be projected; data from pulse rate sensor, data from blood-pressure sensor, data from accelerometer sensor and a microphone can be combined to infer a possible onset of a stroke demanding immediate attention and more aggressive follow-up monitoring. One of the uses of biometric information is that it can be used to authenticate the monitored subjects through 'liveness' detection [18, 19].

By taking full advantage of biometric data capture, both authentication and medical data can be captured and correlated within a system in a secure environment [20]. As with all system implementations involving persons and health data the security concern is always present, and biometric devices are no exception. There are a number of biometric security concerns that must be addressed, especially if the systems are going to contain data about vulnerable users such as those our system is designed for the benefit of. We are closely allaying with the research work of one of the authors of this paper on the definition of a security related framework on biometrics which we believe our work can benefit from [23]. There are various security related issues that need tackling; use of encryption is particularly useful in resolving few of these issues.

We envisage primarily two ways of collecting biometric data: 1) automatically using wearable biometric devices and sensors, 2) by a visiting nurse or a carer using mobile devices. The former requires that the monitored person is semi-permanently connected to biometric devices such as blood pressure and pulse rate monitors; the latter is carried out by a nurse or a carer during regular visits who manually enters the data in the mobile device they carry which then gets transmitted to the local digital transmitter at the end of the visit.

8 Digital Transmitter and Remote Data Logger

The digital transmitter we have been working with belongs to a new range of transmitters designed by Securecom specifically for remote monitoring of properties for security purposes. At its heart is the processing power based on a 32-bit power-efficient ARM Cortex-M4 core technology. These microcontrollers are popular high-performance choice in low-power constrained and cost-sensitive signal processing devices offering Ethernet connectivity.

The digital transmitter is designed to be located at the property to be monitored. It is actually integrated into the alarm panel and is responsible for transmitting alarm data whenever an alarm condition such as intrusion, fire or panic button is triggered. Each property is identified by a unique site code which is also transmitted to a central monitoring station capable of monitoring hundreds of properties fitted with the same transmitter. Fig 7 shows the general components of the digital transmitter. The alarm sensors communicate with the transmitter using a wireless protocol on 868 MHz band using a single channel and employs a contention based protocol where if a sensor needs to send data it establishes a connection with the transmitter and all other sensors are prevented from transmitting their data while this connection is present. The transmitter acknowledges all sensor data. Once the current connection is removed, the transmitter polls all other sensors for data. When no more sensor data is available both the transmitter and the sensors go into sleep mode. This method affords low duty cycle for the sensors thus minimizing their power requirement.

The alarm data is normally transmitted to the central monitoring station immediately using a proprietary format conveying enough information for the central station to act upon in consultation with the user data stored in a customer database. The transmission uses a persistent mode of establishing connection depending on how it is configured. This mode has a sequence of priorities for establishing the transmission path and in order of preference these are: TCP/IP connection via Ethernet interface, GPRS connection via 3G/4G mobile data services, GSM audio with DTMF signaling and finally SMS. In all but the SMS method acknowledgments are possible.

The digital transmitter is capable of supporting enhanced system integrity and reliability using three main methods: 1) embedded MCU watchdog mechanism, 2) application level error detection and correction, and 3) periodic or on demand test signaling. The watchdog method is a standard mechanism for resetting MCU whenever it fails to respond to independent watchdog commands. This method attempts to mitigate mainly obscure embedded software related problems. Test signaling can be done at two levels: central station sends test data to remote digital transmitter and expects an acknowledgement; the digital transmitter sends each remote sensor test data and expects acknowledgements. This can either be done regularly, say once a day or it can be done at times when sensor activity is deemed to be lower than expected.

Working in collaboration with Securecom we have experimented with leveraging the existing remote monitoring and communications capabilities of their digital transmitter in order to facilitate our requirements for assisted-living support. We have managed to use the data from our simulations in order to be able to identify additional requirements that can be integrated with the current design of the transmitter in its

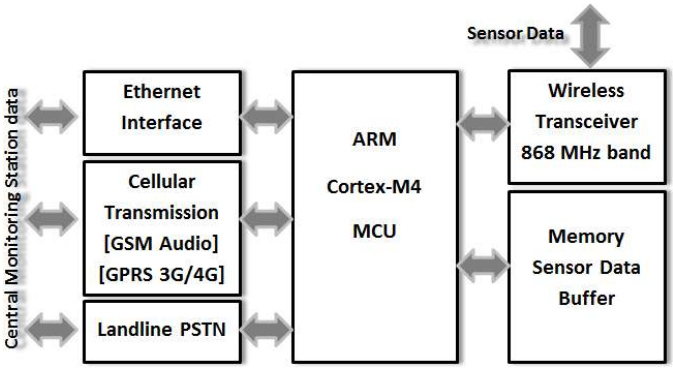


Fig. 7. Digital transmitter block diagram

final stages of development. The following new provisions have been identified for this reason:

- Changes to the proprietary protocol to make allowance for the assisted-living mode of operation
- Changes to transmitter firmware to handle assisted-living protocol requirements
- Changes to transmitter firmware to implement sensor data buffering/logging and periodic transmission of this data

9 Life-style Pattern Recognition and Analysis

The central station software is responsible for various activities that are required to support remote "smart" monitoring and responding on demand to the information it receives. This software is therefore required to offer a range of services; we identified the following essential services:

- Data capture
- Data storage
- Data transformation
- Data analysis
- Identification of exceptions
- Data presentation

We opted for a rule based inference and decision making method of accumulating knowledge and analyzing day-to-day living patterns of persons generated over a period of time. Fig 8 shows an outline of this method. Sensor data are collected in the sensor database in a format that is more compact and amenable to efficient processing than the raw data received. The knowledge base contains sets of rules that are used to facilitate pattern recognition and determination of exception conditions and whether any action should be taken. Initially the basic rules are provided by the domain experts. However some of the sensor data gathered can be transformed into additional

sets of rules thus enabling the knowledge base to adapt to the changing living conditions over time. The inference and decision making engine is fed from both the knowledge base and the sensor database. The results of the process of inferring can be fed back to the knowledge base in order to facilitate evolution of existing and adaptation of new rules over time. Similarly a path from the inference engine to the sensor database can be used to assert, retract or modify sensor data. For example, initially, during the learning period, a monitored aspect could be based on the assertion of the rule "for person X it is normal if he goes to bed and comes out of bed within 24 hours". As time passes this rule will adapt to reflect more precisely the reality and may gradually evolve into the new rule "for person X it is normal if he spends between 5 and 8 hours in bed within 24 hours"; a similar rule for person Y may be based on a different set of personal facts. Therefore the knowledge base will accumulate expertise on life-styles of each of the monitored individuals by adaptively generating sets of rules that closely reflect their way of life.

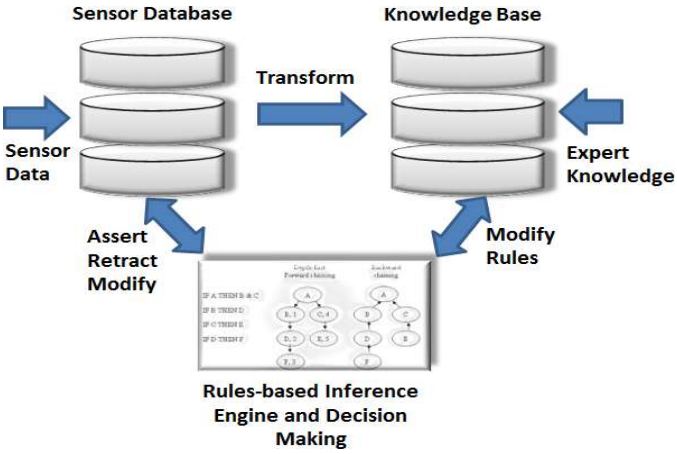


Fig. 8. Adaptive Life-style Pattern Recognition and Analysis System

Over a period of time the monitored sensor data and the knowledge base for each assisted person will grow in size. In order to manage the large amounts of information accumulated it will be necessary to structure the data in a way that makes the time-consuming process of inferring efficient. As the data stored increases the life-style patterns will mature and facilitate increasingly more accurate determination of exception conditions and state-of-health predictions. Fig 9 illustrates a simplified view of data maturity and structuring by base-lining activity 'norms' in order to promote near real-time pattern recognition and analysis. So, instead of ploughing through large amounts of historical data every time sensor data is received the structured base-lining method will enable the inference engine take into account summary of historical data in its reasoning at the same time as considering the most recent data.

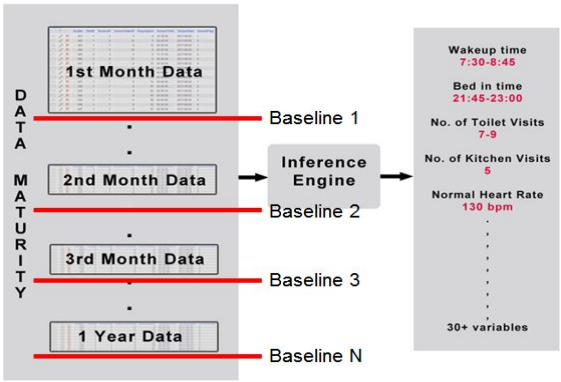


Fig. 9. Sensor data and knowledge base maturity

As always the reality is not as simple as we would like it to be and there are several practical considerations we need to take into account when making inferences if our system is to be workable in practice. We look at three such areas:

- Multiple occupancy
- Accuracy of inferences
- Security of information

9.1 Multiple Occupancy

The proposed system is optimized to work with single occupants in properties. This however presents a dilemma whenever there is more than a single living being in the property. For example, an elderly occupant may be visited by relatives or by a carer; the occupant may have a dog for company. We can identify and authenticate the monitored person or alternatively we can regard the visitors and pets as part of the monitored persons' way of life and hence also part of their life-style pattern. We opted for the latter for two reasons: 1) simpler, cheaper and easier to implement, 2) unusual visits and pet behaviours can also be indicative of exception conditions worth investigating.

9.2 Accuracy of Inferences

In order to enhance the accuracy of decision making the inferences can be based on contextualized sets of actions that take into consideration temporal and spatial indicators. For example, the action of rapidly lowering into a chair needs to be distinguished from falling down. If this takes place in the bathroom or in the hall (as sensed by the motion detectors) this may be construed as a valid fall; if in the living room or bedroom and is followed by pressure sensors triggering then this may indicate actions of sitting in a chair or lying on a bed as opposed to falling down.

We are aware of the potential drawbacks of the proposed rules based inference engine [22]. However we have chosen it as it is relatively easy to implement and is well understood.

9.3 Security of Information

Security of information has two dimensions: 1) snooping and 2) falsifying. Security is particularly important when biometric data is being gathered. Both of these security concerns can be significantly minimized by encrypting both the sensor data received from the sensors and the data the digital transmitter sends to the central station.

10 Prototyping and Results

In order to test our ideas we resolved to implement a proof-of-concept prototype system. The sensor data was provided by both manual means and from the output of the assisted living simulation scenarios as explained before. We used the manual means in order to inject test cases that simulated abnormal data that deliberately and significantly deviated from norms; we used the simulator output to provide rapid succession of sensor data in order to speedily establish life-style norms. A TCP/IP connection was used in order to establish secure communications between the simulator and the simulated central station monitoring software. In the prototype system we did not implement an adaptive rules-based knowledge base; instead we used relatively simple algorithmic means of processing the sensor data. A web-based graphical user interface was also implemented to communicate with the user and to present sensor data; MySQL was used to create access and manage the sensor knowledge database.

Fig 10 shows two views of the web-based prototype user interface. The background view represents the logging screen for central station personnel as well as for the carers and the family members of the persons being monitored; it is envisaged that different types of observers will be presented with information in formats most appropriate for the type of observer. The foreground view displays the colour-coded graphical representation of sensor data gathered from the daily activities of a person being monitored; normal data (green), data needing close watch (yellow) and data that needs to be acted upon urgently (red). The foreground image in Fig 10 depicts an example visual status of four monitored persons. Each person is identified by a unique site id assigned at the time of the installation of the system in their home. The monitored activities and other related parameters are shown at the top of the image. In this example persons 1 and 2 are showing daily activities within their normal ranges. However, persons 3 and 4 are exhibiting exception conditions highlighted in red with person 3's daily kitchen visits below his established 'norm' and person 4's daily toilet visits below her established 'norm'. Another exception condition for person 4 indicates a significant deviation from her daily routine with respect to her use of the main door, for example she may have exited her home or left her door open at an unusual time of the day. This daily activity information can be made available on daily,

weekly or monthly bases in various graphical formats designed to enhance visual impact in the identification of possible exception conditions. This information will also be remotely accessible via mobile devices that will present the information in formats relevant and suitable to the type of users, e.g. doctor, carer or family member.

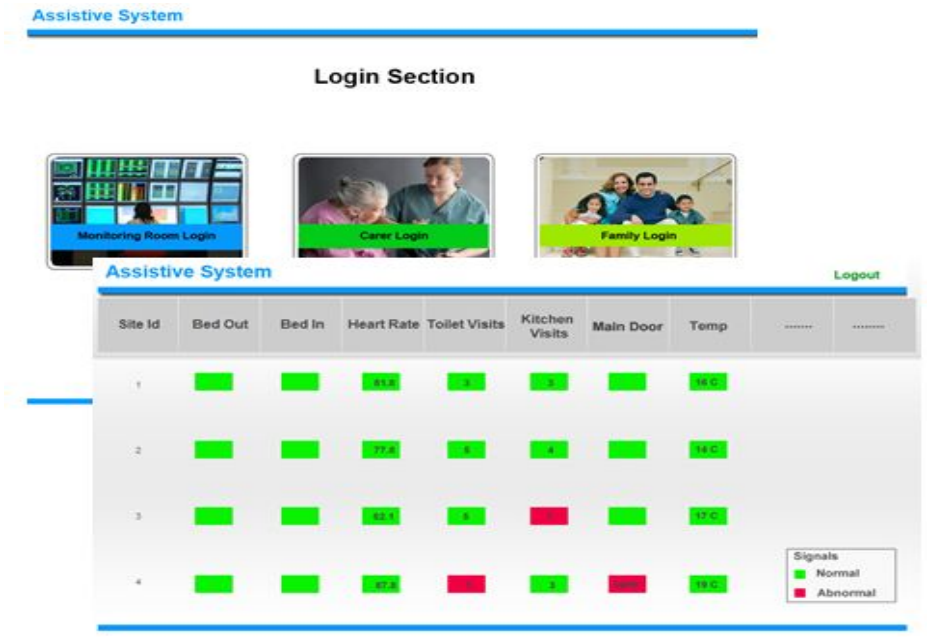


Fig. 10. Central station assisted living support interface

An important reason for developing the prototype was for Securecom to use it as a proof-of-concept to a large housing authority as a potential customer base with a vested interest in the technology. This demonstration was successful and suitably impressed the senior staff of this authority to such an extent that they expressed keen interest in further development work investigating predictive technology where the state of health of persons can be predicted into the future. This is seen to be desirable for facilitating efficient planning and funding of public services such as health care for elderly and disabled. At times of sweeping government cuts in spending on public services this technology is suddenly a welcome prospect.

The full implementation of the monitoring software will be integrated with Securecom's existing central station system normally used to handle monitored alarm incidents. Once this implementation is stable the next stage will be to install sensor devices in several selected properties in order to further evaluate and fine tune it over a period of time before commercially rolling it out. For a larger scale validation we intend to seek the assistance of housing authorities responsible for large numbers of care homes.

11 Looking to the Future

We continue working on the remotely monitored assisted living system described in this paper in collaboration with Securecom Ltd. and expect it to become a commercially viable product in the near future. There is still some work to complete especially in the development of efficient algorithms for life-style pattern recognition and algorithms for inference engine and decision-making that can monitor the daily activities and accurately assess significant deviations from what is regarded as norms as well as detect subtle trends of changes that can point to any future health related problems needing attention sooner than later.

Another area we would like to concentrate on in the future concerns the ability to accurately predict any future health related problems of the monitored subjects. This will require the assistance of biometric devices; our partner company is in the process of developing such devices. The intention here is to be able to use the sensor information from the biometric devices together with the information gathered from other sensors in order to make reasonably accurate extrapolations into the future state of health of the monitored individuals. This process will also help identify gradual degradation in the mobility and health of persons much faster than a human observer can. Local authorities can use the information of this kind in order to better plan and target for future support and funding requirements. Also a more timely medical intervention may be administered enhancing quality of life and improving life expectancy.

12 Conclusions

In this paper we considered a wide spectrum of assisted living issues and technologies. We described our initial and promising investigations into smart remote monitoring using a range of sensors and ubiquitous communications methods for the benefit of elderly and disabled who wish to maintain their independence and dignity as long as possible often living alone. We then went on further and proposed a scalable system that can recognise daily activities as life-style patterns for establishing norms over time and that uses rule-based adaptive knowledge base method in order to detect any alarming deviations from these norms. Working with a commercial partner we aim to help realise the proposed system as a commercial concern thus fulfilling our purpose to support the aging populations. There are many productive and promising research activities in this area and we are hoping that our work will make a modest contribution.

References

1. Khaw, K.: How many, how old, how soon? British Medical Journal 319, 1350–1352 (1999)
2. Age UK factsheet, 2013. Later Life in the UK. Age UK, p. 9 (October 9, 2013)

3. Dohr, A., Modre-Ospiran, R., Drobics, M., Hayn, D., Schreier, G.: The Internet of Things for Ambient Assisted Living. In: Seventh International Conference on Information Technology, Graz, Austria, April 12-14 (2010)
4. Cavallo, F., Aquilano, M., Odetti, L., Arvati, M., Carrozza, M.C.: A first step towards a pervasive and smart ZigBee sensor system for assistance and rehabilitation. In: IEEE 11th International Conference on Rehabilitation Robotics, Kyoto, Japan, June 23-26 (2009)
5. Ando, B., Baglio, S., La Malfa, S., Pistorio, A., Trigona, C.: A Smart Wireless Sensor Network for AAL. In: IEEE International Workshop on Measurements and Networking Proceedings (M&N), Catania, Italy, October 10-11 (2011)
6. Virone, G., Alwan, M., Dalal, S., Kell, S.W., Turner, B., et al.: Behavioural Patterns of Older Adults in Assisted Living. IEEE Transactions on Information Technology in Biomedicine 12(3) (May 2008)
7. Lozano-Tello, A., Botón-Fernández, V.: Analysis of Sequential Events for the Recognition of Human Behavior Patterns in Home Automation Systems. In: Omatu, S., Paz Santana, J.F., González, S.R., Molina, J.M., Bernardos, A.M., Rodríguez, J.M.C. (eds.) Distributed Computing and Artificial Intelligence. AISC, vol. 151, pp. 511–518. Springer, Heidelberg (2012)
8. Chiriac, S., Saurer, B.R., Stummer, G., Kunze, C.: Introducing a low-cost Ambient Monitoring System for Activity Recognition. In: 5th International Conference on Pervasive Computing Technologies for Healthcare, Karlsruhe, Germany, May 23-26 (2011)
9. Pang, G.K.H.: Health Monitoring of Elderley in Independent and Assisted Living. In: International Conference on Biomedical Engineering, Penang, Maleysia, February 27-28 (2012)
10. Carni, D.L., Fortino, G., Gravina, R., et al.: Continuous, Real-time Monitoring of Assisted Livings through Wireless Body Sensor Networks. In: 6th IEEE Internatiuonal Conference on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications, Prague, Czech Republic, September 15-17 (2011)
11. Rodrigues, A.L.B., Gomes, I.C., Bezerra, L.N., et al.: Using Discovery and Monitoring Services to Support Context-Aware Remote Assisted Living Applications. In: International Conference on Computational Science and Engineering, CSE 2009, Rio de Janeiro, Brazil, August 29-31 (2009)
12. Fernandez-Llata, C., Mocholi, J.B., Sala, P., et al.: Ambient assisted living spaces validation by service and devices simulation. In: 33rd Annual International Conference of the IEEE EMBS, Boston, USA, August 30-September 3 (2011)
13. Garcia-Valverde, T., Campuzano, F., Serrano, E., BotiaHuman, J.A.: Behaviours simulation in ubiquitous computing environments. In: Proceedings of the Multi-Agent Logics, Languages, and Organisations Federated Workshops, MALLOW 2010, Lyon, France, August 30-September 2 (2010)
14. Velasquez, C., Soares, C., Morla, R., et al.: A 3D simulation framework for safe ambient-assisted home care. In: The Fifth International Conference on Mobile Ubiquitous Computing, Systems, Services and Technologies, Lisbon, Portugal, November 20-25 (2011)
15. Laue, T., Stahl, C.: Modeling and simulating ambient assisted living environments – A case study. In: Augusto, J.C., Corchado, J.M., Novais, P., Analide, C. (eds.) ISAmI 2010. AISC, vol. 72, pp. 217–220. Springer, Heidelberg (2010)
16. Zhang, S., McCullagh, P., Nugent, C., Zheng, H., Black, N.: An ontological approach for context-aware reminders in assisted living' behavior simulation. In: Cabestany, J., Rojas, I., Joya, G. (eds.) IWANN 2011, Part II. LNCS, vol. 6692, pp. 677–684. Springer, Heidelberg (2011)

17. Bowling, A., Makedon, Z.L.: SAL: a simulation and analysis tool for assistive living applications. In: Proceedings of the 2nd International Conference on Pervasive Technologies Related to Assistive Environments, Article No. 4. ACM Press, New York (2009)
18. Toth, B.: Biometric Liveness Detection. Infomation Security Bulletin 10, 291–297 (2005)
19. Pan, G., Wu, Z., Sun, L.: Liveness Detection for Face Recognition. In: Delac, K., Grgic, M., Stewart Bartlett, M. (eds.) Recent Advances in Face Recognition, pp. 109–124. InTech (2008)
20. Gafurov, D., Snekkenes, E., Bours, P.: Spoof Attacks on Gait Authentication System. IEEE Transactions on Information Forensics and Security 2(3), 491–502 (2007)
21. Farahani, S.: ZigBee Wireless Networks and Transceivers. Newnes (2008)
22. Cosgrove, S.J.: n Reasons Why Production-Rules are Insufficient Models for Expert System Knowledge Representation Schemes. In: Electronics Research Laboratory, Information Technology Division, Research Report ERL-0520-RR, Salisbury, South Australia (1990)
23. Matthew, P.: Autonomous synergy with biometric security and liveness detection. In: Science and Information Conference (SAI 2013), London, UK, October 7-9 (2013)

All Weather Human Detection Using Neuromorphic Visual Processing

Woo-Sup Han[1] and Il-Song Han[2]

[1] Imperial College London, London, UK
{w.han13@imperial.ac.uk}
[2] Korea Advanced Institute of Science and Technology, Daejon, Korea
{i.s.han@kaist.ac.kr}

Abstract. There have been many researches on computer vision for the purpose of vehicle safety applications, making use of diverse methodologies like complex vision algorithms. However, despite its effectiveness, computer vision algorithms sometimes lack the robustness of mammalian visual system for the application in dynamic environments in vehicle driving conditions. We propose that the neuromorphic visual information processing offers an alternative which mimics the robustness and flexibility of the primary visual cortex, based on Hubel and Wiesel's experimentation and Hodgkin-Huxley formalism. The proposed neuromorphic approach is implementable in CMOS VLSI, where the robustness is demonstrated by MATLAB simulation with the real world data sets. The neuromorphic vision based on the orientation selective neuron demonstrated the successful object detection of car license plate, car passenger at accident, or pedestrian on/or off the road. The effectiveness of proposed neuromorphic visual processing system is evaluated for the Electric Vehicle's safety enhancement via the 95% of pedestrian detection rate for electric vehicle's virtual engine sound system, while its applicability or robustness is shown for the case of stereo sensors or the detection of bike rider at the dark and wet weather.

Keywords: Neuromorphic Vision, Pedestrian Detection, ADAS, Stereo Processing, Visual Cortex, Hodgkin-Huxley Formalism.

1 Introduction

Since the experiment of Hubel and Wiesel which investigated mammalian visual cortex, there had been a number of researches carried out on understanding the relationship between the brain and visual perception. Neuromorphic models of visual information processing, inspired by Hubel and Wiesel's experiment and the neurophysiological model of Hodgkin-Huxley formalism, looks to imitate mammalian visual cortex.

One of the possible applications of neuromorphic vision system is in enhancing the safety of automobile vehicle through mimicking the robust and natural visual information processing. In this paper, we will look at how the neuromorphic visual system can be developed for application of the pedestrian detection for the vehicle on the road. The feasibility of this system is based on the successful demonstration of these objectives through the neuromorphic VLSI neuron of visual cortex. The wide application area of

L. Chen, S. Kapoor, and R. Bhatia (eds.), *Intelligent Systems for Science and Information*, Studies in Computational Intelligence 542,
DOI: 10.1007/978-3-319-04702-7_2,

the neuromorphic visual processing is introduced by the stereo vision of distance-based object detection and the detection in the dark and wet weather.

2 Background

While the visual signal environment of pedestrian detection will vary greatly between sensing times and places as the vehicle goes through motion. The resulting quick turnover of environment of the illumination and the background calls for robust object detection algorithm in order to achieve consistency and reliability. While much of existing computer vision algorithms is effective within their usage, they lack the robustness and flexibility of human vision, and will often underperform in varying environment.

2.1 Primary Visual Cortex (V1)

Within the human brain, cognitive and perceptual processes are carried out in the regions of neocortex, with visual input being processed in the occipital lobe. Although there is no definite model of visual cortex, Hubel and Wiesel's research on feline vision have demonstrated the cells in visual cortex responds to the lines of different orientations. At the same time, none of the cells responded to any specific objects like hands, face or body, as human beings would normally perceive visually. It was from this discovery of our visual systems breaking objects into basic constituent parts in order to process them into the visual experience that various theories of object recognition originated from. Theses researches on neurophysiology introduced not only the neural networks software algorithm but also the principles of biologically plausible implementation.

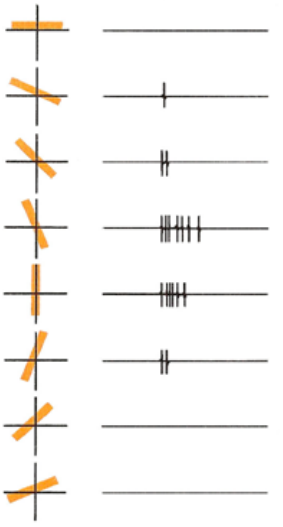


Fig. 1. Response of the cat's cortex when a rectangular slit of light of different orientations is shown [1]

2.2 Neuromorphic Neurons

Neurophysiological model of Hodgkin-Huxley formalism is the most precise spike neuron model with the biological plausibility. The neuromorphic neuron of visual cortex can be implemented by simulating the behaviour of neuron in Hubel and Wiesel's experimentation. The spike neural signal is explained by the widely adopted Hodgkin-Huxley formalism in Fig. 2, with the controlled conductance based equivalent model[5]. Hodgkin-Huxley formalism is unlikely used as much in neural networks or VLSI because of uncompromised large demand in computing complexities in its implementation; however the asynchronous spikes are considered as principle element of high level or large scale neural computing system.

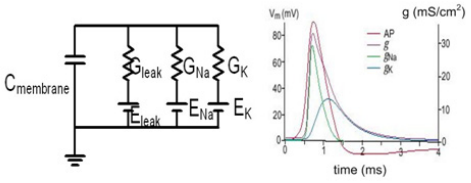


Fig. 2. (left) An electrical equivalent circuit of a neuron, Hodgkin-Huxley formalism; (right) dynamics [5]

The neuromorphic system of neuron and synaptic network was designed for evaluating the feasibility of mimicking the primitive behavior of brain neural system in electronic hardware using the CMOS electronic circuit of transconductance [2,6]. The CMOS transconductance circuit has the wide range of analogue signal processing based on the linearity and programmability, which can be applied to implement various analogue or analogue-mixed circuit applications.

With the neuromorphic neuron formed as in Fig. 3, the various stimuli of six 50 x 50 pixel sized rectangles at different angles are applied as the similar stimulus input to the cat in Hubel and Wiesel's experiment. The simulated result of neuromorphic neuron in Fig. 4. shows the consistent outcome as the observation from the Hubel and Wiesel's experiment in Fig. 1, where the tuned feature orientations are represented as the spike signal outputs.

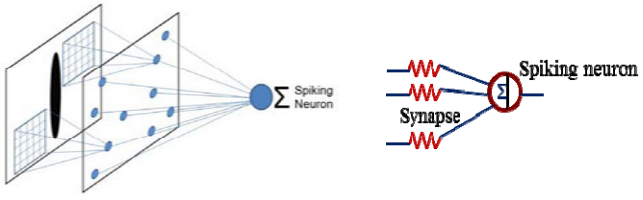


Fig. 3. Spiking neuron and neural networks to mimic the visual cortex's primary neuron

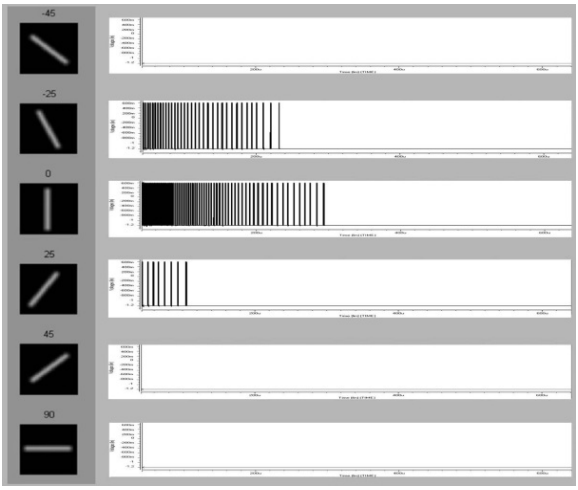


Fig. 4. Simulated behaviour of visual cortex in Figure 1 by CMOS neuromorphic neuron

2.3 Figure Detection by Mimicking Simple Cells of Visual Cortex

The neuromorphic neuron shows its mimicking behavior of simple cell of visual cortex, which suggests a methodology of implementing the robust or flexible recognition of figures. There have been many research activities on visual recognition, either inspired by Hubel & Wiesel's research or not. The detection of particular object or figure is still challenging in terms of the functional robustness or the complexity in implementation. The proposed neuromorphic neuron is applied to the figure detection, for the purpose of investigating its behaviour in changing environment.

The first example is chosen for the license plate detection at the flexible distance to the car, which is to detect the rectangular shape. Two components of orientation (vertical and horizontal) are selected to locate the license plate, which is based on the simple approach to recognize the right angles of rectangular license plate. The position of right angle is expected to yield the outcome from both vertical direction and horizontal direction, with the robustness to minor changes. The detection of license plate is implemented by evaluating the histogram of both horizontal and vertical directions, as in Fig. 5. The threshold of detection is determined by the relative ratio (0.8) of histogram average. The neuromorphic function of orientation selectivity is implemented by the orientation filters in Fig. 6, which simulate the synaptic connections in Fig 3. The overall system in Fig. 5 is implemented by using MATALB, prior to the VLSI implementation in the future. The equivalent values to synaptic connection are {-0.6, 0, 1, 2}, which are also appropriate range for the mixed CMOS VLSI implementation. The process is iterated twice to perform the successful separation of license plate. The test images are captured as color image of 640x480 by the digital camera, from various locations. The license plates from three nations are tested successfully, even though there are challenging license plates with the decorative shiny metal chain surrounded or the mirror-like chrome bar nearby. The numbers of characters on the license plate also contributed to the histogram analysis as illustrated in Fig. 7.

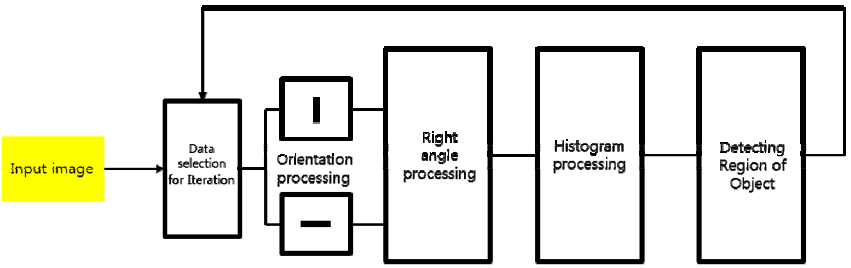


Fig. 5. Neuromorphic visual signal processing for car license plate detection

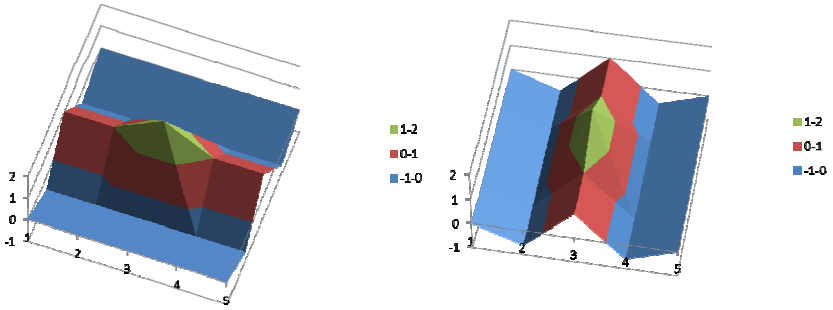


Fig. 6. Orientation filter of neuromorphic vision system in Fig. 5

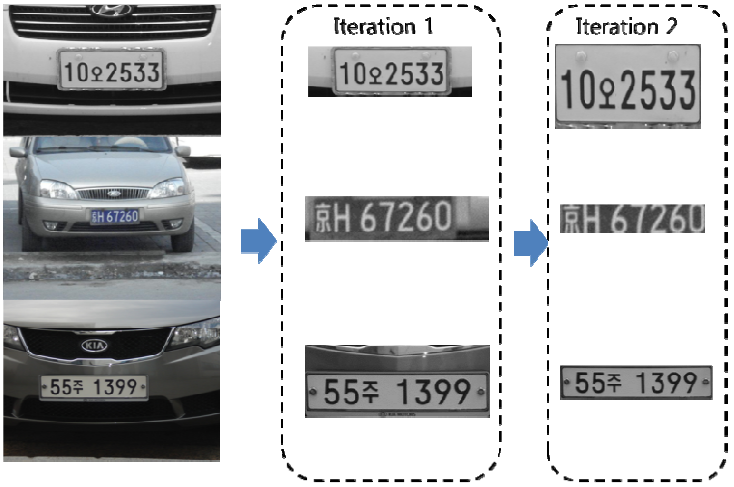


Fig. 7. Car license plate detection and segmentation by neuromorphic vision system of Fig. 5 and Fig. 6

The second example is to detect objects appeared on the road monitored CCTV. The selectivity of 6 orientations is chose to recognize any objects on the road, still based on the simple histogram analysis. The object detection is based on the histogram of sum of 6 orientations, which is different from the license plate detection. The CCTV of 720x480 was located at the local country road side, connected via IP network. The detection of various objects is tested successfully based on the simple histogram analysis of 6 orientations, as shown in Fig. 8. The detected objects are various in shape, such as cars, bikes, or pedestrians.

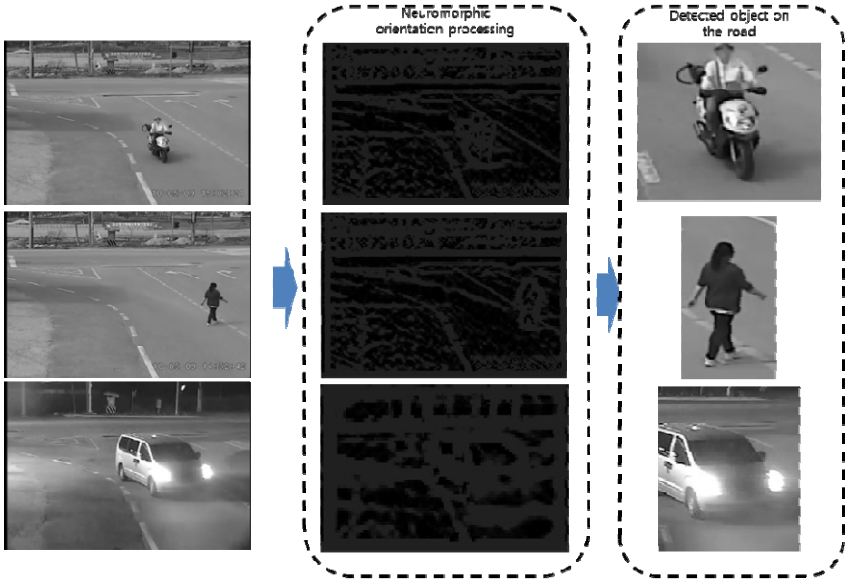


Fig. 8. Neuromorphic visual detection of object on the road

3 Neuromorphic Visual System

The neuromorphic simple cell neuron of visual cortex enables applications of various visual recognitions, as in examples of car license plate detection and object detection on the road, primarily based on the orientation selectivity. The neuromorphic visual signal processing is motivated by the robust performance, especially for the natural outdoors. The pedestrian detection demands the robustness further, if it is for the car-related applications. Particularly, the new Euro NCAP requires the robust pedestrian detection in various conditions, such as coming out from the side way and partly obscured by other vehicles or etc. The situation includes various issues such as the human object only appeared with head and upper body, the limited light conditions of twilight or other poor viewing conditions. There is also a possible application to detect a human object in the car for various purposes, which are for safety or entertainment.

3.1 Neuromorphic Vision for Human Detection

The basic configuration of neuromorphic vision system is shown in Fig. 9, is implemented by MATLAB. The orientation filter function is the first step of the orientation feature extraction using neuromorphic neuron. The second step is to apply neural network to the orientation extracted image for detecting the human head and upper body, yielding the saliency map. The decision of human objects recognition is the final stage by interpreting the saliency map. As we mentioned previously, the target area for this study will be heads of pedestrians or human objects. One of the major challenges faced in the process of pedestrian detection for the enhanced vehicle safety is relation between the reliability of the detection and varying environmental conditions. For an example, most pedestrian detection algorithms have significant drop in its detection rate at when the pedestrian is covered by any material or obstacles. However, as the neuromorphic vision system is based on the orientation selectivity of simple cell and the coarse head-torso shape, rather than pattern matching or complex figure patterns. The neuromorphic visual processing provides the robustness as in Fig. 10.

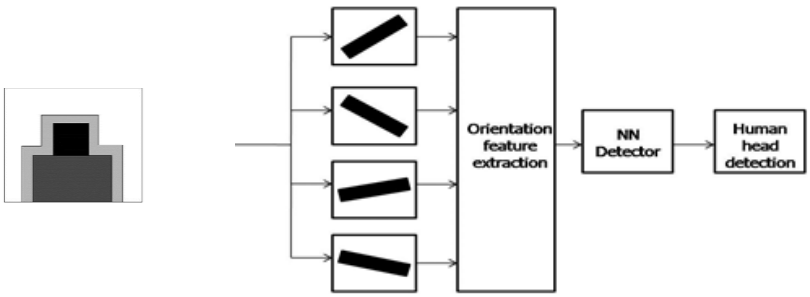


Fig. 9. Neuromorphic vision for human head figure detection, inspired by visual cortex (left); template of human head detection (right)

Fig. 10. Example of neuromorphic visual processing to provide robustness in case where target is shrouded in clothing

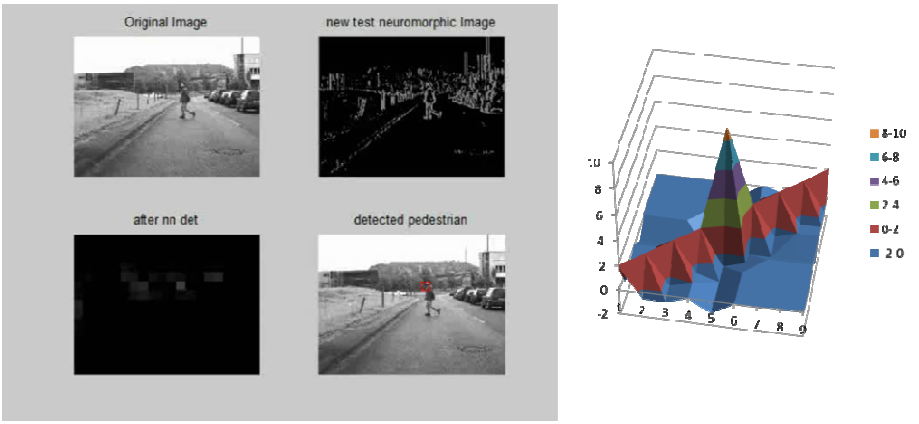


Fig. 11. Pedestrian detection by neuromorphic vision (left); an example of orientation selective filter (45 degrees) employed

The pedestrian detection is applied to the video data set from Daimler research, which is known to be prepared by the planned scenario. The successful detection is observed by applying the orientation filter of 9 x 9 in Fig. 11, which simulates the neuromorhic neuron of orientation.

The inherent robustness of neuromorphic visual processing is observed in Fig. 10, which suggests the feasibility of human object detection in the extreme environment. The example in Fig. 12 shows successful detection of soldiers wearing helmet in a trench from the video clip of military drill for the battle. It also represents that the neuromorphic vision system is robust in detecting camouflaged figured soldiers within low visibility environment, even with the reduced visibility substantially.

Fig. 12. Successful detection of soldiers in trenches despite presence of steel helmets

3.2 Robustness at Night Time and Worse Lighting

The pedestrian detection for conventional ADAS(Advanced Driver Assistance System) is usually for the day light operation, even though the severity of pedestrian accident is higher in the twilight than the day light. The neuromorphic vision system in Fig. 9 exhibits the successful pedestrian detection at dark times, which shows its robustness regardless the change in illumination. Additionally, there isn't so much need to adapt the neuromorphic vision system for different image sensors, as the data sets in Fig. 13 were prepared by using different imaging devices and at different location. The image data in Fig. 13 (a) was produced by a low cost DVR for car (also called as Car Blackbox), while the image data in Fig. 13(b) was produced by a digital camera.

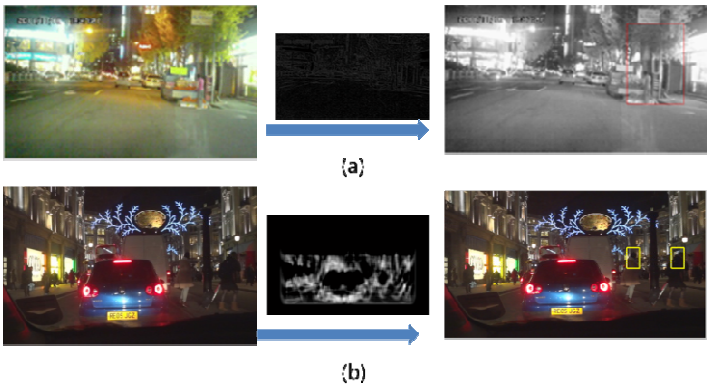


Fig. 13. Night time pedestrian detection using various image sensors (a) car DVR: red box for detection (b) low-end digital camera: yellow box for detection

The neuromorphic vision system shows its robustness at the worse condition of spot light and visible gas. The image data in Fig, 14 is produced at the car collision test by the other Car Blackbox , where the dummy was used. It is usual to record the collision test by special video recording equipment, with the extremely bright elimination. Besides, there exists the gas from the airbag operation, which also causes the degradation of image. The image of Fig.14(a) and Fig. 14(b) are consecutive images, where the inflated airbag is detected by the airbag template (green box). In Fig, 14(a), the airbag is started inflating, while the whole upper body of dummy moved toward front abruptly and occluded by the airbag. The principle of neuromorphic vision is based on the human object detection based on the template in Fig. 9. Therefore, the dummy passenger cannot be identified in Fig. 14 (a). The human detection for dummy is successful at the following frame, even though the size of head is substantially different and the gas from airbag deployment is present.

Fig. 14. Passenger dummy detection at car collision test

The results of Fig. 13 and Fig. 14 illustrate the robustness of recognition process related to the lighting environment.

3.3 Extended Application of Human Object Detection

There are feasible applications of monitoring human on board, either for the safety in driving or for the convenience. The demand of detecting passenger's eye or particular part emerges with the new infotainment service or advanced dash board of 3D display without the eye glasses. The monitoring of passenger's facing becomes important for the enhanced safety since the face direction or eye status of the passenger is applicable to various advanced service for the attention warning or the smart instrument control.

The principle of neuromorphic visual processing for the eye detection is same as for the pedestrian detection. However, for the robust eye detection, both the nose feature and eye feature are integrated together and the neuromorphic processing is based on the still image after locating the passenger's head. The two step processing enhances the image dynamics with localized tuning. Since the outline of the eye and the nose is distinctively different to that of the head, it is important to detect the head of the passenger to start for eye detection. The neuromorphic vision system is unchanged from the system for pedestrian detection, though the image used in the detection was prepared by the different imaging device, in the different location, even the different vehicle model. The successful detection of the passenger head without calibration or any additional settings to match previous detection shows that the system is robust.

With the input image of Fig. 15 (top-left), the orientation features is shown in top-right image of Fig. 15. The orientation features were extracted only from the interior of the vehicle, so that the structure of the vehicle and the outside scenery seen through the window is disregarded in orientation feature extraction. The saliency map after the neural network with human-torso template is applied. From the image it can be seen that the high output levels are concentrated in the centre. And the resulting detection in Figure 15 shows the successful detection of the passenger's head.

Once the area of head is located, the neuromorphic signal processing is iterated again with the same orientation feature extractors and the neural network template of eye detection. The Fig. 15(top-left) is orientation feature extracted from the located head area but it is different to that of Fig. 15 since the orientation feature image is enhanced

by the locally enhanced dynamic range. Since the outlines of the eye and the nose in Fig. 16(top-left) are somewhat clearer than the case of frame difference in Fig. 15(top-right), it is possible to detect an eye of passenger using the appropriate template.

The Fig. 16(top-right) is the saliency map after the template neural network is applied to the Fig. 16(top-left). There are strong signals detected on the left side of the saliency image, possibly due to the noise from other irrelevant objects. The interpretation of saliency map is based on the fact of reasonable size in the saliency map, for eye or nose. The detection result of eye is shown as a winner spot in Fig. 16 (bottom-left), which is represented as the successful eye detection in Fig. 16 (bottom-right).

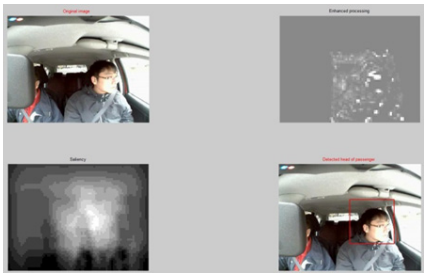


Fig. 15. Detection of passenger's head facing outside

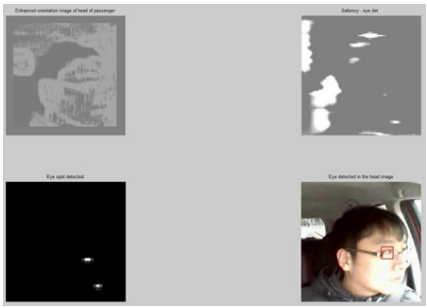


Fig. 16. Detection of passenger's eye after the head detection, iterative orientation processing

4 Vehicle Safety Enhancement through Pedestrian Detection

The use of neuromorphic vision system for the purpose of pedestrian detection in the vehicle could provide substantial enhancement in safety over existing computer vision algorithm. One aspect where the conventional pedestrian detection for the automotive vehicles often struggles is detecting the pedestrian entering the vision from the side. This is partly due to the fact that human detection of those vision algorithms requires presence of head and legs within the image. Neuromorphic vision system provides a more strong performance in this area, as shown on the next column. The test image in Fig. 17 shows the successful pedestrian detection in the middle of street parked cars in the residential area, where the pedestrian was with a hat and an umbrella and appeared from the side in the rainy weather.

One of the challenges faced in pedestrian detection in urban settings is distinguishing between target area and noise present in the image. Neuromorphic vision system was tested 60 frames within urban environment of varying environment in the natural scenes without any scenario, and provided a 95% successful detection rate. The pedestrian detection in Fig. 18 and Fig. 19 illustrate the successful detection of pedestrian presence, which is appropriate for Electric Vehicle or any silent vehicle to generate the sound warning for the safety of pedestrians.

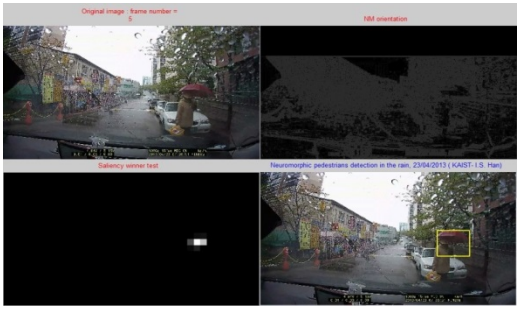


Fig. 17. Successful detection of a pedestrian with a hat and an umbrella, appeared from the street cars in the rainy weather.

The saliency images which come after processing the orientation features show how noise elements of trees and cherry blossoms within the image were successfully enhanced by threshold process of neurons. The orientation image shows a lot of features other than the pedestrians on pavement and background. However the characteristics of head detection can separate the large segment in the saliency as they are usually not relevant to human object detection. Once the neural network processing is applied to the image, most of irrelevant signals other than the target are eliminated.

However, there are some situations where the neuromorphic vision system could not process successfully the noisy object such as tree. There are three instances of failure of false-positive detection, due to the incorrect pedestrian detection by mistaking the trees as human object. The incorrect surplus detections cause no dangerous

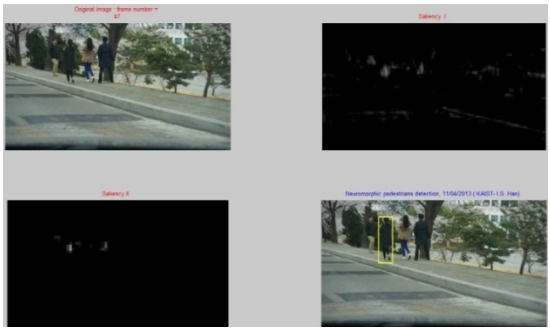


Fig. 18. Example image of successful pedestrian detection within a typical urban environment (one of 60 frames)

Fig. 19. Example image of successful pedestrian detection within a typical urban environment (consecutive image of Figure 18)

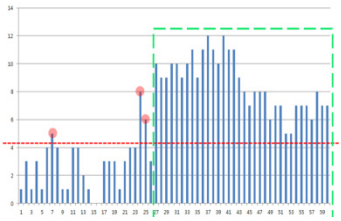


Fig. 20. Neuron activity levels for pedestrian detection, where red circles signifies incorrect detections

issues to Electric Vehicle's safety because its purpose is for activating the sound warning to the pedestrian, while the tree is usually on the road side instead of on the road. The neuron activity level of each frame is summarized in Fig. 20, which shows the false positive detection in red dot.

5 Stereo Neuromorphic Processing for Robust Vision

The stereo vision is expected as a viable solution for the new Euro NCAP of pedestrian detection, which is essential for Autonomous Emergency Braking (AEB). AEB is recommended to be tested for detecting the pedestrian and his/her distance, which requires the distance sensing or the pedestrian detection at particular distance.

The neuromorphic vision system of Fig. 9 can also estimate the distance to a detected pedestrian, using the multiple scales of sight image or template with the fixed optical system. However, for the noisy visual environment on the road, the stereo vision has the advantage of disparity map. The background of the sight sometimes invokes the erroneous operation, as observed in Fig. 20. The tree sometimes yields the many neuromorphic signals of noise for detection, as its figure has various orientations. Under certain conditions such as at night time, the human often mistakes a tree as human object. The principle of stereo neuromorphic processing is based on the difference of two images' neuromorphic orientations as in Fig. 21. The detection range is controlled by the offset of horizontal shifting of one image's orientation,

while the other image's orientation remains unchanged. The effect of shift is illustrated with the stereo image of Fig. 22.

The image of Fig. 22 is from a stereo webcam, which is 640x480 in pixel sizes. The right hand is the closest object, while it has the complicated background of Chinese painting. The image of left sight (webcam) is shown with the neuromorphic orientations in Fig.22. Comparison of left sight and right sight illustrates the shift of horizontal background, which is far-end. By changing the offset of horizontal shift, the orientation components of designated distance remain while other parts are nearly removed.

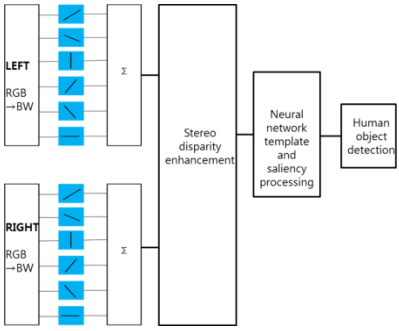


Fig. 21. Stereo neuromorphic visual processing

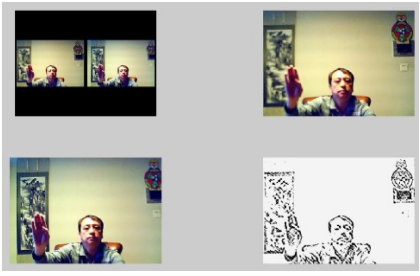


Fig. 22. Images from stereo webcam : (top left) stereo images (top right) left image (bottom left)right image (bottom right) neuromorphic orientations from left image

The red circles in Fig. 23, Fig. 24 and Fig. 25 locate the most dominant features in each stereo Neuromorphic processing. The neuromorphic orientations of Chinese painting in Fig. 22 become much reduced in Fig. 23 as a result of stereo processing. The neuromorphic orientations of hand remain almost unchanged, though those are located in the area of Chinese painting. In Fig. 24, the spotted hand area becomes obsolete, while the head area becomes dominant as in the red circle. There are almost no neuromorphic orientations of human object in Fig. 25 for far-end stereo processing.

.

Fig. 23. Stereo neuromorphic processing for the near end

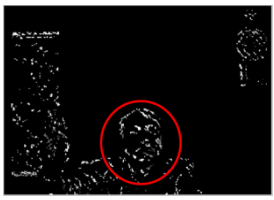


Fig. 24. Stereo neuromorphic processing for the mid range

Fig. 25. Stereo neuromorphic processing for the far-end

The stereo Neuromorphic processing is applied to the pedestrian detection for the automotive safety, where the stereo video was captured from the passenger car as in Fig. 26. There observed crowded trees in the background of both the left image and the right images in Fig. 26

Fig. 26. Stereo video frames(left, right) recorded by a stereo digital camera (location: KAIST, Korea, time: 12:30 ~13:30, July 2013, cloudy weather)

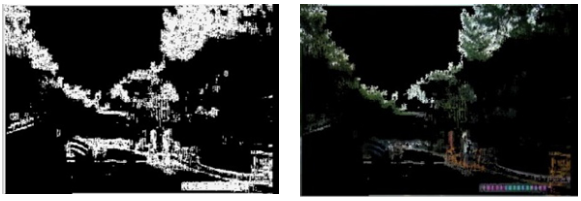


Fig. 27. (left)stereo neuromorphic processing (right) disparity map

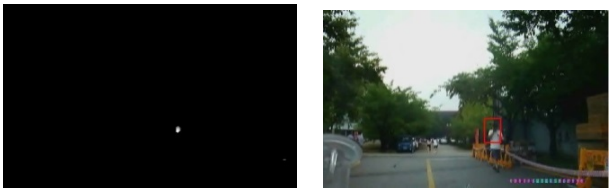


Fig. 28. (left) saliency map (right) detected pedestrian on the shifted image

Fig. 29. Stereo neuromorphic processing of consecutive video frames at the rate of 30 frames/sec.(top to bottom) continuous frames until passing by the pedestrian,(left to right) input image with the detected location of head of pedestrian, disparity map, saliency map

The video was captured without a scenario at the university campus (KAIST, Daejeon, Korea) and the weather was cloudy. The image size is 720 x 480 pixels and the frame rate is 30 frames/sec. The stereo neuromorphic processing is based on the horizontal shift of 30 pixels and most of neuromorphic orientations from trees are removed as in Fig. 27. The disparity map of Fig. 27showes the near-end objects as a result of stereo neuromorphic processing. The passenger car was moving in low speed

while recording the video. Fig. 28 demonstrates the saliency map and pedestrian detection, which is relatively robust to the complex background image.

The images of Fig. 29 demonstrate the applicability of stereo neuromorphic processing in pedestrian detection for automotive safety, with the successful range selective performance.

6 Human Object Detection in Wet Weather and Extreme Condition

The human object detection for automotive safety can be motivated from different reasons for various applications. The pedestrian detection in Fig. 18 or Fig. 19 is for the safety issue of Electric Vehicle (EV), which is silent without the engine sound. The scope of pedestrian detection for EV is more or less for alarming the pedestrian by generating the sound. It is crucial to detect not only the pedestrian on the road in the front but also the human object in the pedestrian pathway. The neuromorphic vision of Fig. 9 is the human object detection based on the head figure and upper torso, which demonstrated the pedestrian detection appropriate to EV as illustrated in Fig. 17, Fig. 18 and Fig. 19.

The status of art in automotive pedestrian detection is usually for the day light and dry weather. There emerges the demand of cyclist detection in rainy days, particularly for the commercial vehicle. In the wet days or bad weather, the commercial vehicle drivers are prone to be negligible at the approaching cyclist from side, where there are high risks of casualty. Hence, the detection of cyclist in wet weather was tested by using the neuromorphic vision based on Fig. 9. The image of Fig. 30 is from the video for such purposes, which was recorded from on-board tour coach in the wet weather (weather: mixed snow and rain, windy, location: Suzhou, China, time: 16:00~17:00, 29th December, 2012). The window wiper in operation shows the wet weather, while the head lights of vehicles and street/building lights confirm the darkness. The image size is 1920x1080 and the frame rate of video is 60 frames/sec.

Fig. 30. A video frame recorded from the commercial vehicle (coach bus) in wet weather and at dark time

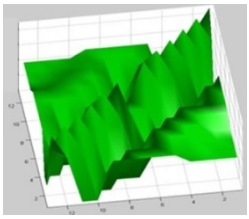


Fig. 31. Weighted synaptic connection of neural networks for neuromorphic visual cortex neuron of orientation selectivity (an example of 45°)

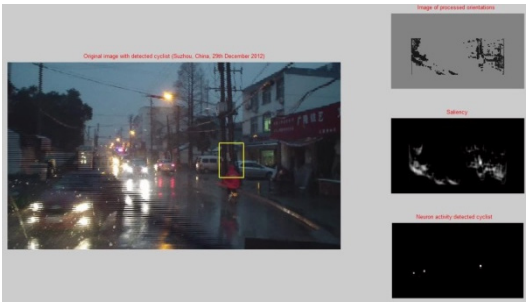


Fig. 32. Neuromorphic bike/cycle rider detection in wet weather, video recorded from the tour coach (location: Suzhou, China, time: 16:00~17:00, 29th December, 2012)

One of key features is the orientation selectivity in neuromorphic vision and is synaptic weights connected to the particular neuron in visual cortex. The synaptic connections of (13x13) are employed in the neuromorphic vision, with the weights based on Fig. 32. For the practical VLSI ASIC implementation, the low resolution integers are used as the weight parameters.

The neuromorphic vision demonstrated the successful human object detection in Fig. 32, while the bike rider wearing the poncho to avoid the wetness. The top figure in right column in Fig. 32 shows the neuromorphic orientation of the image in left column, and the middle figure represents the saliency map. The bottom figure shows the candidate of human object detection. In fact, there are three human objects found at the corresponding locations in the image with the careful inspection. The strongest candidate, who would be at the risk of casualty, is determined by the saliency value, which is located by the yellow box in the left image of Fig. 32.

Fig. 33. Examples various conditions such as moving window wiper (frame rate: 60 frames/sec)

The continuous human object detection is simulated successfully by using 9 consecutive frames, where Fig. 33 illustrates various conditions and the positions of window wiper. The spatial deviation of head area is caused by the single scale application of template, which is not an issue in human object detection for the safety enhancement. The precise head location can be tuned by adding another scale, if necessary.

Driving conditions may vary substantially even within one journey, and accurate detection in extreme conditions would be even more crucial for ensuring the safety issues of automotive vehicles. Furthermore, human object detection in extreme conditions would have diverse applications beyond the scope of road safety, such as surveillance and security.

Following images show still video frames from BBC World News of military drill in the country of higher tension this year, where marines are performing landing manoeuvres. Background of sea water in the image proved challenging to detection process due to the presence of shape-variant wave. Furthermore, it is always a challenge to detect human objects when they are actively attempting camouflage. These test images are for evaluating the robustness and effectiveness of proposed Neuromorphic processing based on the same simulated framework used for the human object detection in the urban environments.

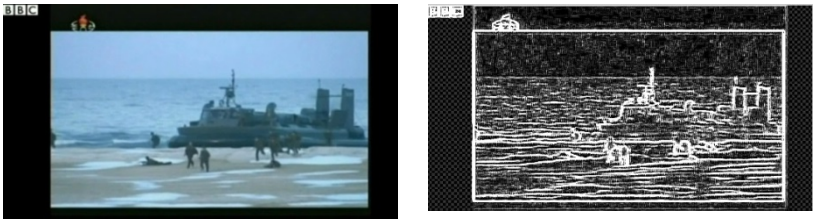


Fig. 34. Original input image (left); Image processed by neuromorphic system

Fig. 35. Sailency map (left); detected human objects(right)

Fig. 36. Images of highlighted boxes of detected human objects

Fig. 34 shows the image processed with orientation features after neuromorphic processing of original image (left). It shows high output levels throughout the picture, reaffirming the challenge of extreme condition, in this case that of background sea and water wave features. However, together with the implementation of neural network, the mimicking of visual cortex function of Fig. 1 allows for effective thresholding of image noise and successful detection of human objects as a result shown in Fig. 35, as the saliency map illustrated the robustness in detecting human objects in the terrestrial environments. The detection based on the feature of human head and torso in Fig. 9.

7 Conclusion

Neuromorphic vision system offers an alternative to existing computer vision algorithms. This is largely due to its robustness which allows for flexibility of application in diverse environment while retaining high accuracy. The proposed neuromorphic vision system is evaluated successfully for its applicability and feasibility to the enhanced safety of electric passenger vehicle, via the active virtual engine sound system by activating the warning and artificial engine sound to alert the pedestrians when they are near the silent electric car.

Further research is under development for the visual recognition of various objects and real-time status in the extreme environment of limited visibility by mimicking and neuromorphic implementation of the simple and complex elements of visual cortex.

Having based on analogue-mixed CMOS VLSI or ASIC/FPGA for the purpose low cost and real time application, the first ASIC-based Neuromorphic vision system demonstrated its feasibility successfully for the enhanced safety of Electric Vehicles for the Automotive OEM industry.

Acknowledgement: This research work was supported in part by the Ministry of Knowledge and Economy of Korea under the Grant for Next Generation Passenger Electric Vehicle Development, and in part by the National Research Foundation of Korea under the Grant 20120006738.

References

1. Hubel, D.H., Wiesel, T.N.: Receptive fields of single neurons in the cat's striate cortex. J. Physiol. 148, 574–591 (1959)
2. Han, I.S.: Mixed-signal neuron-synapse implementation for large scale neural networks. Int. Journal of Neurocomputing, 1860–1867 (2006)
3. Risenhuber, M., Poggio, T.: Hierachical models of object recognition in cortex. Nature Neuroscience, 1019–1025 (1999)
4. Han, I.S.: Bio-inspired neuromorphic processing and its applications. In: WSEAS, Cambridge, U.K. (2013) (Plenary talk)
5. Hausser, M.: The Hodgkin-Huxley theory of action potential. Nature Neuroscience Supplement 3, 1165 (2000)
6. Han, W.J., Han, I.S.: Bio-Inspired Visual Signal Processing based on Neuromorphic Circuit. In: Proc. WSEAS, pp. 131–136 (2010)

Rescue System for Elderly and Disabled Person Using Wearable Physical and Psychological Monitoring System

Kohei Arai

Graduate School of Science and Engineering, Saga University, 1 Honjo,
Saga 840-8502 Japan
arai@is.saga-u.ac.jp

Abstract. Rescue system for disabled and elderly person is proposed together with method and system for frequent health monitoring as vital signs with psychological status monitoring for search and rescue of handicapped person is proposed. Heart beat pulse rate, body temperature, blood pressure, blesses and consciousness is well known vital signs. In particular for Alzheimer diseased persons, handicapped peoples, etc. it is very important to monitor vital signs in particular in the event of evacuation from disaster occurred areas together with location and attitude information. Then such persons who need help for evacuation can be survived. Through experiments wearing the proposed sensors with three normal persons including male and female, young and elder persons and one diseased person, it is found that the proposed system is useful. It is also found that the proposed system can be used for frequent health condition monitoring. Furthermore, physical health monitoring error due to psychological condition can be corrected with the proposed system.

Keywords: wearable computing, e-Health, evacuation, rescue system.

1 Introduction

The purpose of this study is to save human life in particular for disabled and elder persons when disaster occurs. Handicapped, disabled, diseased, elderly persons as well as peoples who need help in their ordinary life are facing too dangerous situation in event of evacuation when disaster occurs. In order to mitigate victims, evacuation system has to be created. Location and attitude of disabled and elderly persons have to known. GPS receiver and attitude sensor

Vital sign is Blood pressure, Bless, Pulse rate, Body temperature, and Psychological status (EEG signals for instance). In addition to these, the number of steps together with calorie consumption is measured. Pulse rate, bless, body temperature, blood pressure, and psychological status. Pulse rate is used to measure for 15 second every 1 minute while Bless is measured for 15 second. Meanwhile, Blood pressure is used to measure for two times with 30 second interval while Body temperature is also measured for two times, 2-6 a.m. and 3-5 p.m. On the other hand, Psychological status

L. Chen, S. Kapoor, and R. Bhatia (eds.), *Intelligent Systems for Science and Information*,
Studies in Computational Intelligence 542,
DOI: 10.1007/978-3-319-04702-7_3,

during four vital signs is measured. Thus some errors due to psychological influences on physical health monitoring can be corrected using psychological status. If the information can be gathered, then disabled and elderly person can be rescued through assignment of the most appropriate person for the person who needs a help for rescue through decision making.

Authors proposed such evacuation system as a prototype system already [1]-[4]. The system needs information of victims' locations, physical and psychological status as well as their attitudes. Authors proposed sensor network system which consist GPS receiver, attitude sensor, physical health monitoring sensors which allows wearable body temperature, heart beat pulse rates; bless monitoring together with blood pressure monitoring [5]-[7]. Also the number of steps, calorie consumptions is available to monitor. Because it is difficult to monitor the blood pressure with wearable sensors, it is done by using the number of steps and body temperature. In addition to these, psychological status is highly required for vital sign monitoring (consciousness monitoring). By using EEG sensors, it is possible to monitor psychological status in the wearable sensor. These are components of the proposed physical health and psychological monitoring system.

The proposed system also allows frequent monitoring. Even for every minute, or every second, it may monitor all the required items. Therefore it is applicable to the patients in ICU. Also, it may find Alzheimer patients who used to walk away from their house and /or hospitals together with physical health and psychological status. Furthermore, it may reduce physical health monitoring error due to psychological status changes. Even for the healthy persons, it occur such errors. For instance, heart beat pulse rate and blood pressure is used to be increased when medical doctor or nurse measures. By using EEG signal analyzed results, such errors may be corrected or at least it can be omitted from the monitored data. These are kinds of bi-products of the proposed system.

Section 2 describes the proposed system followed by experiment method and results. Section 3 also describes rescue system with simulation results. Then conclusion is described together with some discussions.

2 Proposed Method

2.1 System Configuration

Fig.1 shows the entire system configuration of the proposed rescue system for disabled and elderly person together with physical and psychological health monitoring system.

Normal person who may rescue disabled and elderly person are situated somewhere around their district. The locations of disabled and elderly person as well as normal person are known with GPS receivers of which they have in their mobile

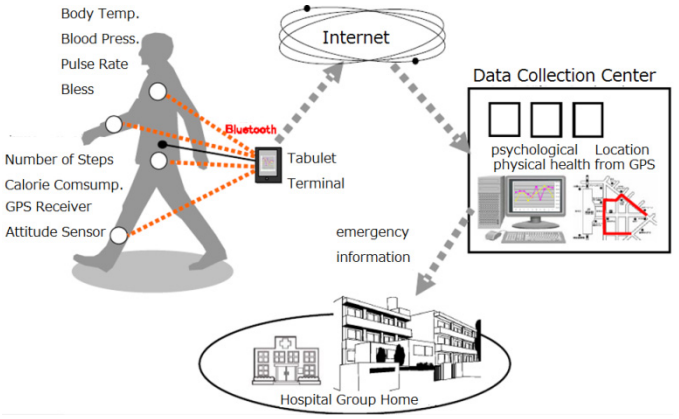


Fig. 1. Entire system configuration of the proposed wearable physical and psychological health monitoring system together with rescue system

devices. Also, their attitudes are known with their attitude sensors which are included in their mobile devices. On the other hand, wearable physical and psychological health sensors are attached to the proposed Arai's glass. These sensor data are transmitted to their mobile devices through Bluetooth communication links. The acquired sensor data together with location and attitude data are transmitted to the Health Data Collection Center: HDCC through WiFi communication links. Using these gathered data of each disabled and elderly person as well as normal person, data collection center makes decisions on who (normal person) rescues whom.(disabled and elderly person) taking into account physical and psychological conditions and traffic condition which can be collected with the different method and system.

Fig.2 shows the entire system configuration of the proposed for physical and psychological health monitoring system.

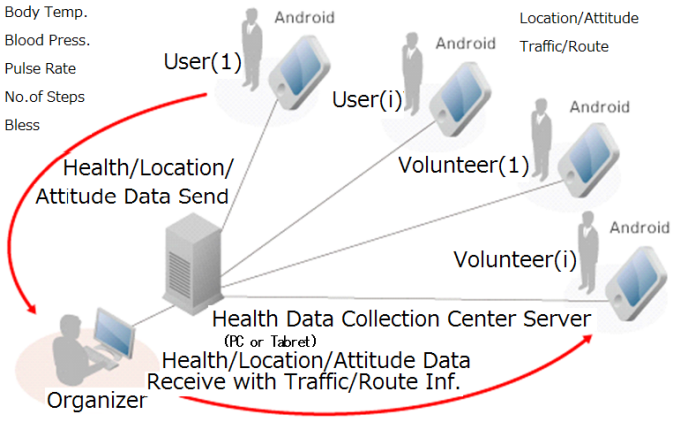


Fig. 2. System configuration

There are two types of stakeholders, patients (users) and volunteers who are responsible for evacuation, rescue, and help patients from disaster area. Patients have physical and psychological health sensors and send the acquired data through Bluetooth and Internet to the HDCC server. On the other hand, volunteers receive health data of the previously designated several patients together with traffic flow information and appropriate route information. When something wrong occurs on the designated patients, HDCC provides information which required for rescue to the designated volunteers then the volunteers rescue patients in an efficient and an effective manner.

2.2 Sensor and Communication System

In order for evacuation and rescue, victims' location and attitude is important. Therefore, GPS receiver and accelerometer are added to the aforementioned measuring sensors for body temperature pulse rate, blood pressure, bless, and EEG, EMG data acquisitions. All sensors should be wearable and can be attached to ones' tall forehead. Acquired data can be transmitted to mobile devices in ones' pockets. Through WiFi network or wireless LAN connection, acquired data can be collected in the designated information collection center. Then acquired data can be refereed from the designated volunteers who are responsible to help victims for evacuation and rescue.

1)Physical Health Sensors Used

Body temperature measuring sensors, Nishimoto Petit Sophia BT-14W, Terumo Electronic Deice C202 is used for experiments. Outlook of C202 and examples of measured body temperature are shown in Fig.3 while Fig.4 shows outlook of pulse rate sensor and measured data. The pulse rate sensor, Nissei Pulse Coach neo HR-40 is used for three time a trial, before exercise, just after the exercise and after cooling down.

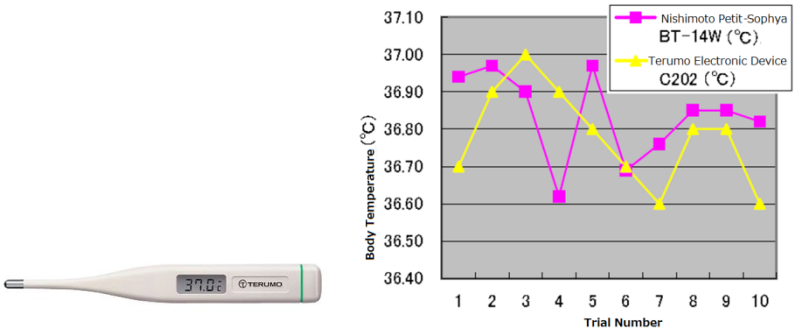


Fig. 3. Body temperature sensor used and examples of measured data

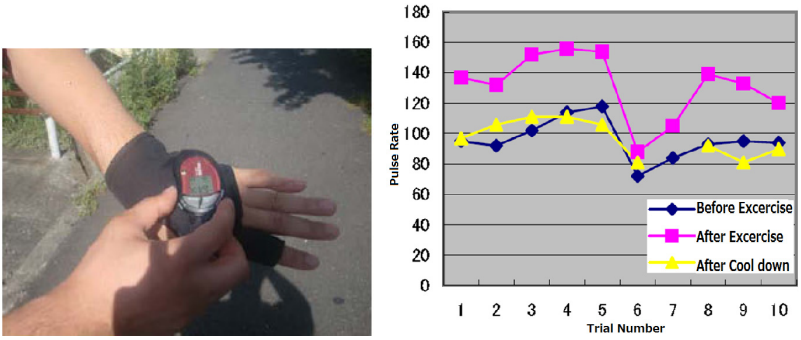


Fig. 4. Pulse rate sensor and examples of measured data

Citizen TW 700 of sensor for the number of steps and calorie consumption is used. Outlook and example of measured data is shown in Fig.5.

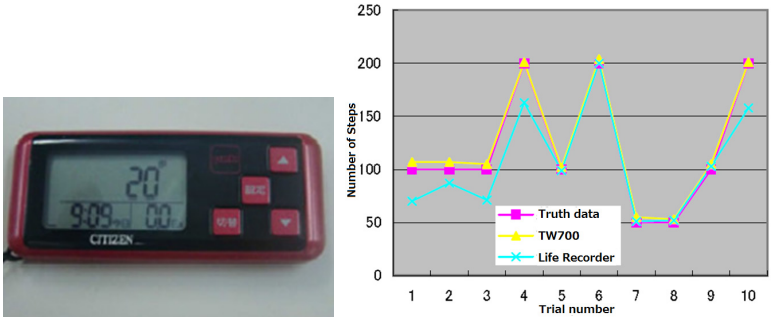


Fig. 5. Sensor (Citizen TW 700) for the number of steps and calorie consumption

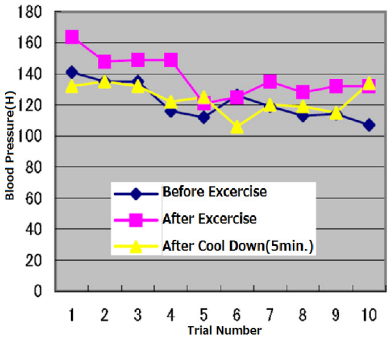
Body temperature sensor head is attached to ear because sensor head has to be attached to Arai's glass. The difference between body temperature measured at armpit and ear is shown in Table 1. Body temperature measured at ear is always less than that measured at armpit by 0.2-0.9 degree Celsius. Therefore, body temperature measured at ear can be corrected by adding 0.55 degree.

Table 1. Body Temperature Difference Measured at Armpit and Ear

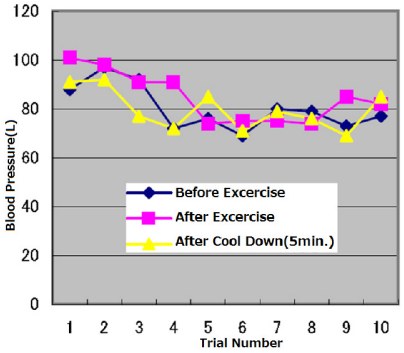
User	Locations of Sensor		Difference ^{0}C	Note
	Armpit (TERUMO) ^{0}C	Ear (Omron MC-510) ^{0}C		
1	36.7	36.5	0.2	1 hour stay in room
2	36.6	36.1	0.5	1 hour stay in room
3	36.9	36.0	0.9	10 minute stay in room
4	35.8	35.1	0.7	5 minute stay in room

Meanwhile, outlook and measured data of blood pressure sensor (Omron HEM-6022) is shown in Fig.6. Blood pressure is the most difficult to measure with Arai's glass. Therefore, blood pressure is estimated with the other parameters, body temperature, the number of steps, etc.

(a)Outlook



(b)Example of measured data of blood pressure high



(c)Example of measured data of blood pressure low

Fig. 6. Blood pressure sensor and examples of measured data of blood pressure high and low

2)Psychological Health Sensors Used (EEG Sensor)

NeuroSky of EEG sensor is used for psychological status monitoring. Outlook and an example of psychological status monitoring display of the NeuroSky are shown in Fig.7. Sensor head of NeuroSky is detached and mounted on Arai's glass. Peak Alpha Frequency: PAF is used for representation of psychological status. Fig.8 shows examples of raw EEG signal and calculated power spectrum while Fig.9 shows PAF data when user is calm and exciting status.

Location of electrode has to be attached to forehead. Preliminary test is conducted for confirmation of EEG signal output between electrodes is attached to the forehead and the center of two eyes. Fig.10 shows the different locations of the electrode and example of EEG signal output between both. Obviously, EEG signals detected at the forehead are much greater than that from the center of two eyes. In the Arai's glass, not only EEG sensor but also body temperature sensor is attached together with battery. Because these are Bluetooth communication interface equipped instruments, it is easy to transmit the acquired data to mobile terminal of which users have. Also, GPS receiver and attitude sensors are equipped in the mobile terminals together with WiFi communication capability.

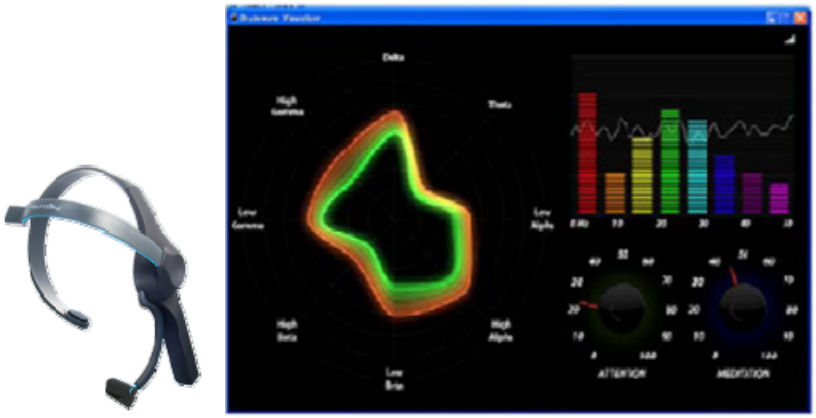
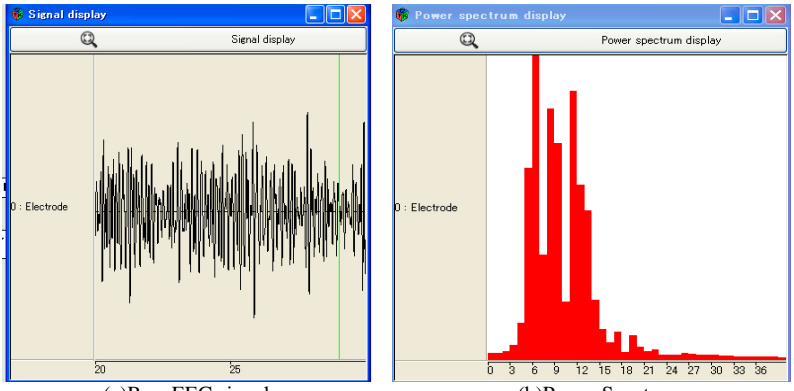
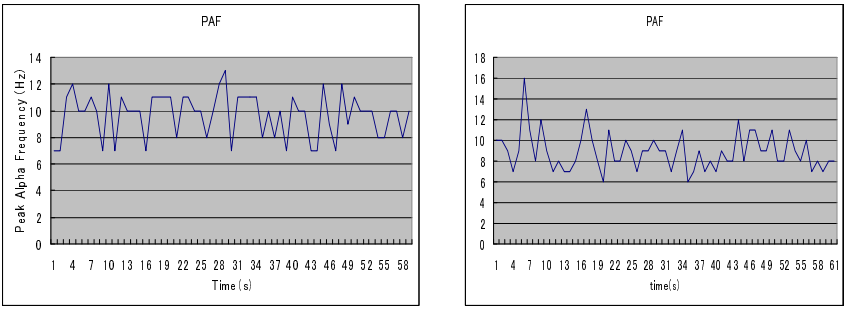


Fig. 7. Outlook and an example of psychological status monitoring display of the NeuroSky



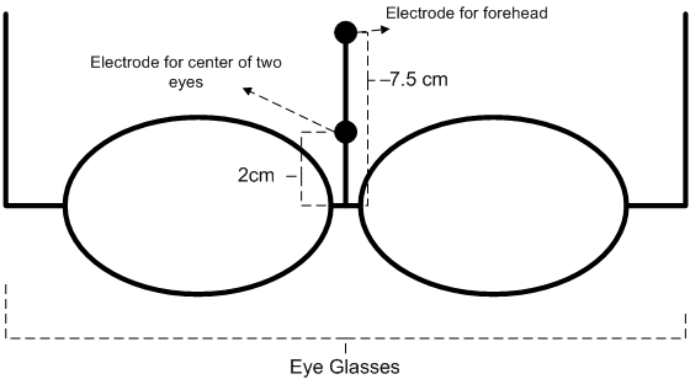
(a)Raw EEG signal (b)Power Spectrum

Fig. 8. Examples of raw EEG signal and power spectrum

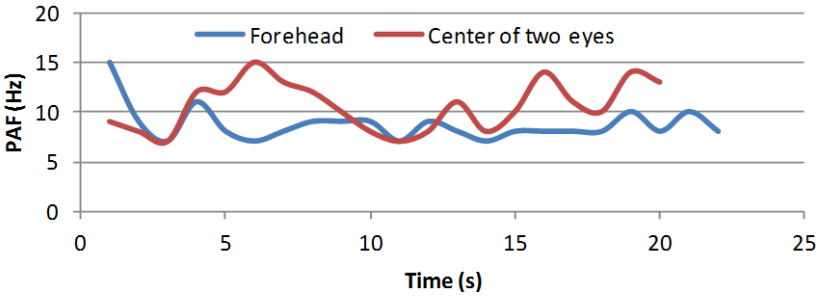


(a)Calm status (b)Exciting status

Fig. 9. Examples of PAF when user is in calm and exciting psychological status



(a)Locations of electrodes at Arai's glass



(b)Examples of EEG signals

Fig. 10. Locations of electrodes and examples for the different two locations of EEG

3 Experiments

3.1 Experimental Method

1) Patients

Four patients are participated to the experiments. The difference due to gender can be discussed through a comparison between patients A and C while the difference due to age can be discussed through a comparison between patients B and C. Meanwhile, the difference due to the degree of Alzheimer can be discussed through a comparison between patients B and D as shown in Table 2.

Table 2. Four Patients

Patient	Male/Female	Age	Remarks
A	Male	37	Good in Health
B	Female	47	Good in Health
C	Female	39	Good in Health
D	Female	91	Weak Alzheimer

Experiments are conducted for eight hours a day for almost every working day (Monday to Friday) for six months starting from May 2012. Measuring time intervals are different by the measuring items. GPS location can be measured every two seconds while accelerometer data can be obtained every 10 seconds. Meanwhile, body temperature, pulse rate can be measured every one minutes while blood pressure is measured every one hour together with eeg and emg signals. The number of steps is measured when the walking event happened. At the end of day, four patients evaluate their physical and psychological conditions which are listed in Table 3.

Table 3. Self Evaluation Item

A1	Feel fever	A6	Limper hurt
B1	Loosing thinking capability	B6	Cannot remember something
A2	Feel tiredness	A7	Head ach
B2	Could not sleep well	B7	Loosing balance
A3	Get tired after exercise	A8	Cannot recover after sleep
B3	Feel bad	B8	Cannot think deeply
A4	Muscle hurt	A9	Throat hurt
B4	Unconfident about health	B9	Loosing concentration
A5	Feel depression	A10	Joint hurt
B5	Do not want to work	B10	Sleep for too long time

2) Subjective Evaluation of Physical and Psychological Health Conditions

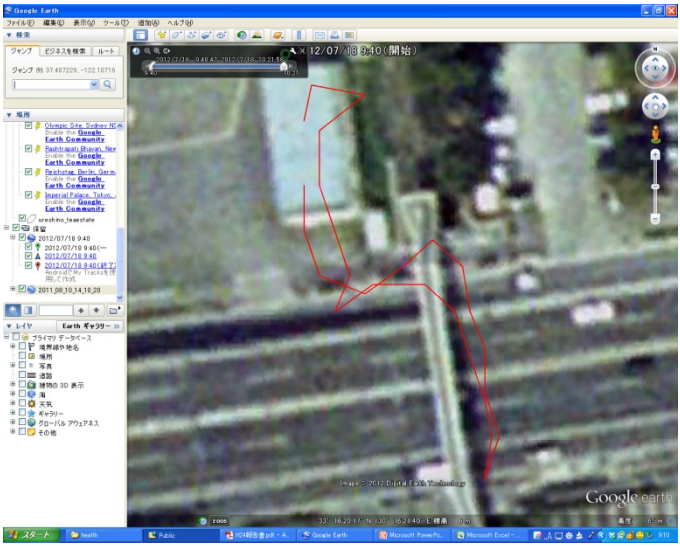
The 20 items listed in the Table 2 are questionnaires for four patients. In the Table, Ai is questionnaire for physical health while Bi is questionnaire for psychological health. The patients respond to the questionnaire above with five levels range from 0 to 4 grades. Total Score is defined as sum of the aforementioned self evaluation of 20 items including physical and psychological health items.

3.2 Experimental Results

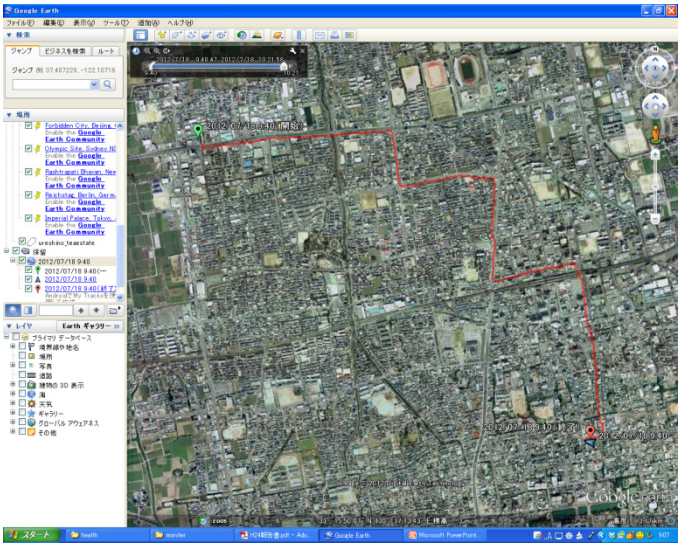
1)Traced Route

Example of traced route measured with GPS receiver on GIS map is shown in Fig.11. Figure 11 (a) is traced route when patient walks on foot while Fig.11 (b) shows the traced route when patient moves by car, respectively.

Fig.12 shows the traced route locations data in the database of the HDCC. A couple of meters of the estimated location error are observed. Also Fig.13 shows an example of measured attitude data in directions of x, y, and z. It is not so easy to estimate the patients' situations (sit, stand up, walking, lay down, etc) from the attitude data derived from the single accelerometer data. As mentioned later, it is much easier to estimate the situations using EEG and EMG sensor data.



(a)On foot



(b)With car

Fig. 11. Example of traced route measured with GPS receiver on GIS map

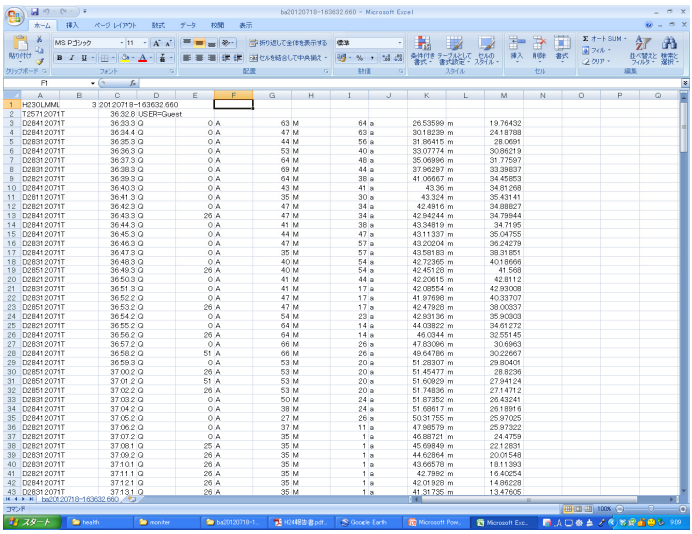


Fig. 12. Traced location in the database of the HDCC

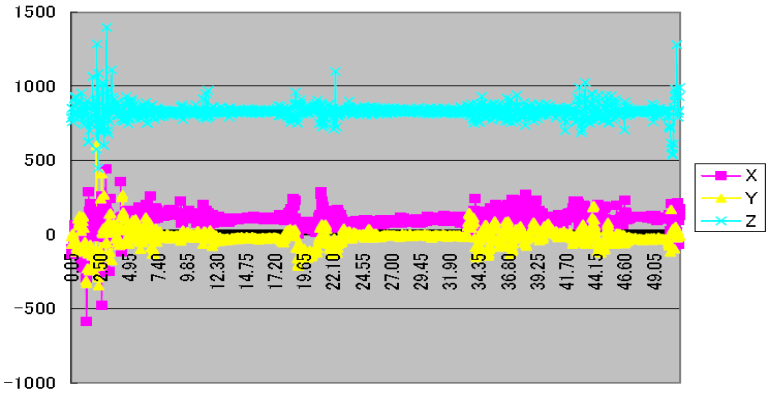


Fig. 13. Example of attitude data (x, y, and z axis movement)

2) Measured Physical Health Conditions

Relation between the measured physical health condition and the self evaluation of physical and psychological health conditions (Total Score) for the patient of weak Alzheimer is plotted in the Fig.14. Total Score denotes sum of the scores of the self evaluation items which are listed in Table 5. Fig.15 (a) and (b) shows physical health data of the weak Alzheimer of patient at the minimum and maximum total scores, 5 and 8, respectively. For both minimum and maximum total score cases, the weak Alzheimer patient walks for 10 minutes (one unit time equals to one hour) for five times every one and half hours. High total score implies high physical and psychological damages. Although blood pressure and pulse rate are increased in accordance with increasing of the number of steps for the minimum total score case, these are not so increased for the maximum total score case.

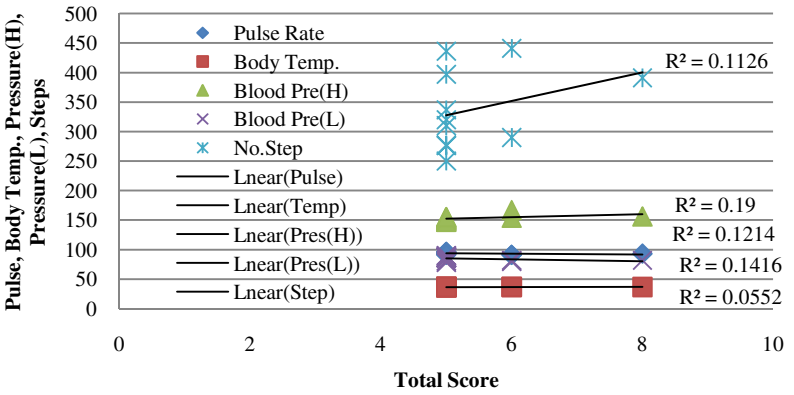
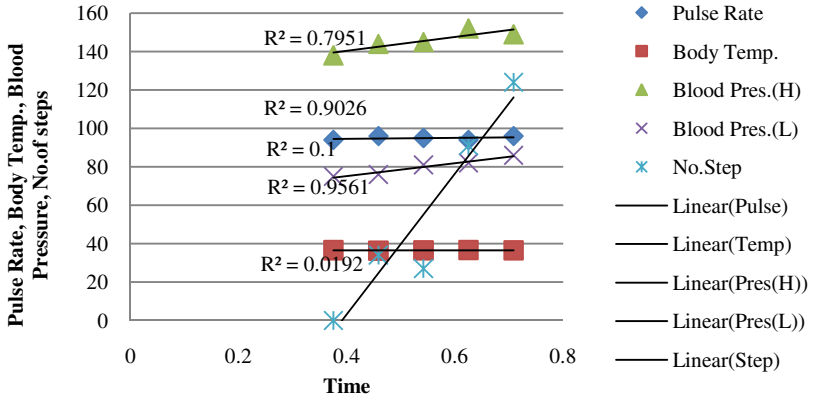
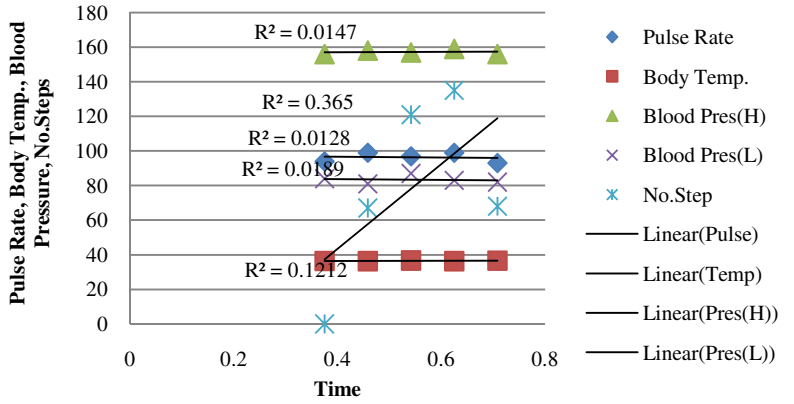


Fig. 14. Relation between the measured physical health condition and the self-evaluation of physical and psychological health conditions (Total Score) for the patient of weak Alzheimer



(a)Minimum total score of 5



(b)Maximum total score

Fig. 15. Physical health data of the weak Alzheimer of patient at the minimum and maximum total scores, 5 and 8, respectively

On the other hand, Fig.16 (a) and (b) shows physical health data of the patient who is good in health at the minimum and maximum total scores, 5 and 8, respectively. For both minimum and maximum total score cases, the patient who is good in health walks for 10 minutes (one unit time equals to one hour) for five times every one and half hours. Although blood pressure and pulse rate are quite stable in accordance with increasing of the number of steps for the minimum total score case, these are decreased for the maximum total score case.

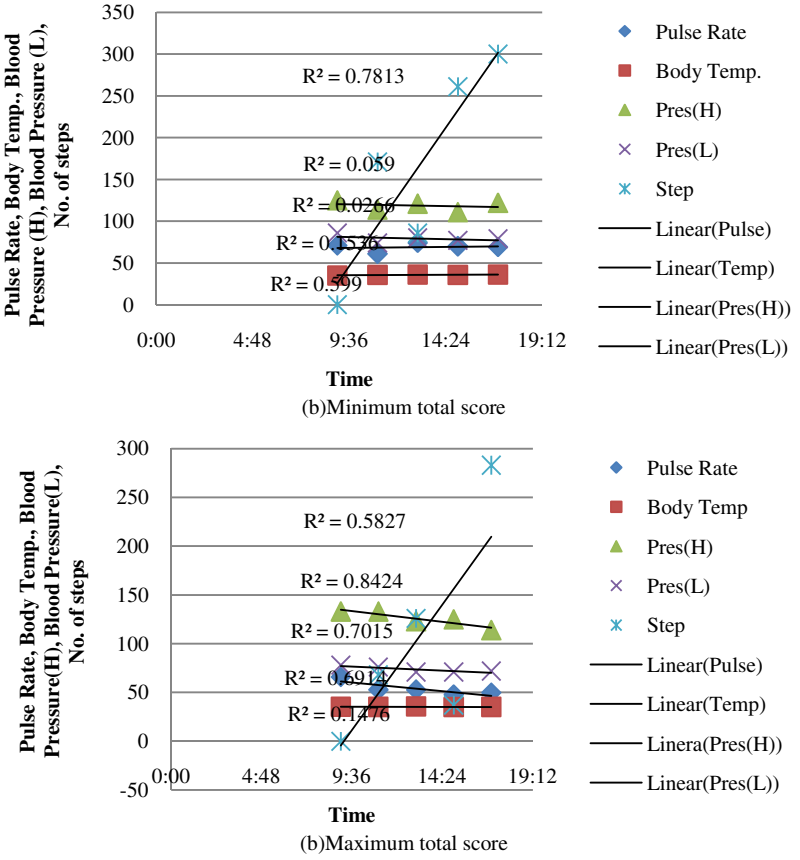


Fig. 16. Physical health data of the patient who is good in health at the minimum and maximum total scores, 7 and 11, respectively

As the results, the followings are concluded,

- Body temperature is relatively stable for a day
- In accordance with increasing of the number of steps, blood pressure (High and Low) is increased
- Even if the number of steps is increased and when blood pressure is stable, then physical and psychological health condition is good in health

- Even if the number of steps is increased and when blood pressure is decreases, then physical and psychological health condition is excellent in health
- There is a correlation between blood pressure (High and Low) and a combination of pulse rate and body temperature

3) Relation Between Blood Pressure and the OtherMeasured Physical Health Conditions

Using all measured physical health data, linear regressive analysis is conducted. Table 4 shows correlation matrix among physical and psychological health conditions. There is relatively large correlation between blood pressure and body temperature and pulse rate. Therefore, the coefficient body temperature and pulse rate multiplied by their correlation coefficients is proposed for regressive analysis. The result from the regressive analysis is shown in Fig.17. Although it is not so easy to measure blood pressure with small size of sensors, it can be estimated with measured body temperature and pulse rate.

Table 4. Correlation matrix among physical and psychological conditions

Body Temp.	Blood Pres.(H)	Blood Pres.(L)	Heart Beet	No.Steps	TotalA	TotalB
-0.0104	0.463	-0.245	0.133	-0.348	-0.171	0.809
	0.122	-0.166	-0.231	0.0321	0.0237	0.440
		-0.504	-0.0562	0.502	-0.482	-0.186
			0.161	0.198	-0.282	-0.420
				-0.387	-0.149	0.0421
					0.340	-0.0180
						0.0784

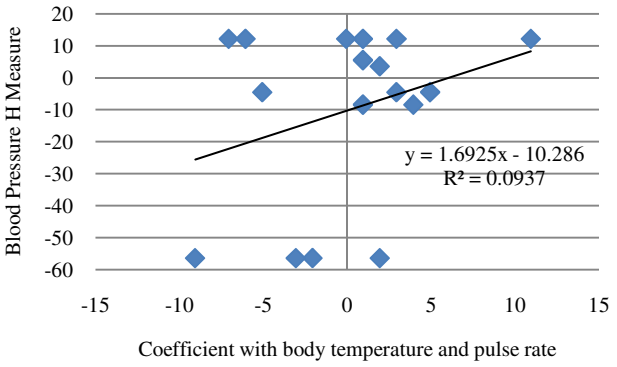


Fig. 17. Results from the regressive analysis between blood pressure high and the coefficient composed with body temperature and pulse rate

5) Measured Psychological Health Conditions

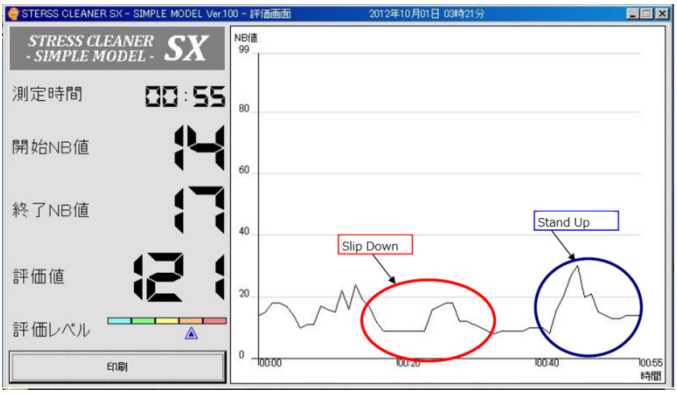
By using EEG analyzer tools, we analyze the fatigue effect between the condition when user is looking at one point and condition when user is looking at four points. In order to analyze fatigue effect, we use Peak Alpha Frequency: PAF [8]-[11.] It is possible to measure psychological status by using PAF derived from EEG signal.

Psychological health condition is measured with Bio Switch MCTOS of Brain Wave Measuring instrument (BM-Set1) manufactured by Technos Japan Co. Ltd. every one hour. Fig.18 shows an example of the measured data of relax indicator, NB value which is derived from EEG and EMG signals.

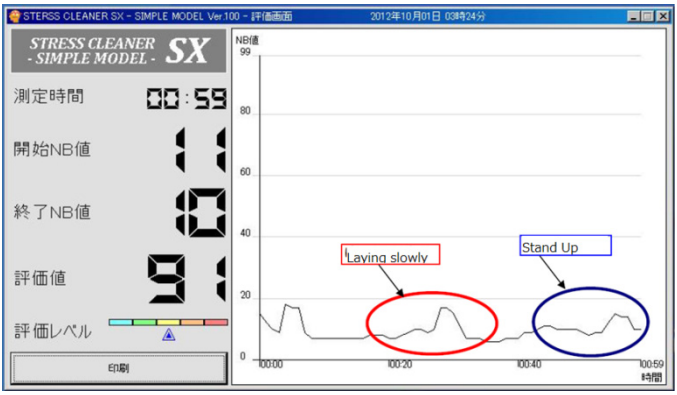
Fig. 18. Example of the measured data of relax indicator, NB value which is derived from EEG and EMG signals

Fig.19 (a) shows the NB value for the patient's action, sit down quickly and then stand up rapidly while Fig.19 (b) shows that for the patient's action of lay down slowly and the stand up normally. Meanwhile, Fig.19 (c) shows NB value for the patient's action, stand up, lay down slowly, stand up and then sit down slowly.

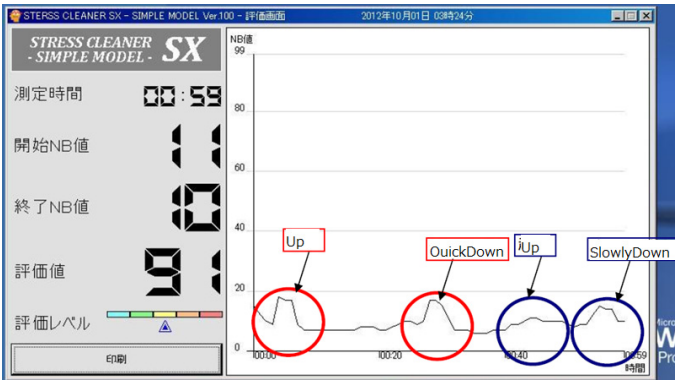
NB values change for the every event of the patient's action. It is found that the NB value changes for slow action are smaller than those for quick action. It is also found that the NB value changes for standup action is much greater than those for lay down and sit down actions as shown in Fig. 20. These NB value change characteristics are almost same for patients A, B, and C. There are the different characteristics between A, B, C, and D as shown in Fig.21.



(a)Case #1



(b)Case #2

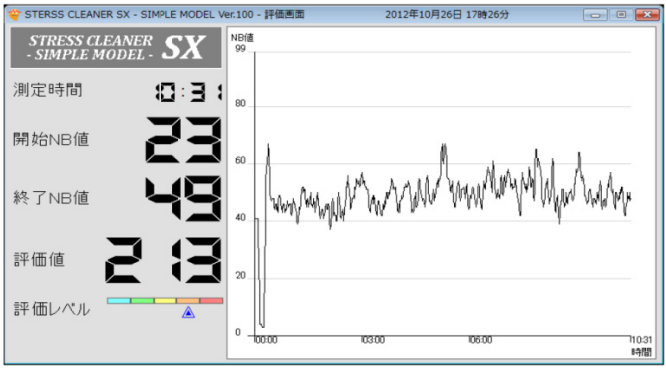


(c)Case #3

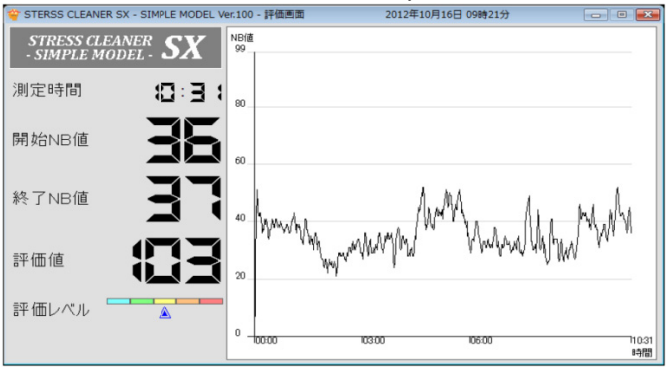
Fig. 19. NB values for the different patient's actions

Fig. 20. Example of NB value changes for the patient action, sit down quickly, standup, slowly sit down and then standup

Fig.21 (a) shows NB value changes for the patient A while Fig.21 (b) shows those for the patient D.



(a)Patient with normal healthy condition



(b)Patient with weak Alzheimer

Fig. 21. NB value changes for the patients

It is concluded that the patient with weak Alzheimer feels much stress due to the actions rather than the patient in normal healthy condition. It also is found that there is no difference of psychological health condition due to age. There is no psychological health condition difference due to gender.

Example of raw EEG signal is shown in Fig.22. Fig.22(a) shows the locations of electrodes and Fig.22(b) shows the examples for the different two locations of EEG.

It is confirmed that it may reduce physical health monitoring error due to psychological status changes. Even for the healthy persons, it occur such errors. For instance, heart beat pulse rate and blood pressure is used to be increased when medical doctor or nurse measures. By using EEG signal analyzed results, such errors may be corrected or at least it can be omitted from the monitored data.

4 Conclusions

Method and system for frequent health monitoring as vital signs with psycholo9gical status monitoring for search and rescue of handicapped person is proposed. Heart beat pulse rate, body temperature, blood pressure, blesses and consciousness is well known vital signs. In particular for Alzheimer diseased persons, handicapped peoples, etc. it is very important to monitor vital signs in particular in the event of evacuation from disaster occurred areas together with location and attitude information. Then such persons who need help for evacuation can be survived. Through experiments wearing the proposed sensors with three normal persons including male and female, young and elder persons and one diseased person, it is found that the proposed system is useful. It is also found that the proposed system can be used for frequent health condition monitoring. Furthermore, physical health monitoring error due to psychological condition can be corrected with the proposed system.

Wearable physical and psychological health monitoring system is proposed. All the sensors which allows monitoring blood pressure, body temperature, pulse rate, measuring sensor for the number of steps, calorie consumption, EEG, GPS receiver, WiFi or Wireless LAN receiver for location estimation, accelerometer are attached to the human body. Measured data are transferred to the mobile devices through Bluetooth. Mobile devices are connected with Internet terminals through WiFi, or Wireless LAN. Thus these physical and psychological health data are collected in the Information Collection Center. Thus those who are wearing the sensors can get a help from the designated volunteer when evacuation from disaster areas.

From the experimental results, the followings are concluded,

- Body temperature is relatively stable for a day
- In accordance with increasing of the number of steps, blood pressure (High and Low) is increased
- Even if the number of steps is increased and when blood pressure is stable, then physical and psychological health condition is good in health
- Even if the number of steps is increased and when blood pressure is decreases, then physical and psychological health condition is excellent in health

- There is a correlation between blood pressure (High and Low) and a combination of pulse rate and body temperature
- It is concluded that the patient with weak Alzheimer feels much stress due to the actions rather than the patient in normal healthy condition. It also is found that there is no difference of psychological health condition due to age. There is no psychological health condition difference due to gender.

Acknowledgment. The author would like to thank all the patients who are contributed to the experiments conducted. The author also would like to thank Professor Dr. Takao Hotokebuchi, President of Saga University for his support this research works.

References

1. Arai, K., Sang, T.X.: Decision making and emergency communication system in rescue simulation for people with disabilities. International Journal of Advanced Research in Artificial Intelligence 2(3), 77–85 (2013)
2. Arai, K., Sang, T.X., Uyen, N.T.: Task allocation model for rescue disable persons in disaster area with help of volunteers. International Journal of Advanced Computer Science and Applications 3(7), 96–101 (2012)
3. Arai, K., Sang, T.X.: Emergency rescue simulation for disabled persons with help from volunteers. International Journal of Research and Review on Computer Science 3(2), 1543–1547 (2012)
4. Arai, K., Sang, T.X.: Fuzzy Genetic Algorithm for Prioritization Determination with Technique for Order Preference by Similarity to Ideal Solution. International Journal of Computer Science and Network Security 11(5), 229–235 (2011)
5. Arai, K., Mardiyanto, R.: Evaluation of Students' Impact for Using the Proposed Eye Based HCI with Moving and Fixed Keyboard by Using EEG Signals. International Journal of Review and Research on Computer Science(IJRRCS) 2(6), 1228–1234 (2011)
6. Arai, K.: Wearable healthy monitoring sensor network and its application to evacuation and rescue information server system for disabled and elderly person. International Journal of Research and Review on Computer Science 3(3), 1633–1639 (2012)
7. Arai, K.: Wearable Physical and Psychological Health Monitoring System. In: Proceedings of the Science and Information Conference 2013, London, UK, October 7-9 (2013)
8. Felzer, T., Nordmann, R.: Alternative text entry using different input methods. In: Proceedings of the 8th International ACM SIGACCESS Conference on Computers and Accessibility, pp. 10–17 (2006)
9. Majaranta, P., Ahola, U., Špakov, O.: Fast gaze typing with an adjustable dwell time. In: Proceedings of the 27th International Conference on Human Factors in Computing Systems, pp. 357–360 (2009)
10. Ng, S.C., Raveendran, P.: EEG Peak Alpha Frequency as an Indicator for Physical Fatigue. In: Proceedings of 11th Mediterranean Conference on Medical and Biomedical Engineering and Computing 2007, pp. 517–520 (2007)
11. Klimesch, W.: EEG alpha and theta oscillations reflect cognitive and memory performance: A review and analysis. Brain Research Reviews, Rev. 29(2-3), 169–195 (1999)

The World as Distributed Brain with Spatial Grasp Paradigm

Peter Simon Sapaty

Institute of Mathematical Machines and Systems
National Academy of Sciences of Ukraine

Abstract. A novel ideology and supporting distributed information and control technology will be presented that can convert any distributed system into an integral spatial brain exhibiting global awareness, consciousness and will, pursuit of global goals, and recovery from indiscriminate damages. The technology is based on implanting of universal intelligent modules into key system points which interacting via any available channels cooperatively interpret high-level Spatial Grasp Language (SGL). The system and mission scenarios in SGL can start from any point, subsequently covering the system at runtime in a self-modifying and self-replicating manner. This can orient its local and global behavior in the way required, and establish spatial infrastructures also using other models and technologies.

Keywords: gestalt theory, spatial grasp language, networked language interpretation, global awareness, system integrity, unmanned systems, distributed brain.

1 Introduction

In our modern dynamic world we are meeting numerous irregular situations and threats where proper reaction on them can save lives and wealth and protect critical infrastructures. One of key problems for achieving suitable and quick solutions lies in adequate system organizations.

The traditional approach to system design, development and management supposes the system structure and system organization to be primary, created in advance, whereas global function and overall behavior appearing as secondary, like in Fig. 1.

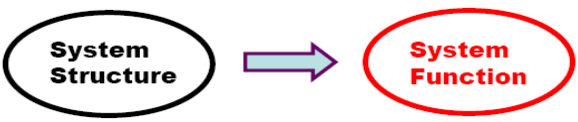


Fig. 1. Traditional approach to system design

The related systems are predominantly clumsy and static, they may often fail to adapt to dynamic and asymmetric situations. If the initial goals change, the whole system may have to be partially or even completely redesigned and reassembled.

L. Chen, S. Kapoor, and R. Bhatia (eds.), *Intelligent Systems for Science and Information*,
Studies in Computational Intelligence 542,
DOI: 10.1007/978-3-319-04702-7_4,

Adjusting the already existing system to new goals and functionality may result in a considerable loss of system's integrity and performance.

Within the approach offered in this paper, the global function and overall behavior are considered, as much as possible, to be primary, and the system structure and organization (command and control including) to be secondary—the latter as a dynamic derivative of the former (see Fig. 2).

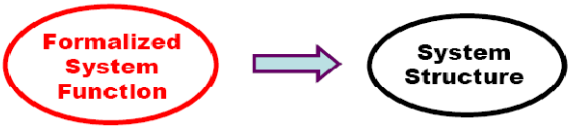


Fig. 2. The system organization considered

The advantages of such (actually, the other way round) organization include high potential flexibility of runtime system creation and management, especially in quick responses to asymmetric events. This allows us formulate top semantics of the needed reaction on world events in a special high level language, shifting most of traditional organizational routines to automated up to fully automatic implementation, with effective engagement of unmanned systems. The underlying paradigm, known as "over-operability" [1,2], resembles of how human brain consciously grasps, comprehends, and manages active distributed worlds with integral, gestalt-like [3, 4, 5] vision, rather than by cooperating parts (or "agents") as traditionally believed [6]. But unlike the brain operation, this approach has been put on a highly parallel and fully distributed technological platform giving it advantages in solving problems in very large and complexly interconnected domains where biological knowledge processing and intuition may fail.

The development history and various philosophical and technological aspects of this Spatial Grasp Technology (SGT), as well as detailed descriptions of the researched areas can be found in existing publications, including [7-20]. Some chronological stages of the technology development performed and/or supervised by the author in different countries are shown in Fig. 3.

1. Simulating power networks: 66-71
2. Creating first distributed computer networks: 69-79
3. Parallel computations, mobile agents: 71-80
4. Nationwide global networked management system: 75-80
5. Intelligent hardwired computers, serially produced: 70-80
6. Distributed macro-pipeline supercomputer: 78-85
7. Wave model of distributed parallel computations: 74-90
8. WAVE system at Basic AI Lab in Czechoslovakia: 84-87
9. Alexander von Humboldt WAVE project, Germany: 88-90
10. Intelligent network management project for Siemens: 90-93
11. WAVE system in Germany, UK, US, Canada: 90-00
12. Distributed simulation of battlefields: UK, DIS project US: 93-98
13. Cooperative robotics, Japan: 01-05
14. European patent: 90-93
15. John Wiley books: 99, 05, new one in progress

Fig. 3. History of SGT development

The current paper summarizes and updates the basics of SGT at its current development stage, and provides examples from different application areas that altogether can symbolically treat the world under SGT as an integral spatial brain capable of solving important and hot problems.

2 The Spatial Grasp Model

The paradigm offered is based on formalized parallel incremental, wavelike, and seamless navigation, coverage or grasping of distributed physical and virtual spaces, as symbolically shown in Fig. 4.

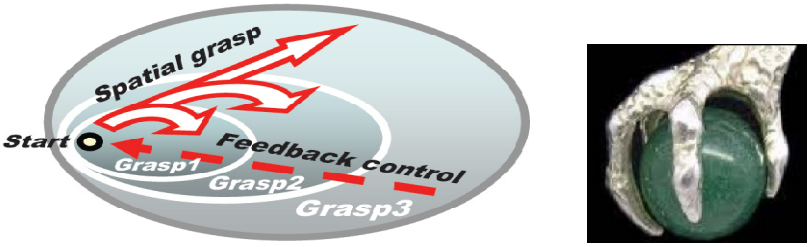


Fig. 4. Incremental grasping of distributed worlds: a) virtual interpretation, b) symbolic physical analogy

It represents holistic, gestalt principles [3-5] rather than cooperating agents [6], having strong psychological and philosophical background, reflecting, for example, how humans (especially top commanders) mentally plan, comprehend and control complex operations in distributed environments.

The approach in general works as follows. A network of universal control modules U, embedded into key system points (like humans, robots, smart sensors, mobile phones, laptops, etc.), collectively interprets mission scenarios expressed in Spatial Grasp Language (SGL), as shown in Fig. 5. These scenarios, capable of representing any parallel and distributed algorithms, can start from any node while covering the whole system or its parts needed at runtime.

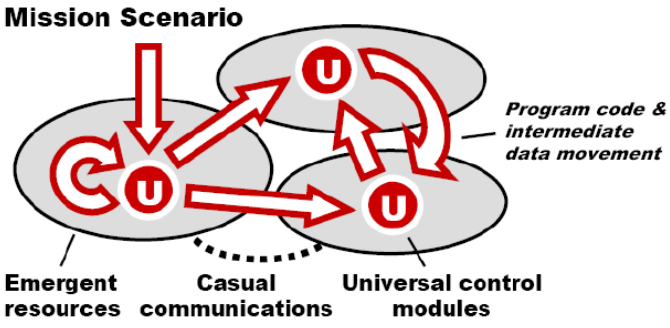


Fig. 5. Collective scenario execution in dynamic environments

SGL scenarios, often expressing top semantics of spatial operations, are very compact and can be created on the fly. Different scenarios can cooperate or compete in a networked space as overlapping fields of solutions. Self-spreading scenarios can create runtime knowledge infrastructures distributed between system components, as shown in Fig. 6. These can effectively support *distributed databases, advanced command and control, global situation awareness, as well as any other computational or control models*.

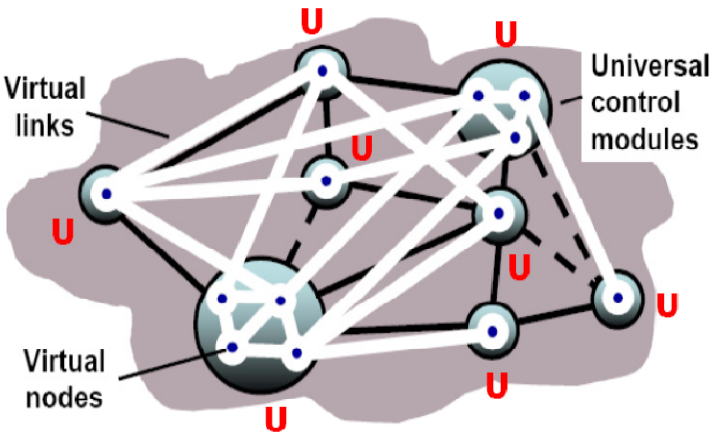


Fig. 6. Creating distributed knowledge infrastructures

3 Spatial Grasp Language

SGL differs radically from traditional programming languages. It allows us to directly move through, observe, and make any actions and decisions in fully distributed environments. SGL directly operates with:

- *Virtual World (VW),* which is finite and discrete, consisting of nodes and semantic links between them.
- *Physical World (PW),* infinite and continuous, where each point can be identified and accessed by physical coordinates, with certain precision.
- *Virtual-Physical World (VPW),* finite and discrete, similar to VW, but associating some or all virtual nodes with certain PW coordinates.
- *Execution world (EW),* consisting of active doers, which may be humans, robots, or any machines capable of operating on the previous three worlds.

Any sequential or parallel, centralized or distributed, stationary or mobile algorithm operating with information and/or physical matter can be written in SGL on any levels, including the highest semantic or the lowest ones, the latter detailing system partitioning, composition, infrastructures, and management. Its universal recursive structure is shown in Fig. 7.

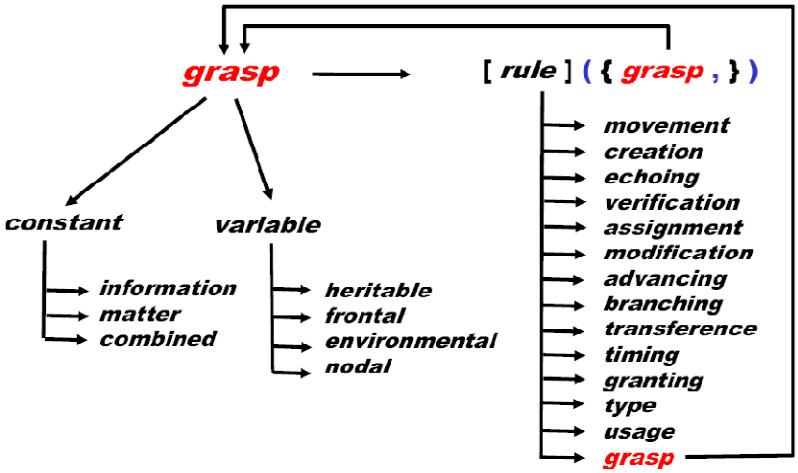


Fig. 7. The SGL recursive syntax

3.1 SGL Main Features

- An SGL scenario develops as parallel transition between sets of progress points (or *props*).
- Starting from a prop, an action may result in new props (which may be multiple) or remain in the same prop.
- Each prop has a resulting *value*, which may be arbitrarily complex, and resulting *state* (one of: `thru`, `done`, `fail`, and `fatal`).
- Different actions may evolve independently or interdependently from the same prop, splitting and parallelizing in space.
- Actions may also spatially succeed each other, with new ones applied sequentially or in parallel from all props reached by the previous actions.
- Elementary operations can directly use states and values of props reached by other actions whatever complex and remote they might be.
- Any prop can associate with a position in PW or a node in VW, VPW, or EW.
- Any number of props can be simultaneously linked with the same points of the VW, VPW, or EW worlds, sharing local information at them.
- Staying with the world points, it is possible to directly access and impact local world parameters in them, whether virtual or physical.
- Overall organization and control of the breadth and depth space navigation and coverage is provided by SGL rules which can be nested.

3.2 SGL Rules

The basic SGL construct, *rule*, of the language may represent any action or decision (all of the same level and status) and can, for example, be as follows:

- Elementary arithmetic, string or logic operation.
- Hop in a physical, virtual, execution, or combined space.

- Hierarchical fusion and return of (remote) data.
- Distributed control, both sequential and parallel.
- A variety of special contexts for navigation in space influencing embraced operations and decisions.
- Type or sense of a value or its chosen usage, guiding automatic interpretation.
- Creation or removal of nodes and links in distributed knowledge networks.
- A rule can be a compound one, integrating a number of other rules; it can also be defined in a result of local or global operations of arbitrary complexity.

3.3 SGL Spatial Variables

Working in fully distributed physical, virtual, or executive environments, SGL has different types of variables, called *spatial*, effectively serving multiple cooperative processes:

- *Heritable variables* – these are starting in a prop and serving all subsequent props, which can share them in both read & write operations.
- *Frontal variables* – are an individual and exclusive prop's property (not shared with other props), being transferred between the consecutive props and replicated if from a single prop a number of other props emerge.
- *Environmental variables* – are accessing different elements of physical and virtual words when navigating them, also a variety of parameters of the internal world of SGL interpreter.
- *Nodal variables* – allow us to attach an individual temporary property to VW, VPW, and EW nodes, accessed and shared by all activities currently associated with these nodes.

These types of variables, especially when used together, allow us to create spatial algorithms working *in between components* of distributed systems rather than *in* them, allowing for flexible, robust, and self-recovering solutions. Such algorithms can freely replicate, spread and migrate in distributed environments (partially or as an organized whole), always preserving global integrity and overall control.

3.4 List of SGL Main Constructs

SGL full description is as follows where syntactic categories are shown in italics, vertical bar separates alternatives, the construct in square brackets is optional, and the parts in braces indicate zero or more repetitions. The remaining characters and words are the language symbols.

grasp	→	*constant* | *variable* | [*rule*] ({ ***grasp***, })
constant	→	*information* | *matter*
variable	→	*heritable* | *frontal* | *nodal* | *environmental*
rule	→	*movement* | *creation* | *echoing* | *verification* | *assignment* | *modification* | *advancing* | *branching* | *transference* | *timing* | *granting* | *type* | *usage* | { ***grasp_*** }

information → '{*character*}' | *number* | *coordinate* | *special*
matter → "{*character*}"
movement → hop | move | shift
creation → create | linkup | delete | unlink
echoing → state | order | rake | element | average | count | sort up | sort down | reverse | add | subtract | multiply | divide | degree | separate | unite | attach | append | remove | common | content | index | access
verification → equal | not equal | less | less or equal | more | more or equal | empty | nonempty | belongs | not belongs | intersects | not intersects
assignment → assign | assign peers
modification → inject | replicate | split
advancement → advance | slide | repeat | destination
branching → branch | sequence | parallel | if | or | and | choose | cycle | loop | sling | whirl
transference → run | call | input | output
timing → sleep | remain
granting → supervise | free | blind | lift | none | stay | seize
type → heritable | frontal | nodal | environmental | matter | number | string | address | coordinate
usage → name | place | center | range | time | speed | doer | node | link | unit
heritable → H{*alphameric*}
frontal → F{*alphameric*}
nodal → N{*alphameric*}
environmental → TYPE | CONTENT | NAME | ADDRESS | QUALITIES | WHERE | BACK | PREVIOUS | DOER | RESOURCES | LINK | DIRECTION | WHEN | TIME | SPEED | STATE | VALUE | COLOR | IN | OUT
special → thru | done | fail | fatal | infinite | nil | first | last | min | max | random | any | all | direct | no back | global | local | fringe | synch | asynch | virtual | physical | executive | existing | first come | reachable

To simplify SGL programs, traditional to existing programming languages abbreviations of operations and delimiters can be used too, substituting certain rules as in the examples throughout this text, always remaining, however, within the general syntactic structure shown in Fig. 7 and above.

3.5 Elementary Examples in SGL

- Assign a sum of values to the variable Result:

```
assign(Result, add(27, 33, 55.6))
```

- Move to two physical locations in parallel:

```
move(location(x1, y3),location(x5, y8)).
```

- Create isolated virtual node Peter:

```
create(Peter)
```

- Extend Peter as father of Alex, the latter as new node:

```
advance(hop(Peter), create(+fatherof, Alex))
```

Graphical representation of these examples with possibilities of using shorter notations similar to traditional languages is shown in Fig. 8.

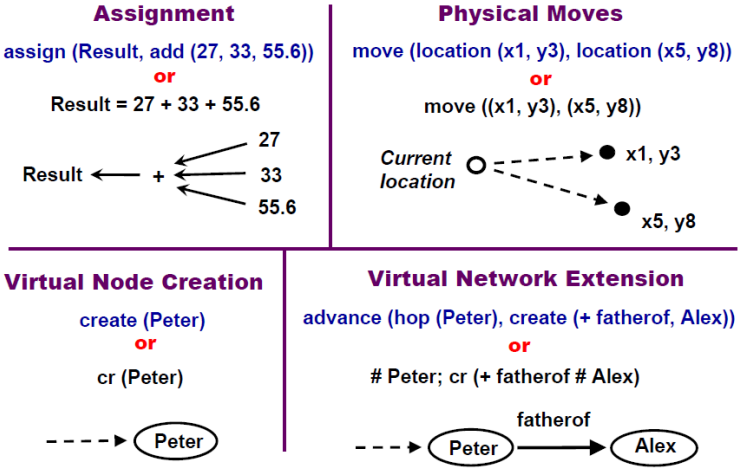


Fig. 8. Elementary programming examples

4 Distributed SGL Interpreter

4.1 General Organization

The internal organization of SGL interpreter [12, 13, 14] (which can have software, hardware or combined implementation) is shown in Fig. 9. The interpreter consists of a number of specialized modules working in parallel and handling & sharing specific data structures supporting both persistent virtual worlds and temporary data and hierarchical control mechanisms. The whole network of the interpreters can be mobile and open, changing at runtime the number of nodes and communication structure between them. Copies of the interpreter can be concealed if are to operate in hostile environments, allowing us to analyze and impact the latter in a stealth manner, if needed.

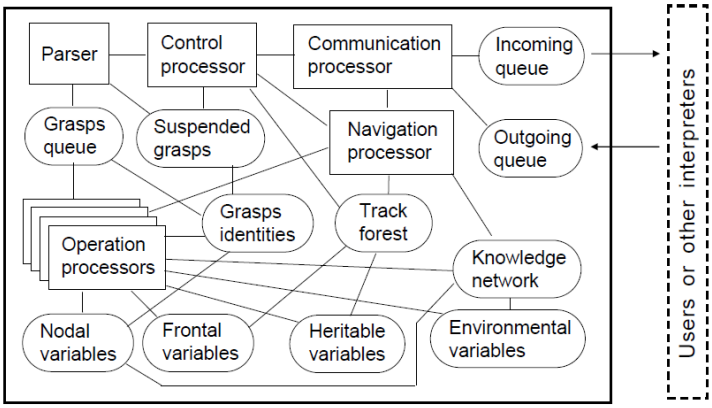


Fig. 9. Organization of SGL interpreter

4.2 Spatial Track System

The "nerve system" of the distributed interpreter is its spatial track system with its parts kept in the Track Forest memory of local interpreters. These being logically interlinked with similar parts in other interpreter copies, forming altogether global control coverage. This forest-like distributed track structure enables for hierarchical control as well as remote data and code access, with high integrity of emerging parallel and distributed solutions, without any centralized resources. The dynamically created track trees (generally: forests), spanning the systems in which SGL scenarios evolve, are used for supporting spatial variables and echoing & merging different types of control states and remote data.

The main components of the track system are shown in Fig. 10.

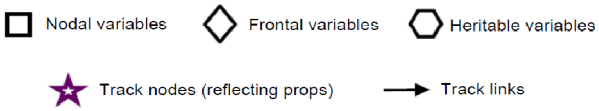


Fig. 10. Track system components

Track operation has alternating forward and backward (or echo) activity. In the forward process, as in Fig. 11, new steps of development of SGL scenarios are reflected by creating new prop nodes (sequentially or in parallel if there are parallel scenario branches), with the previous props and succession links between them retained in the evolving track tree as a history. This track tree also keeps heritable, nodal and frontal variables connected with prop nodes, with frontal variables moving between successive props and always staying with the last, or fringe, props.

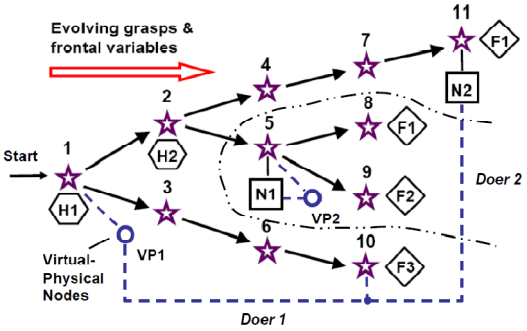


Fig. 11. Tracks creation in forward grasping

After completion of some stage of SGL scenario, the track system can return the generalized control state based on the termination states in all fringe props, which can influence invocation of the next scenario stage. The track system can also self-optimize in the parallel echo processes, as in Fig. 12.

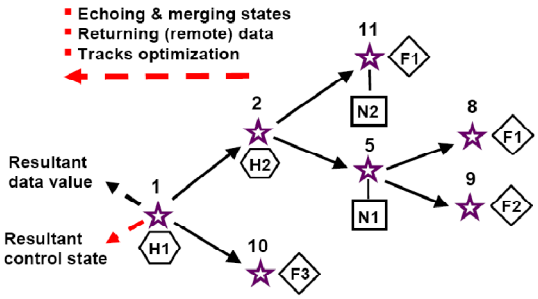


Fig. 12. Track-based echoing and optimization

The optimized track system after the previous scenario stage can route further grasps to the positions in physical, virtual or combined spaces reached by the previous grasps, i.e. defined by the fringe track nodes, uniting them with the frontal variables left there by the preceding stage, as in Fig. 13.

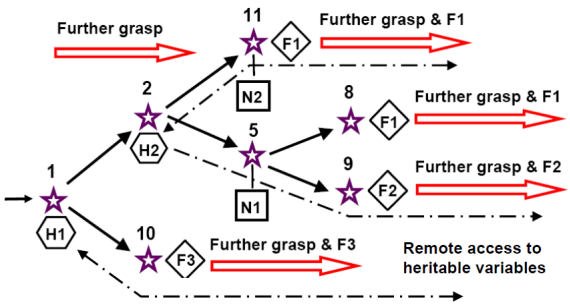


Fig. 13. Development of further grasps

4.3 SGL Interpreter as a Universal Spatial Machine

The dynamically networked SGL interpreters are effectively forming a sort of *universal parallel spatial machine* (as shown in Fig. 14) capable of solving any problems in a fully distributed mode, without any special central resources. "Machine" rather than computer or brain (as symbolically used in the paper's title), as it can operate with matter too and can move partially or as a whole in physical environment, possibly, changing its distributed shape and space coverage. This machine can operate simultaneously on many mission scenarios which can be injected at any time from its arbitrary nodes.

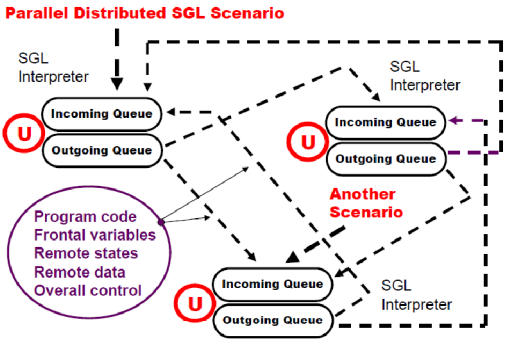


Fig. 14. SGL interpretation network as a universal spatial machine

Installing communicating SGL interpreters into mobile robots (ground, aerial, surface, underwater, space, etc.) on top of their existing functionality allows us to organize effective group solutions (incl. any swarming) of complex problems in distributed physical spaces in a clear and concise way, effectively shifting traditional management routines to automatic levels. Human-robot interaction and gradual transition to fully unmanned systems are drastically assisted too.

5 Infrastructure Operations in SGL

Finding connectivity, different paths, weak and strong components, or any substructures are of growing importance in large dynamic systems, both friendly and hostile. Some examples in SGL are presented below. Solutions of other tasks, including fundamental graph and network problems expressed in the previous versions of SGL (like WAVE and DSL) can be found, for example, in [11, 12].

5.1 Finding Shortest Path in Parallel

A solution for finding the shortest path between two infrastructure nodes (let them be `a` and `e`) combining initial shortest path tree creation and subsequent shortest path collection in an arbitrary network (shown in Fig. 15) can be expressed by the following SGL scenario.

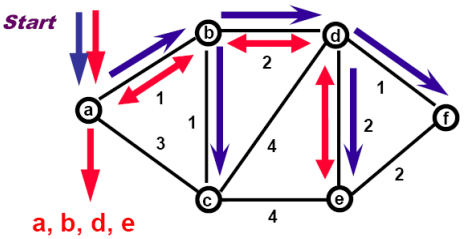


Fig. 15. Finding shortest path in parallel distributed mode

```
frontal(Far, Path); nodal(Distance, Before); hop(a); sequence(
(Distance = 0;
 repeat(hop(all links); Far += LINK;
       or(Distance == nil, Distance > Far);
        Distance = Far; Before = BACK)),
 output(
   Path = NAME;
   repeat(hop(all links); BACK == Before;
         Path &= NAME; if(NAME = e, done))))
```

The result obtained in node a for the network of Fig. 15 will be: (a, b, d, e). It has been found by navigating the network of weighed links in parallel and fully distributed mode, without any central resources.

5.2 Finding Weakest Points in Parallel

To find the weakest nodes in an infrastructure, like articulation points (see Fig. 16), which when removed split it into disjoint parts, the following program suffices, operating in parallel and fully distributed mode too (resulting in node d).

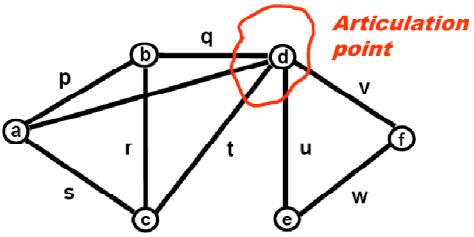


Fig. 16. Finding weakest or articulation points

```
hop(all nodes); COLOR = NAME; nodal(Mark = 1);
and((hop(random, all links);
     repeat(seize(Mark == nil; Mark = 1);
           hop(all links))),
    (hop(all links); Mark == nil),
    output(NAME))
```

This distributed program works in the following steps:

- Starting in each node with personal color, marking it.
- Parallel marking all accessible part of the network with personal color from a randomly chosen neighbor, excluding itself from the marking process.
- Checking if the current node solely connects parts of the network.

5.3 Finding Strongest Parts in Parallel

Cliques (or maximum fully connected sub-graphs of a graph, as in Fig. 17) may be considered as strongest parts of a system. They all can be found in parallel by the following simple program resulting for Fig. 17 in cliques: `(a, b, c, d)`, `(c, d, e)`, and `(d, e, f)`.

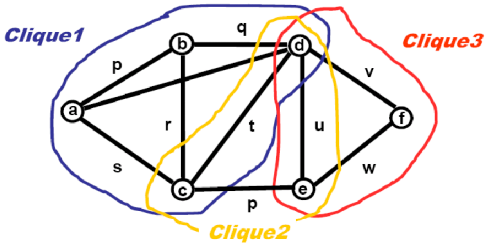


Fig. 17. Finding strongest parts or cliques

```
frontal(Clique); hop(all nodes); Clique = NAME;
repeat(
  hop(all links); not belongs(NAME, Clique);
  if(and_parallel(hop(any links, Clique)),
     if(BACK > NAME, Clique &= NAME, done),
     fail));
if(count(Clique) >= 3, output(Clique))
```

The program operates in the following steps:

- Starting in each node.
- Growing potential clique in a unique node order until possible.
- Reporting the clique grown, with threshold size given.

5.4 Finding Arbitrary Structures by Parallel Pattern Matching

Any structures in distributed networked systems with any topologies can be found by describing them in SGL, like the one in Fig. 18, which can be applied from any network node, evolving subsequently in a parallel replication and pattern-matching mode. The following SGL program reflecting the search pattern (template) of Fig. 18 with variable nodes X1 to X6 is based on a path through all template's nodes, starting from node X1.

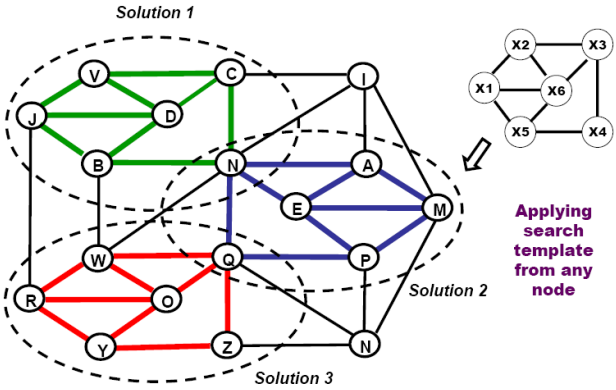


Fig. 18. Finding arbitrary structures in arbitrary networks in a fully distributed mode

```
frontal(Match); hop(all nodes);
repeat_5(append(Match, NAME); hop(all links);
        not belong(NAME, Match));
if(and_parallel(hop(any link, Match[2, 3])),
   (append(Match, NAME); hop(all links, Match[1]);
    hop(any link, Match[5]); OUT = Match))
```

Three substructures found corresponding to the variables X1 to X6 will be as follows:

```
(X1, X2, X3, X4, X5, X6)  →
  (J, V, C, N, B, D), (M, A, N, Q, P, E), (R, W, Q, Z, Y, O)
```

6 Providing Global Awareness

Establishing global electronic supervision over any distributed systems, SGT effectively provides global awareness of complex situations in them. The latter, for example, may be useful for discovering, collecting and distributing hostile targets seeing locally from different points, as shown in Fig. 19 and by the SGL program that follows.

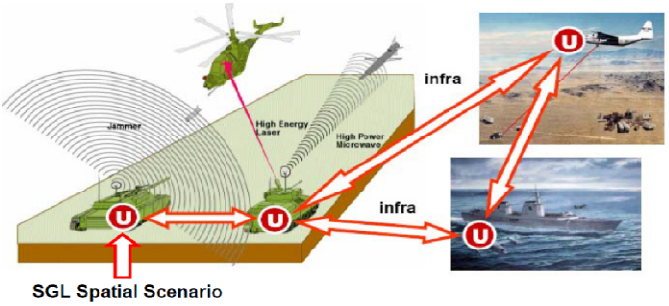


Fig. 19. Providing global awareness and targeting in a distributed space

```
sequence(
  repeat(linkup_first(infra, reachable)),
  loop(frontal(Seen) = repeat(
        free(detect(targets)), hop_first(infra));
      repeat(free(select_shoot(Seen)),
           hop_first(infra))))
```

This looping, replicating, and self-spreading distributed program provides global collection of possible targets throughout the region of concern. It subsequently distributes accumulated targets back to local units, the latter deciding which targets to shoot individually. The program can start from any system component having SGL interpreter installed.

7 Cooperative Robotics

After embedding SGL interpreters into robotic vehicles, we can provide any needed detailed collective behavior of them—from loose swarming to a strictly controlled integral unit obeying external orders. Any mixture of different behaviors within the same scenario can be easily programmed too. Expressing different simple scenarios in SGL and their integration into a more complex, combined one may be as follows.

- Swarm movement scenario, starting from any unit (let us call this **swarm_move**):

```
hop(all nodes);
nodal(Limits = (dx(0,8), dy(-2,5)), Range = 500, Shift);
repeat(Shift = random(Limits);
      if(empty(hop(Shift, Range), move(Shift)))
```

- Finding topologically central unit and hopping into it, starting from any unit (**find_hop_center**):

```
frontal(Aver) = average(hop(all nodes); WHERE);
hop(
  min(hop(all nodes); distance(Aver, WHERE) & ADDRESS):2)
```

- Creating runtime infrastructure starting from the central unit found (**infra_build**):

```
stay(repeat(linkup_first(+infra, reachable)))
```

- Regular targets collection & distribution & impact, from the central unit found (**collect_distribute_impact**):

```
loop(nonempty(frontal(Seen) =
      repeat(free(detect(targets)), hop(+infra)));
    repeat(free(select_shoot(Seen)), hop(+infra)))
```

- Removing previous infrastructure (before creating a new one), starting from any unit (**infra_remove**):

```
stay(hop(all nodes); remove(all links))
```

- And the resultant combined solution (integrating previous SGL programs named in bold), starting from any unit, will be as:

```
branch(
 swarm_move,
 repeat(
   find_hop_center; infra_remove; infra_build;
   or_parallel(collect_distribute_impact,
              sleep(delay))))
```

The obtained resultant scenario combines loose, randomly oriented swarm movement in a distributed space with periodic finding and updating of topologically central unit and setting-updating runtime hierarchical infrastructure between the units. This infrastructure controls observation of distributed territory while collecting potential targets, distributing them back to the vehicles for local selection and impact (a related snapshot, say, for aerial vehicles is shown in Fig. 20).

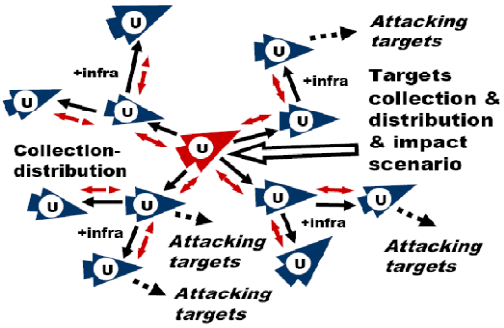


Fig. 20. Collecting, disseminating and impacting targets by an unmanned team

8 Collective Evacuation from a Disaster Zone

In case of major disasters (like earthquakes, hurricanes, flooding, landslides, forest fires, etc.) many people, especially elderly and handicapped, may need a special and urgent help. A related SGL scenario, setting up coordinated massive evacuation from the disaster zone, may be activated by any person caught by such an event or by a special emergency organization. This scenario can be regularly issuing instructions to individuals on where and how to move (say, via mobile phones if still working).

A chained collective movement through the safe passage in a disaster zone is shown by the program below and in Fig. 21, where individuals move in a coordinated (with each other and by the waypoints supplied) way. Only the first individual in this chain is a pure leader (directly following the waypoints), and the last one is a pure follower, whereas all others combine both functionalities (thus moving right after the previous person and directly followed by the next person).

```
cycle(N += 1; assign(free individual, create_node(N)));
(NAME == 1; Waypoints = (w1, w2, w3, … );
```

```
      loop(output('move to', withdraw(Waypoints, 1));
    wait(input == 'ok'))),
(NAME != 1; sling(
  Leader = (hop(direct, NAME - 1); WHERE);
  output(NAME, 'move toward', direction(WHERE, Leader));
  wait(input == 'ok')))
```

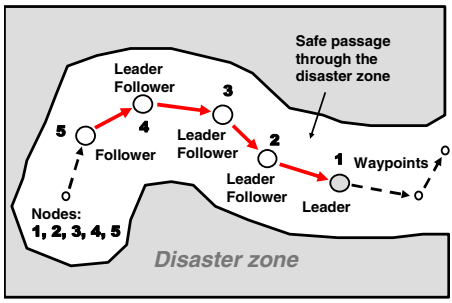


Fig. 21. Chained evacuation from a disaster zone

9 Tracking of Moving Individuals

The elderly people can often be lost, especially in crowded cities. The SGL scenario below shows how whereabouts of people can be regularly checked and traced by mobile spatial intelligence propagating in virtual world while following the movements of persons in physical world, as in Fig. 22.

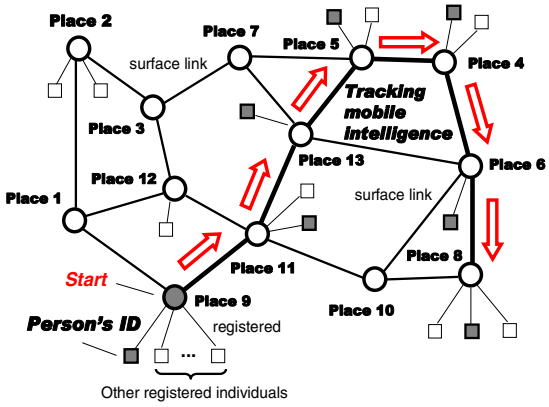


Fig. 22. Networked tracking of individuals

Current positions of individuals can be lifted by their contacts with "semantic surfaces" or by networked video cameras to which key pictures of the persons can be delivered by mobile intelligence accompanying them. The tracking intelligence can

analyze and accumulate behavior of the moving person, demand checking her current physical condition (like heartbeat, blood pressure, body temperature, etc.). It can also alarm the nearest medical facilities in case of irregularities. Many moving persons can be simultaneously and individually checked and served by the SGL scenario shown below.

```
frontal(ID = individual; History);
hop_nodes(places, all);
repeat(loop(belongs(ID, hop_links(registered, all));
          update(History, check (ID, condition));
          if(problems(ID, History),
             alarm(medical staff, nearest));
          delay(delay));
       hop_link(surface, all);
       belongs(ID, hop_link(registered, all)))
```

In a more global scale, tracking of multiple moving objects as threats rather than individuals by spatial intelligence in SGL is considered in the next section.

10 Global Protection of World Infrastructures

This scenario shows how the vast distributed area (which may represent critical infrastructures of a country, a continent, or even the whole world) with scattered, limited in range but communicating with each other sensors can be globally protected with the use of SGT, as shown in Fig. 23.

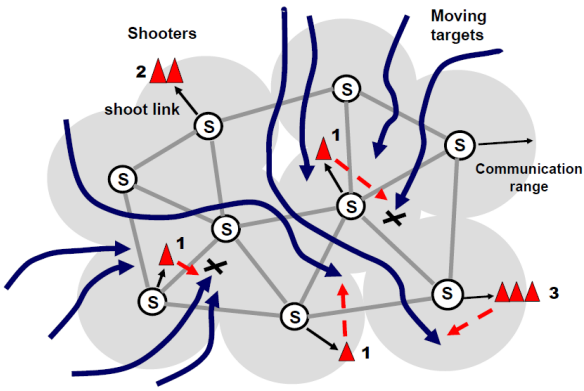


Fig. 23. Multiple objects tracking & shooting for infrastructure protection

The SGL program below is based on the following actions:

- Each sensor is regularly searching for new targets.
- Each new target is assigned individual tracking intelligence which propagates in distributed virtual space while following the target's movement in physical space.

- If there are available shooters in the vicinity and shooting is allowed and technically feasible, a kill vehicle is launched against the target, decreasing the number of available kill vehicles in the region.
- If the target is hit, it is removed from the observation.

```
frontal(Object, Threshold = ..., Range = ...);
nodal(Seen); hop_nodes(sensors, all);
whirl(
  Object = search(moving targets, not belongs(Seen));
  visibility(Object) > Threshold;
  release(repeat(
      append(Seen, Object);
      loop(visibility(Object) > Threshold;
          if((hop(shoot_link); CONTENT > 0;
              allowed (fire, Object);
              shoot(Object); decrement(CONTENT);
              success(shoot, Object)),
             (withdraw(Object, Seen); done)));
      withdraw(Object, Seen);
      max_destination(hop(Range); visibility(Object))))
```

Any further extension of this scenario can be easily provided in SGL including runtime optimization of the use of possibly scattered and limited impact facilities, or self-recovery after indiscriminate damages to the distributed control system.

11 Conclusions

We have briefed a new type of ideology and resulting networking technology aimed at establishing global control and supervision over distributed systems with any electronic means of communication and data processing embedded. Within the technology developed, it is possible to describe in a high-level spatial language any local and global operations and control in both physical and virtual worlds and set up and supervise their behavior needed. The approach also allows us to penetrate into other systems and their organizations, both friendly and hostile, analyze their internal structures and behavior and change them in the way required. On the implementation layer, SGT extensively employs replication and mobile code capability, allowing mission scenarios spread instructions, data and control in distributed worlds, spatially linking them with each other in a parallel pattern matching mode, effectively confronting other networking technologies, computer viruses including. The electronic communications between system components may be local, limited, unsafe, and changing at run time, but the self-spreading interpreted scenarios can always survive and fulfill objectives.

Taking into account the overwhelming world computerization, use of internet, and billions of mobile phone users, SGT can launch and supervise global world missions in a great variety of areas including environmental protection, education, economy, space research, security, and defense. The word “brain” has been used in the paper’s title rather symbolically, to stress capability of the technology to provide effective

global solutions in complex environments with high integrity, overall awareness, and pursuit of global goals—features often referred to localized human brain, which are placed in our case on highly parallel and fully distributed technological platform.

References

1. Sapaty, P.S.: Over-Operability in Distributed Simulation and Control. The MSIAC's M&S Journal Online 4(2) (2002) (Winter Issue)
2. Sapaty, P.: The Over-Operability Organization of Distributed Dynamic Systems for Asymmetric Operations. In: Proc. IMA Conference on Mathematics in Defence, Farnborough, UK (November 19, 2009)
3. Wertheimer, M.: Gestalt Theory, Erlangen. Berlin (1925)
4. Sapaty, P.: Gestalt-Based Ideology and Technology for Spatial Control of Distributed Dynamic Systems. In: International Gestalt Theory Congress, 16th Scientific Convention of the GTA, University of Osnabrück, Germany, March 26-29 (2009)
5. Sapaty, P.: Gestalt-Based Integrity of Distributed Networked Systems. In: SPIE Europe Security + Defence, bcc Berliner Congress Centre, Berlin, Germany (2009)
6. Minsky, M.: The Society of Mind, Simon and Schuster, New York (1988)
7. Sapaty, P.S.: Distributed Air & Missile Defense with Spatial Grasp Technology. Intelligent Control and Automation, Scientific Research 3(2) (2012)
8. Sapaty, P.S.: Withstanding Asymmetric Situations in Distributed Dynamic Worlds. In: Proc. 17th International Symposium on Artificial Life and Robotics (AROB 17th 2012), B-Con Plaza, Beppu, Oita, Japan (January 2012) (invited paper)
9. Sapaty, P.S.: Meeting the World Challenges with Advanced System Organizations. In: Cetto, J.A., Filipe, J., Ferrier, J.-L. (eds.) Informatics in Control Automation and Robotics. LNEE, vol. 85, pp. 29–46. Springer, Heidelberg (2011)
10. Sapaty, P.S.: Distributed Technology for Global Dominance. In: Suresh, R. (ed.) Proc. SPIE 6981, Defense Transformation and Net-Centric Systems, 69810T (2008)
11. Sapaty, P.S.: Ruling Distributed Dynamic Worlds. John Wiley & Sons, New York (2005)
12. Sapaty, P.S.: Mobile Processing in Distributed and Open Environments. John Wiley & Sons, New York (1999)
13. Sapaty, P.: A Distributed Processing System. European Patent No. 0389655, Publ. 10.11.93, European Patent Office (1993)
14. Sapaty, P.S., Corbin, M.J., Seidensticker, S.: Mobile Intelligence in Distributed Simulations. In: Proc. 14th Workshop on Standards for the Interoperability of Distributed Simulations, IST UCF, Orlando, FL (March 1995)
15. Sapaty, P., Klimenko, V., Sugisaka, M.: Dynamic Air Traffic Management Using Distributed Brain Concept. In: Proc. Ninth International Symposium on Artificial Life and Robotics (AROB 9th), Beppu, Japan (January 2004)
16. Sapaty, P., Sugisaka, M.: Optimized Space Search by Distributed Robotic Teams. In: Proc. World Symposium Unmanned Systems 2003, Baltimore Convention Center, USA, July 15-17 (2003)
17. Sapaty, P., Sugisaka, M., Delgado-Frias, J., Filipe, J., Mirenkov, N.: Intelligent management of distributed dynamic sensor networks. Artificial Life and Robotics 12(1-2), 51–59 (2008) ISSN: 1433-5298 (Print), 1614-7456 (Online)

18. Sapaty, P., Sugisaka, M.: Countering Asymmetric Situations with Distributed Artificial Life and Robotics Approach. In: Proc. Fifteenth International Symposium on Artificial Life and Robotics (AROB 15th 2010), B-Con Plaza, Beppu, Oita, Japan, February 5-7 (2010)
19. Sapaty, P., Kuhnert, K.-D., Sugisaka, M., Finkelstein, R.: Developing High-Level Management Facilities for Distributed Unmanned Systems. In: Proc. Fourteenth International Symposium on Artificial Life and Robotics (AROB 14th 2009), B-Con Plaza, Beppu, Japan, February 5-7 (2009)
20. Sapaty, P., Sugisaka, M., Delgado-Frias, J., Filipe, J., Mirenkov, N.: Intelligent management of distributed dynamic sensor networks. Artificial Life and Robotics 12(1-2), 51–59 (2008) ISSN: 1433-5298 (Print) 1614-7456 (Online)

Spatial Relation Approach to Fingerprint Matching

Gabriel Babatunde Iwasokun[1], Oluwole Charles Akinyokun[1], and Cleopas Officer Angaye[2]

[1] Department of Computer Science, Federal University of Technology, Akure, Nigeria
[2] National Information Technology Development Agency, Abuja, Nigeria
maxtunde@yahoo.com, akinwole2003@yahoo.co.uk, cangaye@hotmail.com

Abstract. This paper presents the formulation and implementation of a new fingerprint pattern-matching algorithm. The algorithm uses the spatial relation between the junction points within the 11 x 11 neighbourhood of the core points to generate the pattern matching scores. The junction points were the points of intersections of straight lines connecting the feature points within the neighbourhood. Experiments were conducted using FVC2002 fingerprint database comprising four datasets of images of different sources and qualities. Three statistics; namely False acceptance rate (FAR), False rejection rate (FRR) and the Average Matching Time (AMT) were generated for measuring the performance of the algorithm on the images. The results obtained demonstrated the effectiveness of the algorithm in distinguishing fingerprint images from different fingers. The results also revealed the failure rate of the algorithm when subjected to images with regions of significant degradations. Finally, findings from comparative analysis of the generated results with what obtained for some recently formulated fingerprint pattern matching algorithms revealed best performance for the proposed algorithm.

Keywords: Fingerprint matching, minutiae, spatial relation, FRR, FAR.

1 Introduction

Fingerprint is the results of minute ridges and valleys found on each of the fingers of every person. It is an impression of the friction ridges of all or any part of the finger. As shown in Figure 1, friction ridges are the raised and dark portions of the epidermis on the finger consisting of one or more connected ridge units of friction ridge skin. The valleys are the white and lowered regions.

Each print has an exclusive owner, and there has never been two individuals include ing identical twins recorded with the same print [1-2]. It has also been established that finger's ridges never change, from birth until death and no mater what happens, they will always reappear within a short period of time. The ridges of fingers form five major patterns of left loops, right loops, whorls, arch and tented arch as shown in Figure 2 [1-6]. In the loop pattern, the ridges enter from either side, re-curve and pass out or tend to pass out the same side they entered. In the left loop pattern, the

L. Chen, S. Kapoor, and R. Bhatia (eds.), *Intelligent Systems for Science and Information*, Studies in Computational Intelligence 542,
DOI: 10.1007/978-3-319-04702-7_5,

ridges enter from the right side while the ridges enter from the left side in the left loop. In a whorl pattern, the ridges are usually circular while in the arch pattern, the ridges enter from one side, make a rise in the center and exit generally on the opposite side.

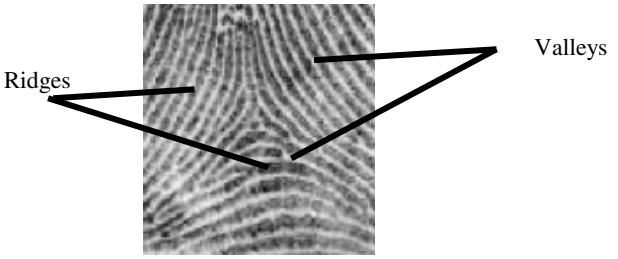


Fig. 1. Fingerprint Ridges and Valleys

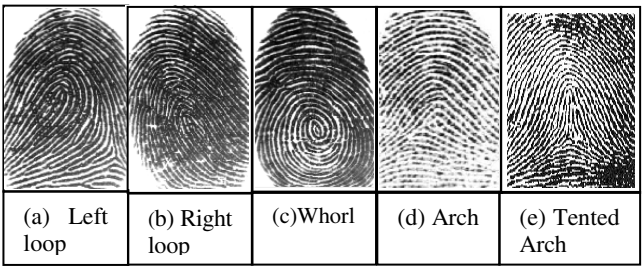


Fig. 2. Types of thumbprints

Fingerprints may be described as captured or latent print [7]. A captured print may be obtained for different purposes. It may be obtained when a person is arrested in connection with a crime. As part of the booking process, the police or other security agent rolled the arrestee's fingertip in ink and then impressed it on a card. The card is subsequently stored in libraries of such cards maintained by local, state or national government agencies. Captured prints may also be obtained by using modern day finger scan system [8-9]. Latent print in contrast, is typically produced at a crime scene and is usually not readily visible. It occurs when the natural secretions of the skin are deposited on a surface through fingertip contact at the crime scene. The best way to render latent fingerprints visible, so that they can be photographed, is complex and depends, for example, on the type of surface involved. A 'developer', usually a powder or chemical reagent, is often used to produce a high degree of visual contrast between the ridge patterns and the surface on which the fingerprint was left [7, 10].

Fingerprint has proved to be a very reliable means of identification. It has enjoyed superiority over all other biometrics including ear, nose, iris, voice, face, gait and signature [11]. The uniqueness of the pattern of friction ridges on a finger makes it immutable and therefore provides a strong mark of identity. More notably, even identical twins are differentiable using their fingerprints. Areas of the use of fingerprints

for identification include access to military installations, control room and high profile offices or stores. Fingerprints are also useful for bank transactions, voting, visa applications as well as, Subscribers Identification Module (SIM) card and examination registrations.

Fingerprint is also used for verification where an input fingerprint is compared with a previously enrolled fingerprint to know if the two fingerprints come from the same finger or not (1:1 match) [9]. The major reasons for the wide use of fingerprints for identification and verification include:

Availability for all individuals in respective of race, gender or age

Availability of easy, smooth operational and cheap fingerprint capturing devices

Unlike in other biometrics such as face, fingerprint does not change in pattern or form over time.

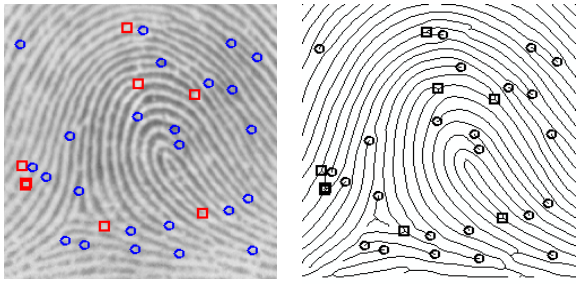
Fingerprint is distinct and highly unique from individual to individual

The important features of fingerprints that are largely responsible for its high performance in identification and verification systems are formed into three levels [9]. Features in level 1 are the macro details such as friction ridge flow, pattern type, and singular points. Level 1 features are used to categorize fingerprints into major pattern types. Level 2 features include minutiae such as ridge bifurcations and endings. The ridge bifurcations (marked by boxes) and endings (marked by circles) for a fingerprint and its thinned image is shown in Figure 3. Level 3 features are the dimensional attributes of the ridge such as ridge path deviation, width, shape, pores, edge contour and other details including incipient ridges, creases, and scars. Level 2 and level 3 features are used to establish a fingerprint's individuality or uniqueness. Pattern matching is mostly conducted when the need for ascertaining the exactness or variations among fingerprint images arises.

During fingerprints pattern matching, match scores are generated using the local and global features [12]. For fingerprints from the same finger, the scores should be high and low for those from different fingers. Challenges facing fingerprint matching include large intra-class variations (variations in fingerprint images of the same finger) and large interclass similarity (similarity between fingerprint images from different fingers). Intra-class variations are caused by finger pressure and placement—rotation, translation, and contact area—with respect to the sensor and condition of the finger such as skin dryness and cuts. Meanwhile, interclass similarity can be large because there are only three major types of major fingerprint patterns (arch, loop, and whorl) [9]. Series of fingerprints pattern matching algorithms had evolved with attendant strengths and weaknesses. In this research, the implementation of a new fingerprint pattern-matching algorithm that is based on the relative distances between the features and the core points is presented. Section 2 presents the review of the related works while Section 3 gives a brief description of minutiae based fingerprint pattern matching. The stages of the algorithm and its implementation are discussed in Sections 4 and 5 respectively. The conclusion drawn from the research is presented in Section 6.

2 Related Works

There are a number of techniques for the matching of fingerprints. One of them is the minutiae based technique that has witnessed a lot of attention from different groups. The technique is popular because minutiae in the fingerprint are widely believed to be the most unique, durable and reliable features. In addition, the template size of the biometric information base on minutiae is much smaller and the processing speed is higher than in other techniques such as graph-based fingerprint matching. These characteristics are very important for saving memory and energy on the embedded devices [13]. Minutiae matching algorithm is often designed to solve two problems; namely correspondence and similarity computation. Each minutia was assigned texture-based and minutiae-based descriptors for the correspondence problem in [14]. An alignment-based greedy matching algorithm was then used to establish the correspondences between minutiae. For the similarity computation, a 17-D feature vector was extracted from the matching result, and the feature vector is then converted into a matching score using support vector classifier. This method is comparative to the best algorithms even though its performances may change when some information such as ridges, orientation and frequency images are not used.



(a) Original image (b) Thinned image

Fig. 3. Fingerprint image and its ridge bifurcations

Latent fingerprint identification is of critical importance to law enforcement agencies in identifying suspects. They are inadvertent impressions left by fingers on surfaces of objects. While tremendous progress has been made in plain and rolled fingerprint matching, latent fingerprint matching continues to be a difficult problem. Poor quality of ridge impressions, small finger area, and large nonlinear distortion are the main difficulties in latent fingerprint matching compared to plain or rolled fingerprint matching. A system for matching latent fingerprints found at crime scenes to rolled fingerprints enrolled in law enforcement databases has been proposed in [15]. Extended features, including singularity, ridge quality map, ridge flow map, ridge wavelength map, and skeleton were used. The matching module consists of minutiae, orientation field and skeleton matching. The importance of various extended features was studied and the experimental results indicate that singularity, ridge quality map, and ridge flow map are the most effective features in improving the matching accuracy. However, the proposed latent matching algorithm is still inferior to the performance of experienced latent examiners, which may be caused by the methodologies

for matching ridge skeleton, minutiae and detailed ridge features. It may also be caused by difference in the approach to utilizing negative evidence.

The Euclidean space and ridge-based relative features among minutiae reinforce each other in the representation of a fingerprint. The authors in [16] introduced a novel algorithm based on global comprehensive similarity with three phases. Firstly, a minutia-simplex that contains a pair of minutiae as well as their associated textures was built to describe the Euclidean space–based relative features among minutiae. Its transformation-variant and invariant relative features were employed for the comprehensive similarity measurement and parameter estimation respectively. Secondly, the ridge-based nearest neighbourhood among minutiae was used to represent the ridge-based relative features among minutiae. With this approach, minutiae were grouped according to their affinity with a ridge. Finally, the relationship between transformation and the comprehensive similarity between two fingerprints was modeled in terms of histogram for initial parameter estimation. Experimental results show the effectiveness and suitability of the method for limited memory Automated Fingerprint Identification Systems (AFISs) owing to its very minimal template size.

With identity fraud in every society going on increasing trend and with rising emphasis on the emerging automatic personal identification applications, the need for biometric-based verification systems continued to increase. Fingerprint-based identification is therefore receiving a lot of attention. The traditional approaches to fingerprint representation suffers shortcomings including difficulty in the automatic detection and extraction of complete ridge structure as well as difficulty in quick matching of fingerprint images containing different number of unregistered minutiae points. The authors in [17] proposed a filter-based algorithm that uses a bank of Gabor filters to capture both local and global details in a fingerprint as a compact fixed length FingerCode. Fingerprint matching is based on the Euclidean distance between the two corresponding FingerCodes. The experimental results show that the algorithm is extremely fast with high verification accuracy, which is only marginally inferior to the best results of minutiae-based algorithms presented in [18]. The proposed system performs better than a state-of-the-art minutiae-based system when the performance requirement of the application system does not demand a very low false acceptance rate.

The basic idea in several minutiae based techniques is connecting the neighbouring minutiae with triangles using a Delaunay triangulation and analyzing the relative position and orientation of the grouped minutiae. Even if rotations, translations and non-linear deformations are present, the obtained triangle structure does not change significantly, except where the feature extraction algorithm fails. This technique provides a good processing time, describes the minutia relationship with consistency and works well with the nonlinear distortions. However, for genuine match, the overlapping area between the matching fingerprints should be large. The approach in [19-20] is similar to the approach taken in the current research. However, we propose new method for generating fingerprints matching scores using the spatial parameters existing for the minutiae points. The method proved suitable enough for handling matching problems due to image ridge orientation and size variations.

3 Minutiae Based Fingerprint Pattern Matching

During minutiae-based fingerprint pattern matching, a match score between two fingerprints is computed using the minutiae characteristics. Minutiae-based pattern matching is mostly used because forensic examiners have successfully relied on minutiae to match fingerprints for a long period. In addition, minutiae-based representation is storage efficient and expert testimony about suspect identity based on mated minutiae is admissible in courts of law [9]. The latest trend in minutiae matching is to use local minutiae structures to quickly find a permissible alignment between two fingerprints and then consolidate the local matching results at a global level. This kind of matching algorithm typically consists of the steps conceptualized in Figure 4.

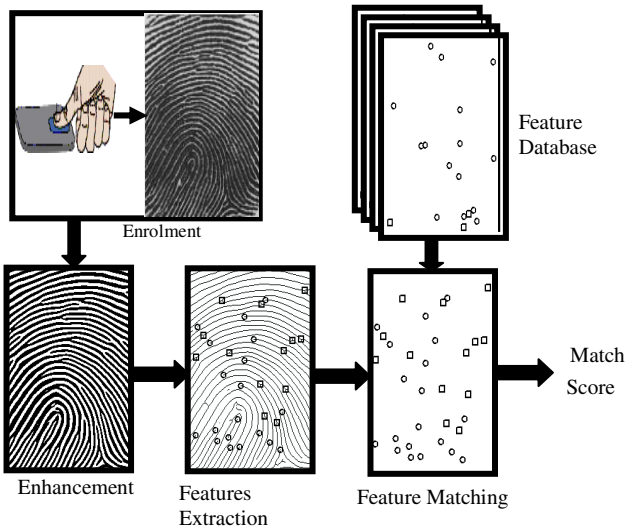


Fig. 4. A typical fingerprint pattern matching steps

The first step of the algorithm is the fingerprint enrolment. Depending on choice, a manual method of ink and paper or the electronic sensing method may be used. The enrolled fingerprint is consequently enhanced leading to smooth and speedy extraction of minutiae. The enhancement of fingerprint involves ridge segmentation, normalization, orientation estimation, frequency estimation, Gabor filtering, binarization and thinning [21-23]. The minutiae points are the points that uniquely describe any fingerprint image. A minutia point is described by type, location and orientation. Algorithms for the extraction of minutiae points from thinned fingerprint images have been proposed or implemented in [12, 21-22, 24]. A number of these algorithms use the 8-nearest neighbours approach to extract ridge points that are bifurcations, ridge endings, isolated, continuing or crossing points. During feature matching, a pair-wise

similarity between minutiae of two fingerprints is computed. This is done through comparison of minutiae descriptors that are invariant to rotation, size and translation.

Any two fingerprints are aligned according to the most similar minutiae pair and the algorithm then establishes minutiae that are close enough in location and direction. A match score is finally computed to show or present the degree of match between two fingerprints. The computation is based on factors such as the number of matching minutiae, the percentage of matching minutiae in the overlapping area of two fingerprints, and the consistency of ridge count between matching minutiae [9].

4 The Proposed Fingerprint Pattern Matching Algorithm

The relative distance algorithm for computing pattern matching scores of fingerprint images uses the distances between the image reference and the feature points. Reference point may be in form of a core or delta point. Some fingerprints (like the one shown in Figures 2(a) and 2(e)) possess the two types of reference points while others (shown in Figure 2(b), 2(c) and 2(d)) contain only one. The core points OA, OC and OD respectively shown in Figure 5(a), 5(b) and 5(c) are the point of maximum orientations of the ridge structures. They are also the point where the directional field becomes discontinuous [25]. The delta point is the point where the ridge points in three directions. The in-between angle of the three directional ridges is approximately 1200 as shown on points OB and OE of Figure 5(a) and 5(d) respectively.

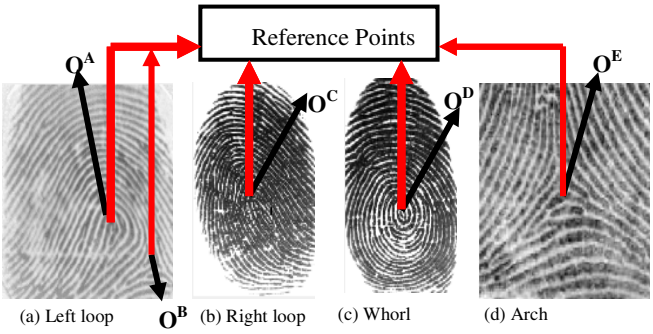
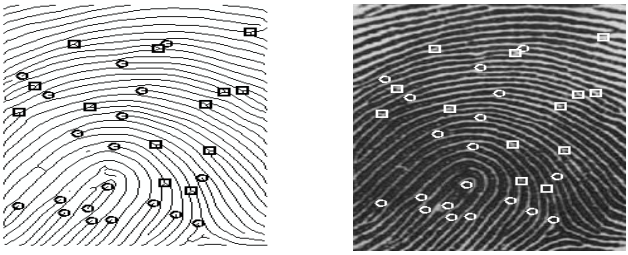


Fig. 5. Fingerprint images and their reference points

The algorithm hinges on the reference point because its position and distance to every feature point do not change irrespective of direction for a specific image size. A feature point is uniquely described by type, location and orientation. The commonest of the feature points are bifurcations and ridge endings [12, 24]. In Figure 6, the bifurcations of the skeleton and real images are denoted by circles while ridge endings are denoted by squares.



(a) Skeleton image (b) Original image

Fig. 6. Feature points for skeleton and original images

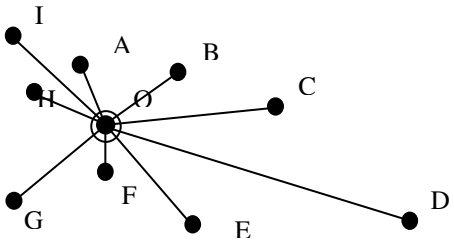


Fig. 7. Interconnecting lines between feature points

The proposed algorithm is formulated based on the assumption that straight line connects every pair of feature points in the image. Figure 7 illustrates typical interconnecting lines between feature point O and nine (9) other feature points in an image. The connecting lines are in various directions with lengths proportionate to the distances between point O and the connecting minutiae points.

The proposed algorithm is in the following phases:

a. Obtain the core point using the steps illustrated in Figure 8. Fingerprint image segmentation is used to separate the foreground region from the background region. The foreground regions contain the ridges and valleys while the background regions are mostly the outside regions which contain the noises introduced into the image during enrolment. The essence of segmentation is to reduce the burden associated with image enhancement by ensuring that focus3 is only on the foreground regions. Normalization on its own is performed on the segmented fingerprint ridge structure for the standardization of the level of variations in the image grey-level values. By normalization, the grey-level values are constrained to a range that is good enough for improved image contrast and brightness.

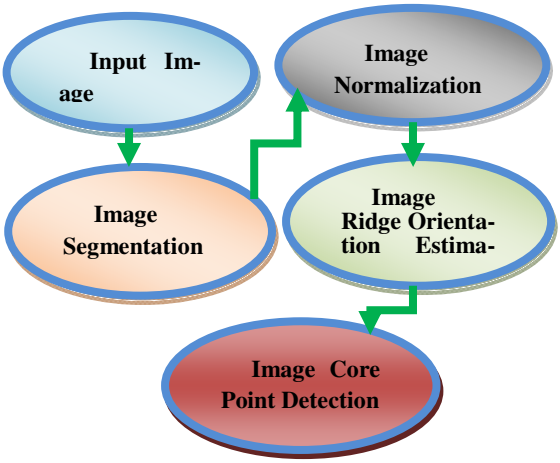


Fig. 8. Steps for fingerprint core point detection

The segmentation and normalization algorithms proposed in [26] were implemented in this research. The ridge orientation estimation algorithm implemented is in the following phases [27, 28]:

Firstly, blocks of size S x S were formed on the normalized fingerprint image.

For each pixel, (p,q) in each block, the gradients ∂_x(p,q) and ∂_y(p,q) were computed as the gradient magnitudes in the x and y directions, respectively. ∂_x(p,q) was computed using the horizontal Sobel operator while ∂_y(p,q) was computed using the vertical Sobel operator [23].

The local orientation for each block centered at pixel I(i,j) is then computed from:

$$V_x(i,j) = \sum_{p=a}^{b}\sum_{q=c}^{d} 2\partial_x(p,q)\partial_y(p,q) \qquad \text{(Error! Bookmark not defined.)}$$

$$V_y(i,j) = \sum_{p=a}^{b}\sum_{q=c}^{d} \partial_x^2(p,) - \partial_y^2(p,q) \qquad \text{(Error! Bookmark not defined.)}$$

$$\theta(i,j) = \frac{1}{2}\tan^{-1}\frac{V_y(i,j)}{V_x(i,j)} \qquad \text{(Error! Bookmark not defined.)}$$

$a = i - \frac{S}{2}$, $b = i + \frac{S}{2}$, $c = j - \frac{S}{2}$, $d = j + \frac{S}{2}$ and $\Theta(i, j)$ is the least square estimate of the local orientation of the block centered at pixel (i, j).

The orientation image is then converted into a continuous vector field defined by:

$$\varphi_x(i,j) = 2\cos^2\big(\theta(i,j)\big), \qquad (4)$$

$$\varphi_y(i,j) = 2\sin\big(\theta(i,j)\big)\cos\big(\theta(i,j)\big), \qquad (5)$$

φ_x and φ_y are the *x* and *y* components of the vector field, respectively. Gaussian smoothing is then performed on the vector field as follows:

$$\varphi'_X(i,j) = \sum_{p=-\vartheta}^{\vartheta} \sum_{q=-\vartheta}^{\vartheta} G(p,q)\varphi_x\,(i-ps, j-qs), \tag{6}$$

$$\varphi'_X \quad \varphi'_y(i,j) = \sum_{p=-\vartheta}^{\vartheta} \sum_{q=-\vartheta}^{\vartheta} G(p,q)\varphi_y\,(i-ps, j-qs), \tag{7}$$

$$\vartheta = \frac{S_\varphi}{2} \tag{8}$$

G is a Gaussian low-pass filter of size S_φ *X* S_φ.

The orientation field O of the block centered at pixel (i,j) is finally smoothed using the equation:

$$O(i,j) = 0.5\cos\left(\left(\left(\varphi'_y(i,j)\right) * (\varphi'_x(i,j))^{-1}\right) * \right.$$

$$\left.\sin\left(\left(\varphi'_y(i,j)\right) * (\varphi'_x(i,j))^{-1}\right)\right) \tag{9}$$

The direction of gravity of progressive blocks (non-overlapping sub block) for each S x S block is determined by using the equations [29]:

$$P = \sum_{l=0}^{2} \sum_{m=0}^{2} \varphi'_X \tag{10}$$

$$Q = \sum_{l=0}^{2} \sum_{m=0}^{2} \varphi'_y \tag{11}$$

Fine tune the orientation field as follows:

If $Q(i,j) \neq 0$ then

$\beta = 0.5 tan^{-1}(Q/P)$

else

$\beta = \pi/2$

if $\beta < 0$ then

if $P < 0$ then

$\beta = \beta + \pi/2$

else: $\beta = \beta + \pi$

else if $P < 0$ then:

$\beta = \pi/2$

The value β is calculated as the orientation value of the image.

The blocks with slope values ranging from 0 to pi/2 are located. Then a path is traced down until a slope that deviates from this range is encountered and that block is marked.

The block that has the highest number of marks will compute the slope in negative y direction and its x and y position will be the core-point.

The equations of the straight lines connecting all the feature points in the 11 x 11 neighbourhood of the core point of the image are calculated. Given that points $P_1(\rho_1, \tau_1)$ and $P_2(\rho_2, \tau_2)$ shown in Figure 9 are two feature points located in this neighbourhood for an image, the equation of the straight line P1P2 is given by:

$$y = \varphi x + \partial \tag{12}$$

φ is the gradient of line P_1P_2 . ∂ is defined by:

$$\partial = 0.5((\tau_1 + \tau_2)(\rho_1 - \rho_2) - (\tau_1 - \tau_2)(\rho_1 + \rho_{2\delta})) * (\rho_1 - \rho_2)^{-1} \tag{13}$$

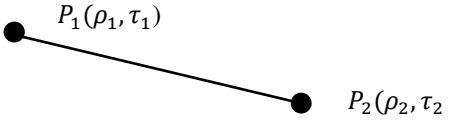


Fig. 9. Presumed feature points

The point of interception of any two straight lines forms a junction point as shown in Figure 10.

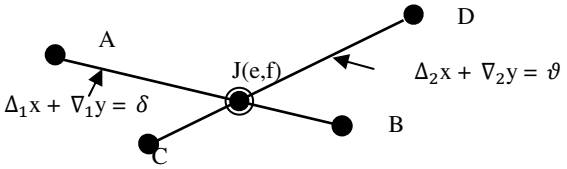


Fig. 10. Junction point of straight line formed by feature points

The locations of all the junction points in the 11 x 11 neighbourhood of the core point are obtained by solving all possible pairing of equations of the straight lines using simultaneous or substitution by elimination approach. Given that straight lines AB shown in Figure 10 is defined by the equation:

$$\Delta_1 x + \nabla_1 y = \delta \tag{14}$$

and the straight line CD is defined by the equation:

$$\Delta_2 x + \nabla_2 y = \vartheta \tag{15}$$

The junction point J(e,f) is obtained from:

$$e = \left(\delta(\Delta_2\nabla_1 - \Delta_1\nabla_2) - \nabla_1(\Delta_2\delta - \Delta_2\vartheta)\right) * (\Delta_1(\Delta_2\nabla_1 - \Delta_1\nabla_2))^{-1} \tag{16}$$

$$and \quad f = (\Delta_2\delta - \Delta_1\vartheta)(\Delta_2\nabla_1 - \Delta_1\nabla_2)^{-1} \tag{17}$$

d. The distance, ω_i between the i^{th} junction point $J_i(e_i,f_i)$ and the image core point $M(\alpha,\beta)$ is obtained from:

$$\omega i = ((e_i - \alpha)^2 + (f_i - \beta)^2)^{0.5} \tag{18}$$

e. For any two images comprising the query and reference images, the degree of closeness, $\gamma_c$is obtained from:

$$\gamma_c = \sum_{i=1}^{n} \frac{|P(i) - I(i)|}{P(i)} \tag{19}$$

n is the smaller of the respective number of junction points in the query and reference image, P(i) and I(i) represent the distance between the i^{th} junction point and the core point for the query and reference image respectively.

The cross-correlation coefficient value, C is then computed as the pattern matching score for the two images by using the formula:

$$C = 1 - \frac{\gamma_c}{100} \tag{20}$$

From this formula, the dissimilar value will be γ_c = 0 for exact or same images and, consequently, the cross-correlation will be C = 1.

5 Experimental Results

The implementation of the proposed fingerprint matching algorithm was carried out using Matlab version 7.6 on window Vista Home Basic Operating System. The experiments were performed on a Pentium 4 – 2.10 GHz processor with 1.00GB of RAM. The purpose of the experiments is to analyze the performance of the algorithm under different conditions of images as well as generate the metrics that could serve the basis for the comparison of the results from the research with results from related works. The experiments are based on the tests from Fingerprint Verification Competition (FVC). The fingerprints were taken from FVC2002 datasets DB1, DB2, DB3 and DB4 [30-31]. The detail of the datasets is presented in Table 1.

Table 1. Details of FVC2002 Fingerprint Database [31]

Data-base	Sensor Type	Image size	Number	Resolution
DB1	Optical Sensor	388 × 374 (142 Kpixels)	100 × 8	500 dpi
DB2	Optical Sensor	296 × 560 (162 Kpixels)	100 × 8	569 dpi
DB3	Capacitive Sensor	300 × 300 (88 Kpixels)	100 × 8	500 dpi
DB4	SFinGe v2.51	288 × 384 (108 Kpixels)	100 × 8	About 500 dpi

The four datasets were of different qualities with each containing 80 fingerprints. The 80 fingerprints are made up of 5 fingerprints from 16 different persons. Dataset DB1 and DB2 were acquired using an optical fingerprint reader, dataset DB3 was acquired using a capacitive fingerprint reader, and dataset DB4 was obtained with computer assistance, using the software SFinGE.

Three indicators; namely False Rejection Rate (FRR), False Acceptance Rate (FAR) and Average Matching Time (AMT) were measured. These indicators were chosen because they are among the commonest indicators used for measuring the performance of any fingerprint pattern matching systems [9]. FRR is the rate of occurrence of a scenario of two fingerprints from same finger failing to match (the matching score falling below the threshold). On the other hand, FAR is the rate of occurrence of a scenario of two fingerprints from different fingers found to match (matching score exceeding the threshold). Matching all the fingerprints from the same finger was used to measure the FRR while measuring FAR was done through matching each fingerprint image of each finger with all fingerprints from the other fingers.

The obtained results show that FRR and FAR are greatly influenced by the nature and quality of the images. The FAR and FRR results obtained for the chosen threshold for datasets DB1and DB2 are presented in Table 2 and Table 3 respectively.

Table 2. FAR and FRR Values for Dataset DB1

Statistics	Value (%)
FAR	0
FRR	15.50

Table 3. FAR and FRR Values for Dataset DB2

Statistics	Value (%)
FAR	0
FRR	12.50

These results reveal that for images obtained using optical fingerprint reader, the proposed algorithm produced an FAR of 0%. Meaning that the algorithm did well in identifying fingerprint images obtained from different fingers under equal conditions in the two datasets. However, the obtained FRR values of 15.5% and 12.5% revealed the inability of the algorithm to match substantial number of fingerprint images in each of the two datasets even though they were enrolled from the same finger. A number of factors including enrolment pressure, rotation, translation and contact area [9] are responsible for this. These factors constrained images from the same finger to show variations in quality, contrast and noise levels. Consequently, there is different level of enhancement and feature extractions in the images leading to matching score that falls below the threshold. The FRR value of 15.5% and 12.5% obtained for dataset DB1 and DB2 respectively indicate the extent to which these factors affect the images in the two datasets.

The FAR and FRR results obtained for the dataset DB3 are presented in Table 4.

Table 4. FAR and FRR Values for Dataset DB3

Statistics	Value (%)
FAR	0
FRR	20.70

The results show that for its images, the proposed algorithm produced an FAR of 0%. This result also shows the algorithm did well in the identification of fingerprint images captured from different fingers using capacitive fingerprint reader under same conditions. However, the obtained FRR value of 20.7% revealed the extent to which the algorithm failed in its efforts to match fingerprint images from same finger. This failure rate is attributed to differences in the qualities of images enrolled from the same finger. Visual inspection of fingerprints images in dataset DB3 reveals significant breaks and smudged effects on the ridge structures.

The enhancement process adversely suffered from these effects as different levels or degrees of artifacts inform of cross over hole or spike structure [12] were introduced into the images. The presence of artifacts led to the extraction of false minutiae (ridge ending and bifurcation) points of various numbers across images. These false minutiae contribute significantly to the high FRR rate of 20.7%. The highest FRR value compared to what obtained for datasets DB1 and DB2 shows the superiority of the two datasets over DB3 in term of quality. Dataset DB4's FAR and FRR values are presented in Table 5. The values indicate that the proposed algorithm produced an FAR of 0% for the dataset. This result also demonstrates the ability of the propose algorithm in distinguishing fingerprint images obtained from different fingers using computer aids. However, the obtained FRR value of 14.35% revealed the failure rate of the algorithm when matching images from the same finger. Visual inspection of fingerprints images in dataset DB4 reveals a high number of gaps across the ridges. Though the enhancement algorithm succeeded in bridging a good number of them, some still resulted in false minutiae points. The recorded FRR value of 14.35% therefore indicates the level of the negative impact of these false minutiae points on the images.

Table 5. FAR and FRR Values for Dataset DB4

Statistics	Value (%)
FAR	0
FRR	14.35

The column chart of the FRR values for the four datasets is presented in Figure 11. It is revealed that dataset DB2 has the lowest FRR value of 12.5% followed by DB4, DB1 and DB3 with FRR values of 14.58%, 15.5% and 20.7% respectively. Overall, the proposed pattern-matching algorithm successfully identified fingerprints from different source (finger) by returning an average FAR of 0% for the fingerprints images obtained from different sources and methodologies in the four datasets. However, an average FRR value of 15.82% shows the failure rate of the proposed algorithm over the four datasets.

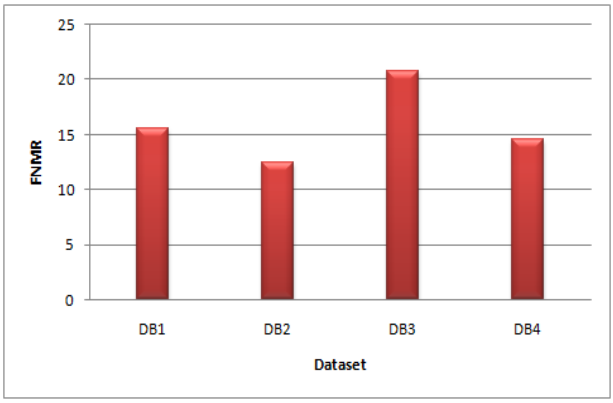


Fig. 11. Column chart of the FRR values for the four datasets

The Receiver Operating Characteristic (ROC) Curve is also generated for experiment on each dataset. An ROC curve depicts the plot of genuine acceptance rate (1-FRR) against false acceptance rate for all possible matching thresholds and measures the overall performance of the system. Each point on the curve is a particular decision threshold. In the ideal case, both the error rates, that is, FAR and FRR should be zero and the genuine distribution and imposter distribution should be disjoint. In such a case, the "ideal" ROC curve is a step function at the zero FAR. On the other extreme, if the genuine and imposter distributions are the same, then the ROC is a line segment with a slope of 45o with an end-point at zero FAR. In practice, the ROC curve behaves in between these two extremes [32]. Figures 12 - 15 show the ROC curves for experiments on datasets DB1, DB2, DB3 and DB4 respectively.

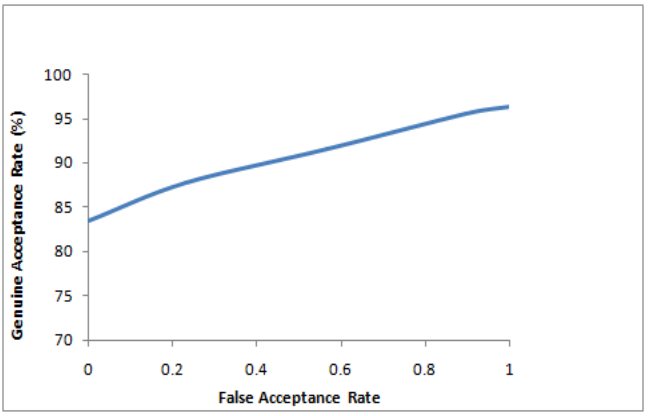


Fig. 12. ROC Curve for experiment on Dataset DB1

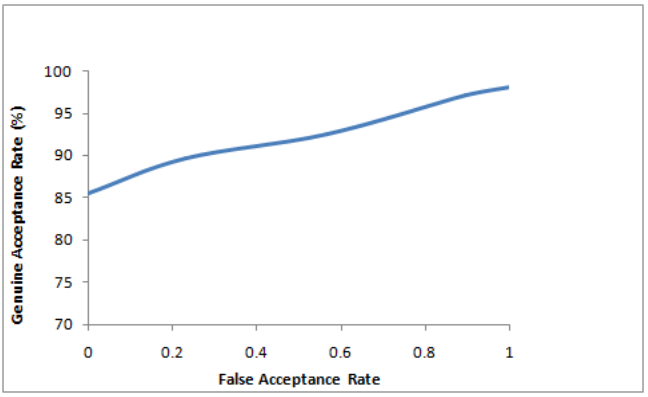


Fig. 13. ROC Curve for experiment on Dataset DB2

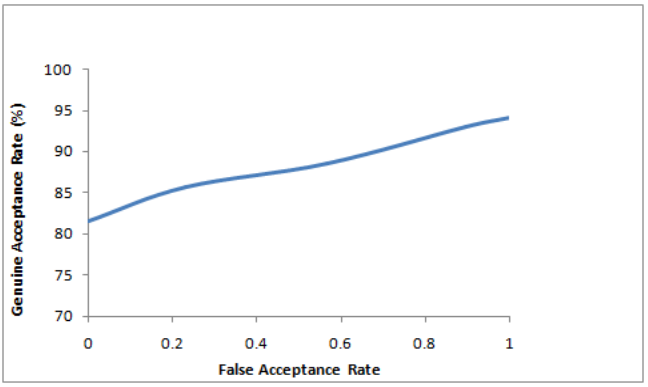


Fig. 14. ROC Curve for experiment on Dataset DB3

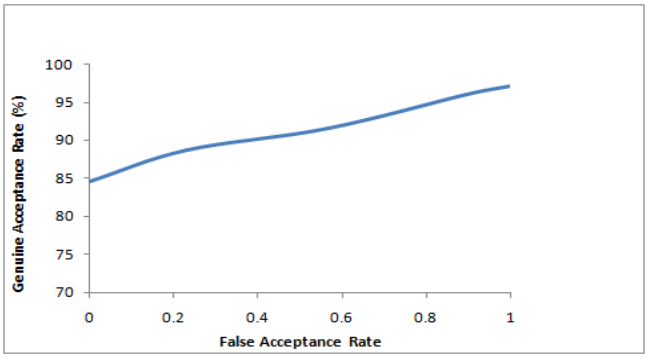


Fig. 15. ROC Curve for experiment on Dataset DB4

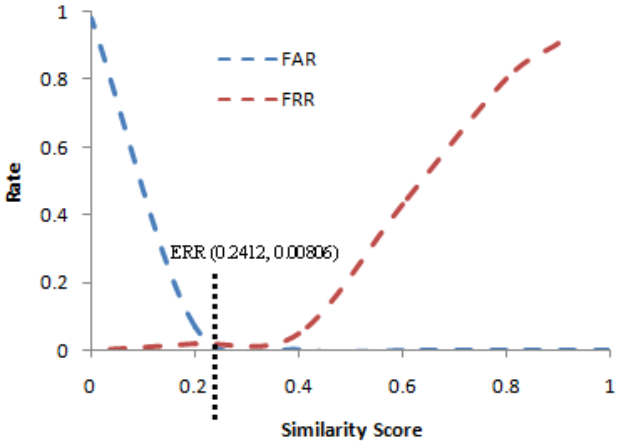


Fig. 16. ROC Curve for Dataset DB1

The Equal Error Rates (EER) were also generated for the experiments. EER is the best single description of the Error Rate of an algorithm and the lower its value, the lower the error rate and adequacy of the algorithm. For each matching threshold, i EER is the value at which FAR (i) and FRR (i) are equal. Figures 16-19 show the obtained EER points for the FAR/FRR function for experiments on Datasets DB1, DB2, DB3 and DB4 respectively.

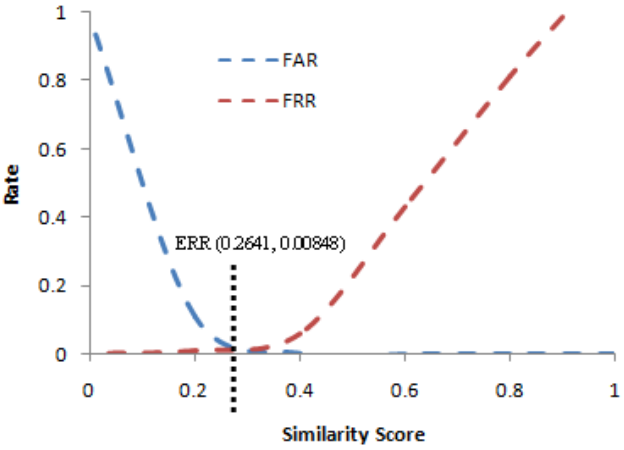


Fig. 17. ROC Curve for Dataset DB2

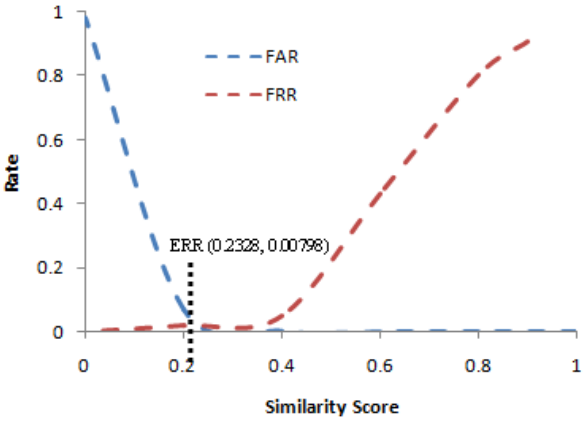


Fig. 18. ROC Curve for Dataset DB3

It is revealed from the Figures that ERR points (0.2412, 0.00806), (0.2641, 0.00848), (0.2328, 0.00798) and (0.2551, 0.00819) were recorded for datasets DB1, DB2, DB3 and DB4 respectively. The implication of these results is that for the matching threshold 0.2412, there is a guarantee of the same FAR and FRR error rates of 0.00806 for the algorithm on Dataset DB1. That is 8.06 out of every 1000 impostors or genuine attempts will succeed or fail. Similarly, 8.48, 7.98 and 8.19 out of every 1000 impostors or genuine attempts will succeed or fail for the images in Dataset DB2, DB3 and DB4 respectively.

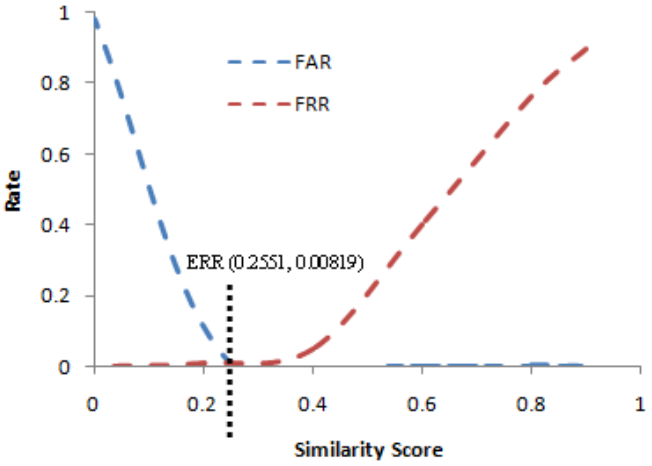


Fig. 19. ROC Curve for Dataset DB4

The average FRR and FAR computation times for the four datasets are presented in Table 6. Dataset DB2 has the lowest FRR average computation time of 0.49 second. It also recorded the lowest average computation time of 0.59 second for FAR. This is followed by DB1, DB4 and DB3 with average FRR: FAR computation time of 0.61:0.69, 0.69:0.79 and 0.81:1.07 second respectively. The lowest average computation time for dataset DB2 is attributed to fewest numbers of minutiae points in the 11 x 11 neighbourhood of the core points and consequently, smallest number of junction points leading to most reduced computations. Similarly, the highest average computation time recorded for dataset DB3 indicates the availability of highest number of both true and false minutiae points in the 11 x 11 neighbourhood of the core points thereby raising the number of computations.

Table 6. Average Computation Time for the Four Datasets

Dataset	Average Computation time (sec)	
	FRR	FAR
DB1	0.61	0.69
DB2	0.49	0.59
DB3	0.81	1.07
DB4	0.69	0.79

Table 7 presents the FRR and FAR values for four different algorithms using the same dataset (FVC2002 fingerprint database). The superior performance of the proposed algorithm over the other algorithms is clearly exhibited with its lowest FRR values recorded for all the datasets. In addition, it is the only algorithm with an FAR value of zero for all the datasets. No reason was given for the non-availability of FAR

values for Datasets DB3 and DB4 in Ref. [33]. The column charts of Figures 20 and 21 depict least figures for the current study thereby buttressing superior performances of our algorithms over those proposed in [33-35].

Table 7. FAR and FRR for Different Algorithms

	Ref. [33]		Ref. [34]		Ref. [35]		Current Study	
Set	FRR	FAR	FRR	FAR	FRR	FAR	FRR	FAR
DB1	16.2	0	52.58	0	89.3	1.7	15.50	0
DB2	12.6	0	50.03	0	88.6	3.7	12.50	0
DB3	NA	NA	73.75	0	91.2	2.4	20.70	0
DB4	NA	NA	65.24	.015	81.3	0.9	14.58	0

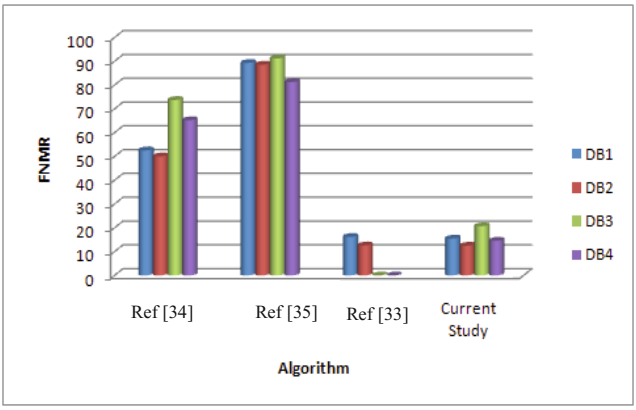


Fig. 20. Colum Chart of FRR values for different fingerprint matching algorithms

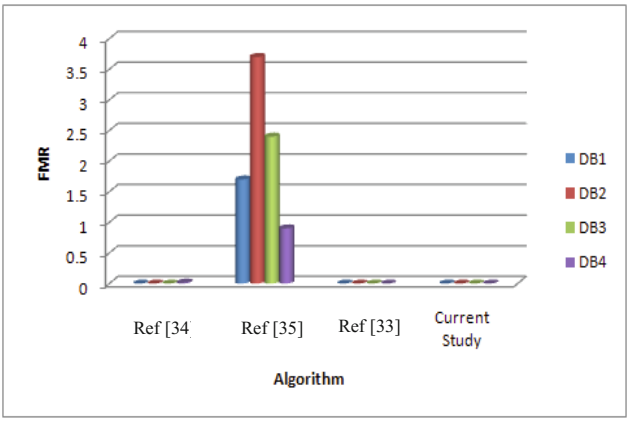


Fig. 21. Colum Chart of FAR values for different fingerprint matching algorithms

Table 8. Average Computation Time in Second for Different Algorithms

	Ref. [34]		Current Study	
Dataset	FRR	FAR	FRR	FAR
DB1	2	1.7	0.61	0.69
DB2	4	3.7	0.49	0.59
DB3	2	2.4	0.81	1.07
DB4	3	0.9	0.69	0.79

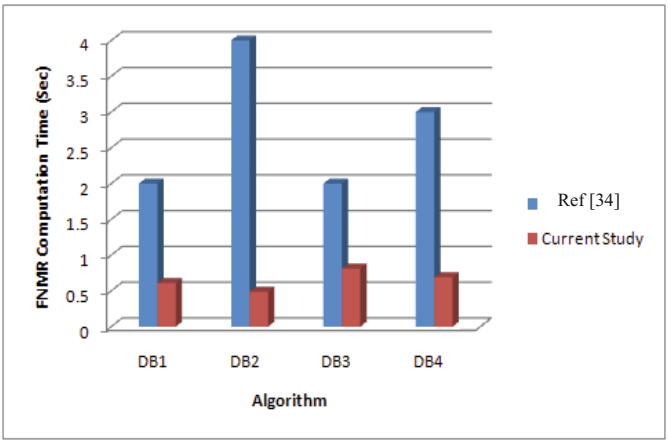


Fig. 22. Colum Chart of FRR matching time for different fingerprint matching algorithms

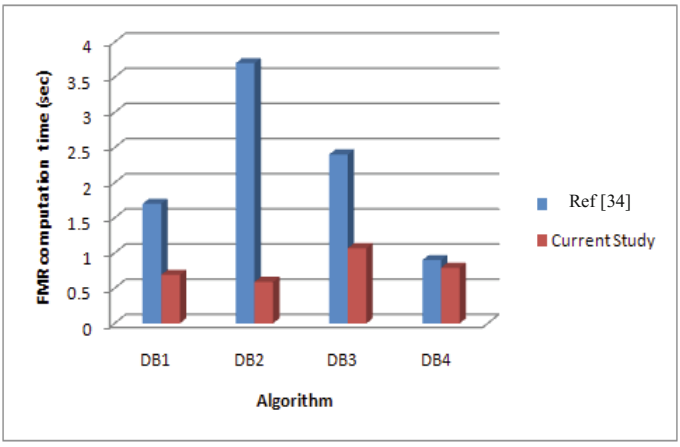


Fig. 23. Colum Chart of FAR matching time for different fingerprint matching algorithms

Table 8 presents the average computation time for FRR and FAR in [34] and the current study. For all the datasets, the proposed algorithm exhibited lower computation time, which confirms its superiority. A further proof of this is presented on the column charts of the values of Table 8 presented in Figures 22 and 23.

6 Conclusion and Future Work

The implementation of a proposed fingerprint pattern matching algorithm that is suitable for building an Automated Fingerprint Identification System (AFIS) has been presented. The algorithm used the relative distances between the minutiae points in the 11 x 11 neighbpourhood of the core points. The algorithm hinged on the premise that for an image size, the minutiae-core point distances within this region do not change irrespective of orientation. The essence of confining matching to this neighbourhood is to reduce the number of computations involved in the determination of the equations of the interconnecting lines and junction points.

The results obtained showed the effectiveness of the algorithm in distinguishing fingerprints from different sources with average FAR of 0%. However, the ability to match images from same source depends on the qualities of such images. Since a number of images in the used datasets are significantly corrupt due to various effects, the algorithm yielded an average FRR values of 15.82% and average ERR of 0.00817 with the third dataset mostly affected. The same order of performance was recorded for the FRR and the average matching time over the datasets. A comparative review of the obtained FRR, FAR and the computation time values with what obtained for some recently formulated algorithms over the same datasets revealed best performance for the proposed algorithm. However, the average matching times are still high when compared to results for other algorithms such as the one proposed in [34] which recorded significantly low average matching results. Emphasis will therefore be directed towards the optimization of the proposed algorithm so that the average matching time is considerably reduced. Similarly, efforts will be directed towards ensuring that the false minutiae play very minimal or zero role on the matching results.

References

1. Eckert, W.G.: Introduction to Forensic Science. Elsevier, New York (1996)
2. FIDIS. Future of Identity in the Information Society. Elsvier Inc. (2006)
3. Salter, D.: Thumbprint – An Emerging Technology, Engineering Technology, New Mexico State University (2006)
4. Wayman, J., Maltoni, D., Jain, A., Maio, D.: Biometric Systems. Springer-Verlag London Limited (2005)
5. Akinyokun, O.C., Adegbeyeni, E.O.: Scientific Evaluation of the Process of Scanning and Forensic Analysis of Thumbprints on Ballot Papers. In: Proceedings of Academy of Legal, Ethical and Regulatory Issues, New Orleans, vol. 13(1) (2009)
6. Yount, L.: Forensic Science: From Fibres to Thumbprints. Chelsea House Publisher (2007)
7. Michael, C., Imwinkelried, E.: A Cautionary Note about Fingerprint Analysis and Reliance on Digital Technology. Public Defence Backup Center REPOR, vol. XXI(3T), pp. 7–9 (2006)
8. Nanavati, S., Thieme, M., Nanavati, R.: Biometrics, Identifying Verification in a Networked World, pp. 15–40. John Wiley & Sons, Inc. (2002)
9. Anil, K.J., Jianjiang, F., Karthik, N.: Fingerprint Matching, pp. 36–44. IEEE Computer Society (2010)

10. McMurray, H.N., Williams, G.: Latent Thumb Mark Visualization Using a Scanning Kelvin Probe. Forensic Science International (2007)
11. Roberts, C.: Biometrics (2005), http://www.ccip.govt.nz/newsroom/informoation-notes/2005/biometrics.pdf
12. Iwasokun, G.B., Akinyokun, O.C., Alese, B.K., Olabode, O.: A Modified Approach to Crossing Number and Post-Processing Algorithms for Fingerprint Minutiae Extraction and Validation. IMS Manthan International Journal of Computer Science and Technology 6(1), 1–9 (2011)
13. Shenglin, Y., Ingrid, M.V.: A Secure Fingerprint Matching Technique (2003), http://www.cosic.esat.kuleuban.be/publications/article-723.pdf (accessed January 23, 2012)
14. Jianjiang, F.: Combining minutiae descriptors for fingerprint matching. Elsevier Pattern Recognition 41, 342–352 (2008)
15. Anil, K.J., Jianjiang, F.: Latent Fingerprint Matching. IEEE Transactions on Pattern Analysis and Machine Intelligence 33(1), 88–100 (2011)
16. He, Y., Tian, J., Member, S., Li, L., Chen, H., Yang, X.: Fingerprint Matching Based on Global Comprehensive Similarity (2005), http://www.fingerpass.net/downloads/papers/Fingerprint%20Matching%20Based%20on%20Global%20Comprehensive%20Similarity.pdf (accessed August 21, 2012)
17. Jain, A.K., Prabhakar, S., Hong, L., Pankanti, S.: Filterbank-Based Fingerprint Matching. IEEE Transactions on Image Processing 9(5) (2000)
18. Jain, A.K., Hong, L., Pankanti, S., Bolle, R.: An identity authentication system using fingerprints. Proc. IEEE 85(9), 1365–1388 (1997)
19. Giuseppe, P.E., Albert, N.: Fingerprint Matching Using Minutiae Triangulation (2003), http://idisk.mac.com/geppy.parziale/Public/Papers/delaunay.pdf (accessed January 23, 2012)
20. Xinjian, C., Jie, T., Xin, Y., Yangyang, Z.: An Algorithm for Distorted Fingerprint Matching Based on Local Triangle Feature Set. IEEE Transactions on Information Forensics and Security 1(2), 169–177 (2006)
21. Raymond, T.: Fingerprint Image Enhancement and Minutiae Extraction, PhD Thesis Submitted to School of Computer Science and Software Engineering, University of Western Australia, pp. 21–56 (2003)
22. Hong, L., Wau, Y., Anil, J.: Fingerprint image enhancement: Algorithm and performance evaluation. In: Pattern Recognition and Image Processing Laboratory, Department of Computer Science, Michigan State University, pp. 1–30 (2006)
23. Iwasokun, G.B., Akinyokun, O.C., Alese, B.K., Olabode, O.: Fingerprint Image Enhancement: Segmentation to Thinning. International Journal of Advanced Computer Science and Applications (IJACSA) 3(1) (2012)
24. Iwasokun, G.B., Akinyokun, O.C., Alese, B.K., Olabode, O.: Adaptive and Faster Approach to Fingerprint Minutiae Extraction and Validation. International Journal of Computer Science and Security 5(4), 414–424 (2011)
25. López, A.C., Ricardo, R.L., Queeman, R.C.: Fingerprint Pattern Recognition, PhD Thesis, Electrical Engineering Department, Polytechnic University (2002)
26. Iwasokun, G.B., Akinyokun, O.C., Olabode, O.: A Mathematical Modeling Approach to Fingerprint Ridge Segmentation and Normalization. International Journal of Computer Science and Information Technology & Security 2(2), 263–267 (2012)

27. Iwasokun, G.B., Akinyokun, O.C., Olabode, O.: A Block Processing Approach to Fingerprint Ridge Orientation Estimation. Journal of Computer Technology and Application 3, 401–407 (2012)
28. Hong, L., Wau, Y., Anil, J.: Fingerprint image enhancement: Algorithm and performance evaluation. In: Pattern Recognition and Image Processing Laboratory, Department of Computer Science, Michigan State University, pp. 1–30 (2006)
29. Navrit, K.J., Amit, K.: A Novel Method for Fingerprint Core Point Detection. International Journal of Scientific & Engineering Research 2(4), 1–6 (2011)
30. Maio, D., Maltoni, D., Cappelli, R., Wayman, J.L., Jain, A.K.: FVC2002: Second Fingerprint Verification Competition. In: 16th International Conference on Pattern Recognition 2002, pp. 811–814 (2002)
31. Li, T., Liang, C., Sei-ichiro, K.: Fingerprint Matching Using Dual Hilbert Scans. In: SITIS, pp. 553–559 (2009)
32. Jain, A.K., Prabhakar, S., Chen, S.: Combining multiple matchers for a high security Fingerprint verification system. Pattern Recognition Letters 20, 1371–1379 (1999)
33. Nandakumar, K.: A Fingerprint Cryptosystem Based on Minutiae Phase Spectrum. In: WIFS 2010, USA (2010)
34. Perez-Diaz, A.J., Arronte-Lopez, I.C.: Fingerprint Matching and Non-Matching Analysis for Different Tolerance Rotation Degrees in Commercial Matching Algorithms. Journal of Applied Research and Technology 8(2), 186–199 (2010)
35. Peer, P.: Fingerprint-Based Verification System A Research Prototype. In: IWSSIP 2010 - 17th International Conference on Systems, Signals and Image Processing, pp. 150–153 (2010)

Different Artificial Bee Colony Algorithms and Relevant Case Studies

Amr Rekaby

Egyptian Research and Scientific Innovation Lab (ERSIL)
Cairo, Egypt
rekaby0@hotmail.com

Abstract. Solving optimization problems can be achieved by many optimization algorithms. Swarm algorithms are part of these optimizations algorithms which based on community-based thinking. Bio-inspired algorithms are these algorithms that are artificially inspired from natural biological systems. Artificial Bee colony algorithm is a modern swarm intelligence algorithm inspired by real bees foraging behavior, and real bees' community communication techniques. This chapter discusses Artificial bee colony algorithm (ABC) and other algorithms that are driven from it such as "Adaptive Artificial Bee Colony" (AABC), "Fast mutation artificial bee colony" (FMABC), and "Integrated algorithm based on ABC and PSO" (IABAP). Comparisons between these algorithms and previous experiments results are mentioned.

The chapter presents some case studies of ABC like traveling salesman problem, job scheduling problems, and software testing. The study discusses the conceptual modeling of ABC in these case studies.

Keywords: Artificial bee colony (ABC), adaptive artificial bee colony (AABC), fast mutation artificial bee colony (FMABC), ABC case studies, swarm intelligence, evolutionary algorithms, swarm intelligence.

1 Introduction

Swarm intelligence is a decentralized, self-organized intelligence that bases on a collecting data from community members towards finding a better solution through their collaborations. Swarm intelligence could be a natural or artificial intelligence. Many real animals' swarms have this perception like ants, bees, group of fishes, and others. No member of this swarm has its own qualified sense. Although, together they can get the decision easily by their collaborations.

Artificial swarm intelligence simulates the same concept of the real ones. In artificial swarm intelligence, no element of the population is capable alone of doing the work, so they all integrate their findings to work more efficiently in the goal achieving.

Optimization problems have many visible solutions in the search space, finding the best solution in the search space is a very time consuming activity (may take

L. Chen, S. Kapoor, and R. Bhatia (eds.), *Intelligent Systems for Science and Information*, Studies in Computational Intelligence 542,
DOI: 10.1007/978-3-319-04702-7_6,

decades), so it is an extremely interesting area to use the swarm algorithms. In solving optimization problems like NP hard (such as traveling salesman problem) or action-response planning (such as a chess game), the target most of the time is finding an acceptable solution. This solution is not the best solution ever, but it is fitting predefined constraints. So swarm algorithms are used to find the appropriate solution without caring about the optimal hidden solution.

Particle swarm optimization (PSO) and Ant algorithm are the most popular and eldest swarm algorithms in solving optimization problems. While Artificial Bee Colony (ABC) is also a highly prevalent algorithm in these problems [4]. Others swarm algorithms like bat algorithm are also members of this intelligence technique family.

The main focus of this chapter is on ABC algorithm, standard one, and other versions of ABC like AABC, FMABC. Also, the chapter will discuss some case studies for applying ABC and how ABC modeling is in these cases.

2 Natural Bee Colony Behavior

In this section, the natural bees' behavior is presented. The main desirable behavior is the foraging process of the bees' colony.

The process starts with a worker bees who hunt food sources. Afterwards, the worker bees return to the hive. During their way back to the hive, the bees dance a unique dancing. This dance is called a waggle dancing.

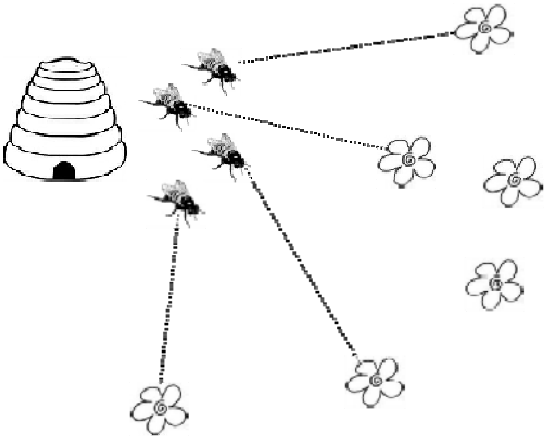


Fig. 1. Real bees foraging behavior

The waggle dancing describes specifically some information about the fetched food sources. This information includes the direction to the source, distance and food assessment. Using the bees' tail as presented in figure 2, the waggle dancing is happening. Other bees in the hive monitor the coming bees dancing to realize which food

source is tempting to be visited. According to the dancing parameters, the waiting bees choose a food source and go for it, then do the same dancing in their return. By that cooperation and the knowledge sharing, the overall bees' community can take a proper decision in the foraging process, although each bee by itself cannot do that without a help.

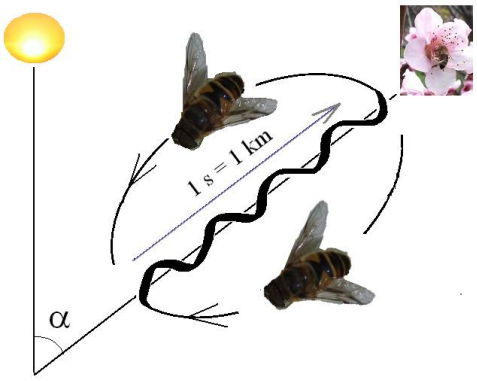


Fig. 2. Bees waggle dancing [9]

3 Artificial Bee Colony (ABC) Algorithm

ABC is bio-inspired swarm algorithm, simulates the behavior of real bees' colony in an artificial searching technique [3]. Like all swarm algorithms, the algorithm bases on a population. This population members are solutions from the problem search space. ABC considers solutions as food sources. The target of ABC is finding an acceptable food source (solution) with particular evaluation. Each solution is evaluated according to the evaluation function. This function definitely changed from a problem to another. Conceptually the solution has a fitness value (function output), and this fitness value is used in solutions selection methods. In ABC, the population compose 3 types of bees [5, 7]:

1. Worker bees: real worker bees are working on the food source allocating and nectar extraction. In ABC, artificial working bees are capturing the visited solutions. According to their solutions' fitness (waggling dance), the other bees types determine their way of working. Each working bee attaches one solution. The solutions' population size is equal to the amount of working bees in the bees' generation.
2. Onlooker bees: referring to the natural bees behavior, there are waiting at hive bees that watch the dancing and choose which source is better, these are the onlooker bees in ABC. Onlooker bees do two main activities: select part of worker bees' solutions according to the selection method, and do a local search activity on these selected solutions.

3. Scout bees: to avoid falling into a local minimum trap, scout bees play the role of introducing entirely new solutions to the bees' generation.

Here, we present ABC algorithm steps as presented in figure 3.

1: Initialize Population
2: **repeat**
3: Place the employed bees on their food sources
4: Place the onlooker bees on the food sources depending on their nectar amounts
5: Send the scouts to the search area for discovering new food sources
6: Memorize the best food source found so far
7: **until** requirements are met

Fig. 3. ABC Algorithm [2]

1. The initial population is created in step one. Initial population is created randomly in most of the implementations. While in some cases, the results from abstract greedy search algorithm are used as an initial optimization generation. This targets starting based on well-defined generation members.
2. In step 2, the iteration starts.
3. In step 3, Worker bees associate to the solutions' generation. Worker bees present the fitness value for the attached solutions to the onlooker to do their work. In some researches/implementations, the worker bee does a small scale local search in the close neighbors. In both cases (doing local search or not), each worker bee should select one solution to be part of the onlookers selecting options.
4. In step 4, onlookers observe all the worker bees' solutions, the through the selection method, they elect a part of these solutions to do a local search around them, the selection procedure in ABC is a roulette wheel method. Each solution probability in the selection process depends on its fitness value comparing to all the solutions' fitness values as presented in equation 1.

$$P_i = fit_i / \Sigma fit_n \qquad \text{where } n=1 \text{ to generation size} \tag{1}$$

5. Step 5, due to the nature of searching in the local areas by onlookers (or even workers) the risk of falling in a local minimum trap increases. Scout bees find new solutions per iteration to add them to the overall pool of solutions and try to avoid the trap. In standard ABC, scouts do that work by generating 100% random solutions.
6. By step 6, ABC has a pool of solutions, this pool is composed of worker solutions, onlooker searching output, and scout random introduced ones. In this step, the new generation solutions are chosen from the solutions' pool to be attached to the next generation worker bees. The selection bases on solutions fitness values.

7. By step 7, the iteration is finished, a new one with the new generation should start if the exit criteria are not met. The exit criteria might be a number of iterations, or fetching a solution with minimum defined fitness value, or any of them [6].

4 Adaptive Artificial Bee Colony (AABC) Algorithm

AABC is an algorithm based on ABC standard logic with some enhancements [2]. AABC considers the same bees' types as in ABC. The main modification in AABC, that the roles of the bees become dynamic. The bee can change its role between worker, onlooker, and scout depends on the fitness values from iteration to another.

As presented in figure 4, there are two additional parameters in AABC:

- Maximum worker count.
- Minimum worker count.

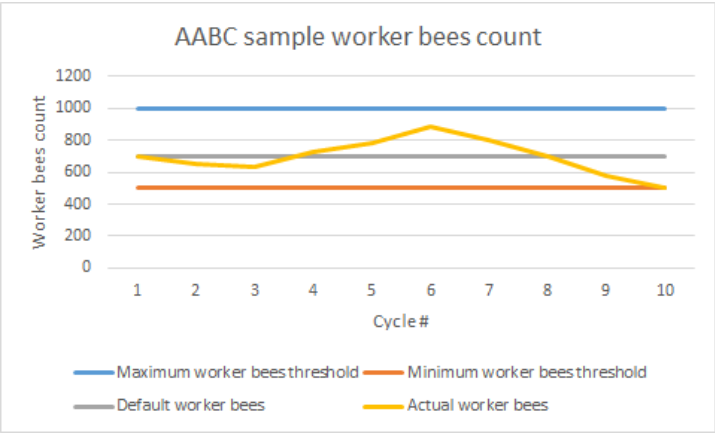


Fig. 4. AABC sample worker bees count

The worker bees' amount may vary within this margin depends on generation fitness quality. The worker, onlooker, and scout bees have a default values like in ABC. AABC introduces the concept of keeping the last iteration fitness mean value. This value is calculated by the mathematical mean of all the last generation fitness values.

During the selection of the new generation worker bee's process, the worker bees' amount might change according to the available solutions comparing to the previous generation fitness mean value. The selection of the worker bees keeps getting the best bees until reach the AABC minimum worker bees count. Then if there are others bees with fitness value greater than the previous generation mean, AABC will consider these bees as workers in the new generation, taking into consideration not exceed the AABC maximum worker bees count limitation.

While the overall bees' amount should be kept immutable, so the delta of worker bees should change their roles to another bees' role. If this delta is positive (current

worker bees > default worker bees), this delta will come from reserved onlooker bees. On the other hand, if the delta is negative (current worker bees < default worker bees), the rest of worker bees change their role to play as scout bees in the next generation. Figure 5 simulates the adoption relation between the different bees' types.

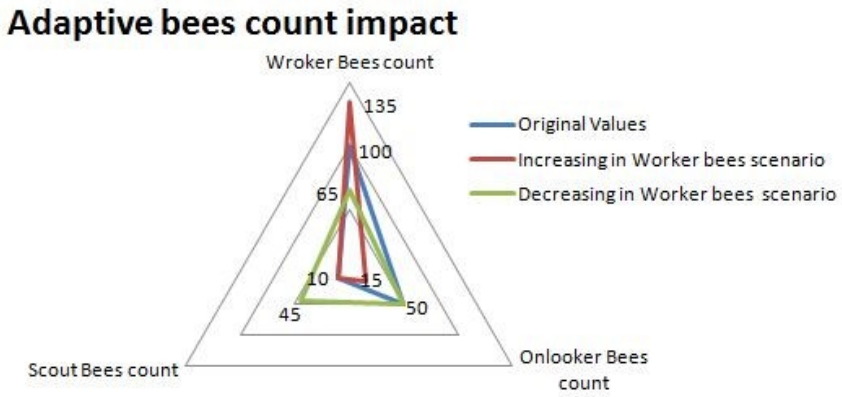


Fig. 5. AABC Adaptive bees count impact [2]

As presented in figure 5, the blue shape shows the original bees count parameter (100 worker, 50 onlooker, and 10 scouts). In case of rich generation (red shape), the worker bees are increased to be 135 bees, these additional bees come from onlooker bees (to be 15 instead of 50), the scout bees now is the same (10 bees). On the other hand in poor generation (green shape), the worker bees are decreased to be 65, this delta number of bees become scouts, the scout here is 45 bees instead of 10, the onlooker bees in this case are resettled to be the default again (50 bees). All these samples values are based on that: the maximum worker bees' parameter is 135, and the minimum is 65. The change in the worker bees can be within this range. It does not have to be just these three values (default, maximum and minimum), but it might have any value within this range (the bees' job exchange works the same as methodology described above).

Figure 6 presents AABC algorithm in a flow chart diagram.

Experiments applied on AABC comparing to ABC (on traveling salesman problem) proves enhancements by around 8% of the fetched solution after the same number of cycles and using the same total bees' generation size.

5 Fast Mutation Artificial Bee Colony (FMABC) Algorithm

FMABC is an improved ABC algorithm presented in research [12]. The main change in FMABC is:

Instead of create random solution by scout bees, the scout bees will do a mutation on the existing solutions.

FMABC is inspired by mutation concept in Genetic algorithms, so the author of it propose using mutation operation on an existing solutions instead of create random one from scratch. The paper [12] states that FMABC provides better performance that ABC while using benchmark data of mathematical functions calculations.

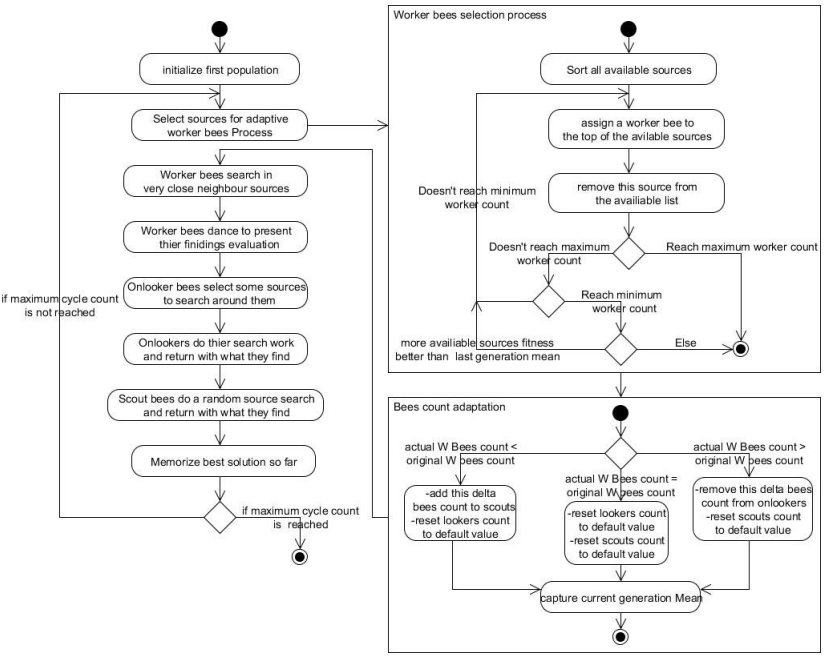


Fig. 6. AABC Algorithm workflow [2]

6 Integrated Artificial Bee Colony Algorithm

Using ABC lonely is one technique. Another technique is combining the usage of ABC with other optimization techniques that what is presented in this section.

6.1 Integration with Particle Swarm Optimization

Particle swarm optimization (PSO) is one of the most commonly known swarm optimization technique. “Integrated algorithm based on ABC and PSO” (IABAP) is a proposed algorithm in research [13]. IABAP provide two communication processes: one to share information from ABC to PSO, another one in the opposite direction.

As presented in figure 7, ABC and PSO algorithms are working in parallel. Per iteration, depends on a probability condition (which might be a random probability, or periodic activity) the information is shared between two algorithms to adapt their behavior based on them.

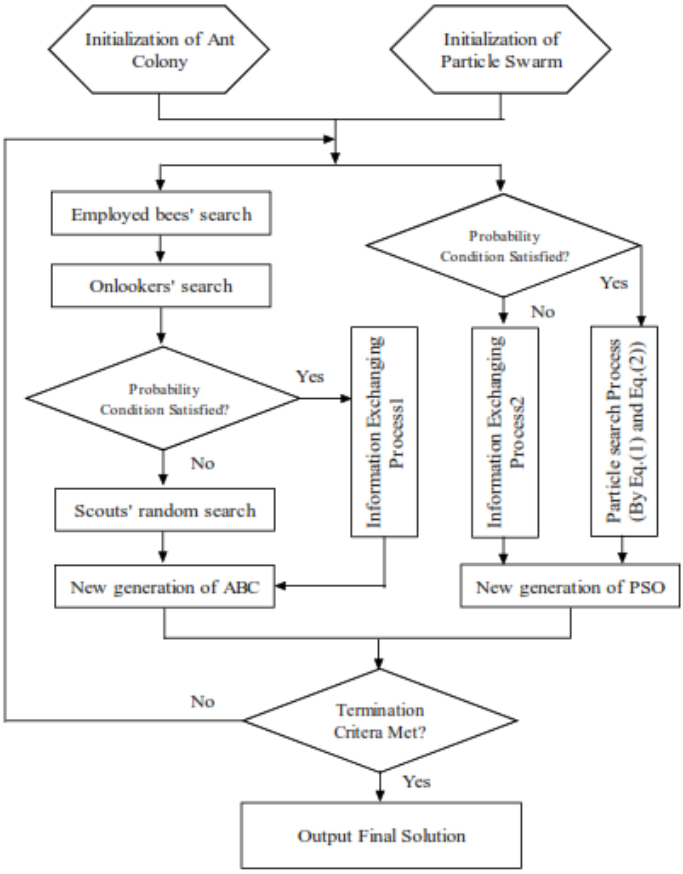


Fig. 7. IABAP algorithm [13]

7 Case Studies

In this section, the chapter presents samples of case studies for ABC algorithm, and how ABC is modeled for them. ABC modeling is not one way. Same problem might be modeled by different ways with the same algorithm, and here we present one of these modeling options.

7.1 Traveling Salesman Problem

Traveling salesman problem (TSP) is an NP hard problem that need to find a path between multiple cities, start and end city is the same, other cities are visited only once with the minimum route cost. Given a graph G= (V, E) where V is a set of cities V= {V1, V2…} and E is a set of edges representing the distance cost between cities is E= {(r, s): r, s ∈ V} [7, 10]. ABC is used as an alternative solution for TSP [1, 8]. Also, TSP is used to compare ABC vs. AABC [2].

ABC-TSP modeling.

A solution in TSP, is a sequence of cities.

$$S= \{V_x, V_y....\} \text{ Where } 1<=X, Y<= \text{ count of cities.} \quad (2)$$

ABC modeling for TSP bases on: each solution is a unique cities order. Initial generation in ABC might be created randomly taking into consideration the solutions valid conditions. Another modeling way is making the initial generation based on output of greedy algorithms like "Nearest neighbor". This would build an exceptionally strong initial population, so the needed enhancements from ABC is minor comparing with the case starts with total random generation.

Onlooker local search in TSP could be modeled as a cities order exchange. By imagination, if the sequence of cities is a solution, the neighbors of this solution will be the same overall cities order with minor order changes (2 cities exchange their order). This model is valid if the problem graph is complete connected with bi-direction edges. If this condition is not happening, exchange any two cities order need to be validated based on the graph connections to ensure the solution validity.

7.2 Job Scheduling Problem

Processors always have multiple tasks to run. Job scheduling problem is how to find the best way to sort the jobs for execution. Greedy algorithms like FCFS solve the scheduling problem, but does not ensure that this is the best solution can be fetched. Using ABC to solve the optimization scheduling problem would lead to a better solution (don't forget that ABC does not target the optimal) [6].

ABC-Job Scheduling Modeling

Sorting the jobs in an order is considered a solution. Not far away from ABC-TSP modeling, having the sequence of the jobs could be randomly created for the initial generation. The fitness function in scheduling problem is different from TSP, but using elements order exchange as an onlooker local search is a good technique. For FMABC, the scout bee's activity can be happened by sequence mutation instead of total new sequence.

The same modeling is fitting into bio-informatics problems, especially in gene and sequence alignment calculations.

7.3 Software Testing

In software testing, one of the main problems is how to generate the test data to reveal the most faults in the software. Many techniques are used in that objective. From input domain, the objective is finding the input data. This search activity can be done by random testing or mutational testing. Some researches use "Metaheuristic algorithms" such as Genetic algorithm, PSO, and ant colony. Also, ABC is considered in this search scope. In research [11], it uses ABC algorithm to generate a test data branches randomly in the initial population. Then, ABC keeps searching for better testing data according to fitness function evaluation, which reflect the coverage of software faults probability.

8 Conclusion

In this chapter, artificial bee colony algorithm is studied. Standard ABC is presented, also improved versions of it like AABC and FMABC are described. Integrated ABC with PSO algorithms is also discussed. Case studies of ABC, and how they can be modeled in ABC are described in this chapter.

This chapter tries to be a short reference for ABC and their updated algorithms. It shows how ABC could be used and implemented with real commonly known problems.

References

1. Barvinok, A., Tamir, A., Fekete, S.P., Woeginger, G.J., Johnson, D.S., Woodroofe, R.: The Geometric Maximum Traveling Salesman Problem. Journal of the ACM 50(5), 641–664 (2003)
2. Rekaby, A., Youssif, A.A., Sharaf Eldin, A.: Introducing Adaptive Artificial Bee Colony Algorithm and Using It in Solving Traveling Salesman Problem. In: An International Conference of Science and Information (SAI), London, UK. IEEE (October 2013)
3. Karaboga, D., Basturk, B.: On the performance of artificial bee colony (ABC) algorithm. Applied Soft Computing 8, 687–697 (2008)
4. Karaboga, D., Akay, B.: A Comparative Study of Artificial Bee Colony Algorithm. Applied Mathematics and Computation Journal, 108–132 (2009)
5. Jones, K.O., Bouffet, A.: Comparison of Bees Algorithm, Ant Colony Optimisation and Particle Swarm Optimisation for Pid Controller Tuning. In: International Conference on Computer Systems and Technologies, CompSysTech 2008 (2008)
6. Wang, L., Zhou, G., Xu, Y., Wang, S., Liu, M.: An effective artificial bee colony algorithm for the flexible job-shop scheduling problem. Int. J. Adv. Manuf. Technol. (September 2011)
7. Wong, L.-P., Low, M.Y.H., Chong, C.S.: A Bee Colony Optimization Algorithm for Traveling Salesman Problem. In: Second Asia International Conference on Modelling & Simulation, pp. 818–823. IEEE (2008)
8. Fatih Tasgetiren, M., Suganthan, P.N., Pan, Q.-K.: A Discrete Particle Swarm Optimization Algorithm for the Generalized Traveling Salesman Problem. In: GECCO 2007. ACM (2007)
9. Brown, P.: `http://www.scilogs.com/from_the_lab_bench/super-hero-experiment-2-the-waggle-dance/` (October 19, 2013)
10. Arora, S.: Polynomial Time Approximation Schemes for Euclidean Traveling Sa-lesman and Other Geometric Problems. Journal of the ACM 45(5), 753–782 (1998)
11. Dahiya, S.S., Chhabra, J.K., Kumar, S.: Application of Artificial Bee Colony Algorithm to Software Testing. In: 21st Australian Software Engineering Conference. IEEE (2010)
12. Bi, X., Wang, Y.: An Improved Artificial Bee Colony Algorithm. In: 3rd International Conference on Computer Research and Development (ICCRD). IEEE (2011)
13. Shi, X., Li, Y., Li, H., Guan, R., Wang, L., Liang, Y.: An Integrated Algorithm Based on Artificial Bee Colony and Particle Swarm Optimization. In: Sixth International Conference on Natural Computation (ICNC 2010) (2010)

Novel Approaches to Developing Multimodal Biometric Systems with Autonomic Liveness Detection Characteristics

Peter Matthew and Mark Anderson

Department of Computing
Edge Hill University
Ormskirk, UK

Abstract. Liveness detection in biometric systems has become an integral part of system viability, but it has innate disadvantages concerning implementation, situational suitability and acceptance. This article looks at the potential for combining liveness detection techniques with autonomous concepts to minimize, negate or even improve the original system. This is done by considering two potential areas within the autonomous system purview, autonomous architectures and the human nervous system paradigm. Within each there are a number of areas that could accept liveness detection incorporation and potentially improve each applicable subsystem. This article will cover an introduction into these topics and a discussion about their suitability.

Keywords: biometric, liveness detection, autonomous systems, template capture, human nervous system paradigm, biometric security, autonomous architectures.

1 Introduction

According to [1] biometrics are "the automated use of physiological or behavioural characteristics to determine or verify identity". This is expanded on but otherwise verified by [2] identifying that the biometric system can be used mainly to do two things, it can verify the user and it can discover the identity of the user [3] [4]. The verification process works because it already knows which template to compare the user to and thus is a quick process. Identification is the complete opposite, as it has to enroll the new user and identify them. In either case the first thing to be complete is to capture a working template record. Therefore some system capability to detect the state of the user is needed to make sure that spoofing of the required biometric does not occur, here liveness detection is needed to allow this level of robustness.

The second aspect of this paper will be concerning autonomous systems. Computers and system complexity are becoming more dynamic and fast paced every year, as [5] postulated systems will beget and evolve new systems that can overcome problems that the parent system could not, thus making more complex and intelligent systems. [6] identified some potential problems which were going to occur in the subsequent years, as systems developed and become more complex but the

L. Chen, S. Kapoor, and R. Bhatia (eds.), *Intelligent Systems for Science and Information*, Studies in Computational Intelligence 542,
DOI: 10.1007/978-3-319-04702-7_7,

corresponding ability of the humans administrators involved in such systems did not improve at the same rate [6].

This paper aims to identify some of the main areas that autonomous computing can be incorporated within biometric liveness detection systems, as well as the best location to incorporate said systems. The first stage is to capture the biometric template data from the user.

2 Biometric Architecture

A biometric device typically has five main components which equate to a sensor, a feature extractor, a template database, a matcher and a decision module [7] [8] [9] [10]. The interaction between these components is depicted in Figure 1. Each feature performs its own task and forwards the information on to the next link in the chain of components.

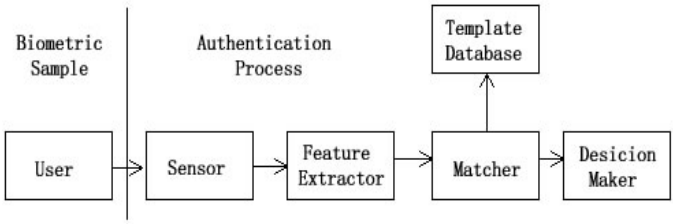


Fig. 1. Biometric Authentication Process

This depicts the general process when dealing with a single biometric method, otherwise known as a unibiometric. There are a number of problems that are associated with unibiometrics, which mainly rotate around the general poor level of security within.

This process highlights the enrolment aspect of a biometric system, it does not identify the process in which general authentication occurs. When a user authenticates within a biometric system, they provided a sample of data, which will need to be mathematically similar to the enrolled template to allow access to the system. This uses a one-to-one match system, thus a user is claiming to be a specific person, and thus the matching occurs with the knowledge of the user's identity, thus being one-to-one. If it is identification instead of authentication this is not the case. Identification is the process of finding out whom a user is. This means it can be a one to many, or a one to few process, both of which contains more system requirements and resources than a simple authentication. Whilst this process is quite simple it is applicable to a lot of different areas within biometrics, however to provide adequate levels of security to biometrics, it has been identified that liveness detection, and various protection schemas are needed to maintain the robust security factors that biometrics require, to be a valid security consideration. Coupled with the drawbacks identified in Table One, a singular implementation of unibiometrics is unlikely to provide the security, robustness and validity that is needed within a system.

Table 1. Unibiometric drawbacks

Limitation	Effect	Multibiometrics Solution
Noise	Can be very susceptible to noise, only one biometric leaves little margin for error.	Multiple biometric allow different form of data capture depending on environment.
Non-university	Can prevent user utilising the system due to lack of biometric sample (Amputee etc.)	If a user has difficulty using one form of biometric then there will be other form available for authentication e.g. cataract causing iris samples to fail, change to fingerprint
Maximum boundary	There is an upper limit of sample improvement, unlike passwords that can be infinitely made more complex, unibiometrics have a limited gambit of scalability	Whilst each individual biometric has the same flaws, the combined levels of multibiometrics increase the potential complexity exponentially as well as providing a vastly improvised degree of scalability, which would lead to a more complex management of sample.
Spoof Attacks	Due to the singular nature of unibiometrics spoof attacks have a much greater chance of being successful, especially within certain ranges, such as vocal biometrics.	Multibiometrics make it exponentially harder to spoof a system, as the spoof need to counter the multibiometrics that are gathering the sample.

This has a number of problems, mainly that most biometric security techniques are not as secure as they should be when considered singularly , instead almost all of the research indicates that multimodal biometrics are much more powerful, and can provide a more robust system. When combined with the relevant security factors, such as threat vector protection, liveness detection, and template protection a multimodal approach is more applicable. The general process architecture of multimodal, or multibiometrics, follows the same path of a single biometric, or unibiometric system, however there are a number of different considerations when dealing with multiple levels of biometric devices. The distinguishing factor that improves multibiometrics is the use of multiple biometric systems to create a more robust and secure collection client, thus enabling a more valid data capture and authentication technique to be identified. This is done by combining multiple forms of biometric data capture concepts [40]. There are four main forms which are sub-sectored here with slight differences in their applicability, the first will scan specific divergences such as 2d and 3d cameras, secondly algorithmic fusion, such as using minutia features as well as ridge features, thirdly unit differentiation such as index and fore fingers, and finally trait divergence, using different biometric forms such as finger and palm scans [41][42].

These features correspond to a grouping of six multimodal styles, each of which has their own advantages and disadvantages as identified within Table 2.

Table 2. Forms of Multibiometrics implamentaon

Sample	Advantages	Disadvantages
Multi-sensor	Can gather complementary data from the sample which improves the level of fusion.	Expensive, need to acquire additional hardware [41][43].
Multi-algorithm	Improves matching data Cost effective	Can cause high computational load on system resources [41]
Multi-Instance	Cost effective (if samples sequencially obtained)	If simultaneous collection is required can require expensive hardware, or high computational resources [41].
Multi-sample	High entropy High upper boundary	Must correctly identify number samples Needs a lot of pre-sample collection work [41]
Multi-modal	High level of security and robustness	Similar characteristics are not as robust as non-similar Expensive Lots of restrictions [41]
Hybrid	Can include an integration of the above sections and thus can include the salient advantages and disadvantages of all.	

As well as the different styles of multibiometrics, there are also methods in which to combine the data being provided and these areas depend on the form the data fusion takes, and what can be accomplished at different areas within the architecture with it. There are two levels that biometric fusion can be accomplished, each with sublevels, the two headings are pre and post matching [41].

There are five areas in which fusion can occur, each reacting differently to the data provided. Sensor level fusion utilises the most feature rich data, the raw data gathered from the user, and whilst this offers a highly abundant set of features it also can contain a lot of noise and thus the corruption of the sample. Sensor fusion occurs primarily between multi-sensor and multi-instance forms of multibiometrics [41] [44]. Feature–level fusion is normally utilised when considering in multi-algorithmic situations, in which the feature sets are normalised and the salient comparisons are undertaken, and whilst this method can potentially provide excellent comparison data there can be issues when using the reduction methods, and the necessity to have a much larger range of training data sets [41][45][46]. Score level fusion is designed to match the data by a fusion of the separate biometric scores and whilst this is often the most used method, there can potentially be major problems, not least the vulnerability to security breaches, as the fusion occurs later within the authentication processes providing an easier route to security threats such as spoof data [47]. Rank level fusion occurs which the biometric sample of identify the user does not authenticating them,

the individual biometric output a ranking system for the user which in turn is then compared and fused together to create an overall rank for the user. This is a very useful form of fusion as it enables am auto normalised scale to be compared and thus is not as resources dependent, and easier to interpret [48]. Finally, decision level fusion is often most used due to the prevalence of commercial device and software in which only the final decisions is often available, and thus the prevalence of Boolean style rules as well as other fusion based techniques such as Bayesian fusion must be used to correlate the overall features set and acceptance level of the sample [41] [49].

All of these aspects provide a number of options when creating a multibiometrics system, and whilst there are a number of advantages and disadvantages associated with each form, overall they provide a high degree of robustness and remove a lot of the inherent problems that occur within unibiometric devices.

It has also been suggested that the False Acceptance Rates (FAR) and False Rejection Rates (FRR) are very important [4]. [3] and [4] then goes on to add an equal error rate (ERR) which is set by the system designers. This rate is a devised by deciding on a level of error that is acceptable to security, cost and user effectiveness. It is useful to remember that no system is completely secure and developing good threat models as set down. [7], [8], [9] and [10] helps produce a system with an acceptable rate of errors.

Biometric authentication is becoming a mainstream technology according to [4]. However, it still has not gathered universal integration [11], as only 15% of organisations within the U.S. were using biometric devices according to CSI/FBI Computer Crime survey in 2005 due to, primarily, concerns about privacy and security [12].

This can be very important at a social level as certain biometric capture styles might cause user uncertainty or fear such iris/retina recognition. Additionally, certain environmental factors such as humidity levels, or brightness levels, might affect a number of devices from fingerprints to facial scans [4], this is known as noise. This noise interference could range from difficulty authenticating a user to template capture problems [13] [14]. Finally algorithmic integrity must be robust enough to only gather the relevant data from a sample and discard the superfluous detritus [13].

3 Biometric Security

There are a number of important features to consider within the design of a biometric security implementation such as how powerful a device must be to gather the best sample. As with many emerging technologies, including biometrics and ubiquitous computing, there is one major problem: a noted lack of coverage, including standardization [15]. These standards had to become significantly more prevalent if the areas were, and are, to be more finely integrated within modern society [4]. This will take a number of years to happen as most of the international electronics vendors would have to be involved, including companies such as LG and Sony [16].

When dealing with biometric devices, it is understood that a number of features will be needed to aid the template-gathering aspect of the device, the security and the reliability. These features can include data validity, maximising data gathering, and

liveness detection. As [17] and [18] postulate using the "gummy" fingers research, liveness detection is possibly the most important security aspect; a fact which was later supported by [19] who proved that a device using liveness detection maintained a success rate of false authentication which dropped from 90% success to 10% success. Liveness detection will be covered in Section V. A secondary, but almost as important, concern is centered upon biometric techniques which might differ depending on the surroundings [13].

As biometric devices can have many different characteristics within each scanning process, a number of technologies have either been adapted or developed to aid the process. These processes include minimal distance, probabilistic methods and neural networks [4] [20]. It has been identified that neural networks are potentially very useful aspects of technology as they learn certain characteristics about an authorized user so that it is quicker to match and accept or deny that authentication request [4]. This would be very useful and, if reliable, could improve the security and efficiency of certain devices. However there are some issues with neural networking [4] [20]. Firstly the configuration of the neural network, making sure that there is enough computational ability for the relevant network size, and considering additional issues regarding storage and specific training would have to be taken into account [4] [20] . This is potentially one of the more problematic issues, as depending on the size of the neural network a vast amount of storage would be needed thereby increasing the cost and decreasing the efficiency. If the intended audience for the biometric device is large then optimising and configuring the system would be a resource intensive task. This potential shows an opportunity for integration within autonomous systems, especially considering self-optimization and configuration. One of the issues that a neural network deals with is the problem of impostor data. By Storing this data the system may quickly find potential impostors without many of the original computational issues [4] [20].

Whilst intrinsic attacks deal with threats that emanate from within the system and thus do not have an external nefarious instigator, adversary attacks are just the opposite. These attacks occur when third parties stage attacks that are designed to intentionally gather information or get access to a system without proper authentication. Adversary attacks are dependent on the discovery and exploration of loopholes/ errors in the system, which can be taken advantage of in some way. These loopholes are often founds in areas where the different hardware and software components of the biometric system are found

There exists a number of attack styles in which these loopholes can be utilised. Firstly as all systems need maintenance and constant administration to function correctly, some error, mishandling of procedure or simple fault can cause errors. These administration threats can cause a route into the system for an external threat as they can exploit the relevant loophole and whilst this issue can be negated somewhat with correct instigation of autonomous system features, such as self-configuration, the potential for the threat is still concurrent. These threats can be caused by a plethora of different reasons ranging from incorrect enrolment: e.g. the systems administrator provides greater system access than is necessary for a new enrolee. There are also the

issues that are caused by coercion which may be centered on either a systems administrator or a legitimate enrolled user; to coerce of a user, which would involve the pressure of the relevant parties to authenticate for a nefarious user. This is an area to consider, as is it possible to detect coercion within a user making use of liveness detection as a standard. The insecurity of the infrastructure may also cause area that are prone to attack as there are many instances were a lack of security in the transmission of data between the different entities in a system has been exploited in some way [8].

This concept culminates with biometric threat vectors, which endeavour to identify the potential areas in which attacks will be most devastating, and most easily implemented. [8] postulated that there were eight points of attack in biometric systems, and whilst these points are quite general they encapsulate the main threats that a system would be susceptible too. This did not identify device specific additional threats that may occur, but was intended as a generalisation for biometric systems [8]. Whilst this initial model was valid, it became obvious that it did not go into sufficient detail on the potential vectors and thus [9] and [10] looked at expanding on these ideas. It is imperative that these vectors are considered, along with the expanded vectors, when considering the inclusion of liveness detection within a system

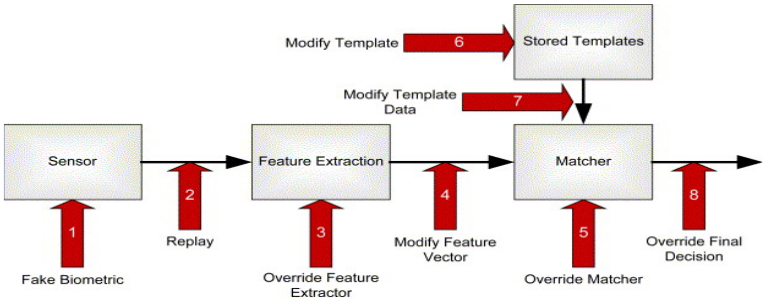


Fig. 2. Biometric threat vectors based on [8] Framework

Figure Two identifies the eight threat vectors that [8] described. The initial attack presented was the attempted enrolment of a fake biometric sample; this could include fingerprint, facial recognition, signature theft etc. such as a gummy finger creation, video/image spoofing, vocal recording, as discussed, respectively, by [18][17] [53]. Solutions such as improved quality of scanners and [21] identified this error as the primary liveness detection environment. Some of these vectors were initially oversimplified however, over the years they have become more technical and applicable to the current field of study. [9] has identified five different subsections of a general biometric systems and indicated that, although each style of biometric could use completely different technologies, there were many generalisations that could be made. One such generalisation was that five sub sectors could encapsulate many of the relevant biometric devices [9]. These additional vectors encapsulated many of the original vectors in [8]'s work as shown in Figure Three.

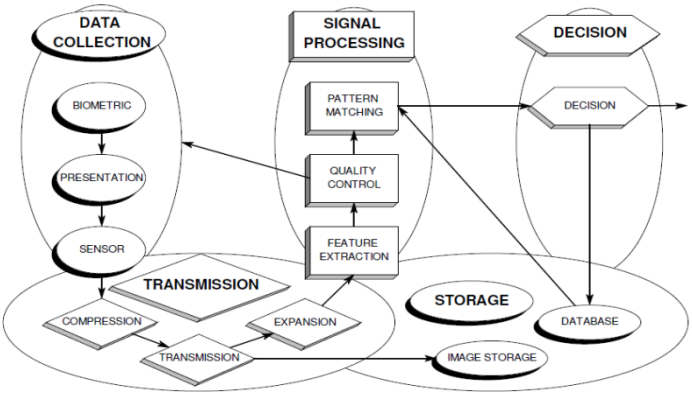


Fig. 3. [9] extended threat vectors

As with [8]'s work, the main are within this model that liveness detection can be implemented is within the data collection environment, thus providing the additionall security levels within the biometric, presentation, and sensor levels. As there are different threats that correspond to each of this levels then the autonomic fusion would allow a more robust integrations, due to the segregation of modules.

Others also improved upon [8]'s work, and this iterative process was conducted by a number of researchers. [7] identified four vector categories, and [10] further extended this work by creating a framework that was based around [9]'s five sub sector model but identified appropriate attack vectors that directly correspondent with it, making a framework with twenty identified attack points and twenty two vulnerabilities.

4 Liveness Detection

The purpose of liveness detection is to verify and confirm the sample being presented to a biometric device is both from a correctly authenticated and living user, who is within the locus during the authentication process [19]. The ideal is that liveness detection removes the risk of a user's samples being used by a third party whilst they are not there. This can be achieved by using spoof biometric samples gathered from their legitimate data source. This could take the form of lifted fingerprints, recorded voice recognition, or facial images for facial recognition [21] [22] [23].

This additional security feature is necessary as it has been demonstrated that, whilst biometric security is very efficient and effective as a security technique, there is a specific need to include liveness detection as spoof data is neither hard to gather nor hard to use, depending on the device under attack [24].By including some form of liveness detection the risk of security breach due to spoofing attacks is dramatically reduced [21].

There are many methods and examples of liveness detection currently in operation. These include pulse, blood pressure or electrocardiograms (ECG)[25]. Whilst most forms of biometric device require this liveness detection as an additional piece of

data, there are some forms that have inbuilt liveness detection due to the characteristics that are being looked for. An clear example is evidenced in ECGs. This is because to gather ECG data then the user must have a living heart [25]. One factor must always be remembered, that whilst liveness detection brings a host of advantages to a system it also brings a number of additional considerations as the system is including an additional layer of complexity and potentially components which need to be secure. Issues that could affect liveness detection sub-systems include performance, such as multi-user capabilities and location variables, as well as other factors such as ease of implementation, heterogenic, user acceptance etc. [3] [21] [22]. There are a plethora of liveness detection techniques that are available and more are being discovered and tested constantly. Whilst there are a number of techniques they all have advantages and disadvantages associated with the medium, Table 3 illustrates a selection of the techniques and the salient points within each.

Table 3. Techniques of liveness detection

Liveness detection technique	Advantages	Disadvantages	Biometric device
Temperature	1. easy to check 2. easy to implement	3. not very secure 4. easy to spoof 5. lots of research on the topic[21]	6. Fingerprint 7. Hand print 8. Earprint etc.
Optical Properties Spectrographic properties	9. Very secure 10. Can detect a number of difference such as glass eyes, dead tissue	11. Main forms of gummy spoofs have similar optical properties to skin thus minimising the use. [21]	12. Fingerprint 13. Hand print 14. Earprint
Pulse	15. Highly acceptable by the general population 16. Easy to gather.	17. Easy to spoof 18. Can differ dramatically due to host of factors – health/fitness of user, medical condition, recent activity etc.	19. Fingerprint 20. Hand print 21. Earprint
Pulse Oximetry	22. Medical application	23. Can be fooled by user providing their own sample covered with spoof i.e. fingerprint. 24. Difficult to gather	25. Fingerprint 26. Hand print 27. Earprint 28. Vein Scan

Table 3. (*continued*)

Relative dielectric permittivity	32. Potentially very accurate 33. Depending on device capture technique can be easy to implement	34. To get an accurate reading the admissible range would include that range of spoof samples	35. Fingerprint 36. Handprint 37. Vein Scan
Infrared/ Ultraviolet/ Thermal scanners	38. Can be very effective at deterring spoof samples. Can show temperature, vein layout etc.	39. Very expensive 40. They have to be installed and thus ad-hoc authentication is not available.	41. Any device that depends on optical scanner.
3d Head Movement	42. Prevents most basic forms of spoof (i.e. pictures of the user)	43. Can still be spoofed using video as a medium. 44. Certain techniques are very prone to noise e.g. depth information vector [50][51]	45. Facial
Micro Movements	46. Quite intuitive 47. Comparison between the muscle movements when speaking occurs 48. Blinking	49. Sometimes relies on additional hardware [51] [52] 50. Biometric dependent	51. Facial
Text Prompting	52. Integrated in the general form of vocal biometric styles. 53. Very cheap to implement	54. Vulnerability from text to speech software's	55. Vocal 56. Facial
Hippus deviation	57. Little is known about it. 58. Good at removing spoof attacks	59. Can be adversely effected during the again process [21]	60. Iris

Liveness detection normally conducts its checks within either the acquisition stage or processing stage of biometric authentication as these are the most relevant areas when dealing with a user's biometric input. This process is normally achieved by implementing the liveness detection in one of the following three methods [22].

Firstly the inclusion of additional hardware which in turn can use the relevant data captured by the biometric device or, depending on the biometric in question, allow the use of the already present liveness features that some biometric devices possess [21]. The primary disadvantage of this form of liveness detection is that there is an acquisition of new hardware which introduces the problems associated with any new piece of hardware such as cost, size, heterogeneity and suitability issues. One primary advantage would be the ability to add complex hardware thus making the overall system more robust and allowing an easier integration of additional features [22].

The second method involves using information that has already been captured by the original biometric device. This method is very cost effective as it does not require additional hardware. However, there would be a number of software related factors to consider [22]. The most important issues is the complexity of the system needed to gather this additional liveness information. This is often prohibitive to include, thus making additional hardware seem a better option [21].

The final method is to use the liveness characteristics inherent in some biometric characteristics, but which has not automatically been gathered when the sample has been provided. However, as this feature is not a universal aspect of biometric selectors there are a number of problems when considering the heterogeneity of the potential systems, as well as consistency of data, and user needs [21] [22].

Whilst the implementation of biometrics is an important initial consideration there is also the categorisation of liveness detection parameters. There are three areas in which liveness detection characteristics fall under [12].

Firstly there are intrinsic properties of a living body. These intrinsic properties can vary depending on type of sample required. For example there could be physical characteristics such as density and elasticity, electrical permutations such as the resistance and capacitance of a sample and spectral characteristics such as colour and opacity [21] [26].

Secondly there are the involuntary signals of a living body. These are data samples are unconscious and can provide a lot of liveness detection data. Examples include perspiration, blood flow and EEG signals. As these samples are primarily medical in nature it would be a small integration to also allow involuntary liveness detection signals to be used for medical purposes as well, such as wellness detection potentially with integrations into smart homes systems [21] [26].

The last area is the way the body reacts to external stimuli. Whilst the previous two sections deal with the body's internal and external functions, this form of liveness detection deals with external stimuli invoking a response from the body. This is normally one of the most regularly used forms of liveness detection and are based on a challenge response paradigm. This necessitates a cooperation between system and users. Examples include changing facial features, typing, changing of fingers for fingerprints, and so on [21] [26].

All of these liveness detection techniques are designed to achieve a single goal; to prove the living status of an authenticating user. They represent a number of options relating to implementation and suitability. However relevance may differ depending on specific scenarios [27]. A system with a high number of users may benefit more from a hardware based solution because of the robustness and quantity of users, whereas a small user base may not require the hardware solution [21].

Whilst liveness detection is a vital component of a biometric security system, there are often a number of perceived issues with these instigations. Hardware is often difficult to manage and maintain [21] [26] however if a gestalt system of both biometric and autonomous components is to be considered then a number of the primary disadvantages can be negated and an increase in security can be achieved. Certain concepts can be improved by integration with autonomous systems, especially when considering certain aspects of the human nervous system paradigm such as self-awareness, self-optimisation and self-configuration. These areas can produce a highly effective synergy between the biometric systems and liveness detection considerations that may be relevant [21] [27].

5 The Human Nervous System Paradigm

Autonomous computing was initially based on the autonomic capabilities of a living body and the human nervous system in particular [6]. Ashby's ultrastable system, was one of the underpinning concepts which identified that an autonomous system must be able to adapt to different behaviours depending on relevant inputs. This would be achieved by having variables that will change the behaviour yet maintain the systems equilibrium at a relevant level [28].

This adaptive behaviour is the most important aspect of an autonomous system and the most significant defining aspect, as by maintaining this state of equilibrium the system is able to survive, or recover, from faults or attacks [6] [28].

This concept has been expanded by comparing the similarities of a system's transmission methods, embodying the use of packets and messages, to the messages within the human nervous system [29] [30]. The functions that the human nervous system conducts are done automatically, allowing the host to devote all the time to higher functions without having to worry about maintaining the equilibrium of the body. Again, this is something that autonomous systems aim to replicate so that all of active runtime resources can be dedicated to specific higher tasks within the system, and all other functions are done so automatically within the purview of the system [29].

The breakdown of these systems identifies that there are a set of initial self* characteristics which expand to eight characteristics as the models have evolved. These characteristics are: self-awareness, self-configuration, self-optimisation, self-healing and self-protecting [31], and was then expanded upon [28] to include context aware, open, and anticipatory knowledge [30] [32]. As the model has evolved even more aspects have become apparent such as self-defining, self-governing and self-reflecting [29] [33] [34]. Whilst there are a number of areas that are applicable within this

scenario this paper will concentrate on the following Self-Configuration, Self-Optimisation, and Self-Protecting

6 Autonomous Synergy with Liveness Detection

Throughout the development of liveness detection subsystems, the main problem is that there are additional levels of complexity that must be considered. These are partially to do with the normal inclusion of additional hardware or software. There are specific areas that integrations may occur at an easier level than others. Some are discussed below.

Liveness detection must be included within the template capture area, as already mentioned this is normally done within the acquisition or processing stages within biometric authentication. Within these stages there is the potential to incorporate autonomous features, by including some aspects of the human nervous system paradigm, such as a self-aware biometric device which can detect for any tampering by keeping a self-state awareness. This would enable some additional spoofing defences as detection of additional nodes, or node tampering could be detected and dealt with [4] [9] [17].

6.1 Human Nervous System Paradigm Component Integration

As already mentioned there are a number of ways the human nervous system paradigm could be used as a base model to improve robustness and reliability within biometric devices. Each area can be used to potentially improve the efficiency of the system, certain aspects can be specifically used to improve liveness detection considerations, especially self-configuration, optimisation and protection [21] [26] [6].

1. Self-Configuration

The main issues relating to integration of liveness detection using hardware as the medium can potentially be negated using self-configuration. This would enable the system to dynamically install and configure the hardware liveness detection system, which in turn would enable a powerful liveness detection parameter that has many other potential features such as robustness and reliability. This would need to be coupled with self-protection as including another piece of hardware would increase the potential security threats that the system faced [21] [26] [6].

As the different forms of liveness detection can include parameters that need to be configured to facilitate the ability to self-configure, this would save time and increase efficiency. If additional hardware was included then there are other avenues of data that could be more easily collected such as medical data when integrating the system within a smart home, or pervasive space.

2. Self-optimisation

As with all system features the need for optimization can be an inefficient yet important process. At a simple level the ability to self-optimise will allow a more

efficient and effective system to run. There are more specific improvements that self-optimisation may allow. If the liveness detection of choice is using the intrinsic properties of the human body as samples, then the relevance of the sample may be very important to consider and having the ability to dynamically change the form the sample takes will allow a much more robust system. This would allow the changing of density and elasticity to colour and opacity for example, depending on the user's needs, profile, or system requirement. This potentially could link into important pervasive components that customizes the relevant features per user endeavoring to provide an accessible environment for biometric authentication, something that would have a very positive impact within a medical situation in case users were unable to provide certain biometric samples due to illness or impairment [21] [26].

3. Self-Protecting

Whilst system defenses are one of the main areas to consider within system implementation, by incorporating self-protection to liveness detection modules a much more robust security system can be implemented. One of the primary problems, especially when incorporating hardware based liveness detection is the added set of vulnerabilities within the system. However, when dealing with self-protection, this vulnerability could be negated, or substantially reduced. Other concerns are threat vectors for biometric devices and the integrations of an effective self-protection algorithm would provide a valuable layer of protection and defense [21] [26] [6].

6.2 Biometric Fusion Consequences

Fusion can occur at a number of points within a biometric system, as identified above. However this adds levels of complexity to a system that always occurs when additional features are implemented into any system. The degree in which fusion between multibiometrics differs provides a number of additional considerations when implementing liveness detection techniques. These considerations can dramatically alter the way in which liveness detection is implemented within a system due to the integral characteristics of biometric fusion. There are a number of areas in which fusion can occur, thus where does liveness detection correspond, is liveness detection implemented on each biometric before the fusion process begins, or does the technique await for the fused biometric. If this is the case then what will occur if the sample within one biometric is successfully spoofed, will the relevant biometric features be available? How will the different forms of both liveness detection such as using hardware, sample data or intrinsic values differ and change when fused with other biometric samples. These are a number of questions that must be considered and identified in the future. Will the samples be relevant when the fusion process occurs, will the sample be applicable for different liveness detection techniques after fusion e.g. if fusion occurs before liveness detection occurs then the ability to request additional data or samples may be impractical or impossible. It is here that self-configuration and optimization would be of vital importance as the system would be able to adapt to the different indicators and change the technique or gathering style of the liveness detection environment, thus offering a more robust and a more highly autonomic system.

6.3 Coercion Detection Characteristics

By incorporating liveness detection and autonomic aspects, a number of potential concepts can be viewed and whilst some may be viable within the inclusion of autonomic elements, they would not be as robust or inclusive. One such area is that of coercion detection. The main focus of biometric devices and systems, is to make sure that a user attempting to access the system is the correct person, this is done by comparing the provided sample to previous control sample or template, the main security concern being other users being able to access the system instead of the designated user. The next problem was the inclusion of gummy or spoof samples, which allowed access to systems using fabricated spoofs such as gelatin fingerprints, videos of users, pictures etc. This is where liveness detection became important, as it became increasingly vital to enable system to understand if the sample being provided was a human or simply a spoof sample. This being combined with autonomic elements enables a more robust, secure, and scalable systems to be developed. The next stage is to detect if a user, that has a legitimate access right to a system, is authenticating under duress. In this scenario a third party is forcing the user to authenticate using some form of coercion, this coercion can take many forms but leads to the same problem, a user is knowingly accessing a system with intend to allow a third party access when authentication has occurred. Thus a method to detect and stop this form of input is vital, there are a number of potential methods in which this could occur, detecting chemical, behavioral etc. characteristics of a user that denotes fear etc., however how accurate could this be, how easily can it be confused with other signals and scenarios. All of which must be considered if the full integration of gestalt systems is to become more widespread.

7 Conclusion

In conclusion it is apparent that there are a wide range of potential areas that liveness detection techniques can be both incorporated and that can improve within an autonomous system framework. Other aspects of the human nervous system paradigm may work just as well as those shown here, if not better. There are a number of areas that liveness detection techniques could be implemented into the autonomous architecture shown above, which would enable a pseudo autonomous set of functions to be incorporated. Overall the concepts identified within this article have potential and more research will help identify how best to move forward.

References

1. International Biometric Group, How is 'Biometrics' Defined? (2007), http://www.biometricgroup.com/reports/public/reports/biometric_definition.html (accessed May 15, 2009)

2. Abernathy, W., Tien, L.: Biometrics: Who's Watching You? (2003), http://www.eff.org/wp/biometrics-whos-watching-you (accessed April 28, 2009)
3. Roberts, C.: Biometric attack vectors and defences. Science Direct Computers & Security 26, 14–25 (2007)
4. Clarke, N., Furnell, S.: Biometrics – The promise versus the practice. Computer Fraud & Security 9, 12–16 (2005)
5. Kurzweil, R.: The Law of Accelerating Returns (March 7, 2001), http://www.kurzweilai.net/the-law-of-accelerating-returns(accessed March 21, 2013)
6. Horn, P.: Autonomic Computing: IBM's Perspective on the State of Information Technology, IBM (2001)
7. Jain, A., Nandakumar, K., Nag, A.: Biometric Template Security. EURASIP Journal on Advances in Signal Pricessing (2007)
8. Ratha, N.K., Connell, J., Bolle, R.: Enhancing security and privacy in biometrics-based authentication systems. IBM Systems Journal 3 (2001)
9. Wayman, J.L.: Technical testing and evaluation of biometric devices. In: Biometrics – Personal Identification in Networked Society, pp. 345–368. Springer, US (2005)
10. Bartlow, N., Cukic, B.: The vulnerabilities of biometric systems – an integrated look and old and new ideas. In: Biometric Consortium Conference (2005a)
11. Gordon, L., Loeb, M., Lucyshyn, W., Richardson, R.: Tenth Annual CSI/FBI Computer Crime and Security Survey (2005)
12. Woodward, J., Horn, C., Gatune, J., Thomas, A.: Biometrics: A Look at Facial Recognition, Santa Monica, CA (2003)
13. Nabti, M., Bouridane, A.: An effective and fast iris recognition system based on a combined multiscale feature extraction technique. Pattern Recognition 41(1), 868–879 (2008)
14. Markowitz, J.: The Many Roles of Speaker Classification in Speaker Verification and Identification. In: Müller, C. (ed.) Speaker Classification 2007. LNCS (LNAI), vol. 4343, pp. 218–225. Springer, Heidelberg (2007)
15. Adolph, M.: Biometric and Standards, International Telecommunication Union (2009)
16. Furnell, S., Evangelatosa, K.: Public awareness and perceptions of biometrics next term. Computer Fraud & Security 1, 8–13 (2007)
17. Matsumoto, T., Matsumoto, H., Yamada, K., Hoshino, S.: Optical security and counterfeit deterrence techniques IV. In: Proceedings of SPIE, Japan, vol. 4677 (2002)
18. van der Putte, T., Keuning, J.: Biometrical Fingerprint Recognition Don't Get Your Fingers Burned. In: Fourth Working Conference on Smart Card Research and Advanced Applications (2000)
19. Clarkson University, Clarkson University Engineer Outwits High-Tech Fingerprint Fraud (2005), http://www.sciencedaily.com/releases/2005/12/051216193022.htm (accessed July 15, 2009)
20. Kung, S., Mak, M., Lin, S.: Biometric Authentication: A Machine Learning Approach. Prentice Hall Information and System Sciences Series (2004)
21. Toth, B.: Biometric Liveness Detection. Infomation Security Bulletin 10, 291–297 (2005)
22. Schuckers, S.A.C.: Spoofing and anti-spoofing measures. Infomation Security Technical Report 7(4), 56–62 (2002)
23. Drahansky, M.: Liveness Detection in Biometrics. In: Advanced Biometric Technologies (2011)
24. Jain, A.K.: An Introduction to Biometric Recognition. IEEE Transactions on Circuits and Systems fir Vudei Technology 14(1) (2004)

25. Biel, L., Pettersson, O., Philipson, L., Wide, P.: ECG analysis: A new approach in human identification. IEEE Trans. Instrum. 50(3), 808–812 (2001)
26. Qureshi, M.K.: Liveness detcion of biomtric traits. International Journal of Information Technology and Knowledge Management 4(1), 293–295 (2011)
27. Johnson, P., Lazarick, R., Marasco, E., Newton, E., Ross, A., Schuckers, S.: Biometric Liveness Detection: Framework and Metrics. In: International Biometric Performance Conference (IBPC) (2012)
28. Hariri, S., Khargharia, B., Chen, H., Yang, J., Zhang, Y., Parashar, M., Liu, H.: The Autonomic Computing Paradigm. Cluster Computing 9, 5–17 (2006)
29. Sterritt, R.: Autonomic computing. Innovations in Systems and Software Engineering 1(1), 79–88 (2005)
30. Parashar, M., Hariri, S.: Autonomic Computing: An Overview. In: Banâtre, J.-P., Fradet, P., Giavitto, J.-L., Michel, O. (eds.) UPP 2004. LNCS, vol. 3566, pp. 257–269. Springer, Heidelberg (2005)
31. Kephart, J., Chess, D.: The vision of autonomic computing. IEEE Computer 6, 41–50 (2003)
32. Agarwal, M., Bhat, V., Li, Z., Liu, H., Khargha, B., Khargharia, B., Matossian, V., Putty, V., Schmidt, C., Zhang, G., Hariri, S., Parashar, M.: AutoMate: Enabling Autonomic Applications on the Grid. In: Proceedings of the Autonomic Computing Workshop, 5th Annual International Active Middleware Services Workshop (AMS 2003), Seattle, WA, USA (2003)
33. Hinchey, M., Sterritt, R.: 99% (Biological) Inspiration... In: Proceedings of the Fourth IEEE International Workshop on Engineering of Autonomic and Autonomous Systems, EASE 2007, Washington, DC, USA (2007)
34. Tianfield, H., Unland, R.: Towards autonomic computing systems. Engineering Applications of Artificial Intelligence 17(7), 689–699 (2004)
35. Murch, R.: Autonomic Computing. I pp. 0-20:25–40. Prentice-Hall (2004)
36. White, S., et al.: An architectural approach to autonomic computing. In: Proceedings International Conference on Autonomic Computing, New York (2004)
37. Ganek, A.G., Corbi, T.A.: The dawning of the autonomic computing era. IBM Systems Journal 42(1), 5–18 (2003)
38. Nami, R.M., Sharifi, M.: A Survey of Autonomic Computing Systems. Intelligent Infomation Processing III, pp. 101–110 (2007)
39. La, H.J., Kim, S.D.: A Practical Framework of Realizing Actuators for Autonomous Fault Management in SOA. In: World Conference on Services - I (2009)
40. Bhanu, B., Govindaraju, V.: Multibiometrics for Human Identification. Cambridge University Press, New York (2011)
41. Ross, A.: Multibiometrics. In: Encylopedia of Biometrics, pp. 967–973. Springer (2012)
42. Hong, L., Jain, A.K., Pankanti, S.: Can multibiometrics improve performance. In: Proceedings of IEEE Workshop on Automatic Identification Advanced Technologies (AutoID), New Jersey (1999)
43. Mitchell, H.B.: Multi-Sensor Data Fusion: An Introduction. Springer (2007)
44. Kiski, D., et al.: Biometrics sensor fusion. In: Thomas, C. (ed.) Sensor Fusion and its Applications, pp. 395–405. Sciyo (2010)
45. Singh, S., Gyaourova, G., Pavlidis, I.: Infrared and visible image fusion for face recognition. In: SPIE Defense and Security Symposium (2004)
46. Zhou, X., Bhanu, B.: Feature fusion of face and Gait for Human Recognition at a distance in video. In: International Conference on Pattern Recognition, Hongkong (2006)

47. Nandakumar, K., Chen, Y., Dass, S.C., Jain, A.K.: Likelihood Ratio-Based Biometric Score Fusion. IEEE Transactions on Pattern Analysis and Machine Intelligence 30(2), 342–347 (2008)
48. Monwar, M., Gavrilova, M.L.: Multimodal Biometric System Using Rank-Level Fusion Approach. IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics 39(4), 867–878 (2009)
49. Prabhakar, S., Jain, A.K.: Decision-level fusion in fingerprint verification. Pattern Recognition 35, 861–874 (2002)
50. Choundhury, T., Kanaujia, A., Li, Z., Metaxas, D.: Multimodal person recognition using unconstrained audio and video. In: International Conference on Audio- and Video-Based Biometric Person Authentication, Washington, DC (1999)
51. Pan, G., Wu, Z., Sun, L.: Liveness Detection for Face Recognition. In: Delac, K., Grgic, M., Steward Bartlett, M. (eds.) Recent Advances in Face Recognition, p. 236. I-Tech, Vienna (2008)
52. Jee, H., Jung, S., Yoo, J.: Liveness detection for embedded face recognition system. World Academy of Science, Engineering and Technology 18 (2008)
53. Thalheim, L., Krissler, J.: Body Check: Biometric access protection devices and thier programs put to the test (2002)

Mobile Augmented Reality: Applications and Specific Technical Issues

Nehla Ghouaiel[1], Jean-Marc Cieutat[1], and Jean-Pierre Jessel[2]

[1] F-64210 Bidart, France
{n.ghouaiel,j.cieutat}@estia.fr
[2] F-31062 Toulouse, France
jean-pierre.jessel@irit.fr

Abstract. Although man has become sedentary over time, his wish to travel the world remains as strong as ever. The aim of this paper is to show how techniques based on imagery and Augmented Reality (AR) can prove to be of great help when discovering a new urban environment and observing the evolution of the natural environment. The study's support is naturally the Smartphone which in just a few years has become our most familiar device, which we take with us practically everywhere we go in our daily lives. In this chapter, we discuss technical issues of augmented reality. We deal especially with building recognition. Our building recognition method is based on an efficient hybrid approach, which combines the potentials of SURF features points and features lines. Our method relies on ANNS (Approximate Nearest Neighbors Search) approach, described by Muja et al. [11]. ANNS approaches are known for their speed but they are less accurate than linear algorithms. To assure an optimal trade-off between speed and accuracy, the proposed method performs a filtering step on the top of the Approximate Nearest Neighbors Search. At the last step, our method calls Hausdorff measure [15] with line models.

Keywords: Mobile Augmented Reality, Building Recognition, Machine Vision.

1 Introduction

The term augmented reality was first used in 1992 by Tom Caudell and David Mizell to name the overlaying of computerized information on the real world. Subsequently, the expression was used by Paul Milgram & Fumio Kishino in their seminal paper "Taxonomy of Mixed Reality Visual Displays" [13]. In this paper, they describe a continuum between the real world and the virtual world (nicknamed mixed reality) where augmented reality evolves close to the real world whereas augmented virtuality evolves close to the virtual world (figure1).

In 1997 Ronald Azuma developed a complementary definition which he completed in 2001 [14] and which, along with Milgram & Kishino's approach, gave two commonly admitted definitions of augmented reality. According to Azuma, an augmented reality system is one which complements the real world with (computer generated) virtual objects so they seem to coexist in the same space as the real world, which in

L. Chen, S. Kapoor, and R. Bhatia (eds.), *Intelligent Systems for Science and Information*, Studies in Computational Intelligence 542,
DOI: 10.1007/978-3-319-04702-7_8,

both cases leads him to define the features of an augmented reality system according to the following three properties:

1. “Combining real and virtual”. In the 3D real world 3D entities must also be integrated.
2. “Real time interactivity”. This namely excludes films even if the previous condition is respected.
3. “3D repositioning”. This enables virtual entities to be made to visually coincide with reality.

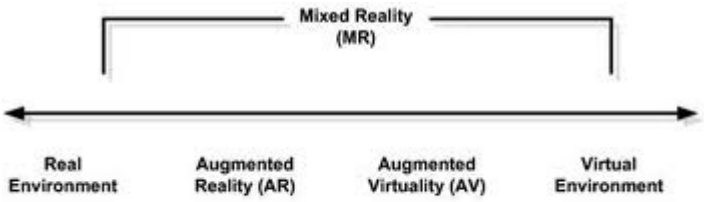


Fig. 1. Virtuality Continum

Displaying augmentations can be done with direct or indirect vision (thus inducing an additional mental load). In the case of direct vision, the display uses metaphors such as mirrors; smartphones open like windows onto the environment, vision through glasses or windows, etc.

In the first part of this chapter, we present our proper definition of augmented reality. Then, we define the general technical constitution of augmented reality systems and mobile augmented reality system. The second part of this chapter summarizes our work on mobile augmented reality. It includes a sensor-based graphic application for urban navigation and a virtual human based augmented reality application. The third part of this chapter, details our hybrid method for building recognition. It combines combines the potentials of SURF features points and features lines. Our method relies on ANNS (Approximate Nearest Neighbors Search) approach, described by Muja et al. [11]. ANNS approaches are known for their speed but they are less accurate than linear algorithms. To assure an optimal trade-off between speed and accuracy, the proposed method performs a filtering step on the top of the Approximate Nearest Neighbors Search. At the last step, our method calls Hausdorff measure [15] with line models.

2 Augmented Reality

2.1 Our Definition of Augmented Reality

In [17][6] we proposed a general definition of augmented reality as being the combination of physical spaces with digital spaces in semantically linked contexts. We can

say that augmented reality is the combination of physical spaces with digital spaces in semantically linked contexts for which the objects of associations lie in the real world. On the contrary, we can define augmented virtuality as being the combination of physical spaces with digital spaces in semantically linked contexts, but where the task's objects lie in the world of computing, states that the systems considered aim to make interaction more realistic.

All the definitions proposed in literature leave little room for multimodality. However, augmented reality has today exceeded the stage of repositioning virtual indices in a video flow and now also proposes sound and even tactile augmentations. To take into account the multimodal aspect of real world, we propose here a new definition of augmented reality: Augmented reality is the superposition of sensory data (digital or analog) to the real world, so that pursuing a definite goal; it seems to coexist with the real world. Our definition of augmented reality includes previous definitions to be more general.

2.2 Mobile Augmented Reality

Technology advances in mobile computing have promoted the development of augmented reality applications. Indeed, handheld computers are increasingly becoming smaller and lighter. They are today more accessible and cheaper thanks to highly competitive industries. Therfore, mobile augmented reality aims a wider audience than ever before, as the users own mobile devices and already know how to handle them. Hollerer et al. [4] depict basic components and infrastructure required for mobile augmented reality systems:

- Mobile Computing Platform
- Displays for Mobile AR
- Tracking and Registration
- Environmental Modeling
- Wearable Input and Interaction Technologies
- Wireless Communication and Data Storage Technologies

2.3 Technical Constitution of an Augmented Reality System

Bimber et al. [5] define general building blocks representing fundamental components of augmented reality :

Base Level: This is the most critical part of an augmented reality system. In the fact, tracking and registration problem are the most fundamental problems in AR research. Much research effort is spent to improve performance, precision, and robustness of tracking systems. In effect, precise alignment between the projected image and the features on the display surface is highly dependent on tracking.

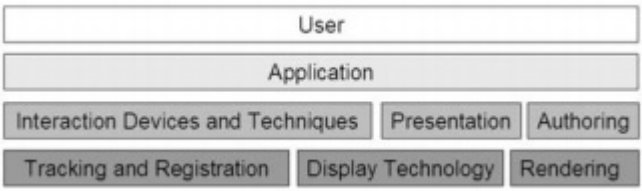


Fig. 2. Buildings Blocks for Augmented Reality [5]

Besides tracking, display technology is another basic building block for augmented reality. Head-mounted displays are the first display technology for AR applications. Today, it is possible to substitute them by Smartphone or tablet screens. The third basic element for augmented reality is real-time rendering. Since AR mainly concentrates on superimposing the real environment with graphical elements, rendering methods should operate in real time.

Second Level: This intermediate level is situated on the top of base level, as can be seen from the figure below. It includes: interaction devices and techniques, presentation, and authoring. Ideas and early implementations of presentation techniques, authoring tools, and interaction devices/techniques for AR applications are just emerging. Some of them are derived from the existing counterparts in related areas such as virtual reality.

Application Level: This level represents the interface to the user. Effectively, it is the user-oriented software part of an augmented reality system. At present, it is possible to totally implement an augmented reality application by the use of dedicated SDK.

User Level: this last layer, is finally the user of the application .User studies have to be carried out to provide measures of how effective augmented reality system is.

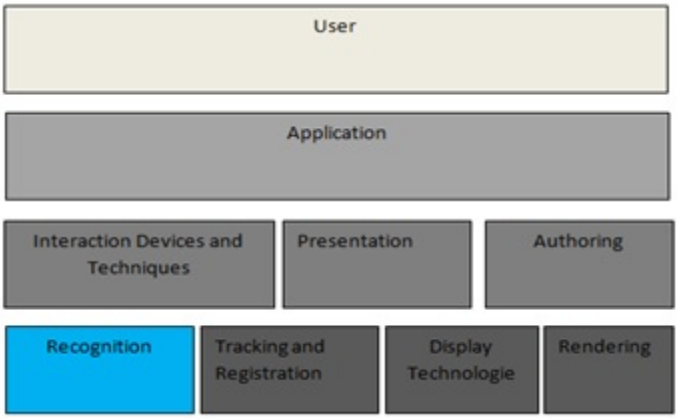


Fig. 3. Modified Technological Constitution of an AR System

We notice that Bimber et al. [5] have omitted to mention recognition in the base level of AR. Therefore, as highlighted by figure 3, we propose a modified version of building blocks of AR. Since augmentation processes treat each object differently, recognition needs to be achieved. Thus, it is primordial for an augmented reality system to identify the object in front of the camera. Hence, in the last section of this article, we deal with technical aspects of object recognition.

3 Exemples of Mobile Augmented Reality Applications

Campus information system is one of the first mobile augmented reality systems. It was proposed in 1997 by Feiner et al. [10] in their paper intitled "A Touring Machine: Prototyping 3D Mobile Augmented Reality Systems for Exploring the Urban Environment". Campus information system aims to assist users in exploring the campus space. As the user moves around the campus, his see-through head mounted display overlays notes on campus buildings, as shown in the figure below. With the emmergence of mobile devices, augmented reality systems turn from heavyweight to lightweight. In fact, thanks to embodied sensors, computing platform and camera, Smartphones or tablets could be used by themselves in a mobile augmeted reality system. In this section, we present examples of mobile augmented reality applications we developped for either indoor or outdoor environments.

3.1 Urban Environment

Augmented Reality Browser. Tourists visiting an urban environment for the first time may face a number of problems. They may, for example, not initially have a precise destination [2]. On the other hand, in any urban environment there are Points of Interest (POIs), which visitors may easily miss if these are less well known or difficult to locate. This type of POI may be described as hidden. D. McGookin [2] shows how visitors can pass by statues without actually seeing them. In this case, the first issue facing tourists confronted with unfamiliar urban environments is: What is worth visiting in the city? We believe that the most appropriate answer to this question in such situations should at least contain all the POIs (the most interesting places to visit in urban environments in this case) with highest priority ranking. Priority ranking POIs are those situated close to the visitor's position as well as those considered to be the city's symbols (this is the case of the Eiffel Tower in Paris). To distinguish common land navigation point by point (in which the destination is determined) from navigation in which the destination is not known in advance, we have chosen to call the latter multipoint navigation.

One of the aims of augmented reality is to enhance perception or the visibility of the physical world. The Smartphone's screen acts as a window onto the real world whose video flow can be augmented. We use the geo-referenced data of objects to

inform users about their location as shown in figure 4, for example, where the location information of different POIs located close by can be seen. Our system calculates the user's position based on GPS data. It then filters the database so as to only display POIs close to the user. Filtering calculates the distance between the user and the referenced objects using the Haversine formula [16]. With regard to the display, annotations are added to the real scene, which are visible on the smartphone's screen as illustrated in figure 2. For this purpose, we use the "Vision See through (VST)" technique [3], widely used in augmented reality applications. Just like the documented reality functionality relating to augmented reality, our video flow can be enriched with information identifying what can be seen with the camera. The layout of annotations informs users about the spatial location of POIs with regard to their geographical position. For example, the annotation in the top left means that the POI in question is in front of the user on the left.

Fig. 4. Visual Interface of Our Augmented Reality Browser

Virtual Human Guide. Virtual humans represent a natural way of communication. Indeed, based on a multimodal interaction mode, a virtual human guide can join gestures to speech, which remember human beings' communication. In this section, we suggest the use of virtual human guides in order to augment touring cultural visits. Educationally rich visits and visitor engagement is also one of the most important factors in the tourism industry. AR has huge potential to actively involve tourists in learning about the visited environment and exploring various museum settings and artifacts like never before.

The church of Sainte Eugenie, named after Napoleon III's wife, Empress Eugenie de Montijo, is a neo-Gothic church of gray stone that dominates the old harbor of Biarritz. To showcase the notable architecture of Sainte Eugenie's church, we integrated a virtual guide in the real scene. The figure below shows the virtual human in a didactic situation.

Fig. 5. Virtual Human in Real Scene

In another application, we overlay digital texture on top of buildings facades. The digital texture holds a virtual human animation. The virtual human highlights the history and the singularities of a particular building. Thus, he attempts to encourage the visitor to enter the point of interest and explore it.

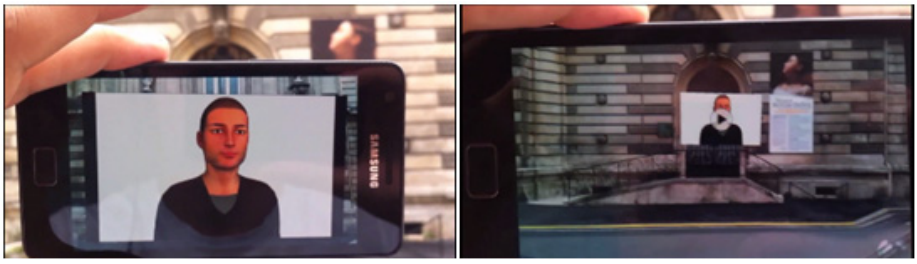


Fig. 6. Virtual Human Animation in AR Scene

4 Recognition

4.1 Features Points

The use of QR codes [12] generates visual pollution. They are also difficult to deploy in outdoor environment. Therfore, in this section we describe alternative solution to QR code which is features points. Features points are interesting points in an image. Obviously, they are rich in terms of local information contents and stable under local and global perturbations in the image domain such as illumination, brightness, and affine transformations.

Harris corner detector [7] is a well-known feature points' detector, which was proposed in 1988. Harris corner detector uses the eigenvalues of the second moment matrix to determine corner points. However, this detector suffers from scale variance.

SIFT detector introduced by Lowe [8] in 2004, is a scale-invariant detector. The relative descriptor, computes a histogram of local oriented gradients around the interest point and stores the bins in a 128 - dimensional vector (8 orientation bins for each of 4 * 4 location bins).

SURF detector and descriptor, is derived from SIFT. It was coined in 2006, by Bay et al. [9], as a novel scale and rotation-invariant detector and descriptor. It shares with SIFT the same concept of local features descriptors based on the neighbourhood of the interest point. Nevertheless, SURF differs in how the interest points are selected and described. SURF detector is based on the Hessian matrix because of its good performance in computation time and accuracy. It relies on the determinant of Hessian matrix for selecting the location and the scale of a feature point. Given a point x = (x, y) in an image I, the Hessian matrix H(x, σ) in x at scale σ is defined as follows:

$$H(x,\sigma) = \begin{bmatrix} L_{xx}(x,\sigma) & L_{xy}(x,\sigma) \\ L_{xy}(x,\sigma) & L_{yy}(x,\sigma) \end{bmatrix} \quad (1)$$

Where $L_{xx}(X,\sigma)$ is the convolution of the Gaussian second order derivative $\frac{\partial^2}{\partial x^2} g(\partial)$ with the image I in point x, and similarly for $L_{xy}(X,\sigma)$and $L_{yy}(X,\sigma)$.The extraction of SURF descriptor is performed in two steps. The first step consists of finding the orientation to a circular region around the interest point. Then, a square region aligned to the selected orientation is constructed, and therefore the SURF descriptor is extracted from it. Thanks to the use of integral images, SURF detector is faster than others points features detectors. An integral image can be rapidly computed from an input image and used to speed up the computation of the SURF descriptors for that image. The value of the integral image I(x) in a point (x, y) is the sum of all the pixel values of the input image I between the point and the origin.

$$I_{\sum}(x) = \sum_{i=0}^{i \le x} \sum_{j=0}^{j \le y} I(i,j) \quad (2)$$

The integral image enables fast computation of the intensities over any upright rectangular area of the image. This process is independent of the size of the image or of the area. The extraction of SURF descriptor consists of two steps. The first step fixes a reproducible orientation based on information from a circular region around the interest point. For that purpose, Haar-wavelet responses are computed in x and y direction, and this in a circular neighbourhood of radius 6s around the interest point, with s the scale at which the interest point was detected. The second step constructs a square region aligned to the selected orientation, and extracts the SURF descriptor from it.

4.2 Matching Approach

The task of finding correspondences between two images of the same scene or object is part of many computer vision applications such as object recognition. Once visual features have been extracted from an image, they are matched against a set of features

extracted from the other image. Feature descriptors contain a vector of real numbers. The simplest way to compare two features is to compute the Euclidean distance (or the squared Euclidean distance) between their associated descriptors. This computation is obviously slower if the dimension is higher, so descriptors with smaller vector (like the 64-dimensional SURF) are preferable over larger ones (like the 128-dimensional SURF). The distance between two descriptor vectors p and q is evaluated using Euclidean metric:

$$dist(p,q) = \sqrt{\sum_{i=1}^{64}(p_i - q_i)^2} \quad (3)$$

Linear search for nearest neighbor is costly for real-time applications. Hence, many methods are interested on approximate nearest neighbor search. Our approximate nearest neighbor search is based on the FLANN (Fast Library for Approximate Nearest Neighbors) library proposed by Muja et al [11]. FLANN contains a collection of algorithms for solving approximate nearest neighbors problem. These algorithms use, among many others, the hierarchical k-means tree or multiple randomized kd-trees. Their library automatically selects the optimal algorithm performing the best approximate nearest neighbor searches for a given dataset.

4.3 Recognition Algorithm

Approximate nearest neighbor search [11] is faster than linear search. However, it generates loss in accuracy. Obviously, it does sometimes not return optimal neighbors. This figure shows false matches generated by FLANN.

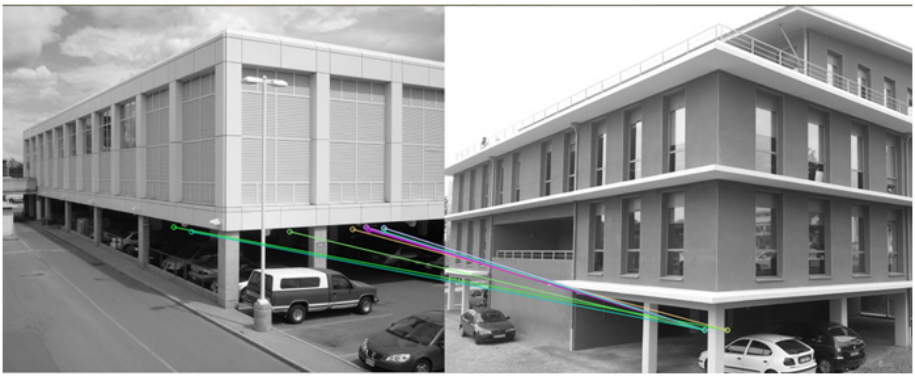


Fig. 7. False Matches with FLANN

To overcome this problem, we propose a filtering method on top of approximate nearest neighbor algorithm. Indeed, our recognition algorithm is split into two steps. The first step consists on the approximate nearest neighbor method followed by a filtering method. The second step is based on the Hausdorff distance [15] applied on line models. We note that our recognition method focuses on building recognition.

First, the test image is compared to image dataset using the approximate nearest neighbor method [11]. For each pair (test image, model image), the minimum distance between descriptors is computed. Next, the median of minimum distances is calculated. For pairs (test image, model image) that minimum distance is less than the computed median, we calculate the number of matches. At this stage, a match is considered positive if it fulfills this condition:

$$d < median \tag{4}$$

d : The relative descriptor distance

$median$: The previous computed median of minimum distances

Subsequently, we retain images giving a number of matches equal or higher than 4. This phase is called filtering.

In the second step of our recognition method, each selected model image is aligned according to the test image, using SURF correspondences. In fact, SURF correspondences are used to calculate the homography [20] relating the test and the model images. Given the 2D homography, points of the model image are transformed with respect to the test image. Next, lines segments are extracted from selected images. Line segments detection is achieved by Hough transformation [18]. Then, we carry out the clustering method proposed by Nieto et al. [19], in order to keep only orthogonal lines, which contribute to vanishing points computation. Hence, we obtain a line-based representation of building as shown in the following figure (figure 8).

Fig. 8. Line model of Building

Next, Hausdorff distance [15] is computed for each pair (test image, model image). The pair giving the smallest Hausdorff distance is considered to be the correct match.

4.4 Tests and Results

We carried out an experimental study in order to measure the performance of the proposed recognition method. In fact, we compared a test image to a dataset containing fifty images of buildings.

The figure 9 shows the obtained values of minimum distances between matched descriptors. The median of minimum distances values of this dataset is 0.089661.

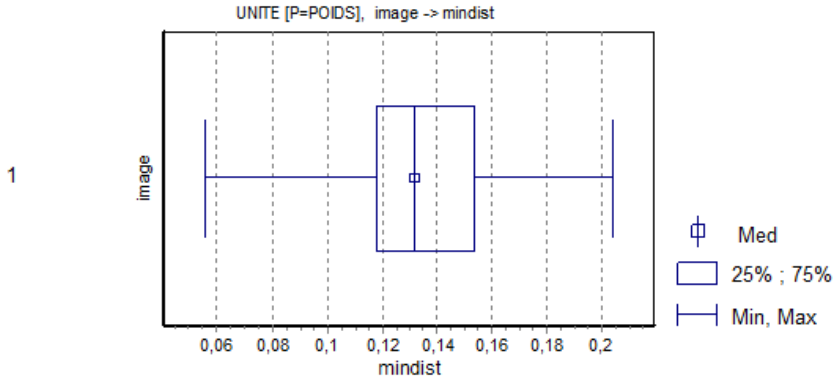


Fig. 9. Median diagram of minimum distances

Figure 10 shows the number of matches after and before filtering step, drawn respectively in blue and red. Only 22% of images participated to the second step of our algorithm. In this last step, an image is discarded if its alignment with the test image failed, otherwise, Hausdorff distance is computed. The obtained results gave that all the alignment processes failed expect the one performed with the correct match. The correct match returned a Hausdorff distance value equal to 10.098.

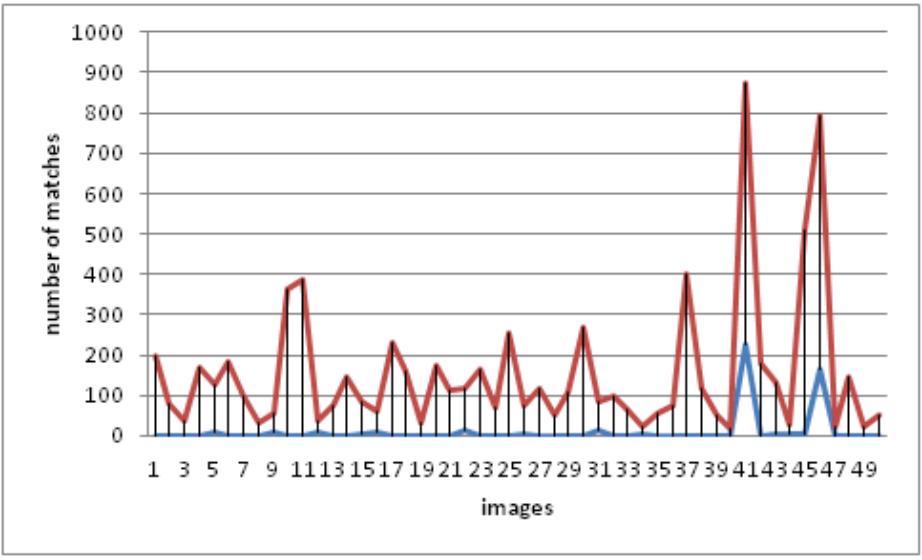


Fig. 10. Impact of Filtering Step

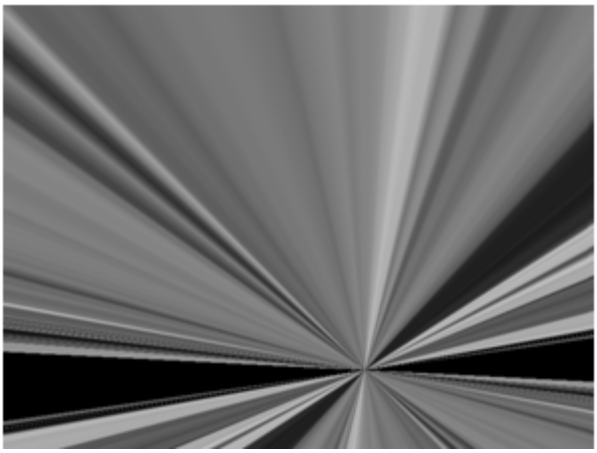


Fig. 11. Alignment Failure

Fig. 12. Successful Alignment. From left to right: test image, model image and rectified image

5 Conclusion

Augmented Reality describes the overlaying of computerized information on the real world. In this chapter, we gave our definition of augmented Reality: **Augmented reality is the superposition of sensory data (digital or analog) to the real world, so that pursuing a definite goal; it seems to coexist with the real world.** Our definition of augmented reality includes previous definitions but it is more general.

Object recognition is a primordial process in augmented reality. Thus, an augmented reality system should identify points of interest (e.g. Buildings, and artifacts) existing in the real scene, in order to apply correspondent augmentations to them. Hence, at the last section of this chapter, we depict our method for building recognition. Our proposition presents an hybrid method, which relies on both points and lines features. The obtained results show the performance of this method.

References

1. Azuma, R.: A survey of augmented reality. Presence: Teleoperators and Virtual Environments 6(4), 355–385 (1997)

2. McGookin, D., Brewster, S., Priego, P.: AudioBubbles: Employing Non-speech Audio to Support Tourist Wayfinding. In: Altinsoy, M.E., Jekosch, U., Brewster, S. (eds.) HAID 2009. LNCS, vol. 5763, pp. 41–50. Springer, Heidelberg (2009)
3. Kanbara, M., Okuma, T., Takemura, H., Yokoya, N.: A Stereoscopic Video See-through Augmented Reality System Based on Real-time Vision-based Registration. In: Proceedings of the IEEE Virtual Reality Conference, pp. 255–262. IEEE Computer Society (March 2000)
4. Hollerer, T.H., Feiner, S.K.: Mobile Augmented Reality. In: Karimi, H., Hammad, A. (eds.) Chapter Nine in book Telegeoinformatics: Location-Based Computing and Services. Taylor & Francis Books Ltd. (January 2004)
5. Bimber, O., Raskar, R.: Spatial Augmented Reality: Merging Real andVirtual Worlds. A K Peters, Ltd. (2005)
6. Cieutat, J.M., Hugues, O., Ghouaiel, N.: Active Learning based on the use of Augmented Reality Outline of Possible Applications. International Journal of Computer Applications 46(20), 31–36 (2012)
7. Harris, C., Stephens, M.: A combined corner and edge detector. In: Proceedings of the Alvey Vision Conference, pp. 147–151 (1988)
8. Lowe, D.: Distinctive image features from scale-invariant keypoints, cascade _ltering approach. IJCV 60(2), 91–110 (2004)
9. Bay, H., Ess, A., Tuytelaars, T., Van Gool, L.: Speeded-up robust features (surf). Computer Vision Image Understanding 110(3), 346–359 (2008)
10. Feiner, S., Macintyre, B., Höllerer, T.: A Touring Machine: Prototyping 3D Mobile Augmented Reality Systems for Exploring the Urban Environment (1997)
11. Muja, M., Lowe, D.G.: Fast Approximate Nearest Neighbors with Automatic Algorithm Configuration. In: VISAPP International Conference on Computer Vision Theory and Applications, pp. 331–340 (2009)
12. Interlando, J.C., Padilla, C.: Decoding the (41,20,10) Quadratic Residue Code Beyond its Error-Correcting Capability. Applied Mathematical Sciences 5(46), 2261–2269 (2011)
13. Milgram, P.: A taxonomy of Mixed Reality Visual Displays. IEICE Transactions on Information Systems E77-D(12), 1321–1329 (1994)
14. Azuma, R., Baillot, Y., Behringer, R., Feiner, S., Julier, S., MacIntyre, B.: Recent advances in augmented reality. IEEE Computer Graphics and Applications, 34–47 (November 2001)
15. Huttenlocher, D.P., Klanderman, G.A., Rucklidge, W.J.: Comparing images using the Hausdorff distance. IEEE Trans. on Pattern Analysis and Machine Intelligence 15(9), 850–863 (1993)
16. Sinnott, R.W.: Virtues of the Haversine. Sky and Telescope 68(2), 159–159 (1984)
17. Bottecchia, S., Cieutat, J.-M., Merlo, C., Jessel, J.-P.: A new AR interaction paradigm for collaborative teleassistance system: the POA. Dans: International Journal on Interactive Design and Manufacturing 3(1) (2009)
18. Hough, I.P.V.C.: A method and means for recognizing complex patterns, U.S. Patent No. 3069654 (1962)
19. Nieto, M., Salgado, L.: Real-time robust estimation of vanishing points through nonlinear optimization. In: IS&T/SPIE Int. Conf. on Real-Time Image and Video Processing. SPIE, vol. 7724, p. 772402 (2010)
20. Agarwal, A.A., Jawahar, C.V., Narayanan, P.J.: A survey of planar homography estimation techniques. Centre for Visual Information Technology, Tech. Rep. IIIT/TR/2005/12 (2005)

Violinists Playing with and without Music Notation: Investigating Hemispheric Brainwave Activity

Valerie Ross[1], Zunairah Haji Murat[2], Norlida Buniyamin[2], and Zaini Mohd-Zain[3]

[1] Faculty of Music,
[2] Faculty of Electrical Engineering,
[3] Faculty of Medicine
Universiti Teknologi MARA
vross@salam.uitm.edu.my, nbuniyamin@salam.uitm.edu.my,
zunai194@salam.uitm.edu.my

Abstract. Music has been known to improve learning and cognition. The ways in which musicians think and perform have increasingly become subjects of interest to scientists particularly in light of advances in neuroscience research. This study examines the brainwave activity of a group of violinists as they perform. Using electroencephalography (EEG), the left and right brainwaves of the musicians were recorded when they played a piece of music by first reading the score and then without reading the score. The results indicated that playing with music notation enhances left brain activity while playing without music notation enhances right brain activity. In addition, alpha brainwaves increased significantly on the right side of the brain when the violinist plays with and without score.

Keywords: left brain activity, right brain activity, music-neuroscience, violinists, and music learning strategies.

1 Introduction

Learning to play an instrument necessitates intense practice, patience, musical aptitude and perseverance. Music offers complex and multifaceted stimuli for the brain by engaging a vast network of temporal, frontal, parietal, cerebellar and limbic areas that govern auditory perception, syntactic and semantic processing, emotion and mood control, attention and memory as well as motor skills [1].

Studies of musical training in humans have shown that music has the capacity to induce long-term plasticity in the brain as indicated by changes in neurotransmitter levels, grey and white matter volumes in cortical and subcortical areas and in synaptic plasticity [2, 3] Such large scale activation and modification of the brain, especially the emotional and memory circuits in response to music, has fuelled research in the field of music neuroscience particularly in the last 20 years. Accumulating evidence from neuroimaging studies of participants using electroencephalography (EEG), magnetoencephalography (MEG), positron emission tomography (PET) and functional magnetic resonance (fMRI) suggest that the acoustic features of music triggers a

L. Chen, S. Kapoor, and R. Bhatia (eds.), *Intelligent Systems for Science and Information*, Studies in Computational Intelligence 542,
DOI: 10.1007/978-3-319-04702-7_9,

series of cognitive, emotive and motor responses in the brain at the onset of decoding acoustical information into neural signals [2-4].

Nevertheless, there is still a critical need for interdisciplinary dialogue and cross-disciplinary collaboration in the domain of music's impact on brain plasticity and human well-being particularly among music performers. While the positive effects of music listening has been well documented, less is known of the neural dynamics that occur during performances by instrumentalists playing under different conditions.

This study examines the brainwaves of nine advanced level violinists (mean age ± S.D 25 ± 5.09; 3 males and 6 females) recorded using electroencephalography (EEG) during musical performance. It illustrates the left and right brain hemispheric activity of the musicians performing while reading music notation (using visual stimuli) and performing the same piece of music without reading music notation (using recall / memory) drawing some observations from the outcomes.

2 Music Learning and Playing Strategies

The violin is one of the most studied instruments in the world. Every year violinists benchmark their playing standards through graded music examinations. In 2009 alone, 549,510 music students took the Associated Board of the Royal Schools of Music examinations of which 55,036 were violin candidates [5, 6]. These students learn repertoire from set syllabus which they generally play reading the score. Some commit pieces to memory. Thus, a study of the instrument's impact on brainwaves during performance is apt and serves to advance understanding about the functions of music in neuro processing.

Brattico et al. [7] studied the dissociation between neural processes occurring during affective vs. cognitive listening modes and judgments of music, positing the existence of distinct neural structures underlying cognitive-affective modes and judgments while listening to music. The study found that the role of the right-hemisphere for the generation of the late potentials elicited by musical sounds was predominant, regardless of the listening strategies adopted. Furthermore, the uniquely ordered structure of sensory patterns and the inherent ordered structure of music can engage, organize and alter behavior by stimulating a range of global neurolofical functions that affects the affective, cognitive and sensorimotor domains [8, 9].

Musicians possess heightened perceptual abilities, typically acquired through years of practice. As such, musicians represent a good model of how our brains adapt with experience. Theoretical models of mental representations in music performance have identified three requisite cognitive skills namely, goal imaging, motor production and self-monitoring. A musician's goal image guides performance, whether the image is built from the visual cues of printed notation or from musical information stored in memory [10-12].

3 Human Brainwave Activity

The modern era of split-brain research began in the late 1950s with pioneers such as Michael Gazzaniga and Roger Sperry who tested the functioning of each hemisphere

independently of the other in split-brain patients. They found that severing the entire corpus callosum blocks the inter-hemispheric transfer of perceptual, sensory, motor, gnostic and other forms of information in a dramatic way, paving the way for further research into hemispheric differences and the mechanisms through which the two hemispheres interact. The study of brain asymmetry examines neuroanatomical differences between the left and right sides of the brain and the lateralization of its functions [13, 14].

One way of looking at learning styles and preferences is to determine one's hemispheric dominance to ascertain whether the individual is more left or right brain dominant. While brain research confirms that both sides of the brain are involved in nearly every human activity, it is generally known that the left side of the brain is the seat of language and processes in a logical and sequential order. The right side is more visual and processes intuitively, holistically, and randomly. Generally, it is perceived that left brain personalities are more logical and scientifically inclined individuals compared to the more right-brain creative personalities such as musicians. Figure 1 provides a general view of left and right brain dominance features [15].

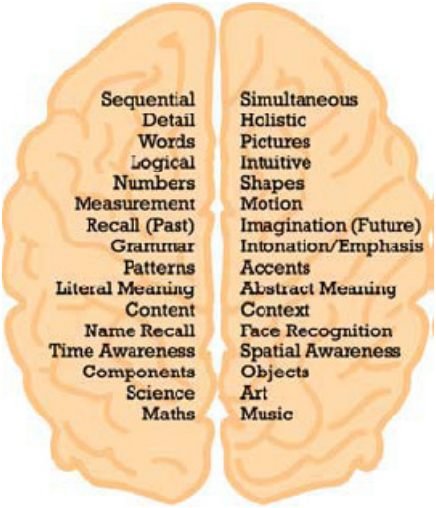


Fig. 1. Whole brain functions [15]

3.1 Brainwaves

Electrical impulses are generated by parallel-working neurons in the human brain. The synchronization of neurons enhances the potential (amplitude) of electrical oscillations while the speed of these neurons plays a role in enhancing the frequency of these oscillations. These two parameters, amplitude and frequency, act as the primary characteristic of brainwaves. The fundamental brain patterns of an individual are obtained by measuring the subject's brain signals during his/her relaxed condition. Brain patterns usually form sinusoidal waves that range from 0.5μV to 100μV peak-to-peak amplitude. During the activation of a biological neuron, this complex electrochemical

system is able to generate electrical activity, represented in terms of waves comprising of four frequency bands, namely, Delta, Alpha, Theta and Beta [16].

Previous studies have determined that among these four groups, the Beta band (13-40 Hz) has the highest frequency with the lowest amplitude while the Delta band (0.1-3 Hz) has the lowest frequency with the highest amplitude [17, 18]. The Alpha and Beta waves reflect a conscious or awake state of mind while Delta and Theta waves indicate the unconscious state. Table 1 illustrates the brainwave bands and their relation to amplitude, frequency and functions [19].

Table 1. Brainwave bands and relation to amplitude, frequency and functions

Brain-waves	Frequency (Hz)	Amplitude (μV)	Functions
Delta	0.1 – 3	Highest	Instinct : Survival, Deep Sleep, Coma, Dreaming
Theta	4 – 7	High	Emotion : Feelings, Dreams, Drowsy,
Alpha	8 – 12	Low	Consciousness : Awareness of body, Integration of feelings, Relax
Beta	13 – 40	Lowest	Concentration Thinking, Perception, Mental Activity, Alert

Beta waves are regarded as “fast brain waves” that exist when a person focuses, analyses, calculates or thinks about the external environment. An adult has significantly higher amounts of beta brain waves in comparison to a child. On the other hand, high beta brain wave activity in the right hemisphere is linked to anxiety, tension, and worry. The high Beta in the left hemisphere is considered healthy but Beta brain waves in excess are associated with disorders such as anxiety, insomnia, and obsessive-compulsive disorder. Stressful events and tension are also known to increase beta waves [16, 19].

Alpha brainwaves are associated with meditative states, visualization and idleness. Day dreaming, relaxing or closing one’s eyes (but not sleeping) increases alpha waves. Normal alpha waves are usually found to be balanced in the right and left hemispheres. Children who suffer from depression or often daydream are known to have high alpha waves. Alpha brainwaves are commonly observed in the rear parts of the brain. Excessive alpha brain waves in the left-hemisphere or frontal part may indicate a depressed state. Alpha brain waves link the conscious mind with the subconscious [16, 19].

Theta brainwaves are commonly linked to enhanced levels of creativity, emotion and spontaneity. Theta is high when one is feeling depressed, daydreams and feels distracted and anxious. High theta is also linked to fuzzy thinking, poor decision making, impulsivity, and slowed reaction time. Children generally have higher theta

waves compared to adults. This state have been known to facilitate the recovery of long-term memory and repressed emotions or improve spiritual connection [16, 19].

Delta waves are commonly associated with deep sleep patterns and represent the dominant brain-wave patterns among infants. High-amplitude rhythmic delta brain waves in adults are often found to accompany brain injury or disorders. Arrhythmic delta has been observed in college students during problem solving tasks. Delta brain waves may also be observed in the EEG of children with attention-deficit hyperactivity disorder (ADHD) accompanied by Theta. Loss of physical awareness or body awareness is accompanied by delta waves. Delta waves are observed in the EEG readings of unconscious persons [16, 19].

4 Left-Right Brain Hemisphere

The past decade has seen a rapid development of brain research. Roger Wolcott Sperry (1913-1994), an American scientist and a professor of psychobiology, reported one of the most important finding of the 1970's. Awarded the Nobel Prize in 1981 for his work on the functional specialization of the cerebral hemispheres of split brains Sperry and his students proved that "if the two hemispheres of the brain are separated by severing the Corpus Callosum (the large band of fibers that connects them), the transfer of information between the hemispheres ceases, and the coexistence in the same individual of two functionally different brains can be demonstrated" [20]. He showed that a conscious mind exists in each hemisphere. The left hemisphere is dominant in activities involving language, speech, arithmetic, and analysis whereas the right hemisphere is superior in perceiving, thinking, remembering, emoting and understanding as shown in Table 2 and Figure 2, [20-21]. Both hemispheres may be conscious simultaneously, even in mutually conflicting situations with parallel mental processing [21]. The discovery of left and right brain functions have inspired new dimensions in brain studies and music neuroscience research.

Table 2. Left and right brain capabilities

Left Hemisphere Capabilities	Right Hemisphere Capabilities
Logical and Sequential Operations	Analytical and Conceptual Operations
Communication Skills	Orientation and Awareness Skills
Comprehension & Learning Skills	Performing Complex Physical Tasks
Processing of Experiences	Technical Skill for Precise Physical actions

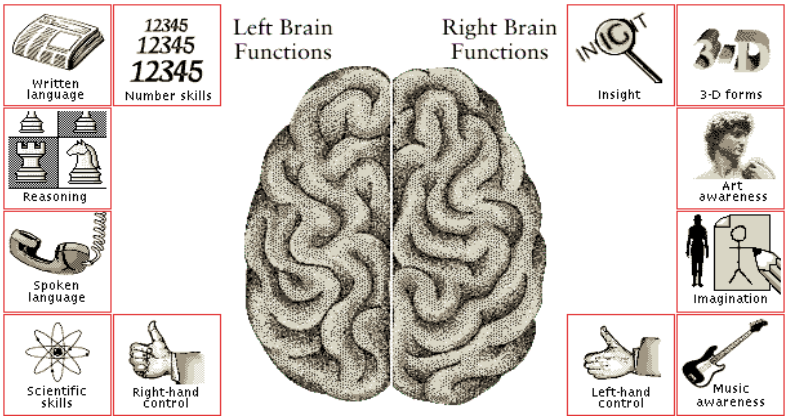


Fig. 2. Left and Right Brain Functions (Source: encarta.msn.com)

5 Electroencephalogram

EEG measurements or electroencephalograms are widely used in medical and neuroscience research. A medical imaging technique that captures electrical activity on the scalp generated by neurons, EEG records electrical activity from the scalp using metal electrodes and conductive media [16,22]. EEG measured externally and directly from the scalp surface is named electrocortiogram. When depth probes are used, it is called electrogram [16]. In this study, only EEG measured from the head surface is considered. EEG procedures are completely non-invasive and can be applied repeatedly to subjects with virtually no risks or limitations [26]. Furthermore, EEG is relatively tolerant of subject movement, unlike many other neuroimaging techniques, and there exist methods for minimizing and even eliminating movement artefacts in EEG data [23].

When a large population of neurons is activated, local current flows are produced and captured by EEG in terms of electrical potential in the time domain [24]. The weak electrical signals are massively amplified, and then displayed on paper or stored to computer memory for offline analysis. Due to its capability to reflect both the normal and abnormal electrical activity of the brain, EEG has been found to be a very powerful tool in the field of neurology and clinical neurophysiology [16, 24].

Additionally, the positioning of the electrodes is an important decision. In 1958, the International Federation in Electroencephalography and Clinical Neurophysiology adopted standardisation for electrode placement - '10-20 electrode placement' [24]. Figure 3 shows the International 10-20 systems for EEG electrodes connection [24]. The head is divided into proportional distances from prominent skull landmarks to provide adequate coverage of all regions of the brain. Label 10-20 designates proportional distance in percentage between ears and nose where points for electrodes are chosen. Placements are labeled according to adjacent brain areas: F (frontal), C (central), T(temporal), P (posterior), and O (occipital).The letters are accompanied by odd numbers at the left side of the head and with even numbers on the right side [24].

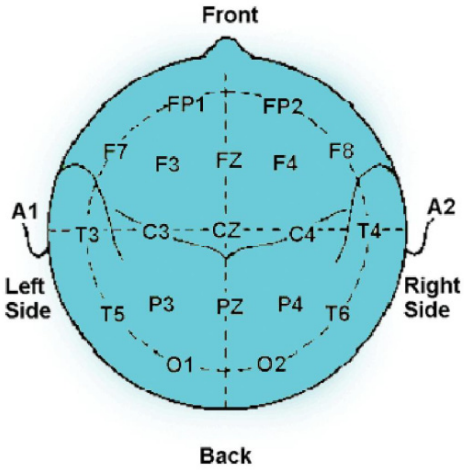


Fig. 3. International 10-20 System EEG Electrodes connections [24]

In this experiment the electrodes were placed in the frontal area and earlobes at FP1, FP2, A1 and A2. Electrodes connection of FP1 and A1 are for channel 2 which recorded the brainwaves of the left hemisphere while FP2 and A2 were for channel 1 giving the readings for the right hemisphere. The reference electrode was connected at the center between FP1 and FP2 which were in line with FZ.

5.1 Data Analysis and Equipment

Various methods have been introduced and developed to analyse EEG signals. Some of the commonly used methods are Wavelet Transform, Neural Networks and Independent Component Analysis (ICA) [24]. Each of these methods has its advantages and drawbacks. For instance, Wavelet Transform has been used in the analysis of EEG signals from epileptic seizure patients to detect or predict the onset of seizures [25-27]. Nevertheless, the wavelet transform theory is a relatively new discipline [28]. Furthermore, attempts to apply Wavelet Transform to analyse on-going EEG activity have encountered problems [26]. Artificial Neural Network (ANN) has been successfully used in the broader area covering approximation, time series prediction and modelling, system identification, pattern and sequence recognition as well as for medical diagnosis [29-30].

ANN employs the concept of interconnected groups of artificial neurons using mathematical or computational model for information processing. Researchers have successfully applied ANN to detect spike in EEG recording and in fact ANN is the best method in this application [30]. ICA is a blind source separation method used to uncover a hidden structure underneath a set of observations [31]. ICA effectively removes artefacts and separates the sources of the signals and further decomposes the remained mixed signals into subcomponents that may reflect the physiological

activity of the signals [32]. In another investigation, analysis of brainwave signal using ICA was applied to obtain the independent components and reject useless components [33].

For this experiment, the EEG data acquisition system used was the Wave Rider Pro which comes with optical isolation on all data lines. It has a sampling rate per channel of 128 Hz and FFT frequency resolution of 1 Hz, thus simplifying the collection of EEG signals and subsequent graphical output created as it is only necessary to average the amplitude signals of each category of brainwaves. In addition, the Wave Rider Pro (see Figure 4) has a dedicated channel for reading GSR and four multi-purpose independent low noise differential channels for reading signals from the brain, heart and muscles. The Analogue to Digital Resolution is 8 bit with 0.5 Hz high pass filter and 40 Hz (-72 dB at 60 Hz) of low pass filter. The common mode rejection ratio is 100dB minimum.

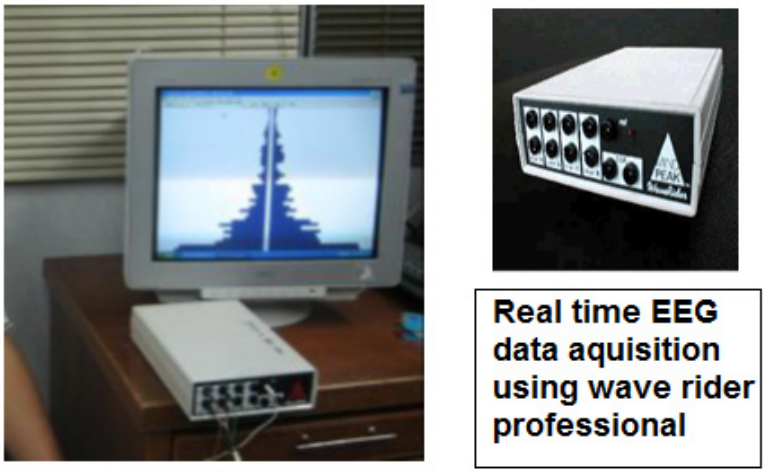


Fig. 4. The Wave Rider Pro used

6 Experiment and Procedure

This study was conducted at the Biomedical Research Lab for Human Potential, Faculty of Electrical Engineering, University Teknologi MARA (UiTM). The main objective of this experiment was to compare the brainwave patterns of violinists during two ways of performance, namely playing while reading a piece of music (visual stimuli) and playing the same piece of music by not reading music (memory / recall).

The participants in the study consisted of 9 advanced level violinists aged between 21 and 38 years (mean age ±S.D: 25±5.09; 3 males and 6 females). All the participants are first study violinists, four of them hold a Bachelor of Music qualification and five are in the final year of their undergraduate music degree studies.

Each of them played 'Rigaudon' by Fritz Kreisler for duration of 4 minutes each. The first playing was recorded with the violinist reading the score while the second playing was recorded without reading the score after a rest period of 3 minutes in between. At each performance, the piece was repeated at least once until the 4 minute duration was reached. The work was chosen for its level of technical and musical difficulty. The piece is in simple quadruple time (4/4) with mainly running semiquaver note patterns throughout the piece. The piece was played at a moderate tempo (♩ = 80-100). Figure 5 shows an excerpt of the first eight bars of the work.

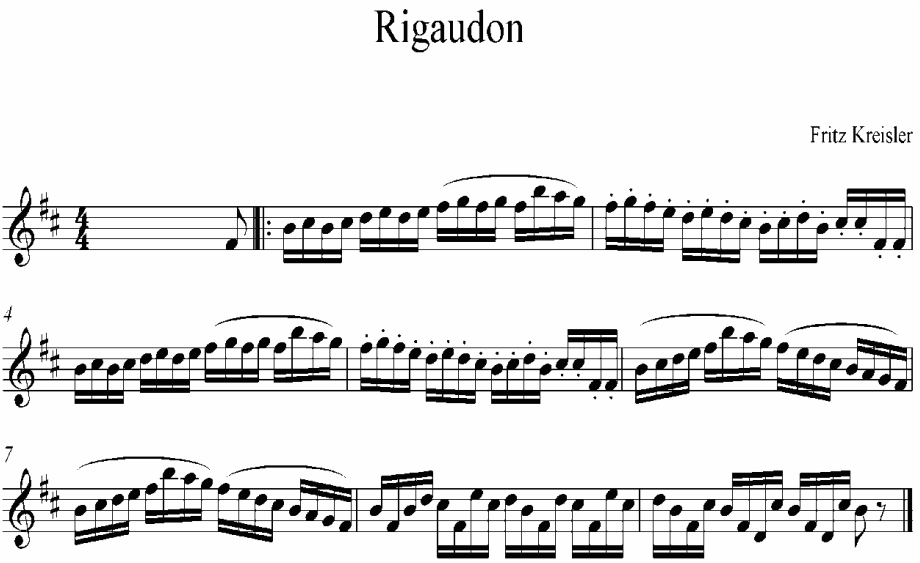


Fig. 5. Excerpt of the music played

Ethics approval to conduct the experiment was obtained from the UiTM Research Ethics Approval Committee (600-RMI 5/1/6) and written consent was obtained from each participant. The procedures used for taking the EEG recordings were as follows.

1. The subject's skin on the ears and forehead was cleaned using non-abrasive alcohol to remove dirt and dead cells for better contact with electrodes.
2. The electrodes with conductive paste were connected. Two channels of bipolar connections were connected to both ear lobes, and the left and right sides of the forehead. Channel one captured the right brainwave while channel two captured the left brainwave.
3. Before playing, the subject was directed to remain calm and relaxed for three minutes while the initial condition of the subject's brainwaves was recorded.
4. The violinist was then instructed to play for four minutes using the score while the brainwaves signals were recorded.
5. After the performance, the subject was again directed to remain calm and relaxed for three minutes to record the EEG signals.

6. The violinist was then instructed to play the same piece again for four minutes without using the score while the brainwaves signals were recorded.
7. After the performance, the subject was again directed to remain calm and relaxed for three minutes to record the EEG signals.

Figure 6 shows EEG being taken during the performance and Figure 7 illustrates an sample of the EEG brainwaves signals captured during performance using the Wave Rider (the EEG recording instrument).

Fig. 6. EEG being taken during performance

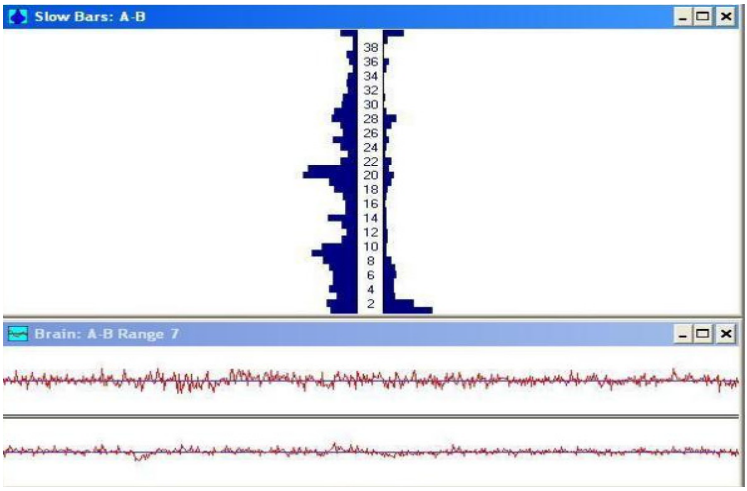


Fig. 7. Sample of EEG Signals captured using Wave Rider

The time frame protocol for the EEG experiment is shown in Table 2. The total time taken to perform the experiment was approximately 24 minutes including 14 minutes of EEG recording. The electrodes were connected throughout the experiment.

Table 3. The time frame protocol for experiment for all subjects

	Minutes				
	10	3	4	3	4
Events	Preparation	Baseline	Experiment 1	Rest	Experiment 2
	Preparations for electrodes placement. The forehead skin and earlobes of subject was cleaned with alcohol before electrodes were attached.	The subject was relaxed with eyes closed and conscious. EEG was recorded to the obtain (baseline)	In this section EEG was recorded while the subject was playing while reading the score.	At the end of experiment 1, the subject rested for 3 minutes.	In this section EEG was recorded while the subject was playing without reading the score.
Graphs		Depicted in Figure 4 (i)	Depicted in Figure 4 (ii)	Depicted in Figure 4 (iii)	Depicted in Figure 4 (iv)

7 Results and Discussion

Inspecting brain activity during a cognitive task offers an opportunity to assess the cognitive performance associated with various visualization and memory recall methods. In this instance, it captures the hemispheric brain activity of a skilled musician during the performance of a piece of music under two conditions, that is, while reading the score and without reading the score.

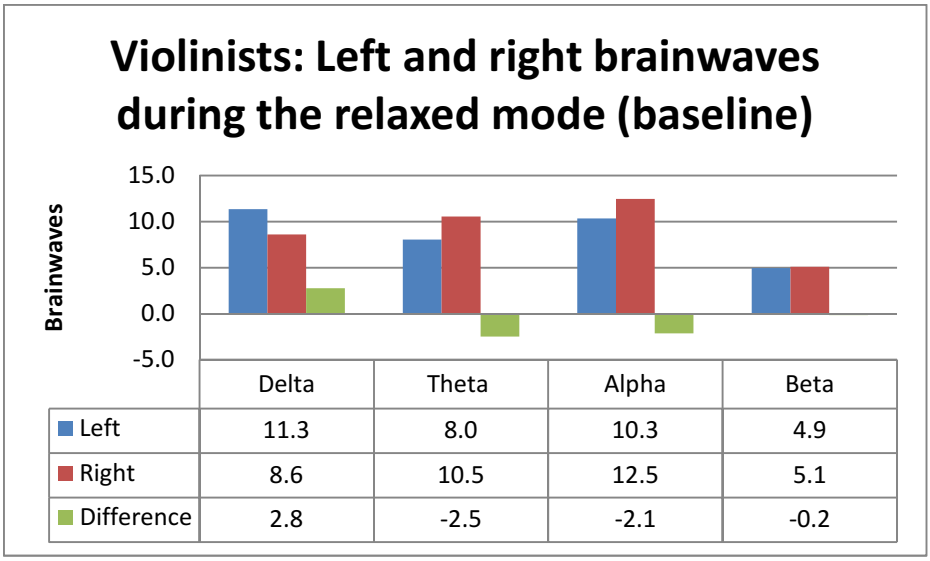
Figures 8-12 are graphs of the left and right brainwave readings under the conditions of (i) relaxed mode (baseline) (Figure 9), (ii) playing with the score (reading music notation) (Figure 10), (iii) back to relaxed mode (Figure 11), and finally (iv) playing without the score (not reading music notation) (Figure 12) . As this is a study of a small group of musicians, inherent limitations in sample size are inevitable. Nevertheless, the results do offer some significant outcomes where several observations may be drawn, namely,

1. The subjects were slightly right dominant as indicated by their baseline brain activity, the Theta, Alpha and Beta waves being higher on the right compared to the left (as depicted in Figure 8). This is generally true for individuals who are involved in creative activities which use more right brain cognition. The delta brainwave can be ignored as it is affected when subjects are in deep sleep or are unconscious.
2. Figure 12 illustrates the brainwave frequencies of each session. Playing with the score (experiment 1) indicated an increase in both sides of brainwave activity with slightly higher increment and higher level for the left side compared to the right side. Here the subject was reading, analysing and interpreting the score while playing, resulting in an increase in left side brain activity. However, as shown by Figure 9, when each of the brainwave type was analysed separately, there was a marked increase of the alpha brainwave on the right side of the brain.

3. Playing without the score (experiment 2) also increases both sides of brainwave activity. Interestingly, the right side was slightly higher than the left side as shown in Figure 12. In addition, the right alpha brainwaves increased considerably and is higher than playing with the score.
4. As shown in Figures 8-12, greater brainwave activity was recorded while playing compared to the relaxed mode suggesting that music promotes thinking and brain plasticity.

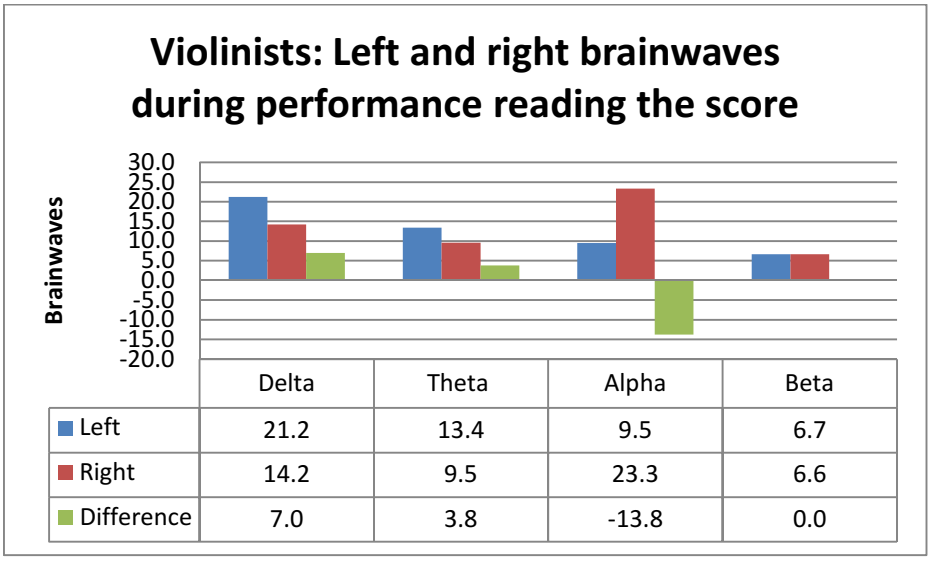
An interesting observation is that the Alpha brainwave on the right side increases considerably compared to the increment on the Alpha left for both playing with the score and playing without the score. This condition ascertains the fact that playing music could increase the activity of the right brain in particular for the Alpha wave, possibly enhancing one's creativity.

The results of the experiment indicate greater cognitive processing during musical performance supporting current research findings that music listening, learning and performance promote brain plasticity. The increase in left brain activity during performance whilst reading notation suggests that the musical score acted as a visual stimulation which necessitated immaculate reasoning and interpretation. Further research is needed in examining working memory during performance without the help of visual stimuli to better understand brain activity responsible for the retrieval, manipulation, the processing of task-related information and the emotional responses to musical stimuli [34-36]. Listening to music is also known to generate higher brain functional skills in reading, literacy and mathematical abilities [1-3, 10, 13, 37].



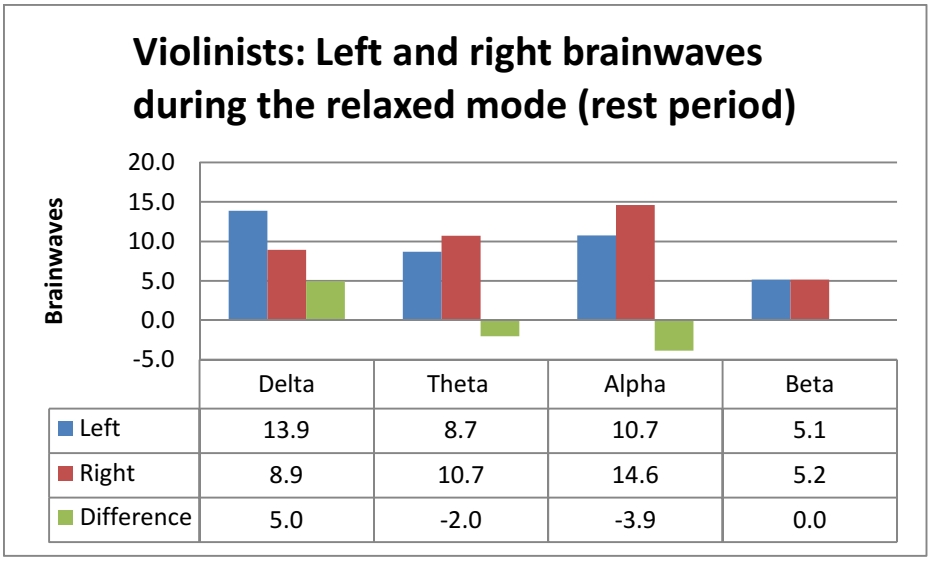
	Delta	Theta	Alpha	Beta
Left	11.3	8.0	10.3	4.9
Right	8.6	10.5	12.5	5.1
Difference	2.8	-2.5	-2.1	-0.2

Fig. 8. Baseline EEG readings of violinists



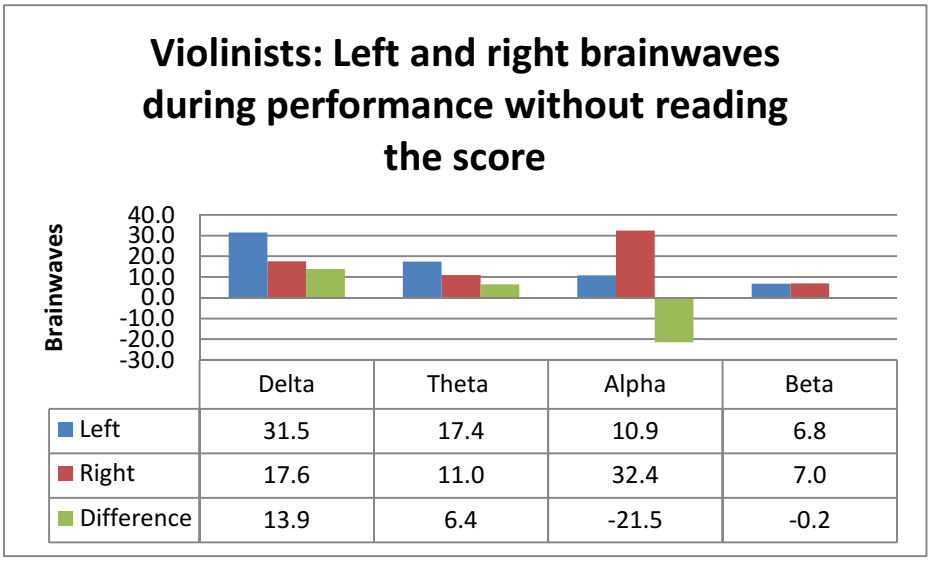
	Delta	Theta	Alpha	Beta
Left	21.2	13.4	9.5	6.7
Right	14.2	9.5	23.3	6.6
Difference	7.0	3.8	-13.8	0.0

Fig. 9. EEG readings of violinists playing with the score



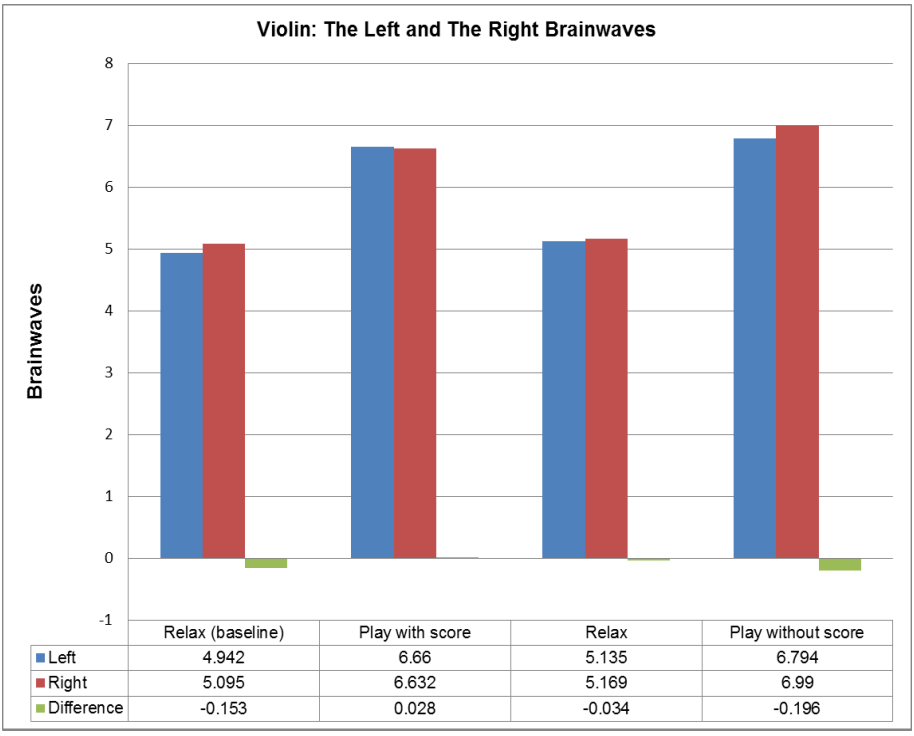
	Delta	Theta	Alpha	Beta
Left	13.9	8.7	10.7	5.1
Right	8.9	10.7	14.6	5.2
Difference	5.0	-2.0	-3.9	0.0

Fig. 10. EEG readings of violinists during the rest period



	Delta	Theta	Alpha	Beta
Left	31.5	17.4	10.9	6.8
Right	17.6	11.0	32.4	7.0
Difference	13.9	6.4	-21.5	-0.2

Fig. 11. EEG readings of violinists playing without the score



	Relax (baseline)	Play with score	Relax	Play without score
Left	4.942	6.66	5.135	6.794
Right	5.095	6.632	5.169	6.99
Difference	-0.153	0.028	-0.034	-0.196

Fig. 12. Right and Left Brainwave Amplitude for each session

8 Conclusion

In conclusion, the results point to the greater use of the left brain when the subjects played with the score, supporting the notion that reading music enhances left brain cognition and that music performance in general, promotes brain plasticity. The outcomes of this research are congruent with studies which posit that musical experience engages cognition. In this instance, music's positive effect on performers is evidenced by an increase in the amplitude of the alpha brainwave. The study supports the practice of music to enhance brain activity, foster creativity and encourages its use in rehabilitation programmes.

This investigation is part of a larger study which compares the learning approaches of musicians who are trained by reading music notation and traditional music practitioners who are trained by the oral tradition. It explores the cognitive-affective strategies adopted by skilled individuals whose complex music processing techniques intrigue those working in the dynamic domain of music science research.

Acknowledgment. We would like to thank all staff at the Biomedical Research Lab for Human Potential, Faculty of Electrical Engineering, UiTM for supporting this study and to everyone who has assisted in making this study possible. The authors also wish to thank the Ministry of Higher Education, Malaysia and Universiti Teknologi MARA for funding this research under a FRGS Grant No 600-RMI/SSP/FRGS 5/3/FSP (38/2011).

References

1. Sarkamo, T., Tervaniemi, M., Laitinen, S., et al.: Music listening enhances cognitive recovery and mood after middle cerebra(ii) l artery stroke. Brain 131(pt. 3), 866–876 (2008)
2. Gaser, C., Schlaug, G.: Brain structures differ between musicians and non musicians. J. Neurosci. 23(27), 9240–9245 (2003)
3. Hyde, K.L., Lerch, J., Norton, A., et al.: Musical training shapes strutural brain development. J. Neurosci. 29(10), 3019–3025 (2009)
4. Janke, L.: The plastic human brain. Restor. Neurol. Neurosci. 27(5), 521–538 (2009)
5. ABRSM, Statistics of Examination Candidates, Associated Board of the Royal Schools of Music (2011), http://www.abrsm.org (retrieved August 20, 2011)
6. Ross, V.: External music examiners: micro-macro tasks in quality assurance. Journal of Music Education Research 11(4), 473–484 (2009)
7. Brattico, E., Jacobsen, T., De Baene, W., Glerean, E., Tervaniemi, M.: Cognitive vs. affective listening modes and judgments of music – An ERP study. Biological Psychology 85, 393–409 (2010)
8. Peretz, Shahin, A., Roberts, L., Chau, W., Trainor, L., Millera, L.: Music training leads to the development of timbre-specific gamma band activity. NeuroImage 41, 113–122 (2008)
9. Cohen, J.D., Perlstein, W.M., Braver, T.S., Nystrom, L.E., Noll, D.C., Jonides, J., Smith, E.E.: Temporal dynamics of brain activation during a working memory task. Nature 386, 604–608 (1997)

10. Lehmann, A.C., Davidson, J.W.: Taking an acquired skills perspective on music performance. In: Colwell, R., Richardson, C. (eds.) The New Handbook of Research on Music Teaching and Learning, pp. 542–560. Oxford University Press, New York (2002)
11. Woody, R.H.: Explaining expressive performance: Component cognitive skills in an aural modeling task. Journal of Research in Music Education 51, 51–63 (2003)
12. Anderson, E.W., Potter, K.C., Matzen, L.E., Shepherd, J.F., Preston, G.A., And Silva, C.T.: A User Study of Visualization: Effectiveness Using EEG And Cognitive Load. In: Hauser, H., Pfister, H. (eds.) Eurographics / IEEE Symposium On Visualization 2011 (Eurovis 2011), vol. 30(3) (2011)
13. Hassan, H., Murat, Z.H., Ross, V., Mohd-Zain, Z., Buniyamin, N.: Enhancing Learning Using Music to Achieve a Balanced Brain. In: 3rd International Congress on Engineering Education (ICEED 2011), Kuala Lumpur, Malaysia, December 7- 8, pp. 70–74 (2011)
14. Gazzaniga, M.S.: Cerebral specialization and interhemispheric communication. Does the corpus callosum enable the human condition? A Journal of Neurology 123, 1293–1326 (2000)
15. Gregory, J.: Brain Warmup Exercises forAuthor Creativity (2009), http://publishingacademy.com/authors/get-bookideas/brain-warmup-exercises-for-authorcreativity
16. Teplan, M.: Fundamentals of EEG Measurement. Measurement Science Review 2, 1–11 (2002)
17. Manjarrez, E., Vázquez, M., Flores, A.: Computing the center of mass for traveling alpha waves in the human brain. Brain Research, 239–247 (2007)
18. Will, U., Berg, E.: Brain wave synchronization and entrainment to periodic acoustic stimuli. Neuroscience Letters 424, 55–60 (2007)
19. Murat, Z.H., Taib, M.N., Hanafiah, Z.M., Lias, S., Kadir, R.S.S.A., Rahman, H.A.: Initial Investigation of Brainwave Synchronization After Five Sessions of Horizontal Rotation Intervention Using EEG. In: 5th International Colloquium on Signal Processing & Its Applications (CSPA), pp. 350–354 (2009)
20. Sperry, R.W.: Left -Brain, Right Brain. In: Saturday Review: Speech Upon Receiving the Twenty-Ninth Annual Passano Foundation Award, pp. 30–33 (1975)
21. Sperry, R.W.: Some Effects of Disconnecting the Cerebral Hemispheres. In: Division of Biology, California Institute of Technology, Pasadena, California, pp. 1–9 (1981)
22. Will, U., Berg, E.: Brainwave Synchronization and Entrainment to Periodic Acoustic Stimuli. Neuroscience Letters 424, 55–60 (2007)
23. O'Regan, S., Faul, S., Marnane, W.: Automatic detection of EEG artefacts arising from head movements. In: 2010 Annual International Conference of the IEEE Engineering in Medicine and Biology, pp. 6353–6356 (2010)
24. Sanei, S., Chambers, J.A.: EEG Signal Processing. Wiley (2007)
25. Adeli, H., Zhou, Z., et al.: Analysis of EEG Records in an Epileptic Patient Using Wavelet Transform. Journal of Neuroscience Methods 123, 69–87 (2003)
26. Durka, P.: From Wavelets to Adaptive Approximations: Time-Frequency Parametrization of EEG. BioMedical Engineering OnLine 2, 1 (2003)
27. He Sheng, L., Tong, Z., et al.: A Multistage, Multimethod Approach for Automatic Detection and Classification of Epileptiform EEG. IEEE Transactions on Biomedical Engineering 49, 1557–1566 (2002)
28. Da-Zeng, T., Ming-Hu, H.: Applications of Wavelet Transform in Medical Image Processing. presented at Machine Learning and Cybernetics (2004)

29. Faro, A., Giordano, D., et al.: Transcranial Magnetic Stimulation (TMS) to Evaluate and Classify Mental Diseases Using Neural Networks. In: Artificial Intelligence in Medicine, pp. 310–314 (2005)
30. Miller, A.S., Blott, B.H., et al.: Review of Neural Network Applications in Medical Imaging and Signal Processing. Medical and Biological Engineering and Computing 30, 449–464 (1992)
31. Van Dun, B., Wouters, J., et al.: Improving Auditory Steady-State Response Detection Using Independent Component Analysis on Multichannel EEG Data. IEEE Transaction on Biomedical Engineering 54 (2007)
32. Rajapakse, J.C., Cichocki, A., et al.: Independent Component Analysis and Beyond in Brain Imaging: EEG, MEG, fMRI, and PET. In: Proceedings of the 9th International Conference on Neural Information Processing, ICONIP 2002 (2002)
33. Lin, C.T., Chuang, S.W., et al.: EEG Effects of Motion Sickness Induced in a Dynamic Virtual Reality Environment. In: 29th Annual International Conference of the IEEE Engineering in Medicine and Biology Society, EMBS 2007, pp. 3872–3875 (2007)
34. Juslin, P., Västfjäll, D.: Emotional responses to music: The need to consider underlying mechanisms. Behavioral and Brain Sciences 31, 559–575 (2008)
35. Patterson, R.D., Uppenkamp, S., Johnsrude, I.S., Griffiths, T.D.: The processing of temporal pitch and melody information in auditory cortex. Neuron 36, 767–776 (2002)
36. Tillmann, B., Janata, P., Bharucha, J.J.: Activation of the inferior frontal cortex in musical priming. Cognitive Brain Research 16, 145–161 (2003)
37. Hassan, H., Murat, Z.H., Ross, V., Buniyamin, N.: A Preliminary Study on the Effects of Music on Human Brainwaves. In: International Conference on Automation and Information Sciences (ICCAIS 2012), Ho Chi Minh City, Vietnam, December 26-29, pp. 176–180 (2012)

A Novel Organizational Model for Real Time MAS: Towards a Formal Specification

Mohamed Amin Laouadi[1], Farid Mokhati[2], and Hassina Seridi[3]

[1] Computer Science Department, Ferhat Abbes University, Setif -1-, Algeria
Laouadiamin@yahoo.fr
[2] Computer Science Department, Larbi Ben M'hidi University, Oum El Bouaghi, Algeria LAMIS Laboratory, Tebessa University
mokhati@yahoo.fr
[3] Computer Science Department, Badji Mokhtar-Annaba University, P.O. Box 12, 23000 Annaba, Algeria LabGED Laboratory
seridi@labged.net

Abstract. In this paper we present our approach allowing the translation of Real Time Multi-Agents Systems (RT-MAS) organizational requirements described by extended AUML (Agent UML Language) diagrams into a formal specification written in Real Time Maude language (RT-Maude). In fact, the approach is an extension of our previous work [1] that consists in extending AUML diagrams (Temporal AUML organization use case diagram and Temporal AUML organization class diagram) by using stereotypes notions and meta-model organizations entities for taking into account RT-MAS specificities. Once elaborated, these different diagrams undergo a validation to assure inter-and intra model coherence. The formal and object oriented language RT-Maude, base on rewriting logic, supports formal specification and programming of concurrent systems. The main motivations of this work are: (1) formalizing the organizational requirements of RT-MAS by using RT-Maude language, and (2) integrating the validation of the coherence models, since the analysis phase.

Keywords: Organizational Requirements; Formal Framework; Agent UML; Rewriting Logic; RT-Maude.

1 Introduction

Currently, Agent Oriented Software Engineering (AOSE) is a very active research domain. For many years, MAS designers have development methodologies and modeling language without reflects the different temporal restrictions that RT-MAS may have. Indeed, it is not easy to conceptualize RT-MAS concepts using conventional agent software engineering approaches. Consequently, a critical research issue for the real time agent community has been the definition of suitable organizational model for analyzing and designing their main properties.

Moreover, even the proposed methodologies for the development of MAS as stressed in our previous work [1] and those proposed for RT-MAS development: (like

L. Chen, S. Kapoor, and R. Bhatia (eds.), *Intelligent Systems for Science and Information*, Studies in Computational Intelligence 542,
DOI: 10.1007/978-3-319-04702-7_10,

'RT-Message' [2], 'BDI-ASDP extended for real time' [3] and 'Development Method of Lichen Zhang'[4]), are inadequate. They have certainly made important responses in the development process of RT-MAS. However, the methodological aspect is not yet mastered. Indeed, none of these methodologies take into account the formalization of the organizational requirements for the future system. The quality of model analysis has an extreme importance for the remainder of the development process phases. Their formal specification and validation allow avoiding many problems that may affect the development quality.

Formalizing the organizational aspects of RT-MAS is in our opinion, an importance way for both analysis and design activities. Furthermore, the RT-MAS design requires the involvement with formal languages. Among these languages: RT-Maude [5] is probably the best known and most widely used languages for object oriented formal specification. There is currently no work applying RT-Maude to RT-MAS formal organizational specification and both to real time applications.

The present work takes place in the context of our project, whose objective is to develop a generic organizational formal framework for organization- oriented specification of RT-MAS aspects.

As first step in our project, we have presented in a precedent work [1], extensions made on AUML diagrams for describing Real-time Multi-Agent Systems organizational requirements.

However, AUML models [6] suffer of a lack of formal semantics. It may contain inconsistencies which are difficult to detect manually. Formal methods represent an interesting solution to this problem. The formal specifications will have the effect of eliminating the ambiguities in the models interpretation. For that, the combination of AUML and RT-Maude will formally validate the developed organizational model.

This work presents a systematic approach supporting the translation of RT- MAS organizational requirements represented by extending AUML diagrams [1] into a formal specification writing in RT-Maude language. This last is multi-paradigms language which combines the functional programming and object-oriented programming. Furthermore, RT-Maude is very powerful in terms of specification, validation and verification of concurrent systems, making it a good candidate for specification and validation of RT-MAS.

The aim of this approach is to translating extended AUML diagrams into a RT-Maude formal framework to integrate the formal validation of the consistency of the organizational model, since the analysis phase.

The remainder of this paper is organized as follows. Section 2 presents a general overview of similar works. In section 3 we give a survey of used diagrams. In section 4 we give a brief overview of rewriting logic and RT-Maude language. The proposed approach and the translation process are presented in Sections 5. In section 6 a discussion of our contribution is given. Finally, we give a conclusion and some future work directions in section 7.

2 Related Works

Several methods to modeling organization oriented MAS have been analyzed [1], such as AGR [7], Moise+ [8], INGENIAS [9], and Gaia [10]. They have modified the MAS construction through giving rise to new organizational concepts, but without taking into account the agent temporal behavior. For this reason we will present briefly in this section three methodologies that directly addressing the design of real-time multi-agent systems, we are interested by: the Methodology RT-Message [2], the extended BDI-ASDP methodology for real time [3] and the development method of Lichen Zhang [4]. For a description of real-time agents, these three methodologies use different models namely: domain model, role model, and timed model (Table 1.).

Table 1. Real-Time Agent Models Identification Approaches.

	Domain Model	Role Model	Timed Model	Organizational Model
The RT- Message Methodology [2]	✓	✓	✓	very simple model
Extended BDI ASDP methodology for real time [3]	✓		✓	
Zhang development method[4]	✓		✓	

In RT-MESSAGE, an organization model is built viewing the system as a single entity. The model shows its interactions within the environment, identifying the events which it must react to. It is also necessary to estimate approximate response times for all the identified events. In our opinion, this model is abstract and still remains largely unresolved, which we believe should be described more clearly and deeply. Also, the concept of role is only present in RT-Message methodology [2], where the roles are identified independently of the agent system.

Regarding the modeling of temporal constraints of real-time agents, each methodology offers an approach: for the RT-Message case, extensions made on the different models imported from the MESSAGE method [11] [12] allow analyzing the MAS for real-time environments. For example, "the Goal / Task model" has been modified to incorporate a taxonomy of goals (Goal taxonomy) which takes into account temporal criteria. When specifying the goal's different types, it is necessary to extend the goal and task patterns of the method "message" for integrating the real-time features. The artifacts obtained are a set of 'implications diagrams' showing the relationship between goals and tasks. Subsequent, in extended BDI-ASDP for real time, proposed by Melián et al. [3], the temporal constraints modeling is done through "the timing diagrams" specified in UML 2.0. To satisfy the need to model real-time systems by the agent approach, Zhang [4] proposed to extend UML by introducing a new stereotype, called <<agents>>. The timing characteristics are specified as an instance of this stereotype called <<TimeAspect>>.

However, the RT-MAS modeling is frequently linked to organizational specifications in the sense that those specifications provide a basis for describing the organizational requirements of agents, applying a set of software engineering techniques. This concept has been neglected in these analyzed methodologies, no one of them takes in to account the temporal dimension jointly with organizational concepts (Table 1).

Although these methodologies brought much important answers in the development process and in particular for describing RT-MAS requirements, they offer only informal or semi-formal descriptions for representing RT-MAS organizational requirements. Our approach offers a joint representation of the organizational requirements while profiting from the advantages of the semi-formal and the formal approaches. Furthermore, the proposed formal approach, allows reducing confusion and misunderstanding risks between developers and users.

3 Used Diagrams

3.1 Temporal AUML Organization Use Case Diagrams

In [1], we have adopted some extensions to conventional use case diagrams (Agent-O Use Case; Temporal Agent-O Use Case; Agent-O; External Agent-O; and Real Time Agent-O) by using the extension mechanism of "stereotyping", to describe roles responsibilities, functions and behaviors in an organization.

3.2 Temporal AUML Organization Class Diagrams

This type of diagrams allows us to reflect the static relationships among roles with taking into account temporal restriction of real time agent in organizations. Our idea was to added new compartments (Temporal Organization; Groups; Social norms; Goal; and Task) to the classical AUML class diagrams based on stereotype concept [1].

3.3 Modeling Dynamic Behaviors of Agent Organizations

The proposed model, describes the desired behavior of the agent organization and its general structure (roles, interactions, and temporal constraints) by means of AUML state-chart and protocol diagrams that describes agents' individual and collective behaviors respectively.

In what follows we briefly present Real-Time Maude language and the basic concepts related to the proposed translation process.

4 Real Time Maude

Maude is a language for specifying and programming systems. It is based on rewriting logic [13] [14] [15]. Maude allows describing easily the intra and inter-object concurrency. In rewriting logic, the logic formulas are called rewriting rules. They have the

following forms: R: [t] => [t'] or R: [t] => [t'] if C. Rule R indicates that term t becomes (is transformed into) t'. On its second form, a rule could not be executed except that a certain condition C if verified. Term t represents a partial state of a global state S of the described system. The modification of the global state S of the system to another state S' is realized by the parallel rewriting of one or more terms that express the partial states. The distributed state of a concurrent system is represented as a term whose sub-terms represent the different components of the concurrent state [16].

Real-Time Maude [5] is an extension of Maude that was designed to exploit the concepts of the real-time rewrite theory. A real-time rewrite theory is a Maude rewrite theory, which also contains the specification of:

sort Time to describe the time domain,

sort GlobalSystem with a constructor '{_}": {_} : System -> GlobalSystem

And a set of tick rules that model the elapsed time in the system that have the following form: {t} => {t'} in time if condition μ

Where μ is a term which may contain variables, of sort Time that denotes the length of the rule, and the terms t and t' are terms of sort System, which denotes the state of the system. The rewriting rules that are not tick rules are rules supposed to take a time instant zero. The initial state must always have the form {t"}, where t" is a term of sort System, so that the form of tick rules ensures that time flows uniformly in all parts of the system. Real-time rewrite theories are specified in Maude as timed modules or timed object-oriented modules.

5 Translation Process

In this section, we present our approach that allows obtaining a RT-Maude formal specification.

The proposed translation process aims to translate the extended AUML diagrams described above (Section 3) to describing RT-MAS organizational requirements to RT-Maude formal specifications. This process is divided into three major steps (Fig. 1): (1) description of RT-MAS organizational requirements using AUML diagrams, (2) inter-diagrams validation, and (3) generation of RT-Maude formal specification.

The first step is the usual analysis phase of software development process. The second step aims to validate the coherence between the designed models. The last step is the systematic generation of RT-Maude source code from the considered AUML diagrams.

The formal framework proposed (Fig. 2) is composed of several RT-Maude modules: functional modules, object-oriented modules, and timed object-oriented modules. For reason of limitation of space, we present only the main modules of the proposed formal framework. We use the suffix "O" to indicate "organization".

The Module GOAL describes agents' organization goals; the TASK module describes the tasks that an agent-O can performs. These two last modules and the GROUP module (which is used to define the Agent-O groups) are imported into ORGANIZATION module to define the organizations related to agent's goals.

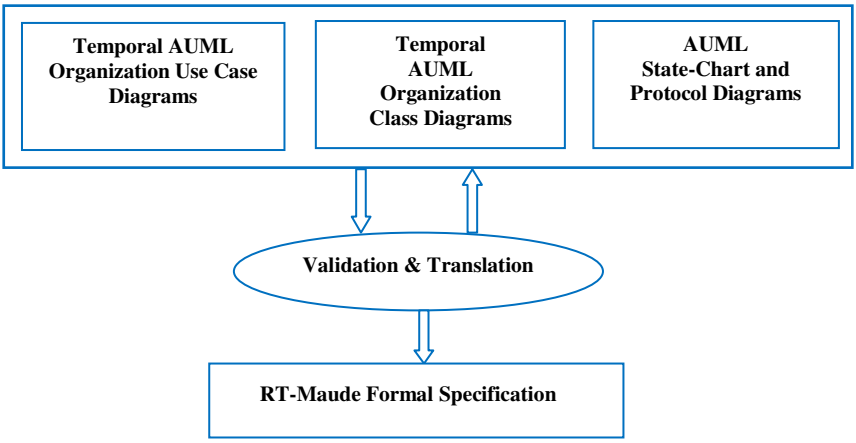


Fig. 1. Overview of the Translation Process

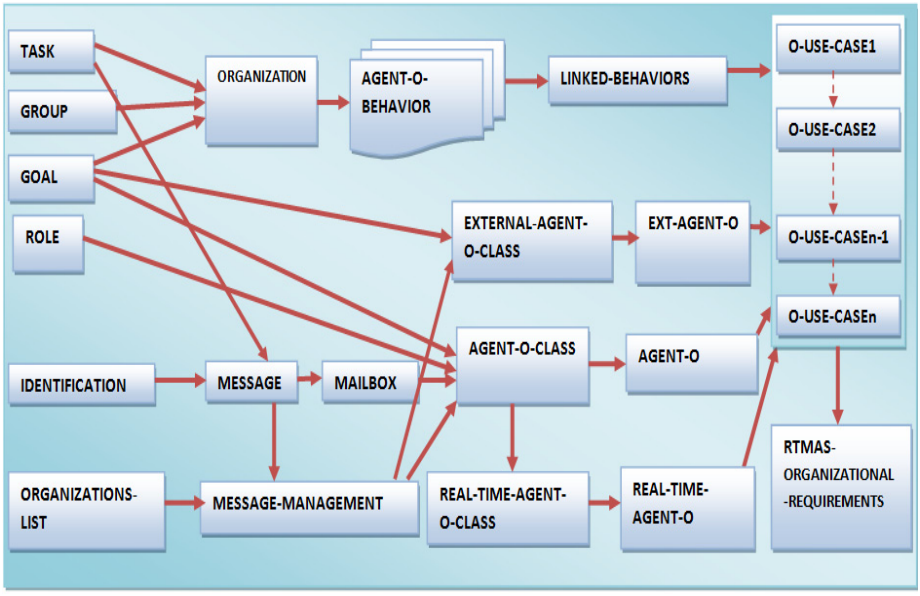


Fig. 2. Generated Modules

AGENTi-O-BEHAVIOR modules that import ORGANIZATION module are used to illustrate the behavior of individual Agents-O. In order to respect interactions between the different Agents-O, connections between them are performed through the LINKED-BEHAVIOR module, which reuses AGENTi-O-BEHAVIOR modules. The identification mechanism for Agents-O is defined by the IDENTIFICATION module, and message structure description exchanged between the various Agents-O is done via MESSAGE module that imports the IDENTIFICATION, and TASK modules.

Communicating agents are generally endowed with a Mailbox containing the received messages of other Agents-O and a list of its organizations. For that, we define the functional modules MAILBOX and ORGANIZATIONS-LIST to manage respectively Mailboxes and organizations lists of agents. Agents-O roles are defined in the module ROLE. To describe the sending/ receiving operations, we define module MESSAGE-MANAGEMENT which imports ORGANIZATIONS-LIST and MESSAGE modules.

The object oriented module EXTERNAL-AGENT-O-CLASS (Fig. 3) is used to define the base class of external agents organization, with attributes *CurrentGoal* and *OrgList* (line [1]) which represent the agent's current goal and its organizations list. This module imports GOAL, and MESSAGE-MANAGEMENT modules.

```
(omod EXTERNAL-AGENT-O-CLASS  is
protecting  GOAL . protecting  MESSAGE-MANAGEMENT .
class ExtAgentO | CurrentGoal : Goal, OrgList :
OrganizationList .        ---[1]   endom)
```

Fig. 3. The O.O Module EXTERNAL-AGENT-O-CLASS

In the object oriented module AGENT-O-CLASS (Fig. 4), we define the internal agents' organization base class structure. This class (line [1]) has as attributes: *PlayRole*, *CurrentGoal*, *MBox* and *OrgList* to contain in this order: the role played by the agent, its current goal, its mailbox and its organizations list. This module imports all the modules: GOAL, ROLE, MAILBOX, and MESSAGE-MANAGEMENT.

```
(omod AGENT-O-CLASS is protecting GOAL. protecting
ROLE.
protecting MAILBOX .  protecting MESSAGE-MANAGEMENT .
class AgentO | CurrentGoal : Goal, PlayRole : Role,
OrgList: OrganizationList, MBox : MailBox .--[1] endom)
```

Fig. 4. The O.O Module AGENT-O-CLASS

To describe the Real-Time Agents Organization, we have defined the *RealTimeAgentO* class with the attribute *Clock* (line [1]) in the timed object oriented module REAL-TIME-AGENT-O-CLASS (Fig. 5) as a subclass of *Agent-O* Class (line [2]).

```
(tomod REAL-TIME-AGENT-O-CLASS is extending AGENT-O-
CLASS .
class RealTimeAgentO | Clock : Time . ---[1]
subclass RealTimeAgentO  <  AgentO .   ---[2]  endtom)
```

Fig. 5. The Timed O.O Module REAL-TIME-AGENT-O-CLASS

To each organization use case is associated one timed O.O module O-USE-CASEi (Fig. 6), which has the same name as the corresponding organization use case. In each module O-USE-CASEi are defined the rewriting rules describing the different interaction scenarios between the agents-O defined in the different AUML Protocol diagrams,

instances of the organization use case. Note that these rules may be instantaneous rules or tick rules, conditional or unconditional.

```
(tomod O-USE-CASEi is inc EXTERNAL-AGENTS-O . inc
AGENTS-O . including REAL-TIME-AGENTS-O. including
LINKED-BEHAVIORS.
rl [1] : Configuration1 => Configuration2. …
rl [m] : Configuration 2m-1 =><Configuration2m. endtom)
```

Fig. 6. The Timed O.O Module O-USE-CASEi

Once generated, all O-USE-CASEi modules are imported in the timed object oriented module RTMAS-ORGANIZATIONAL-REQUIREMENTS (Fig. 7) which describes all system's organizational requirements.

```
(tomod  RTMAS-ORGANIZATIONAL-REQUIREMENTS  is
including O-USE-CASE1. … including O-USE-CASEm. endtom)
```

Fig. 7. The Timed OO Module RTMAS-ORGANIZATIONAL-REQUIREMENTS

The tick rule used to ensure the progress of time in the system is given in Fig. 8, where we have defined the message *Timer* to change the Real Time Agent Organization clock defined by the attribute *Clock* (line [1]).

```
crl [tick] :{Timer(TimeOut) < A : RealTimeAgentO |
CurrentGoal : G, Clock : T, PlayRole: Initiator> --[1]
REST:Configuration} => { Timer(TimeOut monus 1)
< A : RealTimeAgentO |CurrentGoal: G, Clock : T plus 1>
REST:Configuration }   in time 1 if (TimeOut > zero).
```

Fig. 8. The Tick Rule

6 Discussion

As illustrated in our previous work [1], the proposed approach considers jointly functional, static and dynamic aspects of RT-MAS from AUML extended diagrams. Among possible techniques used in the literature to validate and verify informal and/or semi-formal models is that which consists of translating these three views into Real-Time Maude formal descriptions to produce precise descriptions and also offer a better support to their verification and validation processes.

We propose that in the initial requirements phase, an organizational model is defined, detailing the main actors, roles, goals and dependencies. In this model, several organizational elements are considered (such as a simple structure, agent roles, and goals) and during the translation process, we have developed a formal framework, where several modules are generated, like illustrates in Fig. 2, to integrate the formal validation of the consistency of the organizational model, since the analysis phase.

In fact, our idea consists basically in using jointly AUML extended diagrams presented in [1], and RT-Maude formal framework together to support the formal specification of RT-MAS organizational requirements. In this formal organizational framework, the strengths of both approaches are unified, but it still lacks suitable tools for analysis and design. We are currently working on the validation of this generated RT-Maude specifications (modules), from the considered AUML extended diagrams, using a concrete case study.

7 Conclusions

The formalization of organizational requirements represents an important activity during development process of real time multi-agent systems. It produces a rigorous description and offers a solid basis for the verification and the validation activities. Several methodologies describing MAS organizational requirements are proposed.

However, they only offer informal or semi-formal descriptions. In this paper we proposed a generic approach that allows firstly, capturing functional aspect (temporal AUML organization use case diagram), static aspects (temporal AUML organization class diagram), and dynamic aspects (AUML protocol diagrams together with AUML state-chart diagrams) of RT-MAS based on organizational perspective, and secondly, translating the graphical description in a formal description RT-Maude. This later characterizes by the power of description and integrates several tools of verification and validation.

Although, this formal organizational framework represents an important approach for modeling RT-MAS, it is a recent extension which combines the AUML extended diagrams [1], with RT-Maude specification. In this framework, the strengths of both approaches are unified, but it still lacks suitable tools for analysis and design.

As future directions to this work, we plan on the development of additional formal analysis techniques including temporal analysis (using model-checking) to verify some properties of Real-Time Multi-Agent System organizational requirements specification, by integrating possibilities offered by Real-Time Maude.

References

1. Laouadi, M.A., Mokhati, F., Seridi-Bouchelaghem, H.: Towards an Organizational Model for Real Time Multi-Agent System Specification. In: Proc. SAI, October 7-9. IEEE, London (2013)
2. Julián, V., Soler, J., Moncho, M.C., Botti, V.: Real-Time Multi-Agent System Development and Implementation (2004)
3. Melián, S.F., Marsá, I., Ukrania, M., Miguel, D.-R., Carmona, A.-L.: Extending the BDI ASDP methodologie for real time (2005)
4. Zhang, L.: Development Method for Multi-Agent Real Time Systems. Faculty of Computer Science and Technology Guangdong University of Technology. International Journal of Information Technology 12(6) (2006)
5. Olveczky, P.C.: Real-Time Maude 2.3 Manual. Department of Informatics, University of Oslo (2007)

6. Odell, J., Parunak, H.V.D., Bauer, B.: Extending UML for Agents. In: Proceedings of the Agent-Oriented Information Systems Workshop at the 17th National Conference on Artificial Intelligence. ICue Publishing, Austin (2000)
7. Ferber, J., Gutknecht, O., Michel, F.: From agents to organizations: An organizational view of multi-agent systems, pp. 443–459 (2004)
8. Hubner, J.F., Sichman, J.S., Boissier, O.: A model for the structural, functional, and deontic specification of organizations in multi-agent systems. In: Bittencourt, G., Ramalho, G.L. (eds.) SBIA 2002. LNCS (LNAI), vol. 2507, pp. 118–128. Springer, Heidelberg (2002)
9. Gomez, J., Fuentes, R., Pavon, J.: The INGENIAS Methodology and Tools. Agent oriented Methodologies, pp. 236–276. Idea Publishing Group (2005)
10. Wooldridge, M., Jennings, N.R., Kinny, D.: The Gaia Methodology for Agent-Oriented Analysis and Design. Journal of Autonomous Agents and MAS 3(3), 285–312 (2000)
11. Message web site: `http://www.eurescom.de/public/projects/P900-series/p907/`
12. Message, Metamodel web site, `http://www.eurescom.de/~public-webspace/P900-series/P907/MetaModel/index.HTML`
13. Clavel, M., Duran, F., Eker, S., Lincoln, P., Marti-Oliet, N., Meseguer, J., Talcott, C.: Maude Manual (version 2.2). SRI International, Menlo Park, CA 94025, USA (2005)
14. Meseguer, J.: Rewriting as a Unified Model of Concurrency. SIGPLAN OOPS Mess. 2(2), 86–88 (1991)
15. Clavel, M., Durán, F., Eker, S., Lincoln, P., Martí-Oliet, N., Meseguer, J., Quesada, J.F.: Maude: Specification and Programming in Rewriting Logic. Theoretical Computer Science (2001)
16. Mokhati, F., Badri, M., Zerrougui, S.: A Novel Conformance Testing Technique For Agent Interaction Protocols. In: Proceedings of Science and Information Conference (SAI), London (2013)

Challenges in Baseline Detection of Arabic Script Based Languages

Saeeda Naz, Muhammad Imran Razzak, Khizar Hayat,
Muhammad Waqas Anwar, and Sahib Zar Khan

COMSATS Institute of Information Technology, Abbottabad, Pakistan
King Saud bin Abdulaziz University for Health Sciences, Riyadh, Saudi Arabia
(saeedanaz,khizarhayat)@ciit.net.pk, imranrazak@hotmail.com

Abstract. In this chapter, we present baseline detection challenges for Arabic script based languages and targeted Nastaliq and Naskh writing style. Baseline is an important step in the OCR as it directly affects the rest of the steps and increases the performance and efficiency of character segmentation and feature extraction in OCR process. Character recognition on Arabic script is relatively more difficult than Latin text due to the nature of Arabic script, which is cursive, context sensitive and different writing style. In this paper, we provide a comprehensive review of baseline detection methods for Urdu language. The aim of the chapter is to introduce the challenges during baseline detection in cursive script languages for Nastaliq and Naskh script.

Keywords: Baseline, Optical Character Recognition, Naskh, Nastaliq, Arabic script OCR.

1 Introduction

Optical Character Recognition (OCR) enables the computer system to convert the scanned image of printed/handwritten text into machine-encoded form such as ASCII/Unicode format. The objective of character recognition is to imitate the human reading ability to the machine by simulating the intelligent behavior so that it can act like human and do the similar activity the human do with text with human accuracy but with higher speed i.e. target performance is at least 5 characters per second with a 99.9% recognition rate [39]. It is a vital component in many applications; office automation, check verification, and a large variety of banking, business, postal address reading, sorting and reading hand-written and printed postal codes, and data entry applications, reading machine of blind people etc. [2], [3]. The tremendous advancement in the computational intelligence algorithms have provided significant improvement in the development of intelligent character recognition systems and it becomes one of the most successful applications of artificial intelligence, image processing and pattern recognition [4]. Optical Character Recognition (OCR) has been accelerated recently due to high demand of conversion of documents such as books, holy books, magazines, news-papers, poetry books, handwritten documents etc. into computer format text such as ASCII/Unicode, which would be manipulated

L. Chen, S. Kapoor, and R. Bhatia (eds.), *Intelligent Systems for Science and Information*,
Studies in Computational Intelligence 542,
DOI: 10.1007/978-3-319-04702-7_11,

by word processing software. OCR is a basic tool for various applications like a large variety of banking, document automation, data entry applications, business, checks verification, advanced scanning, reading machine for blind people etc. These applications can perform well if the characters from text images are classified and recognized accurately.

OCR is an active area of research and it improves human-machine interaction and communication. Various recognition methods have been proposed in past and reported high recognition rates for Latin script [7-10]. Whereas Arabic script character recognition has not received much researchers attention as thoroughly as Latin, Japanese, or Chinese scripts. This lag of research on Arabic script character recognition as compared with other scripts may be due to the complexity of this script as well as the lack of benchmark database. From the family of Arabic script base language, Arabic received significant attention whereas relatively little efforts have been done for Urdu script. Based on the similarity between Urdu and Arabic, it seems that work done for Arabic must work well for Urdu. However, it not the case in reality, Urdu writing style Nastaliq has several challenges that do not exist in Naskh. Hence, Nastaliq requires more sophisticated and very advanced feature extraction and recognition techniques as compare to Naskh. This chapter present challenge in baseline detection for Arabic script based language form both Nastaliq and Naskh writing style. Section 2 describes the properties of Arabic script followed by section 3 that describes current state of the art. Section 4 presents the challenges in baseline detection whereas section 5 presents the methods used for Nastaliq writing style so far. Future directions are discussed in section 6

2 Arabic Script Writing Systems

The Arabic writing has appeared in 6th century and it took its origins in the Phoenician script. Arabic script based languages are used by more than ¼th of world population directly or indirectly and considered 2nd to Latin as it's adopted by more than 25 languages[31-33]. Persian, Ottoman Turkish, Urdu, and other Indic languages barrowed Arabic letters and several new shapes have been invented to represent sounds that cannot be represented by existing Arabic character. This enhancement in basic Arabic character set to deal with other sounds that exist in Arabic script based languages is mostly the addition of diacritical marks on the basic shape expect few character for Urdu and other Indic languages. Thus, with respect to basic character set, Urdu may be the superset of all Arabic script based languages. Fig.1.a shows the basic character shapes of Urdu that are followed by Arabic script based languages.

The calligraphic development of Arabic script led towards several writing style i.e. Nastaliq, Naskh, Koufi, Rouqi Thuluthi and Diwani shown in Fig 1.b. Nastaliq and Naskh are the most common writing style followed by Arabic script based languages. Naskh is used by Arabic, Farsi, Pashto etc. usually [17] whereas Nastaliq is mostly followed by Urdu, Punjabi etc. There is a significant difference between these two writing style [18]. Nastaliq writing style makes Urdu scripts languages different than Arabic in appearance and introduce more challenges and complexities in the Urdu characters segmentation.

ا ب ج د ر س ص ط ع ف ق

ک گ ل م ں و ہ ء ی ے ھ

Nastaliq	أبجد هوز حطي كلمن سعفص قرشت ثخذ ضظغ
Koufi	أبجد هوز حطي كلمن سعفص قرشت ثخذ ضظغ
Thuluthi	أبجد هوز حطي كلمن سعفص قرشت ثخذ ضظغ
Diwani	أبجد هوز حطي كلمن سعفص قرشت ثخذ ضظغ
Rouq'i	أبجد هوز حطي كلمن سعفص قرشت ثخذ ضظغ
Naskh	أبجد هوز حطي كلمن سعفص قرشت ثخذ ضظغ

Fig. 1. a. Ghost Character Set b. Writing style follow by Arabic Script

Arabic script based languages are written cursively from right to left and cursive in nature. Due to cursive nature of this script, the letters are joined with its neighbor letters normally within word or sub-word. The letters shape is fully dependent on the property of joiner and non-joiner letters in case of Naskh and its position in the ligature as well in the character in case of Nastaliq writing style. Each character has two to four different shapes depend on its position in the ligature i.e. beginning of word, middle of word and end of word in a connected sequence or in isolation form. However, few letters have just two forms of final and isolated and they may join with their precedence letters but do not connect with letters which are written after them. These are called non-joiners. But, if we talk about Nastaliq writing style, it more complex than Naskh. The character shapes may be up to 32 instead of only four. In case of Nastaliq, character shapes does not depends only on the position of the character in ligature but it also depends upon the associated character on both sides as shown in figure 2.

There is also a special character Hamza with single shape only that does not join with any character and lies above or below a letter. In addition to these languages are rich in diacritical marks that appear at above, below, start or inside of the character. The diacritics are divided into common diacritics i.e. Toy, Hamza, and Madaa etc. and non-common.e. zaber, zer, pesh and shadd etc.

Some diacritical marks are compulsory whereas some diacritical marks are optional and only added to help in pronunciation. Optional diacritical marks are not often used by the native speaker i.e. Arabic and Urdu speaker who do not use the optional diacritical marks which are only added for the nonnative speaker. These diacritics and dots change the pronunciation and the meaning of the word and differentiate letters with each other of similar shape. Moreover Arabic word may comprise of more than

one ligatures and isolated letter or a single long ligature e.g., the word Pakistan has 2 ligatures and one isolated letter and Saeed (one ligature). The algorithm proposed for Arabic character recognition cannot be applied on Urdu exactly [19] whereas as work done for Urdu may also work for other Arabic script based languages.

3 State of the Art

Several baseline detection methods for Naskh and Nastaliq have been used and got more attention of researchers in the field of offline and online OCR for Arabic script based languages. Current state of the art shows that the researchers are using horizontal projection among baseline detection. The baseline is defined as a virtual line, and characters of cursive or semi cursive script's languages touch the line. Generally baseline is kept in mind during both writing and reading. Baseline detection is not only used for automatic character recognition but it is also necessary for human. Without baseline detection it is very difficult to read the text even for human due to improper visibility and error rate increase up to 10% while the context sensitive interpretation is involved in human reading. It is used in writing and in reading for vertical positioning of each character and sub-word/ligature. It is also used for the distinction of graphic representation of a symbol or character known as grapheme, as they are aligned/rest on the baseline [6].

Nastaliq is written diagonally from right to left and top to bottom, thus the baseline of Nastaliq is not straight along the horizontal line instead it is depended on the baseline following glyph and there is no single baseline for Nastaliq. Different character appears at different descender line. Due to the complexity of Nastaliq over Naskh, one character may appear at different descender line depending upon the associated characters whereas in Naskh style last character appears on the one baseline and does not depends upon it connected character. Thus baseline estimation for Nastaliq written text is more complex than Naskh style. There are several methods for baseline detection in the literature but the most known methods are following.

3.1 The Horizontal Projection Method

The horizontal projection based approach counts the elements on horizontal line and assumes that maximum number of elements on horizontal line is the baseline [7], [8], [9]. Horizontal projection provides a line of maximum number of pixels, which are used for the baseline [10], [11]. Although, it is robust and very easy to implement for Naskh but it need long straight line of text but in the case of handwritten text especially for online handwritten the length of line may be very short. Thus the histogram projection mostly failed in estimating the correct baseline for isolated handwritten text and ligatures having greater number of ascender and descender. But this method has unsatisfactory results for Nastaliq writing style due to dots and overlapping. Furthermore, it is very sensitive to the skew [12]. In [13], horizontal projection method is used with painting algorithm for tracing and detecting baseline and skew correction. Recently, Abu-Ain combined [14] directional feature of primary ligature with

horizontal profile method to estimate the baseline including dots and diacritics on Handwritten Arabic text.

3.2 Voronoi Diagram

Voronoi diagram is defined as "lines which bisect the lines between a center point and its surrounding points and these points are perpendicular of each other" and it consists of Voronoi edges, Voronoi vertex and Voronoi regions. The voronoi diagram point is measured by using edge detection, contour tracing and sampling, then use just the horizontal Voronoi edges for Cursive script language. This method is insensensitive to diacritics, dots and skew and worked well in detection of zigzag baseline [15].

3.3 Word Contour Representation

Word contour representation predicts critic points of the word contour. The points can be local minima or maxima points of the word contour. This method works well with or without diacritics, but performance of method can be affected if diacritics sizes are large relative to the main word. It is also more flexible with both machine printed text and handwritten text of different word handwriting styles.

3.4 Word Skeleton

In this method, the skeleton of the word is created by polygonal approximation and then features can extract using polygonal skeleton. This is a fast method to extract baseline relevant features and work better both for offline and online baseline detection even with diacritics and dots. The diacritics and dots do not a affect the performance of method but it needs more time than other baseline detection method, because complex calculations are performed in this method.

3.5 The Hough Transform Method

A parametric representable group of points such as a circle or straight line are identified by mapping into a parameter space in an image [16]. The parameter space consists of accumulators having n-dimensional matrix. For shape of interest, n represents number of parameters. After considering all the pixels in the given document image, votes in the accumulator are used to find the strength of evidence for a straight line or angle of writing lines skew with the corresponding parameters [17], [18]. The maximum accumulator gives the baselines for word processing. It is an expensive method due to taking time in commutation but can use on several lines at a time.

3.6 The Entropy Method

The entropy is used for measurement of the information of image. This method uses the horizontal projection of contours image on y-axis according to several inclined

axes. For each projection, the histogram density and corresponding entropy are calculated and this calculated entropy finds the orientation according to which the word is the most compact. This method also takes more calculation time [19].

3.7 Principal Components Analysis (PCA)

PCA is also common technique for baseline detection, which is used for pattern extraction in high dimensional data, skew detection and data compression. It is a mathematical method that covers covariance, standard deviation, and Eigen vectors and values. The orthogonal transformation used in this method for converting a set of observations of possibly correlated variables into a set of values of linearly uncorrelated variables, is known as principal components. Basically, this method is used to find the direction of cursive script word's base-line according to distribution of foreground or background pixels. PCA detected an angle and estimated direction using PCA Eigen vector, then rotated the image and then applied horizontal projection in [20], [21]. This method can work with dots and diacritics, but the results are better without dots and diacritics [20].

4 Challenges in Baseline Detection

Optical Character Recognition for cursive script languages like Urdu, Pashto, Jawi, Sindhi and Farsi is at infancy stage. A large number of characters and similar shape characters in Urdu scripts' alphabet set make it more complex and challenging task to the researcher [22]. Due to context sensitivity, position of character introduces complexities when they are placed at initial, middle or at the end of a word etc. [23], [24], [25]. There are many writing styles in Urdu but the most common writing styles are Nastaliq and Naskh. Nastaliq writing style of Urdu language is written diagonally with no fixed baseline, slopping, highly cursive and context sensitive due to shape, filled loops and false loops, and position of characters and overlap is present in characters and also in the ligatures [26] [28] [25]. All these complexities pose significant challenges to the technology for Nastaliq script than Naskh script. The main comparative issues between Nastaliq and Naskh are present here from baseline detection point of view:

Arabic Naskh style usually uses a horizontal baseline where most letters have on a horizontal segment of constant width, irrespective of their shape and we can easily segment these horizontal constant-width segments from a ligature in Naskh style shown in Fig 11. However, this is not the case with Nastaliq which has multiple baselines, horizontal as well as sloping shown in Fig 12, making Nastaliq a complex style for character segmentation and recognition. As Nastaliq is written diagonally from right to left and top to bottom, thus the baseline of Nastaliq is not straight along the horizontal line instead it is depended on the baseline following glyph and there is no single baseline for Nastaliq. Similarly the position of the glyph is depended on the position of the following glyph. The ligatures are tilted at 30-40 degrees approximately. Furthermore, the ascenders and descenders cause incorrect detection of the

baseline because of oblique orientation and long tail of ascenders and descenders especially in case of Nastaliq. Whereas, in case of Naskh writing style, all character appear on horizontal baseline thus it easy to find the baseline and literature showed excellent results for baseline extraction. A wide survey summarizes the different approaches [29], using the horizontal projection and peaks detection, skeleton analysis using linear regression for Arabic [30] and in for Urdu Nastaliq [31]. Razzak et al. estimated the baseline locally on primary stroke with additional knowledge of previous words for Online Urdu script [31].

Approaches based on the horizontal projection histogram can be used for Naskh writing style of Arabic, Farsi, Pashto, Urdu and Sindhi languages but they may be ill-suited for Nastaliq writing style of Urdu. Urdu uses Nastaliq script for writing. Horizontal projection histogram of pixels calculates maxi-mum number of pixels in a row which will be a candidate for the baseline in Naskh because all characters of the ligatures rest on the horizontal virtual horizontal line as shown in Fig. 1 but it may be resulted with false baseline in Nastaliq due to its diagonality nature because only last character rest on the virtual horizontal line as shown in Fig. 2.

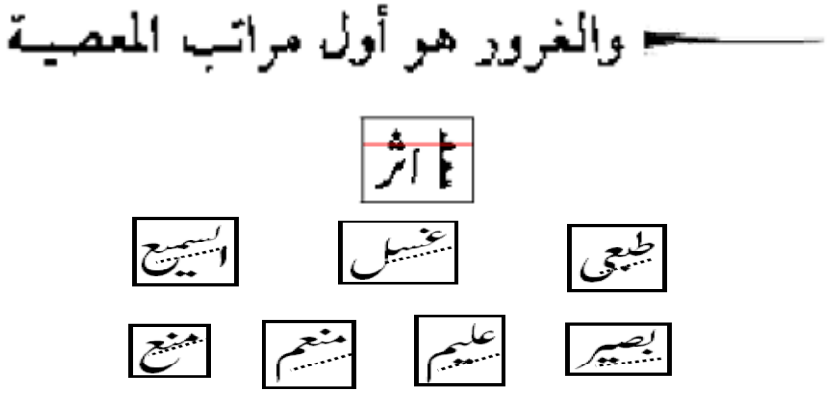


Fig. 2. a. Baseline in Naskh due to Horizontal Projection Histogram b. False Baseline in Nastaliq due to Horizontal Projection Histogram c. Multiple baseline[39]

Detection of Baseline for Nastaliq script as a straight line is still unsolved problem in the field of OCR because words composed of more than one sub-word or ligature, and ligatures distribution in the same word makes baseline detection difficult, as shown in Fig. 3.

Fig. 3. Each word is composition of more than one sub-word

There are four forms of a character in Naskh in Fig. 4 but various forms of initial, medial and isolated etc. in the Nastaliq, which increases the number of ligatures than Naskh as shown in Fig. 5 and also introduces more challenges.

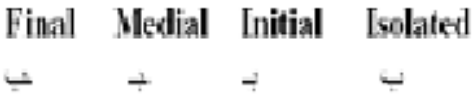


Fig. 4. One Form of Initial, Medial, Final Bey in Naskh

Fig. 5. Various Initial Forms of Bay in Nastaliq

Other issues in baseline detection are due to diacritics and dots, which may lie in the range of baseline. The diacritics may affect significantly the performance in both accuracy and time in cursive script baseline detection. Therefore, the diacritics and dots should be eliminated in the process of baseline detection but it will increase the processing time especially more in Nastaliq than Naskh. The dots are laid above or below the main horizontal line in Naskh while the dots may lie on the virtual baseline in Nastaliq case. For example dot in “chey” in Nastaliq style rest on baseline in Fig. 6.

Fig. 6. Dots Placement in Nastaliq and Naskh

The diagonality of words is also a challenging issue in Nastaliq baseline for researchers. As Nastaliq is written diagonally from right to left and top to bottom, thus the baseline of Nastaliq is not straight along the horizontal line instead it is depended on the baseline following glyph. Similarly the position of the glyph is depended on the position of the following glyph. The ligatures are tilted at 30-40 degrees approximately. Whereas, in case of Naskh writing style, all character appear on horizontal baseline, as shown in Fig. 7.

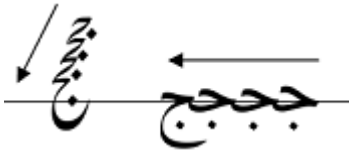


Fig. 7. Diagonality in Nastaliq and Naskh writing Style

Complexity of overlapping is present in characters and portion of connected characters (ligatures) in Urdu scripts. The characters are overlapped vertically, and they do not touch each other. The overlapping in ligature is required to avoid unnecessary white space.

There are two types of overlapping i.e. inter ligature overlapping and inter ligature overlapping [29], [30]. Intra Ligature Overlapping means that different characters within same ligature are overlap vertically and do not touch each other, as shown in Fig. 8

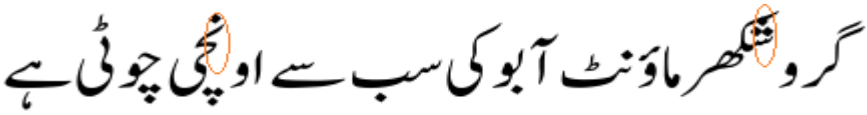


Fig. 8. Inter Ligature Overlapping

Inter Ligature Overlapping means that individual characters from different subwords are also overlap vertically and do not touch each other, as shown in Fig. 9. All these complexities make the baseline detection and recognition of individual characters quite difficult.

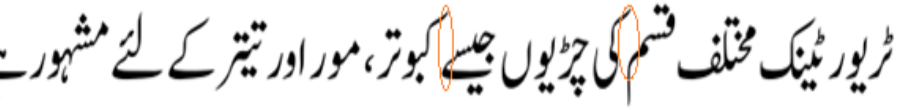


Fig. 9. Intra Ligature Overlapping

Naskh writing style uses one horizontal baseline where most letters rest on it irrespective of their shape in Naskh style in Fig. 10.

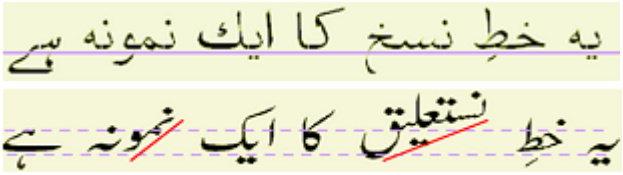


Fig. 10. a. Baseline in Naskh, b. Baseline in Nastaliq

However, this is not the case with Nastaliq which needs multiple baselines, horizontal as well as sloping, making Nastaliq a complex style for baseline detection in Fig. 11. The baseline of Nastaliq writing style is not a straight horizontal line; instead, the baseline of each character/ligature is dependent on the baseline of following character/ligature [27]. Similarly, the position and shape of a particular character is relative to the position of the character following it [28].

The text lines in printed Naskh script have large spacing between lines as compare to Nastaliq script. Moreover, in case of Naskh, text appear on one baseline whereas in case of Nastaliq is written diagonally from right-to-left and top-to-bottom. Horizontal

projection is used for line segmentation in Arabic scripts and show that there are zero valleys between-line in the projection profile as shown in Fig 8. While small interline spacing is a case of Nastaliq script (Urdu) which result with no zero valleys between line in the projection profile and this method is not working robustly as shown in Fig 9.

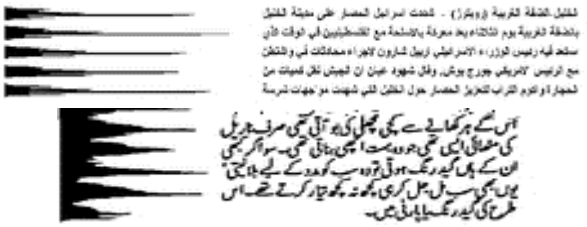


Fig. 11. A. Horizontal Profile in Naskh and Nastaliq [41]

The distance of isolated letters and all ligatures bellow the virtual baseline for Naskh and Nastaliq Urdu cannot be same, for example the letter "bari yeh" last character of the text line rest on the baseline in the Nastaliq writing style, while it lies down the baseline in Naskh script as shown in Fig. 12. This characteristic makes the baseline detection method difficult for the researchers and needs more computational time.

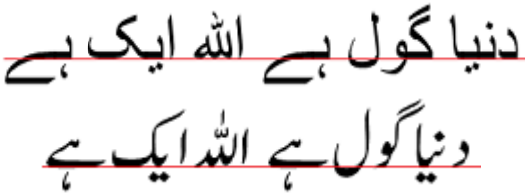


Fig. 12. The descender distance of different characters in Naskh and Nastaliq script

5 Review of Baseline Detection Methods in Urdu

Urdu baseline detection methods have not received more attention of researchers in the field of offline and online OCR. The researchers are using the most common method of horizontal projection among baseline detection methods which was published in [5]. Javed and Hussain presented improved pre-recognition process for Urdu Nastaliq OCR as a part of a large project on font 36, whose objective is to construct a complete Urdu OCR system [35]. They have achieved 100% and 94% accuracy for baseline identification and ligature identification respectively by using the horizontal projection by considering that last letter of each ligature of Nastaliq lies on a baseline, and the diacritics do not lie on, touch or cross the horizontal line. They have performed number of heuristic check for avoidance of false baseline along with projection

method. A threshold value is computed for the size of the main body versus the diacritics, as the latter are much smaller in size. All the connected bodies crossing baseline but having the size smaller than the threshold are still considered as diacritics.

In [36], existing system for Roman script in [37] studied and analyzed and applied that algorithm on Urdu text document images for baseline detection with some modification. This work was partially funded by the BMBF (German Federal Ministry of Education and Research). The Shafait et al. analyzed empty white space rectangles for pages with columns, and then binarization and noise removal by connected components has been performed. As illustrated in section 3 that Urdu characters are below from the baseline with variable distance as illustrated in Fig. 15. There-fore, they extracted baseline of Urdu text by introducing two descender lines for Urdu image by modification of single-descendent text line detection algorithm which published in [38] for Roman script named as Recognition by Adaptive Subdivision of Transformation Space (RAST)as shown in Fig. 14.

Fig. 13. Baseline Issues, a) false baseline, b) ligature Alif does not rest on base-line, c) diacritics and dots touch the baseline [35]

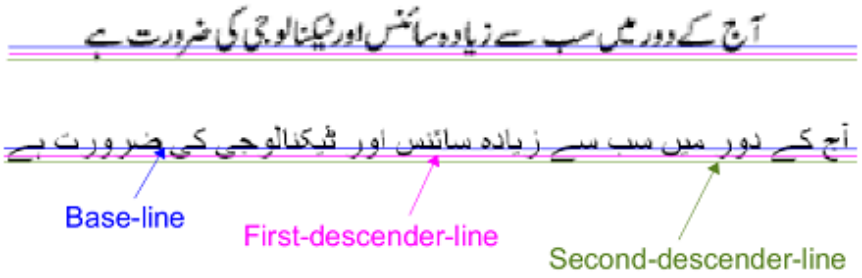


Fig. 14. Urdu baseline modeling using two descender lines [36]

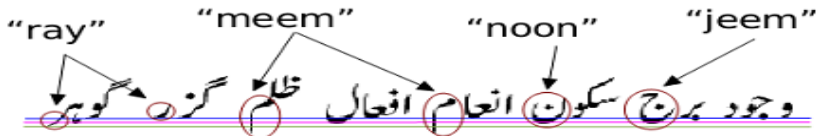


Fig. 15. the descender distance of different characters in Nastaliq writing script[36]

The RAST has three parameters that are r (distance of the baseline from the origin), (angle of the baseline from the horizontal axis), d (distance of the line of descenders from the baseline). The RAST algorithm finds globally optimal baselines using these parameters and images statistics about average width and height of a letter. But the results are not as robust as Roman's results due to dots, diacritics and cursive natures of ligature or compound letters.

The modified method is applied and tested on 25 scanned document images from different sources like books, magazines, and newspapers. The results show high text-line detection on scanned images of Urdu books and magazines with accuracy of above 90%. The algorithm also works reasonably well and achieved accuracy upto 80% on digest images due to small inter-line spacing and presence of enumerated lists. The text-line detection accuracy decreased to about 72% for newspaper documents due to many font sizes within the same image, small inter-line spacing, inverted text, and poor quality of page resulting in lot of noise.

Razzak et al. presented baseline extraction to detect Urdu online handwriting baseline for Nastaliq and Naskh script by combining two methods [27]. Horizontal projection is used for detection primary baseline after separation of secondary strokes or diacritics. Features of each ligature with additional knowledge of previous words of Urdu scripts and primary baseline used for estimation of local baseline as illustrated in Fig.16 and Fig.17. The system achieved accuracy rate 80.3% for Nastaliq and 91.7% for Naskh writing style.

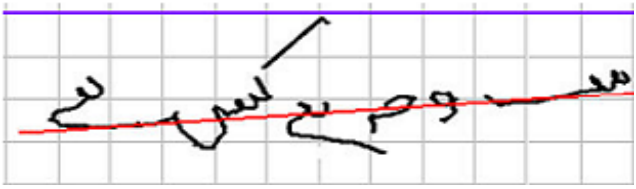


Fig. 16. Primary Baseline Estimation based on projection [27]

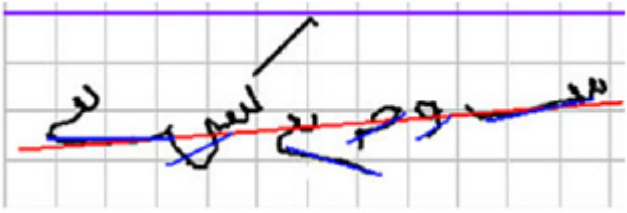


Fig. 17. Local Baseline Estimation Based on Features [27]

Razzak et al. performed analysis of skeleton using linear regression and tried to combine the information of offline and online for analyzing of spatial morphology of the Nastaliq scripts' for baseline detection [29]. Minimum enclosing rectangle and drawing vertical projection is also used in order to detect the baseline [30]. In related attempts [32], [33], fuzzy based biologically technique performed locally on baseline estimation, stroke mapping, slant correction etc. as a preprocessing steps and based on the fuzzy and context knowledge. The angle of current word computed with additional knowledge of previous word angle for detection of baseline and to-tally depended on fuzzy rules. In [34], Sardar et al. calculated primary ligatures statistics for sketching two horizontal lines on the 35% and 50% height of the main ligatures and find out an average horizontal line as shown in Fig. 18. Sabour and Safiat have also used maximum horizontal projection for baseline identication [11].

Fig. 18. Showing Base line and Average horizontal lines [33]

6 Future Direction and Conclusion

This chapter presents the challenges in baseline detection for OCR of Naskh and Nastaliq writing style as well as we presented the baseline detection methods in Nastaliq writing style. The objective of the chapter is to highlight the baseline issues. The study concluded that still, there is no perfect and robust method available for Urdu baseline detection due to multiple baselines that introduced due to diagnality nature of Nastaliq. It is clear that this area of research needs further enhancement and is open for researchers. It is concluded, that the proposed methods in the literature of Arabic, Urdu, Farsi are well with Naskh printed text, but we have to develop baseline detection methods for printed Nastaliq text and handwriting Naskh and Nastaliq text by combining two or more methods or by finding new other techniques, which should work without angle and should not be effected by the diacritics.

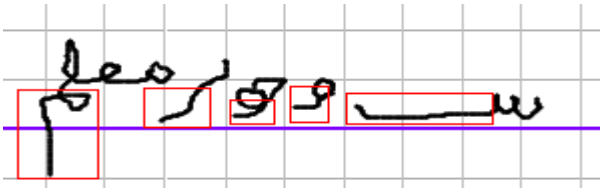


Fig. 19. Features for baseline estimation

Baseline detection can be efficiently performed on the ghost character shapes. It is better of find some features that lies on the base line for Nastaliq as presented [27]. The addition of some baseline features along with several candidate baselines would better help in finding the baseline as shown in figure 19. Moreover, local baseline based on few sub word would also provide good result especially in case of handwritten text. This baseline detection approach can be performed three run. In the first run, candidate baseline will be extracted. Then features lie on the candidate baseline will be extracted and then finally, true baseline will be extracted by fusing the feature and pre-extracted baseline knowledge. The result can also be improved by using more than one method for baseline detection.

References

1. Razzak, M.I., Mirza, A.A., et al.: Effect of ghost character theory on Arabic script based languages character recognition. Przeglad Elektrotechniczny, ISSN 0033-2097
2. Raza, A., Siddiqi, I., Abidi, A., Arif, F.: An unconstrained benchmark Urdu handwritten sentence database with automatic line segmentation. In: International Conference on Frontiers in Handwriting Recognition (2012)
3. Farooq, F., Govindaraju, V., Perrone, M.: Pre-processing methods for hand-written Arabic documents. In: Proceedings of the 2005 Eight International Conference on Document Analysis and Recognition (ICDAR 2005), pp. 267–271. IEEE (2005)
4. Al-Rashaideh, H.: Preprocessing phase for Arabic word handwritten recognition. Russian Academy of Sciences 6(1), 11–19 (2006)
5. Parhami, B., Taraghi, M.: Automatic recognition of printed farsi texts. Pattern Recognition 14, 395–403 (1981)
6. Boubaker, H., Kherallah, M., Alimi, A.M.: New algorithm of straight or curved baseline detection for short arabic handwritten writing. In: 10th International Conference on Document Analysis and Recognition, ICDAR 2009, pp. 778–782. IEEE (2009)
7. Natarajan, P., Belanger, D., Prasad, R., Kamali, M., Subramanian, K.: Baseline Dependent Percentile Features for Oine Arabic Handwriting Recognition. In: International Conference on Document Analysis and Recognition (ICDAR 2011), pp. 329–333. IEEE (2011)
8. Al-Badr, B., Mahmoud, S.A.: Survey and bibliography of Arabic optical text recognition. Signal Processing 41(1), 49–77 (1995)
9. Amin, A.: Online arabic character recognition: the state of the art. Pattern Recognition 31(5), 517–530 (1998)
10. Shah, Z.A.: Ligature based optical character recognition of urdu-nastaleeq font. In: International Multi Topic Abstracts Conference, INMIC 2002, p. 25. IEEE (2002)
11. Sabbour, N., Shafait, F.: A segmentation-free approach to arabic and urdu ocr. In: IS&T/SPIE Electronic Imaging, pp. 86580–86580. International Society for Optics and Photonics (2013)
12. Pechwitz, M., Margner, V.: Baseline estimation for arabic handwritten words. In: Proceedings of the Electrochemical Society of the Eighth International Workshop on Frontiers in Handwriting Recognition (IWFHR) Frontiers in Handwriting Recognition (IWFHR), p. 479 (2002)
13. Nagabhushan, P., Alaei, A.: Tracing and straightening the baseline in hand-written persian/arabic text-line: A new approach based on painting-technique. The Proceeding of Int. Journal on Computer Science and Engineering, 907–916 (2010)

14. Abu-Ain, T., Sheikh Abdullah, S.N.H., Bataineh, B., Omar, K., Abu-Ein, A.: A novel baseline detection method of handwritten Arabic-script documents based on sub-words. In: Noah, S.A., Abdullah, A., Arshad, H., Abu Bakar, A., Othman, Z.A., Sahran, S., Omar, N., Othman, Z. (eds.) M-CAIT 2013. CCIS, vol. 378, pp. 67–77. Springer, Heidelberg (2013)
15. AL-Shatnawi, A., Omar, K.: A comparative study between methods of Arabic baseline detection. In: International Conference on Electrical Engineering and Informatics, ICEEI 2009, vol. 1, pp. 73–77. IEEE (2009)
16. Li, Q., Xie, Y.: Randomised hough transform with error propagation for line and circle detection. Pattern Analysis & Applications 6(1), 55–64 (2003)
17. Yamani, M., Idris, I., Razak, Z., Zulkiee, K.: Online handwriting text line segmentation: A review. IJCSNS International Journal of Computer Science and Network Security 8(7) (2008)
18. Likforman-Sulem, L., Hanimyan, A., Faure, C.: A hough based algorithm for extracting text lines in handwritten documents. In: Proceedings of the Third International Conference on Document Analysis and Recognition, vol. 2, pp. 774–777. IEEE (1995)
19. Maddouri, S.S., Samoud, F.B., Bouriel, K., Ellouze, N., El Abed, H.: Baseline extraction: Comparison of six methods on ifn/enit database. In: The 11th International Conference on Frontiers in Handwriting Recognition (2008)
20. Burrow, P.: Arabic handwriting recognition. m.sc. thesis. Master's thesis, University of Edinburgh, England (2004)
21. Al-Shatnawi, A.M., Omar, K.: Methods of arabic language baseline detection, the state of art. ARISER 4, 185–193 (2008)
22. Pal, U., Sarkar, A.: Recognition of printed urdu script. In: Proceedings of the Seventh International Conference on Document Analysis and Recognition, ICDAR 2003 (2003)
23. Ahmad, Z., Orakzai, J.K., Shamsher, I.: Urdu compound character recogni-tion using feed forward neural networks. In: 2nd IEEE International Conference on Computer Science and Information Technology, ICCSIT 2009, pp. 457–462. IEEE (2009)
24. Sattar, S.A., Haque, S., Pathan, M.K.: Nastaliq optical character recognition. In: Proceedings of the 46th Annual Southeast Regional Conference on XX, pp. 329–331. ACM (2008)
25. http://en.wikipedia.org/wiki/Nastaliq_script
26. Javed, S.T., Hussain, S., Maqbool, A., Asloob, S., Jamil, S., Moin, H.: Segmentation free nastalique urdu ocr. Word Academy of Science, Engineering and Technology (2010)
27. Razzak, M.I., Sher, M., Hussain, S.A.: Locally baseline detection for online Arabic script based languages character recognition. International Journal of the Physical Sciences 5(7), 955–959 (2010)
28. Wali, A., Gulzar, A., Zia, A., Ghazali, M.A., Rafiq, M.I., Niaz, M.S., Hussain, S., Bashir, S.: contextual shape analysis of Nastaliq
29. Razzak, M.I., Hussain, S.A., Sher, M., Khan, Z.S.: Combining offline and online preprocessing for online urdu character recognition. In: Proceedings of the International MultiConference of Engineers and Computer Scientists, vol. 1, pp. 18–20 (2009)
30. Razzak, M.I., Anwar, F., Husain, S.A., Belaid, A., Sher, M.: Hmm and fuzzy logic: A hybrid approach for online urdu script-based languages character recognition. Knowledge-Based Systems 23(8), 914–923 (2010)
31. Razzak, M.I., Husain, S.A., Mirza, A.A., Belad, A.: Fuzzy based preprocessing using fusion of online and oine trait for online urdu script based languages char-acter recognition. International Journal of Innovative Computing, Information and Control 8, 1349–4198 (2012)

32. Razzak, M.I., Husain, S.A., Mirza, A.A., Khan, M.K.: Bio-inspired multilayered and multilanguage Arabic script character recognition system. International Journal of Innovative Computing, Information and Control 8 (2012)
33. Razzak, M.I.: Online Urdu Character Recognitio. In: Unconstrained Environment. PhD thesis, International Islamic University, Islamabad (2011)
34. Sardar, S., Wahab, A.: Optical character recognition system for Urdu. In: 2010 International Conference on Information and Emerging Technologies (ICIET), pp. 1–5. IEEE (2010)
35. Javed, S.T., Hussain, S.: Improving Nastalique specific pre-recognition process for Urdu OCR. In: IEEE 13th International Multitopic Conference (INMIC 2009), pp. 1–6 (2009)
36. Shafait, F., Keysers, D., Breuel, T.M., et al.: Layout analysis of Urdu document images. In: Multitopic Conference, INMIC 2006, pp. 293–298. IEEE (2006)
37. Breuel, T.M.: High performance document layout analysis. In: Proceedings of the Symposium on Document Image Understanding Technology, pp. 209–218 (2003)
38. Breuel, T.M.: Two geometric algorithms for layout analysis. In: Lopresti, D.P., Hu, J., Kashi, R.S. (eds.) DAS 2002. LNCS, vol. 2423, pp. 188–199. Springer, Heidelberg (2002)
39. Sattar, S.A., Shah, S.: Character Recognition of Arabic Script Languages. In: ICCIT 2012 (2012)
40. Naz, S., Hayat, K., Anwar, M.W., Akbar, H., Razzak, M.I.: Challenges in Baseline Detection of Cursive Script Languages. In: Science and Information Conference 2013, London, UK, October 7-9 (2013)
41. Mukhtar, O., Setlur, S., Govindaraju, V.: Experiments on urdu text recognition. In: Guide to OCR for Indic Scripts, pp. 163–171 (2010)

Gaze Input for Ordinary Interfaces: Combining Automatic and Manual Error Correction Techniques to Improve Pointing Precision

Enrico De Gaudenzi and Marco Porta

Dipartimento di Ingegneria Industriale e dell'Informazione - Università di Pavia
Via Ferrata 1 - 27100 - Pavia - Italy
enrico@degaudenzi.eu, marco.porta@unipv.it

Abstract. Although eye tracking technology has greatly advanced in recent years, gaze-based interaction is still not as comfortable and effective as traditional mouse or touchpad input. In this paper we present the solutions we have developed to enhance eye pointing in ordinary interfaces, often characterized by small graphical elements. The described approach combines both automatic and manual error correction techniques (namely *eye data filtering*, *virtual magnetization*, *magnifying glass* and *cursor manual shift*) to allow the use of gaze as an alternative or supplementary communication channel. A modified keyboard with two additional keys is also proposed, with the purpose to speed up gaze interaction. Experiments have shown that the approach can provide better performance than touchpad pointing and compares fairly with the mouse. The obtained results, while preliminary, can be the starting point for deeper and more focused investigations, driving further research on the topic.

Keywords: alternative communication, eye input, eye pointing, eye tracking, gaze-added interfaces, gaze-based interaction.

1 Introduction

The WIMP interaction paradigm, introduced in the eighties and based on Windows, Icons, Menus and Pointing devices, transformed the personal computer into a household instrument, usable for many kinds of purposes. The greater intuitiveness of graphic interaction has simplified the approach to the computer for novices and has also increased the productivity of those who use it for their professional work.

However, as we move towards a world where information technology will affect almost any aspect of our life, the need arises for more intuitive ways of interacting with the computer and electronic devices in general. Perceptive User Interfaces — also called Perceptual User Interfaces when integrated with multimedia output and other possible forms of input [1] — try to provide the computer with perceptive capabilities, so that implicit and explicit information about the user and his or her environment can be acquired: the machine thus becomes able to "see", "hear", etc.

L. Chen, S. Kapoor, and R. Bhatia (eds.), *Intelligent Systems for Science and Information*, Studies in Computational Intelligence 542,
DOI: 10.1007/978-3-319-04702-7_12,

Interface research is now going in several directions, and new and more natural input modalities will probably find application in graphical user interfaces (GUIs), joining and partly replacing traditional interaction paradigms based on keyboard and mouse. While a few years ago one of the most recurrent keywords in the computer field was multimedia, another term is now contending with it for the first place: multimodal. Multimodal systems combine natural input modes — such as speech, hand gestures, eye gaze, and head/body movements — with multimedia output. Sophisticated multimodal interfaces can integrate complementary modalities to get the most out of the strengths of each mode, and overcome weaknesses [2].

In particular, a computer that can understand what we are looking at could undoubtedly enhance the quality of human-machine interaction, opening the way to more sophisticated communication paradigms. Eye Tracking relates to the capability of some devices to detect and measure eye movements, with the aim to precisely identify the user's gaze direction. An eye tracker is a device able to detect the user's gaze while looking at a screen or at other elements in real-life settings [3].

Techniques exploiting computer-controlled video cameras have been the standard for the last two decades. However, first-generation eye tracking methods (until about the end of the 1990s) were very invasive, requiring special equipment to be mounted on the user's head. Fortunately, current eye tracking systems have evolved to the point that the user can almost freely move in front of the camera (within certain limits). Video-based eye trackers exploit infrared or near-infrared lighting, which is practically invisible and thus not disturbing, and follow the eyes by measuring how light is reflected by the cornea and by the retina through the pupil. Several fourth-generation eye tracking systems are now commercially available, which differ in implementation details and in the techniques employed to improve the eye position recognition process.

Eye-tracking methodologies have been studied and applied in several contexts. Many researches can be found, for example, in psychology, psychophysics, neuroscience, usability and advertising. In these fields, eye tracking is typically used to obtain the path followed by the user's eyes when looking at something (e.g. a picture or a web page), as well as to get indications about those areas of the screen which were most watched.

Present monitor-based eye trackers look almost like ordinary LCD screens, and limit user movements very little. The acquired data can then be recorded for subsequent use, or directly exploited to provide commands to the computer.

Gaze input is very important for people who cannot use their hands or cannot move at all. Eye tracking as an assistive technology has therefore been extensively studied and a variety of ad-hoc applications have been developed, ranging from writing (e.g. [4], [5]) to web surfing (e.g. [6]) and videogame control (e.g. [7]). These tools can significantly improve the accessibility of software interfaces, making them also usable by disabled people. In presence of serious motor impairments, that impede easy hand movements, using a keyboard and a mouse is impossible. In these situations, the eyes are usually the only communication channel for an individual.

However, there is now an increasing attention to the use of gaze input also for ordinary human-computer interaction, towards more effective and natural ways to

communicate with the computer. Although a main hindrance to the diffusion of eye trackers is currently their high cost, it is not unlikely that they may spread as a new engaging technology in the near future, even incorporated into laptops [8] or tablets [9]. The combination of traditional input devices and gaze-based approaches is a challenging research area that has received relatively little attention to date. The aim is to build gaze-added interfaces in which the eyes are an additional (potentially more efficient) input channel for the computer [10].

Unfortunately, accurate eye pointing is hampered by both physiological and technological hurdles. Eye movements occur as sequences of very fast (< 100 ms) saccades, followed by relatively stable fixation periods (~ 100 ÷ 600 ms). Even during fixations, however, the eye is not completely still, but is characterized by some jitters [11] (Fig. 1). In addition, even the precision of very recent eye trackers is limited (usually, around 0.5 degrees).

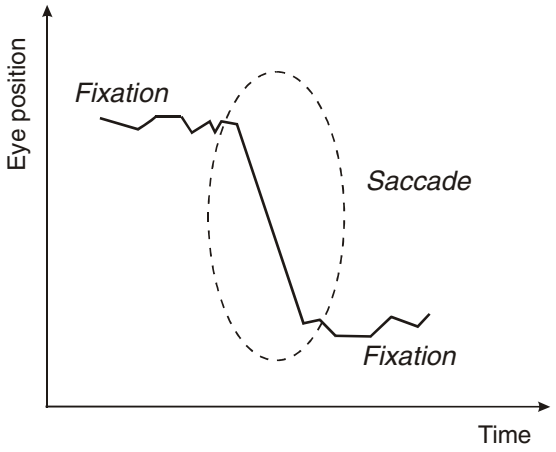


Fig. 1. Example of eye signal

It is thus almost impossible that two consecutive gaze samples are detected in the same screen point. For reliable human-computer interaction, mechanisms are therefore necessary to stabilize the eye signal.

To solve the limited precision problem, most eye-controlled interfaces are characterized by big graphical components, which can be easily selected even with a trembling pointer. In a general usage scenario, however, the eye tracker must be also exploitable to work within ordinary window-based operating environments, which contain several small elements.

In this paper we consider an interaction approach in which both automatic and manual error correction techniques are employed together, to improve the accuracy of gaze input. In particular, the first category of techniques includes an eye data filtering procedure, used to smooth the detected gaze coordinates, and a virtual magnetization method, employed to ease the pointing of small interface elements. The second group includes a classical screen magnification solution and a keyboard-based approach in

which the cursor, when necessary, can be manually shifted to the right position. The proposed solutions are part of our ongoing research aimed at making eye tracking a reliable and practical technology for human-computer interaction.

The paper is structured as follows. Section 2 briefly presents some previous research projects related to general exploitation of eye tracking as a replacement for the mouse. Section 3 describes the main features of the developed system. Section 4 illustrates the carried out experiments and the obtained results. Section 5, at last, draws some conclusions and provides hints for future research.

2 Related Work

Exploiting eye input as a replacement for the mouse is a potentially valuable way to interact with the computer, as already demonstrated more than twenty years ago in a famous study by Jacob [12], who identified and tested six different possible applications for eye-based interaction (namely "object selection", "continuous attribute display", "object motion", "eye-controlled scrolling text", "menu command selection", and "active window choice"). In particular, Sibert and Jacob [13] have demonstrated that, for the purpose of selecting an object on the screen, eye gaze interaction is faster than selection with a mouse. More recently, Oyekoya and Stentiford [14] have carried out some experiments to compare the speed of the eye and the mouse to control an interface. Testers were asked to find a target image from a series of 50 grid displays of 25 stimuli (24 distractors and one target image). Results showed faster target identification for the eye interface than the mouse. Although the speed per se cannot be considered an advantage (as it should be evaluated in the context of the specific application), these are undoubtedly very interesting findings. In fact, pointing actions are intrinsically connected with eye fixations [15]: when we use the mouse, we first identify the target, and then shift the cursor to it through an accurate ocular-hand coordination process.

To stabilize gaze data, several methods have been devised; a recent survey by Špakov [16] presents and compares some of the most exploited filters (weighted averaging over dynamic time window, two-mode low-pass, Savitzky-Golay, Kalman, weighted on-off, and some modified versions of them). Besides plain smoothing techniques, more complex strategies have also been considered, such as those described by Zhang et al. [17], all aimed at adjusting eye cursor trajectories by offsetting eye jitters.

Zooming is a widespread and relatively simple solution to improve eye pointing, and has been applied in several implementations. For example, one of the first applications of this method, in which the screen region being fixated is automatically displayed magnified after a "dwell time", is described by Lankford in [18]. A more recent project by Kumar et al. [19] requires the user to explicitly press a keyboard's key to show the enlarged content. Studies have also been carried out to compare eye-only and eye-with-zoom interaction in pointing tests [20]. As a variant of zooming, the "fish eye" lens effect allows the user to keep an overview of the screen while selectively zooming in on a specific area [21].

For very reliable gaze pointing, special pointers have also been devised. For instance, the cursor described in [22] by Porta et al. can be easily shifted among the icons on the screen or within windows, thus relieving the user from precise pointing actions. This cursor is very precise, although its speed cannot be compared with that of mouse operations.

Eye-hand mixed input solutions have been considered as well, with the purpose to improve the performance of common mouse-based activities. In the MAGIC project by IBM [23], for example, gaze is only exploited to roughly position the cursor, while the small shifts necessary to precisely move it are made by hand. Starting from the observation that it is unnatural to overload a perceptual channel such as vision with motor control duties, the authors propose an alternative approach in which gaze is used to dynamically redefine the "home" position of the pointing cursor, while the small movements necessary to precisely position it are made by hand. Practically, when the user looks at a certain object on the screen, the cursor is automatically and instantaneously moved within a circle area which, with a 95% probability, contains that object. Then, if the user's intention is really to select the item, the cursor is moved as usual through the mouse. Such an approach has the main advantage of not requiring extremely accurate eye tracking for cursor control, while providing an easy way to reduce mouse use.

The MAGIC principle has also inspired some other works. Among these, Yamato et al. [24] propose a technique for button selection in general GUIs in which the cursor position is derived from both automatic and manual adjustment; Räihä and Špakov [25] and Blanch and Ortega [26] describe an approach in which eye gaze is used to select the currently active cursor among several cursors displayed on the screen at the same time; Drewes and Schmidt [27] combine gaze input with a touch mouse to ease pointing tasks; and Stellmach and Dachselt [28] use an ordinary touch device to confirm eye selections and facilitate small target pointing.

3 System Description

As said in the Introduction, the proposed solution exploits both automatic and manual error correction techniques, to improve eye pointing precision as much as possible.

3.1 Eye Tracker and Development Environment

As an eye tracker, we use the Tobii 1750 (Fig. 2 left), which integrates all the necessary components (camera, infrared lighting, etc.) into a 17'' (1280x1024) monitor [29]. With an accuracy of 0.5 degrees and a relatively high freedom of movement, the system is ideal for real-use settings, where it would be intolerable to constrain users too much in their activities. For correct use of the device, at least one eye (better if both) must be within the field of view of the infrared camera, which can be represented as a box with size 20x15x20 cm placed at about 60 cm from the screen (Fig. 2 right). The sampling rate of the apparatus is 50 Hz, i.e. gaze data is recorded 50 times a second.

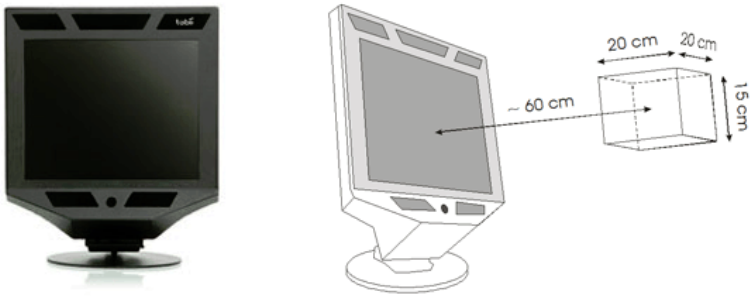


Fig. 2. The Tobii 1750 eye tracker (left) and field of view of the infrared camera (right)

The project has been developed in C# using the Microsoft .NET framework [30].

3.2 Optional Mouse Buttons Replacement

In the developed system, eye pointing is used to control the cursor, while the mouse is still exploited for left and right button clicks. However, it is also possible to think of a modified keyboard with two additional keys, for example placed above the left and right arrow keys (which are usually free areas). At a prototype level we have implemented such a tool, simply using the electronics of an ordinary mouse and a cardboard artifact (Fig. 3).

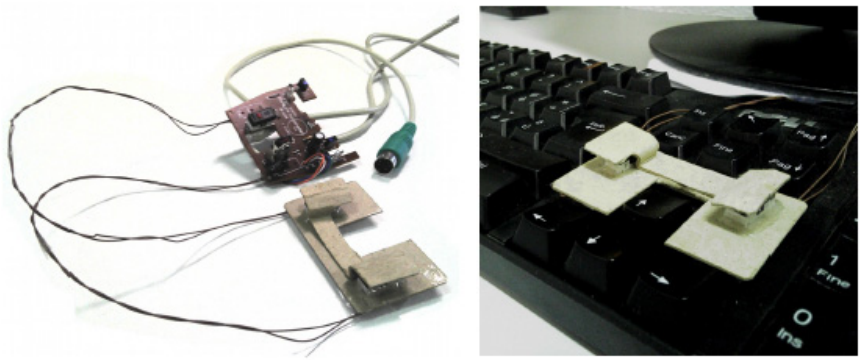


Fig. 3. Prototype implementation of an on-keyboard substitute for mouse buttons

The advantage of employing such a customized keyboard is that the user can both perform left and right clicks and, without removing the hand, adjust cursor position using the cursor manual shift technique that will be described in Section 3.4. While not strictly necessary, mouse buttons replacement results in a speed up of gaze interaction operations (especially in the first stages of system use), as experimented in several informal trials.

3.3 Eye Data Filtering

The filtering algorithm used to denoise raw eye data exploits two buffers, which we will call B_1 and B_2. B_1 contains the last 10 samples (gaze points, i.e. x and y gaze coordinates) acquired by the eye tracker. At each acquisition (which replaces the oldest sample), the arithmetic mean p_1 of the 10 points in B_1 is calculated.

Practically, B_1 can efficiently handle large cursor shifts with a maximum delay of 20 ms • 10 = 200 ms. When eye movements are smaller, however, this lag is too high, and makes cursor control more difficult. We therefore use a second buffer, B_2, which contains 10 elements like B_1. The value 10 derives from several trials, and allows to achieve a good tradeoff between system response times and comfortable use. As schematized in Fig. 4, B_2 is progressively filled with the arithmetic means computed on the points in B_1.

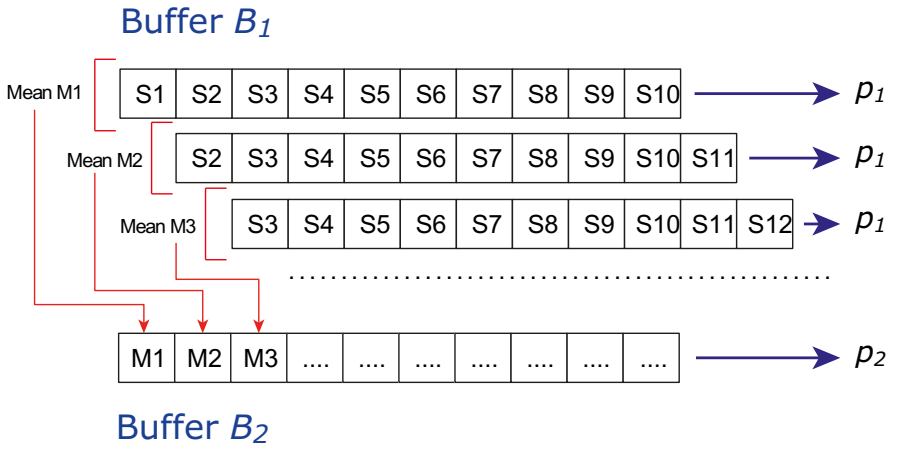


Fig. 4. Schematization of the eye data filtering process: buffer B_1 is progressively filled with the last 10 gaze samples S_i, while buffer B_2 is progressively filled with the arithmetic means M_j of the samples in B_1

After each sample acquisition, the mean p_2 of the values in B_2 is calculated. As illustrated in Fig. 5, at each step n, only one of the two points p_1 and p_2 determines the cursor position, according to the following rule: if p_1 is contained in the rectangular area delimited by the points in B_1 at step n-1, then the cursor will be displayed in p_2; otherwise it will be displayed in p_1. Buffers B_1 and B_2 represent a double structure which on the one hand provides a fast but less accurate eye data stabilization, and on the other hand allows a slower but more precise gaze control.

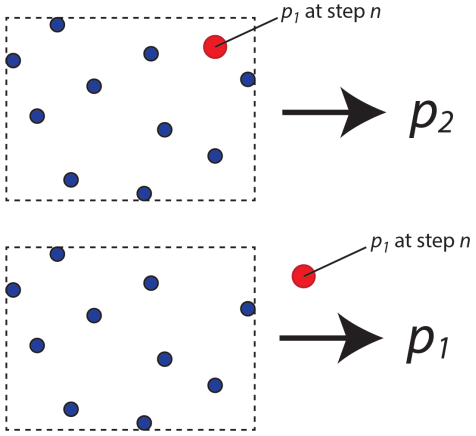


Fig. 5. Choice between points p_1 and p_2 as the new stabilized gaze position

Although more complex solutions could be used to filter eye data (e.g. those presented in [16]), the above described gaze control algorithm has the advantage of being simple and fast enough for our purposes.

3.4 Virtual Magnetization

To simplify the selection of GUIs' control elements, the cursor is automatically "captured" when the gaze is perceived near them. The approach, to some extent similar to the force field method described in [17], exploits MS Windows native functions to find the positions of graphical elements. MS Windows visual elements (windows and other graphical objects) are organized into a hierarchical structure that makes it relatively easy to access their properties.

The virtual "magnetization" is implemented within the eye data filtering algorithm: if the cursor distance from a graphical object (be it a menu, a button, etc.) is less than a threshold, then half of the elements in buffer B_1 are replaced with the coordinates of the center of the object. This way, the cursor is only "softly" caught by the GUI's element, and can be easily moved away when necessary.

While more complex techniques exist to facilitate gaze-to-objects mapping (a good survey can be found in [31]), the solution used in this project turns out to be both simple and effective for our needs.

3.5 Cursor Manual Shift

In spite of the (always necessary) initial calibration procedure, an eye tracker may be characterized by a systematic error in gaze detection, usually depending on the position of the observed area of the screen and possibly also on the specific user.

The developed system allows to manually fix this error by means of the keyboard's arrow keys. Practically, the screen is virtually subdivided into 20x10 pixel rectangles

(size deriving from empirical trials and modifiable through a configuration file). Each time the user realizes that the cursor is not displayed in the actually watched spot, he or she can employ the four arrow keys to move it to the correct location (Fig. 6).

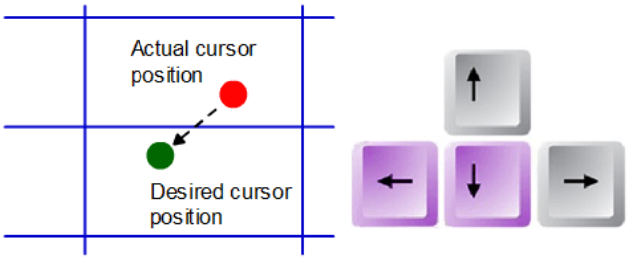


Fig. 6. Manual correction of cursor position

The system stores this information in a file (for each user), so that it can be subsequently exploited in case the same situation occurs again, also in next executions. In other words, if the wrong position of the cursor was within a rectangle r_1 and the user shifted it to a new location l within a rectangle r_2, then, when the gaze will be perceived again within r_1, the cursor will be automatically moved to l, inside r_2.

It is important to stress again that, with an implementation like that shown in Fig. 3, the user can both press the left/right buttons and adjust the cursor's position without removing the hand from the keyboard (Fig. 7).

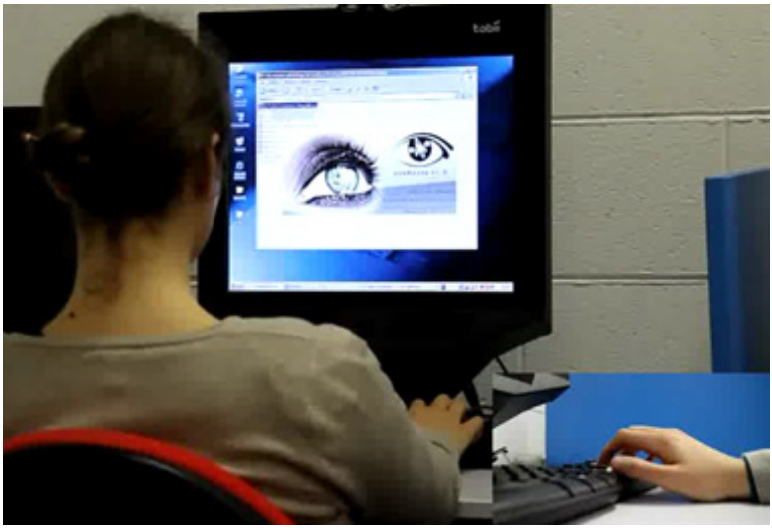


Fig. 7. Example of system use

3.6 Magnifying Glass

To allow precise pointing of very small elements, the press of the right CTRL key on the keyboard displays a square window showing an enlarged version of the area surrounding the current cursor position (Fig. 8).

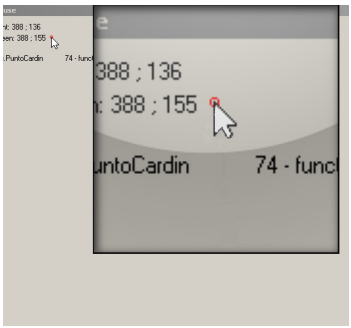


Fig. 8. Magnifying glass

Like the other parameters, the magnifying factor can be chosen according to one's preference (3 in our experiments). While not new (as explained in Section 2, zooming has been exploited in several gaze-based interfaces), the magnifying glass represents a simple and effective way to ease eye pointing. The right CTRL key can be easily pressed without the need to remove the hand from the arrow keys or from the buttons of the tool shown in Fig. 3.

4 Experiments

Besides informally testing the system many times during its development, we have also carried out two more structured tests (T_1 and T_2) once it was fully implemented, aimed at quantitatively and qualitatively assessing its performance.

4.1 Test T_1

Participants and Procedure. Test T_1 involved 18 volunteer participants (aged between 21 and 50, mean = 32, 10 males and 8 females). All testers reported normal or corrected-to-normal vision and were regular computer users. Only two of them had been previously involved in other eye tracking experiments (of different kind). The test was composed of three tasks (t_1, t_2 and t_3) regarding the use of standard components of the MS Windows XP operating system.

Each task was subdivided into subtasks, namely:

Task t_1:	*a.*	Open Windows calendar (double click in the bottom right corner of the screen)
	b.	Close the calendar (click on the 'X' button in the upper right corner of the window)
Task t_2:	*a.*	Open Internet Explorer (through the 'Start' menu at the bottom left corner of the screen)
	b.	Close Internet Explorer (through the 'Exit' option of the 'File' menu)
Task t_3:	*a.*	Open Windows calculator (menu 'Start' → 'Programs' → 'Accessories' → 'Calculator')
	b.	Perform the operation 2 + 2
	c.	Close the calculator (click on the 'X' button in upper right corner of the window)

Prior to the beginning of the experiments, participants were instructed on how to use the eye input system and could autonomously practice with it for 15 minutes. In this stage, they could also perform manual corrections, in order to adjust gaze detection errors. A calibration procedure was performed both before the initial free trial and before the actual tests. The calibration consisted of following a moving circle on the screen and lasted a few seconds.

Each task was performed by each tester with mouse, touchpad and eye pointing (in random order), using a 1024x768 screen resolution. For eye pointing, the modified keyboard of Fig. 3 was employed. For touchpad input, a laptop was connected to the eye tracker (and positioned in front of it), so that the touchpad could be used to control the cursor on the eye tracker's screen. The dependent variable of the study was the execution time of each task.

Results. Table 1 shows the means (*M*) and standard deviations (*SD*) obtained for the three input solutions.

Table 1. Results of test T_1 (times in seconds)

		Mouse		**Touchpad**		**Eye Point.**			
		M	*SD*	***M***	*SD*	***M***	*SD*	*F*	*p*
t_1	***a***	**2.8**	1.3	**5.9**	1.4	**3.3**	0.9	108.60	<.001
	b	**2.0**	0.8	**2.9**	0.8	**1.5**	1.3	8.37	.001
t_2	***a***	**3.7**	0.6	**4.1**	0.6	**4.2**	0.5	2.88	.07
	b	**3.3**	0.9	**4.4**	0.5	**3.5**	0.8	11.01	<.001
t_3	***a***	**6.3**	1.5	**9.8**	1.4	**9.3**	1.9	31.66	<.001
	b	**3.9**	0.6	**7.6**	1.0	**7.0**	1.0	202.24	<.001
	c	**1.3**	0.2	**3.0**	0.9	**2.5**	1.6	10.39	<.001

The last two columns contain, for each subtask, the *F(2,34)* and p values of a repeated-measures (within-subjects design) ANOVA.

The histogram in Fig. 9 graphically compares the means for the three methods and the three tasks.

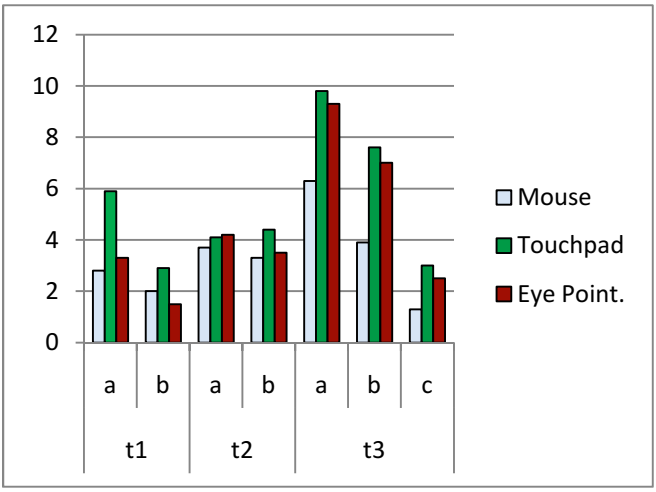


Fig. 9. Histogram of mean times (in seconds) for test T_1

As can be seen from the table, the differences between means are statistically significant (5% significance level), except those of subtask t_{2a}. Apart from this case, the mean times for eye pointing are all lower than those for the touchpad, even if not always markedly. As could be easily expected, the mouse — the pointing device we are all accustomed to using in our everyday computer work — shows the best performance.

At the end of the test, each participant was asked to judge (within a 0-10 scale) the eye input modality on the whole, in relation to its "usefulness" (can it effectively substitute ordinary input devices?), "intuitiveness" (first impression) and "comfort" (user-friendliness). Table 2 shows the outcomes for the three qualities.

Table 2. User subjective judgment (mean and standard deviation, 0-10 scale)

	M	*SD*
Usefulness	**8.2**	0.8
Intuitiveness	**9.2**	0.8
Comfort	**7.8**	0.8

Testers were also free to express their opinions about the proposed gaze-based input system. Comments were mainly related to the initial difficulty in controlling the cursor, especially to point small interface elements. However, they also stressed the usefulness of the manual correction of cursor position, that progressively improved pointing accuracy. The prototype on-keyboard substitute for mouse buttons was also appreciated, as

well as the magnifying glass (especially in areas near the borders of the screen). One participant reported a little eye strain, probably due to the high effort put in the test.

4.2 Test T_2

Participants and Procedure. The purpose of Test T_2 was to appraise and compare the gaze detection precisions of the eye tracker per se and of the developed eye data filtering algorithm.

Three testers (two males and one female, aged 24, 25 and 27) watched ten target points appearing on the screen in random positions, each one being displayed for two minutes, and the measured variable was the distance (in pixels) of the cursor from the target point.

Results. Considering a 100% "success percentage" (*sp*) when the distance of the cursor from the target point is zero and a 0% *sp* when the distance is greater than half the screen diagonal, the mean *sp* for the eye tracker raw signal was 97.44%, while the mean sp for the stabilized signal was 99.57%. As an example, Fig. 10 shows a graphical representation of the detected gaze points for a specific tester and target.

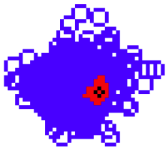


Fig. 10. Example of raw (blue) and filtered (red) gaze positions detected for a tester looking at a target point

The average distances of the eye tracker's raw gaze points and of the stabilized points from the target were, respectively, 15 pixels and 2.5 pixels.

Although three testers are of course too few to draw unquestionable conclusions about the reliability of the stabilization process, test T_2 provided an initial (and positive) quantitative feedback.

5 Conclusions

Due to the limited precision of current eye trackers, most existing eye-controlled interfaces are specifically designed to make up for such weakness. For example, they are characterized by big graphical elements that can be easily selected even if the user's gaze is not detected exactly on them. Program suites developed for commercially-available eye trackers (e.g. [32]), usually targeted at assistive technology, are collections of applications sharing graphical look and interaction mechanisms, designed on-purpose for eye gaze interaction.

However, while a dedicated environment for the execution of eye-controlled programs has certainly a number of advantages, it has some limitations as well. First of all, it constrains the user to employ only software tools that are available in the suite:

any other application installed on the computer cannot be controlled by means of the eyes (or, if so, the task is very difficult, because elements of ordinary graphical user interfaces are usually small and not designed for easy eye pointing). Moreover, program suites are often associated with specific eye trackers: if, for any reason, the user wants to change the device, the old applications may not work properly on the new system. When the market of eye trackers will expand (in a hopefully not too far future), the decrease of prices is likely to accentuate such problem.

In particular, in this paper we have presented some solutions for improving gaze input in ordinary computer use. This work is part of our ongoing research aimed at making gaze-added interfaces (in which gaze input is an additional input channel, besides keyboard and mouse) more robust and effective.

The combined exploitation of automatic techniques and manual strategies has the purpose to seamlessly reduce user effort, while coping with intrinsic limits of current technology. Specifically:

- The proposed eye data filtering algorithm turns out to be simple but helpful for our purposes, being precise and flexible at the same time;
- Virtual magnetization eases eye pointing with small GUI elements, "capturing" the cursor and, in a sense, anticipating user intentions;
- Manual correction allows gaze detection errors to be corrected in the first stages of system use, thus progressively increasing its reliability;
- The magnifying glass helps the user to focus on interface details;
- The on-keyboard replacement for mouse buttons allows effective combined use of automatic and manual gaze correction techniques.

We think that the relatively positive performance of the described input solution compared to the touchpad is a significant result: in those cases in which the mouse is not available or not practical (such as in laptops), gaze cursor control can potentially speed up the interaction. Moreover, testers' encouraging judgments about system usefulness, intuitiveness and comfort (Table 2) are a stimulus to continue our research.

Future work will include:

- Further experiments, aimed to confirm the obtained outcomes and investigate new possible improvements;
- The use of variable-size rectangles for the cursor manual shift method, to take into account the different precisions of the eye tracker depending on gaze position;
- The joint use of gaze interaction and traditional mouse input, towards a real integration of eye pointing into daily computer-based activities;
- The extension of the system to a completely hands-free interaction, especially useful for people who cannot employ the hands. As shown in Fig. 11, our idea is to display a popup "toolbar" each time the user fixates a point of the screen for more than a certain time (e.g. one second). The toolbar, displayed with a semi-transparent effect, contains four buttons denoting the four basic mouse operations *Click*, *Double Click*, *Drag* and *Right Click*. By watching one of these buttons, the user will be able to perform the corresponding operation.

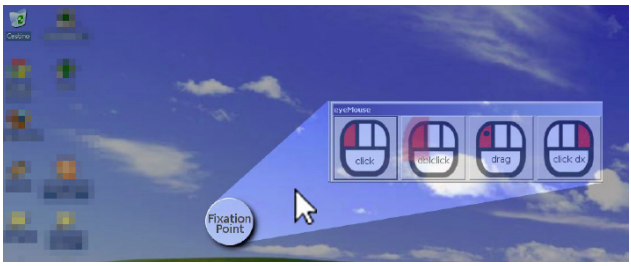


Fig. 11. Gaze-activated toolbar for hands-free interaction

References

1. Pentland, A.: Perceptual intelligence. Communications of the ACM 43, 35–44 (2000)
2. Oviatt, S.: Ten Myths of Multimodal Interaction. Communications of the ACM 42(11), 74–81 (1999)
3. Duchowski, A.T.: Eye Tracking Methodology – Theory and Practice, 2nd edn. Springer, London (2007)
4. Ward, D.J., Mackay, J.C.: Fast Hands-free Writing by Gaze Direction. Nature 418, 838 (2002)
5. Porta, M., Turina, M.: Eye-S: a Full-Screen Input Modality for Pure Eye-based Communication. In: Proc. ETRA 2008 (Symposium on Eye Tracking Research and Applications), pp. 27–34. ACM Press (2008)
6. Porta, M., Ravelli, A.: WeyeB, an Eye-Controlled Web Browser for Hands-Free Navigation. In: Proc. HSI 2009 (Conference on Human System Interaction), pp. 210–215. IEEE Press (2009)
7. Istance, H., Hyrskykari, A., Immonen, L., Mansikkamaa, S., Vickers, S.: Designing Gaze Gestures for Gaming: an Investigation of Performance. In: Proc. ETRA 2010 (Symposium on Eye Tracking Research and Applications), pp. 323–330. ACM Press (2010)
8. Eisenberg, A.: Pointing With Your Eyes, to Give the Mouse a Break. New York Times (March 26, 2011), retrieved from `http://www.nytimes.com/2011/03/27/business/27novel.html?r=2` (retrieved from September 10, 2012)
9. Westover, N.: Tobii Brings Eye Control to Tablets. PCMAG.COM (September 19, 2013), retrieved from `http://www.pcmag.com/article2/0,2817,2424565,00.asp`
10. Salvucci, D.D., Anderson, J.R.: Intelligent gaze-added interfaces. In: Proc. CHI 2000 (Conference on Human Factors in Computing Systems), pp. 273–280. ACM Press (2000)
11. Yarbus, A.L.: Eye Movements and Vision. Plenum Press, New York (1967)
12. Jacob, R.J.K.: The Use of Eye Movements in Human-Computer Interaction Techniques: What You Look At is What You Get. ACM Transactions on Information Systems 9(3), 152–169 (1991)
13. Sibert, L.E., Jacob, R.J.K.: Evaluation of Eye Gaze Interaction. In: Proc. CHI 2000 (Conference on Human Factors in Computing Systems), pp. 281–288. ACM Press (2000)
14. Oyekoya, O., Stentiford, F.W.M.: A Performance Comparison of Eye Tracking and Mouse Interfaces in a Target Image Identification Task. In: Proc. EWIMT 2005 (2nd European Workshop on the Integration of Knowledge, Semantics and Digital Media Technology), pp. 139–144 (2005)

15. Smith, B.A., Ho, J., Ark, W., Zhai, S.: Hand Eye Coordination Patterns in Target Selection. In: Proc. ETRA 2000 (Symposium on Eye Tracking Research and Applications), pp. 117–122. ACM Press (2000)
16. Špakov, O.: Comparison of Eye Movement Filters Used in HCI. In: Proc. ETRA 2012 (Symposium on Eye Tracking Research and Applications), pp. 281–284. ACM Press (2012)
17. Zhang, X., Ren, X., Zha, H.: Improving Eye Cursor's Stability for Eye Pointing Tasks. In: Proc. CHI 2008 (Conference on Human Factors in Computing Systems), pp. 525–534. ACM Press (2008)
18. Lankford, C.: Effective Eye Gaze Input into Windows. In: Proc. ETRA 2000 (Symposium on Eye Tracking Research and Applications), pp. 23–27. ACM Press (2000)
19. Kumar, M., Paepcke, A., Winograd, T.: EyePoint: Practical Point and Selection Using Gaze and Keyboard. In: Proc. CHI 2007 (Conference on Human Factors in Computing Systems), pp. 421–430. ACM Press (2007)
20. Bates, R.: Multimodal Eye-Based Interaction for Zoomed Target - Selection on a Standard Graphical User Interface. In: Proc. Interact 1999, vol. II, pp. 7–8 (1999)
21. Ashmore, M., Duchowski, A.T., Shoemaker, G.: Efficient Eye Pointing with a Fisheye Lens. In: Proc. GI 2005 (Graphics Interface), pp. 203–210 (2005)
22. Porta, M., Ravarelli, A., Spagnoli, G.: ceCursor, a Contextual Eye Cursor for General Pointing in Windows Environments. In: Proc. ETRA 2010 (Symposium on Eye Tracking Research and Applications), pp. 331–337. ACM Press (2010)
23. Zhai, S., Morimoto, C., Ihde, S.: Manual And Gaze Input Cascaded (MAGIC) Pointing. In: Proc. CHI 1999 (Conference on Human Factors in Computing Systems), pp. 246–253. ACM Press (1999)
24. Yamato, M., Monden, A., Matsumoto, K.: Button Selection for General GUIs Using Eye and Hand Together. In: Proc. AVI 2000 (Conference on Advanced Visual Interfaces), pp. 270–273. ACM Press (2000)
25. Räihä, K., Špakov, O.: Disambiguating Ninja Cursors with Eye Gaze. In: Proc. CHI 2009 (Conference on Human Factors in Computing Systems), pp. 1411–1414. ACM Press (2009)
26. Blanch, R., Ortega, M.: Rake Cursor: Improving Pointing Performance with Concurrent Input Channels. In: Proc. CHI 2009 (Conference on Human Factors in Computing Systems), pp. 1415–1418. ACM Press (2009)
27. Drewes, H., Schmidt, A.: The MAGIC Touch: Combining MAGIC-Pointing with a Touch-Sensitive Mouse. In: Gross, T., Gulliksen, J., Kotzé, P., Oestreicher, L., Palanque, P., Prates, R.O., Winckler, M. (eds.) INTERACT 2009. LNCS, vol. 5727, pp. 415–428. Springer, Heidelberg (2009)
28. Stellmach, S., Dachselt, R.: Look & Touch: Gaze-supported Target Acquisition. In: Proc. CHI 2012 (Conference on Human Factors in Computing Systems), pp. 2981–2990. ACM Press (2012)
29. Tobii Technology AB: Tobii 1750 Eye Tracker (2006), `http://www.tobii.com`
30. Prosise, J.: Programming Microsoft NET. Microsoft Press, Redmond (2002)
31. Špakov, O.: Comparison of Gaze-to-Objects Mapping Algorithms. In: Proc. NGCA 2011 (Conference on Novel Gaze-Controlled Applications), article 6. ACM Press (2011)
32. MyTobii User Manual, Version 2.4. Retrieved from `http://www.tobii.com/Global/Assistive/Downloads_Training_Documents/MyTobii_P10/PDF/User%27s%20Manual/MyTobii_User_Manual_English.pdf` (October 26, 2013)

Data Mining Approach in Host and Network-Based Intrusion Prevention System

Alaa H. Al-Hamami[1] and Ghossoon M.Waleed Al-Saadoon[2]

[1] College of Computer Sciences and Informatics
Computer Science Dep.
Amman Arab University
Amman, Jordan
[2] Applied Science University
Administrative Sciences College
Management Information Systems Dep.
Manama, Bahrain
alaa_hamami@yahoo.com, ghowaleed2004@yahoo.com

Abstract. Intrusion Prevention Systems (IPS) as a security solution have their own characteristics in analysing, detecting and preventing intruders' acts. It provides a quite good service in securing the network, which goes further than the functionality of Intrusion Detection Systems (IDS), firewalls, antivirus and any security applications. This is by actively responding to attacks and affording great flexibility when dealing with security threats.

Host based IPS mostly depend on a static signature mechanism to identify intruders, which in turn needs to be updated from time to time to insure the most accurate detection. The use of improved Network Intrusion Prevention System (NIPS) based on two mechanisms is to detect patterns of known intrusions (misuse detection) and to distinguish anomalous network activity of intrusion from normal network traffic (anomaly detection) effectively. The Data Mining methods have been used in this chapter to enhance NIPS based on anomaly detection.

In this chapter we try to enhance intruders' detection, by replacing the static database with a dynamic one, and even more adding intelligence to the detecting mechanism through Data Mining. A feedback to the whole process is being made to help in making future inspections to be more realistic.

The use of Data Mining methods will result in the development of a Network Intrusion Prevention System (NIPS) as an internal security gateway for defending against attacks and threats from within and outside the computer network system. In addition, it will help to detect anomalous activity comprising suspicious probing inside the network before it launches any network attacks with damaging effects.

The study aims to enhance the Snort tool, which consists of a NIPS based on both misuse- and anomaly-detection mechanisms, by using two sub-phases of Data Mining approaches: an improved K-mean clustering algorithm and a PF-growth algorithm. The integration of these two sub-phases helps to discover new rules, especially those related to internal network scans; in addition, the unsupervised learning process in the K-mean algorithm is used to discover new

L. Chen, S. Kapoor, and R. Bhatia (eds.), *Intelligent Systems for Science and Information*, Studies in Computational Intelligence 542,
DOI: 10.1007/978-3-319-04702-7_13,

clusters which may represent a new type of attack depending on the decisions of analysts.

The Host based IPS will contribute to achieving enhancement in the following: evolving the techniques of investigating activities due to the use of Data Mining, integrate or could eliminate antivirus programs installed on Personal Computer (PC), and Maximize the level of security of the whole network through securing single host.

Integrating of two of Data Mining approaches (K-mean clustering and PF-Growth algorithm) helps to discover new rules especially those related to internal network scans, besides unsupervised learning process in K-mean algorithm is used to discover new cluster may represent a new type of attack depending on decisions of analysts.

All that work, helps to enhance and develop NIPS tool, by involving Data Mining approaches in investigating anomalies. Besides achieve objective to be a complete system performs requirements such as detect probe attack inside source of network and prevent it before launch network attack to the target machine with high performance, reduce false alarm, easy building system with low cost, and compatibility with any operating system. Furthermore, maximize the effectiveness in identifying attacks, thereby helping the users to construct more secure information systems.

Keywords: Intrusion Prevention System, Data Mining, K-Means, PF-G Algorithm, Internet Protocol.

1 Introduction

The most important points and challenges facing the technology are the information security and preservation of its integrity from any intrusion. In spite of the huge development in technology and networks with increased value of the information stored accompanied by an increase in software and hardware products that specializes in security and trying to maintain the integrity of this information but was accompanied by an increase in the risk of intrusion and hacker exploited weaknesses in the security products.

Recently, most of the information about an individual is stored online by companies and government organizations; for example, a finance company and mortgage can keep information on customer financial credit rating, social security number, bank account numbers, and a lot more personal information of the customer. The intruders may break into the system and copy data, and the user never knows. Therefore, the damage from digital personal data loss may be far greater than loss of physical data; also, damages caused by a hacker either breaking into a network or using a computer to launch an attack on another networks are possible. Computer attacks become more sophisticated and skilled; organizations today are keenly aware of the need to provide effective security and protect their information system, and existence of networks requires the protection of the gateway and the nodes.

Although both Intrusion Detection Systems (IDS) and Intrusion Prevention System (IPS) can detect an intruder, IPS have a plus for preventing intrusion by taking certain

actions once it is being detected. Among all security solutions, IPS has its own special characteristics. These special characteristics made IPS a new trend for researchers at the moment, many papers have discussed ways for choosing the best IPS, and other papers discussed new attack detecting mechanisms.

Although many security applications are available such as IDS , IPS, anti-virus/spam systems; firewalls have been proposed to control the attacks, securing distributed systems and networks is still extremely challenging. Each one of these security applications covers one component of the total network security picture. Some of these are distinguished by features such as detection without prevention, the use of a mechanism (either anomaly-based or signature-based detection/prevention), or the ability to defend the network from outside threats, but they are limited when it comes to detecting threats coming from within the network computer system. Other tools focus on improving latency, or "the time it takes to respond and take appropriate action… this period of time is critical in the success of an attack", without taking into account any undesirable increase in false negatives: "any malicious traffic that makes it through the security applications to the production network" or false positives: "any legitimate traffic that the security applications drops because it appears to be anomalous" [1].

2 Statement of Problem

Host based IPS mostly depends on a static signature mechanism to identify intruders, which in turn needs to be updated from time to time to insure the most accurate detection. To overcome the static signature detecting mechanism, we introduce in this chapter a four layers host based IPS, which uses Data Mining technique, namely decision tree, as a detecting mechanism instead. Data Mining technique that will be used is the association analysis, namely the decision tree J48 algorithm. Association analysis will be used to find out anomaly cases, where the behaviour of the user is very different from the normal, which could indicate a threat.

This chapter will use one of the IPS system , which use rule (signature)-based prevention integrated with Data Mining, which can detect and expect anomaly attack. As well as the capacity and effectiveness of Data Mining in dealing with the huge information stored in the database of the network and high speed processing search and extract the appropriate decision. Network administrators can determine the security policy violations using analysis of enough data collected. Unfortunately, the data is so huge even for a small network and traditional methods of analysis so time consuming and difficult even with computer assistance because foreign features can make it harder to detect suspicious behaviour patterns, complex relationship exist between the features, which are particularly impossible for humans to discover, the solution to this problem is using Data Mining approach. Using Intrusion Detection/Prevention and Data Mining are capable of working together efficiently to provide network security.

The objective of this chapter is to propose a system to protect the internal gateway of the computer network as well as its hosts from signature and anomaly attacks, and able to detect all threats especially those occur inside the network. Furthermore, it will decrease the load of the intrusion prevention task on every individual host, reduce the

possibility human errors that occur during contribution to make the right decision and try to achieve the desired reduction of the rate false positive, false negatives and minimized latency.

This chapter includes an example of implementation for using Data Mining which applied the rules of Intrusion Prevention Systems that use; this methodology can detect and predict anomaly attacks using the capacities and effectiveness of Data Mining.

3 Intrusion Prevention System

Intrusion Prevention System (IPS) any device (hardware or software) that has the ability to monitor, detect attacks and activities for malicious or unwanted behaviour and can react in real time to prevent the attack from being successful.
Any Enterprises without security strategy prevent its information from external and internal threats open the door to unacceptable risks, costs, undesired access, malicious content, and rate-based attacks. The main objective of security techniques system reviews network weakness and takes steps to maintain security and protect information corporate assets and intellectual property from spyware and other intruders.

An IPS is typically used on the outer boundary of a network to prevent any malicious traffic from reaching potentially vulnerable systems inside the network that may contain sensitive information. Currently, many research studies tend to focus on improving the work of IPS for computer networks, some focus on the various types of mechanism detection and prevention, and others on choosing the best IPS architecture for a network. Data Mining could contribute to the enhancement of the applications of network intrusion detection/prevention systems. Data Mining uses one or more techniques in the context of intrusion detection which analyse network data in order to gain intrusion-related knowledge [3, 4].

IPS is classified into two categories: host-based systems and network-based systems. Network Intrusion Prevention Systems (NIPSs) base their decisions on data obtained by monitoring the traffic network to which the hosts are connected. Typically they comprise a hardware product. Usually, NIPSs are inline, and sit between the network traffic flows, between two or more network interfaces, monitoring network traffic at a collection point; they respond to an event almost immediately. The true power of NIPS is their ability to dynamically block the offending traffic [5].

NIPSs are a great way to prevent attacks from happening on the network. They check every packet that passes through them, analysing traffic for known attack patterns designed to infect, disable or take over another computer system. When a pattern is matched and the NIPS detects an attack, it takes an action in the form of an alert, log, send and reset, typically modifying firewall rules or blocking the corresponding packet stream, preventing the attack from happening again and generating a notification that leads to the prevention of successful intrusions [7, 6].

The benefit of IPS position in the line of network traffic is that can detect attacks and intrusions more accurately and reliably through less dependence on signatures and more on intelligent methods of detection, so the IPS generates far fewer false

alarms. The summery of actions taken by IPS when observes any suspicious activity: generating alarm, preventing attack activity, resetting the connection, modifying firewall rules, dropping attack pocket and allowing pass other normal packet, and logging the event activity and updating the log event database.

4 Requirements of Successful IPSs

There are some requirements commonly used when evaluating the fine tuning of IPS or any security techniques that can be further used to analyze the successfulness of IPS.

1. **Accuracy:** The most important requirement in an IPS is accuracy. Having false positives must be absolutely unacceptable in an IPS. A false positive is any legitimate traffic that will drop because it appears to be anomalous. False positives are commonly generated by security systems that depend on a single detection method, and by ones that cannot be configured at different levels to fit into the operational environment. If legitimate traffic is blocked, then problems appear for authorized users. This creates Denial of Service (DoS) attacks that originate from the prevention system itself. For example a valid business transaction may act like an attack. In such a case, this packet may first be dropped and then the entire data flow and may be the source is critical business and the recipient will be prevented from accessing resources.
2. **Performance:** The importance of IPS is importance. One of the problems with IPS is that it tends to occur a network bottleneck. Network traffic needs to flow through IPS to be analyzed and if they don't operate quickly enough, they drop packets or pass packets, increasing the possibility of false negatives. A false negative is any malicious traffic that makes it pass through the IPS to the production network. Thus, IPSs have to work equated with line of speed.
3. **Flexibility:** Ability of prediction unknown attacks and easy signature update for new attacks. An IPS system must provide flexible methods to update new attack signatures constantly, as well as these systems should have capabilities to deal with entirely new classes of attacks without depending on database signature updates. IPSs use methods such as inverse exclusion method where all given destination requests except that legal are dropped. Protocol validation method, where illegal protocol request are dropped. Another method is attack-independent blocking where hostile attackers are identified, and all traffic from the attacker is dropped, regardless of whether the attacks are known or not.
4. **Reliability and Availability:** An IPS system should be reliable and high available. Reliability refers to the ability of a system to perform its functions properly without interfering with other systems on the network also IPSs should cooperate with these systems such as firewalls, antivirus systems, etc. While the availability is the amount of downtime of the system, due to shutdown, crashes, or maintenance. An IPS gives the network security administrator many facilities; it is capable of detecting attacks and intrusions and directly affects limiting or blocking network traffic. IPSs have an easy interface for setting and changing configurations on its system.

5. **Minimize Latency:** Latency is the time it takes for a packet to pass through the IPS to the destination system and return to the user. This is typically measured in Round-Trip Time (RTT). With all the necessary time to analyze and detect the content of packet before being sent to the destination system.

5 IPSs Approaches

There are some of IPS methods being used that presented as the following:

6. **Protocol Anomaly Detection is used to ensure that packets meet to the protocol requirements and have no ambiguities.** Protocols should be well defined, this lead to high accuracy detection of the deviations from the protocol standard. For example, by IPS spoofing of FTP PORT commands, the attacker can tell the FTP server to open a connection to a victim's IP address and then transfer a Trojan Horse to the victim. Checking for a match between the IP address in the FTP PORT command and the client's IP address can prevent this anomaly.
7. **Traffic Anomaly Detection** is operating on the basis of deviations from expected behaviour. Attackers often use a port or network scan as a precursor to an attack and the scanning techniques that used by attackers have made it possible that worms can affect the entire vulnerable of system in 10s of seconds or less, so fast that no traditional Anti-attack response is possible. Network Intrusion Prevention system implement throughput and threshold triggers that alert to such scanning activity, increasing the possibility that prevented an attack.
8. **State-based Signature Detection** is based on the context specified by the user, looks at related portions of traffic by tracking state, to detect attacks. It is not completely automated as the user needs to have previous knowledge about the attack. For example the love Letter worm can be detected by a rule that would read as follows: "LOOK for 'I LOVE YOU' in the subject field only, ignore this string anywhere else in the email". Of course false positives can be generated in this case, since harmless emails with the same title may have been sent.
9. **Pattern matching using regular expressions** use to detect attack patterns that are slightly different from the fixed ones because the simply change like a space or a tab in the attack code could be enough to avoid detection. So regular expressions provide wild-card and complex pattern matching, and are able to prevent attacks.
10. **Signature detection** is used in cooperation with the above mentioned techniques to prevent combined attack types seen on today's networks.
11. **Hybrid approch** typically used in NIPS, is use various detection methods, including protocol anomaly, traffic anomaly, and signature detection work together to determine an imminent attack and block traffic coming from an inline router.
12. **Software-based heuristics** this approach usually using on HIPS is similar to anomaly detection system using neural networks to act against new or unknown types of intrusion.

13. **Sandbox approach** use on HIPS is a mobile code like ActiveX, Java applets or any scripting language is quarantined in a sandbox, an area with restricted access to the rest of the system. This system then runs the suspect mobile code in the sandbox and monitors its behaviour. If the code not meets a predefined security policy, it is stopped and prevented from executing.
14. **Kernel-based protection** typically used in HIPS. Kernel based IPS prevents execution of malicious system calls. The kernel controls access to system resources like memory, input/output devices and CPU. Programming code errors enable exploits as buffer-overflow attacks to overwrite kernel memory space and crash or take over computer systems. To prevent these types of attacks, a software agent is loaded between the user application and the kernel. The software agent intercepts system calls to the kernel, inspects them against an access control list defined by a policy and then either allows or denies access to resources.
15. **Address space randomization** is a technique used to fortify systems against buffer overflow attacks. The idea is to introduce artificial diversity by randomizing the memory location of certain system components, and checking whether the code about to be executed by the operating system came from a normal application or an overflowed buffer, these attacks can be stopped.
16. **Protecting System Resources** used to prevent alteration of system resources by hacking tools such as Trojan Horses, root kits, and backdoors and can change in system resources like libraries, files/directories, registry settings, and user accounts. This system disallows install hacking tools.
17. **Stopping Privilege Escalation Exploits** privilege escalation attacks try to take ordinary users root or administrator privileges. This type can prevent change privilege levels, disallowing attacks access to resources, and block exploits.
18. **Prohibit Access to E-mail Contact List** many worms spread by mailing a copy to those in the Outlook's contact list. This approach could be preventing these worms by prohibiting e-mail attachments from accessing Outlook's contact list.
19. **Prevent directory traversal** an approach that would prevent the hacker access to the web server files outside its normal range could prevent malicious activities. Sometimes the directory traversal has vulnerability in different web servers that allow the hacker to access files outside the web servers range.

6 Data Mining and Intrusion Prevention System

Data Mining is the process of examining data to uncover patterns and deviations as well as determining any changes or events that have taken place within the data structure. It can improve Host and network Intrusion detection system by adding a new level of observation to detection of network data in differences and identifying the boundaries for usual network activity so it can distinguish common activities from uncommon activities.

Data Mining involves the use of data analysis to discover previously unknown, valid patterns and relationships in large data sets. These tools can include statistical models, mathematical algorithms, and machine learning methods. Data Mining consists of more

than collecting and managing data; it also includes analysis and prediction. Recently Data Mining started to enter information security field such as Intrusion detection/prevention. Many applications of Data Mining techniques demonstrate that information security became an interesting field for researchers who have the tendency to discover new methods to identify the most frequent issues regarding data security in networks. In these researches, techniques such decision trees, association rules, clustering and others are well known, and used for vulnerabilities detection.

Data Mining improves intrusion detection/prevention system using a variety of different methods:

1. **Code Variants:** Data Mining is based on the process of scanning for abnormal activity through code variants instead of unique signatures. For example, a buffer overflow code has been changed would be considered a fraud by attempting to escape an intrusion detection system that uses signatures.
2. **Data Reduction:** Data Mining can significantly reduce data overload through its capability to extract specific amounts of data for identification and analysis. This helps the system to determine which data is most relevant time and processing.
3. **Filter out Valid Network Activity:** Data Mining is used to help intrusion detection by being able to better identify valid network activity so it can filter it out to make detection of abnormal activity in data easier.
4. **Attacks without Signatures:** since Data Mining is not signature-based like intrusion detection, it is more efficient in detecting abnormalities that do not contain signatures. If network activity contains a specific profile and rules of protocol, an abnormality is easily detected and can be extended to individual hosts, entire networks, specific users, and overall traffic patterns on the network at specific times.

Data Mining is powerful assisting for most applications that required data analysis. Recently, Data Mining is becoming an important component in intrusion detection/prevention system. Data Mining could contribute to the enhancement of the applications of network intrusion detection/prevention systems, Data Mining use one or more techniques are used in the context of intrusion detection which analyze network data to gain intrusion related knowledge, such as:

1. Data summarization with statistics, including finding outliers that lead to finding anomalies activity that discovers a real attack.
2. Clustering, including segmentation of the data into natural categories that lead to identifying different IP address has same activity, this ongoing pattern can be a type of attack.
3. Association of rule discovery, including defining normal activity and enabling the discovery of anomalies that help to separate normal activity from suspicion data to allow focus on real attacks.
4. Classification, including predicting the category to which a particular record belongs data, this identify which data generate alarm and attack signatures.

Usually using Data Mining techniques to analysis of collected data in an offline database, this important in performing Network Intrusion Prevention Systems (NIPS)

because all connections have already finished therefore these techniques can process and check all features without drop packets when flooded with data became faster than process, as well as offline database provides the ability to transfer data from multiple hosts to central host for analysis, detection and prevention that a way to increase the performance and accuracy of Network Intrusion Prevention Systems (NIPSs).

6.1 Clustering

Human labelling of network audit data instances or even used traditional methods are time-consuming and expensive because these amounts of available data is huge therefore beings used clustering approach which is a technique for statistical data analysis used in many fields such as machine learning and Data Mining. Clustering is the process of labelling data and assigning it into groups. Clustering algorithms can be partition the data set into subsets or clusters; so that the objects in each cluster share some common feature often proximity according to some defined distance measure. Clustering techniques can be categorized into:

- Pair wise clustering, pir wise clustering unifies similar data instances based on a data-pair wise distance measure.
- Central clustering classes, while central clustering that also called centroid-based or model-based clustering, models each cluster by its "centroid", and more efficient than pairwise clustering algorithms.

Clustering is used to detect attack in any cluster that modelled according to predefined metrics and common features of sets data belonging to this cluster by discovering complex intrusions occurred over extended periods of time and different spaces, correlating independent network events and in another mean the clustering is useful in intrusion detection as attack activity should cluster together, separating it from normal activity.

One of the common clustering techniques is K-means clustering which used to find natural groupings of similar alarm records; this depends on the records that are far from any of these clusters indicate unusual activity that may be part of a new attack.

Most of the clustering techniques are the basic steps involved in identifying intrusion. These steps are as follows:

*Find the largest cluster, i.*e., the one with the most number of instances, and label it normal.

Sort the remaining clusters in an ascending order of their distances to the largest cluster.

Select the first K1 "No. Of clusters" so that the number of data instances in these clusters sum up to1/4 'N, and label them as normal, where 'N is the percentage of normal instances.

Label all the other clusters as attacks.

6.2 Classification

Classification is similar to clustering in that it also partitions data records into distinct segments called classes. But it differs from clustering, classification require more because it need also labeling data set for training stage and classification is much less exploratory than clustering because the end-user decides on the attribute to use define the classes and each record has a value for these attribute, so classification is not used to explore the data or approximate its values to discover interesting segments but to assign new data has a specific value to predefined categories or classes.
A classification based IDS even IPS attempts to classify all traffic as either normal or abnormal class, this techniques has been popular to detect individual attacks, but has been suffered the problem of high false positives and false negatives rate so begin applied with complementary fine-tuning techniques to reduce its troubles.
Classifications algorithms can be classified into three types:

- Extensions to linear discrimination (e.g., multilayer perception, logistic discrimination).
- Decision tree and rule-based methods (e.g., C4.5, AQ, CART).
- Density estimators (Naive Bayes, Multi-Bayes, K-nearest neighbour, LVQ (Learning Vector Quantization), SOM (Self Organizing Maps)).

Data classification for intrusion detection can be achieved by the following basic steps:

1. First to learn he classification models of the normal and abnormal system call sequences, it needs to supply it with a set of training data containing pre-labelled normal and abnormal sequences. The mechanism models based on any type of classification algorithms, all these can be used to scan the normal network paths and create a list of unique sequences of system calls. This list is generally named as normal list.
2. Next the second step is to scan each of the intrusion paths. For each sequence of system class, first looks it up in the normal list. If an exact match can be found then the sequence is labelled as normal, otherwise it is labelled as abnormal.
3. Finally must ensure that the normal paths include nearly all possible normal short sequences of system calls, because an Intrusion Path contains many normal sequences in addition to the abnormal sequences since the illegal activities only occur in some places within a network path.

Classification technique in the domain of intrusion detection or prevention system needs the large amount of data needed to be collected to apply classification. To build the traces and form the normal and abnormal groups, significant amount of data need to be analyzed to ensure its proximity. Using the collected data as empirical models, false alarm rate in such case is significantly lower when compared to clustering.

In intrusion detection, Data Mining classification can be applied to a standard set of malicious virus and benign executable using derived features, then classification

approach can be useful for both misuse detection and anomaly detection, but it is more commonly used for misuse detection.

Thus the various classification approaches can be employed on network data for obtaining specific information and detecting intrusion and then prevention, for example the Naive Bayes and multi-Bayes classifiers can be used to detect malicious virus code. While the decision Tree can be exploited to formulate genetic algorithm to create rules that match a set of anomalous connection. Nearest neighbour classifier approaches based on SOM and LVQ can be used to refine the collected network data in intrusion detection.

6.3 Association Rule

The association rule is particularly designed using in data analysis. Association rule mining finds interesting association or correlation relationships among huge set of data items. Association rule shows attribute value conditions that occur frequently together in a given dataset. The association rule considers each pair (attribute/value) as in item. In each single network request an item set is a combination of items. The algorithm scans through the dataset trying to find item sets that tend to appear in many network data. The objective behind using association rule based Data Mining is to derive multi-feature (attribute) correlations from database table.

Association rules construct information or rules in the form of "if-then" statements. Association rules are probabilistic in nature. In addition to the antecedent (the "if" part) and the consequent (the "then" part), an association rule has two numbers that express the degree of uncertainty about the rule. In association analysis the antecedent and consequent are sets of items that are disjoint. The first number is called the support for the rule. The support is simply the number of transactions that include all items in the antecedent and consequent parts of the rule. The other number is known as the confidence of the rule. Confidence is the ratio of the number of transactions that include all items in the consequent as well as the antecedent to the number of transactions that include all items in the antecedent.
Many association rule algorithms can be classified into two categories:

- Candidate-generation-and-test approach such as Apriori.
- Pattern-growth approach.

Association rule algorithms are multiple scans of transaction database and a large number of candidates therefore became use association rule in analyzing network data in intrusion detection. Basic steps for integrating association rule for intrusion detection as follows:

- First network data have to be constructed into a database table where each row is an attribute record and each column is a value field of the attribute records.
- There is index that intrusions and user activities shows frequent correlations among network data. For example, "program policies", which modify the access rights of privileged programs, are concise and capable to detect known attacks is in that the intended behaviour of a program, e.g., read and write files from certain directories

with specific permissions is very consistent. These consistent behaviours can be captured in association rules.

- With the association rule, rules based on network data can continuously merge the rules from a new run to the aggregate rule set of all previous runs, and then can get the capability to capture behaviour in association rule for correctly detecting intrusion and hence lowering the false alarm rate.

6.4 Outlier Detection

An outlier is an uncommon observation that significantly deviates from the characteristic distribution of other observations. The value of outliers indicates that individuals or groups that have very different behaviour from most of the individuals of the dataset. Many times, outliers are removed to improve accuracy of the estimators. Outlier detection has many applications, such as data cleaning, fraud detection and network intrusion.

Anomaly detection algorithms require a set of purely normal data to train the model. Assume that anomalies can be treated as previously unobserved patterns. Since an outlier may be defined as a data point which is very different from the rest of the data, than can employ several outlier detection schemes for intrusion detection which are based on statistical measures, clustering methods and Data Mining methods.

Commonly used outlier techniques in intrusion detection are Mahalanobis distance, detection of outliers using Partitioning Around Medias (PAM), any Bay's algorithm for distance-based outliers. Outlier detection is very useful in anomaly based intrusion detection. With outlier detection approach, can detect novel attack/intrusion by identifying them as deviation from normal behaviour. The basic steps in detecting intrusion based on outlier detection are as follows:

1. As outlier detection technique is used in anomaly detection, first step have to identify normal behaviour. This behaviour can be data set or pattern of some events on the network.
2. Then useful set of feature need to be constructed.
3. And similarity function needs to be defined between them.
4. Also will need to run specific outlier detection algorithm on the set of feature. The algorithm can be based on a statistical based approach, a distance based approach, or a model based schema. All these approaches are based on finding the deviation between collected and scanned data sets.
5. In case of intrusion detection, the collected set of data set will be the set of events and their relation to intrusion. Such relation can be calculated based on normal behaviour and any other behaviour which significantly deviates from normal behaviour. As with such deviation we can pre-empt attacks based on their behavioural deviation. Outlier detection approaches can useful for detecting any unknown attacks.

This is the primary reason that makes outlier detection a popular approach for intrusion detection systems. Statistical based outlier detection scheme uses a probabilistic model as representation of underlying mechanism of data generation. Such

probabilistic model can be useful in intrusion detection environment to decide the probability before alarming the system for intrusion.

So the outlier detection is very useful in anomaly based intrusion detection systems that are involved in detecting abnormal behaviour or deviating patterns. It can help to identify abnormal behaviour from the set of normal behaviour and enable to detect any unknown intrusions.

7 Intrusion Prevention System Using Data Mining Tools

This chapter will use one of the IPS systems that uses rule- (signature)-based prevention integrated with Data Mining, which can detect and predict anomaly attacks in addition to having the capacity and effectiveness of Data Mining for dealing with the huge quantity of information stored in the network database and high-speed processing to search for and extract the appropriate decision.

The system consists of four main phases, as described in Figure 1. Each phase performs a certain task. These phases integrate with each other, eventually to manufacture the complete system. The appropriate technology was used in each phase when building the system in order to meet its requirements.

1. The sniffer phase: The first phase starts with a packet sniffer that uses network sniffer tools, which capture packets from the network with different levels of detail and display them on the control unit in the next phase in order to examine data from a live network or from a capture file on a disk.
2. The NIPS phase: This phase includes several actions; it starts with analysis of the captured packet, then takes the appropriate decision, such as passing or dropping, and logs it into a file (storing the log file in an offline DB) based on a set of rules defined in the Snort tool.
3. Advanced analysis phase: This phase generates new rules which help to improve defined rules that were used in the previous phase. It consists of two sub-phases of the Data Mining approach.
4. Control Analyst Interface: This final phase is used to choose recent rules that enable anomaly detection. It also takes suitable decisions about the new rules, which lie in the middle of clustering. In addition to updating the proposal, a smart new rule file will be searched for in case normal traffic has still not resumed or there is suspicion about the appropriate actions.

8 Methodology

8.1 Implementation of Sniffer Phase

In the proposed system, the Winpcap tool [11] was used for NIPS; this is represented by the Snort application that gives instructions to the Winpcap tool to start capturing data. According to a selected command line of the Snort tool, the response from the Winpcap tool will be based on the command used.

Implementation NIPS Phase

The implementation of NIPS using the Snort [12] tool, here an open-source network intrusion prevention and detection system uses both a rule-based signature and anomaly inspection methods.

After the Snort tool has been installed, then the "Snort.Conf" file, which comprises configuration files of the Snort tool, must be modified to suit the requirements of the proposed system and the tools used with it. The "Snort.Conf" file is activated by the selected rules file, and this will apply the activation rules that are selected in the "Snort.Conf" file to each sniffer packet in order to decide whether an action should be taken based upon the rule type in the file. In order to run the Snort tool in Network Intrusion Prevention System (NIPS) mode and logger mode, the following instructions must be followed.

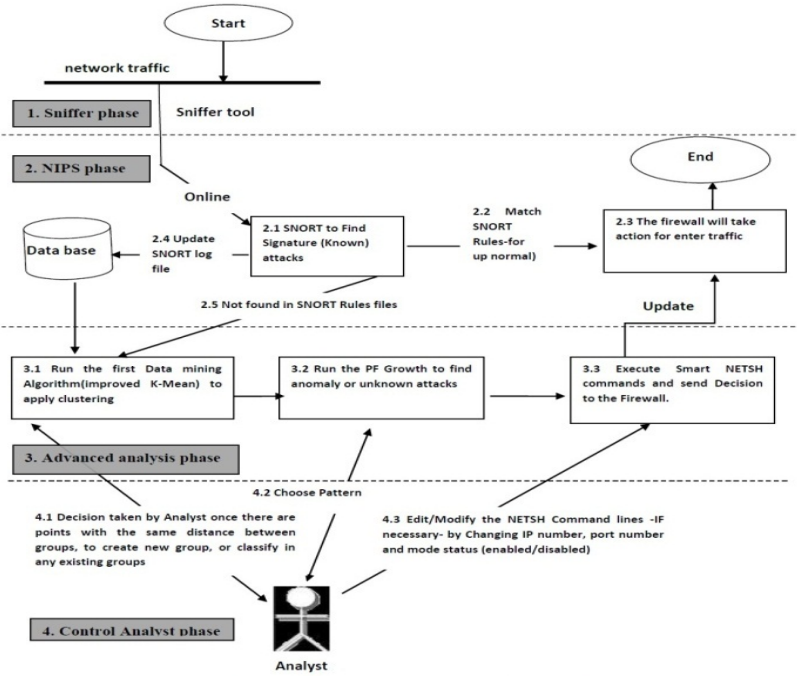


Fig. 1. Schema of the proposed system

This is done by configuring the Snort tool to run in its NIPS form, logging packets that trigger rules specified in the "Snort.Conf" file to disk. Snort runs in the logger mode, collecting every packet it sees and placing it in a directory to the disk.

Implementation of an Advanced Analysis Phase

In this phase, a deep analysis is needed for the Snort log file to make it possible to find behaviour that is hidden between the source and the destination of the connection

traffic. Reliance on human ability in the analysis of this file is not an effective method, and it additionally lacks accuracy.

This is because this file has many records, and the database is huge, requiring a long time for analysis as well as a high level of analyst skill in order to discover all the hidden behaviour [9]. The suggestion to use Data Mining algorithms, due to their effectiveness in working with a huge database, would help an analyst to discover new rules from hidden traffic behaviour between the source and the destination, which the Snort tool cannot see as obvious rules. In order to do this; two sub-phases of the Data Mining approach will be relied upon. The first sub-phase is an improved K-mean algorithm to classify the log file of the Snort tool to clusters in an effective manner. Records will be taken from the resulting clusters as an input to the second sub-phase, which is the FP-growth algorithm that will generate the frequent patterns to these records, which in turn will help to create new rules from these patterns, leading to improvements to the Snort tool. The use of Data Mining techniques to analyse the collected data in an offline database is important in performing NIPS because all connections have already finished; therefore, these techniques can process and check all features without dropping packets when flooded with data, resulting in a faster process and increasing the performance and accuracy of NIPS [8, 10].

9 Quality Evaluation

The system is a LAN network that consists of four nodes connected to one main server. The Snort tool was installed on the server in order to monitor the traffic coming in and out through the workstations. The following steps are used to test and execute the system:

1. Execute the Snort tool as an inline mode (NIPS) that sniffs network traffic and takes the appropriate decisions to pass or drop the traffic, depending on the Snort rules and the Snort alerts that are saved in the Snort log file.
2. Begin to call or enter the Snort log file to be analysed by the proposed system.

The first and second steps are illustrated in the following two cases:

The first case: 10 records were taken from the first log file to be extracted from the execution of the Snort file on 15/6/2011.

Here, the Snort tool made a decision to drop the third category of traffic, classifying it as “bad traffic” and displayed it either as an alert shown on a screen, a log file, or both. All the remaining records were set to pass. However, the system showed an alert for the second and tenth records, as these were attempts to use privileged access. The overview of Snort decisions about these transactions is passing the traffic that in general is normal, or dropping the bad traffic, depending on the Snort rules. As a result, nine types of traffic are classified as a normal case, while one type of traffic is dropped and classified as an attack case, depending on the Snort tool that analyses the content of this traffic and the ports used.

The proposed system will apply this log file to advanced analysis, with two sub-phases of the Data Mining approaches. In the first sub-phase, which is an improved K-mean algorithm, the input records were classified as follows:

Four records put in a normal cluster, based on the initial centre-point that was chosen for normal traffic; in this case K=2.

One record put in an attack cluster, based on the second initial centre-point that was chosen for the attack traffic; in this case K=2.

Three records put in the test cluster, because it was out of the normal or attack groups, so the analyst created it based on a new cluster, called 'Test', and chose record 7 as an initial centre-point; in this case K will become equal to 3.

Two records were considered to be intermediate records between the normal and attack clusters, as these comprised an attempt to use the access privileges; based on this, the analyst decided to put these two points in a new group, named the Probe cluster, and chose record 2 as an initial centre-point; in this case K will become equal to 4.

The second step is to use the FP-growth algorithm in order to select records from the four clusters mentioned above. The analyst will select records from the attack, probe and test clusters, and then execute the FP-growth algorithm, which will finally display the results in a table that has a suffix and frequent items.

Examples of frequent *items are shown below:*

IPs PORTsIPd PORTd

"98.136.154.147, 80, 192.168.0.198, 4727" frequent item appears more than twice.

"98.136.154.147, 27374, 192.168.0.198, 4727" frequent item appears once.

"98.136.154.148, 3676, 192.168.0.198, 4727" frequent item appears once.

The patterns above indicate that the IP source 98.136.154.147 uses port 80 to target IP 192.168.0.198 through port 4727; IP source 98.136.154.147 also uses port 27374 to access the same IP 192.168.0.198, 4727 and port destination. IP source 98.136.154.148 by port 3676 targets the same IP 192.168.0.198.

As can be seen, the IP source targeted the same IP destination but with two different ports, one of which was legal (port 80) and one of which was illegal, because its signature is for the Sub Seven worm, which is port 27374. Therefore, the system prevented the IP source from accessing the IP destination by using a command line firewall.

The final results were:

4 records were set in the normal cluster.

1 record was set in the attack cluster.

3 records were set in the test cluster.

2 records were set in the probe cluster.

Eventually, using these two sub-phases of Data Mining made it possible to discover rules that the Snort was unable to detect.

The second case: Several records that were taken from the other log file were the result of executing the Snort on a different date. They were generated by using the

ping Dos command line utility, which can help to determine whether or not a particular network resource is responding on a network. This kind of test was conducted on IP 192.168.0.198 and 192.168.0.195, targeting the destination IP 192.168.0.196 on a different date but at approximately the same time. Here, the Snort tool logs in the file at the IP destination 192.168.0.196 and replies to an echo ping request from IP source 192.168.0.198 or 192.168.0.195. However, the Snort tool does not send an alert about this frequent testing.

The proposed system will use this log file in the two sub-phases of the Data Mining approach for performing advanced analysis. In the first sub-phase, which is an improved K-mean algorithm, the input records classified most of the records and put them in the probe cluster, except for the records that have IP 192.168.0.195, which were put in the test cluster that was previously defined by the analyst in the first case.Now, using the second sub-phase FP-growth algorithm, the recent records of a probe and a test cluster can be selected. The execution FP-growth algorithm will appear as a frequent item in the final results table. These frequent items will be as follows:

IPs IPd

"192.168.0.195, 192.168.0.198" frequent item appears three times in this test; "192.168.0.198, 192.168.0.196" frequent item appears six times; "192.168.0.196 and 192.168.0.195" frequent item appears three times; "192.168.0.195" frequent item appears three times; "192.168.0.198" frequent item appears ten times; and "192.168.0.196" frequent item appears seven times.

The patterns above indicate that IP source 192.168.0.195 is related to IP source 192.168.0.198, because IP sources 168.0.198 and 192.168.0.195 access the same destination IP, 192.168.0.196. As a result, the Data Mining can discover these frequent tests and show a hidden relationship between "192.168.0.195 and 192.168.0.198", which may originate from the same source as the malicious user and the target IP destination, 192.168.0.196.

10 Conclusion

The improvement of the work of the Snort tool integrated with Data Mining methods in an implementation mechanism of an anomaly- and signature-based prevention system was effective, and the installation of this system worked as an internal gateway for the network; this system will contribute to significant improvements as follows:

1. The improved NIPS, based on both the misuse and anomaly-detection approach, can expose such an individual and result in the prevention of more damage being done. Cooperation between Data Mining and Snort helps to reduce the analysis effort of the NIPS when a high level network is recording availability. This will lead to the achievement of minimized latency. The Data Mining approach is complementary to the Snort tool, and will help to increase the overall attack coverage and

particularly insider attack. The use of the centralized database is to store all the network records and analyse them by using Data Mining sub-phases. If there is a new attack or any internal threat has been detected, the attack file in the centralized database will be updated.

2. The anomaly-detection approach represented by two Data Mining sub-phases is effective for detecting many insider attacks, where authorised users attempt to access the source with the aim of mimicking an attack. The malicious behavior shown by such a user often differs from normal behavior, and can therefore be identified as anomalous behavior. As no security mechanism is fully guaranteed, and undetected successful insider attacks create the equivalent of outsider attacks, this is dangerous. Using the proposed approach to analyse and detect insider attacks will increase a power of security from inside the network, securely guarding networks both from within and externally.
3. Using two sequential analysis stages that are represented by two Data Mining sub-phases will achieve increases in the accuracy of the analysis and reduce the effort for human analysts and NIPS. In addition, this effectively helps in the detection of hidden malicious paths between the source and the destination. Therefore, this will achieve a reduced rate of false-positive and false-negative alarms. Because the primary goal is to discover the process of suspicious reconnaissance within the network before launching any network attacks, with devastating results, the focus was on the use of the IP source, port source, IP destination and port destination as the main attributes of the network records. Additionally, using these specific attributes leads to accuracy in the results, minimizes the consumption and process time of the analysis, and reduces overload, even when huge network records are available for analysis.

References

1. Kizza, J.M.: A Guide to Computer Network Security. In: Computer Communications and Networks, ch. 13. System Intrusion Detection and Prevention. Springer-Verlag London Limited (2009)
2. Tamagna-Darr, L.: Evaluating the Effectiveness of an Intrusion Prevention / Honeypot Hybrid, Masters thesis, Rochester Institute of Technology, B. Thomas Golisano College of Computing and Information Sciences, Department of Network Security and Systems Administration (August 2009)
3. Al-Hamami Alaa, H.: Data Mining: concepts, techniques and application. Ithraa Publishing and Distribution, Amman (2008)
4. Brugger Terry, S.: Data Mining Methods for Network Intrusion Detection, PhD thesis, University of California Davis (2004)
5. Zois, C., Bos, H.: Intrusion Prevention System, Vrije Universities (2006)
6. Sequeira, D.: Intrusion Prevention Systems- Security-Silver Bullet?, GSEC Version 1.4B,OPTION 1, SANS Institute, Reading Room site (2002)
7. Andres, S.K., Andrés, B.: Security Sage's Guide to Hardening the Network Infrastructure, Understanding Intrusion Detection and Prevention Basics. Syngress Publishing, Rockland (2004)

8. How Data Mining is Used for Intrusion Detection, spam laws (2010), accessed from: http://www.spamlaws.com/how-data-mining-helps-intrusion-detection.html (last accessed: April 1, 2010)
9. Siraj Ambareen, B., Rayford, V., Bridges, S.M.: Intrusion Sensor Data Fusion in an Intelligent Intrusion Detection System Architecture (2004)
10. Bringas, P.G., Penya, Y.K.: Next-Generation Misuse and Anomaly Prevention System. In: Filipe, J., Cordeiro, J. (eds.) Enterprise Information Systems. LNBIP, vol. 19, pp. 117–129. Springer, Heidelberg (2009)
11. WinPcap, accessed from: http://www.winpcap.org/ (last accessed May 3, 2011)
12. Gaur, N.: Snort: Planning IDS for your enterprise. Linux Journal (July 11, 2011), accessed from: http://www.linuxjournal.com/article/4668?page=0,0 (last accessed May 3, 2011)

Two Types of Deadlock Detection: Cyclic and Acyclic

Takao Shimomura and Kenji Ikeda

University of Tokushima, Tokushima, Japan
{simomura,ikeda}@is.tokushima-u.ac.jp

Abstract. Concurrent programs are difficult to test and debug due to their non-deterministic execution. For deadlocks, traditional deadlock detection algorithms depend on finding cycles in lock graphs created from application programs. This paper first introduces three kinds of blocked relations, lock-blocked, wait-blocked, and join-blocked for Java multi-threaded programs. Previous work does not consider the wait-blocked relations, nor the influence of thread interruption. The paper then proposes two types of deadlocks based on these blocked relations, that is, block-cycle type deadlocks and waiting-block type deadlocks which are acyclic. It also presents an example of implementation to detect these types of deadlocks, and addresses future directions.

Keywords: Acyclic, blocked relations, deadlock, lock table, synchronization.

1 Introduction

Unlike sequential programs, concurrent programs are difficult to design, make, test, and debug due to their non-deterministic execution. To help understanding concurrency-related errors and developing techniques for detecting them, Fiedor, J. et al. [12] provide a uniform taxonomy of concurrency errors common in concurrent programs such as data races, atomicity violation, order violations, deadlocks, missed signals, non-progress behavior, live locks and blocked threads. Bradbury, J.S. et al. [5] propose thirteen programming anti-patterns for concurrent Java such as deadlock, starvation, resource exhaustion, and missing or nonexistent signal anti-patterns, and implement a clone-based detection tool for detecting concurrency anti-patterns in Java programs. For each of 105 randomly selected real world concurrency bugs from 4 representative server and client open source applications, MySQL, Apache, Mozilla and OpenOffice, Lu, S. et al. [26] examine its bug report, corresponding source code, related patches, and programmers' discussion to provide concurrency bug patterns, manifestation, and fix strategies. ConMem [46] monitors program execution, analyzes memory accesses and synchronizations, and detects severe concurrency-memory bugs that can lead to program crashes in C/C++ applications. UNICORN [37] monitors pairs of memory accesses, combines the pairs into problematic patterns, and ranks the patterns by their suspiciousness scores. It detects non-deadlock concurrency bugs in Java and C++

L. Chen, S. Kapoor, and R. Bhatia (eds.), *Intelligent Systems for Science and Information,* Studies in Computational Intelligence 542,
DOI: 10.1007/978-3-319-04702-7_14,

such as order violations and both single-variable and multi-variable atomicity violations. Checkmate [11] is a static analyzer of multi-threaded Java programs to detect data races. El-Zawawy, M.A. et al. [10] propose a static detector for data-race problems in programs written in a simple language, called m-while. A deadlock is a common form of bugs. Sun's bug database [32] shows that 6,500 bug reports out of 198,000 contain the keyword "deadlock" [20]. As for deadlocks, for example, Nir-Buchbinder, Y. et al. [30] investigate how to exhibit potential deadlocks and how to heal multi-threaded programs so that they will not get into deadlocks. Traditional deadlock detection algorithms depend on finding cycles in lock graphs created from application programs.

Here are some examples of typical deadlocks. When a thread A possesses a lock for object **a** and a thread B possesses a lock for object **b**, if thread A performs a lock action on **b** and thread B performs a lock action on **a**, they will deadlock. For another example, when a thread A possesses a lock for object **a** and invokes B.**join**() to wait for a thread B to terminate, if thread B performs a lock action on **a** and no threads invoke A.**interrupt**(), then threads A and B will also deadlock. Conventional techniques need to find cycles in the blocked relations of threads to detect deadlocks. Therefore, they cannot detect deadlocks caused by a blocked thread tree whose root thread is waiting. For example, when a thread A invokes **a.wait**() and then a thread B locks object **a** and invokes **b.wait**(), if no threads invoke **b.notify**() or B.**interrupt**(), threads A and B will deadlock. In this situation, even if another thread invokes **a.notify**() or A.**interrupt**(), thread A cannot resume as explained in detail later. This type of deadlock cannot be detected by previous work. For another example of the deadlock the previous work cannot detect, when a thread A possesses a lock for objects **a** and **b** and invokes **b.wait**() and then a thread B locks object **b** and performs a lock action on **a**, even if another thread invokes **b.notify**(), threads A and B will deadlock. In this situation, a block cycle that consists of threads A and B has been formed.

This paper introduces three kinds of blocked relations, lock-blocked, wait-blocked, and join-blocked for Java multi-threaded programs. Previous work uses lock graphs and mainly addresses lock-blocked relations, and it does not take into consideration wait-blocked and join-blocked relations [16], [18], [15], [27], [38], [7], [2], [44], [9], [21]. Although lock-blocked and join-blocked relations are addressed in [31] [19], they do not consider wait-blocked relations, nor the influence of thread interruption. On the other hand, based on these three kinds of blocked relations, this paper defines block graphs and then proposes two types of deadlocks. One is block-cycle type deadlocks, and the other is acyclic deadlocks, that is, waiting-block type deadlocks. Deadlocks are defined as follows in [12]:

> *A program state contains a set S of deadlocked threads iff each thread in S is blocked and waiting for some event that could unblock it, but such an event could only be generated by a thread from S.*

The two types of deadlocks this paper proposes satisfy this definition even though waiting-block type deadlocks are acyclic. The paper also describes an example of implementation to detect these types of deadlocks, and addresses future directions.

The remainder of this paper is organized as follows: Section 2 briefly describes multi-threaded programming with Java. Section 3 gives the definition of block graphs this paper proposes and classification of deadlocks. Section 4 illustrates the deadlock detection algorithm. Section 5 demonstrates some examples of deadlock detection. Section 6 discusses the evaluation of the deadlock detection this paper presents. Section 7 describes the related work. Finally, Section 8 summarizes the paper.

2 Java Multi-threaded Programming

This section briefly describes multi-threaded programming with Java [33]. The Java Virtual Machine (JVM) allows an application to have multiple threads of execution running concurrently.

2.1 Creation of Threads

There are two ways to create a new thread of execution. One is to declare a class to be a subclass of Thread. This subclass should override the **run**() method of class Thread. An instance of the subclass can then be allocated and started as follows:

```
SubclassOfThread T =  new SubclassOfThread();
T.start();
```

The other way to create a thread is to declare a class that implements the Runnable interface. That class then implements the **run**() method. An instance of the class can then be allocated, passed as an argument when creating Thread, and started.

```
Thread T = new Thread(new ClassImplementsRunnable());
T.start();
```

In both cases, the **run**() method will be executed as a new thread.

2.2 Synchronization

Synchronization is implemented using monitors. Each object in Java is associated with a monitor, which a thread can lock or unlock. Only one thread at a time may hold a lock on a monitor. Any other threads attempting to lock that monitor are blocked until they can obtain a lock on that monitor.

A synchronized statement "**synchronized** (**a**) { ... }" attempts to perform a lock action on that object (**a**)'s monitor and does not proceed further until the lock action has successfully completed. After the lock action has been performed, the body of the synchronized statement is executed. If execution of the body is ever completed, either normally or abruptly, an unlock action is automatically performed on that same monitor. The invocation of a synchronized method "**synchronized** m() { ... }" is equivalent to the invocation of "m() { **synchronized** (this) { ... } }".

2.3 Wait and Notification

Every object, in addition to having an associated monitor, has an associated wait set that consists of threads. When a thread T possesses a lock for object **a**, a wait action occurs upon invocation of **a.wait**() method on object **a**. Thread T is added to the wait set of object **a**, and performs an unlock action on **a**. Thread T does not execute any further instructions until it has been removed from **a**'s wait set. The thread may be removed from the wait set due to any one of the following actions:

1. A notify action being performed on **a** in which T is selected for removal from the wait set.
2. A notifyAll action being performed on **a**.
3. An interrupt action being performed on T.

Then, thread T performs a lock action on **a** to resume.

When a thread possesses a lock for object **a**, a notification action occurs upon invocation of **a.notify**() and **a.notifyAll**() method on object **a**. If **a**'s wait set is not empty in the invocation of **a.notify**(), a thread that is a member of **a**'s current wait set is selected and removed from the wait set. In the invocation of **a.notifyAll**(), all threads are removed from **a**'s wait set.

An interruption action occurs upon invocation of T.**interrupt**() method for a thread T. If thread T was removed from **a**'s wait set due to an interrupt, the **a.wait**() method throws InterruptedException. It should be noted that this exception will be effective after the thread acquires a lock for object **a** as described in Section 3.1 (2).

3 Block Graphs

3.1 Timing at Which Threads are Blocked

This paper presents three types of thread blocked situations, lock-blocked, wait-blocked and join-blocked. A thread is blocked by another thread in one of these situations. Figure 1 illustrates the moments at which threads are blocked in these three situations.

(1) Lock blocked
When a thread A executes a synchronized statement "**synchronized** (**a**) { ... }", thread A will be blocked if another thread B possesses a lock for object **a**. This blocked relation is denoted by "B <— lock(a) blocked A" as shown in Fig. 1 (a).

(2) Wait blocked
After a thread A invokes **a.wait**(), thread A will be blocked if another thread B locks the monitor of object **a** (hereafter, "locks object **a**", in short) as shown in Fig. 1 (b). Thread A cannot resume even if it is notified or interrupted as explained in Section 2.3. This blocked relation is denoted by "B <— wait(a)

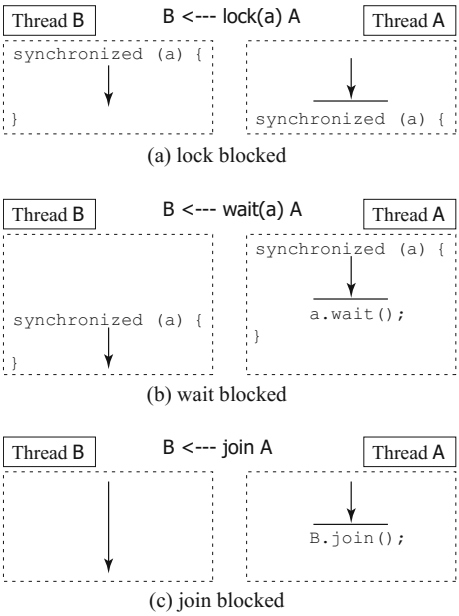


Fig. 1. Blocked moments and their blocked types

blocked A". This relation is not formed when thread A invokes **a**.**wait**() because it possesses a lock for object **a** and then performs an unlock action on object **a**. This wait-blocked relation is not addressed in previous work.

Let's consider the following example, where the current thread T is waiting by invoking **obj.wait**():

```
try {
    synchronized (obj) {
        while (!condition) {
            obj.wait();
            System.out.println("Notified");
        }
    }
} catch (InterruptedException e) {
    System.out.println("Interrupted");
}
```

If another thread invokes **obj.notify**(), this thread will perform a lock action on **obj** and then output "Notified". If another thread invokes T.**interrupt**() for this thread T, this thread will perform a lock action on **obj** and then output "Interrupted". However, this thread cannot resume until it can acquire a lock for object **obj**. Therefore, if this thread is wait-blocked, it cannot resume even if **obj.notify**() or T.**interrupt**() is invoked.

(3) Join blocked
When a thread A invokes B.**join**() method for a thread B, thread A will wait for thread B to die. Thread B will die either by returning from the call to the **run**() method or by throwing an exception that propagates beyond the **run**() method. This blocked relation is denoted by "B <— join blocked A" as shown in Fig. 1 (c). If some thread interrupts the current thread A, InterruptedException will be thrown.

3.2 Definition of a Block Graph

Definition 1 (Block graph). *A block graph BG = (V, E) is a directed graph, where V is a set of threads and $E \subseteq V \times V$, an element of which represents a blocked relation described in Section 3.1. An arc (A, B) represents one of the relations "B <— blocked A".*

Most previous work calls this kind of graph a lock graph. However, in this paper, it is called a block graph because these relations are not only formed by synchronization locks but also formed by wait-blocked and join-blocked relations. Some work addresses join-blocked relations, but it does not consider the influence of interruption.

Definition 2 (Block cycle). *A block cycle BC of block graph BG is a set of threads, which forms a cycle in the graph.*

Definition 3 (Block tree). *When a thread is blocked, there exists only one thread that blocks the thread. Therefore, if a connected component in the block graph does not have a cycle, it forms a directed tree. This is called a block tree.*

Definition 4 (Waiting block tree). *If the root thread of a block tree is waiting, this block tree is called a waiting block tree. The root thread of a waiting block tree is a waiting thread that is not wait blocked by any thread.*

3.3 Properties of Block Cycles

Theorem 1 (BCDL1). *If a block cycle is formed only by lock-blocked relations and wait-blocked relations, it will deadlock.*

Proof. Even if a thread in the block cycle is notified or interrupted, it cannot resume as described in Section 2.3 and Section 3.1 (2).

Theorem 2 (BCDL2). *If join-blocked threads in a block cycle BC are not interrupted by any threads outside BC, the block cycle BC will deadlock.*

Proof. If a thread is interrupted when it is join blocked in the block cycle, it will resume. However, it cannot be interrupted by any thread.

It should be noted that a block cycle will not always deadlock if a join-blocked thread in it is interrupted by another thread.

3.4 Properties of Waiting Block Trees

Lemma 1 (WBT1). *If the root thread of a waiting block tree invokes* ***a.wait****(), there exist no threads that are wait(a) blocked.*

Proof. Suppose that there exists a thread B that is wait(a) blocked by X. The root thread will also be wait(a) blocked by X. This contradicts the prerequisite condition that the root thread is not blocked.

Lemma 2 (WBT2). *There may exist multiple waiting block trees the root threads of which invoke* ***a.wait****() for the same object* ***a****.*

Proof. If a thread invokes **a**.**wait**() after another thread invokes **a**.**wait**(), there will exist two waiting block trees the root threads of which invoke **a**.**wait**().

Lemma 3 (WBT3). *There may exist a thread in a waiting block tree that is wait blocked.*

Proof. When a thread A locks **b** and invokes **b.wait**(), if a thread B locks **b** and invokes **a.wait**(), thread A will be wait(b) blocked by thread B. Blocked relation "B <— wait(b) blocked A" will be formed. Because thread B is waiting, threads A and B form a waiting block tree. Thread B is a root thread of this waiting block tree, and thread A is a wait-blocked thread in this waiting block tree.

Theorem 3 (WBDL). *For a set of waiting block trees, if all the root threads of the waiting block trees are not notified or interrupted and if join-blocked threads in the waiting block trees are not interrupted, then they will deadlock.*

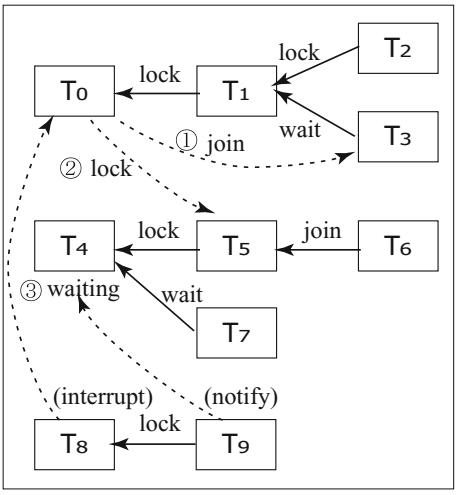


Fig. 2. A block graph

Proof. Suppose that a thread A in those waiting block trees will resume. If thread A is the root thread of one of those waiting block trees, thread A must be notified or interrupted by another thread. This contradicts the prerequisite condition. If thread A is join blocked by thread B, thread A must be interrupted by another thread or thread B must die. This contradicts the prerequisite condition.

Figure 2 shows an example of blocked relations between threads. There are ten threads from T0 to T9. The arcs in the block graph indicate blocked relations such as lock-blocked, wait-blocked, and join-blocked. Here exist three block trees BT1, BT2, and BT3 the root threads of which are T_0, T_4 and T_8, respectively.

(1) If thread T_0 invokes T_3.**join**(), blocked relation "T_3 <— join blocked T_0" will be formed. Then a block cycle BC will be formed by threads T_0, T_1, and T_3. If a join-blocked thread T_0 in BC is not interrupted by any threads outside BC, BC will deadlock.

(2) If thread T_0 performs a lock action on an object **a** for which thread T_5 possesses a lock, blocked relation "T_5 <— lock(a) blocked T_0" will be formed. Then the block tree BT2 will be expanded to include BT1 as its subtree.

(3) If the root thread T_4 invokes **b**.**wait**() for an object **b**, the block tree BT2 will become a waiting block tree WBT. If the root thread T_4 of WBT is not notified or interrupted by any threads outside WBT and if a join-blocked thread T_6 in WBT is not interrupted by any threads outside WBT, then WBT will deadlock.

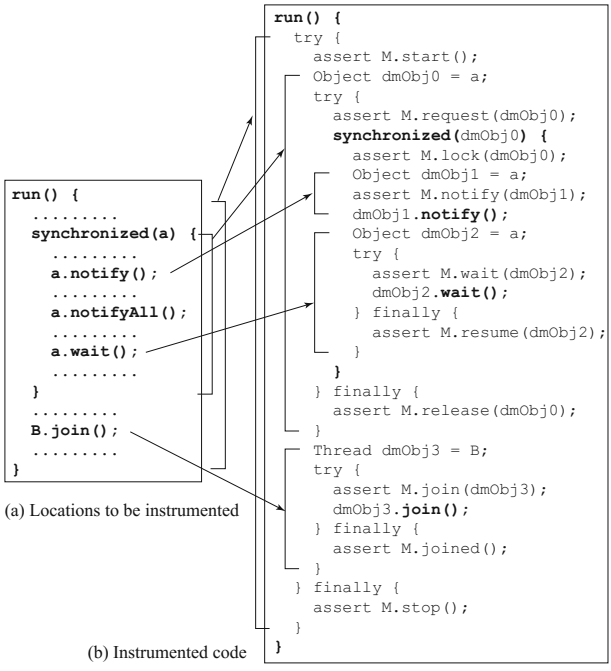


Fig. 3. Instrumentation for block process analysis

3.5 Classification of Deadlocks

This paper proposes the following two types of deadlocks.

(1) Block-cycle type deadlocks (BCDL)
According to Theorem 2, if join-blocked threads in a block cycle BC are not interrupted by any threads outside BC, the block cycle BC will deadlock. In addition, according to Theorem 1, if a block cycle is formed only by lock-blocked relations and wait-blocked relations, it will deadlock. These are called block-cycle type deadlocks (BCDL).

(2) Waiting-block type deadlocks (WBDL)
According to Theorem 3, for a set of waiting block trees, if all the root threads of the waiting block trees are not notified or interrupted and if join-blocked threads in the waiting block trees are not interrupted, then they will deadlock. These are called waiting-block type deadlocks (WBDL) [42].

4 Deadlock Detection

4.1 Thread Execution Methods

Definition 5 (Thread execution method). *The thread execution method (tem) of a thread is the method that starts the thread. For example, thread execution method "ap.Ap.main" denotes the* ***main****() method of ap.Ap class that starts the main thread. Thread execution method "ap.Ap.run" denotes the* ***run****() method ap.Ap class implements that starts a thread. For a thread T, tem(T) denotes a thread execution method that has started T. For a set of threads* $\{T\}$*, tem(*$\{T\}$*) denotes a set of the thread execution methods* $\{tem(T)\}$*.*

Definition 6 (ITS(tem)). *For a thread execution method tem, ITS(tem), the Interrupting Thread-execution-method Set of tem, denotes a set of the thread execution methods M that satisfies the following condition. For a thread T started by tem, a thread started by* $m \in M$ *may directly or indirectly invoke T.****interrupt****(). "Thread A indirectly invokes T.****interrupt****()" means that A starts another thread B and that B invokes T.****interrupt****().*

Definition 7 (NTS(tem)). *For a thread execution method tem, NTS(tem), the Notifying Thread-execution-method Set of tem, denotes a set of the thread execution methods M that satisfies the following condition. For a thread T started by tem, if T invokes* ***a.wait****() for an object* ***a****, a thread started by* $m \in M$ *may directly or indirectly invoke* ***a.notify****() or* ***a.notifyAll****(). "Thread A indirectly invokes* ***a.notify****()" means that A starts another thread B and that B invokes* ***a.notify****().*

Table 1. Processes of probes

Probes	**Processes**
start()	tem(A) is recorded.
stop()	A BCDL and a WBDL will be detected if they exist.

request(**a**)	**if** (object **a** is not registered in the lock table) {
	LT(**a**) = (**a**, -, {A/0}, ϕ, 0);
	} **else if** (LT(**a**) == (**a**, -, RTS, WTS, n)) {
	LT(**a**) = (**a**, -, RTS $\cup$ {A/0}, WTS, n);
	if (RTS == ϕ and n == 0) {
	NNRW = NNRW - WTS;
	}
	} **else if** (LT(**a**) == (**a**, L/r, RTS, WTS, n) and L != A) {
	LT(**a**) = (**a**, L/r, RTS $\cup$ {A/0}, WTS, n);
	Blocked relation "L <— lock(a) blocked A" will be formed.
	}
	A BCDL and a WBDL will be detected if they exist.

lock(**a**)	**if** (LT(**a**) == (**a**, -, RTS $\cup$ {A/r}, WTS, n)) {
	LT(**a**) = (**a**, A/r, RTS, WTS, n);
	Blocked relation "A <— lock(a) blocked R" will be formed for each R in RTS.
	Blocked relation "A <— wait(a) blocked W" will be formed for each W in WTS.
	} **else if** (LT(**a**) == (**a**, A/r, RTS, WTS, n)) {
	LT(**a**) = (**a**, A/r+1, RTS, WTS, n);
	}

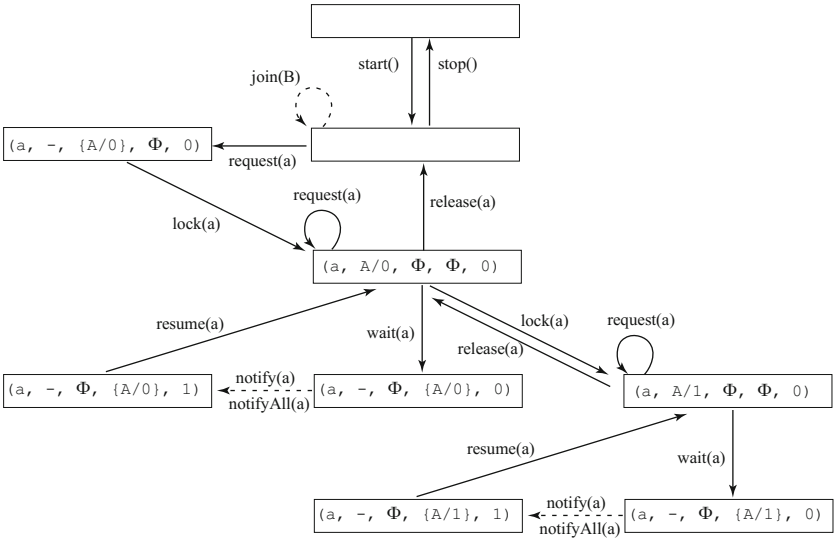
release(**a**)	Let LT(**a**) = (**a**, A/r, RTS, WTS, n).
	if (r > 0) {
	LT(**a**) = (**a**, A/r-1, RTS, WTS, n);
	} **else if** (r == 0) {
	LT(**a**) = (**a**, -, RTS, WTS, n);
	Blocked relation "A <— lock(a) blocked R" will be removed for each R in RTS.
	Blocked relation "A <— wait(a) blocked W" will be removed for each W in WTS.
	if (RTS == ϕ and n == 0) {
	NNRW = NNRW $\cup$ WTS;
	}
	}

wait(**a**)	Let LT(**a**) = (**a**, A/r, RTS, WTS, n).
	LT(**a**) = (**a**, -, RTS, WTS $\cup$ {A/r}, n);
	Blocked relation "A <— lock(a) blocked R" will be removed for each R in RTS.
	Blocked relation "A <— wait(a) blocked W" will be removed for each W in WTS.
	if (RTS == ϕ and n == 0) {
	NNRW = NNRW $\cup$ WTS $\cup$ {A};
	}
	A WBDL will be detected if it exists.

resume(**a**)	Let LT(**a**) = (**a**, -, RTS, WTS $\cup$ {A/r}, n). LT(**a**) = (**a**, A/r, RTS, WTS, n - 1); Blocked relation "A <— lock(a) blocked R" will be formed for each R in RTS. Blocked relation "A <— wait(a) blocked W" will be formed for each W in WTS.

notify(**a**)	Let LT(**a**) = (**a**, A/r, RTS, WTS, n).
	if (#WTS > n) {
	LT(**a**) = (**a**, A/r, RTS, WTS, n + 1);
	}

notifyAll(**a**)	Let LT(**a**) = (**a**, A/r, RTS, WTS, n). LT(**a**) = (**a**, A/r, RTS, WTS, #WTS);
join(B)	Blocked relation "B <— join blocked A" will be formed. A BCDL and a WBDL will be detected if they exist.
joined()	The join-blocked relation the current thread has will be deleted.



The second element "-" of the lock table indicates that no thread possesses a lock for object **a**.

Fig. 4. State transition of a lock table

4.2 Execution Inspection

Deadlock detection has been implemented by a variety of methods such as source code instrumentation [13], [25], bytecode instrumentation [27], [16], [44], [2], [19] and JVM profiler interface [31]. To detect the deadlocks this paper proposes, program execution made by **run**(), **synchronized** (**a**), **a**.**wait**(), **a**.**notify**(), **a**.**notifyAll**(), and B.**join**() need to be inspected. Figure 3 shows an example of source code instrumentation for execution inspection. It also indicates the timings at which program execution should be inspected regardless of which method implements the deadlock detection this paper proposes. For example, a synchronized statement could cause an exception when its expression becomes null. In addition, its synchronized block might be exited by the execution of a break, continue, or return statement. Therefore, the source code instrumentation of "**synchronized** (exp) { ... }" should be something as follows:

```
Object a = exp;
try {
  assert M.request(a);
  synchronized (a) {
    assert M.lock(a);
    .....
  }
} finally {
  assert M.release(a);
}
```

To ensure that programs can be executed in both modes of normal and monitored, these probes may be inserted as assertions, where the probes always return true.

4.3 Lock Tables for Deadlock Detection

For an object **a**, lock table LT(**a**) is (**a**, L/r, RTS, WTS, n), where L is a thread that possesses a lock for **a**. If no thread possesses a lock for **a**, L will be null, which is represented by "-". When thread L possesses a lock for **a**, (r + 1) indicates the number of lock actions performed by thread L on object **a** that have not been matched by unlock actions. It is equivalent to the number of times thread L have entered nested **synchronized** (**a**) blocks. RTS (Requesting Thread Set) is a set of threads that have requested a lock for object **a** immediately before **synchronized** (**a**) statement. WTS (Waiting Thread Set) is a set of threads that have invoked **a.wait**() and have not yet resumed. The number n is the number of threads in WTS that have been notified.

If L is null, one of the threads in RTS or one of the notified threads in WTS will be chosen by JVM, and it will lock object **a**. Figure 4 shows the state transition of lock table LT(**a**) when the probes described in Section 4.2 are executed by thread A although **a.notify**() and **a.notifyAll**() must be invoked by another thread.

4.4 Non-notified Root Waiting Thread Set

Let NNRW (Non-Notified Root Waiting thread set) be a set of threads any element W of which satisfies the following condition: W $\in$ WTS such that LT(**a**) = (**a**, -, ϕ, WTS, 0) for some object **a**. That is, thread W invokes **a.wait**(), but it is not wait blocked nor notified, and there exist no threads that request a lock for object **a**. This table makes waiting-block type deadlock detection efficient as described in Section 4.7.

4.5 Processes of Probes

Table 1 shows the processes of the probes introduced in Section 4.2. Thread A is the current thread that is executing the probes. "tem(A)" represents the thread execution method of the current thread A. Probe start() records it by inspecting an array of stack trace elements of the current thread. The second element "-" in the lock table LT(**a**) indicates that no thread possesses a lock for object **a**. #WTS represents the number of threads in WTS. The probes updates lock table LT(**a**) for object **a** and NNRW. According to the transition of the lock table, some blocked relations will be formed, and a BCDL and a WBDL will be detected if they exist.

4.6 Block-Cycle Type Deadlock Detection

Block-cycle type deadlocks will be checked every time probe request(**a**) or join(B) is processed as described in Section 4.5. When a thread gets blocked, it must

```
 1 public class BcWait implements Runnable {
 2   static int numOfWaits = 0;
 3   static synchronized void waitForTheOtherThread()
       throws Exception {
 4     while (numOfWaits == 0) {
 5       numOfWaits++;
 6       BcWait.class.wait();
 7     }
 8     BcWait.class.notify();
 9   }
10   static Thread mainThread;
11   Object a;
12   Object b;
13   BcWait(Object a, Object b) {
14     this.a = a;
15     this.b = b;
16   }
17   public void run() {
18     try {
19       waitForTheOtherThread();
20       synchronized (a) {
21         a.notify();
22         synchronized (b) {
23         }
24       }
25     } catch (Exception e) {
26       e.printStackTrace();
27     }
28   }
29   public static void main(String[] args)
       throws Exception {
30     mainThread = Thread.currentThread();
31     Object a = new Object();
32     Object b = new Object();
33     Thread thread = new Thread(new BcWait(a, b));
34     thread.start();
35     synchronized (b) {
36       synchronized (a) {
37         waitForTheOtherThread();
38         a.wait();
39       }
40     }
41   }
42 }
```

Fig. 5. Source code of BcWait that generates a block-cycle type deadlock with a wait-blocked thread

be the root thread of a block tree. As illustrated in Fig. 2, if it is blocked by a thread in the same block tree, this will form a block cycle. If it is blocked by a thread outside the block tree, the block tree will become a part of another block tree.

When thread A gets blocked by thread B, a block-cycle type deadlock can be detected where the root thread of the block tree that includes B equals A and join-blocked threads {T} in the block cycle BC are not interrupted by any threads outside BC. Let ETS (Existing Thread Set) be all the threads that have started and not stopped. A block-cycle type deadlock will be detected if tem(ETS - BC) $\cap$ ITS(tem({T})) = ϕ.

4.7 Waiting-Block Type Deadlock Detection

Waiting-block type deadlocks will be checked every time probe stop(), request(**a**), wait(**a**) or join(B) is processed as described in Section 4.5. When a thread performs one of these processes, a waiting-block type deadlock can be detected if the root thread R of a waiting block tree WBT is not notified or interrupted and if join-blocked threads {T} in WBT are not interrupted. Let ETS be all the threads that have started and not stopped. A waiting-block type deadlock will be detected if tem(ETS - WBT) $\cap$ (ITS(tem(R)) $\cup$ NTS(tem(R)) $\cup$ ITS(tem({T}))) = ϕ. The root threads of waiting block trees that have not be notified can be obtained from NNRW (Non-Notified Root Waiting thread set), which is described in Section 4.4.

5 Examples of Deadlock Detection

5.1 Lock Table and Block Graph for Example Programs

This section illustrates the changes of a lock table, an NNRW table, and a block graph for six example programs, one of which (program BcWait) is shown in Fig. 5. Each program creates and starts a thread T in the second way explained in Section 2.1. In the program, two threads, the main thread M and thread T are running. The waitForTheOtherThread() method is used to wait for the other thread to invoke this method so that both threads will resume execution at the same time as shown in Fig. 5. Threads M and T will deadlock in the following ways:

(1) program BcLock
Figure 6 (a) shows the changes of a lock table, an NNRW table, and a block graph for program BcLock. This will cause a BCDL(Lock), where blocked relations "T <— lock(a) blocked M" and "M <— lock(b) blocked T" are formed.

(2) program BcWait
Figure 6 (b) shows the changes of a lock table, an NNRW table, and a block graph for program BcWait. This will cause a BCDL(Wait), where blocked relations "T <— wait(a) blocked M" and "M <— lock(b) blocked T" are formed. It

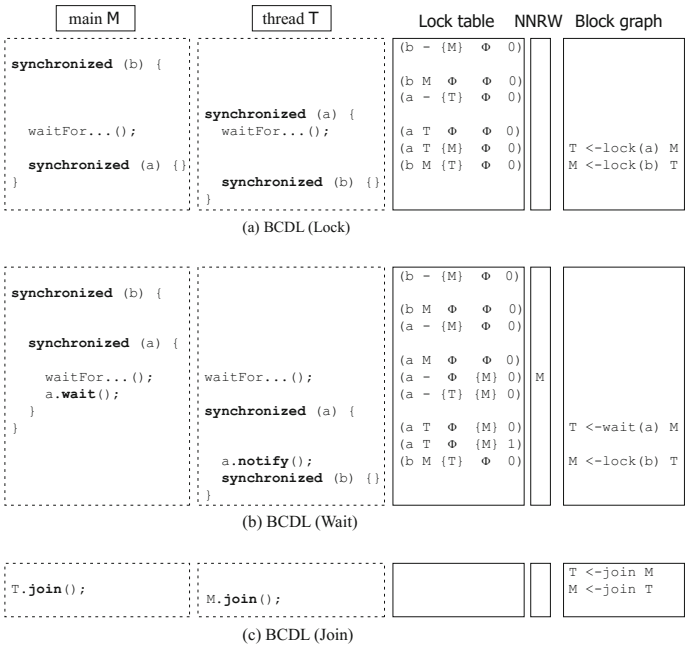


Fig. 6. Transition of a lock table and block graph for block-cycle type deadlocks

should be noted that the main thread M has been notified by thread T. However, it cannot resume as it is wait blocked by thread T. Here, NTS(BcWait.main) = {BcWait.run}.

(3) program BcJoin
Figure 6 (c) shows the changes of a lock table, an NNRW table, and a block graph for program BcJoin. This will cause a BCDL(Join), where blocked relations "T <— join blocked M" and "M <— join blocked T" are formed.

(4) program WbLock
Figure 7 (a) shows the changes of a lock table, an NNRW table, and a block graph for program WbLock. Because NTS(WbLock.run) = ϕ, this will cause a waiting-block type deadlock that consists of only T as soon as T waits. On the other hand, if "NTS(WbLock.run) = {WbLock.main}" is given, this will cause a WBDL(Lock) that consists of two threads T and M, where T is waiting on **b** without notification and blocked relation "T <— lock(a) blocked M" is formed.

(5) program WbWait
Figure 7 (b) shows the changes of a lock table, an NNRW table, and a block graph for program WbWait. This will cause a WBDL(Wait), where T is waiting on **b** without notification and blocked relation "T <— wait(a) blocked M"

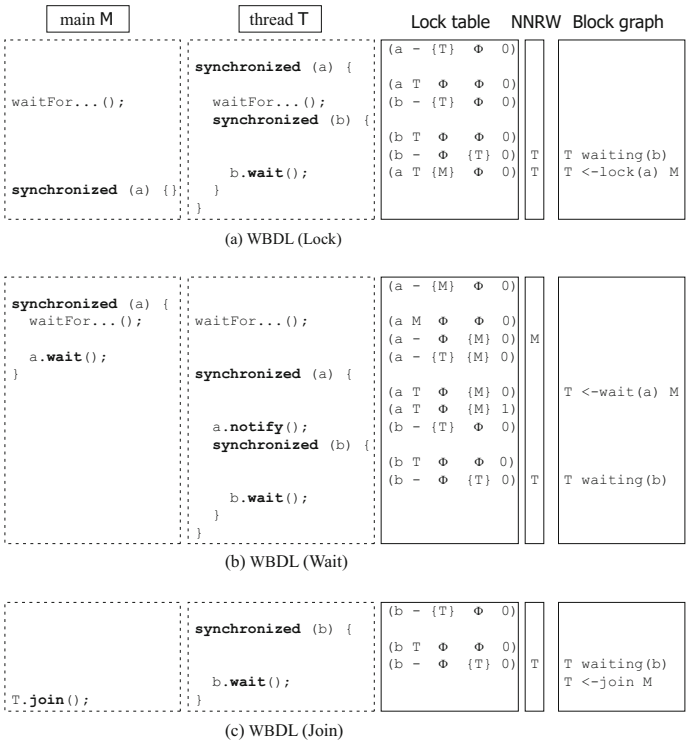


Fig. 7. Transition of a lock table and block graph for waiting blocked tree deadlocks

is formed. It should be noted that the main thread M has been notified by thread T. However, it cannot resume as it is wait blocked by thread T. Here, NTS(WbWait.main) = {WbWait.run}.

(6) program WbJoin
Figure 7 (c) shows the changes of a lock table, an NNRW table, and a block graph for program WbJoin. Because NTS(WbJoin.run) = ϕ, this will cause a waiting-block type deadlock that consists of only T as soon as T waits. On the other hand, if "NTS(WbJoin.run) = {WbJoin.main}" is given, this will cause a WBDL(Join) that consists of two threads T and M, where T is waiting on b without notification and blocked relation "T <— join blocked M" is formed.

5.2 Deadlock Detection Examples

A prototype system DLM (DeadLock Monitor) has been developed to detect the deadlocks this paper proposes. For a thread execution method tem, ITS(tem) and NTS(tem) should be determined by static analysis. On the other hand, instead of static analysis, DLM uses ITS.conf and NTS.conf files where ITS(tem) and

```
=================== Block-Cycle Type Deadlock Detected ===================
[ 1]  Thread (Thread[ Thread-0,5,main] : eb.BcWait.run(BcWait.java:22)) lock(java.lang.Object@1bab50a) blocked by
[ 2]  Thread (Thread[ main,5,main] : eb.BcWait.main(BcWait.java:38)) wait(java.lang.Object@e53108) blocked by
Thread (Thread[ Thread-0,5,main] )
```

(a) Block-cycle (wait) type deadlock with 2 threads

```
=================== Waiting-Block Type Deadlock Detected ===================
[ 1]  Thread (Thread[ Thread-0,5,main] : eb.WbWait.run(WbWait.java:49)) waiting(java.lang.Object@9931f5)
[ 2]  -- Thread (Thread[ main,5,main] : eb.WbWait.main(WbWait.java:83)) wait(java.lang.Object@1bab50a) blocked
```

(b) Waiting-block (wait) type deadlock with 2 threads

Fig. 8. Deadlock detection results with 2 threads

```
=================== Block-Cycle Type Deadlock Detected ===================
[ 1]  Thread (Thread[ Thread-0,5,main] : ea.ThreadN.run(ThreadN.java:248)) lock(java.lang.Object@1034bb5) blocked by
[ 2]  Thread (Thread[ Thread-4,5,main] : ea.ThreadN.run(ThreadN.java:258)) lock(java.lang.Object@1820dda) blocked by
[ 3]  Thread (Thread[ Thread-9,5,main] : ea.ThreadN.run(ThreadN.java:258)) lock(java.lang.Object@1d9dc39) blocked by
[ 4]  Thread (Thread[ Thread-8,5,main] : ea.ThreadN.run(ThreadN.java:258)) lock(java.lang.Object@110b053) blocked by
[ 5]  Thread (Thread[ Thread-7,5,main] : ea.ThreadN.run(ThreadN.java:258)) lock(java.lang.Object@723d7c) blocked by
[ 6]  Thread (Thread[ Thread-6,5,main] : ea.ThreadN.run(ThreadN.java:258)) lock(java.lang.Object@8814e9) blocked by
[ 7]  Thread (Thread[ Thread-5,5,main] : ea.ThreadN.run(ThreadN.java:258)) lock(java.lang.Object@e7b241) blocked by
[ 8]  Thread (Thread[ Thread-3,5,main] : ea.ThreadN.run(ThreadN.java:258)) lock(java.lang.Object@7ffe01) blocked by
[ 9]  Thread (Thread[ Thread-2,5,main] : ea.ThreadN.run(ThreadN.java:258)) lock(java.lang.Object@471e30) blocked by
[ 10]  Thread (Thread[ Thread-1,5,main] : ea.ThreadN.run(ThreadN.java:258)) lock(java.lang.Object@e53108) blocked by
Thread (Thread[ Thread-0,5,main] )
```

(a) Block-cycle type deadlock with 10 threads

```
------------------- Waiting-Block Type Deadlock Detected -------------------
[ 1]  Thread (Thread[ main,5,main] : ea.ThreadN.createThreads(ThreadN.java:191)) waiting(class ea.ThreadN)
[ 2]  Thread (Thread[ Thread-0,5,main] : ea.ThreadN.run(ThreadN.java:233)) waiting(class ea.ThreadN)
[ 3]  -- Thread (Thread[ Thread-9,5,main] : ea.ThreadN.run(ThreadN.java:258)) lock(java.lang.Object@e53108) blocked
[ 4]  -- -- Thread (Thread[ Thread-1,5,main] : ea.ThreadN.run(ThreadN.java:258)) lock(java.lang.Object@1034bb5) blocked
[ 5]  -- -- -- Thread (Thread[ Thread-2,5,main] : ea.ThreadN.run(ThreadN.java:258)) lock(java.lang.Object@117a8bd) blocked
[ 6]  -- -- -- -- Thread (Thread[ Thread-3,5,main] : ea.ThreadN.run(ThreadN.java:258)) lock(java.lang.Object@1fae3c6) blocked
[ 7]  -- -- -- -- -- Thread (Thread[ Thread-4,5,main] : ea.ThreadN.run(ThreadN.java:258)) lock(java.lang.Object@743399) blocked
[ 8]  -- -- -- -- -- -- Thread (Thread[ Thread-5,5,main] : ea.ThreadN.run(ThreadN.java:258)) lock(java.lang.Object@a981ca) blocked
[ 9]  -- -- -- -- -- -- -- Thread (Thread[ Thread-6,5,main] : ea.ThreadN.run(ThreadN.java:258)) lock(java.lang.Object@19efb05) blocked
[ 10]  -- -- -- -- -- -- -- -- Thread (Thread[ Thread-7,5,main] : ea.ThreadN.run(ThreadN.java:258)) lock(java.lang.Object@111a3ac) blocked
```

(b) Waiting-block type deadlock with 11 threads

Fig. 9. Deadlock detection results with 10 threads

NTS(tem) are required, respectively. For example, a line "ap.Ap.run ap.Ap.main ap.Ap.run" in the ITS.conf file indicates that "ITS(ap.Ap.run) = {ap.Ap.main, ap.Ap.run}", that is, a thread started by "ap.Ap.run" may be interrupted by the main thread started by"ap.Ap.main" and by another thread started by "ap.Ap.run".

When a deadlock is detected, in addition to reporting it, it is important to give some information to assist further debugging. However, it should not be complicated. Therefore, this system gives programmers the following information:

(1) When a block-cycle type deadlock is detected, the threads that form a block cycle are displayed in turn, which includes each thread, the source code location at which the thread was blocked, and its blocked relation.

(2) When a waiting-block type deadlock is detected, the threads that form the set of the waiting block trees are displayed in turn, which includes each thread, the source code location at which the thread was blocked, and its blocked relation. In addition, the indentation of each line indicates the hierarchy of the

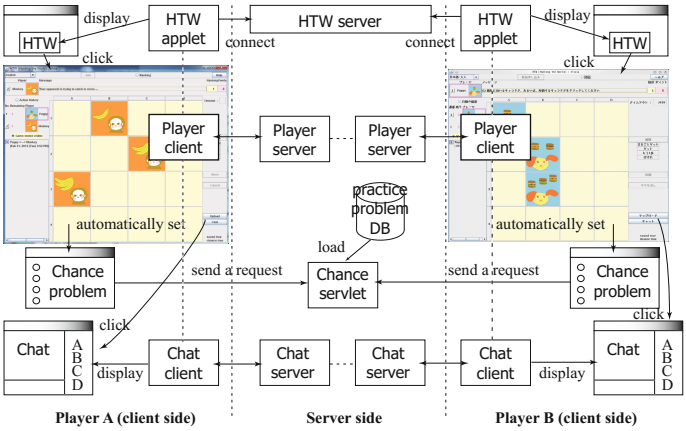


Fig. 10. HTW game structure

waiting block trees. The threads with no indentations indicate the root threads of their waiting block trees.

Figures 8 and 9 show output examples of deadlock detection for the execution of some programs. Figure 8 (a) and (b) show the deadlock detection of BCDL(Wait) and WBDL(Wait) for the example programs, respectively. Figure 9 (a) shows an output example of a block-cycle type deadlock with 10 threads, and Fig. 9 (b) shows an output example of a waiting-block type deadlock with 10 threads, where there exist two waiting block trees, Thread [1] and Threads [2] to [10]. If programmers would like to know more details such as the exact call stack of each thread, they can use other tools like jps, jstack, and so on [35].

6 Evaluation

6.1 Application to an HTW Game

The method this paper proposes has been applied to a part of an e-Learning system Apty [40], [41]. Apty provides a collaborative game HTW (Hunting The World) to promote students' motivation of learning. Anyone can start and join the game anytime. If some people join the game at the same time, the first two people will become a pair of players and the game will start. At the same time, multiple pairs can play the game and chat with all the players. During each game, a chance problem will be automatically given. The first player who answers it correctly can get an extra character, which will result in a higher score.

Figure 10 shows the system configuration of the game. The Java applet is running on the client side while communicating with several programs on the server side. When a student clicks on the HTW button in a subject home Web page, the applet will send a subject name to the server, and receive a port number for later connection. A new window will be opened to display a game

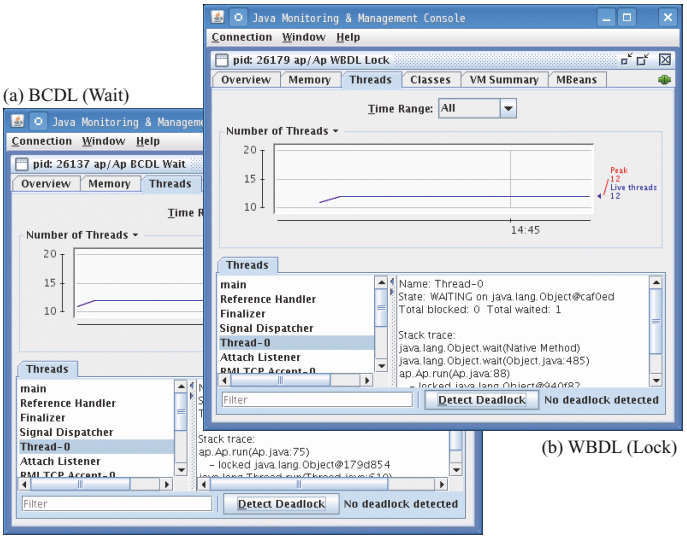


Fig. 11. Deadlock detection results for BCDL and WBDL with a jconsole tool

board, and a client player thread will start and connect with a player server by using the returned port number. When the student joins the game, the player's information will be recorded on the server side. When his or her opponent joins the game, the game will start. During the game, when the player server informs the client player thread of a chance problem being set, the client player thread will open a Web browser window and send a request to the Chance servlet (server-side program). The Chance servlet will retrieve an appropriate question from the practice problem DB table of the subject, and return a response to show the question in a Web page of the browser window. The same question will be shown to the pair of players at the same time. If one player answers the question correctly earlier than the other, he or she will gain an extra character.

In the HTW system, when the first student joins, the corresponding player server thread (playerServer1) will invoke **subjectInfo.wait**() to leave the first student waiting for his or her opponent to join. When his or her opponent joins, the corresponding player server thread (playerServer2) will invokes **subjectInfo.notify**(). The notified playerServer1 will next invoke **playInfo.wait**() to wait for the game to start. Then, playerServer2 will invoke **playInfo.notifyAll**() to start the game. When a chance problem is being given, both of these two threads will invoke **playInfo.wait**() to wait for the players to answer it. Therefore, in this program, **playInfo.notifyAll**() is invoked to start and resume the game. However, because playerServer2 invoked **playInfo.wait**() by mistake when it possessed a lock for **subjectInfo**, a blocked relation "playerServer2: waiting(playInfo) <— wait(subjectInfo) blocked playerServer1" was formed,

Table 2. Probes and their computational complexities (CC) for N threads

Probes	**Processes that affect computational complexities**	**CC**
start()	None	O(1)
stop()	(1) Remove join-blocked threads. (2) Detect a waiting block tree type deadlock.	O(N)
request(a)	(1) Update the root thread for its block tree. (2) Detect a block-cycle type and a waiting-block type deadlock.	O(N)
lock(a)	(1) Add blocked threads and update their block trees' root thread.	O(N)
release(a)	(1) Register waiting threads. (2) Remove blocked threads.	O(N)
wait(a)	(1) Remove blocked threads. (2) Detect a waiting block tree type deadlock.	O(N)
resume(a)	(1) Add blocked threads and update their block trees' root thread.	O(N)
notify(a)	None	O(1)
notifyAll(a)	None	O(1)
join(B)	(1) Add the current thread and update its block tree's root thread. (2) Detect a block cycle type and a waiting block tree type deadlock.	O(N)
joined()	None	O(1)

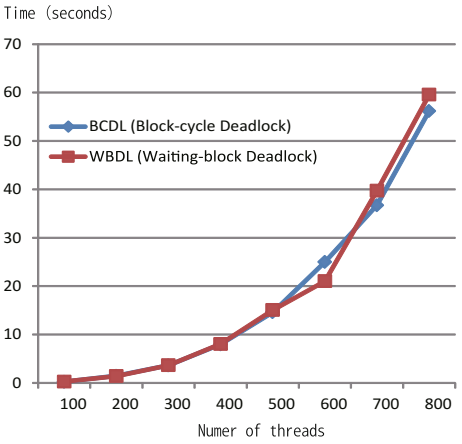


Fig. 12. Process time of DLM

where NTS(PlayerServer.run) = {PlayerServer.run, AnswerChance.run, TimeoutMonitor.run}. When a chance problem is not given to a pair of players, AnswerChance servlet threads and TimeoutMonitor threads are not running. As a result, this waiting-block type deadlock was detected by DLM for the first pair of players.

It cannot be predicted what will cause program faults. For example, you might lock an object **a** and then invoke B.**join**() to wait for thread B to stop while possessing a lock for **a**. Furthermore, you might lock an object **a** and then invoke **b**.**wait**() on another object **b**. These types of coding could cause faults. To detect these faults, the two types of deadlock detection this paper proposes will be important. Existing tools cannot detect all the deadlocks this paper proposes. The tool jconsole [34] only detects a BCDL(Lock) among six kinds of deadlocks described in Section 5.1 as shown in Fig. 11. Some IDEs cannot detect a BCDL(Join), a WBDL(Lock), a WBDL(Wait) or a WBDL(Join) [36].

6.2 Computational Complexities

Let us consider the performance of DLM. A test program ThreadN has been made which creates and starts multiple threads, the number of which is specified as a parameter of the program. It will deadlock in two ways, a block-cycle type deadlock and a waiting-block type deadlock. Table 2 shows the computational complexity of each probe for N threads. To detect a block-cycle type deadlock, DLM has only to check whether the current thread equals the root thread of the block tree that includes a thread blocking the current one, which requires O(1). To detect a waiting-block type deadlock, DLM needs to check whether the root thread of a waiting block tree is not notified or interrupted, which requires O(N). Introducing the NNRW table described in Section 4.4 made this performance possible. As a result, the process time of DLM becomes O(N).

Figure 12 shows that the process time of test program ThreadN with DLM on a Linux machine (Intel Core i7 2.4GHz, 2GB memory) will increase according to the number of threads. It can be seen from the figure that the process time of test program ThreadN with DLM becomes $O(N^2)$ because N threads invoke probes in DLM each of which requires O(N).

6.3 Soundness and Completeness of the Deadlock Detection

The deadlock detection this paper presents is sound but not complete. It reports no false positives. The detected deadlocks are all faults. Implementations of JVM are permitted to perform "spurious wakeups", that is, to remove threads from wait sets and thus enable resumption without explicit instructions to do so. Such spurious wakeups caused by operating systems do not need to be considered for deadlock detection. They will cause other types of faults than deadlocks because they wake up threads against programmers' intention. Therefore, like the code snippet shown in Section 3.1 (2), it is recommended that **wait**() methods should always be used only within a loop that terminates only when some logical condition that the thread is waiting for holds. Suppose that a set of threads whose

deadlock is detected by the algorithm this paper presents have resumed due to the spurious wakeup. Because the resumption is not intended by programmers, it may be regarded as a fault of deadlock.

On the other hand, the deadlock detection DLM (DeadLock Moniter) employs is not complete for two reasons. DLM can detect all block-cycle type deadlocks if they are formed at run-time. Especially, it can detect a block-cycle type deadlock that includes wait-blocked threads, which the previous work was unable to detect. However, firstly, it usually detects a waiting-block type deadlock that consists of only one waiting block tree. A block graph can include more than one waiting block tree at the same time. If a set of waiting block trees satisfies Theorem 3, which is considered to rarely happen, they will deadlock. DLM does not detect this case by default. It is a trade-off between the accuracy and the performance of deadlock detection. Secondly, NTS(tem) is defined for not an object but a thread. Suppose that two threads A and B are running and that NTS(tem(A)) = {tem(B)}. Even if thread A invokes **a.wait**() and then it is waiting without notification, DLM will not detect a waiting-block type deadlock that consists of thread A since NTS(tem(A)) = {tem(B)}, that is, thread B might invoke **a.notify**(). However, thread B might invoke not **a.notify**() but **b.notify**() when thread A invokes **b.wait**().

7 Related Work

7.1 Static Analysis for Deadlock Detection

Methods in object-oriented concurrent libraries often encapsulate internal synchronization details. As a result of information hiding, clients calling the library methods may cause thread safety violations by invoking methods in an unsafe manner. This is frequently a cause of deadlocks. Given a concurrent library, Deshmukh, J.V. et al.[9], [8] present a technique for inferring interface contracts that specify permissible concurrent method calls and patterns of aliasing among method arguments. The derived contracts guarantee deadlock-free execution for the methods in the library. Their technique combines static analysis with a symbolic encoding scheme for tracking lock dependencies. Shanbhag, V.K. [39] uses static analysis to fetch a list of potential deadlock scenarios from Java libraries by using a coarser level of granularity in lock-order graph construction. Williams, A. et al. [45] propose a method for static detection of deadlock in Java libraries to determine whether client code exists that may deadlock a library. Flow-sensitive, context-sensitive analysis determines possible deadlock configurations using a lock-order graph. Kahlon, V. et al. [23], [24] propose a static data race detection technique for multi-threaded C programs with asynchronous function calls.

Lock capabilities [15] statically verify that multi-threaded programs with locks will not deadlock. Although most previous work on deadlock prevention requires a strict total order on all locks held simultaneously by a thread, lock capabilities do not enforce a total order and support idioms that use fine-grained locking, such as mutable binary trees, circular lists, and arrays where each element has a different lock. Pun, K.I. et al. [38] propose a static deadlock detection by using a

behavioral type and effect system that, in the first stage, checks the behavior of each thread or process against the declared behavior, which captures potential interaction of the thread with the locks. In the second step on a global level, the state space of the behavior is explored to detect potential deadlocks. Johnsen, E. B. et al. [18] propose a static type and effect system to prevent non-lexical lock errors for a formal, object-oriented calculus which supports non-lexical lock handling and exceptions. Naik, M. et al. [29] propose a static deadlock detection algorithm for multi-threaded Java programs that detects some problems such as reachable, aliasing, escaping and parallel. However, because of the static analysis of Java programs, it may also report a few false positives.

Some systems apply model checking techniques to deadlock detection. JCAT [7] detects deadlock situations in Java programs that make use of multi-threading mechanisms. An abstract formal model is generated from the Java source using the Java2Spin translator. The model is expressed in the PROMELA language, and the SPIN tool is used to perform its formal analysis. Because JCAT uses a model that is extracted from the parallel processes of a source program, it cannot detect deadlocks in real environments and it does not support polymorphism. Bogor[28] is a model checker for automatic deadlock detection. All parts of a Java program that may cause a deadlock such as global variables and shared resources are translated as the input to Bogor, and Bogor creates an automaton whose states represent all possible program's states and shows whether there is any deadlock in the program. If Bogor finds a deadlock, it will show a counter example to help programmers to fix the problem. Bensalem, S. et al. [3] propose efficient deadlock detection by combining the information from the invariant with model checking techniques and strategies for reducing the memory footprint. It generates the exact set of reachable specification violations along with traces to demonstrate the error. Gadara [43] automates dynamic deadlock avoidance for conventional multi-threaded programs. It employs whole-program static analysis to model programs, and to synthesize lightweight, decentralized, highly concurrent logic that controls them at runtime. Model checking [6] based on counterexample-guided abstraction refinement is applied to deadlock detection in message passing based C programs. Model checking is usually limited to relatively small, especially critical programs due to the state explosion problem, and could detect some false positives.

7.2 Dynamic Analysis for Deadlock Detection

JTWFG [31] is a Java thread-wait-for graph to represent synchronization waiting states in the execution of a Java program to detect deadlocks. The graph consists of join waiting relations, notification waiting relations, and monitor waiting relations. Notification waiting relations are obtained by static analysis of the program. The other relations are obtained by using the JVM profiler interface (JVMPI). JTWFG cannot form notification waiting relations in the graph if the invocation of **notify**() is missing in the program and it cannot detect a BCDL(Wait) described in Section 5.1. CHECKMATE [19] first generates a trace program from the execution of a multi-threaded annotated and instrumented

program, and then a model checker explores all possible executions of the trace program to report possible deadlocks. It can detect a missed notification of Java. However, it requires programmers to identify synchronization predicates using an annotation mechanism of Java.

Some systems employ bytecode instrumentation. Multicore SDK [27] implements a two-phase deadlock detection algorithm. It modifies Java class bytecode at certain points to insert instrumentation and examines a single execution trace obtained by running the instrumented program. The algorithm reduces lock graphs based on program locations. Certain locks are filtered out that cannot participate in a deadlock by analyzing the lock graph created in the first phase. Even smaller lock graph is analyzed for cycles to find potential deadlocks in the application. Java PathFinder [16] implements two runtime analysis algorithms for detecting the latency of data races and deadlocks. This system uses a home grown Java Virtual Machine (JVM) to check bytecode directly. JDeadlockDetector [44] can detect deadlocks without source code because it is built on the official Java Virtual Machine (JVM). It uses a dynamic instrumentation mechanism to insert the interception code into the monitored Java applications. However, it can only detect a BCDL(Lock) described in Section 5.1. It might be because JDeadlockDetector uses thread states obtained by the invocation of a thread's getState() method. They are roughly classified as NEW, RUNNABLE, BLOCKED, WAITING, TIMED_WAITING, and TERMINATED. BLOCKED indicates that a thread is blocked waiting for a monitor lock. WAITING indicates that a thread is waiting indefinitely for another thread to perform a particular action. JPAX [2], [17], [4] implements a framework for confirming deadlock potentials detected by runtime analysis of a single run of a multi-threaded program. It automatically instruments the bytecode class files of the multi-threaded program by adding new instructions that when executed generate the execution trace consisting of lock and unlock events needed for the analysis. A lock graph is constructed which can reveal deadlock potentials in the form of cycles. It may yield false positives to indicate warnings rather than errors.

Communix [22],[21] is a collaborative deadlock immunity framework for Java programs. Dimmunix detects deadlocks at runtime and generates signatures to avoid reoccurrences of the same deadlocks. Communix distributes the deadlock signatures produced by Dimmunix. Agarwal, R. et al. [1] propose run-time algorithms to detect potential deadlocks in programs that use locks (block structured as well as non block structured), semaphores, and condition variables. Flanagan, C. et al. [14] propose a dynamic analysis for atomicity that is a fundamental correctness property in multi-threaded programs. The analysis reasons about the exact dependencies between operations in the observed trace of the target program, and it reports error messages if and only if the observed trace is not conflict-serializable. DEADLOCKFUZZER [20] consists of two phases to detect potential deadlocks and create real ones. In the first phase, it executes a multi-threaded program and finds potential deadlocks that could happen in some execution of the program. This phase identifies potential deadlocks even if the observed execution does not deadlock, and provides suitable debugging

information to identify the cause of the deadlock. This debugging information is used by the second phase to create real deadlocks with high probability. In this phase, a random thread scheduler is biased to generate an execution that creates a real deadlock reported in the previous phase with high probability.

8 Conclusion and Future Directions

This paper has proposed block-cycle type deadlocks and waiting-block type deadlocks for Java multi-threaded programs and presented an example of implementation to detect these deadlocks. ITS(Interrupting Thread Set) and NTS(Notifying Thread Set) the paper defines are important to locate these deadlocks among a lot of threads running in a concurrent program. These sets should be determined in advance by static analysis of the program, which will augment the power of the deadlock detection this paper presents. The deadlocks caused by non-lexical lock primitives like Lock.lock() and Lock.unlock() will be detected by enhancing the proposed method. These problems remain as future work.

References

1. Agarwal, R., Stoller, S.: Run-time detection of potential deadlocks for programs with locks, semaphores, and condition variables. In: Proceedings of the 2006 Workshop on Parallel and Distributed Systems: Testing and Debugging, pp. 51–60. ACM (2006)
2. Bensalem, S., Fernandez, J., Havelund, K., Mounier, L.: Confirmation of deadlock potentials detected by runtime analysis. In: Proceedings of the 2006 Workshop on Parallel and Distributed Systems: Testing and Debugging, pp. 41–50. ACM (2006)
3. Bensalem, S., Griesmayer, A., Legay, A., Nguyen, T., Peled, D.: Efficient deadlock detection for concurrent systems. In: 2011 9th IEEE/ACM International Conference on Formal Methods and Models for Codesign (MEMOCODE), pp. 119–129. IEEE (2011)
4. Bensalem, S., Havelund, K.: Dynamic deadlock analysis of multi-threaded programs. In: Ur, S., Bin, E., Wolfsthal, Y. (eds.) HVC 2005. LNCS, vol. 3875, pp. 208–223. Springer, Heidelberg (2006)
5. Bradbury, J., Jalbert, K.: Defining a catalog of programming anti-patterns for concurrent java. In: Proc. of the 3rd International Workshop on Software Patterns and Quality (SPAQu 2009), pp. 6–11 (2009)
6. Chaki, S., Clarke, E., Ouaknine, J., Sharygina, N., Sinha, N.: Concurrent software verification with states, events, and deadlocks. Formal Aspects of Computing 17(4), 461–483 (2005)
7. Demartini, C., Iosif, R., Sisto, R.: A deadlock detection tool for concurrent java programs. Software: Practice and Experience 29(7), 577–603 (1999)
8. Deshmukh, J., Emerson, E., Sankaranarayanan, S.: Symbolic deadlock analysis in concurrent libraries and their clients. In: Proceedings of the 2009 IEEE/ACM International Conference on Automated Software Engineering, pp. 480–491. IEEE Computer Society (2009)
9. Deshmukh, J., Emerson, E., Sankaranarayanan, S.: Symbolic modular deadlock analysis. Automated Software Engineering, 1–38 (2011)

10. El-Zawawy, M., Nayel, H.: Type systems based data race detector. Computer and Information Science 5(4), 53–60 (2012)
11. Ferrara, P.: A generic static analyzer for multithreaded java programs. Software: Practice and Experience (2012), doi:10.1002/spe.2126
12. Fiedor, J., Křena, B., Letko, Z., Vojnar, T.: A uniform classification of common concurrency errors. In: Moreno-Díaz, R., Pichler, F., Quesada-Arencibia, A. (eds.) EUROCAST 2011, Part I. LNCS, vol. 6927, pp. 519–526. Springer, Heidelberg (2012)
13. Filman, R., Havelund, K.: Source-code instrumentation and quantification of events. In: FOAL 2002 Workshop (at AOSD 2002), pp. 45–49 (2002)
14. Flanagan, C., Freund, S., Yi, J.: Velodrome: a sound and complete dynamic atomicity checker for multithreaded programs. ACM SIGPLAN Notices 43(6), 293–303 (2008)
15. Gordon, C., Ernst, M., Grossman, D.: Static lock capabilities for deadlock freedom. In: Proceedings of the 8th ACM SIGPLAN Workshop on Types in Language Design and Implementation, pp. 67–78. ACM (2012)
16. Havelund, K.: Using runtime analysis to guide model checking of java programs. In: Havelund, K., Penix, J., Visser, W. (eds.) SPIN 2000. LNCS, vol. 1885, pp. 245–264. Springer, Heidelberg (2000)
17. Havelund, K., Rosu, G.: Java pathexplorer–a runtime verification tool. In: The 6th International Symposium on Artificial Intelligence, Robotics and Automation in Space: A New Space Odyssey, pp. 18–21 (2001)
18. Johnsen, E.B., Tran, T.M.T., Owe, O., Steffen, M.: Safe locking for multi-threaded java with exceptions. The Journal of Logic and Algebraic Programming 81, 257–283 (2012)
19. Joshi, P., Naik, M., Sen, K., Gay, D.: An effective dynamic analysis for detecting generalized deadlocks. In: Proceedings of the Eighteenth ACM SIGSOFT International Symposium on Foundations of Software Engineering, pp. 327–336. ACM (2010)
20. Joshi, P., Park, C., Sen, K., Naik, M.: A randomized dynamic program analysis technique for detecting real deadlocks. ACM Sigplan Notices 44(6), 110–120 (2009)
21. Jula, H., Andrica, S., Candea, G.: Efficiency optimizations for implementations of deadlock immunity. In: Khurshid, S., Sen, K. (eds.) RV 2011. LNCS, vol. 7186, pp. 78–93. Springer, Heidelberg (2012)
22. Jula, H., Tozun, P., Candea, G.: Communix: A framework for collaborative deadlock immunity. In: 2011 IEEE/IFIP 41st International Conference on Dependable Systems & Networks (DSN), pp. 181–188. IEEE (2011)
23. Kahlon, V., Sinha, N., Kruus, E., Zhang, Y.: Static data race detection for concurrent programs with asynchronous calls. In: Proceedings of the the 7th Joint Meeting of the European Software Engineering Conference and the ACM SIGSOFT Symposium on the Foundations of Software Engineering, pp. 13–22. ACM (2009)
24. Kahlon, V., Yang, Y., Sankaranarayanan, S., Gupta, A.: Fast and accurate static data-race detection for concurrent programs. In: Damm, W., Hermanns, H. (eds.) CAV 2007. LNCS, vol. 4590, pp. 226–239. Springer, Heidelberg (2007)
25. Koskinen, E., Herlihy, M.: Dreadlocks: efficient deadlock detection. In: Proceedings of the Twentieth Annual Symposium on Parallelism in Algorithms and Architectures, pp. 297–303. ACM (2008)
26. Lu, S., Park, S., Seo, E., Zhou, Y.: Learning from mistakes: a comprehensive study on real world concurrency bug characteristics. ACM Sigplan Notices 43(3), 329–339 (2008)

27. Luo, Z., Das, R., Qi, Y.: Multicore sdk: A practical and efficient deadlock detector for real-world applications. In: 2011 IEEE Fourth International Conference on Software Testing, Verification and Validation (ICST), pp. 309–318. IEEE (2011)
28. Mahdian, F., Rafe, V., Rafeh, R.: A framework to automatic deadlock detection in concurrent programs. Przeglad Elektrotechniczny, 182–184 (2012) iSSN 0033-2097, R. 88 NR 1b/2012
29. Naik, M., Park, C., Sen, K., Gay, D.: Effective static deadlock detection. In: IEEE 31st International Conference on Software Engineering, ICSE 2009, pp. 386–396. IEEE (2009)
30. Nir-Buchbinder, Y., Tzoref, R., Ur, S.: Deadlocks: From exhibiting to healing. In: Leucker, M. (ed.) RV 2008. LNCS, vol. 5289, pp. 104–118. Springer, Heidelberg (2008)
31. Nonaka, Y., Ushijima, K.: A run-time deadlock detector for concurrent java programs. In: Eighth Asia-Pacific Software Engineering Conference, APSEC 2001, pp. 45–52. IEEE (2001)
32. Oracle Corporation: Java Bug Database (2013), `http://bugs.sun.com/`
33. Oracle Corporation: Java Language Specification Java SE, 7th edn. (2013), `http://docs.oracle.com/javase/specs/jls/se7/jls7.pdf`
34. Oracle Corporation. jconsole (2013), `http://docs.oracle.com/javase/7/docs/technotes/guides/management/index.html`
35. Oracle Corporation. jstack (2013), `http://docs.oracle.com/javase/7/docs/technotes/tools/`
36. Oracle Corporation: NetBeans IDE (2013), `http://netbeans.org/`
37. Park, S., Vuduc, R., Harrold, M.: A unified approach for localizing non-deadlock concurrency bugs. In: 2012 IEEE Fifth International Conference on Software Testing, Verification and Validation, pp. 51–60. IEEE (2012)
38. Pun, K., Steffen, M., Stolz, V.: Deadlock checking by a behavioral effect system for lock handling. Journal of Logic and Algebraic Programming 81(3), 331–354 (2012)
39. Shanbhag, V.: Deadlock-detection in java-library using static-analysis. In: 15th Asia-Pacific Software Engineering Conference, APSEC 2008, pp. 361–368. IEEE (2008)
40. Shimomura, T.: Easy, Enjoyable, Effective E-Learning. Nova Science Publishers, Inc. (2008)
41. Shimomura, T., Ikeda, K.: Extensible web-based learning architecture. In: The 2012 International Conference on Software Engineering Research and Practice, pp. 1–7 (2012)
42. Shimomura, T., Ikeda, K.: Waiting blocked-tree type deadlock detection. In: Proceedings of Science and Information Conference 2013 (SAI 2013), pp. 45–50 (2013)
43. Wang, Y., Kelly, T., Kudlur, M., Lafortune, S., Mahlke, S.: Gadara: Dynamic deadlock avoidance for multithreaded programs. In: Proceedings of the 8th USENIX Conference on Operating Systems Design and Implementation, pp. 281–294. USENIX Association (2008)
44. Wen, Y., Zhao, J., Huang, M., Chen, H.: Towards detecting thread deadlock in java programs with jvm introspection. In: 2011 IEEE 10th International Conference on Trust, Security and Privacy in Computing and Communications (TrustCom), pp. 1600–1607. IEEE (2011)
45. Williams, A., Thies, W., Awasthi, P.: Static deadlock detection for java libraries. In: Gao, X.-X. (ed.) ECOOP 2005. LNCS, vol. 3586, pp. 602–629. Springer, Heidelberg (2005)
46. Zhang, W., Sun, C., Lu, S.: Conmem: detecting severe concurrency bugs through an effect-oriented approach. ACM SIGARCH Computer Architecture News 38(1), 179–192 (2010)

Exploring Eye Activity as an Indication of Emotional States Using an Eye-Tracking Sensor

Sharifa Alghowinem[1,4], Majdah AlShehri[2],
Roland Goecke[3,1], and Michael Wagner[3,1]

[1] Australian National University, Canberra, Australia
[2] King Saud University, Riyadh, Saudi Arabia
[3] University of Canberra, Canberra, Australia
[4] Ministry of Higher Education, Kingdom of Saudi Arabia

Abstract. The automatic detection of human emotional states has been of great interest lately for its applications not only in the Human-Computer Interaction field, but also for its applications in psychological studies. Using an emotion elicitation paradigm, we investigate whether eye activity holds discriminative power for detecting affective states. Our emotion elicitation paradigm includes induced emotions by watching emotional movie clips and spontaneous emotions elicited by interviewing participants about emotional events in their life. To reduce gender variability, the selected participants were 60 female native Arabic speakers (30 young adults, and 30 mature adults). In general, the automatic classification results using eye activity were reasonable, giving 66% correct recognition rate on average. Statistical measures show statistically significant differences in eye activity patterns between positive and negative emotions. We conclude that eye activity, including eye movement, pupil dilation and pupil invisibility could be used as a complementary cues for the automatic recognition of human emotional states.

Keywords: Affective computing, eye tracking, emotion recognition.

1 Introduction

Affective computing – the study of automatic recognition of human emotional states and their utilisation in a computer system – has had much interest lately due to its multidisciplinary applications. For example, Human-Computer Interaction (HCI) is concerned with enhancing the interactions between users and computers by improving the computer's understanding of the user's needs, which include understanding the user's emotional state [23]. In the education field, understanding the affective state of a student could lead to more effective presenting style and improved learning [7]. A current interest is in the personalisation of commercial products, which could be enhanced by understanding the client's preference based on their mood [31]. Moreover, such understanding of the user's emotions could enhance other applications such as virtual reality and smart surveillance [29]. Such automatic recognition of emotions could also be useful to support psychological studies. For example, such studies could give a baseline for the emotional reaction of healthy subjects, which could be compared and used to diagnose mental disorders such as autism [14] or depression [1].

L. Chen, S. Kapoor, and R. Bhatia (eds.), *Intelligent Systems for Science and Information,*
Studies in Computational Intelligence 542,
DOI: 10.1007/978-3-319-04702-7_15,

Eye-tracking applications cover several domains, such as psychology, engineering, advertising, and computer science [9]. As an example, eye-tracking techniques have been used to detect driver fatigue [15]. Moreover, cognitive load of a learner could be determined using eye-tracking and pupil measuring [4,18]. Moreover, some studies have been conducted on investigating eye responses on emotional stimuli [5,24,25].

In this study, we investigate whether eye activity holds discriminative power for recognising the emotional state, i.e. positive and negative emotions. We extract features for eye movements, pupil dilation and pupil invisibility to the eye tracker using a Tobii X120 eye tracker. We examine the performance of these features using artificial intelligence techniques for classification on an emotional stimulation experiment of 30 young adult and 30 mature adult subjects. Beside the machine learning techniques, we also analyse the statistical significance of the features between the two emotion and age groups. To separate memory from emotional effect on eye activities, we also compare eye activity of participants who have seen the positive clip of participants who have not.

The remainder of the paper is structured as follows. Section 2 reviews related background literature on using pupil features in affective studies. Section 3 describes the methodology, including the data collection, feature extraction and both statistical and classification methods. Section 4 presents the results. The conclusions are drawn in Section 5.

2 Background

Psychology research on pupil dilation has shown that not only light contributes to the pupil's response, but also memory load, cognitive difficulty, pain and emotional state [4]. Only few studies have been conducted for investigating eyes' response to emotional stimuli [5,13,24,25]. In the [13] study, it has been found that extreme dilation occurs for interesting or pleasing stimuli images. In [24,25], while their pupil dilation was recorded, subjects listened to auditory stimuli of three emotional categories: neutral, negative and positive. In [25], it has been found that pupil size was significantly larger during both emotionally negative and positive stimuli than during neutral stimuli. Using more controlled stimuli in [24], results showed that the pupil size was significantly larger during negative highly arousing stimuli than during moderately arousing positive stimuli. Another study [5] monitored pupil diameter during picture viewing to assess effects, showing that pupil size was larger when viewing pleasant and unpleasant emotional pictures.

One of the main issues with pupil variation measurement is the elimination of variation of pupillary light reflex, including materials with colour, luminance, and contrast. A review and suggestions of pupillary studies to eliminate luminance effects is presented in [11]. Avoiding visual stimuli variations, [24] and [25] used auditory stimuli. Another study used several statistical normalisation methods to minimise the variations in luminance, such as averaging pupil size and principal component analysis (PCA) [22]. Another way to fix this problem is to design a control paradigm that is identical for each subject to reduce variability [22]. Associating pupil measurement with other psychophysical measures, such as skin conductance, heart rate and brain signals could also be used to validate the reason of pupillary response [18]. In this study, we use both statistical normalisation methods and eliminate luminance effects (as far as possible).

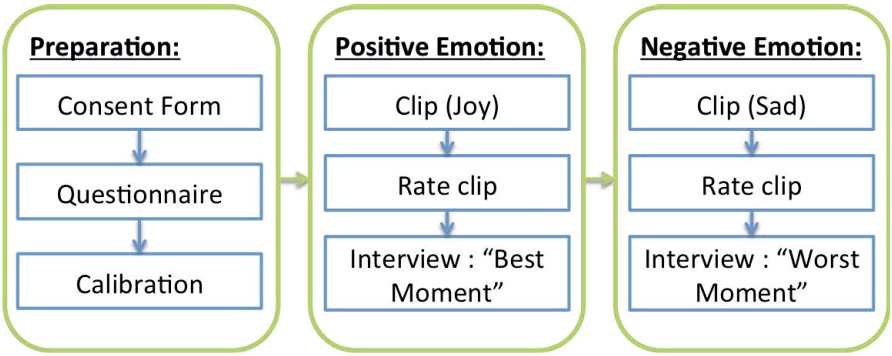


Fig. 1. Emotion eliciting paradigm and data collection process

However, there has been little research on eye activity that accompanies emotional responses to stimuli [16,28]. A study comparing eye activity as a response to positive and negative pictures, found greater eye blink reflex and corrugator muscle activity when viewing negative pictures [28]. In the [16] study, a decrease on eye blink and corrugator activity was found while suppressing negative emotions. It has been suggested that eye activity can help predict subjective emotional experience [21]. Therefore, in the current study, not only pupil dilation, but also the eye activity will be investigated to recognise the emotional state.

To the best of our knowledge, using an eye tracker sensor to analyse eye activity, pupil dilation and invisibility to the eye tracker is novel for the task of automatic emotion recognition. In this paper, we use a Tobii X120 eye tracker to extract eye activity features in order to analyse the differences between negative and positive emotions showing in eye behaviour in an emotion eliciting experiment. Beside analysing the general differences in eye movement, we specifically investigate the influence of age in the classification of emotions by comparing two age groups. After extracting features from each frame, we use a hybrid classifier where we build a Gaussian Mixture Model (GMM) for each subject's emotion, then feed it to Support Vector Machines (SVM) for classification. We also analyse the eye activity features statistically to identify how eye movement differs based on emotional state.

3 Method

A basic framework is designed to elicit positive and negative emotions only, using two video clips and two interview questions. The paradigm of our data collection is shown in Figure 1.

3.1 Emotion Elicitation Paradigm

Induced Emotions. Video clips have proven to be useful in inducing emotions [12] and been used for several emotional studies [30,17]. A universal list of emotional clips

(a) “Heidi” (joy) clip

(b) ‘Nobody’s Boy: Remi” (sad) clip

Fig. 2. Sample video frames from the most highly rated video clips demonstrating positive (joy) and negative (sad) emotions

is available [12]; however; these clips are in English from ‘Western society’ movies. Considering cultural and language differences between the Western countries and Arab countries, it is possible that some of the validated clips, even when dubbed, will not obtain similar results. Moreover, using Arabic subtitles in those clips was not an option, since the measurement of the eye activity will be jeopardised. Given the unique culture of Saudi Arabia, where the study was conducted, and to ensure acceptance of all participants, an initial pool of 6 clips inducing positive and negative emotions was selected from classic cartoon animation series dubbed in Arabic. A basic survey to rate the emotion induction from those 6 clips was conducted on 20 volunteers. Those volunteers were not included in the latter eye activity data collection. The most highly rated video clips demonstrating positive (joy) and negative (sad) emotions were selected, namely: “Heidi” and “Nobody’s Boy: Remi”, respectively (see Figure 2). The two selected clips had almost similar duration ($\sim 2.5min$). The positive (joy) emotion clip shows a scene of rich depiction of nature and landscape where Heidi breathed fresh mountain air, felt the warmth of the sun on her skin, and happily met goatherd Peter. The negative (sad) emotion clip shows a scene of Remi learning the terrible truth of his beloved master Vitalis’s death.

Spontaneous Emotions. Beside inducing emotions, watching video clips served as a preparation of the participant’s mood for the subsequent spontaneous emotion recording part. To gain spontaneous emotions, participants were interviewed about emotional events in their life. That is, after watching the positive emotion clip, the participants were asked about the best moment in their life. For negative emotion, after watching the negative emotion clip, the participants were asked about the worst moment in their life.

3.2 Participants

The data collection procedure recruited 71 native Arabic speakers from a convenience sample (65 females, 6 males). The participants’ age ranged from 18 to 41 years ($\mu = 25.6, \sigma = 4.8$). As regular participants’ mood and mental state are important in the

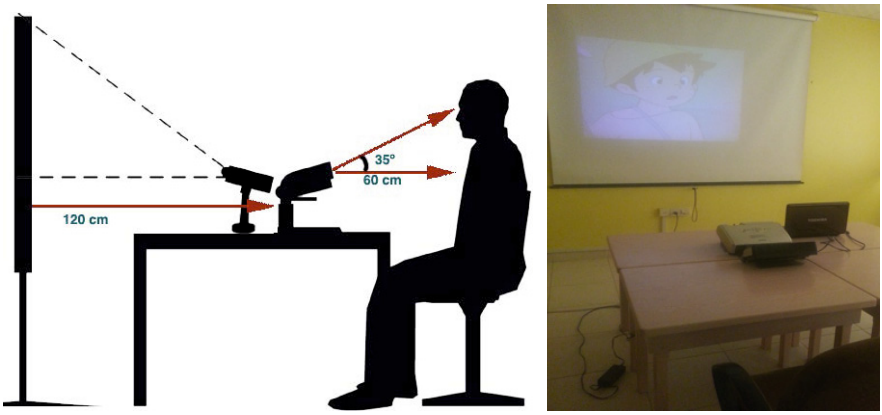


Fig. 3. Recording setup and environment

study, participants were asked about any current or history of mental conditions and about their usual mood: None of the participants had a mental disorder, 72% of the participants reported they are in a neutral mood, 7% always sad, and 22% always happy.

In this experiment, only 60 native female Arabic speakers were selected from the total recruited sample (30 young adults, and 30 mature adults), to insure age balance and reduce gender difference variability. The young adult participants' age ranged from 18 to 24 years ($\mu = 21.6, \sigma = 1.12$), while the mature adults' age ranged from 25 to 41 years ($\mu = 29.5, \sigma = 3.8$). Out of the selected subjects, 65% of participants had normal vision, the rest were using either glasses or lenses for correction.

3.3 Hardware and Recording Environment Settings

We used a Tobii X120 eye-tracker, attached to a Toshiba Satellite L655 laptop. We used a PowerLite 1880 XGA Epson projector screen as an extended monitor to the laptop, to ensure that the participant looked at similar coordinates while watching the clips and while talking to the interviewer. While the participant watch the clips, the interviewer leaves the room to reduce distraction and to allow the participant to freely watch the clips. The interviewer enters the room for the interview question and locates themselves in the middle of the projector screen. The screen resolution and distance from the projector screen and the eye tracker was fixed in all sessions. Although we had limited control over light in the recording room, we normalised the extracted features for each segment of each participant to reduce the light variability coming from the video clips themselves and the room light (see Figure 3).

3.4 Procedure

Consent and also a general demographic questionnaire asking about age, cultural heritage, physical and mental health, etc. were obtained prior to enrolling subjects in the study. Subjects were briefed about the study and were tested individually. Before the

beginning of the experiment, the subjects were instructed that they were going to watch the clips (without mentioning the specific type of emotion state) and told that the all film clips will be in Arabic. They were asked to watch the film as they would normally do at home and were told that there would be some questions to answer afterwards about the film clip and about their feelings. The eye movements of the subject were calibrated using a five-point calibration. This calibration was checked and recorded and upon successful calibration the experiment was started. Subjects were shown the instruction element screen asking them to clear their mind of all thoughts and then the clip began. After each clip, a post-questionnaire was done asking whether the subject had seen the clip previously, to investigate pupillary response due to memory activity. Moreover, to validate the induced emotion from the clips, participants were asked to rate the emotional effect of each clip in 11-points scale as: 'none (score: 0) to 'extremely (score: 10). Once the recording is over, subjects were thanked and no incentives were given.

Normalisation. To eliminate variability of pupil response not caused by emotional response, several aspects were considered:

- The design of the collection paradigm was controlled to be identical for each subject to reduce variability. The same clips were shown to each subject and also same interview questions asked.
- The screen resolution and distance from the projector screen and the eye tracker was fixed in all sessions. Further, recordings were done in the same room with similar daylight conditions.
- A projector screen was selected over a monitor screen, as once the interview starts, the interviewer locates themselves in the middle of the projector screen, to be in a similar position as the clips and to the eye-tracker.
- Moreover, percentile statistical normalisation methods have been applied to the extracted features from each subject to reduce within subjects variability as described bellow.

Preparation for Analysis. For each subject recording, clip watching and interview questions tasks of both positive and negative emotions were segmented using Tobii Studio (version 2.1). Having a total of 4 segments per subject, we extract raw features from each segment. To get clean data, we exclude frames where the eyes were absent to the eye tracker. The absence of the eyes is determined by the confidence level of the eye tracker of each eye (range 1 to 4). We only select frames where both confidence level of both eyes equal to 4.

3.5 Feature Extraction

Low-level Features. Excluding frames where the eyes were not detected by the eye tracker, we calculated 9 features per frame (30 frames per second) of raw data extracted from the Tobii eye tracker, as follow:

- Distance between eye gaze points position from one frame to the next for each eye, and its speed (Δ) and acceleration (Δ,Δ) were calculated to measure the changes

in eye gaze points; the longer the distance the faster the eye gaze change (2×3 features).
- Difference between the distance from left eye and the eye tracker and the distance from right eye and the eye tracker were calculated, to approximately measure head rotation (1 features).
- Normalised pupil size for each eye, to measure emotional arousal (2×1 features).

Statistical Features. Over the low-level features mentioned above, we calculated 147 statistical functional features to measure the pattern of eye activity, those features are:

- The mean, standard deviation (std), variance (var), maximum, minimum and range for all low-level features mentioned above (6×9).
- Even though, blink features are not available in Tobii X120, we measured the absence of the pupil in the video frames. Absence of left pupil only indicates left head rotation, and vice versa. Absence of both pupils could represent blinks, occluded eyes or head rotation being out of eye tracker range such as extreme looking up/down or left/right. We measure the mean, standard deviation (std), and variance (var) of the absence of left, right and both pupils (3×3)
- We also calculate several statistical features such as maximum, minimum, range and average of the duration, as well as its rate to total duration and count of occurrence of:
  - fast and slow changes of eye gaze for each eye ($6 \times 2 \times 2eyes$).
  - left, and right head rotation, calculated from the differences of the distance from both eyes and the eye tracker (6×3).
  - Large and small pupil size for each eye ($6 \times 2 \times 2eyes$).
  - The absence of left, right and both eyes (6×3)

The above duration features are detected when the normalized feature in question is higher than a threshold. The threshold is the average of the normalized feature in question plus the standard deviation of that feature for each segment.

3.6 Statistical Test

In order to characterise the eye activity patterns, the extracted statistical functionals from positive and negative emotions were compared. A two-tailed T-test was used for this purpose. In our case, the two-tailed T-tests for two samples were obtained assuming unequal variances with significance $p = 0.05$. The state of the T-test were calculated to identify the direction of effect.

3.7 Classification and Evaluation

For the low-level features, a Gaussian Mixture Model (GMM) with 8 mixture components was created for each segment for each participant. In this context, the GMM serves as dimensionality reduction, as well as a hybrid classification method [2]. The Hidden Markov Model Toolkit (HTK) was used to implement a HMM using one state to train the GMM models. In this work, diagonal covariance matrix was used, and the number

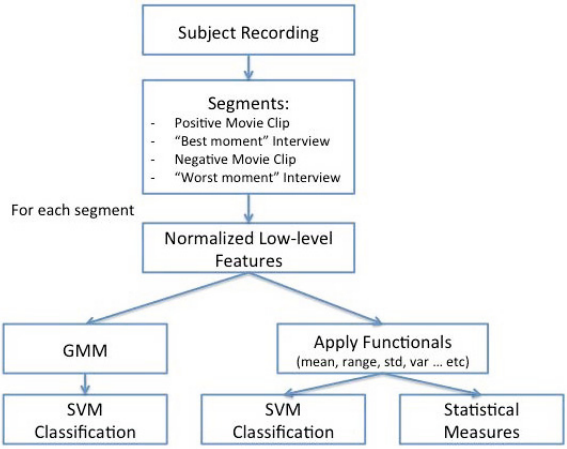


Fig. 4. Structure and Steps of the System

of mixtures was chosen empirically then fixed to ensure consistency in the comparison. This approach was beneficial to get the same number of values of the extracted features that to be fed to the Support Vector Machine (SVM) regardless of the duration of the participant's segment. The means, variance and weight of the 8 mixtures of GMM were used as a supervector that have been fed to the SVM classifier for each subject.

To test the effect of the eye activity patterns in each emotion (positive and negative) on the classification results, SVM was used on the statistical functional mentioned earlier. Comparing the low-level features modelling with statistical functional features classification is also beneficial to identify best modeling method for the task.

The segments for all participants were classified in a binary subject-independent scenario (i.e. positive / negative) using Support Vector Machine (SVM), which can be considered as a state-of-the-art classifier for some applications since it provides good generalisation properties. In order to increase the accuracy of the SVM, the cost and gamma parameters need to be optimized. In this paper, we used LibSVM [6] to implement the classifier, with a wide range of grid search for the best parameters. To mitigate the effect of the limited amount of data, a leave-one-subject-out cross-validation was used, without any overlap between training and testing data.

The main objective was to correctly classify the segment of each subject as positive or negative based on the eye activity patterns. The performance of a system can be calculated using several statistical methods, such as recall or precision. In this paper, the average recall (AR) was computed. Figure 4 summarize the general structure and the steps of our system.

3.8 Feature Selection

In order to maximise the recognition rate, manual and automatic feature selection were experimented with on the statistical features. Manual selection is based on the statistical tests mentioned above, as we manually select features that passes the T-test from

mature, young and all participants groups. For automatic feature selection, principal component analysis (PCA) [26] is a dimensionality reduction method, where high-dimensional original observations are projected to lower dimensions called principal components, which maximize the variance. As a result, the first principal component has the largest possible variance and so on for the next principal components. In this study, we perform a PCA on the statistical features, then used only the first 20% of the principal components for the classification, to be comparable (in number of features) with the manual feature selection.

4 Results

4.1 Initial Observations

Due to ethic restrictions in King Saud University regarding video-recording of participants, observations have been made only by the interviewer at the time of the interview and were not recorded. Regarding negative emotions, while watching the clip, 39% of participants rated the clip to have strong affect (more than 8 out of 10), though only almost 1% cried over the clip. On the other hand, while answering negative emotion interview question, 70% of the participants cried (including one male participant). Since the negative clip shows a death scene, almost 85% participants talked about their negative emotion during losing a loved person in their life. Other topics included injustice, failure, and conflict with a close person. Those late findings, indicate that watching the video clips prepared the participant mood for the spontaneous emotions in the interview. Since the number of male participants was not enough to make any reliable gender comparisons, more data needs to be collected. For the positive emotion, while watching the movie clip, 53% of participants rated the clip to have strong affect (more than 8 out of 10). On the other hand, while answering the positive emotion interview question, only 0.7% of the participants cried while expressing their joy (none of which were males). Our observations, indicate that unlike happiness crying, sadness crying was associated with eye contact avoidance.

As mentioned earlier, subjects were asked if they have seen the clip before the experiment. For the joy clip, 57% of the participants have seen the clip before. On the other hand, only 10% have seen the sad clip. Therefor, and to examine how the memory might effect the eyes activity, participants who have seen the joy clip were compared with participants who have not seen the clip, as can be seen later on. Figure 5 shows the number of participants who have seen the joy clip of each group.

Moreover, participants were asked to rate how much each clip had an affect on their feelings in scale from 0-10, given that 10 is a high affect and 0 has no affect. The joy clip got an average of 8 points as positive affect, and the sad clip got an average of 7 points as negative affect (see Figure 6).

4.2 Memory Effect on Eye Activity

In order to find out how memory could affect the eyes activity, participants were asked if they have seen each clip before. As mentioned earlier, 57% of the participants have

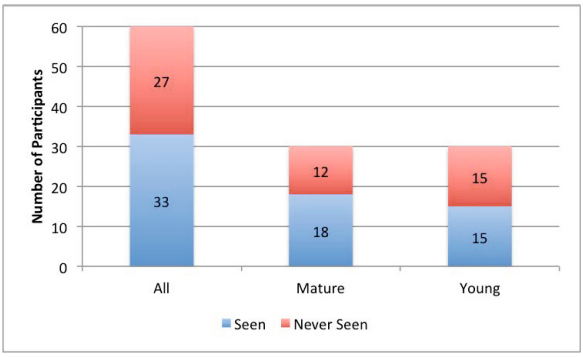


Fig. 5. Average rating of 'Joy' and 'Sad' clips of effectiveness in Inducing Emotions

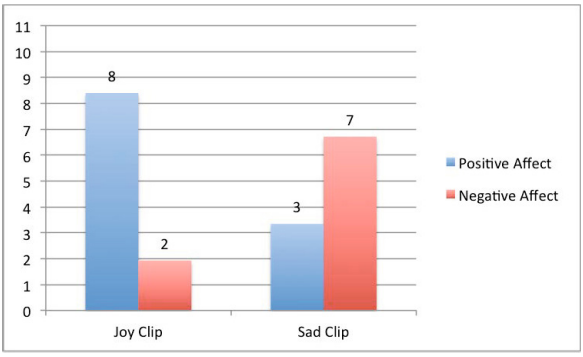


Fig. 6. Average rating of 'Joy' and 'Sad' clips of effectiveness in inducing emotions

seen the joy clip before, while only 10% have seen the sad clip. Since the number of participants who have seen the sad clip is very small, a comparison using the sad clip is not feasible. On the other hand, number of participants who have seen the joy clip is comparable to the ones who have not seen it. We investigated the differences between the two groups statistically and using artificial intelligent techniques.

Statistical Analysis. While comparing statistical features between participants who have seen the joy clip from participants who have not, only few features passed the t-test (see Table 1). We expected to get at least a slight difference in pupil size with memory provocation, as reviewed in [19]. However, it was not the case. The lake of significance might be due to the period since the clip have been seen. As the joy clip is extracted from classical animation series that was first aired in Saudi Arabia in 1983, and continued to air every two years till 1998 [8]. On the other hand, the only features that had significant differences was the speed and acceleration of changing gaze direction, being slower for participants who have seen the clip. This finding could support that memory for a

Table 1. Significant T-tests results of eye activity features Comparing Seen (S) vs. Unseen (N) Joy Clip for All Participants

Emotion	Feature	Direction	P-value
Joy Clip	Speed range of left eye gaze	S < N	0.050
	Speed range of right eye gaze	S < N	0.050
	Acceleration average of left eye gaze	S < N	0.040
	Acceleration average of right eye gaze	S < N	0.030

scene could have longer fixation duration and shorter saccade length, as this is still a controversial study, as reviewed in [27], more investigation is needed.

Classification Results. Classifying participants who have seen the clip against participants who have not, the average recall results were almost chance level (Table 2). Low-level features and functional features classification, as well as feature selection method are investigated to for any eye activity pattern differences from participants who have seen the clip against the ones who have not. The classification results supports our statistical analysis as there are no differences found by the classifiers. That finding might be due that the participants have seen the clip very long time ago, and they no longer remember the clip.

Table 2. Classification Average Recall Results in(%) for Participants Who Have Seen vs. Who Have Not Seen the Joy Clip

	Joy Clip		
Features	All Participants	Mature Adults	Young Adults
Low-level Features	48.7	37.5	41.7
All Statistical Features (146)	58.3	56.8	53.6
PCA Selection of Statistical Features (20%)	52.5	55.0	56.2

4.3 Emotional Effect on Eye Activities

In our paradigm, positive and negative emotions were elicited in two ways: using movie clips that induce emotions and by spontaneously answering questions about emotional events. In order to investigate the effect of elicited emotions on eye activity, we compared eye movements while experiencing positive and negative emotion statistically as well as artificial classification as follows.

Statistical Analysis. Regarding induced emotions from watching video clips, some statistical features were significant as shown in Table 3. The maximum and range of left and right pupils' size were larger in induced negative emotions. This finding is inline with [24], where it has been found an increase in pupil size while listening to sound containing negative emotion. Even though blinking rate or duration could not be extracted using the eye tracker, we extracted the absence of the pupils in the video frames. Absence of both pupils could represent blinks, occluded eyes or head rotation being out of eye tracker range such as extreme looking up / down or left / right. Calculating the

Table 3. Significant T-test results of eye activity features comparing positive (P) vs. negative (N) emotions for all participants

Emotion	Feature	Direction	P-value
Induced	Maximum left pupil size	P < N	0.027
	Range of left pupil size	P < N	0.010
	Maximum right pupil size	P < N	0.026
	Range of right pupil size	P < N	0.012
	Average absence of both pupils	P < N	0.005
	Absence duration rate of both pupils	P < N	0.005
	Absence duration range of both pupils	P < N	0.038
	Maximum absence duration of both pupils	P < N	0.038
Spontaneous	Maximum duration of left head rotation	P < N	0.053
	Maximum duration of large left pupil size	P < N	0.014
	Maximum duration of large right pupil size	P < N	0.002
	Maximum absence duration of left pupil	P < N	0.035
	Absence duration range of left pupil	P < N	0.047
	Maximum absence duration of right pupil	P < N	0.004
	Absence duration range of right pupil	P < N	0.015
	Maximum absence duration of both pupils	P < N	0.017
	Absence duration range of both pupils	P < N	0.027

differences in the absence of the pupils for both positive and negative emotions could be used as an indicator of emotions. For example, sadness is characterised by downward eyebrows and head [10]. In our study, the statistical functional of absence duration of both pupils was significantly longer in negative emotions, which might be a sign of extreme head rotation, blinks, or occluded eyes. This finding is supported by the general characteristic of expressing sadness [10].

Table 3 shows some of the significant statistical features while expressing positive and negative spontaneous emotion in the interview. In this study, we found that the maximum duration of left head rotation (measured by the different in the distance between the eyes and the eye tracker) is significantly longer while expressing spontaneous negative emotions. That could be explained by avoiding eye contact, as there is evidence that during discussing personal topics and during speech hesitation there is less eye contact [3]. Regarding pupil size, we found that the maximum duration of large pupil size for both left and right eye were significantly larger in spontaneous negative emotions, which is inline with our finding in induced emotions. As in induced emotions, the absence duration of left, right and both pupils are measured. Range and maximum absence duration of left, right and both pupils were longer while expressing negative emotions, indicating extreme head movements and occluded eyes. Giving that more than %70 of the participants cried while being interviewed in the negative emotion questions, the absence of both pupils are explained by the occluded eyes.

Classification Results. Automatic classification of positive and negative emotions from eye activity features were performed and the results are shown in Table 4. In general, the recognition rate of emotion types from eye activities was reasonable, giving 66% on average. This indicate that eye activities could be used as a complementary cue in identifying emotional states. Although both induced and spontaneous emotions have relatively reasonable recognition rates (65.3% and 66.7% on average, respectively), generally while expressing spontaneous emotions the average recognition

Table 4. Average recall results in (%) for positive vs. negative emotion classification

	Induced Emotions			Spontaneous Emotions		
Features	All Participants	Mature-adults	Young-adults	All Participants	Mature-adults	Young-adults
Low-level Features	60.8	66.4	62.3	65.1	68.5	68.4
Manual Selection of Statistical Features (passed T-test)	68.3	68.3	75	65.8	68.3	71.6
PCA automatic selection of Statistical Features (20%)	60.1	60	66.6	64.2	63	65

rate of positive and negative emotions is slightly higher than induced emotions. This finding indicates that spontaneous emotions might have stronger eye activity patterns than induced emotions.

Unlike statistical features, low-level features modelled only eye activity excluding pupils invisibility to the eye tracker, where the pupils of either eyes were not detected. Regardless of the high dimensionality, modelling functional features performed approximately equivalent to low-level features, if not better. Current findings indicate that statistical modeling loses less information compared to the low-level modeling. Comparing manual (T-test) and automatic (PCA) feature selection of the functionals, the manual selection based on the T-tests results was better than using the automatic selection with PCA.

Previous studies have investigated age effects on emotion intensity, finding a reduction in intensity between young adult and middle aged groups [20]. In our study, mature adults sustained almost equivalent recognition rates for both induced and spontaneous emotions. On the other hand, young adults' recognition rate for both induced and spontaneous emotion was significantly higher than mature adults. The sustained recognition rate in the spontaneous emotion for mature adults might be explained by the adaptation, i.e. older people have been exposed to more emotional incidents and, thus, been adapted [20]. Regarding young adults, the finding is not surprising as in [20] it was expected to have better recognition rate in young adults for both induced and spontaneous emotions.

5 Conclusions

Automatic detection of human emotional states has had much interest lately for its applications not only in HCI, but also for its applications in psychological studies. Given the rich source of the eye activities regarding the emotional state, we investigated whether eyes activities, including movements, and pupil dilation or invisibility to the eye tracker hold discriminative power for detecting emotional states.

A framework to elicit positive and negative emotions was implemented using two video clips for induced emotions and using interview for spontaneous emotions. We collected eye movements from 71 participants, were only 60 female participants were selected for the analysis to reduce gender variability. We extracted several low-level features as well as applying statistical functionals over them. We analyzed statistical functional features statistically as well as using artificial technique classifications.

In general, the classification results using eye activities were reasonable, giving 66% on average, proving that eye activities holds effective cues in identifying emotional states. Generally while expressing spontaneous emotions, the recognition rate of positive and negative emotions is slightly higher than induced emotions. This finding indicates that

spontaneous emotions might have stronger eye activity patterns than induced emotions. Comparing low-level eye activity features with statistical features, we found that statistical features modeling is as good in capturing information compared to the low-level modelling, therefore, performed better in recognising emotional states. We also investigate manual and automatic feature selection, the different between classification results from both methods were statistically significant, we conclude that manual selection based on T-test performs better.

We found that pupil dilation size and duration increase while expressing negative emotions. We also found less eye contact, explained by head rotation. Calculating the absence of pupils in the video frames indicates extreme head rotation away from the camera and occluded eyes was longer while expressing negative emotions.

Investigating the age effect on expressing emotions, we compared young and mature adults. While mature adults sustained almost equivalent recognition rates for both induced and spontaneous emotions, the young adults' recognition rates for induced and spontaneous emotion were significantly higher than mature adults. Moreover, the sustained recognition rate in mature adults with induced and spontaneous emotions, might be due to adaption.

To separate memory from emotional effect on eye activities, we compared eye activities from participants who have seen the joy clip from participants who have not. The classification results were almost chance level, as there were no differences from the two groups. Statistical analysis showed no significant different in all features but slower speed of change gaze direction for participants who have seen the clip. As this finding is still controversial in the psychology literature, more investigation is needed using more participants.

We conclude that eye activity including eye movement, pupil dilation and invisibility to the eye tracker could be a complementary cues for the automatic recognition of human emotional states.

6 Limitation and Future Work

A known limitation in this study is the lack of an official rated list of eliciting emotion Arabic clips. A separate study investigating affective Arabic clips is needed for such cultural emotional studies. Future work in this current collected database will analyse speech and investigate its correlation with eye activities. The current database collection was limited to voice recording and eye tracking, future collection will expand not only on the number of participants, but also on using more sensing devices such as skin conductivity, temperature, heart rate, and Brain-Computer Interfaces (BCI) including measuring eye blinking duration and rate.

Acknowledgment. The authors would like to acknowledge the Deanship of Scientific Research at King Saud University for funding the work through the research group project Number RGP-VPP-157.

References

1. Alghowinem, S., Goecke, R., Wagner, M., Epps, J., Breakspear, M., Parker, G.: From Joyous to Clinically Depressed: Mood Detection Using Spontaneous Speech. In: Proc. FLAIRS-25 (2012)
2. Alghowinem, S., Goecke, R., Wagner, M., Epps, J., Breakspear, M., Parker, G.: A Comparative Study of Different Classifiers for Detecting Depression from Spontaneous Speech. In: IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP) (1), pp. 26–31 (2013)
3. Argyle, M., Dean, J.: Eye-contact, distance and affiliation. Sociometry, 289–304 (1965)
4. Beatty, J.: Task-evoked pupillary responses, processing load, and the structure of processing resources. Psychological Bulletin 91(2), 276 (1982)
5. Bradley, M.M., Miccoli, L., Escrig, M.A., Lang, P.J.: The pupil as a measure of emotional arousal and autonomic activation. Psychophysiology 45(4), 602–607 (2008)
6. Chang, C.C., Lin, C.J.: Libsvm: a library for svm, pp. 1–30 (2001), `www.csic.ntu.edu.tw/rcjlin/papers/lib.svm` (March 4, 2006)
7. Craig, S., Graesser, A., Sullins, J., Gholson, B.: Affect and learning: an exploratory look into the role of affect in learning with autotutor. Journal of Educational Media 29(3), 241–250 (2004)
8. Ministry of Culture, A, Information.: Saudi arabia channel one (2013), `http://www.sauditv1.tv/`
9. Duchowski, A.T.: A breadth-first survey of eye-tracking applications. Behavior Research Methods 34(4), 455–470 (2002)
10. Frijda, N.H.: The emotions. Cambridge University Press (1987)
11. Goldwater, B.C.: Psychological significance of pupillary movements. Psychological Bulletin 77(5), 340 (1972)
12. Gross, J.J., Levenson, R.W.: Emotion elicitation using films. Cognition & Emotion 9(1), 87–108 (1995)
13. Hess, E.H.: Pupillometrics: A method of studying mental, emotional and sensory processes. In: Handbook of Psychophysiology, pp. 491–531 (1972)
14. Hobson, R.P., Ouston, J., Lee, A., et al.: Emotion recognition in autism: Coordinating faces and voices. Psychological Medicine 18(4), 911–923 (1988)
15. Horng, W.B., Chen, C.Y., Chang, Y., Fan, C.H.: Driver fatigue detection based on eye tracking and dynamk, template matching. In: 2004 IEEE International Conference on Networking, Sensing and Control, pp. 7–12. IEEE (2004)
16. Jackson, D.C., Mueller, C.J., Dolski, I., Dalton, K.M., Nitschke, J.B., Urry, H.L., Rosenkranz, M.A., Ryff, C.D., Singer, B.H., Davidson, R.J.: Now you feel it, now you don't frontal brain electrical asymmetry and individual differences in emotion regulation. Psychological Science 14(6), 612–617 (2003)
17. Jerritta, S., Murugappan, M., Nagarajan, R., Wan, K.: Physiological signals based human emotion Recognition: a review, pp. 410–415. IEEE (2011)
18. Kahneman, D., Tursky, B., Shapiro, D., Crider, A.: Pupillary, heart rate, and skin resistance changes during a mental task. Journal of Experimental Psychology 79(1p1), 164 (1969)
19. Laeng, B., Sirois, S., Gredebck, G.: Pupillometry: A window to the preconscious? Perspectives on Psychological Science 7(1), 18–27 (2012)
20. Larsen, R.J., Diener, E.: Affect intensity as an individual difference characteristic: A review. Journal of Research in Personality 21(1), 1–39 (1987)
21. Niemic, C.P., Warren, K.: Studies of emotion. A Theoretical and Empirical Review of Psychophysiological Studies of Emotion (Department of Clinical and Social Psychology). JUR Rochester 1(1), 15–19 (2002)

22. Oliveira, F.T., Aula, A., Russell, D.M.: Discriminating the relevance of web search results with measures of pupil size. In: Proceedings of the 27th International Conference on Human Factors in Computing Systems, pp. 2209–2212. ACM (2009)
23. Pantic, M., Pentland, A., Nijholt, A., Huang, T.: Human computing and machine understanding of human behavior: a survey. In: Proceedings of the 8th International Conference on Multimodal Interfaces, pp. 239–248. ACM (2006)
24. Partala, T., Jokiniemi, M., Surakka, V.: Pupillary responses to emotionally provocative stimuli. In: Proceedings of the 2000 Symposium on Eye Tracking Research & Applications, pp. 123–129. ACM (2000)
25. Partala, T., Surakka, V.: Pupil size variation as an indication of affective processing. International Journal of Human-Computer Studies 59(1), 185–198 (2003)
26. Pearson, K.: Liii. on lines and planes of closest fit to systems of points in space. The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science 2(11), 559–572 (1901)
27. Rayner, K.: Eye movements in reading and information processing: 20 years of research. Psychological Bulletin 124(3), 372–422 (1998)
28. Sutton, S.K., Davidson, R.J., Donzella, B., Irwin, W., Dottl, D.A.: Manipulating affective state using extended picture presentations. Psychophysiology 34(2), 217–226 (2007)
29. Tao, J., Tan, T.: Affective computing: A review. In: Tao, J., Tan, T., Picard, R.W. (eds.) ACII 2005. LNCS, vol. 3784, pp. 981–995. Springer, Heidelberg (2005)
30. Zeng, Z., Pantic, M., Roisman, G.I., Huang, T.S.: A survey of affect recognition methods: Audio, visual, and spontaneous expressions. IEEE Transactions on Pattern Analysis and Machine Intelligence 31(1), 39–58 (2009)
31. Zhou, F., Ji, Y., Jiao, R.J.: Affective and cognitive design for mass personalization: status and prospect. Journal of Intelligent Manufacturing, 1–23 (2012)

Finding Robust Pareto-optimal Solutions Using Geometric Angle-Based Pruning Algorithm

Sufian Sudeng and Naruemon Wattanapongsakorn

King Mongkut's University of Technology Thonburi,
Department of Computer Engineering,
126 Pracha-Uthit Rd. Bangmod, Thoongkhru Bangkok, Thailand
sufian@sudeng.org
naruemon@cpe.kmutt.ac.th
http://cpe.kmutt.ac.th

Abstract. Evolutionary multi-objective optimization algorithms have been developed to find a representative set of Pareto-optimal solutions in the past decades. However, researchers have pointed out that finding a representative set of Pareto-optimal solutions is not sufficient; the task of choosing a single preferred Pareto-optimal solution is also another important task which has received a widespread attention so far. In this paper, we propose an algorithm to help the decision maker (DM) choose the final preferred solution based on his/her preferred objectives. Our algorithm is called an adaptive angle based pruning algorithm with independent bias intensity tuning parameter (ADA-τ). The method begins by calculating the angle between a pair of solutions by using a simple arctangent function. The bias intensity parameter of each objective is introduced independently in order to approximate the portions of desirable solutions based on the DM's preferred objectives. We consider several benchmark problems including two and three-objective problems. The experimental results have shown that our pruning algorithm provides a robust sub-set of Pareto-optimal solutions for the benchmark problems.

Keywords: Multi-objective optimization, pruning algorithm, Pareto-optimal solutions.

1 Introduction

In real world problems, several objective functions have to be optimized simultaneously. The typical goal for such multi-objective optimization problems is to approximate the entire set of Pareto-optimal solutions and its image in objective space, the Pareto front. Over the years, evolutionary multi-objective optimization methodologies have adequately demonstrated their usefulness in finding a well-spread set of Pareto-optimal solutions. Evolutionary multi-objective optimization (EMO) procedures have been widely applied and gained a great attention in EMO community. However, recent works [2],[9],[11],[13] have discovered that the EMO methodologies (e.g. NSGA-II, SPEA-2) have faced difficulties in solving problems with a large number of objectives. The difficulties are as

L. Chen, S. Kapoor, and R. Bhatia (eds.), *Intelligent Systems for Science and Information,* Studies in Computational Intelligence 542,
DOI: 10.1007/978-3-319-04702-7_16,

follows : (1) visualization of the multi-dimensional objectives space is difficult which might restrict EMO methodologies from finding the entire Pareto-optimal set, (2) there is a need of an exponential number of points to represent a higher-dimensional Pareto-optimal solutions, and (3) The large number of objective functions may not produce enough multi-objective selection pressure. In real-world applications, the DM is not interested in the whole Pareto front since often the final decision is just a single best solution. The main goal of multi-objective optimization algorithms (MOEAs) is to assist the DM in selecting the final alternative which satisfies the most or all of his/her preferences. To simplify the decision making task, the DM can incorporate his/her preferences into the search process. These preferences are used to guide the search towards the preferred parts of Pareto front [1],[4],[8], called region of interest (ROI). The ROI is the preferred portion of the Pareto front from the DM's perspective. Several strategies have been developed to model the DM's preference information such as weight preference [7],[16], weight reference point [15], solution ranking [12] and so on. Some strategies have been developed in order to perform this in post Pareto-optimal phase such as presented in [3] and [11]. However, the set of desired solutions from many preference strategies are still reflect difficulties in choosing the final best solution. In addition, most of the preference strategies require some level of background knowledge from the DM. Considering these reasons, we propose a new pruning mechanism that can filter out the undesired solutions and provide more robust trade-off solutions to the DM.

The contributions of our work can be summarized as follows:
1. We propose a pruning algorithm which can be used in post Pareto-optimality analysis.
2. The main issue of our algorithm is to filter out marginally improved solutions. Only outstanding or significantly improved solutions are selected to be the non-dominated set.
3. The algorithm has independent objective biasing capability which allows the DM to prioritize each objective according to his/her preference.

The remainder of this article is organized as follows. In Section 2, we present several basic concepts of multi-objective optimization and multi-objective evolutionary algorithms. In Section 3, we present literature review of preference incorporation techniques solved with multi-objective evolutionary algorithms (MOEAs). In Section 4, we describe our pruning algorithm. The results are presented and discussed in Section 5. Finally, we give our research conclusions and suggestion on future research directions.

2 Multi-objective Optimization

Most often, the multiple objectives are in conflict and compete with each other. The decision maker (DM) has to decide on an individual solution based on certain preferences and objectives' priorities. In its general form, a multi-objective optimization problem can be formulated as follows

$$\text{“}Minimize\text{”} z = f(x)$$

Subject to

$$x \in X$$

where,

$$f(x) = (f_1(x), .., f_i(x), .., f_p(x))^T$$

p-vector of objective functions

$$x = (x_1, ..., x_n)^T$$

decision vector

$$X \subseteq R^n$$

feasible decision space

$$z = f(x)$$

objective vector

$$Z = f(X)$$

feasible objective space (solution space)

R^n might be restricted with constraints of the following types to form X:

$$g(x) \leq 0$$

inequality constraints

$$h(x) = 0$$

equality constrains

2.1 Decision Domain, Decision Space and Objective Space

The decision domain Ω and the constraints $g_i(x)$ and $h_k(x)$ define a feasible region A, where

$$A = x \in \Omega : g(x) \leq 0, h(x) = 0$$

Subject to all the points that do not belong to A, which constitute the infeasible region.

The objective function *F(x)* maps the decision vector from the decision space to the objective space. Thus, a feasible region is also defined in the objective space. Fig. 1 shows an example of an optimization problem with two-decision variables (x_1, x_2) and two-objective functions defined by:

$$F(x) = min([f_1(x), f_2(x)])$$

$$f_1(x) = x_1 + x_2, where 0 \leq x_1 \leq 2, f_2(x) = x_1^2 + x_2^2, where 0 \leq x2 \leq 1, X \in \Re$$

Fig.1 illustrates how the function *F(x)* maps the constrained decision space onto the objective space. In addition, it is possible to see that both objectives (f_1, f_2) are incompatible because there is no single solution that minimizes both objectives at once. In instead of having a single solution, there is a set of Pareto-optimal solutions depicted as a bold front in the objective space. Next section explain in detail the concept of Pareto-optimal solutions.

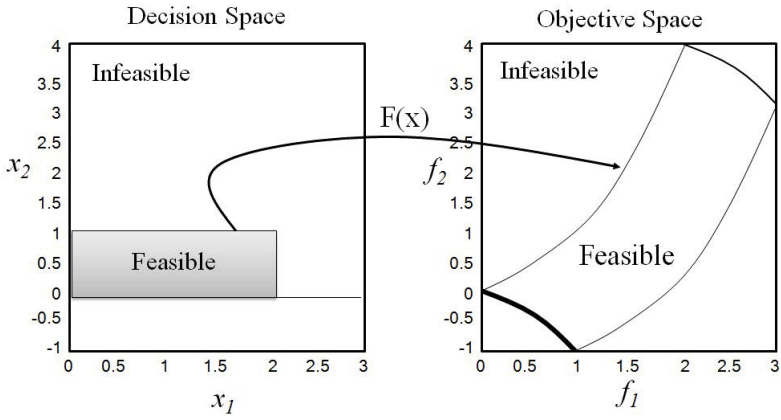


Fig. 1. Example of optimization problem

2.2 Pareto-optimality

In multi-objective problems, the concept of "dominance" is used to determine if one solution is better than others. A solution 'x' is said to dominate a solution 'y' if the following two conditions are true: (1) 'x' is no worse than 'y' in all objectives and (2) 'x' is better than 'y' in at least one objective. In this case 'y' is said to be "dominated" by 'x', or alternatively, 'x' is said to be "non-dominated" by 'y'. The concept of dominance is exemplified in a two-objective minimization example as shown in Fig. 2.

Since both functions are to be minimized, the following dominance relationships can be observed: solution 2 dominates solutions 1, 3 and 5; solution 3 only dominates solution 5 and solution 4 only dominates solution 5. Conversely, solutions 2 and 4 are non-dominated because there is no solution that dominates them.

Note that even if solution 2 is equal in one objective to solutions 1 and 3, it still dominates them, given the concept of dominance. The non-dominance

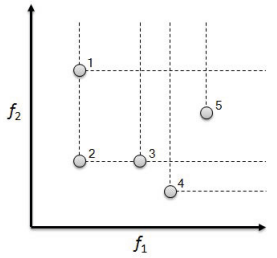


Fig. 2. Concept of dominance

relationship determines the concept of Pareto optimality. A solution is said to be **Pareto optimal** if it is non-dominated by any other solution. In other words, a Pareto optimal solution cannot be improved in one objective without losing in another one. In this case, solutions 2 and 4 are Pareto-optimal solutions. All solutions that are Pareto optimal constitute the **Pareto set**. The objective values of the Pareto set in the objective space constitute the **Pareto frontier**.

2.3 Multi-objective Evolutionary Algorithms

Evolutionary Algorithms (EAs) work with a number of solutions at a given time. EAs were early attempted to use in Multi-objective optimization problem focused on classical approaches (e.g., weighted-sum or ϵ -constrained). Then researchers soon started to prove novel algorithms that exploited the name of EA. The Vector Evaluated Genetic Algorithm (VEGA) developed in 1984 is considered the first multi-objective evolutionary algorithm.

After VEGA, the next decisive milestone was proposed by Goldberg [10] for the use of Pareto optimality as fitness criteria. In this case, the population is ranked in fronts (Pareto ranking). The non-dominated solutions obtain the highest rank (associated with highest fitness). The next front is given the second highest rank and so on. The Goldbergs' Pareto ranking is illustrated in Fig. 3. Goldberg also proposed the use of a "niching mechanism" to maintain the diversity of the solutions along the Pareto front.In niching mechanism, the fitness of each individual is modified according to the distance to its neighbors. Individuals that are too close to one another have their fitness reduced. Therefore, this approach is also called 'fitness sharing' as shown in the following equation.

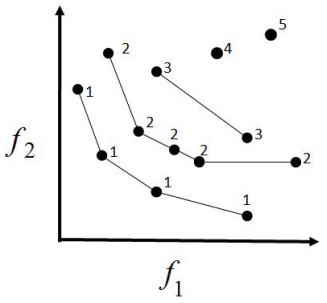


Fig. 3. Goldberg's Pareto Ranking

$$f_{si} = \frac{f_i}{\sum_{j=i}^{N} \oslash(d_{ij})} \tag{1}$$

where,

$$\oslash(d_ij) = \begin{cases} 1 - \frac{d_ij}{\sigma_share}, d_ij < \sigma_share \\ 0, otherwise \end{cases}$$

$f_s i$ is the shared fitness of f_i, N is the number of neighboring solutions, and $\oslash(d_i j)$ is a niche count, $d_i j$ indicated by the distance between solutions i and j, and $\sigma_s hare$ is the niche radius. Goldberg's theoretical background served as the basis for several MOEAs developed in later years. These are considered the first generation of MOEA including the Multi-objective Genetic Algorithm (MOGA) proposed in 1993, and the Non-dominated Sorting Genetic algorithm (NSGA) proposed in 1994. In 1999, the Strength Pareto Evolutionary Algorithm (SPEA) was proposed by Zitzler et al. started the second generation MOEA, and Deb et al. proposed the Non-dominated Sorting (NSGA). A year later, revised versions of SPEA and NSGA were launched and named as SPEA-2 [17] and NSGA-II [18], which have gained widespread attention and have begun to be applied to a diversity of practical problems and become state-of-the-art MOEAs.

Recently, many multi-objective optimization evolutionary algorithms (MOEAs) have been developed, a very well-known and high performance MOEA had been introduced, called MOEA/D, a decomposition based multi-objective evolutionary algorithm. The algorithm decomposes a multi-objective optimization problem into a number of scalar optimization sub-problems and optimize them simultaneously. Each sub-problem is optimized by using only information from its neighboring sub-problems. MOEA/D has proved to have lower computational complexity and higher performance than others, such as NSGA-II, SPEA2 and MOGLS (multi-objective genetic local search) [6]. We plan to incorporate our pruning algorithm with decomposition based MOEA in our future works, we then decide to approximate Pareto-optimal solutions using MOEA/D. The core issue of MOEA/D is a choice of an appropriate scalarizing function for a particular multi-objective problem. The simple scalarizing function can be divided into weighted sum or weighted Tchebycheff. We implemented the basic version of MOEA/D as follows:

1. Weighted Tchebycheff approach is employed.
2. Let λ^1,....., λ^N be a set of even spread weight vector.
3. z^* is the reference point.
4. The problem of approximation of the Pareto front can be decomposed into N scalar optimization sub-problems by using the Tchebycheff approach and the objective function of the j^{th} sub-problems is:

$$g^{te} = (x|\lambda^j, z^*) = max\left\{\lambda^j | f_i(x) - z_i^* |\right\} \tag{2}$$

where $\lambda^j = (\lambda_1^j, .., \lambda_m^j)^T$. MOEA/D minimizes all these N objective functions simultaneously in a single run. Note that g^{te} is continuous of λ , the optimal solution of $g^{te}(x|\lambda^i, z^*)$ should be close to that of $g^{te}(x|\lambda^j, z^*)$ if λ^i and λ^j are close to each other. Therefore, any information about these g^{te} 's with weight vectors close to λ^i should be helpful for optimizing $g^{te}(x|\lambda^i, z^*)$. A neighborhood of weight vector λ^i is defined as a set of its several closest weight vector in $\lambda_1, ..., \lambda_N$. The neighborhood of the i^{th} sub-problem consists of all the sub-problems with the weight vectors from the neighborhood of λ^i . The population composes of the best solution found so far for each sub-problem. At each generation t, MOEA/D with the Tchebycheff approach maintains:

- a population of N points $x^1, .., x^N \in \Omega$,where x^i is the current solution to the i^{th} sub-problem.
- $FV^1, , FV^N$, where FV^i is the F-value of x^i .
- $z = (z_1, ., z_m)^T$, where z^i is the best value found so far for objective f_i .
- An external population (EP), which is used to store non-dominated solutions.

Input: MOP(1); A stopping condition N: number of sub-problems A uniform spread of N weight vectors: $\lambda_1, , \lambda_N$ T: the number of the weight vectors in the neighborhood of each weight vector. Output: External Population (EP)

MOEA/D algorithm:

- Step 1 Set EP = $\oslash$.
- Step 2 Compute Euclidean distance between two weight vectors and work out T closest weight vector.
- Step 3. Generate initial population, initialize z.
- Step 4 Reproduction
- Step 5 Improvement
- Step 6 Update z
- Step 7 Update Neighboring solutions
- Step 8 Updated EP
- Step 9 Repeat Steps 4 - 8 until stopping criteria is met.

2.4 Requirements of Multi-objective Evolutionary Algorithms

Solving a multi-objective problem involves satisfying three areas as follows: (1) Convergence- The approximation set achieved for a multi-objective optimization problem is required to be as close as possible to the Pareto front (all possible optimal solutions). (2) Diversity- The set of Pareto optimal solutions is also required to be well spread and uniformly covering wide areas of the Pareto front. (3) Converging to Region Of Interest (ROI)-Recently, many researchers indicated that approximating the entire Pareto-optimal solutions does not help decision maker (DM) so much. The third goal of converging to the region that appeals to decision maker is considered. The methods of converging to ROI should support the DM in finding the most preferred solution as the final one.

2.5 Preference Articulation Techniques

In Multi-objective decision making literature, the idea of solving a multi-objective optimization problem is understood as helping a decision maker (DM) in considering the multiple objective simultaneously and in finding a Pareto-optimal solutions that please him/her the most. The DM is assumed to take part in the solution process. Methods that the DM is assumed to take part in the solution process are classified into 3 following categories [8]:

(1) Priori methods - In a priori method, the DM articulates preference information and one's aspirations and then the solution process tries to find a Pareto-optimal solution satisfying them as much as possible. This is a straightforward approach but the difficulty is that the DM does not necessarily know

the possibilities and limitations of the problem beforehand and may have too optimistic or pessimistic expectations. (2) Posteriori methods - where a representation of the set of Pareto-optimal solution is first generated and then the DM is supposed to select the most preferred one among them. This approach gives the DM an overview of different solutions available. However, if there are more than two objectives in the problem, it may be difficult for the DM to analyze the large amount of information. This is because visualizing the solutions is no longer as straightforward as in a bi-objective case. On the other hand, generating Pareto-optimal solutions may be computationally expensive. Typically, multi-objective optimization algorithms (MOEAs) belong to this class. (3) Interactive methods - The previous mentioned methods consider either no DM takes part in the solution process or she/he expresses preference relations before or after the running process of the algorithm. This last method devotes to interactive methods which are the most extensive methods [8]. In interactive approaches, an iterative solution algorithm which can be called a solution pattern is formed and repeated several times. After each iteration, some information is given to the DM and he/she is asked to specify preference information in the form that the method in question can utilize, e.g., by answering some questions. One can say that the analyst aims at determining the preference structure of the DM in an interactive way. The noteworthy part is that the DM can specify and adjust one's preferences between each iteration and at the same time learn about the interdependencies in the problem as well as about one's own preferences.

3 Incorporation of Preference Articulation to MOEAs

The incorporation of preference articulation has been developed in several reasons. This section reviews some well-known techniques that incorporate to the multi-objective optimization problem solving. Many researches in evolutionary multi-objective optimization attempt to approximate the complete Pareto optimal frontier by a set of well-distributed representatives of Pareto-optimal solutions. On the other hand, in most practical applications, the DM is eventually interested in only a single solution at some point during the optimization process. The role of decision maker (DM) is illustrated in Fig. 4.

Branke et al. (2001) proposed a method for utilizing the decision maker's preferences for guiding the search towards the regions of interest. The method asks the decision maker to specify his/her trade-offs between each pair of objectives. After that, it constructs the minimal and maximal utility functions. These utility functions are used for modifying the dominance scheme according to the decision maker's preferences [1]. The authors also suggested the Guided Multi-Objective Evolutionary Algorithm (G-MOEA) that uses this method. They showed that the algorithm is able to converge to the desired regions. The concept is illustrated in Fig. 5(b). The G-MOEA can be implemented by a simple transformation of the objective: it is sufficient to replace the original objectives with two auxiliary objectives and use these together with the standard dominance principle.

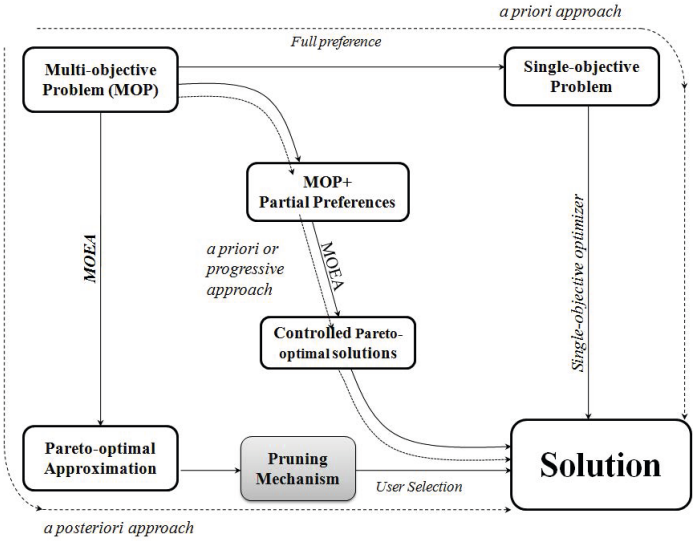


Fig. 4. The role of decision maker (DM) in multi-objective optimization

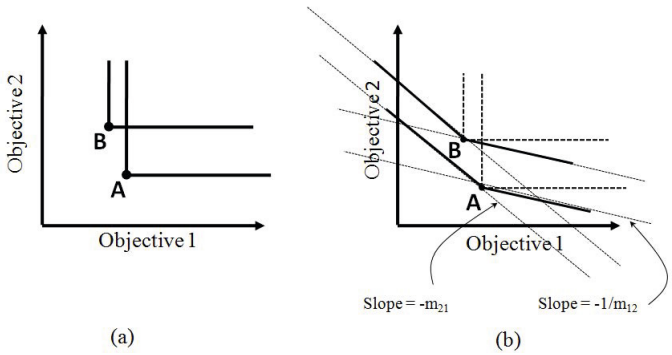


Fig. 5. The concept of guided dominance

The extended dominance can be calculated as follows.

$$x \succ y \Leftrightarrow [(f_1(x) + m_{12} f_2(x)) \leq (f_1(y) + m_{12} f_2(y))] \wedge (\mathrm{m}_{21} f_1(x) + f_2(x)) \leq (m_{21} f_1(y) + f_2(y)) \quad (3)$$

In equation 3, the parameters $m_1 2$ and $m_2 1$ denote correspondingly the maximum acceptance amount of degradation for objectives 1 and 2 which are compensated by a single unit of improvement in terms of objectives 1 and 2, respectively. The guided dominance concept is illustrated in Fig. 5(b) on a simple bi-objective optimization problem. In 2004, Branke et al. also proposed the "Knee regions concept". They suggested the solutions in the knee area that are generally preferable compared to other solutions [19]. The obtained solutions from this approach are the solutions that are located at the convex regions.

Deb (2003) suggested a method that seeks to find a set of solutions based on biased sharing mechanism extended with a better control of the region of interest and separate parameter controlling the strength of the bias [20]. For a solution i on a particular front, the biased crowding distance measure D_i is redefined as follows,

$$D_i = d_i(\frac{d_i'}{d_i})^{\alpha} \quad (4)$$

where d_i and d_i' are the original crowding distance and the crowding distance calculated based on the location of the individuals projected onto the plane P with direction η , respectively. Fig 6 illustrates the concept. α denotes a parameter controlling the bias intensity to control the extent of the bias. Larger α results in a stronger bias.

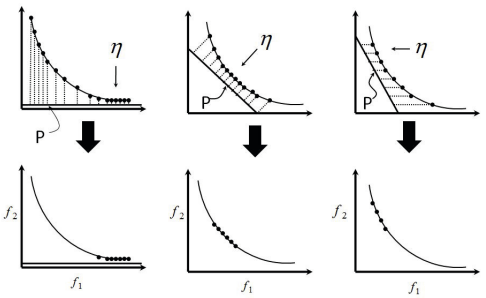


Fig. 6. Pareto bias approach

Koksalan and Karahan (2010) developed a preference-based multi-objective evolutionary algorithm that interacts with the DM during the course of optimization. The author created a territory around each solution where no other solutions are allowed. The smaller territories around preferred solutions were defined in order to obtain denser coverage of these regions. At each interaction, the algorithm asks the DM to choose his/her best solution among a set of

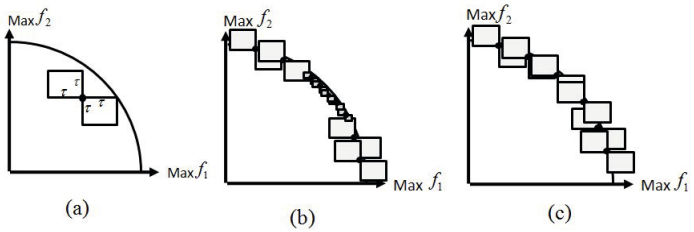


Fig. 7. Territory defining approach

representative solutions to guide the search toward the neighborhood of the selected solution [5]. The algorithm aims to converge to a final preferred region of the DM.

An illustration of the territory of an individual in bi-objective space is given in Fig. 6. A territory size parameter is defined around the individual closest to the offspring. Fig. 6 (b) illustrates different territory size. Small territory size indicates the preferred region. Fig. 6 (c) illustrates the equal territory size to preserve diversity.

Leesutthipornchai and Wattanapongsakorn (2010) proposed a new reason of pruning Pareto-optimal solutions. The efficient solutions are approximated by generic MOEAs (user-preferred), then the pruning mechanism called adaptive angle based pruning algorithm (ADA) is applied [11]. Undesired solution will be filtered-out by removing the solutions that only marginally improve in some objectives. The algorithm uses the concept of extended dominance by modifying the regular dominance as indicated in Branke et al [17]. The contributions of ADA are based on following assumptions: the DM does not prioritize multiple objective functions, and the algorithm emphasizes the final solutions that balance all objective values.

4 Proposed Pruning Algorithm

The idea behind adaptive geometric angle-based pruning algorithm (ADA) is justified by the guided dominance technique proposed in [1]. In ADA, a geometric angle is calculated between pair of solutions of each objective. A threshold angle of each objective is introduced in order to approximate the portions of desirable solutions based on the DM's opinions. The pruning rationale is to increase the dominated area for the purpose of removing solutions that only marginally improves in some objectives while being significantly worse in other objectives. The extra angles are expanded from the regular dominated area. The expanded area means some solutions that have marginal improvement are discarded. Only significantly improved solutions are selected to be the non-dominated set.

Our pruning method begins by calculating the angle between a pair of solutions by using a simple geometric function that is an inverse tangent function. The angle between two non-dominated solutions is calculated by using Equation

(5). The geometric angle is denoted by θ_n where n is the n^{th} objective. For the minimizing objective context, θ_n is given by

$$\theta_n = tan^{-1}\left[\frac{\sqrt{\sum_{m=1,m\neq n}^{N}(\Delta f_m)^2}}{\Delta f_n}\right] \tag{5}$$

where, N denotes the number of objective functions. n denotes the n^{th} objective functions.

Δf_n denotes the difference of the nth objective value between two non-dominated solutions.

For example, considering multi-objective optimization with two-objective functions f_1 and f_2 , the angle of extended dominated area for solution A are shown in Fig. 7.

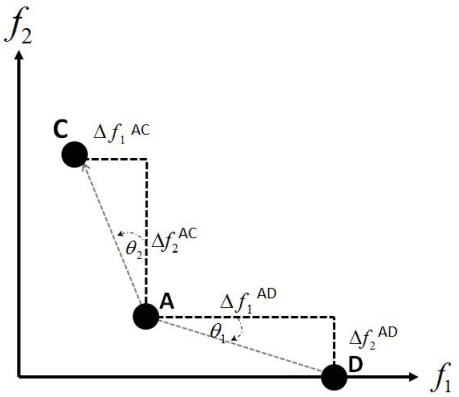


Fig. 8. The angle calculation of two-objective functions

$$\theta_1 = tan^{-1}\left[\frac{\sqrt{(\Delta f_2^{AD})^2}}{\Delta f_1^{AD}}\right]$$

$$\theta_2 = tan^{-1}\left[\frac{\sqrt{(\Delta f_1^{AC})^2}}{\Delta f_2^{AC}}\right]$$

we define the dominance condition as follows :

$$i \succ j \Leftrightarrow \sum_{n=1}^{N}([f_n(i) \leq f_n(j)] \wedge [\theta_n(i,j)| \leq |\delta_n|])) > 0 \tag{6}$$

From equation 6, solution i dominates solution j is represented by $i \succ j$. For example, a two-objective optimization has a multi-objective criteria such that

$$i \succ j \Leftrightarrow ([f_1(i) \leq f_1(j)] \wedge [\theta_1(i,j)| \leq |\delta_1|]) + ([f_2(i) \leq f_2(j)] \wedge [\theta_2(i,j)| \leq |\delta_1|]) > 0$$

Solution i is better than solution j if and only if there exists at least one function that is true. The dominated solution is usually located in only one side of the extended areas. (extension from f_1 geometric angle or f_2 geometric angle)

For the threshold angle (δ_n) , we implemented as following steps below:

1. Each non-dominated solution is sorted in ascending order for each objective.
2. Inter-quartile range of sorted data of each objective is calculated, denoted by IQS_n .
3. The average distance of of n_{th} objective value between two consecutive non-dominated solutions is calculated, denoted by IQ_n

The threshold angle of each objective value can be calculated as follows:

$$\delta_n = \left[tan^{-1} \left(\frac{IQS_n}{IQ_n} \right) \right]^{\tau_n} \tag{7}$$

where n is objective number; τ_n is bias intensity of each objective, ranging from 0-1, stronger bias results in less prefer objective.

The overall pruning mechanism is implemented with the following procedures.

- Step 1: Approximate the Pareto-optimal solutions using multi-objective evolutionary algorithm.
- Step 2: Specify the bias parameter for each objective independently for each threshold. (n thresholds for n objectives)
- Step 3: Normalize all solutions with the same range in each objective
- Step 4: Convert all objectives into minimization context (maximization can be converted to minimization by multiplying by -1)
- Step 5: For each non-dominated solution
  - Step 5.1: Calculate the threshold angle by using equation 7.
  - Step 5.2: Compare with the other solutions using Angle based comparison as shown in equation 6. Select only the solution that are not dominated by other solutions.

5 Experimental Result

We begin our experiment by selecting several benchmark problems considering different complexities such as concave, convex, discrete and continuous shapes. We randomly test the selected benchmark problems on two and three objectives regarding the shape complexity. The experiment was set as follows: a) Approximate the Pareto-optimal solutions of selected benchmark problems by MOEA/D MOEA/D Parameters: Population Size = 500 Maximum Iteration = 150000 Crossover Probability = 1.0 Mutation Probability =0.1 Scaling factor for mutation = 0.5 Distribution Index = 20 b) Apply our pruning algorithm on each Pareto-optimal set. c) Tuning bias intensity τ^i for each objective function $i, 1 \leq i \leq N$. d) Examine the result.

Result discussion: To observe the result effectively, five different forms of benchmark problems are considered in our experiment, where each problem has its unique characteristic and complexity. We observe the result as follows:

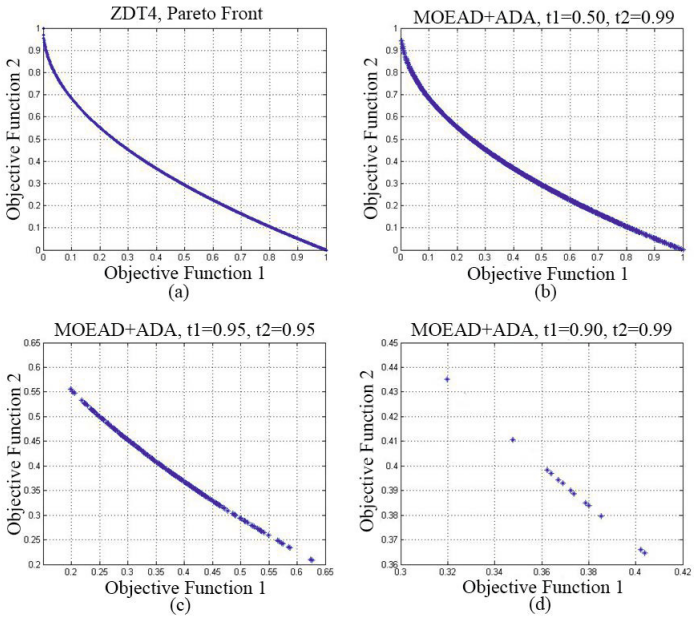


Fig. 9. ZDT4 benchmark problem

ZDT4: There is less significant of pruning result when biasing objective 2 with stronger value while lowering the bias in objective 1 (lower bias results in more preferred objective). There is a significant result when stronger bias is applied to both objectives as shown in Fig. 9 (d). The pruned Pareto-optimal solutions distribute favoring objective 1 when stronger bias is applied to both objectives.

WFG1: WFG1 problem has more complicated shape than ZDT4 (multi-modal form). There is a significant result when stronger bias is applied to objective 1 as shown in Fig. 10. (c) The pruned Pareto-optimal solutions usually located at extreme regions favoring objective 2 (Fig. 10 (d)), where f_2 value is no greater than 0.9.

WFG2: WFG2 problem has discrete shape of Pareto front. The pruning result is quite interesting, applying appropriate bias values ($\tau_1 = \tau_2 = 0.95$) to each objective constitute the pruned Pareto-optimal solutions distribute and locate at knee regions of Pareto front as shown in Fig. 11. (c). When applying stronger bias to objective 1 (objective 2 has more priority), the pruned Pareto-optimal solutions located at extreme regions favoring objective 2 as shown in Fig. 11. (d), where f_2 value is around 0.8.

DTLZ5: In DTLZ5 problem, we scale the objective number to 3 objectives and observe the result in 3-dimensional plan. It is quite difficult to observe the result of different intensity value of each bias intensity parameter for each objective in 3D plan. We observed that there are large numbers of pruned Pareto-optimal solutions when tuning the bias intensity parameter in following order: $1 \geq 2 \geq 3$

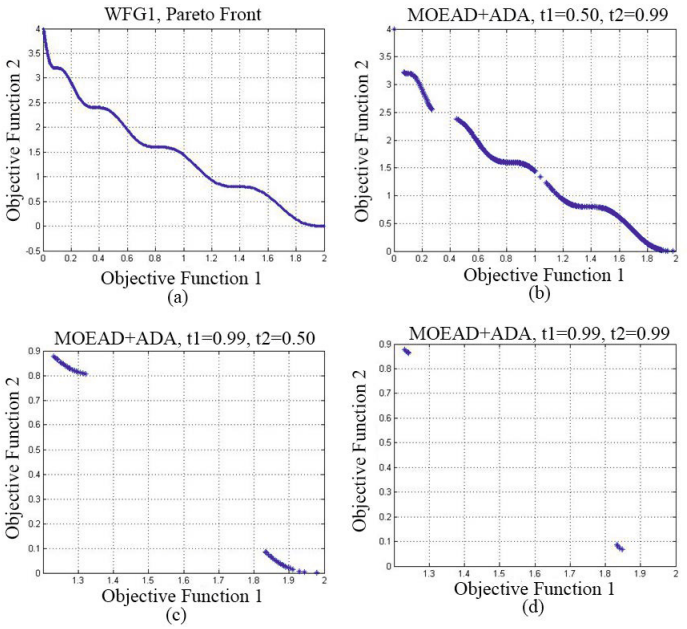


Fig. 10. WFG1 benchmark problem

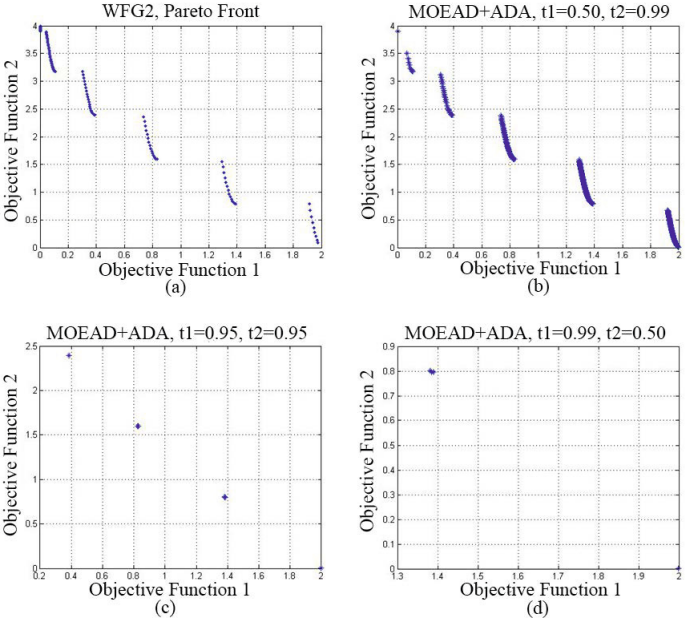


Fig. 11. WFG2 benchmark problem

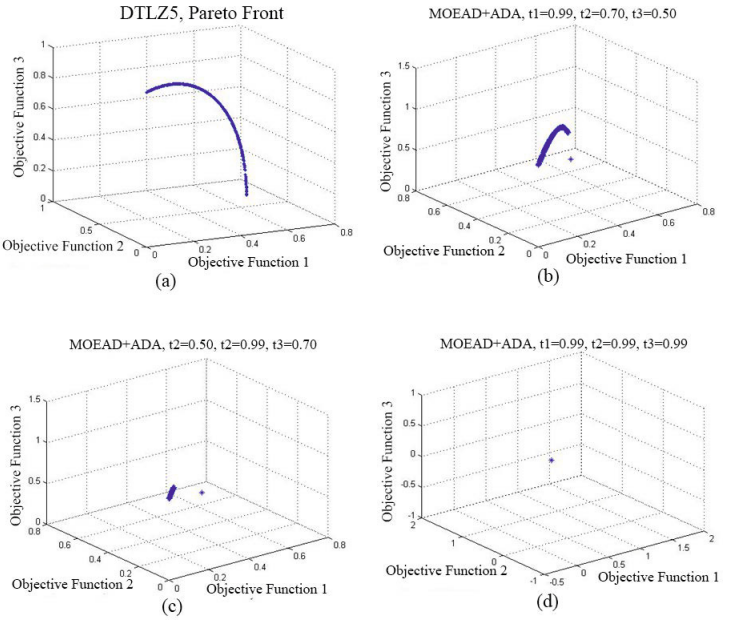


Fig. 12. DTLZ5 benchmark problem

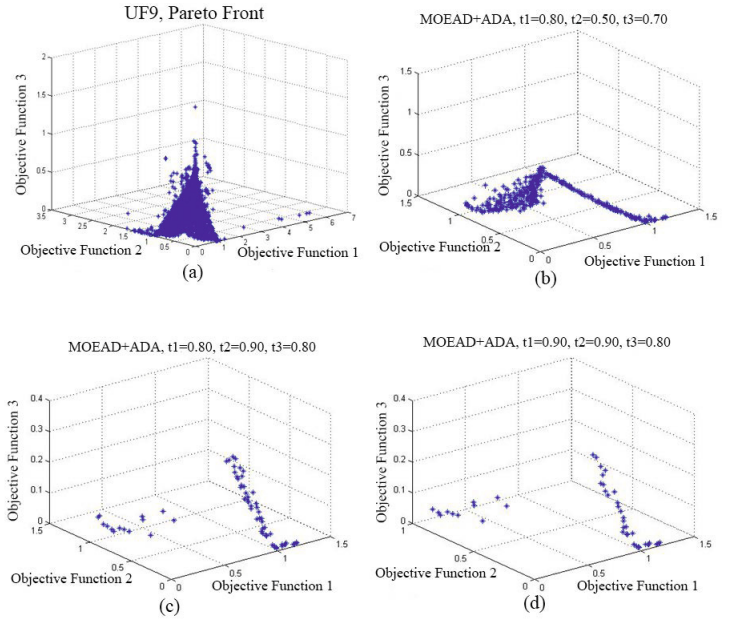


Fig. 13. UF9 benchmark problem

(objective 3 is most preferred) as shown in Fig. 12. (b). In addition, only one remaining solution is obtained when the strongest bias of each parameter is applied equally ($\tau_1 = \tau_2 = \tau_3 = 0.99$) as shown in Fig. 12. (d).

UF9: UF9 is considered a complicated Pareto set. We observed that the bias procedure is different from other problem. After varying the bias intensity parameters, we can get the pruned Pareto-optimal solutions located on the edge area as shown in Fig.13(c)-(d).

Performance Evaluation

For assessing the performance of the algorithms to the test problems, two different issues are normally taken into account: (1) to minimize the distance of the Pareto front generated by the algorithm to the exact Pareto front (convergence), and (2) to maximize the spread of solutions (diversity). Various quality indicators have been proposed to be used in comparative studies for solving multi-objective optimization problems. They can be classified into those measuring convergence (e.g. GD), diversity (e.g. Spread) and both (e.g. Hypervolume). Since our algorithm focuses on specific regions of Pareto front, all quality indicators that measuring diversity are not useful to evaluate the algorithm.

$$GD = \frac{\sqrt{\sum_{i=1}^{n} d_i^2}}{n} \tag{8}$$

It's also an opportunity to measure the performance of our algorithm in terms of accuracy by using convergence metric such as GD or IGD. GD indicator was introduced for measuring how far the elements in the set of non-dominated vectors found are from those in the Pareto-optimal set [6]. The GD is formulated in equation 8, where n is the number of vectors in the set of non-dominated solutions, and d_i is the Euclidean distance (measured in objective space) between each solution found and nearest member of the Pareto-optimal set. It is clear

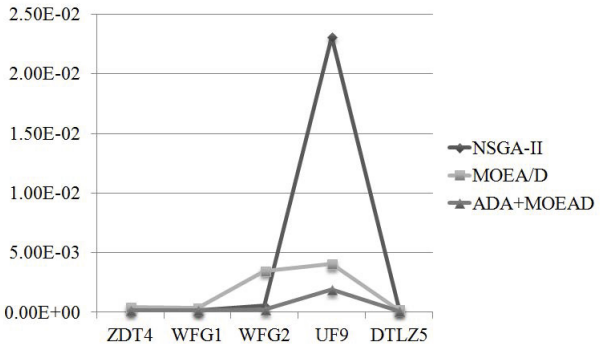


Fig. 14. Generational distance performance metric

that a value of GD = 0 means that all the generated elements are in the Pareto-optimal set. The results in Fig. 14 have proved the accuracy of our algorithm. We can get better GD value if is decreased. It's clearly shown that even with stronger bias, our algorithm still outperform others. Therefore, our pruning algorithm can filter out less accurate solutions and keep more accurate and robust solutions effectively.

Conclusion

We proposed a geometric angle-based pruning algorithm (ADA) with independent bias intensity tuning parameter in computing extended dominance by calculating the angle between pair of solutions using simple a geometric function. The features of our pruning algorithm can be listed as follows: (1) Filter out marginally improvement solutions in some objectives. (2) Introduce independent bias intensity tuning parameter. (3) The pruning result still reflects diversity of the solutions even when the strongest bias is applied. (4) The pruned Pareto-optimal set is distributed to multiple regions instead of single region. (5) It is clearly shown in benchmark problems that the pruned Pareto-optimal solutions are located in the knee regions of the Pareto-front. We use MOEA/D algorithm to approximate Pareto-optimal solutions for each problem, then the pruning algorithm is applied and the result is observed. The experimental results have shown that our pruning algorithm provides robust sub-set of Pareto-optimal solutions for several benchmark problems. In our future work, we plan to integrate our pruning algorithm with decomposition-based MOEA and observe the result interactively. In addition, we will allow the DM to choose his/her prefer region during the running process of pruning algorithm.

References

1. Branke, J., Kauber, T., Schech, H.: Guidance in Evolutionary Multi-objective Optimization. J. on Adv. Eng. Software 32, 449–507 (2001)
2. Kim, J.H., Han, J.H., Choi, S.H., Kim, E.S.: Preference-Based Solution Selection Algorithm for Evolutionary Multiobjective Optimization. IEEE Tran. on Evolutionary Computation 16, 20–34 (2012)
3. Konak, S.K., Coit, D., Baheranwala, F.: Pruned Pareto-optimal Sets for the System Redundancy Allocation Problem Based on Multiple Prioritized Objectives. J. of Heuristics 14, 335–357 (2008)
4. Soylu, B., Ulusoy, S.K.: A preference ordered classification for a multi-objective max-min redundancy allocation problem. Computer and Operation Research 13, 1855–1866 (2011)
5. Karahan, I., Koksalan, M.: A Territory Defining Multiobjective Evolutionary Algorithms and Preference Incorporation. IEEE Trans. on Evolutionary Computation 14, 636–664 (2010)
6. Zhang, Q., Li, H.: MOEA/D: A multiobjective evolutionary algorithm based on decomposition. IEEE Trans. on Evolutionary Computation 11, 712–731 (2007)
7. Coit, D., Konak, A.: Multiple Weighted Objectives Heuristic for the Redundancy Allocation Problem. IEEE Trans. on Reliability 55, 4471–4479 (2006)

8. Jaszkiewicz, A., Branke, J.: Interactive Multiobjective Evolutionary Algorithms. In: Branke, J., Deb, K., Miettinen, K., Słowiński, R. (eds.) Multiobjective Optimization. LNCS, vol. 5252, pp. 179–193. Springer, Heidelberg (2008)
9. Branke, J.: Consideration of Partial User Preferences in Evolutionary Multioobjective Optimization. LNCS, pp. 157–178. Springer, Berlin (2008)
10. Goldberg, D.E.: Genetic algorithms in search, optimization & machine learning. Addison-Wesley, Reading (1989)
11. Leesutthipornchai, P.: Multi-Objective Optimization for Grooming, Routing and Wavelength Assignment in Optical Network Design. PhD dissertation. Department of computer engineering. King Mongkut's University of Technology Thonburi, Thailand (2010)
12. Wang, R., Purshouse, R.C., Fleming, P.J.: Local Preference-inspired Coevolutionary Algorithms. In: Genetic and Evolutionary Computation Conference (GECCO 2012), Philadelphia, Pensylvania, USA, pp. 513–520 (2012)
13. Bechikh, S., Said, L.B., Ghedira, K.: Negotiating decision makers' reference points for group preference-based Evolutionary Multi-objective Optimization. In: The 11th International Conference on Hybrid Intelligent Systems (HIS), pp. 377–382. IEEE Press, Melacca (2011)
14. Branke, J., Deb, K.: Integrating User Preferences into Evolutionary Multi-objective Optimization. KangGAL Technical report, Report Number 20004004 (2004)
15. Mohammadi, A., Omidvar, M.N., Li, X.: Reference Point Based Multi-objective Optimization Through Decomposition. In: World Congress on Computational Intelligence, WCCI 2011, Brisbane, Australia, pp. 1150–1157 (2012)
16. Friedrich, T., Kroeger, T., Neumann, F.: Weighted Preferences in Evolutionary Multi-objective Optimization. In: 24th Australian Joint Conference on Advances in Artificial Intelligence, Perth, Australia, pp. 291–300 (2011)
17. Zitzler, E., Laumanns, M., Thiele, L.: SPEA2: Improving the Strength Pareto Evolutionary Algorithm. Computer Engineering and Network Laboratory (TIK), Department of Electrical Engineering, Swiss Federal Institute of Technology (ETH) Zurich, Swithzerland, pp. 1–21 (2001)
18. Deb, K., Pratab, A., Agarwal, S., Meyarivan, T.: Fast and Elitist Multi-objective Genetic Algorithm: NSGA-II. IEEE Trans. on Evolutionary Computation 6, 182–197 (2002)
19. Branke, J., Deb, K., Dierof, H., Osswald, M.: Finding knees in multi-objective optimization. In: The Eight Conference of Parallel Problem Solving from Nature (PPSSN VIII) (2004)
20. Deb, K.: Multi-objective Evolutionary Algorithms: Introducing Bias Among Pareto-optimal Solutions. Adv. in Evolutionary Computing, 263–292 (2003)

Intelligent Collision Avoidance for Multi Agent Mobile Robots

Aya Souliman[1], Abdulkader Joukhadar[1],
Hamid Alturbeh[2], and James F. Whidborne[2]

[1] Aleppo University, Aleppo, Syria
aya_eng_88@yahoo.com, joukhadar2703@yahoo.co.uk
[2] Cranfield University, Bedfordshire, UK
{h.alturbeh,j.f.whidborne}@cranfield.ac.uk

Abstract. This chapter presents a newly developed mobile robot based multi-agent system with capabilities of robust motion control and intelligent collision avoidance. The system consists of three mobile robots. One main robot acts as a master and the other two act as slaves. The master intelligently takes decisions as to which action to perform to avoid obstacles and collisions. The master mobile robot has the capability to swerve around a static or moving object when necessary. All possible conditions have been coordinated in a fuzzy knowledge base which is used to make a decision on the required maneuver to avoid a collision with a slave robot that the mobile robot may encounter on its driving lane. The proposed research has been carried out to simulate a real car driving regime on roads where the driver may not react properly. The system is implemented on a robot experimental test bench and some experimental results are presented and discussed.

Keywords: Fuzzy Logic Control (FLC), Multi-agent Mobile Robot, Collision Avoidance.

1 Introduction

Automobile collision avoidance systems have received wide attention by industry and academia over the last ten years, with the aim of reducing road accidents caused by driver error. The key issue in most collision avoidance systems is the decision making required to avoid collision. Most recent techniques utilized by collision avoidance systems depend on deterministic calculations for deciding the required action to safely avoid a collision and subsequent accident [1–3]. This paper instead investigates on an intelligent collision avoidance system that implements the decision-making process by Fuzzy Logic Control (FLC) [4–7]. This helps the system to consider several critical cases to avoid a collision, namely velocity reduction, lateral path variation (or swerving), safe stopping, and velocity change of the slave mobile robots. Communication between the master and slave robots is via an RF communication module [8] and [9].

The remainder of the chapter is organized as follows. Section 2 provides a brief description of the proposed multi-agent robot system. Section 3 discusses the

L. Chen, S. Kapoor, and R. Bhatia (eds.), *Intelligent Systems for Science and Information,* Studies in Computational Intelligence 542,
DOI: 10.1007/978-3-319-04702-7_17,

proposed control strategy. Section 4 describes the system wireless communication and data exchange. The high level control is explained in Section 5. The low level control system is exhibited in Section 6. Section 7 discusses the practical results conducted from a practical test. Conclusions are provided in Section 8.

Notation

R_{12} denotes the distance between robot R_1 and robot R_2
R_{13} denotes the distance between robot R_1 and robot R_2
V_1 denotes the speed of master robot R_1
V_3 denotes the speed of slave robot R_3
S denotes velocity membership function 'Slow"
H denotes velocity membership function 'High"
S denotes distance membership function 'Small"
M denotes distance membership function 'Medium"
B denotes distance membership function 'Big"
VB denotes distance membership function 'Very Big"

2 Proposed Multi Agent Robot System

Multi-agent systems have received wide research attention by academic and industry. This is due to their inherited system complexity in behavior and to the challenges for providing an intelligent control strategy, which meets all disciplines of multi-agent robot motion control while moving in a specific domain environment. A different application of multi-agent mobile robot systems, e.g., unmanned air craft and spaceflight systems, unmanned underwater vehicles, walking robots and automotive and transportation robot systems etc., require such a control strategy to enable the system achieve the desired goals which has been designed for [10–14].

For any multi-agent robot system applications there must be motion coordination, which aims to intelligently coordinate the motion of the agents in a specific working domain as well as arranging what is most needed to avoid collision between one or more agents while moving.

Fig. 1 shows a schematic of the proposed mobile robot system for robot-to-robot collision avoidance. The multi-agent mobile robot system consists of three robots. The master robot, R_1, and slave, R_2, move on a circuit path in one direction, and the second slave robot, R_3, travels in the opposite direction on a parallel path.

A virtual trajectory is used to enable an S-shape overtaking maneuver by the master robot as an option to avoid collision. The S-shape maneuver allows the master robot to 'swerve' around the third-party mobile. The decision to perform an S-shape motion is based on an intelligent fuzzy inference engine, as

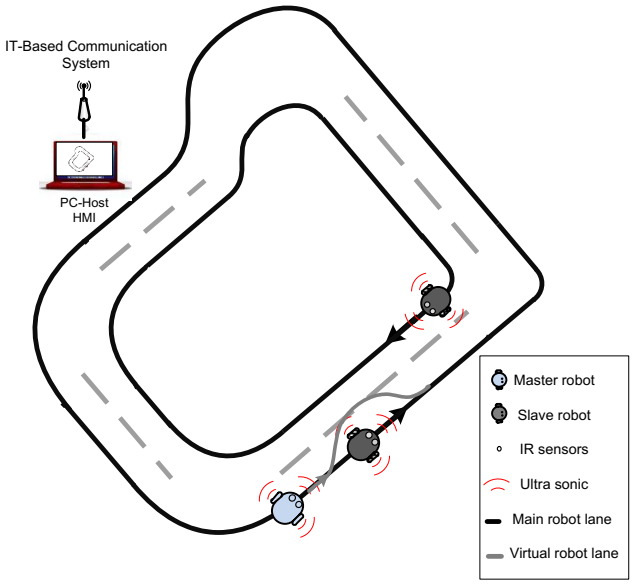


Fig. 1. Multi-agent Robot System

is the decision for completing the S-shape performance is also based on a fuzzy inference engine [4, 5].

The proposed intelligent collision avoidance approach including the fuzzy logic controller has been implemented on I3-3GHz PC using C language. Communication control between the mobile robots and the host PC-based processor is based on an RF module.

3 Proposed Control Strategy

3.1 Sensors

Collision avoidance is considered a very challenging research area since it depends on many criteria and measurements from the agents in the working domain. Precision in collision avoidance is required thus high-cost measurement devices should be used. These generally include sonar, ultrasound, infrared and radar devices. These depend on sending waves from the transmitter, the waves bounce off the nearest objects and travel back to the receiver component of the sensor. In some cases fusion sensors are used to provide high precision measurement of an agent's velocity, displacement as well as distances between agent and object or an agent and other moving agent [7].

In ultrasonic sensors, the time of flight (ToF) method is used for finding the distance between the transmitter and the object. The transmitter sends out a

burst of pulses and the receiver detects the reflected echo. The time delay between the corresponding edges of the transmitted and received pulses is measured by a microcontroller unit (MCU); this gives the time of flight as an echo pulse on the output of the ultrasonic sensor. From this pulse, much information can be determined like distance between the transmitter and the target, position, and relative velocity instantly, and if any sudden changes in those factors could potentially cause a collision.

Substituting the time delay and the velocity of ultrasound in air ($c = 340$m/sec) in (1), the distance d between the transmitter and the target is simply determined by

$$d = \frac{c \times ToF}{2} \tag{1}$$

where ToF is the time of flight from the transmitter to the object and back to the receiver. Fig. 2 shows the echo pulse generation.

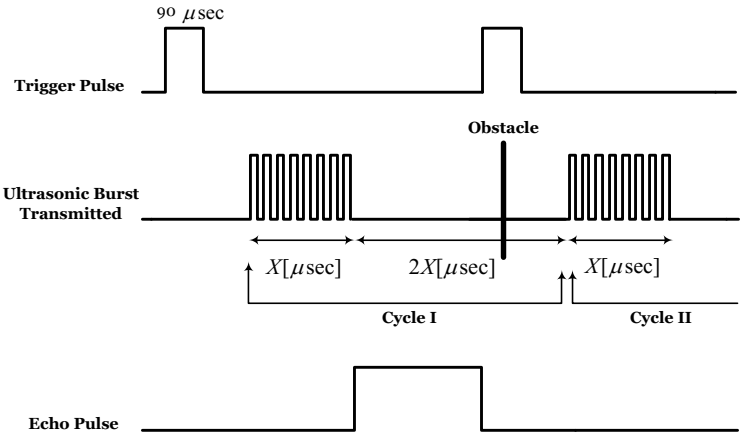


Fig. 2. Echo pulse generation

3.2 Collision Avoidance

Generally, there are two types of collision avoidance systems; passive and active. In passive systems, when the system reaches a critical situation, it just alerts the driver. In active systems, the active collision avoidance system takes an action to prevent the collision, for example: stop the vehicle, decrease the velocity, change the direction, and so on.

The fuzzy collision avoidance system has been designed to be dependent on the velocities of the robots and the distances between them. Fig. 3 shows a block diagram of the collision avoidance decision-making using FLC. The strategy consists of three stages [6, 7]:

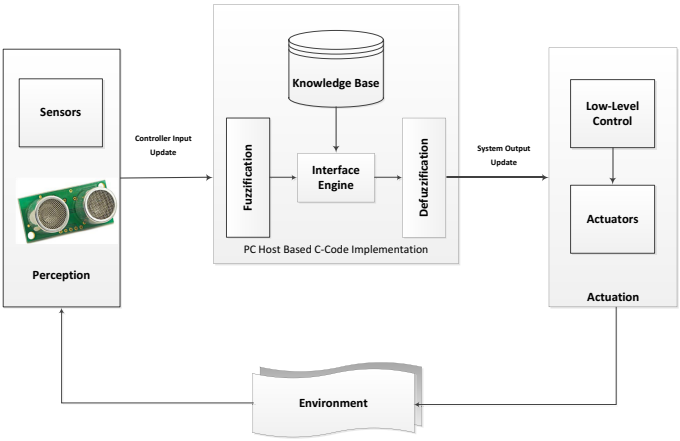


Fig. 3. Block diagram of collision avoidance using FLC

1. **Perception**: in this stage, the velocities of mobile robots are determined using speed sensors and the distances between robots are calculated using ultrasonic sensors distributed around the master robot body.
2. **FLC**: a Mamdani-style fuzzy inference process is performed in three steps [15]:
   (a) **Fuzzification**: in this step the output of the sensors are transformed from crisp data to a fuzzy set in a fuzzy universe of discourse. Each crisp value will have two values in the fuzzy universe of discourse.
   (b) **Decision Maker**: a decision is made using fuzzy logic based on a knowledge base.
   (c) **Defuzzification**: this step transforms the decision in the fuzzy universe of discourse into a crisp control action demand.
   Details of the operation of the FLC are provided by means of an example that follows in Section 3.3.
3. **Actuation**: at the final stage, the control demand is sent to a low level control circuit that controls the actuators to execute the decision (see Section 5).

Figs 4 and 5 show the membership functions of the velocity of the master robot, V_1, and velocity of slave robot, V_3, respectively. Fig. 6 shows the membership functions of the distance R_{12}, while Fig. 7 shows the membership functions of the distance R_{13}.

Tables 1-4 show the possible rules for the decision making handled by the robot where 'X' denotes to remain at previous state (i.e. no change), 'Increase' denotes that the master robot increases speed, 'Decrease' denotes that the master robot decreases speed, 'S-shape' denotes that the master robot commences the S-shape maneuver to overtake R_2.

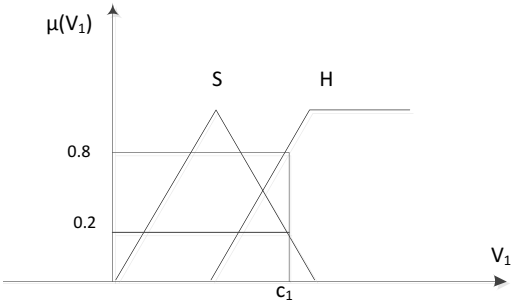


Fig. 4. The distribution of the velocity of R_1 membership functions

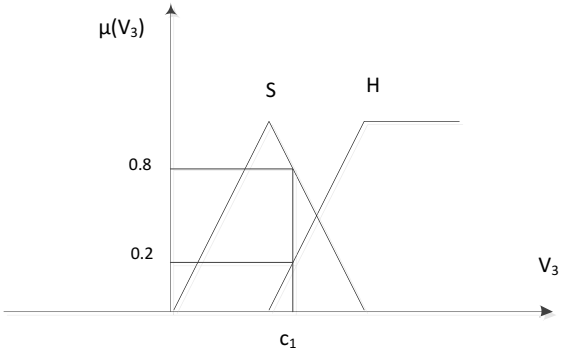


Fig. 5. The distribution of the velocity of R_3 membership functions

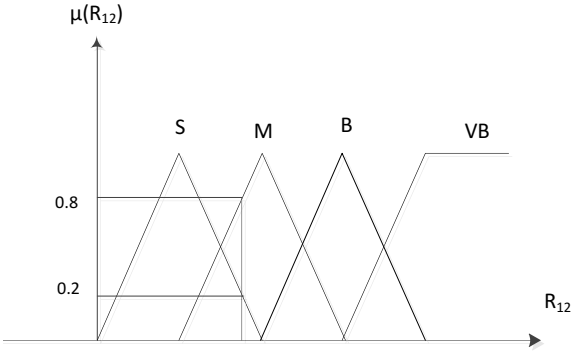


Fig. 6. The distribution of the R_{12} membership functions

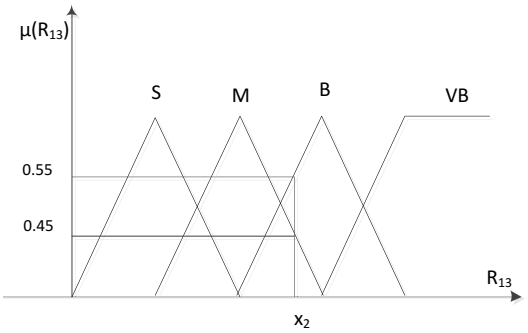


Fig. 7. The distribution of the R_{13} membership functions

Table 1. Fuzzy Inference Engine — S-shape start when V_1 = Slow and V_3 = Slow

R_{13} \ R_{12}	Big	Small
Big	Increase OR X	S-Shape
Small	Increase OR X	Decrease

Table 2. Fuzzy Inference Engine — S-shape start when V_1 = Slow and V_3 = High

R_{13} \ R_{12}	Big	Small
Very Big	Increase OR X	Decrease OR S-Shape
Medium	X	Decrease

Table 3. Fuzzy Inference Engine — S-shape start when V_1 = High and V_3 = Slow

R_{13} \ R_{12}	Very Big	Medium
Big	X	S-Shape
Small	X	Decrease

Table 4. Fuzzy Inference Engine — S-shape start when V_1 = High and V_3 = High

R_{13} \ R_{12}	Very Big	Medium
Very Big	X	S-Shape
Medium	X	Decrease

3.3 FLC

The operation of the FLC is illustrated by the example below.

Fuzzification. Each crisp value will have two values in the fuzzy universe of discourse. For example, from Figs 3, 4, 5 and 6:

$$V_1 = c_1 \Rightarrow \mu(c_1) = 0.2(\text{Slow})\&0.8(\text{High}) \quad (2)$$

$$V_3 = c_3 \Rightarrow \mu(c_3) = 0.8(\text{Slow})\&0.2(\text{High}) \quad (3)$$

$$R_{12} = X_1 \Rightarrow \mu(X_1) = 0.8(\text{Medium})\&0.2(\text{Small}) \quad (4)$$

$$R_{13} = X_2 \Rightarrow \mu(X_2) = 0.55(\text{Big})\&0.45(\text{Medium}) \quad (5)$$

Decision Maker. This stage consists of two steps

1. **Rule evaluation**: the union of two crisp sets consists of every element that falls into either set. The fuzzy union is chosen to be the maximum. Applying the maximum operation on (2)-(5)

$$\mu(V_1 = c_1) = 0.8(\text{High}) \max 0.2(\text{Slow}) \Rightarrow V_1 = \text{High} \quad (6)$$

$$\mu(V_3 = c_3) = 0.2(\text{High}) \max 0.8(\text{Slow}) \Rightarrow V_1 = \text{Slow} \quad (7)$$

$$\mu(R_{12} = X_1) = 0.8(\text{Medium}) \max 0.2(\text{Small}) \Rightarrow R_{12} = \text{Medium} \quad (8)$$

$$\mu(R_{12} = X_1) = 0.55(\text{Big}) \max 0.45(\text{Medium}) \Rightarrow R_{13} = \text{Big} \quad (9)$$

2. **Aggregation of the rule outputs**: depending on the velocities the appropriate inference engine table (Tables 1-4) is chosen and then depending on the distance between the robots the control decision is taken from the table. For example, (6) and (7) show that Table 3 should be used. From Table 3 with (8) and (9), the decision is that an S-shape should be executed.

3.4 Overtaking Maneuver

The overtaking maneuver is divided into two parts as shown in Fig 8. When the master robot has completed the first S-shape and moved into the parallel path, a second decision is required as to when to complete the maneuver by an 'End S-shape' decision that will take the master robot back onto its original path. So the complete trajectory is not predetermined, but is dependent upon the master and slave robot velocities. Hence a fuzzy inference engine controller is also proposed to enable the master robot to securely complete the maneuver. Table 5 shows the fuzzy-rules for completing the trajectory by the End S-shape decision. Figure 8 shows the stages of the collision avoidance via the S-shape trajectory maneuver. Stop R_3 denotes that the slave robot R_3 should halt to prevent a collision.

4 Communication System

The communication system is implemented using radio frequency (RF) [8] and [9].The master robot and PC-Host computer are provided with an RF

Table 5. Fuzzy Inference Engine — Completing S-Shape

R_{13} \ R_{12}	Big	Medium	Small
Big	End S-shape	X	X
Medium	End S-shape	End S-shape	X
Small	End S-shape	End S-shape	Stop R_3

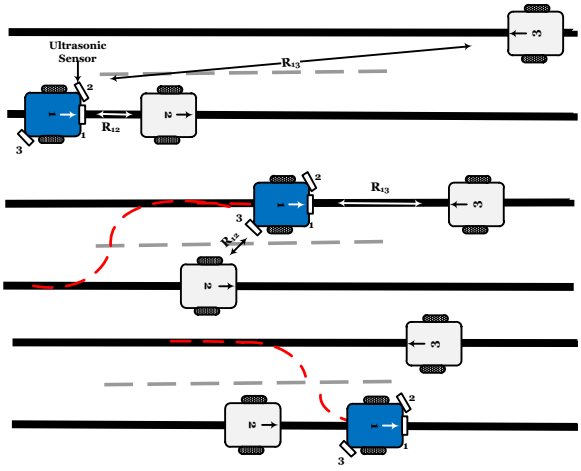


Fig. 8. Collision avoidance stages

transmitter and receiver. The master robot sends ultrasound sensor signals to the computer and receives from it a reply as the command. The PC-Host receives ultrasound signals from the master robot that are used to calculate the distance between the robots and subsequently to apply the FLC algorithm to make the correct decision. Finally it sends the decision to the master robot for execution. The slave robots are provided with RF receivers to receive velocity commands from the PC-Host. The user can enter the desired velocities of the robots through the GUI shown in Fig 9. Figure 10 shows the communication system of the master robot and Fig 11 the communication system of the slave robot. The command sent by the PC-Host to the master robot consists of 16 bits coded as follows:

1. 4 bits dedicated to trigger the ultrasonic sensors,
2. 4 bits dedicated to control the master robot, achieved using two MUX (one to each wheel) as indicated in Table 6.
3. 8 bits to represent the desired velocities of the right and left wheels (4 bits each) of the master robot.

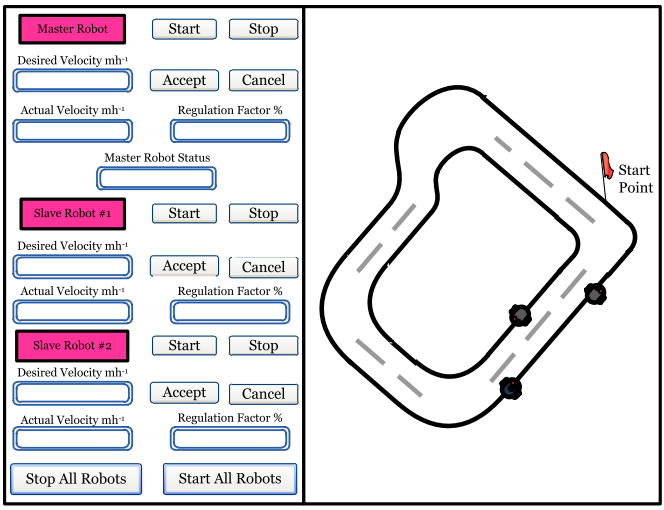


Fig. 9. Interface GUI

The commands sent by the PC-Host to the slave robots consist of 12 bits coded as follows:

1. 4 bits to control the slave robot as indicated in table VI,
2. 8 bits to represent the desired velocities of the right and left wheels of the slave robot.

Table 6. State of the Robots

MUX	Master robot state	Slave robot state
00	Stop	Stop
01	On the main trajectory	On the main trajectory
10	S-shape	Don't care
11	Don't care	Don't care

5 High Level Control System

To control the mobile robot trajectory, the robot right and left actuators velocities, ω_r and ω_l respectively, are controlled [17]. Figure 12 shows the proposed closed loop system. The required trajectory is given as a desired robot state as a function of time:

$$[X_d(t)\ Y_d(t)\ \theta_d(t)\ V_d(t)\ \omega_d(t)]^T \tag{10}$$

where $X_d(t)$, $Y_d(t)$ are the desired position in Cartesian space, $\theta_d(t)$ is the desired robot orientation, $V_d(t)$, $\omega_d(t)$ are the desired linear and angular velocities. $\dot{(X)}_d t$,

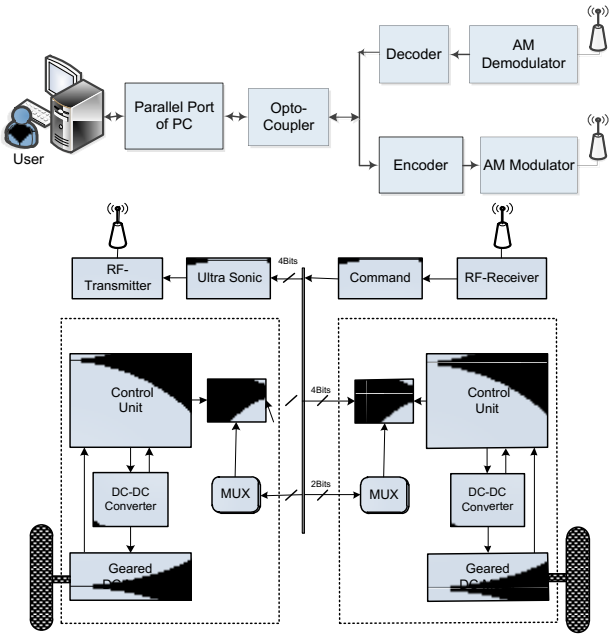


Fig. 10. Communication system - Transmitter/Receiver system of the master robot

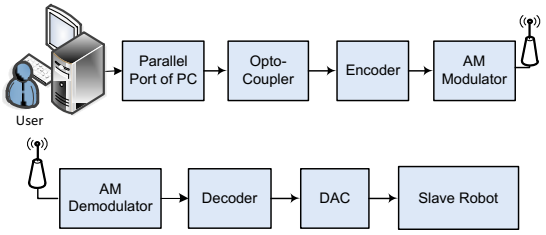


Fig. 11. Communication system - Transmitter/Receiver system of the slave robot

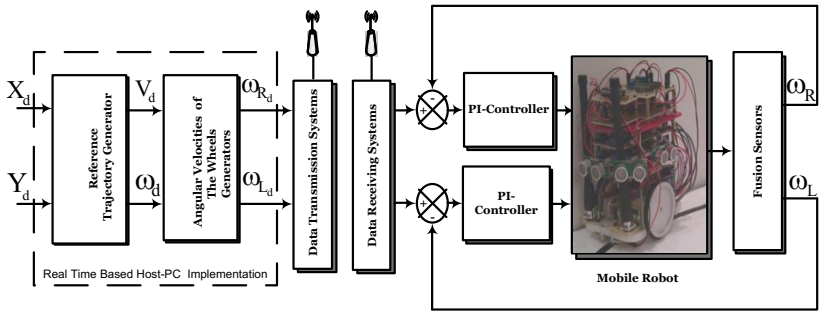


Fig. 12. High Level Control

$\dot{Y}_d(t)$ and $\ddot{X}_d(t)$, $\ddot{Y}_d(t)$ can be calculated by applying the first and the second order derivatives to the desired positions.

State variable trajectories $V_d(t)$, $\omega_d(t)$ and $\theta_d(t)$ can be calculated by [9, 17]:

$$\theta_d(t) = \text{atan2}(\dot{Y}_d(t), \dot{X}_d(t)) \tag{11}$$

$$V_d(t) = \pm\sqrt{\dot{Y}_d^2(t) + \dot{X}_d^2(t)} \tag{12}$$

$$\omega_d(t) = \frac{\ddot{Y}_d(t)\dot{X}_d(t) - \ddot{X}_d(t)\dot{Y}_d(t)}{\dot{Y}_d^2(t) + \dot{X}_d^2(t)} \tag{13}$$

Note that if, at some time t, the reference linear velocity $V_d(t)$ is zero, neither the desired angular velocity $\omega_d(t)$ nor the desired angle $\theta_d(t)$ can be defined by (11) and (13), and hence these must be given explicitly [17].

The desired right and left actuators angular velocities can be obtained by [18]:

$$\omega_{Rd}(t) = \frac{V_d(t) + \frac{d}{2}\omega_d(t)}{r} \tag{14}$$

$$\omega_{Ld}(t) = \frac{V_d(t) - \frac{d}{2}\omega_d(t)}{r} \tag{15}$$

where $2d$ is the distance between the robot wheels and r is the wheel radius.

6 Low Level Control System

The master robot of the proposed robot system has been implemented for the purpose of testing the developed IR sensors, ultrasonic sensors, PI controllers, RF based and the DC/DC controlled converter with the opto-couplers [19]-[21].

Figure 13 shows the block diagram of a single driving wheel control. The Pulse Width Modulation (PWM) technique has been used to control a step down DC/DC transistorized controlled converter which is used for the robot wheels motor drive control. The PWM signal was generated using natural sampling PWM which is based on comparing the control signal (control law - the

output of the PID controller) with a carrier frequency signal (triangular waveform). The opto-coupler is important to provide optical isolation in order to protect the robot controller board from any hazardous voltages that may occur in the DC/DC converter motor drive board. PID-based controllers have been designed to enable precise mobile robot velocity control and to enhance thrust force control for smooth and stable uphill and downhill motion control (if encountered).

The mobile robot is equipped with different types of sensors, namely IR, ultrasonic, and speed sensors. The IR sensor is used to enable following of desired guiding line marking the path. The following is achieved by a bang-bang controller. The ultrasonic sensor is used for detection of both mobile and static obstacle. This information is used by the master robot for the collision avoidance as explained in Section III. The speed sensor provides the feedback signal for the PI speed control. Note that the robot velocity command consists of two signals, the desired speed which is received from the PC-Host and a threshold as shown in the speed command block in Fig 13.

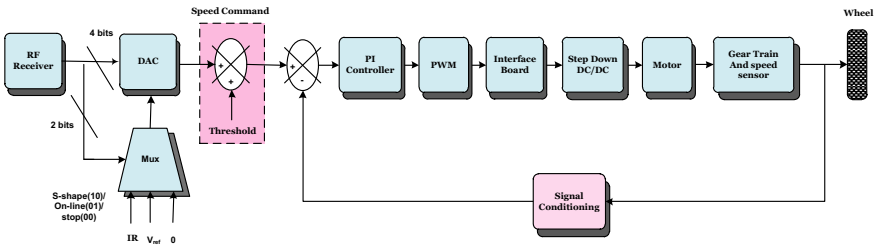


Fig. 13. Motion control system of the robot

7 System Implementation, Practical Results and Discussion

This section provides practical results of each part of the robot, PWM generating, PI-control, IR sensor, ultrasonic sensor and real time system implementation.

Figure 14 shows the generated PWM modulator signal with the saw-tooth waveform, and the control signal from the PI velocity control. The two signals are fed to a comparator. The output of the comparator is the PWM signal which is fed into the opto-coupler to provide optical isolation between the electronic controller board and the motor drive board. PWM signal from the opto-coupler is taken to control the DC/DC controlled converter.

Figure 15 shows the IR sensor output response and the PI controller response. The IR sensor has been tested by a rotating disk consisted of a certain number of black and white sections. The disk was fixed to a rotating shaft. As seen the IR sensor responds rapidly to the white and black color changes imposed by the rotating disk. The response time is within approximately 33msec.

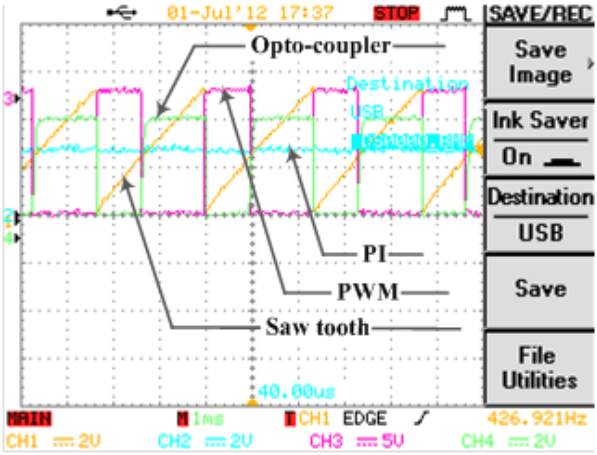


Fig. 14. Generating PWM modulator signal

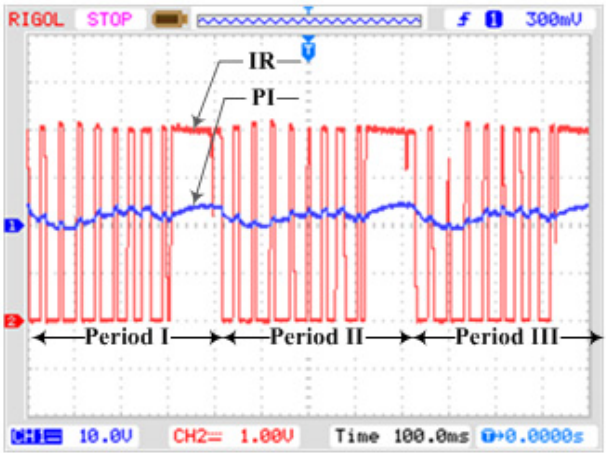


Fig. 15. IR and PI responses

The IR sensor is augmented with a proportional controller which results in a fast response to rapid switching in the black and white colors. This ensures that performance of the mobile robot line-following control will not be compromised by sensors. Furthermore, it should be noted that the IR sensor has a high repeatability factor since there was no error in repeating signals of the IR responses for three sequences of disk cycles (see Fig 15). Also note from Fig 15 that the PI-controller responds rapidly to any change in the IR states which is working as a bang-bang controller.

Figure 16 shows real time velocity PI controller response to speed command signals. As seen, the response is stable and fast with only a small amount of overshoot and oscillation. Also note that the steady state error is negligible.

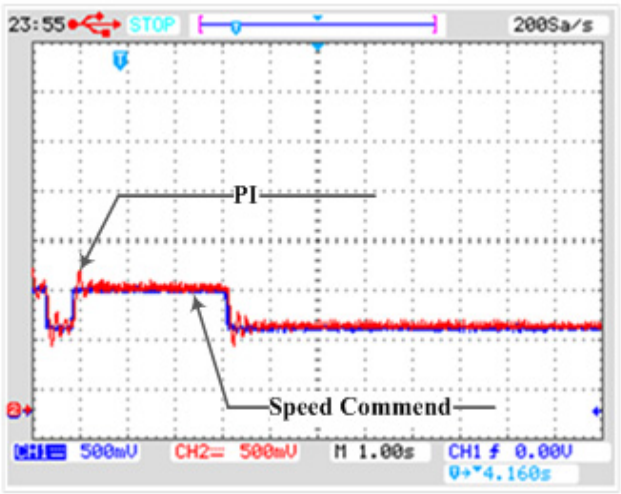


Fig. 16. Real time PI velocity controller response

The ultrasonic sensor has been tested based on a real time controller board to examine the accuracy of the distance measurement algorithm. Figures 17 and 18 show the trigger pulse sent to the ultra-sonic sensor and the received echo. These are respectively obtained from the mobile robot system and from the real time controller board based PC-Host (using RF receivers).

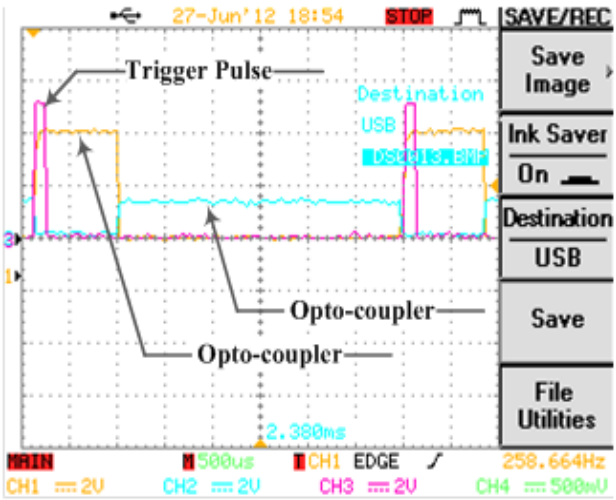


Fig. 17. Echo and trigger pulses for a distance of 50 cm (mobile robot system)

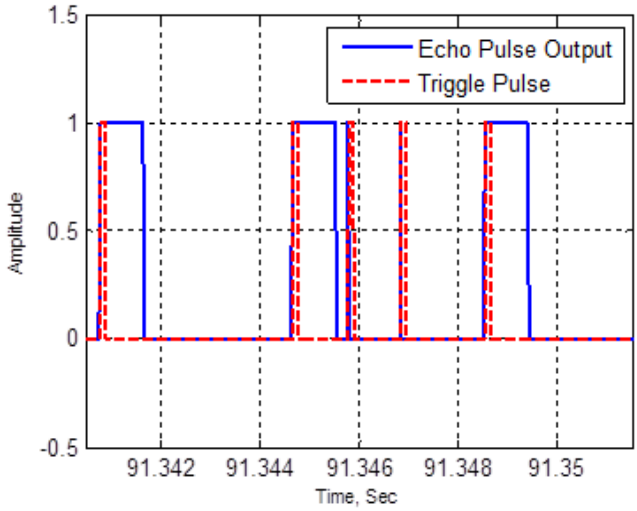


Fig. 18. Detail of echo and trigger pulses (PC-Host system)

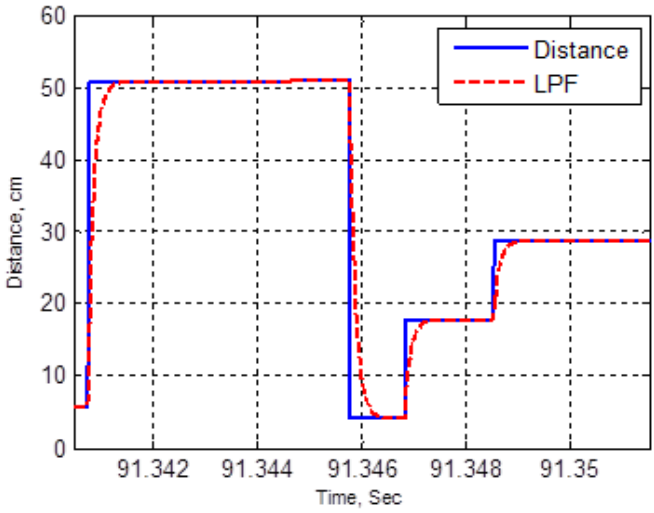


Fig. 19. Calculated distance

Figure 19 shows the calculated distances from the real time controller board on the PC-Host. As can be seen, the distance measurement is accurate to allow good measurement of the distance between the master robot and moving or static obstacles.LPF has been added to get smooth variation in distance measurement.

The proposed FLC algorithm, the High level control algorithm, has been written using C language and implemented based on a host PC. Figure 20 shows

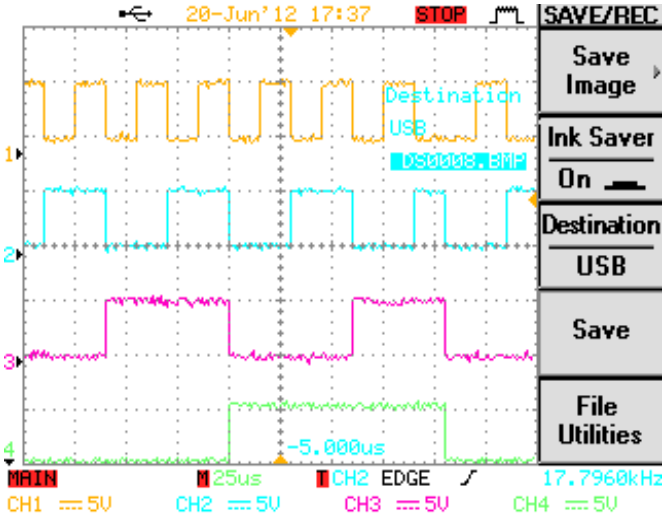


Fig. 20. Real time commands on LPT port

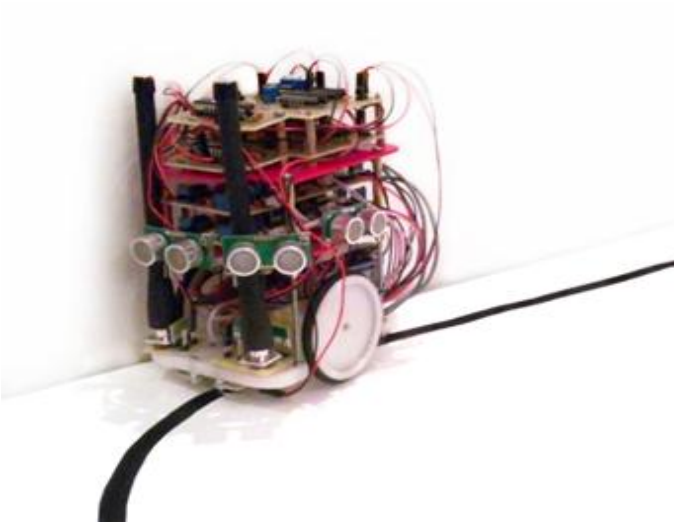


Fig. 21. Master Mobile Robot

a message of 4-bit sent on the LPT port to the master robot. The sample time of the whole control algorithm was set to $15\mu s$.

Figure 21 shows the prototype master robot made in the Mechatronics laboratory at University of Aleppo.

8 Conclusions

This chapter has described a newly developed intelligent collision avoidance system with an adaptive S-shape path following. A multi-agent robot system has been proposed to simulate on road traffic. Fuzzy Logic controller has been proposed for decision making to avoid collision between moving robots in a preproposed path. Fuzzy rules which cover all possibilities have been developed, fuzzy linear membership function have been used as a fuzzy distribution of the distances R_{12} and R_{13} and velocities V_1 and V_3.

Dynamic performance of the mobile robot has been enhanced by utilizing a PI type velocity control which has been implemented with a pulse width modulation strategy using analogue components. Infrared (IR) sensors have been developed with a P type controller to further enhance the path following trajectory of the mobile robot. The communication with the mobile robots have been accomplished using radio frequency (RF) techniques to provide smooth and an ease of communication with the robots in the working field. The master mobile robot has been equipped with a group of ultrasound sensors.

Triggering the sensors and receiving the echo back from the sensors have been completely done using a wireless communication system based on RF protocol. Indoor and outdoor tests have been carried out to proof the validation of the wireless communication approach and the validation of the FLC algorithm for collision avoidance. It is worth mentioning that all control processes with distance calculation between robots are accomplished based on the wireless communication system on a host PC.

References

1. Isermann, R.: Mechatronics Developments for Automobiles. In: 12th Mechatronics Forum Biennial International Conference, Zurich, Switzerland, June 28-30 (2010)
2. Seiler, P., Song, B., Hedrick, J.K.: Development of a Collision Avoidance System, SAE Special Publications, ITS Advanced Controls and Vehicle Navigation Systems, vol. 1332, pp. 97–103 (1998)
3. Jansson, J., Johansson, J., Gustafsson, F.: Decision Making for Collision Avoidance Systems, paper 2002-01-0403, Society of Automotive Engineers (2002)
4. Souliman, A., Joukhadar, A.: Mobile Robot-Based Motion Control and Intelligent Collision Avoidance. In: 12th International Workshop on Research and Education in Mechatronics, Kocaeli, Turkey, September 15-16 (2011)
5. Souliman, A., Joukhadar, A.: Mobile Robot-based Multi Agent Systems "Motion Control and Automatic Collision Avoidance". In: 2nd International Engineering Sciences Conference, Aleppo, Syria, April 18-20 (2011)
6. Bai, Y., Zhuang, H., Wang, D. (eds.): Advanced Fuzzy Logic Technologies in Industrial Applications. Springer, London (2006)
7. Llorca, D., Milanpés, V., Alonso, I., Gavilán, M., Daza, I., Pérez, J., Sotelo, M.: Autonomous Pedestrian Collision Avoidance Using a Fuzzy Steering Controller. IEEE Transactions on Intelligent Transportation Systems 12(2), 390–401 (2011)
8. Klancar, G., Skrjanc, I.: Tracking-error model-based predictive control for mobile robots in real time, Science direct. Robotics and Autonomous Systems 55, 460–469 (2007)

9. Oriolo, G., Luca, A., Vandittelli, M.: WMR Control Via Dynamic Feedback Linearization: Design, Implementation, and Experimental Validation. IEEE Transactions on Control Systems Technology 10(6), 835–852 (2002)
10. Maes, P.: Modeling adaptive autonomous agents. In: Langton, C.G. (ed.) Artificial Life: An Overview, pp. 135–162. The MIT Press, Cambridge (1995)
11. Hernandez-Martinez, E.G., Aranda-Bricaire, E.: Convergence and Collision Avoidance in Formation Control: A Survey of the ArtificialPotential Functions Approach Multi-Agent Systems - Modeling, Control, Programming, Simulations and Applications. In: Alkhateeb, F. (ed.) Multi-Agent Systems - Modeling, Control, Programming, Simulations and Applications. InTech (2011), doi:10.5772/14142
12. Wooldridge, M.: An Introduction to Multi agent Systems. John Wiley (2002)
13. Momani, M.: Collision Avoiding System (CAS). Contemporary Engineering Sciences 5(7), 341–354 (2012)
14. Liu, J., Wu, J.: Multi-agent Robotic Systems. International Series on Computational Intelligence. CRC Press (2001)
15. Negnevitsky, M.: Artificial Intelligence A Guide to Intelligent Systems, 2nd edn. Pearson Education (2005)
16. Souliman, A., Joukhadar, A., Marzi, H.: Model-Based Robust Trajectory Tracking of Wheeled Mobile Robot System Control. In: IEEE IES ISIE 2013, Taipei, Taiwan, May 28-30 (2013)
17. Brezak, M., Petrović, I., Perić, N.: Experimental Comparison of Trajectory Tracking Algorithms for Nonholonomic Mobile Robots. In: 35th Annual Conference of IEEE Industrial Electronics, IECON 2009 (2009)
18. Mireles Jr., J.: Kinematic Models of Mobile Robots, EE 5325/4315 - Kinematics of Mo-bile Robots (2004) (Summer)
19. Bräunl, T.: Embedded Robotics, 2nd edn. Springer, Porto, portugal, November 3-5 (2006)
20. Fu, K.S., Gonzalez, R.C., Lee, C.S.G.: Robotics: Control, Sensing, Vision and Intelligence. McGraw-Hill (1987)
21. Niku, S.B.: Introduction to Robotics. Pearson Education (2001)

Fuzzy Logic Based Network Bandwidth Allocation: Decision Making, Simulation and Analysis

Julija Asmuss and Gunars Lauks

Institute of Telecommunications, Riga Technical University,
Azenes str. 12, Riga LV-1048, Latvia
{julija.asmuss,gunars.lauks}@rtu.lv

Abstract. We present a fuzzy logic based methodology of decision making on bandwidth allocation in a substrate network with DaVinci architecture, according to which the physical substrate network is divided into virtual networks. This methodology describes a fuzzy modification of the adaptive bandwidth allocation mechanism introduced in order to optimize decision making under uncertain network conditions by using fuzzification and defuzzification principles and the expert knowledge database of fuzzy rules. The system of fuzzy rules is optimized using inequalities for fuzzy values of linguistic variables. The effectiveness of this methodology is evaluated on the link level for two traffic types within simulation experiments realized by using Coloured Petri Nets Tools.

Keywords: bandwidth allocation problem, fuzzy rules, decision system, simulation, coloured Petri net.

1 Introduction

Nowadays Internet provides different services (VoIP, video conferencing, music, IPTV, web pages, e-mail, etc). Some services require low delay mechanisms, other services high throughput mechanisms, and their requirements may be incompatible with each other. Network virtualization principles [2], [11] can be used for constructing experimental platforms that run multiple virtual networks. Network virtualization is a popular technique, which is discussed among researchers in the networking field. From network point of view network virtualization splits a network into multiple virtual networks. Such virtual networks provide us with the opportunity to classify and distinguish traffic. Each virtual network is logically separated and can be modified for a particular class of traffic. Resources that are offered by a substrate network are shared between all virtual networks. Finding the suitable bandwidth allocation to virtual networks is one of the main problems of network virtualization [3], [5], [11], [17], [18], [19], [20].

DaVinci approach (Dynamically Adaptive Virtual Networks for a Customized Internet) describes a technique of network virtualization, according to which all virtual networks are constructed over the physical substrate network by subdividing each physical node and each physical link into multiple virtual nodes and

L. Chen, S. Kapoor, and R. Bhatia (eds.), *Intelligent Systems for Science and Information,* Studies in Computational Intelligence 542,
DOI: 10.1007/978-3-319-04702-7_18,

virtual links [6]. We examine the problem of bandwidth resource management in a substrate network basing on DaVinci principles. Under such assumptions it is a maximization problem for the aggregate utility of all virtual networks [10], which effective solution depends on the design of dynamically adaptive bandwidth allocation mechanisms. In the case when different types of traffic coexist over the same network substrate, each virtual network could control a subset of resources at each node and link. At a smaller period of time each virtual network maximizes its own utility. The main question here is whether optimization of virtual networks combined with the bandwidth share adaptation scheme performed by the substrate network in reality maximizes the aggregate utility.

Standard bandwidth allocation mechanisms cannot make decisions in uncertain conditions, which are dominantly persistent in the nowadays networks. The dynamic traffic demand in the fast changing environment almost completely eliminates the possibility of fast online reasoning. Considering that fuzzy logic serves as the excellent tool to deal with uncertain and multivariable data, this giving the flexibility and robustness for decision making while using fuzzy rules, in this paper we propose a fuzzy approach for network bandwidth management. For the evaluation of performance of the proposed fuzzy logic based network bandwidth management policy for two types of traffic we present the design and simulation scheme of dynamically adaptive bandwidth allocation using Coloured Petri Nets [4], [7], [8], [9], [10], [15].

This paper is organized in the following way. Section 2 describes DaVinci approach for modelling of substrate networks accordingly to network virtualization principles. The problem of network bandwidth allocation is considered in Section 3. A fuzzy approach to decision making on bandwidth allocation for delay sensitive traffic and throughput sensitive traffic is described in Section 4. Section 5 presents a short introduction on Coloured Petri Nets used in Section 6 for modelling and simulating the proposed bandwidth allocation mechanisms. The paper finishes with Conclusion.

2 Description of DaVinci Architecture

The DaVinci architecture [6] allows us to describe how a single substrate network can support multiple traffic classes, each with a different performance objective. The problem of bandwidth allocation in a substrate network is a maximization problem for the aggregate objective of multiple applications with diverse requirements. According to the DaVinci approach, each traffic class is carried on its own virtual network with customized traffic-management protocols. The substrate network by assigning resources to each virtual link gives each virtual network the illusion that it runs on a dedicated physical infrastructure.

Let the topology of a substrate network SN be described by a graph $G_S = \{V_S, E_S\}$, given by a set V_S of nodes (or vertices) and a set E_S of links (or edges). We suppose that links of E_S are with finite capacities C_l (links are denoted by $l : l \in E_S$). Correspondingly to $G_S = \{V_S, E_S\}$, we consider DaVinci model with

N virtual networks, indexed by k, where $k = 1, 2, \ldots, N$. Let the key notations be the following:

$y^{(k)}$ - bandwidth of virtual network k , $k = 1, 2, \ldots, N$;
$z^{(k)}$ - path rates for virtual network k , $k = 1, 2, \ldots, N$;
$\lambda^{(k)}$ - satisfaction level degree of virtual network k , $k = 1, 2, \ldots, N$;
$O^{(k)}$ - performance objective for virtual network k , $k = 1, 2, \ldots, N$.

Bandwidth values $y^{(k)} = (y_l^{(k)})_{l \in E_S}$ for each substrate link $l \in E_S$ are assigned by the substrate network, taking into account such local information as current satisfaction indicators and performance objectives. The substrate network periodically reassigns bandwidth shares $y^{(k)}$ for each substrate link between its virtual links. Thus, values $y^{(k)} = (y_l^{(k)})_{l \in E_S}$ and $O^{(k)}$ are periodically updated by the substrate network and used to compute virtual link capacity $y^{(k)}$.

At a smaller timescale, each virtual network runs according to a distributed protocol that maximizes its own performance objective independently. Under such combined conditions in a dynamically changing virtual network environment a fundamental problem of resource allocation is the design of dynamically adaptive bandwidth allocation protocols.

3 Description of Bandwidth Allocation Problem

The goal of the substrate network is to optimize the aggregate utility of all virtual networks

$$\sum_{k=1}^{N} w^{(k)} O^{(k)}(z^{(k)}, y^{(k)}),$$

where $w^{(k)}$ is the weight the substrate assigns to represent the importance of virtual network k. If the substrate wants to give virtual network strict priority, then $w^{(k)}$ can be assigned a value several orders of magnitudes larger than the other weights.

First we consider an optimization problem for the performance objective of each virtual network:

$$\begin{aligned} \text{maximize} \quad & O^{(k)}(z^{(k)}, y^{(k)}), \\ \text{subject to} \quad & H^{(k)} z^{(k)} \leq y^{(k)}, \\ & g^{(k)}(z^{(k)}) \leq 0, \\ & z^{(k)} \geq 0, \\ \text{variables} \quad & z^{(k)}, \end{aligned}$$

which represents also constraints of each virtual network $k = 1, 2, \ldots, N$. Usually it is supposed that the objective function $O^{(k)}$ depends on both virtual link rates $z^{(k)}$ and virtual link capacity $O^{(k)}$. The objective is subject to a capacity constraint and possibly other constraints described in terms of $g^{(k)}(z^{(k)})$. The capacity constraint requires the link load

$$r^{(k)} = H^{(k)} z^{(k)}$$

to be no more than the allocated bandwidth. To compute $r^{(k)}$ we use routing indexes

$$H_{lj}^{(k)i} = \begin{cases} 1, \text{ if path } j \text{ of source } i \text{ in virtual network } k \text{ uses link } l, \\ 0, \text{ otherwise,} \end{cases}$$

and path rates $z_j^{(k)i}$ that determine for source i the amount of traffic directed over path j.

Now we formulate the optimization problem for the aggregate utility:

$$\begin{aligned} \text{maximize } & \textstyle\sum_{k=1}^{N} w^{(k)} O^{(k)}(z^{(k)}, y^{(k)}), \\ \text{subject to } & \textstyle\sum_{k=1}^{N} y^{(k)} \le C, \\ & H^{(k)} z^{(k)} \le y^{(k)},\ k = 1, 2, \ldots, N, \\ & g^{(k)}(z^{(k)}) \le 0,\ k = 1, 2, \ldots, N, \\ & z^{(k)} \ge 0,\ k = 1, 2, \ldots, N, \\ \text{variables } & z^{(k)},\ y^{(k)}\ k = 1, 2, \ldots, N. \end{aligned}$$

An optimization scheme follows directly from DaVinci principles. First, the substrate network determines how satisfied each virtual network is with its allocated bandwidth. Satisfaction level degree $\lambda_l^{(k)}$ (for link l of virtual network k) is one indicator that a virtual network may want more resources. Next, the substrate network determines how much bandwidth virtual network k should have on link l: the substrate network increases or decreases value $y_l^{(k)}$ in dependence on the satisfaction level $y_l^{(k)}$ on link l and the relative importance $w^{(k)}$ of virtual network k.

Given that each virtual network is acting independently, the question is whether virtual networks together with the bandwidth share adaptation performed by the substrate network actually maximize the overall performance objective.

4 A Fuzzy Approach to Decision Making on Link Level

Our work focuses on two traffic types: delay sensitive (the objective is to minimize the average delay) and throughput sensitive (the objective is to maximize the average link rate). Accordingly to DaVinci principles we consider two virtual nets ($N = 2$) for two types of traffic.

We denote by t_j, where $j = 1, 2, \ldots$, moments of system adaptation, i.e. moments of decision on bandwidth shares $y_l^{(1)}(t_j)$ and $y_l^{(2)}(t_j)$. Such decision is based on the results of monitoring the performance of the system during interval $[t_{j-1}, t_j]$ and values $y_l^{(1)}(t_{j-1})$ and $y_2^{(1)}(t_{j-1})$. We suppose that values $y_l^{(1)}(t_0)$ and $y_2^{(1)}(t_0)$ are given and consider the behavior of the system for $t \ge t_0$.

The first step of fuzzy decision making system design (see e.g. [14], [16]) is to define fuzzy variables. The proposed fuzzy solution requires three input variables for each link l and each traffic class k:

$U_l^{(k)}(t_j)$ - the average utilization of the virtual link;
$L_l^{(k)}(t_j)$ - the average length of queue;
$D_l^{(k)}(t_j)$ - the average delay of packets.

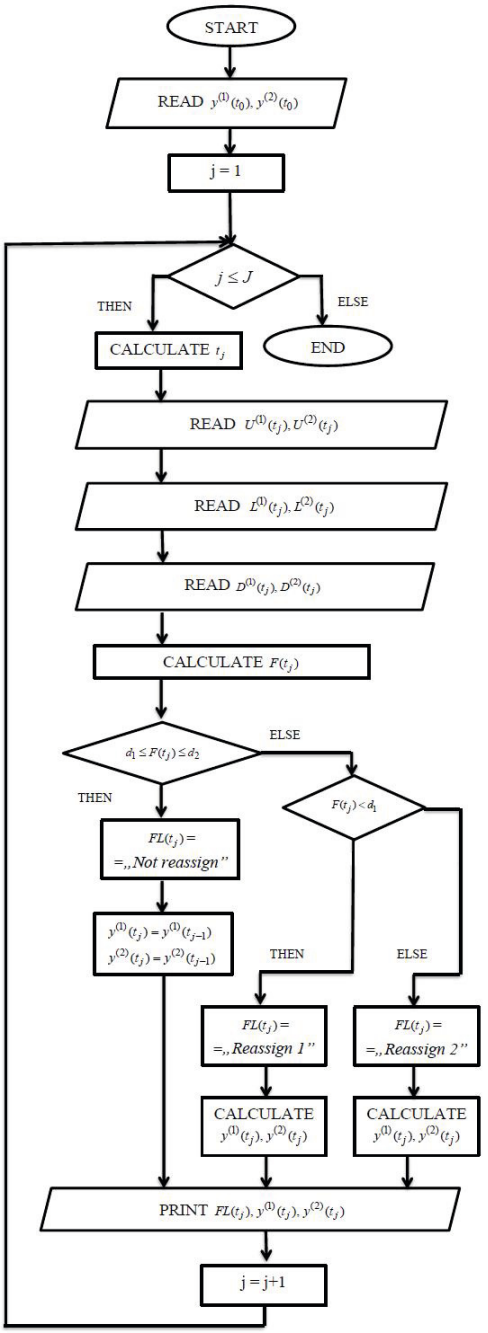


Fig. 1. Flowchart of decision process on link level

These variables are evaluated for the period $[t_{j-1}, t_j]$ and have linguistic values according to their membership functions. Input variables are used to describe the system state and the starting point of adaptation decision. We use the denotations $LU_l^{(k)}(t_j)$, $LL_l^{(k)}(t_j)$ and $LD_l^{(k)}(t_j)$ for the linguistic values of $U_l^{(k)}(t_j)$, $L_l^{(k)}(t_j)$ and $D_l^{(k)}(t_j)$ correspondingly.

The introduced fuzzy solution requires one output variable for each link l, which determines a fuzzy logic decision at t_j:

$F_l(t_j)$ - a fuzzy response, which is a defuzzified output value of the fuzzy inference module.

We use the denotation $LF_l(t_j)$ for the linguistic value of this variable.

The decision process on link level can be described by the following flowchart (see Fig. 1). Considering the link level we omit index l.

Now we determine the following parameters of decision making in order to achieve the advanced goals (see e.g. [14], [16]):

- the number of linguistic values and the membership functions of linguistic values for each input and output variable;
- the base of if-then rules;
- the method of defuzzification of output parameters;
- the type of decision making system.

The COG (center of gravity) defuzzification method was used in this investigation. We apply Mamdani fuzzy inference technique, introduced in [12], [13] (see also [14], [16]).

For all input variables we consider three linguistic values "Low", "Medium" and "High" (we use the denotation L,M and H as indexes). But for the output variable linguistic values are: "Reassign 1" (meaning that bandwidth share increases for the first type of traffic and decreases for the second type, correspondingly), "Not reassign" and "Reassign 2". The decision process on link level can be described by the following flowchart (see Fig. 1). Considering the link level we omit index l.

The membership function for each linguistic value is given by a triangular or trapezoidal fuzzy number. For example, Fig. 2 shows the graphs of the membership functions of linguistic values U_L, U_M, U_H for the utilization variables. In the form of trapezoidal fuzzy numbers:

$$U_L = (0, 0, 0.6, 0.8),$$
$$U_M = (0.7, 0.8, 0.8, 0.9),$$
$$U_H = (0.8, 0.9, 1.0, 1.0).$$

These membership function are considered to be independent on l, k and j. Obviously, such assumption is is not true for membership functions of linguistic values for the delay variables. The graphs of the membership functions used for linguistic values $D_L^{(1)}$, $D_M^{(1)}$, $D_H^{(1)}$, for a delay sensitive traffic are given by Fig. 3, but the graphs of $D_L^{(2)}$, $D_M^{(2)}$, $D_H^{(2)}$, for throughput sensitive traffic by Fig. 4.

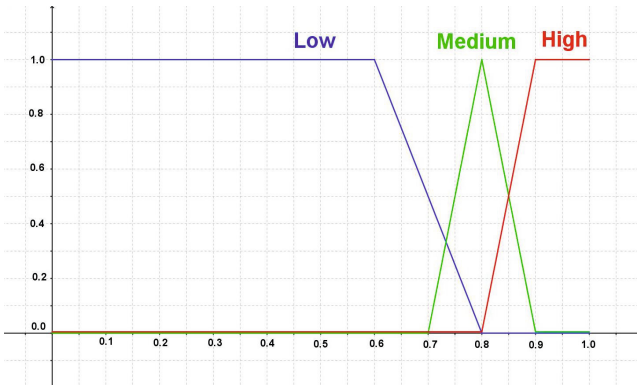


Fig. 2. Membership functions of the linguistic values U_L, U_M, U_H

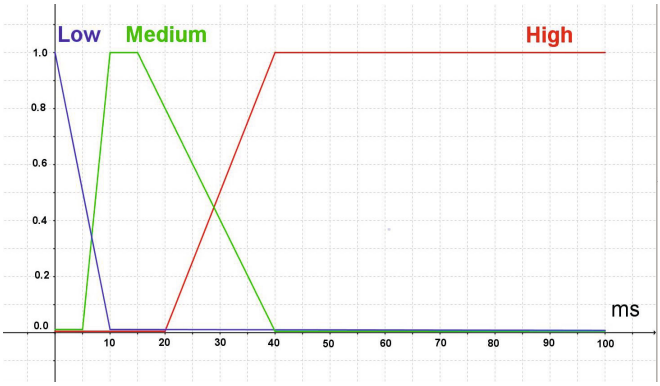


Fig. 3. Membership functions of the linguistic values D_L^1, D_M^1, D_H^1

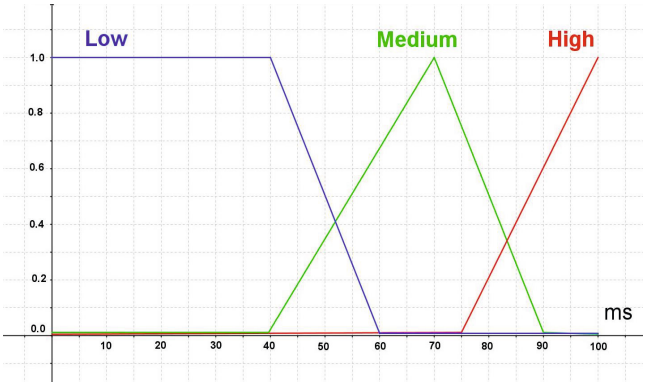


Fig. 4. Membership functions of the linguistic values D_L^2, D_M^2, D_H^2

The knowledge base for current implementation of the algorithm is made as rule database which is based only on the expert knowledge, where the experts are the authors of the paper, and the rules are assumed as logical assumptions, e.g.:

- if $UL^{(1)}$ ="High" and $LL^{(1)}$ ="Low" and DL^1 ="Low" and $UL^{(2)}$ ="High" and $LL^{(2)}$ ="Low" and $DL^{(2)}$ ="Low", then FL ="Not reassign";
- if$UL^{(1)}$ ="Low" and$LL^{(1)}$ ="Medium" andDL^1 ="High" and$UL^{(2)}$ ="High" and $LL^{(2)}$ ="Low" and $DL^{(2)}$ ="Medium", then FL ="Reassign 1";
- if $UL^{(1)}$ = "High" and $LL^{(1)}$ = "Low" and DL^1 = "Low" and $UL^{(2)}$ = "High" and $LL^{(2)}$ = "Medium" and $DL^{(2)}$ = "Low", then FL = "Reassign 2";
- if $UL^{(1)}$ = "High" and $LL^{(1)}$ = "Low" and DL^1 = "Medium" and $UL^{(2)}$ = "Medium" and $LL^{(2)}$ = "Low" and $DL^{(2)}$ = "Low", then FL = "Reassign 1";
- if $UL^{(1)}$ = "High" and $LL^{(1)}$ = "Low" and DL^1 = "Low" and $UL^{(2)}$ = "Medium" and $LL^{(2)}$ = "Low" and $DL^{(2)}$ = "Low", then FL = "Reassign 1".

The rules can be freely modified as well as the membership function definitions. The assumed fuzzy rules and the membership functions of input and output values were used for the proposed approach performance evaluation by simulation process and the impact of their modification is considered as the field for the future research.

We investigate the possibility to simplify the base of if-then rules by using inequalities for linguistic variables. For example, instead of two if-then rules

- if XL = "Low", then $YL = 0$,
- if XL = "Medium", then $YL = 0$,

it is possible to consider only one

- if $XL \leq$ "Medium", then $YL = 0$,

if the interpretation of the expression $XL \leq$ "Medium" is defined correctly. In this expression $\leq$ means "less than or equal to" or "not greater than". For understanding of if-then rules it is enough to define the membership function of the linguistic values "not greater than XL" and "not smaller than XL" for a given linguistic value XL. For XL, which is given in the form of trapezoidal fuzzy number

$$XL = (a, b, c, d),$$

we state the following meanings:

"Not greater than XL"$= (0, 0, c, d)$,
"Not smaller than XL"$= (a, b, +\infty, +\infty)$.

Fig. 5 and Fig. 6 illustrate the membership functions of the linguistic values "Not greater than U_M" and "Not smaller than U_M" for the linguistic value "Medium" given by Fig. 2.

We apply such fuzzy rule base simplification technique to our system of if-then rules. When a linguistic description is higherdimensional (i.e., when it consists

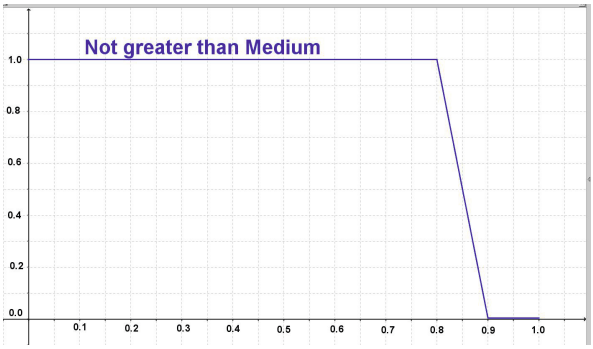


Fig. 5. Membership function of the linguistic value "Not greater than U_M"

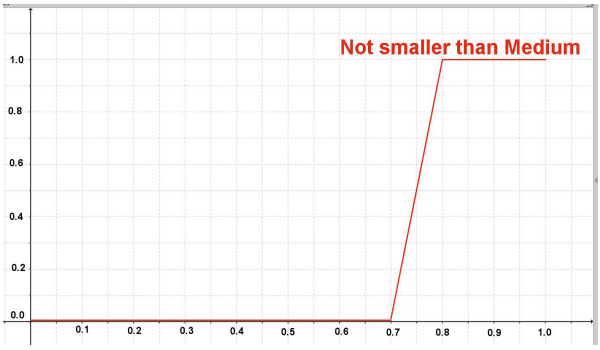


Fig. 6. Membership function of the linguistic value "Not smaller than U_M"

of if-then rules with more than one antecedent variables) then this method is not so simple. The proposed simplification framework leads to very good fuzzy rule base reduction also in our case with six antecedent variables.

In two next Sections of the paper we focus mainly on practical realization of the proposed approach by means of Coloured Petri Nets (CPN) using CPN Tools.

5 Coloured Petri Nets Based Modeling

The concept of Coloured Petri Nets, developed by K. Jensen (see e.g. [7]), is an extended version of classical Petri Nets. Additional to places, transitions and tokens, the concept of types or colour sets has been added. This concept allows to include simple or complex information into the tokens and gives the possibility to use tokens that carry data values and can therefore be distinguished from one another. Each token could be attached with a colour, representing the identity of the particular token. Furthermore, each place and each transition has attached

a set of colours. A transition can fire according to each of its colours. When a transition fires, tokens are removed from the input places and added to the output places in the same way as it happens in original Petri Nets, with the exception that a functional dependency is specified between the colour of the transition that fires and the colours of the involved tokens.

A Coloured Petri Net is a tuple

$$CPN = (P, T, V, \Sigma, W, C, G, H, I)$$

fulfilling the following requirements:

- P is a finite set of places;
- T is a finite set of transitions;
- V is a set of directed arcs;
- Σ is a finite set of types (colour sets);
- W is a finite set of typed variables, $Type : W \rightarrow \Sigma$ is a type function assigning types (colour sets) to variables;
- $C : P \rightarrow 2^{\Sigma}$ is a colour function assigning colour sets to each place;
- $G : T \rightarrow EXPR(W)$ is a guard function assigning a guard to each transition;
- $H : V \rightarrow EXPR(W)$ is an arc expression function assigning an expression for each arc;
- I is an initialisation function assigning an initial marking to each place.

Triple (P, V, T) constitutes the net structure, pair (Σ, W) describes types and variables and tuple (C, G, H, I) defines the net inscriptions. In this case we overleap the details and the explanations on marking iterations and do not discuss how transitions alter the marking of places. Due to this iteration scheme, CPN is one of efficient mathematical modeling languages for the description of discrete event systems. CPN combines a well-developed mathematical theory with an outstanding graphical representation. This combination is without doubt the key reason for the huge success of CPN in modeling of the dynamic behaviour of systems [4], [7], [9].

Coloured Petri Nets, proposed by Kurt Jensen, have been developed by the CPN group at Aarhus University, Denmark since 1979. The first version was a part of the PhD Thesis of Kurt Jensen, which was published in 1981. The CPN group has developed and distributed industrial-strength computer tools, such as Design/CPN in 1990 and CPN Tools in 2003. Our simulation scheme is based on Coloured Petri Nets Tools [9], [15]. We extend CPN Tools model by adding specially designed bandwidth adaptation module, which executes bandwidth allocation mechanism, which is described in the previous Section. As a result, we obtain a discrete event modeling computer tool, which supports interactive and automatic simulations, state spaces and performance analysis and combining Coloured Petri Nets and fuzzy logic based decision making system. By making simulations of networks CPN models with this additional module it is possible to investigate different scenarios and explore the behaviour of the system, in order to use simulation based performance analysis for further decision making and adaptation processes.

6 Coloured Petri Nets Based Simulation Scheme

The goal of our simulation experiment is to obtain a visualization of virtual network switching due to demand for extra link bandwidth. To simplify the issue we simulate the adaptive bandwidth allocation on link level. The simulation study concentrates on two traffic types: delay sensitive and throughput sensitive. We consider two virtual nets for two types of traffic denoted by A and B. Up to now we experiment with two nodes topology (Fig. 7) and use the following notations:

G_A, G_B - packet generators;
D_A, D_B - destination nodes.

Colours A and B can be effectively used for modeling and simulating such systems [1]. The colours are associated with the tokens which in their turn represent packets. According to DaVinci architecture all data packets which are generated by both generators are handled and transmitted over virtual links. The queuing model of each virtual link is described independently. The CPN Tools based model of the system is given in Fig. 8. Packets are generated by the traffic generators (Arrivals) and stored in the FIFO (first in, first out) queues (Buffer). The goal is to route packets to the output port (Transmitted). Packets cross the FIFO queue and are transmitted under the condition that the transmission link is free.

The transmission time depends on the size of a packet and the bandwidth of the corresponding virtual link and is calculated for each packet transmission. Module Link Load Data assigns a bandwidth value for each virtual link. Two traffic FIFO queues and two virtual links are separated due to colours A and B and two flows of packets are parallel controlled and analyzed.

This model involves bandwidth shares adaptation module (see Fig. 8). The adaptation scheme is realized accordingly to the proposed fuzzy logic based algorithm described by Fig. 1. Bandwidth allocation adaptation modules are designed to update the virtual link resources allocation for dynamically changing traffics. Special predicate and observation functions and data collection monitors

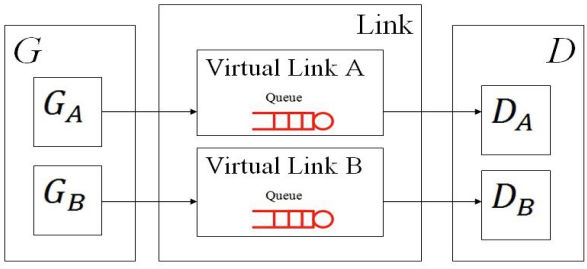


Fig. 7. Simulation scheme for two nodes topology

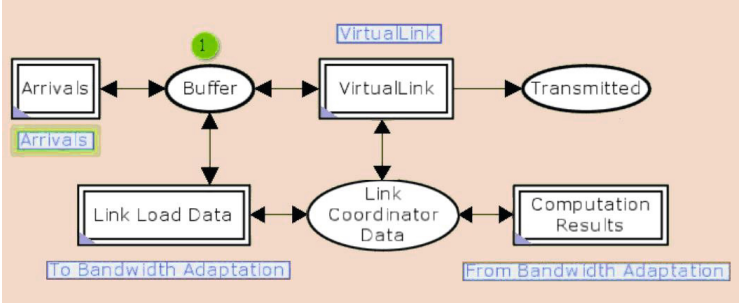


Fig. 8. Bandwidth adaptation simulation scheme using Coloured Petri Net Tools

(Buffer Delay, Buffer Length, Virtual Link Utilization) are included for monitoring the performance of the system. A monitoring mechanism is used not only to control, but also to modify a simulation of the net. It is done by reassigning bandwidth shares between virtual links A and B. A decision making system is based on data collection monitors that allow to calculate the system performance measures such as the delay in each queue, the length of each queue, the utilization of each link.

Comparative result analysis is provided for DaVinci model realization with and without the use of fuzzy logic. The results of the comparative study, estimating the performance of both the fuzzy logic algorithm and the classic algorithm for the number of decisions equal to 500, are illustrated with Fig. 9 and Fig. 10. Carrying out the analysis of the average values, we can see that the average level of packet delays for delay sensitive traffic (for the virtual link A) decrease from 30 ms in the case of the classic algorithm to 23 ms in the case of the fuzzy logic based algorithm (see Fig. 9). At the same time the average level of utilization for throughput sensitive traffic (for the virtual link B) stabilizes at 0.96, showing the better result in comparison with 0.92 utilization level when using the classic algorithm (see Fig. 10).

Before comparing DaVinci model realization with and without the use of fuzzy logic, the analysis and comparison of results with static and DaVinci model realization without the use of fuzzy logic was performed. Network characteristics were analysed while using static and dynamic network resource management techniques. The results of the comparative study, estimating the performance of both the static and the DaVinci algorithms for the number of decisions equal to 500, are illustrated with Fig. 11 and Fig. 12. DaVinci approach, opposing the statistical approach, foresees a periodical resource reallocation between parallel links during the whole virtual network lifecycle. The resources are reallocated depending on link traffic load and request number. As a result the system became more stable than the statistical and the adaptive reallocation of resources guaranteed the increase of the system functionality efficiency. The obtained results give a picture of the two resource allocation approaches obvious difference and the DaVinci approach advantages.

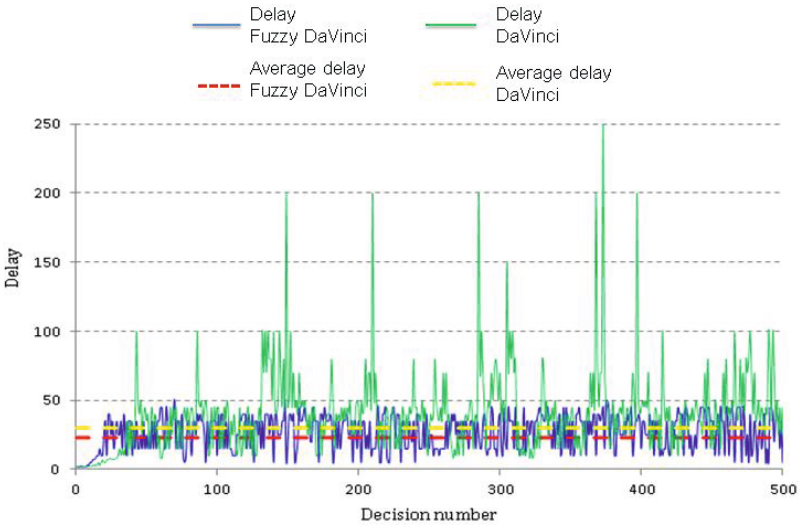


Fig. 9. Simulation results: packet delays (average delays) in the virtual link A (delay sensitive traffic) in the case of DaVinci and Fuzzy DaVinci allocation

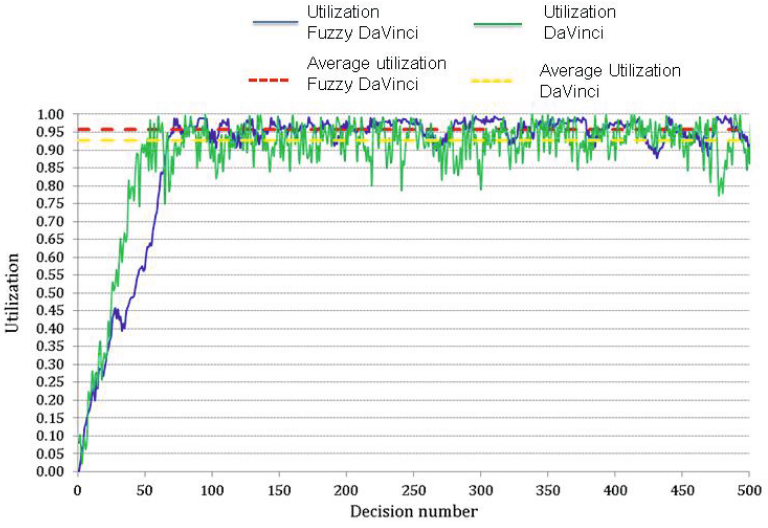


Fig. 10. Simulation results: utilization (average utilization) in the virtual link B (throughput sensitive traffic) in the case of DaVinci and Fuzzy DaVinci allocation

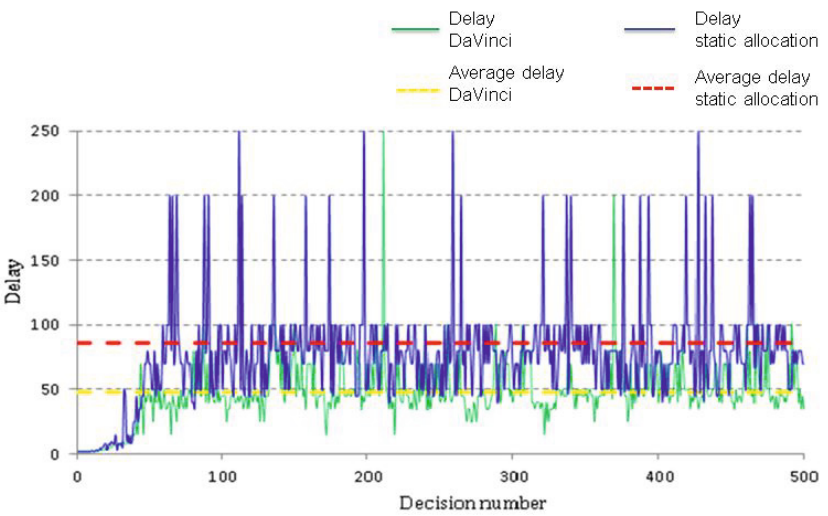


Fig. 11. Simulation results: packet delays (average delays) in the virtual link A (delay sensitive traffic) in the case of static and DaVinci allocation

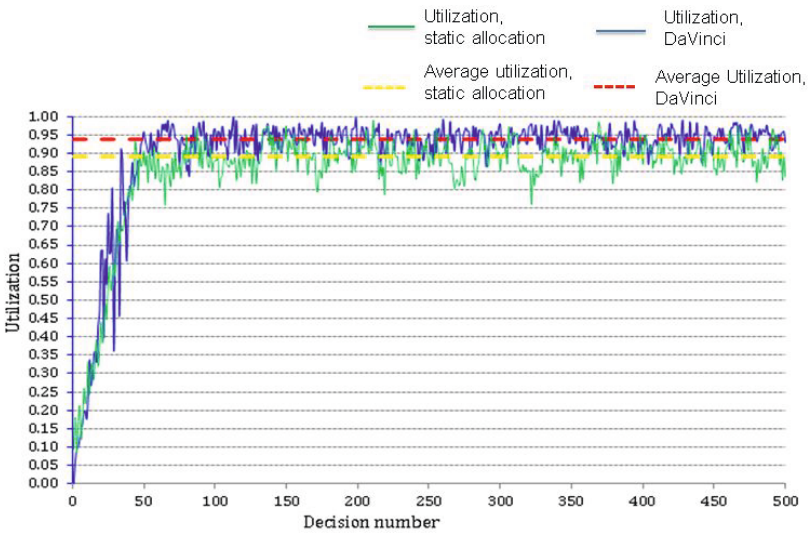


Fig. 12. Simulation results: utilization (average utilization) in the virtual link B (throughput sensitive traffic) in the case of static and DaVinci allocation

In our experiments the initial resource allocation to virtual links is uniform, we set the capacity of the physical network as 100 Mbps. Packets A and B are generated with exponentially distributed arrival time and uniform distributed size. An important issue is the frequency of adaptation. Bandwidth resources are reassigned every 10000 MTU. By changing traffic parameters we observe the adaptation process. The simulations show that the proposed fuzzy logic based adaptation scheme has good results. We experiment with delay sensitive traffic and throughput sensitive traffic as A and B correspondingly, as well as with two delay sensitive traffic classes and with two throughput sensitive traffic classes. Our simulation results clearly show that the adaptive bandwidth allocation mechanism can dynamically and efficiently react to traffic changes in both cases: when traffic classes are with different performance objectives or with the same one.

7 Conclusion

In this paper we presented the simulation scheme of an adaptive bandwidth allocation mechanism which is implemented for two nodes topology. Our aim was to create a fuzzy logic driven controller, which could be used to modify DaVinci network model with two virtual links. Coloured Petri Nets (CPN Tools) were used to realize the DaVinci model, which was based on two virtual networks. With the use of simulation experiments optimization mechanisms of network resource allocation were analysed. Network characteristics were analysed while using static and dynamic network resource management techniques. DaVinci model, when improved with fuzzy logic based controller, gained the feasibility of selective traffic management technique, which provided the possibility of proactive network resource distribution, while maintaining QoS parameters within the feasible limits. Comparative result analysis is provided for DaVinci model realization with and without the use of fuzzy logic based controller.

The experimental realization of the fuzzy logic based algorithm showed that in comparison with the classic algorithm packet delays for delay sensitive traffic at the average level decreased for 23 percents, at the same time the link utilization for throughput sensitive traffic at the average level increased for 5 percents. As the result, fuzzy logic effectiveness is illustrated, if compared to the classic DaVinci model realization while using both network resource management techniques for two nodes topology and two virtual networks.

Our future work will mainly focus on extension of this simulation scheme in the following directions: from two traffic classes to multiple traffic classes and from the local link level to a global network level. Also the modification of the base of if-then rules and membership functions of decision variables is required. At the same time our aim is to modify and improve monitoring and decision making systems in order to optimize adaptive bandwidth management, to develop the experimental network system and simulation environment as well as to accomplish the practical realization of the classic and fuzzy logic based decision making systems under similar conditions to conduct a further comparison of the

produced results. Comparing with the classical decision making system the fuzzy solution, which is based on the successive evaluation of multiple parameters and large scale bases of if-then rules, can ensure fast and effective real-time modification of traffic management policy within the MPLS routers, adopting it to an unsteady network environment.

References

1. Asmuss, J., Zagorskis, V., Lauks, G.: Simulation of dynamically adaptive bandwidth allocation protocols using Coloured Petri Nets. In: Proceedings of the 24th European Modeling and Simulation Symposium, EMSS 2012, pp. 408–413 (2012)
2. Chowdhury, M., Boutaba, R.: A survey of network virtualization. Computer Networks 54(5), 862–876 (2010)
3. Dramitinos, M.L.: A bandwidth allocation mechanism for 4G. In: Proceedings of European Wireless Technology Conference, pp. 96–99 (2009)
4. Gehlo, V., Nigro, C.: An introduction to system modeling and simulation with coloured Petri nets. In: Proceedings of Winter Simulation Conference, pp. 104–118 (2010)
5. Haider, A., Potter, R., Nakao, A.: Challenges in resource allocation in network virtualization. In: Proceedings of the 20th ITC Specialist Seminar, Hoi An, Vietnam (2009)
6. He, J., Zhang-Shen, R., Li, Y., Lee, C.-Y., Rexford, J., Chiang, M.: DaVinci: dynamically adaptive virtual networks for a customized Internet. In: ACM CoNEXT Conference, New York (2008)
7. Jensen, K.: Coloured Petri nets: basic concepts, analysis methods and practical use, vol. 13. Springer (1992, 1997)
8. Jensen, K., Kristensen, L.: Coloured Petri nets. Modelling and validation of concurrent systems. Springer (2009)
9. Jensen, K., Kristensen, L., Wells, L.: Coloured Petri Nets and CPN Tools for modelling and validation of concurrent systems. Int. J. on Software Tools for Technology Transfer 9, 213–254 (2007)
10. Lin, X., Shroff, N.: Utility maximization for communication networks with multipath routing. IEEE Trans. Automatic Control 51, 766–781 (2006)
11. Liu, W., Li, S., Xiang, Y., Tang, X.: Dynamically adaptive bandwidth allocation in network virtualization environment. Advances in Information Sciences and Service Sciences 4(1), 10 (2012)
12. Mamdani, E.H.: Applications of fuzzy algorithms for control of simple dynamic plant. Proceedings IEEE 121(12), 1585–1588 (1974)
13. Mamdani, E.H., Assilian, S.: An experiment in linguistic synthesis with a fuzzy logic controller. Int. J. of Man-Machine Studies 7(1), 1–13 (1975)
14. Piegat, A.: Fuzzy modeling and control. Physica-Verlag, Heidelberg (2001)
15. Vinter Ratzer, A., Wells, L., Lassen, H.M., Laursen, M., Qvortrup, J.F., Stissing, M.S., Westergaard, M., Christensen, S., Jensen, K.: CPN Tools for editing, simulating and analysing coloured Petri net. In: van der Aalst, W.M.P., Best, E. (eds.) ICATPN 2003. LNCS, vol. 2679, pp. 450–462. Springer, Heidelberg (2003)
16. Ross, T.E.: Fuzzy logic with engineering applications. McGraw-Hill, New York (1995)

17. Szeto, W., Iraqi, Y., Boutaba, R.: A multi-commodity flow based approach to virtual network resource allocation. In: Proceedings of the Global Communications Conference IEEE GlobeCom, pp. 3004–3008 (2003)
18. Zhang, Y., Wang, C., Gao, Y.: A virtualized network architecture for improving QoS. Int. J. on Information Engineering and Electronic Business 1, 17–24 (2009)
19. Zhou, Y., Li, Y., Sun, G., Jin, D., Su, L., Zeng, L.: Game theory based bandwidth allocation scheme for network virtualization. In: Proceedings of the Global Communications Conference IEEE GlobeCom, Miami, Florida, USA (2010)
20. Zhu, Y., Ammar, M.: Algorithms for assigning substrate network resources to virtual network components. In: Proceedings of the 25th IEEE International Conference on Computer Communications, pp. 1–12 (2006)

Finding Relevant Dimensions in Application Service Management Control

Tomasz D. Sikora and George D. Magoulas

Department of Computer Science and Information Systems
Birkbeck, University of London
Malet Street, London WC1E 7HX
sikora.t@gmail.com, gmagoulas@dcs.bbk.ac.uk

Abstract. The increased interest in autonomous control in Application Service Management environments has driven a demand for analysis of multivariate datasets in this area. This paper proposes a feature selection method using metrics time series analysis. The method exploits four metrics called Similarity, Dependency, Consequence, Interference which are combined in order to perform a multivariate evaluation. This allows more efficient search for similarities in the time-series, selection of most relevant dimensions, and easier control in the reduced space, which would ultimately reduce maintenance effort. This is further used to create causal models of the controlled system, significantly simplifying evaluation of defined elements utilization dependencies. We show that methods based on these metrics can be applied in service control practice under several scenarios.

Keywords: Application Service Management, Time Series, Dimensionality Reduction, Feature Selection, Adaptive Controller, Service Level Agreement, Performance, Metrics.

1 Introduction

Enterprise systems, employed by a large organization to offer high quality of computing services, are typically expected to be capable of handling large number of calls and volumes of data supporting the organization business processes. More and more enterprises support Software as a Services (SaaS) model where provided functionalities are exposed to external or internal clients, deployed on in-house infrastructure or with use of (Private, Public or Hybrid) Cloud based environments. Rapid changes in the market cause constant need for adjustment or even redesign and reimplementation of significant parts of the systems. Very often organizations release newer versions, cyclically modifying existing implementations of their internal business processes. Because of the frequency of changes, the need for high quality of service, complex distributed integration architectures, implemented processes and used algorithms, the responsibility for maintenance and development is often taken by dedicated teams working for the organization. Such teams of system operators and developers work closely

L. Chen, S. Kapoor, and R. Bhatia (eds.), *Intelligent Systems for Science and Information,* Studies in Computational Intelligence 542,
DOI: 10.1007/978-3-319-04702-7_19,

together on collected business requirements, delivering implementations for business needs and solve faced technical issues.

Ideally metrics are acquired from all significant elements of existing distributed systems infrastructure and from all tiers of their architectures. Starting from the physical tier, through the virtualization tier, operating system metrics, services tier (database, application servers, web containers, messaging systems, etc.) and finishing on the application tier metrics (i.e. active users count, front-end execution count/time, application external/internal APIs and web/facade interfaces execution count/time) [1] [2] [3].

A form of control, either manual or automatic, is essential to maintain system stability and optimize quality of service under changing conditions. Diagnostic, monitoring, and auditing components of enterprise class systems may produce thousands of metrics dimensions forming a knowledge base, which is being constantly evaluated in order to specify the control rules. Application Service Management (ASM) is a discipline which focuses on monitoring and managing the performance and quality of service in complex enterprise systems. An ASM controller needs to react adaptively to changing system conditions in order to optimize a defined set of Service Level Agreements (SLA) as objective functions.

Previous authors' work on autonomous controller in ASM field focuses on a blackbox approach, where no model of the system is present [4].A neural-control system learns by observation of actions, resources utilization and reactions of the applied control. On that basis further control rules are generated and sent to actuators installed in the application runtime environment. Multidimensional similarity matching of metrics signals deserves special attention because it gives an opportunity to view a sequence of changes which were associated with a specific event or group of actions which occurred in the system, so only most relevant dimensions can be considered. In this chapter we propose a novel feature selection approach based on metrics time series analysis. The method performs multivariate evaluation calculating the strength of dependency between set dimensions investigating metrics sequences for introduced measures: Similarity, Consequence, Interference, and Clarity (SCIC). So far such time series analysis for dimension reduction as a preliminary processing for the benefit of autonomous control phase in ASM has not been researched.

In order to apply efficient and constructive (more stable) control there must be established knowledge about the system run-time characteristics, as each change can impact other elements. A typical enterprise ASM control system can define hundreds to thousands of distinct actions, tens to hundreds of resources, and tens to hundreds of SLAs used as both key performance indicators and for revenue recognition. Due to many non-linearities autonomous control in such systems has to be applied in a background of well recognized reduced space, as a multidimensional space is difficult to control/regulate. The smaller the number of dimensions, the more rigid the control applied [5]. Thus the right selection of the most relevant dimensions for the autonomous ASM control is crucial. It is imperative to identify which elements (especially controllable actions) have the biggest impact on SLA (indirectly through resources utilization).

The chapter is organized as follows. Section 2 describes the fundamentals of the problem area. Sections 3 shows previous work done in the related areas. Section 4 defines the method proposed. Section 5 illustrate aspects of the performance of SCIC approach with real and synthetic data, respectively. Section 6 highlights some open areas for future work. Section 7 ends the chapter by presenting a few concluding remarks.

2 Problem Definition

There are many complex interdependencies in an enterprise system. In order to illustrate the problem we take, as an example, a simple model of operation where (a) incoming requests on actions cause higher resources utilization, (b) higher utilization over certain limit causes longer execution times in the system, and effectively higher SLA penalties. Of course following (b) there is an impact on actions execution and therefore on requests count (this has usability implications, i.e. user clicks but the system does not respond in timely fashion, so users tend to click again). Also, actions can be substantially driven by input parameters values depending on functional specifics. We will not consider these aspects in this book chapter due to space limitations.

The above assumption formulates a causal model chain [6]. Thus the typical causal model of enterprise class system can be defined by the following equation:

$$a \rightarrow r \rightarrow f_{SLA} \tag{1}$$

Set of actions $a = a(t)$ cause effects on resources $r = r(a, r)$. Actions and resources utilization cause changes in SLA functions values $f_{SLA} = f_{SLA}(a, r, t)$. Some actions are controllable $a_c \in a$.

The main purpose of this study is to find dependency measure which can be used to investigate dimensions relevance for better (more focused) control, where relevance criteria are similarity between metrics sequences and strength of utilizations dependence. Dimensionality reduction is applied to the field by feature selection [7], so only the most depending on controllable actions and therefore relevant for control are selected.

Although the control approach researched does not contain an analytic model of the system, cause-and-effect dependencies between action requests, resources utilization and SLA are defined. Effectively the general causality direction is here quite well set. Consequently events or related changes in the system can be investigated using similarity of metrics changes in time. Even though SLA definitions are clearly defined and directly related to parameters used, the non-linear system especially under high load may react surprisingly unstable. Therefore a method for finding causality strength by matching metrics variations (selecting events times), so actuators can be set in areas where resources and SLA were found having the biggest causality, thus control should have the strongest effect.

Every event detection approach needs specific treatment, as events nature differs amongst problems [8]. Thus it is important to note how we define an event. There are two significant factors in ASM area which we assume in this

work: (a) similar signals slopes and shapes (this can be tested additionally with first/second derivative, described in section 4.1), and (b) first appearance of an effect is immediate (no delays).

Four frequently occurring effects/situations which outline time series structure in the ASM field are: (a) Full Match: no delays, this is normal situation when all necessary resources are available, which can be rare in highly utilized environments, (b) Shift: delays occur due to some internal system characteristics (i.e. actions execution time → disk queue), (c) Extension: mainly when system is highly utilized, more and more actions wait for execution (i.e. actions execution time → CPU consumption), (d) Saturation: problem often occurring together with point c, when a bottleneck issue is faced.

It is important to highlight that pure sequence similarity or correlation is not sufficient to measure dimensions dependency. This is an issue that is investigated later in section 4.1. Correlation on the whole dataset does not focus on sequences of signal changes, so there is a need to consider time dimension and interdependencies between dimensions (other events impacts). It is necessary to investigate the changes of signals in time, see section 4.1 and Fig. 1.

Even though process monitoring, and non model-based control has been researched thoroughly [9], application of dimensionality reduction preceding control in the ASM area have not been widely studied yet.

3 Previous Work

In this work we exploit ideas from causal modeling, event detection, time series processing and establishing similarities (by correlation and distances measures), therefore it is necessary to review briefly most notable works from relevant areas.

Cause-and-effect relationship were researched widely in the past; statistical analysis theory was applied to such disciplines as econometrics, finance, control theory and many more [10]. The core finding is that causality is not guaranteed by correlation. In ASM field though we know the causal chain a priori, so investigating changes in time domain can infer events [11].

On the basis of causality models it was proposed a Path Analysis, which helped establishing path coefficients. This was studied by Wright (1934) who introduced a simple set of path tracing rules, for calculating the correlation between two variables [12] [13]. Pearl has proposed a theory of causal inference, formulating causal modeling and cause–effect relations, based on directed graph definitions [6]. Granger proposed a pragmatic test for mainly economics use as a new model of causation applying cross-spectral methods [14]. Extended Granger causality was researched by Chen in the domain of multiple nonlinear time series analysis [15]. More recently, Eichler at al. worked on Granger Causality and path diagrams for multivariate time series analysis [16]. Guyon at al. examined areas in which the knowledge of causal relationships benefits feature selection, by explaining relevance in terms of causal mechanisms, distinguishing between features, predicting the consequences of actions performed by external agents [17].

Event detection and distance measures in time series is the core area of this work. Amongst a variety of works in this field, an interesting distribution based outliers filtering method was studied by Ihler et al. for adaptive event detection [18]. Knowledge Based Event detection in complex time-series was researched by Hunter [19]. Correlation ranking for feature reduction were investigated by Geng et al [20], and Wei et al. [21]. Multivariate time series in process controllability was researched by Seppalla at al. who investigated time series methods for dynamic analysis of multiple controlled variables to enhance feedback controllers efficiency [22].

For the problem expressed above it is important to review distance measures used in time series processing. Simple general form metric distances, like Minkowski, L1 norm (Manhattan Distance), L2 norm (Euclidean Distance), L∞ norm (Supremum Distance) are natural, simple to use, producing distance which is calculated without a shift (delay). Dynamic Time Warping (DTW), which provides distance between two time series which is less sensitive to lags (effect delays), was researched by Berndt and Clifford (1994) [23], and Keogh at al. (2002) [24]. Piecewise Linear Approximation (PLA) for segmentation of plane curves studied by Pavlidis and Horowitz (1974) [25], a modification of DTW as Piecewise Aggregate Approximation (PAA) by Keogh et al. (2000) [26], Adaptive Piecewise Constant Approximation (APCA) introduced by Keogh et al. (2001) [27], Elastic Partial Matching (EPM) [28], are a few examples of widely used similarity measures methods in time series field. More details and references of comparison of representations and distance measures a reader can find in paper by Ding at al. [8].

Other methods include correlation-based feature selection for machine learning studied by Hall (1999) [29] and dimensionality reduction for fast similarity search in large time series databases researched by Keogh (2001) [30]. A review of correlation as distance metric in non-linear time series analysis can be found in [31].

Events correlation in ASM was researched by Grushke in 1998. He described an event management framework added to a system where a component correlating events based on a dependency graph is the core element [32]. The paper does not mention times series analysis nor reduction of the space being controlled. Similarly Keller and Kar worked on dependency query facility providing query mechanisms for dependency models definitions as consolidated dependency graph, which can be used as input for event correlators and service management frameworks [33]. Agarwal at al. researched a method using co-occurrence and relevance scores to learn and match fault patterns for automatically generating "change points" defining problem signatures using administrator feedback [34], but the research did not focus on providing autonomous control. Jiang et al. propose a rank building mechanism for fault management considering threshold based rules, so the importance of alerts can be better assessed [35]. Kiciman in his PhD thesis [36] uses extensively correlation techniques in times series analysis, in order to detect usage anomalies, and goodness-of-fit test to confirm deviation between showing component interactions, for more focused application monitoring.

4 ASM Control Approach

Time series analysis for dimension reduction as a preliminary processing for the benefit of autonomous control phase in ASM field has not been researched yet. A novel method is introduced below to address the high dimensionality problem.

Before the ASM control system can be started, the metrics time series repository must contain data collected during normal operating system run. In order to secure the knowledge about boundary criteria, it is strongly advised to run as many actions in isolation to each other. This information is most valuable since it describes not-interfered impacts of actions on the system state. So the evaluator is able to detect which elements of the system use which resources, and effectively establish causal dependencies in the system. The steps of the procedure are listed below.

1. Isolated run – in this step the metrics evaluation process collects data about the impact of various actions running in isolation; discussed in Section 4.1. Planned activities such as user acceptance testing (UAT), performance load and stress testing on the final infrastructure are recommended in this phase.
2. Normal work – the control system collects data about normal operating conditions, so it builds knowledge about casual work signals distributions and interferences between actions.
3. Dimensions selection – this step focuses on statistical evaluation and dimension selection procedures which are needed to reduce the dimensionality of the control space.
4. Controlled run – the controller adapts the application to the changing conditions; the adaptation is set up to consider only metrics gathered in the last time window of activity, which is reseted after changed functionalities during next releases.

In this chapter we describe a method tackling the dimensions selection step (3). Controller run step (4) was addressed by the authors in [4].

4.1 Method Description

The proposed approach is based on calculating similarities between measured dimensions and ascertaining strengths of causal relations. To this end we introduce, for example, the Clarity measure which depends not only on how source and output signals dimensions match each other, but also on how consistent is the similarity relation against time and if the similarity is not interfered by other substantially similar signals in a given time window. This approach allows to establish purity of impact dependencies, focusing on isolation of patterns of interest (high similarity) from the background of other signals.

Hence, the proposed algorithm establishes the dependency of dimensions based on metrics times series – source s and output o signals are input parameters. Signal pairs are measured using four different factors: Similarity, Consequence, Interference and the final score Clarity. Thus the method is called SCIC. Computed values describe edges in the assumed causal model graph. When Clarity values are

computed a ranking is built, which directly defines the relevance of dimensions between the tiers of the examined system.

Although the similarity match is not strictly statistical approach, we can uncover some structure of causality in the signals changes against time [6]. To infer causality we assume that signals representing activity, resources consumption or services utilizations should contain similarly monotonically behaving components.

To improve the efficiency of the main algorithm collected data are preprocessed in two steps. Firstly data are linearly scaled (normalized) so all amplitudes are standardized to the range $[0, 1]$. This step is to ensure that all dimensions are being equally evaluated (curves matching) regardless of actual values ranges which can varies widely. Secondly the data are denoised, assuming that if any of the metrics values in the time series were not collected (null value) a previous non-null value is taken into consideration. During the preliminary investigations it was found that fair results are obtained by Simple Moving Average (SMA) [37]. Similarity check of denoised signals gives much better results, as more underlying causative features are visible, see Fig. 1.

The algorithm is searching for the strength of signals causation dependencies:

1. For each of the dimensions of sets of actions a, set of resources r, and set of SLA functions f_{SLA} calculate Similarity S, Consequence κ, and Interference I measures between pairs of source and output signals. Build matrix of the final relation strength – called Clarity C. All measures are calculated in steps, using sliding window approach of constant n-width.
2. Build ranking based on descending order of strength of clarity causation. If any of the output dimensions correlate with other, than this with lower clarity value is not considered in the first order.

The Similarity measure S (simx) between two metrics time series, source s and output o signals, is $S_{(s,o)} = \sum_i \mathbf{S}_{(s_i,o_i)}$, where s_i, o_i are the i-th window of source and output vectors, and the i-th vector of similarity series $\mathbf{S}$ is calculated according to the following formula:

$$\mathbf{S}_{(s_i,o_i)} = 1/(\delta(\hat{s_i}, \hat{o_i}) + 1) * \overline{\hat{s_i}} * \overline{\hat{o_i}}, \tag{2}$$

where $\hat{s_i} = (s_i - \min s)/(\max s - \min s)$ is a scaled (normalized) vector of signals s, $\delta(s_i, o_i)$ is distance function of source s and output o signals in the i-th sliding window. We evaluated the simple euclidean distance, the DWT [24], and the distance derived from Pearson's, Kendall's, and Spearman's correlation coefficients. Fig. 2 presents comparison of distances tested.

In order to improve source and output signal shapes check, another step which is based on root-finding of the first derivative of smoothed signal functions can be optionally added. This minimize a negative effect of high similarity of long high level signals. In such situations distances would be mainly influenced by one of the high signals values (see eq. 2). This step requires the original signals to be with

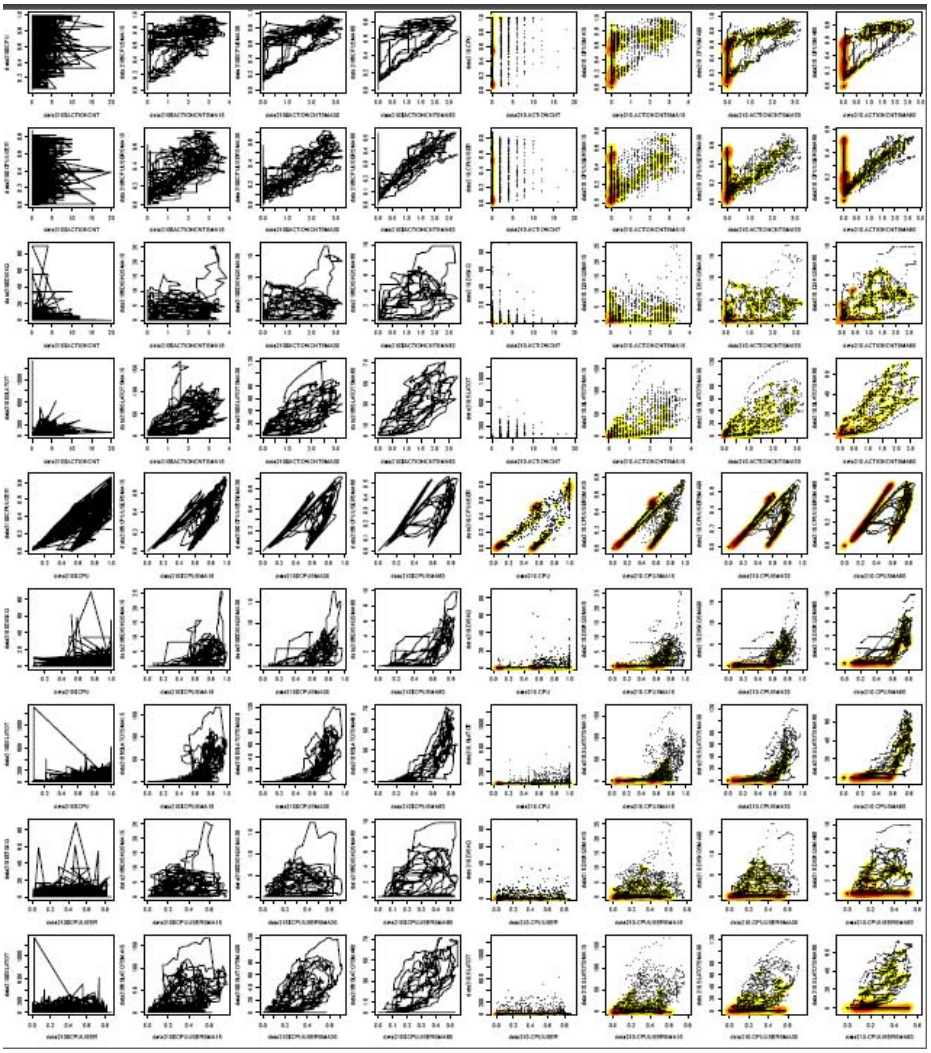


Fig. 1. Comparison of four different SMA window size: 0 (no SMA), 15, 30, and 60 sec, applied to time-series of metrics data, and denoted respectively in dimensions names suffixes. The figure contains two views trajectory-maps as vectors between system state, showing sequence of changes (four columns on left), and measurements distributions, showing statistical properties of measurements apportioning (four columns on right). Each row shows a different bivariate trajectory-map/distributions illustrating the relationship between a source and an output dimension (signal). The wider the moving average window is the less noisy the data get and clearer is the relation between dimension pairs of high causality. By applying SMA we can uncover some characteristics often hidden in raw data representing system states. Data presented in second example scenario, see section 5.2.

use of Cubic Smoothing Spline[1] [38] $\mu_{s_i} = \mu(s_i)$, then the first derivative μ'_{s_i} root is calculated with use of linear interpolation approach.

Further enhancement of the method is a measure built by calculating the degree of cross-correlation. This optional extension of Similarity confirms dependence of output from input signals evaluated. The Dependence measure is calculated following the key assumption of causation theory that in a causal system a response never precedes an input [6]. Thus the measure focus on checking positive lags of output to input signal. Dependency is a result of cross-correlation [10] on input and output signals being evaluated, where the maximal matching lag (the biggest cross-correlation lag in a declared time window) derive the Dependency strength: $D = (l_{max} - t_l)/l_{max}$, where t_l is the maximum product of cross-correlation function estimation $t_l = \max(ccf(s_i, o_i, l_{max})) : t_l \geq 0$ in a set maximum lag window l_{max}. Of course negative lags are not considered, as are found against the assumed causality chain. Shorter lags (delays) are promoted over longer ones, so Dependency is the highest when $t_l = 0$ and minimal, if lag is equal to the set maximal window length $t_l = l_{max}$. Conceptually Dependency D is very related to the Similarity measure, and as a scaling coefficient of the enhanced Similarity $S_{[D]} = S * D$ directly impacts the Clarity measure C explained later (see eq. 5). This is a significant extension of SCIC method, therefore SCIC using the measure is denoted as SDCIC.

The Consequence measure κ (conx) of similarity is a scalar value, indicating how often higher source signal s causes higher output o in the whole time scale; the more often similarity $S_{(s,o)}$ (effect) is found the stronger the relation. This measure is calculated to minimize coincidental similarities, and to lower risk of promoting relations without causal relation. Simply speaking consequently found similarities are rewarded, and those which do not reoccur or occur rarely effect lower Clarity measures.

$$\kappa_{(s,o)} = \frac{S_{(s,o)}}{\sum \hat{s} + \sum \hat{o} - S_{(s,o)}} \tag{3}$$

The measure of Interference I (intx) has been introduced to limit the significance of high similarity in situations where more than one source signal matches with the same output signal being investigated. In such cases there is generally lower certainty of the causation strength, and therefore weaker dependency relation. We define interference as a sequence of cumulative similarity[2] of all source signals except currently considered k-th source $s^{(k)}$; the i-th element of interference vector $\mathbf{I}$ is computed according to the below equation:

$$\mathbf{I}_{(\mathbf{s_i^{(k)}}, \mathbf{o_i})} = \sum_{j:j \neq k} \mathbf{S}_{(s_i^{(j)}, o_i)} \tag{4}$$

[1] This is to avoid error multiplication problem, since derivative action is sensitive to measurement noise [38] [39]

[2] All vectors with similarity patterns between source signals are added up creating a cumulative sequence.

The scalar interference value $I_{(s,o)} = \sum_i \mathbf{I}_{(s_i,o_i)}$, where s_i, o_i are the i-th window of source and output vectors.

The Clarity C (clax) is the final measure of dependence of the proposed method. It represents the strength of impacts dependencies of source to output dimensions. It is based on similarity $\mathbf{S}$, its consequence κ and interference $\mathbf{I}$ of other source signals following below equation, for the i-th element of the vector:

$$\mathbf{C}_{(s_i,o_i)} = \frac{\mathbf{S}_{(s_i,o_i)} * \kappa_{(s,o)} c_\kappa}{\sum \mathbf{I}_{(s_i,o_i)} c_\mathbf{I} + 1} \tag{5}$$

The scalar clarity value $C_{(s,o)} = \sum_i \mathbf{C}_{(s_i,o_i)}$.The equation consists also Consequence c_κ and Interference $c_\mathbf{I}$ scaling coefficients, as additional constant parameters configuring strength of measures in Clarity calculation.

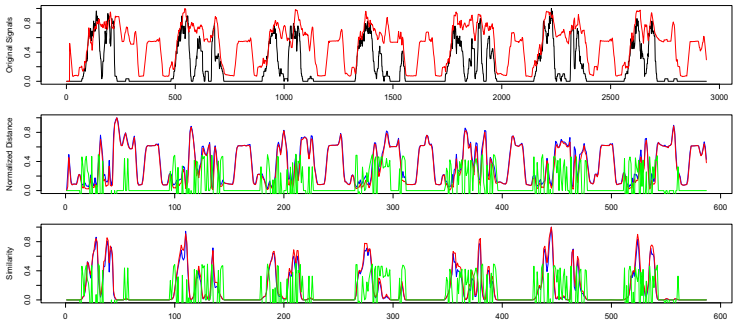


Fig. 2. Comparison of times series matching (top chart with input signals, middle with normalized distances, bottom with Similarity sequence) with use of three different distance measures Euclidean (red), DTW (blue), Spearman correlation (green) in a window of 5 points (seconds).

5 Testbed Design and Experiments

All empirical evaluations were done with use of R scripting, which can be directly called from Java hosted ASM controller. Several experiments were performed on datasets of synthetic data (5.1), and captured from the test-bed system (5.2) to test the effectiveness of the approach.

To comparatively evaluate the performance of the distance measures applied in SCIC, we performed a test of different sliding window length n, see Fig. 3; all charts contain normalized values adjusted by the size of the sliding window. The comparison shows that the ranking method is stable against sliding window length, and fairly similar across presented dimensions method.

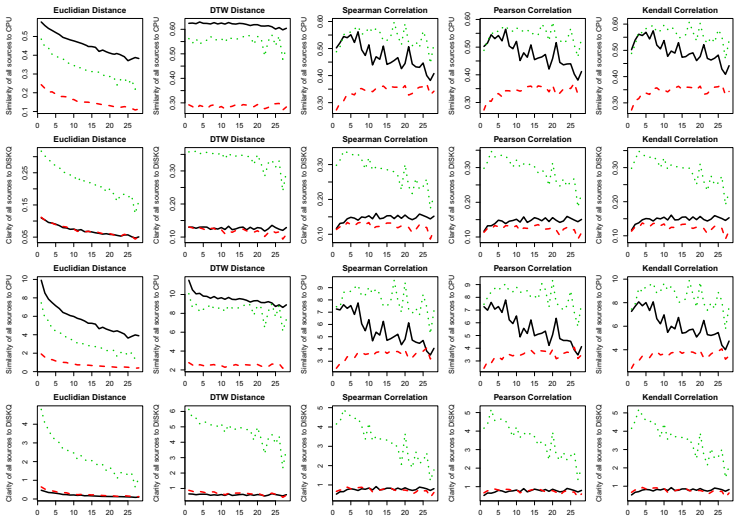


Fig. 3. Figure presents comparison of Similarity (first two rows) and Clarity (last two rows) of five feature selection tests using different distances measures (in columns). Each of the tests computed the dependency relation of actions to resources of data taken from Scenario 2 (5.2), where lines denote different departments actions: solid-black: Reservations, dashed-red: Operations, dotted-green: Finance. Tests were executing SCIC algorithm for different sliding window lengths, from 1 to 30.

5.1 Scenario 1

This example contains synthetic data sample of aggregated values collected during 24 hours of normal operation of hypothetical world-wide used system with 3 departments using different system actions. The first component is Reservations based in Europe and America – which are characterized by high use, and very frequent calls; Operations based in India – cause high system use, frequent calls, and long sessions. Financial Reporting based in America – rare actions but heavy IO/CPU tasks; see Fig. 4 for more details. Due to different geographic locations the load is distributed in time.

In this case action summary of execution time metrics have been considered, so data evaluated are aggregates in a given time bucket. To simplify the presentation only two resources are being analyzed CPU and DISKQ. As mentioned earlier the highest computational load comes around midday from Reservations (ACTRES), and later at the end of the day from Finance (ACTFIN) departments when also the disk was mainly utilized. Above highlighted actions-resources matchings were found, what was showed in similarity matrix on Fig. 4. Although Reservations (ACTRES) were substantially interfered by Operations (ACTOPS) activity, the Similarity and Consequence measures were firm enough to conclude the highest Clarity level across evaluated pairs. The Clarity matrix shows that the disk usage was found to be relevant mainly for Finance actions. The Dependency measure has not been applied in this scenario as there is no significant effect of delayed system resources signals nor SLA response.

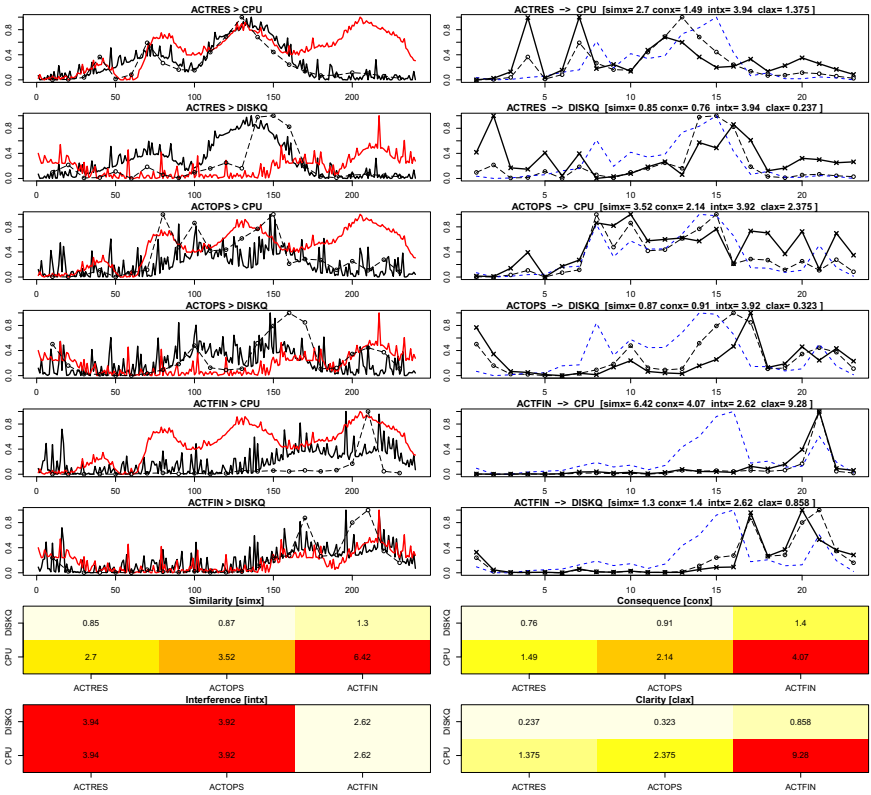


Fig. 4. Figure represents matching of Actions to Resources for 24-hours of operation of a hypothetical world-wide-used system. The bottom of the figure presents four matrices with Similarity, Consequence, Interference, and Clarity values for each of signals pairs evaluated. Left column shows input (black), output (red) signals, and Similarity sequence (dashed). Right column contains comparison of sequences of Similarity: dashed with o–points (where Spearman correlation coefficient as the distance method was used), Interference: dashed blue, and Clarity: solid–bold with x–points.

5.2 Scenario 2

In contrast to the earlier example this test dataset was captured during 48 minutes load test[3] containing 7 subsequent runs of a real testbed application, see Fig. 7. The causal model is defined using three tier approach: actions, resources, and SLA. There are five actions, four resource dimensions and three SLAs. Each of actions is called with the same pattern but with different load, start of the load or length of the load. The load test was prepared in such a way that only first action (ACTIONCNTSMA15) causes significant CPU and IO utilization. The rest of actions consume minimal resources. Nevertheless without the model of the system, input and output dimensions evaluation must be performed on normalized level as pure

[3] During the load test applied monitoring was capturing a detailed state of the system every second, thus there is a set of 2880 sample points.

actions quantity values do not provide information required to uncover particular actions impacts on resources. To hide the real use of the system from action perspective only action counts metrics are considered (actions execution time would give much better hint on the actual system usage). Raw metrics data were preprocessed with use of SMA of 15 sec window. Two of the actions were called exactly with the same pattern length as the first, but with shift of 100 and 200 seconds respectively, so they are overlapping each other. Another two actions load pattern was extended two and four times. These preparations effectively impacted the Interference measure.

Fig. 5 provides insight into the details of the Similarity-Dependency and the Interference-Clarity pairs. In order to illustrate better the dynamics of the SCIC all measures sequences have been normalized, so values on the charts do not reflect the real values, but the reader can compare the shapes easier.

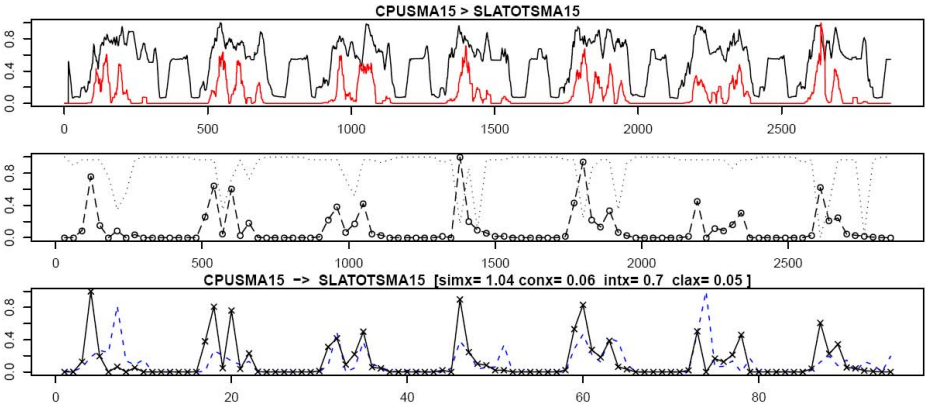


Fig. 5. Figure shows example of evaluated dimensions and sequences of SCIC measures calculated. From the top, input (CPUSMA15) and output (SLATOTSMA15 in red) signals, then Similarity: dashed with o–points with Dependency: dotted, and at the bottom Interference: dashed blue, and Clarity: solid–bold with x–points.

Fig. 6 presents the results of SCIC algorithm execution. Left column shows input (actions) and output (resources) signals. In the bottom of the figure four matrices represent values of main factors calculated for each of signal pairs evaluated. The highest similarity measure was found for ACTIONCNTSMA15 and CPUSMA15, but high values were also for CPUUSERSMA15. Consequence and Interference matrices confirm the strength of these relations. Please note that Interference matrix contains high values of ACTIONCNT1SMA15 and ACTIONCNT1BSMA15 which most often overlap with other entry signals, so even if similarities for these signals were high, there would be difficult to identify which ones of those had impact on the resources. Interference of ACTIONCNT1CSMA15 is the lowest, but Similarity and Consequence are too low to promote this entry signal as relevant for resources utilization. Despite the fact that the load test was obfuscated with other non relevant for resources consumption actions, the method evaluation was able to isolate enough information to set the final dependency measure (Clarity) in accordance with real workload.

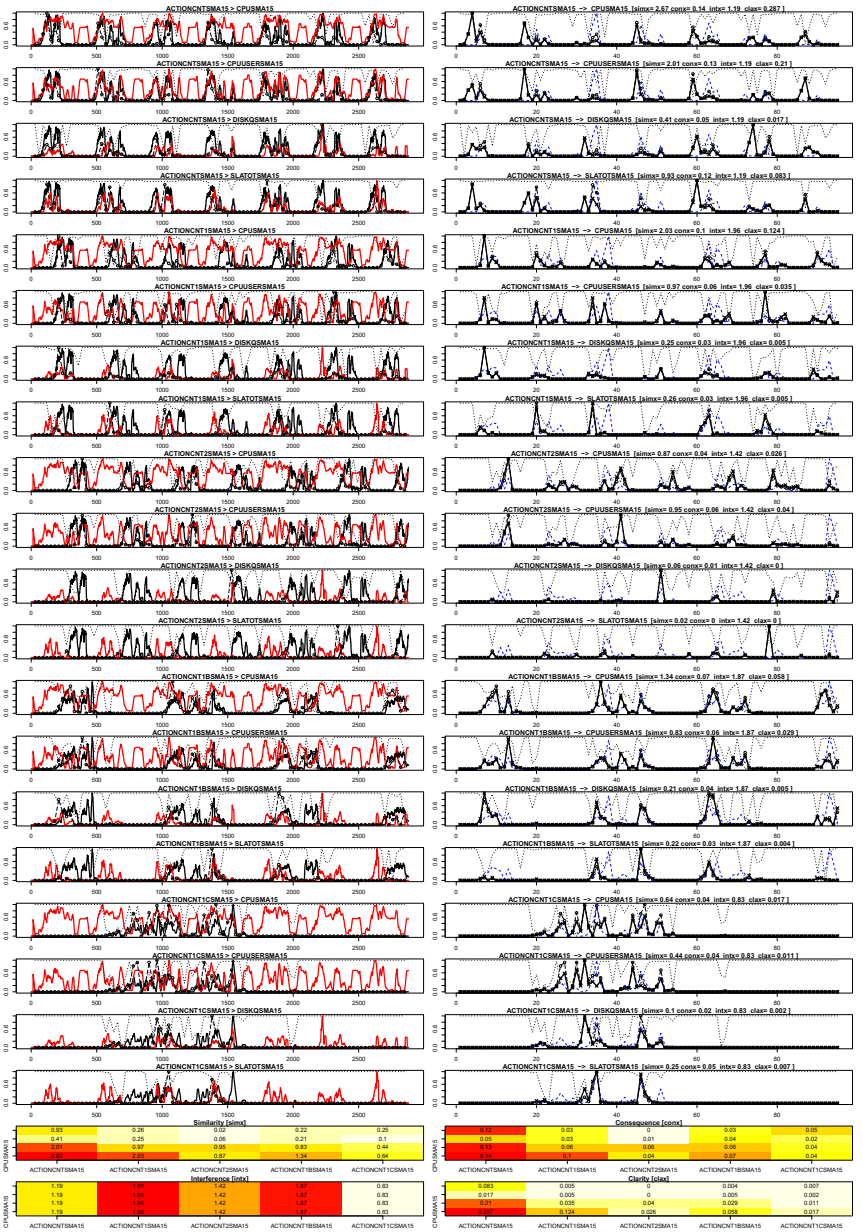


Fig. 6. The process of SDCIC-based matching of Actions to Resources. The bottom of the figure presents four matrices with Similarity, Consequence, Interference, and Clarity values for each of signals pairs evaluated. Left column shows input and output signals (black is input, red is output signal). Right column contains comparison of all measures normalized sequences; Similarity: dashed with o–points, Dependency: dotted, Interference: dashed blue, and Clarity: solid–bold with x–points.

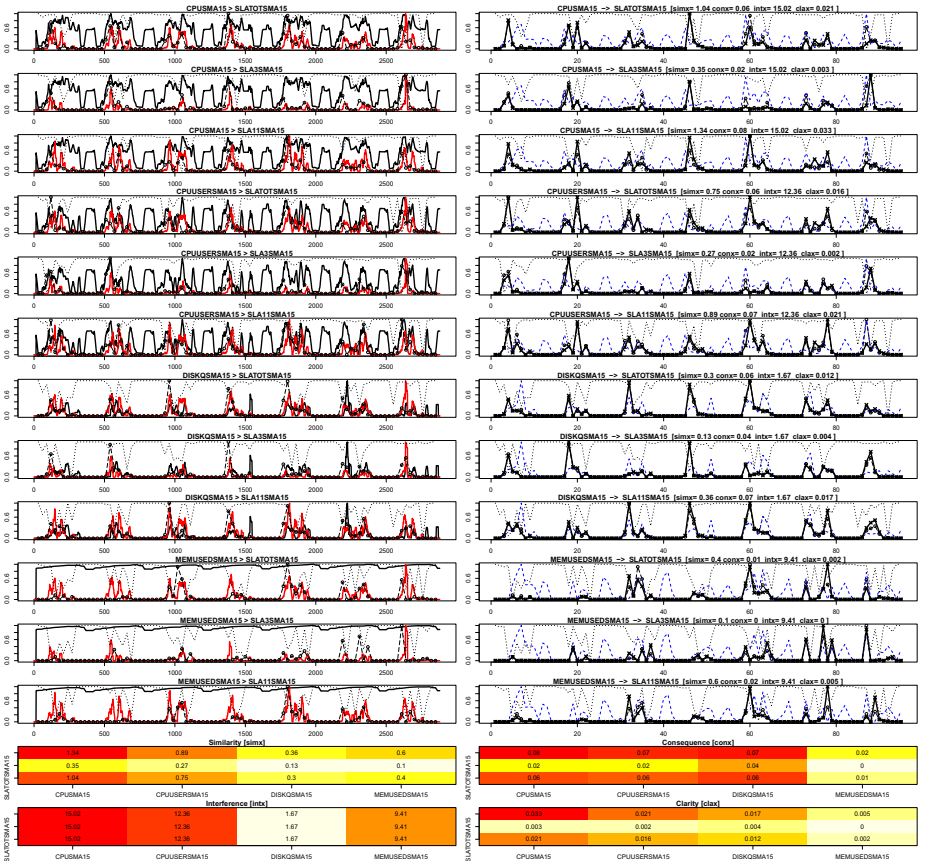


Fig. 7. The process of SDCIC-based matching of Resources to SLAs. This is an analogous figure to Fig. 6, all measures are denoted in the same way.

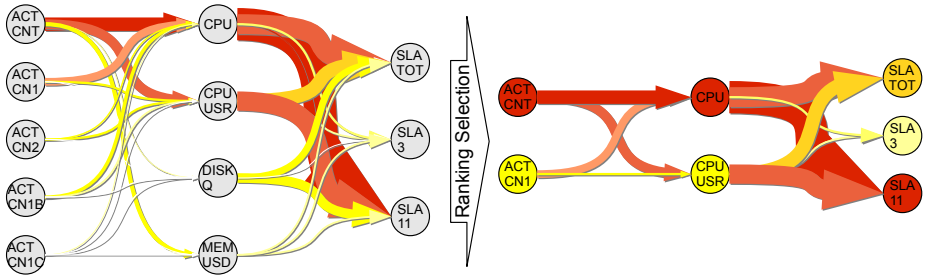


Fig. 8. Figure represents dependency graphs of dimensions used in the three tiers of causal chain. The graph on left shows all dimensions matched (see Fig. 6 and 7) before the future selection step. The graph on right consists only the strongest affected dimensions according to SDCIC method after ranking based selection.

Analogical processing was executed for the second tier of the dependency model Resources and SLAs, see Fig. 7. Please note the very low Clarity of MEMUSEDSMA15 pairs, mainly due to low Consequence. Notable is also the very low level of Interference of DISKQSMA15, which contains fairly rare high signals so it was not interfering with others, but had too low similarity with any other SLA dimension to set Clarity to high level.

On the basis of the SDCIC processes performed against defined tiers of causal chain (Actions to Resources, Fig. 6; and Resources to SLAs, Fig. 7) the casual dependency graph was generated, which is an alternative representation of the Clarity matrices. After selecting top of the ranked Clarity measures, the graph was reduced containing only the strongest dimensions in the causal chain, see Fig. 8.

6 Limitations and Future Work

The proposed approach can be sensitive in situations where many source signals are similar to each other (input dimension values change in time in similar way, resulting in high Similarity measure between them). Then due to the Inference measure definition such similarities are strongly minimized in the Clarity. If the system works only under such conditions then it would be difficult to infer which elements of the causal model are dependent on each other, and the model is inefficiently evaluated.

Moreover the method can fail to provide precise results in situations where: not enough run-time descriptive metrics are available; actions significantly change the system run-time characteristic after the execution, or data are collected for very short time; so uncertainty is high and the effective probabilities cannot be set or the causal model is misspecified.

The feature selection technique can be further improved by applying a preprocessing step of ASM signal separation, which is a crucial problem for non-isolated metrics space areas, where many actions use the same resource, and impact its utilization level metric. Blind or Semi Blind Signal Separation (BSS/SBSS) tackle a similar to defined problem in signal processing domain [40]. Equation (6) below, which is common to BSS techniques, describes hidden components in a resource signal r_k effected by actions a (actions and resources dimensions were used only as examples):

$$r_k = \sum_i p_i(r_k)a_i + n_k \ , \tag{6}$$

where r_k is the k-th resource, a_i is the i-th action p_i is unknown coefficient of component causing changes in r_k after a_i calls, n_k is noise.

Although this chapter does not propose a new method to blindly unmix ASM metrics data, it is worth to outline the main difficulties in ASM signals decomposition, which we plan to tackle in future work. The core obstacle to use out–of–shelf BSS is that the collected metrics sequences distribution can be statistically dependent – thus, many of BSS techniques from signal processing can not be applied directly.

Another issue is related to non-addable linearly signals, i.e when a given resource is highly or fully utilized, actions using it are queued by the system. During saturation the signals are distorted/deformed: in such situations resources are flattened and extended in time while actions execution time signals form much higher peaks. Thus the main assumption of most BSS methods presented in equation (6) is not met.

On the other hand all signals source data are stored in a database, so it is possible to start addressing such problems by building models of statistical properties, so derivation of hidden components can assess the distributions of signals being evaluated. Moreover, assuming that the cause-and-effect delay is known (in simpler cases it is equal to zero) particular hidden components can be matched using similarity measures for the unknown coefficient values searched. The ASM signals deconvolution should be also supported by the assumption that resources r are utilized by actions a, thus cumulative time of action execution is directly related to used resource until a given time.

7 Conclusions

Although ASM control is gaining more and more attention in the enterprise, there is still lack of well researched features selection methodology, which could be used to improve autonomous controllers and support ASM operators. To alleviate this situation, the chapter discusses two methods called SCIC and SDCIC. Causal dependencies in ASM field are known in precision to tier or type of metric (action count, action execution time, resources, SLA values). On that basis we are able to evaluate dependencies between dimensions by measuring similarities of input and output signals.

Methods discussed build sequences of Similarities, Interferences and Clarities values – therefore provides details of interpreted data against time, which can be helpful for understanding the relationships evaluated. In cases when only some actions are controllable $a_c \in a$, it is possible to find out which controllable actions cause most impact on resources and SLAs and include that in the ranking accordingly. This can be an interesting audit property of the method when applied in engineering field.

Additionally the introduction of the Dependency measure complements and enhances the Similarity measure, used in the original SCIC method, by applying further causality observations based on cross-correlation. The use of this measure leads to lower Clarity scores in situations when observed effects were delayed against the set causality chain. The evaluated scenarios confirm the practical potential of the proposed approach. Our research in the area of features selection for more focused autonomous control in the ASM field is ongoing. In particular we investigate various aspects relating to signal decomposition, unmixing hidden components, and incorporating knowledge about the system reactions delays, which were highlighted in the chapter.

Acknowledgment. This work was partially supported by Solid Software Solutions (`http://www.solidsoftware.pl/`).

References

1. Haines, S.: Pro Java EE 5 performance management and optimization. Apress (2006)
2. Sydor, M.J.: APM Best Practices: Realizing Application Performance Management. Apress (2010)
3. Grinshpan, L.: Workload characterization and transaction profiling. In: Solving Enterprise Applications Performance Puzzles: Queuing Models to the Rescue, pp. 57–94 (2012)
4. Sikora, T., Magoulas, G.D.: Neural adaptive control in application service management environment. In: Jayne, C., Yue, S., Iliadis, L. (eds.) EANN 2012. CCIS, vol. 311, pp. 223–233. Springer, Heidelberg (2012)
5. Pérez, P.A., Antonio, S., Sala, A.: Multivariable control systems: an engineering approach. Springer (2004)
6. Pearl, J.: Causality: models, reasoning and inference, vol. 29. Cambridge Univ. Press (2000)
7. Guyon, I., Elisseeff, A.: An introduction to variable and feature selection. The Journal of Machine Learning Research 3, 1157–1182 (2003)
8. Ding, H., Trajcevski, G., Scheuermann, P., Wang, X., Keogh, E.: Querying and mining of time series data: experimental comparison of representations and distance measures. Proceedings of the VLDB Endowment 1(2), 1542–1552 (2008)
9. Box, G.E., Luceno, A., del Carmen Paniagua-Quiñones, M.: Statistical control by monitoring and adjustment, vol. 898. Wiley (2011)
10. Box, G.E., Jenkins, G.M., Reinsel, G.C.: Time series analysis: forecasting and control, vol. 734. Wiley (2011)
11. Holland, P.W.: Statistics and causal inference. Journal of the American statistical Association 81(396), 945–960 (1986)
12. Wright, S.: The method of path coefficients. The Annals of Mathematical Statistics 5(3), 161–215 (1934)
13. Wright, S.: Correlation and causation. Journal of Agricultural Research 20(7), 557–585 (1921)
14. Granger, C.W.: Investigating causal relations by econometric models and cross-spectral methods. Econometrica: Journal of the Econometric Society, 424–438 (1969)
15. Chen, Y., Rangarajan, G., Feng, J., Ding, M.: Analyzing multiple nonlinear time series with extended granger causality. Physics Letters A 324(1), 26–35 (2004)
16. Eichler, M.: Granger causality and path diagrams for multivariate time series. Journal of Econometrics 137(2), 334–353 (2007)
17. Guyon, I., Elisseeff, A., Aliferis, C.: Causal feature selection (2007)
18. Ihler, A., Hutchins, J., Smyth, P.: Adaptive event detection with time-varying poisson processes. In: Proceedings of the 12th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 207–216. ACM (2006)
19. Hunter, J., McIntosh, N.: Knowledge-based event detection in complex time series data. In: Horn, W., Shahar, Y., Lindberg, G., Andreassen, S., Wyatt, J.C. (eds.) AIMDM 1999. LNCS (LNAI), vol. 1620, pp. 271–280. Springer, Heidelberg (1999)
20. Geng, X., Liu, T.-Y., Qin, T., Li, H.: Feature selection for ranking. In: Proceedings of the 30th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 407–414. ACM (2007)
21. Wei, H.-L., Billings, S.A.: Feature subset selection and ranking for data dimensionality reduction. IEEE Transactions on Pattern Analysis and Machine Intelligence 29(1), 162–166 (2007)

22. Seppala, C., Harris, T., Bacon, D.: Time series methods for dynamic analysis of multiple controlled variables. Journal of Process Control 12(2), 257–276 (2002)
23. Berndt, D., Clifford, J.: Using dynamic time warping to find patterns in time series. In: KDD Workshop, Seattle, WA, vol. 10(16), pp. 359–370 (1994)
24. Keogh, E.: Exact indexing of dynamic time warping. In: Proceedings of the 28th International Conference on Very Large Data Bases. VLDB Endowment, pp. 406–417 (2002)
25. Pavlidis, T., Horowitz, S.L.: Segmentation of plane curves. IEEE Transactions on Computers 100(8), 860–870 (1974)
26. Keogh, E.J., Pazzani, M.J.: Scaling up dynamic time warping for datamining applications. In: Proceedings of the Sixth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 285–289. ACM (2000)
27. Keogh, E., Chakrabarti, K., Pazzani, M., Mehrotra, S.: Locally adaptive dimensionality reduction for indexing large time series databases. ACM SIGMOD Record 30(2), 151–162 (2001)
28. Latecki, L.J., Megalooikonomou, V., Wang, Q., Lakämper, R., Ratanamahatana, C.A., Keogh, E.J.: Elastic partial matching of time series. In: Jorge, A.M., Torgo, L., Brazdil, P.B., Camacho, R., Gama, J. (eds.) PKDD 2005. LNCS (LNAI), vol. 3721, pp. 577–584. Springer, Heidelberg (2005)
29. Hall, M.A.: Correlation-based feature selection for machine learning. Ph.D. dissertation, The University of Waikato (1999)
30. Keogh, E., Chakrabarti, K., Pazzani, M., Mehrotra, S.: Dimensionality reduction for fast similarity search in large time series databases. Knowledge and Information Systems 3(3), 263–286 (2001)
31. Kantz, H., Schreiber, T.: Nonlinear time series analysis, vol. 7. Cambridge University Press (2003)
32. Gruschke, B., et al.: Integrated event management: Event correlation using dependency graphs. In: Proceedings of the 9th IFIP/IEEE International Workshop on Distributed Systems: Operations & Management (DSOM 1998), pp. 130–141 (1998)
33. Keller, A., Kar, G.: Determining service dependencies in distributed systems. In: IEEE International Conference on Communications, ICC 2001, vol. 7, pp. 2084–2088. IEEE (2001)
34. Agarwal, M.K., Sachindran, N., Gupta, M., Mann, V.: Fast extraction of adaptive change point based patterns for problem resolution in enterprise systems. In: Large Scale Management of Distributed Systems, pp. 161–172. Springer, Heidelberg (2006)
35. Jiang, G., Chen, H., Yoshihira, K., Saxena, A.: Ranking the importance of alerts for problem determination in large computer systems. Cluster Computing 14(3), 213–227 (2011)
36. Kıcıman, E.: Using statistical monitoring to detect failures in internet services. Ph.D. dissertation, Stanford University (2005)
37. Pyle, D.: Data preparation for data mining, vol. 1. Morgan Kaufmann (1999)
38. Reinsch, C.H.: Smoothing by spline functions. Numerische Mathematik 10(3), 177–183 (1967)
39. Wang, Y.: Smoothing spline models with correlated random errors. Journal of the American Statistical Association 93(441), 341–348 (1998)
40. Comon, P., Jutten, C.: Handbook of Blind Source Separation: Independent component analysis and applications. Academic Press (2010)

Polarized Score Distributions in Music Ratings and the Emergence of Popular Artists

Tianqi Cai[1], H.J. Cai[1,⋆],
Yuanyuan Zhang[2], Ke Huang[3], and Zhengquan Xu[4]

[1] International School of Software, Wuhan University, Wuhan, Hubei, China
[2] School of Software and Microelectronics, Peking University, Beijing, China
[3] School of Economics and Management, Wuhan University, Wuhan, Hubei, China
[4] State Key Laboratory of Information Engineering in Surveying, Wuhan University, Wuhan, Hubei, China
{cai.tianqi,hjcai}@whu.edu.cn,
vivizhyy@gmail.com, flowerhugo@126.com, xuzq@whu.edu.cn

Abstract. It is found that most users have a favored score based on the dataset provided by Yahoo!, which is used most frequently in music ratings. We introduce the polarization ratio, indicating the frequency of a user's most favored score since the rating distributions depict apparent polarizations: 66.99% of the total users have the polarization ratio larger than 0.5; while 80.37% of the total users' most favored score is either 90 or 0. Our statistical analysis shows that the rating score distributions are highly polarized for average raters as well as the so called heavy raters regardless of the total times a user rated. Our analysis of the ratings on four categories of music items, tracks, albums, artists and genres, reveals the emergence of popular artists, which signifies the importance of the introduction of a user's most favored score and its polarization ratio in quantitative descriptions of human behavior in music ratings.

Keywords: Polarized behaviors, Music rating, Distribution, KDD Cup.

1 Introduction

As one of the biggest music service providers, Yahoo! Music has amassed billions of user ratings for musical objects. To provide better service, offering a smart recommendation system and a personalized service is crucial, which requires knowing users' tastes according to their rating history.

Previous studies have been conducted to analyze the rating behavior. Tuomas[1] uses an empirical comparison of two common paradigms of emotion representation in music, opposes a multidimensional space to a set of basic emotions and generates a linear model conditionally with a better view when consisting of a certain range number of predictors. While Berns[2] uses the functional magnetic resonance imaging (fMRI) to elucidate the neural mechanisms, which are associated with social influence with regard to music, and states that the principal

⋆ Corresponding author.

L. Chen, S. Kapoor, and R. Bhatia (eds.), *Intelligent Systems for Science and Information,* Studies in Computational Intelligence 542,
DOI: 10.1007/978-3-319-04702-7_20,

mechanism whereby popularity ratings affect customer choice is through the anxiety generated by the mismatch between one's own preferences and other's.

Others also attempt to explain the skewed patterns or peaks in the rating figures deriving from users' behaviors. One idea is "users who rate more items tend to have considerably lower mean ratings" [3]. They called those who explore and rate tens of thousands of items "heavy raters", and assumed that the heavy raters tend to rate more items which do not match their own musical tastes and preferences[4]. However, those explanations failed to commentate on the peaks in high ratings, especially the remarkable peak in the score 90, which are reported earlier in[5].

In this paper, we first calculated and depicted a general scatter plot to get a whole picture of users' behaviors, and then took a close look into the polarized score distribution of those typical user groups in 0 and 90.

2 The Yahoo! Music Dataset and Zipf-like Distributions

Yahoo! Music has created a large scale dataset for the KDD Cup 2011 contest, releasing over 250 million ratings performed by over 1 million users. Those ratings are aiming at four types of musical items: tracks, albums, artists and genres. Concentrating on the users' tastes and behaviors, the dataset is considered as a whole firstly, giving us a general idea about the ratings. As the research moves on, it will be grouped according to the types to show some details.

The KDD Cup contest gave out two datasets (Track 1 and Track 2) with similar properties, whereas the Track 2 dataset omits the dates and times, and refers to a smaller user population. To be more precise, we choose the Train data in Track 1, which is much richer and larger than any others. Train 1 owns 252,800,275 ratings of 624,961 items rated by 1,000,990 users from year 1999 to 2010. Each user has at least 10 ratings. Rating scores are integers varying from 0 to 100; while another popular 1-to-5 star scale ratings are translated by the dominance of the peaks at 0, 30, 50, 70 and 90 in our paper, according to the KDD Cup.

All ratings are grouped in item types, and in users' ratings on item types and total ratings of item types in Figure 1. In Figure 1(a), the horizontal axis represents users' ratings on tracks, and the vertical axis stands for the number of users. In Figure 1(b), the horizontal axis represents total ratings of tracks, and the vertical axis stands for the number of tracks. Similarly, Figure 1 (c) and (d) are for albums, (e) and (f) are for artists, and (g) and (h) are for genres respectively. Particularly, Figure 1(i) describes the users' ratings on all items, and Figure 1(j) demonstrates total ratings of all items.

Distributions in Figure 1 display Zipf-like features. Dashed lines are used to show the slopes of each distribution. The slopes of Figure 1(a) - (j) are approximately -1.35, -1.67, -1.52, -1.22, -2.08, -1.19, -2.78, -1.85, -1.47 and -1.56, respectively. It is notices that Figure 1(i) and 1(j) depict almost perfect Zipf-like distributions.

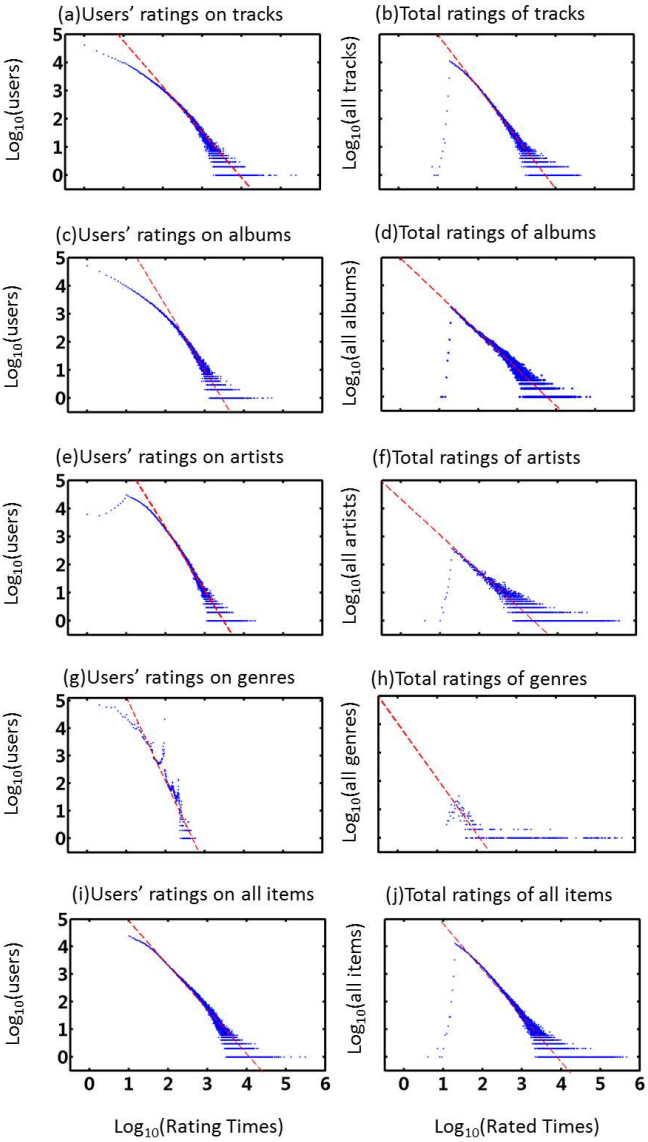


Fig. 1. Users' ratings on item types and total ratings of item types grouped in four item types, tracks, albums, artists and genres, and their summation, respectively.

3 Definitions of the Favored Score and the Polarization Ratio

A favorable music recommendation system is supposed to *know* which songs a user would like to listen to. To discover one's taste, understanding the user's behavior comes first. Thus, we put up two parameters to indicate two important values in a user's ratings.

One is the most favored score of each user, presented as the symbol S_{mf}. It denotes a specific score which occurs in most occasions in one's ratings. When Smf is a high score, we can figure out the user's favorable items by analyzing the objects; otherwise, we can infer the ones the user dislikes so that we will try to avoid them for the certain user.

The other is the ratio of occurrence of one's favorite score in one's whole ratings, or the frequency of S_{mf} We call it *polarization ratio* and mark it with the symbol α_p.

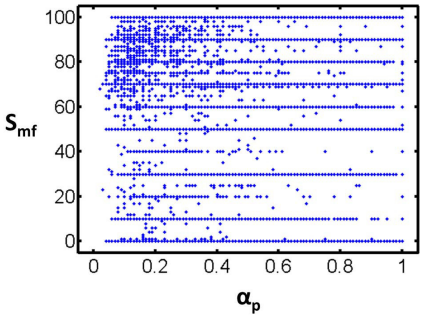


Fig. 2. The general view of the users' favorite scores

Table 1 describes some simple statistic results of user numbers with different pairs of S_{mf} and α_p. The columns are concerned with S_{mf}, while rows are with different α_p levels. Each block has three numbers, respectively indicating the number of users, ratio of the user group out of all users, and the average rating times within the group. Figure 2 depicts the user number of each group, where we can see the most remarkable peak in score 90 and the second peak in 0, forming an interesting polarized distribution, which signifies the importance of the introduction of S_{mf} and α_p.

Although, it is also clearly shown that 64.27% users take 90 as their favorite score, while the second popular score is 0 which concerns 16.10% users. Unlike the user ratios, the average ratings times per user with $S_{mf} = 0$ always have the biggest number, usually times more than that with $S_{mf} = 90$. Thus, $S_{mf} = 0$ is also a critical behavior trend in music rating, which strengthens the bipolar distribution in favorite rating score. Another interesting thing is, when $\alpha_p > 0.5$,

the total user number achieves 66.99% of all users, when $\alpha_p > 0.3$, it consists of 92.40% of all users, which depicts a strong preference in music rating among most people. Therefore, it comes no surprise to assume that most people have a certain partial score and feel like using it in most ratings.

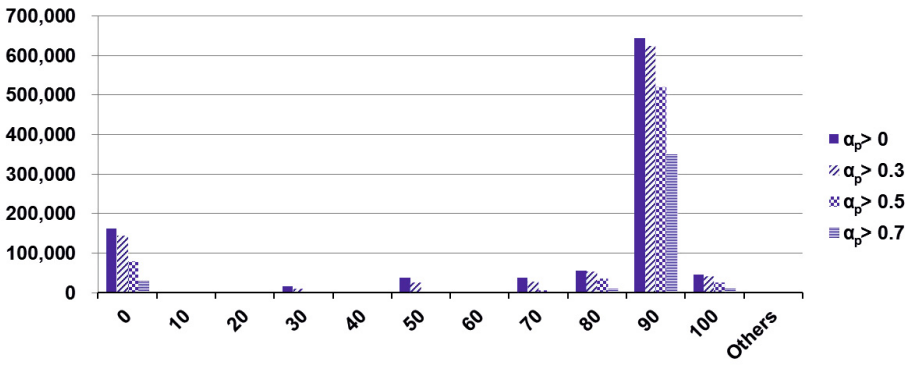


Fig. 3. The number of the users of each favorite rating score

4 Typical Polarized Rating Distributions

Since the users with favorite score of 0 and 90 occupy most places in both ratings and user numbers, it is natural to take a close look in those two groups.

Figure 4 displays the number of users and their average ratings, respecting different α_p values with a single unit equaling to 0.02. From previous analysis, we have known that, the distribution is polarized instead of normal. In particular, Figure 4(a) depicts that when $S_{mf} = 90$ and $\alpha_p > 0.98$, there exists an extraordinary sharp peak in number of users, however, the average rating times is the lowest at the meantime. As for $S_{mf} = 0$, shown in Figure 4(b), user numbers appear to follow an approximate normal distribution, while ratings are amassed at two ends.

It is inferred from Figure 4 that: (1) users in $S_{mf} = 90$ (whose most favored score is 90) have relatively less average rating times in the order of 100; (2) users in $S_{mf} = 0$ have higher average rating times in the order of 1,000.

Those findings can be applied in a recommendation system. Most customers tend to only rate on the items they like, so it is possible to find out their real interests quickly.

In order to get more details of polarization ratios, we have picked out several typical α_p values (0.7, 0.5, 0.4, 0.3 and 0.2) to observe the specific rating distributions, as displayed in Figure 4, so that the none-favorite ratings can be seen clearly. Apparently, there are always peaks at the score 0 and 90, regardless of variations in α_p.

Table 1. The number of users, their ratios and average rating times of each favorite rating score

S_{mf}	$\alpha_p > 0$	$\alpha_p > 0.3$	$\alpha_p > 0.5$	$\alpha_p > 0.7$
0	161206 16.10% 678.76	142831 14.27% 690.77	78941 7.89% 827.89	29503 2.95% 1103.13
10	2516 0.25% 176.11	2264 0.23% 82.07	1627 0.16% 36.66	526 0.05% 15.46
20	137 0.01% 662.72	45 0.00% 169.78	10 0.00% 22.5	3 0.00% 14
30	14966 1.50% 625.54	9952 0.99% 640.79	1385 0.14% 759.85	221 0.02% 612.27
40	109 0.01% 447.44	28 0.00% 561.64	11 0.00% 882	5 0.00% 10.6
50	37193 3.72% 557.11	23996 2.40% 531.79	2538 0.25% 427.1	296 0.03% 358.34
60	1503 0.15% 206.74	1163 0.12% 88.8	938 0.09% 28.71	662 0.07% 27.57
70	37233 3.72% 440.8	27104 2.71% 387.56	4461 0.45% 272.19	751 0.08% 224.74
80	55963 5.59% 55.17	51953 5.19% 36.69	35519 3.55% 30.52	8390 0.84% 23.71
90	643332 64.27% 124.21	624879 62.43% 116.09	520223 51.97% 89.27	350342 35.00% 70
100	45679 4.56% 278.67	40261 4.02% 254.64	24804 2.48% 221.35	11998 1.20% 183.51
Others	1153 0.12% 226.8	409 0.04% 195.1	124 0.01% 72.06	47 0.00% 137.04
Total	1000990 100% 252.55	924885 92.40% 230.73	670581 66.99% 181.67	402744 40.23% 148.76

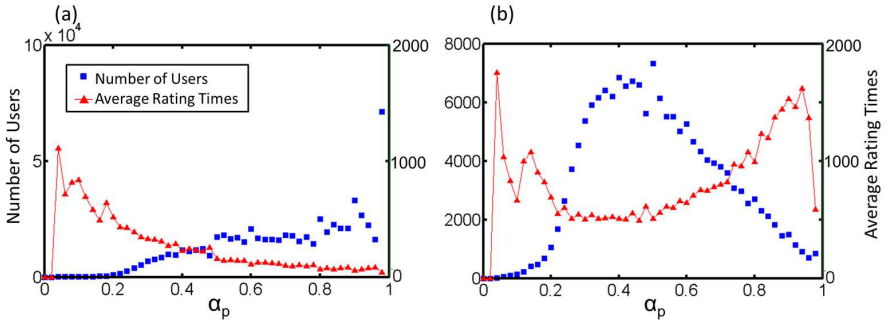


Fig. 4. Number of users and their average rating times of two groups with $S_{mf} = 90$ and $S_{mf} = 0$, respectively

However, when $\alpha_p < 0.5$, the score 50 becomes a new center peak and more peaks form around 0, 50 and 90, but we still regard such distribution as polarized for $\alpha_p > 0.5$, consists most users. Thus it leads to the polarizations in each ratio's Last, it is also important to point out that as 1-to-5 star scale ratings are translated in odd number (see section two), most peaks form in the odd numbers.

5 Polarized Ratings by Heavy Raters

Previous researches argue that, "users who rate more items tend to have considerably lower mean ratings" [1]. To test that, we picked out the top 20 heavy raters' statistic data and depicted the rating score distributions of the top 10, as shown in Table 2 and Figure 6 respectively. Obviously, scores are highly polarized in distribution.

In Table 2, we can see that, there are 6 out of the top 20 raters without 0 as their S_{mf}, and 2 0-favored users with the $\alpha_p < 0.4$. In particular, according to Figure 6, (a) and (d) have one polar, which can be explained by previous studies. However, (c) depicts that the 3rd heaviest rater only rated with fairly high scores. Figure 6(f) shows quite a lot of high scores as well. What is more, (e) and (h) show clear bipolar features.

6 The Emergence of Popular Artists

In previous sections, we have observed the polarized distributions in Yahoo! Music ratings. To take a closer look into those ratings, we group them into 4 categories: tracks, albums, artists and genres, and list the ratings of each category in Table 3.

We denote total ratings as R_t and frequency of favored/dominated score as α_p. Obviously, artists have a fairly high ratio, 62%, which is as twice as that of

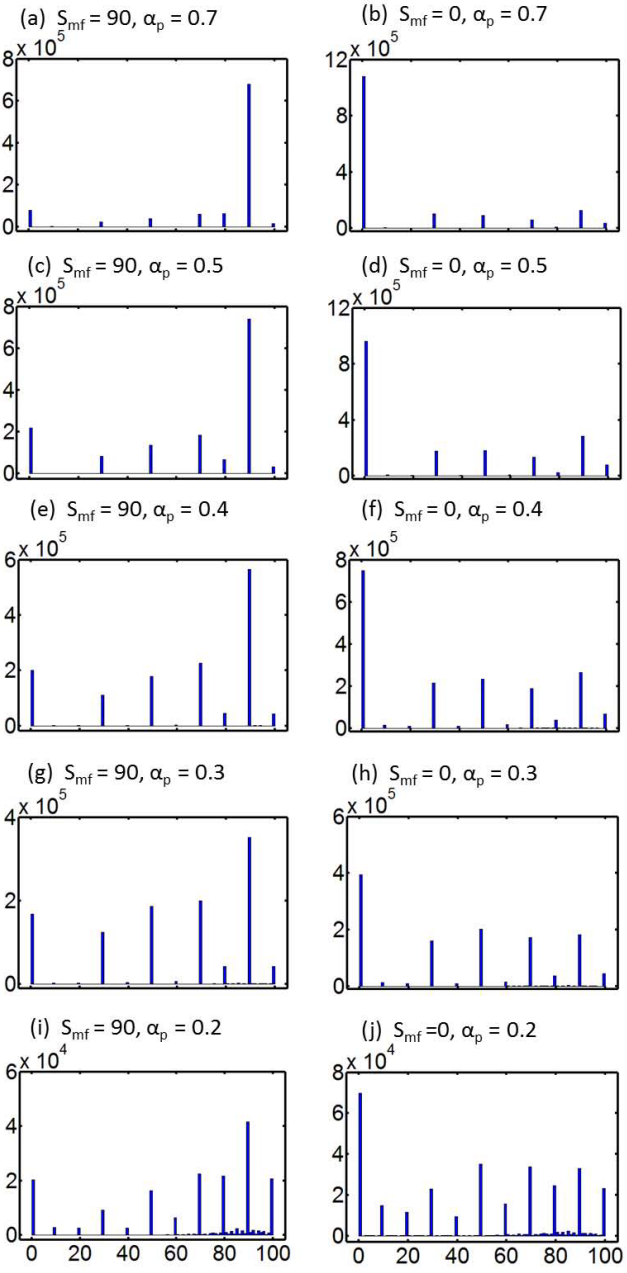


Fig. 5. Rating score distributions by users with specific favored score and polarization ratio

Table 2. Statistics of Top 20 Heavy Raters

User ID	Total Ratings	Mean Score	S_{mf}	Deviation	α_p
362170	307201	7	0	25	0.91
789893	224466	16	0	24	0.63
464661	132583	92	100	7	0.46
171695	89073	7	0	20	0.86
535879	74595	52	90	41	0.38
140482	74505	65	70	24	0.15
406338	66904	0	0	5	0.97
905572	65890	50	0	41	0.37
169188	64665	3	0	15	0.95
932163	64116	12	0	20	0.68
860621	59423	7	0	17	0.85
974216	57889	15	0	12	0.25
413054	55553	0	0	4	0.98
567135	53267	9	0	17	0.74
949460	52898	3	0	13	0.92
527571	52517	34	50	20	0.44
722623	51339	77	100	36	0.56
110226	50428	10	0	27	0.78
292379	46798	45	50	26	0.28
545603	46602	8	0	17	0.61

Table 3. Ratings of Each Music Type

	Quantity	R_t	α_p Quantity	> 0.5 Ratio
Track	507,172	118,439,743	159,491	31%
Album	88,909	48,060,315	31,642	36%
Artist	27,888	72,897,743	17,383	62%
Genre	992	13,402,474	539	54%
Total	624,961	252,800,275	209,455	34%

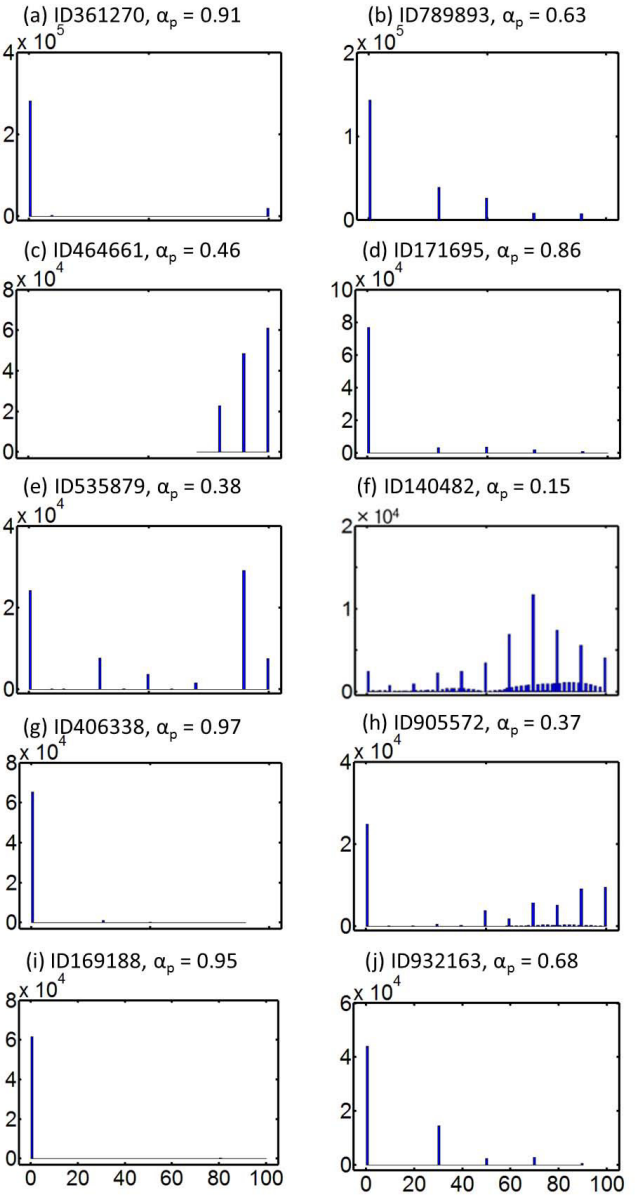


Fig. 6. Rating score distributions by top 10 heavy raters

tracks, when α_p is greater than 0.5. In other words, most artists are rated in a dominated score, which is quite different from tracks and albums.

With the discovery of most artists' high α_p, now we wonder whether those artists have huge amount of ratings or not. If so, they will definitely affect the whole distribution of artists.

Figure 7 depicts the R_t of each α_p of four categories. On the left side (Figure7(a)(c)(e)(g)) are the ratings from the user view, and each dot is a user's total times rating on the category with the user's dominated score of frequency α_p. On the right side (Figure7(b)(d)(f)(h)) are those from the item view, and each dot is total number of times an item in the category is rated with the dominated score of frequency α_p. In the user view, the dots are more concentrated on where p is greater than 0.5, while in item view, the dots are more concentrated on where α_p is less than 0.5. For example, Figure 7(a) has about 53% dots that are on where α_p bigger than 0.5 (323,071 out of 606,860 users), and 57% on the same area in Figure 7(c), 77% in Figure 7(e) and an amazing 89% in Figure 7(g). However, it is different in item view: Figure 7(b) and Figure 7(d) have 31% and 36% of dots, respectively, on where α_p is larger than 0.5.

The most remarkable thing is the amazing triangle area marked in red color in Figure 7(f). The triangle is on the right-top part of the image, indicating that those artists are rated for a huge amount of times and have a comparatively higher α_p. We call them popular artists. Those artists are so outstanding that we conduct a further analysis on them.

We pick 332 popular artists from the red area, the same number of artists from the left-top part of the image, and compare the two groups of artists. 332 popular artists are rated for 30,071,050 times (about 41% of all ratings of artists), including 18,094,025 times of favored score 90 with a mean α_p of 0.6. However, the 332 left-top artists only receive 9,810,738 ratings out of 72,897,743 and have a mean α_p of 0.3. Based on the two group artists, we extract the ratings of the 664 artists and draw out their distributions, displaying in Figure 8.

There is an obvious peak at score 90 in figure 8(a), which appearing 18,094,025 times, 60% of all popular artists' ratings and a second peak at score 0. While in figure 8(b), we can see a clear polarized distribution with a peak at score 0 and the second peak at score 90 respectively, which is consistent with the results in our earlier sections. It is noticed that if the two peaks of score 0 and score 90 are removed, the remaining scores' distributions of both case share the third peak at score 50, and their distributions look like normal.

The emergence of popularity occurs in the field of artists only, and this interesting phenomenon may have various causes. We believe that a most crucial factor derives from human nature of self-assertiveness demand[6]. For most people, they prefer healthy and attractive artists and give them high scores. Those scores are the bases of popularity. Once others notice the artists with more ratings, people tend to follow and support them as well, even without knowing much about the artists. In such a way, more and more attentions are gained which contribute to the emergence of the popu-larity of artists.

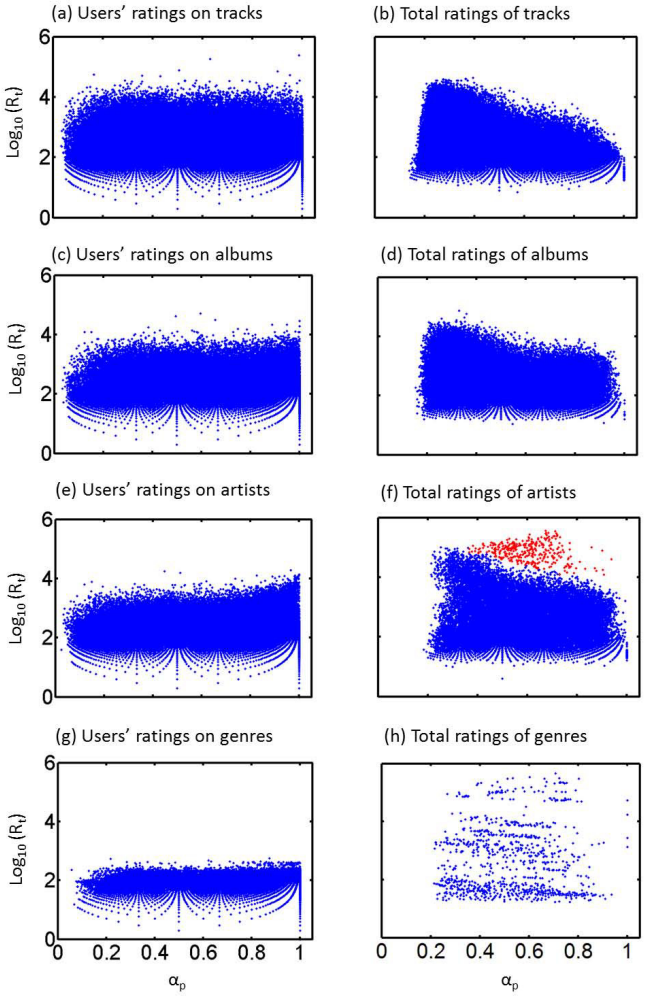


Fig. 7. Users' ratings on and total ratings of four types of music items respectively

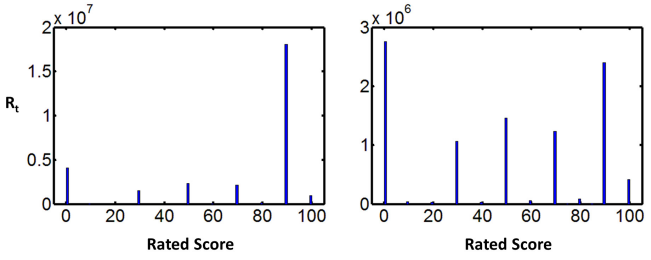


Fig. 8. Rated score distributions of popular artists and left-top group

7 Conclusion

In this paper, we have introduced two parameters: the most favored score S_{mf} and the polarization ratio α_p. Our analysis shows that the rating score distributions are highly polarized for average raters as well as heavy raters regardless of the total times a user rated. We also analyze the ratings on four categories (tracks, albums, artists and genres) of music items and find out the emergence of popular artists in Yahoo! Music rating systems. Without the introduction of S_{mf} and α_p, the emergence of popular artists may not be easily found out.

It is noticed that many users will rate on both the pleasant and the unpleasant music items and most of them tend to only score on those they are interested in and avoid rating too many times on those they do not like. Such phenomenon is true in almost every situation of S_{mf} and α_p, which provides a more thorough understanding of music rating behavior compared to a previous study tested through the data of Yahoo! Music's Launch Cast Radio, arguing that "most users tend to rate songs that they love more often than songs they feel neutral about and somewhat more often than songs that they hate" [7]). Such music rating behavior could be explained by Anchoring, which describes the common human tendency to rely too heavily on one trait or piece of information, while ignore other important information when making decisions[8]. Once most users start to rate in their beloved or dislike music items, they will simply follow their opinion on giving high rate for their favored music items or low rate for their hated ones, while pay little attention to other music items, due to such focus illusion.

Acknowlegements. This study is supported by the Basic Research Program of China/973 Project, Basic Research on Data Storage System Theory and Technology for Complex Application Environment under contract No. 2011CB302306.

References

1. Eerola, T., Lartillot, O., Toivainen, P.: Prediction of multidimensional emotional ratings in music from audio using multivariate regression models. In: 10th International Society for Music Information Retrieval Conference, pp. 621–626 (2009)
2. Gregory, S.B., Capr, C.M., Moore, S., Noussair, C.: Neural mechanisms of the influence of popularity on adolescent ratings of music. NeuroImage 49, 2687–2696 (2010)
3. Gideon, D., Noam, K., Yehuda, K., Markus, W.: The Yahoo! Music dataset and KDD-Cup 2011. In: KDD-Cup 2011 Workshop (2011)
4. Gideon, D., Noam, K., Yehuda, K.: Yahoo! Music recommendations: modeling music ratings with temporal dynamics and item taxonomy. In: ACM Recommender Systems 2011 (RecSys 2011), Chicago, Illinois, USA, ACM (2011)
5. Zhang, Y.: User clustering and rating score predictions with Yahoo! Music Datasets. Thesis, Wuhan University, Wuhan, China (2011)
6. Cai, H.J.: The Historical Context of the Rise of China and the Entry Point of the Transformation of the Development Pattern. Emergence and Transfer of Wealth 2(1), 1–6 (2012), doi:10.4236/etw.2012.21001
7. Marlin, B.M., Zemel, R.S., Roweis, S., Slaney, M.: Collabo-rative filtering and the missing at random assumption. In: 23rd Conference on Uncertainty in Artificial Intelligence, pp. 267–275 (2007)
8. Tversky, A., Kahneman, D.: Judgment under Uncertainty: Heuristics and Biases. Science. New Series. 185, vol. 4157, pp. 1124–1131 (1974)

Shape from Shading with and without Boundary Conditions

Lyes Abada and Saliha Aouat

Artificial Intelligence Laboratory (LRIA)
Computer science Department
University of sciences and technology(USTHB) Algiers, Algeria
{labada,saouat}@usthb.dz

Abstract. The shape from shading field attracts the attention of many researchers. Several methods have been proposed in several domains of computer vision. Two classes of methods are used: local and global resolution methods. Local methods deal with each pixel and its neighbors. Global methods, however, deal with all the pixels of the image at the same time. Other methods of resolution are integration methods which may be local or global. Integration methods, solve the problem of shape from shading throw two steps: the generation of the needle-map then its integration to generate the 3D object. This chapter proposes a new needle-Map integration method. The needle-Map is calculated from an image generated by a perspective camera. At first the boundary conditions was supposed to be known to solve the problem, then an improvement is performed to integrate the needle map without boundary conditions thanks to the utilization of a singular point of the image. The proposed technique was tested on synthetic and real images.

Keywords: Shape from Shading, needle-Map, perspective camera model, boundary conditions, singular points.

1 Introduction

The reconstruction of a 3D object from one gray level image is a difficult operation. This operation is called Shape From Shading. Several studies have been proposed to model the problem. Most of the researchers modelize the Shape From Shading by mathematical equations based on several field such as physical optics, photometry and other fields[13,3,6,9]. It is already proved that the Shape From Shading is an ill-posed problem, justifying it by the obtained equations. These equations give several solutions i.e. several objects can be generated from the same image. To make the unique possible solution , the image must verify some constraints on the object to be reconstructed. The constraints refer to The light source and the camera model with the knowledge of some initial data on the captured object[15]. Several classifications has been proposed on the methods of resolution. Denis [3,4] proposed a classification into three categories: Methods of Resolution of PDEs [15,16,17,18,19], Optimization Methods[7] and Methods Approximating the Image Irradiance Equation [3,11]

L. Chen, S. Kapoor, and R. Bhatia (eds.), *Intelligent Systems for Science and Information,* 369
Studies in Computational Intelligence 542,
DOI: 10.1007/978-3-319-04702-7_21,

Among the methods of solving Shape From Shading there are methods of integration, where the resolution is done in two steps: the generation of the needle-Map then the integration of the obtained needle-map to reconstruct the 3D object. In this chapter, a new method of integrating the normal field is proposed. Our method is based on the local resolution technique in which the orientation of the surface of each pixel depends on their neighbors. The normal field is assumed to be known.

The chapter is organized as follow:

In Section 2, we give an overview of Shape from Shading (SFS) principles and some techniques known in the literature. In Section 3 we will detail the proposed method for the integration of the normal field. In Section 4 we will show the steps of the reconstruction and the obtained results in the case of using the Boundary Conditions (BC) as the initial data. Then in Section 5, we will show the reconstruction steps using only one singular point without boundary conditions.

2 Shape from Shading

The first who introduced the notion of the Shape From Shading is Horn in 1970 [20]. The Shape From Shading (SFS) consists of reconstructing the 3D shape of the object from a single 2D image. This problem is ill-posed because there is an ambiguity in the generation of the 3D scene (an image can generate more than one 3D object). To make the problem well-posed most of the researchers put some hypotheses and constraints on the nature of the scene, the model of the camera (perspective or parallel), the position and the nature of the light source (punctual, located at the optical center or to infinity)...etc. The methods of resolution can be classified into three main categories[3]:

- Methods of resolution of partial differential equation(PDE): in this category the problem is modelized as a PDE for example Denis [5] designed the Shape From Shading by the perspective Eikonal equation, Prados[19] designed the SFS by PDE and proposed a solution using the theory of viscosity solution, among the methods in this category: Characteristic Strips Expansion, Power Series Expansion, Approximation of Viscosity Solutions, the Falcone and Sagona's Method, Level-Set Methods[3].
- Optimization methods: this category is based on the minimization of a certain error. Among these methods: the Choice of a Functional, Choice of an Energy, Choice of a Minimization Method, Daniel and Durou's Method [3].
- Approximation of the image Irradiance equation methods: There is local, linear and Tsai and Shah methods [3].

2.1 Image Formation

The camera receives light rays projected on the photosensitive. These rays transformed into electrical signals, then into a numerical image. Such operation is based on photometric and physical calculations. The SFS is the reverse operation

of the image formation most of resolution technique is based on the equations of the images formation.

Our method for integrating the normal field (needle-Map), as we will see in the next section, is based on some constraints:

- The model of the camera is perspective.
- The light source located to infinity.
- The surface is lambertian.
- The surface is continuous and differentiable.

There are several methods for generating normal field based on these constraints, among these methods Denis [3], Rosenfeld [12], Li Jin [11]...

The Li Jin method's [11] is the most used is the literature, therefore we apply it in our first approach for the generation of the normal field[1]. Li and Jin method requires the knowledge of the boundary conditions [8].

we will also present in this chapter our second approach that deals without boundary conditions by using a singular point of the image.

There are various methods proposed to represent and calculate the normal of a surface, Lee and Rosenfeld [12] Li Jin [11] represent a normal by the two angles TILT and SLANT. TILT is the angle between the projected of the normal vector on the image plan and the origin axis(OX). SLANT is the angle between the Z-axis and the normal vector. The majority of researchers in the Shape From Shading field start with the equations of the image formation in their approaches. Among these equations, there is the basic equation of the images formation for more details see the reference [2]

$$E = \frac{\pi}{4}(\frac{p}{f})^2 I \cos^4 \delta L \tag{1}$$

Equation 1 uses the internal parameter of the camera:
E : is the gray level of the image normalized between 0 and 1
δ : is the angle between the two vectors the light source and the z-axis unit vector.
p : is lens diameter.
f : is the focal distance.
I : is the intensity of the light source.

The luminance (brightness) of a Lambert surface is given by [2]:

$$L = \frac{\rho}{\pi}(\overrightarrow{N}.\overrightarrow{S}) \tag{2}$$

Such that L is the brightness of the pixel.
By replacing (eq. 2) in (eq. 1) the equation (eq. 1) becomes:

$$E = \frac{\rho}{4}(\frac{p}{f})^2 I \cos^4(\delta)(\overrightarrow{N}.\overrightarrow{S}) \tag{3}$$

Since the variables ρ, p, f and I are constant, we denote $K = \frac{\rho}{4}(\frac{p}{f})^2 I$, $cos^4(\delta) \simeq 1$:

$$E = K\overrightarrow{N}.\overrightarrow{S} \tag{4}$$

The two vectors N and S are unit vectors, so the scalar product of the two vectors is the co-sinus of the angle between the two vectors :

$$E = K\cos(\alpha) \tag{5}$$

The proposed method belongs to the category of "integration methods" which consists of reconstructing the Scene in two steps: the construction of the normal field (the needle-Map) then the determination of the scene from it.

2.2 Reconstruction of the Normal Field(Needle-Map)

In this work the normal field is considered known. The following formula is used to compute the normal field for the syntheses images.

$$N = \frac{-p, -q, 1}{\sqrt{p^2 + q^2 + 1}} \tag{6}$$

such as $p = Z_{i+1} - Z_i$ and $q = Z_{j+1} - Z_j$.

In the real test images, we used the method of generating the normal field proposed by Lee and Rosenfeld [12]. This method is used later by Li and Zhang [11] and many other researchers. The principle of this method is the generation of the needle-Map from the two angles TILT and SLANT. The first step is to calculate the SLANT. For a maximum gray level, the vector of light source S coincides with the normal then $\phi = 0$, the equation (eq. 7) can be rewritten as follow:

$$E_{max} = K\cos(0) = K \tag{7}$$

Under the constraint that the vector of the source is parallel to the z-axis, the angle between the normal vector and the source is equal to SLANT.

$$\phi = arccos(\frac{E}{E_{max}}) \tag{8}$$

The equation (eq. 2) can be written using the gradients of the normal vector $N = (p, q, -1)$ and the light source vector $S = (p_s, q_s, -1)$:

$$E = K\frac{pp_s + qq_s + 1}{\sqrt{p^2 + q^2 + 1}\sqrt{{p_s}^2 + {q_s}^2 + 1}} \tag{9}$$

p : is the gradient of normal(N) with respect X-axis.
q : is the gradient of normal(N) with respect Y-axis.
p_s : is the gradient of the light source S with respect to X-axis.
q_s : is the gradient of the light source S with respect to Y-axis.

by replacing :

$$p = tan(\phi)cos(\theta)$$

$$q = tan(\phi)sin(\theta)$$

$$p_s = tan(\phi_s)cos(\theta_s)$$

$$q_s = tan(\phi_s)sin(\theta_s)$$

The TILT can be computed [11] as:

$$\theta = arctan(\frac{E_y cos\theta_s - I_s sin\theta_s}{E_x sin\theta_s cos\phi_s + E_y cos\theta_s sin\phi_s}) \tag{10}$$

E_x : is the partial differential of E with respect X-axis.
E_y : is the partial differential of E with respect Y-axis.

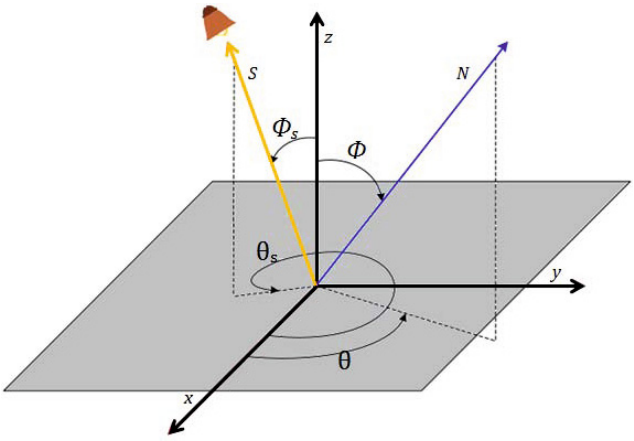


Fig. 1. TILT and SLANT of normal and source light vectors

The normal can be calculated from TILT and SLANT as follows:

$$N \begin{cases} N_x = sin(\phi)cos(\theta) \\ N_y = sin(\phi)sin(\theta) \\ N_z = cos(\phi) \end{cases}$$

3 Modeling of the 3D Scene and Integration of the Normal Field

There are several methods for integrating the normal field to generate a 3D surface. It is easier to generate the surface in the case of the parallel projection model, because the distance between the projection of two neighboring pixels on the scene is constant (Fig. 2.a). In this case the only information we have to know is the depth (Z). There are several methods for direct integration (with a single pass). But in the perspective projection model, there are two unknown parameters: the depth and the distance between two points (Fig. 2.b).

Denis [2] gives details of the difference between the two cases, and proposed an iterative solution for the integration of the normal field in the case of perspective model.

The main issue in Shape from Shading techniques is due essentially to the important processing time because the majority of techniques are iterative, therefore they are time-consuming.

in this chapter, we try to avoid the problem of previous methods. In the following, we give and detail our method which is a non-iterative method of integration of the normal field (needle-Map) in the perspective model.

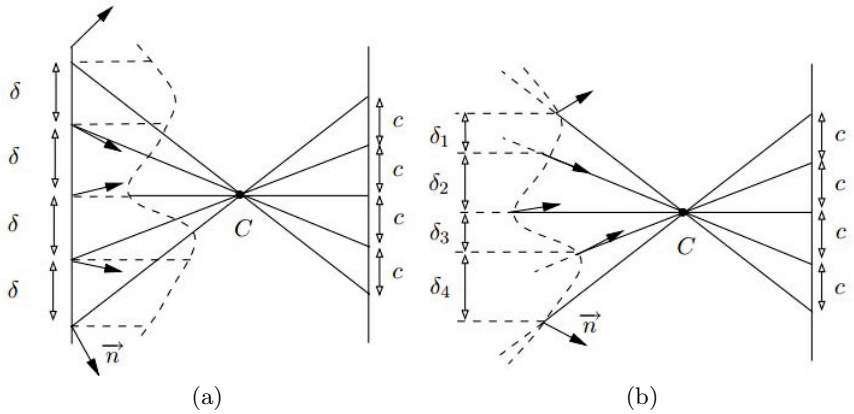


Fig. 2. (a)parallel model. (b)Perspective model [2]

The image is composed by a set of pixels, each pixel represents the projection of a small area of the scene on the photosensitive of the camera. In our method each pixel represents a small surface plan defined by four points around the pixel (px1 (top), px2 (bottom), py1 (right) and py2 (left)). the four points define a lozenge (Fig. 3), the projection of (px1,px2,py1,py2) on the scene with respect the optical center is (PX1, PX2, PY1, PY2) respectively.

The four points are calculated as follows:

$$P(i,j)\begin{cases} p_{x1}(i) = (i \times dx + (dx/2), j \times dy, f) \\ p_{x2}(i) = (i \times dx - (dx/2), j \times dy, f) \\ p_{y1}(j) = (i \times dx, j \times dy + (dy/2), f) \\ p_{y2}(j) = (i \times dx, j \times dy - (dy/2), f) \end{cases}$$

i, j are the pixel indices.
dx, dy represents the size of the pixel .

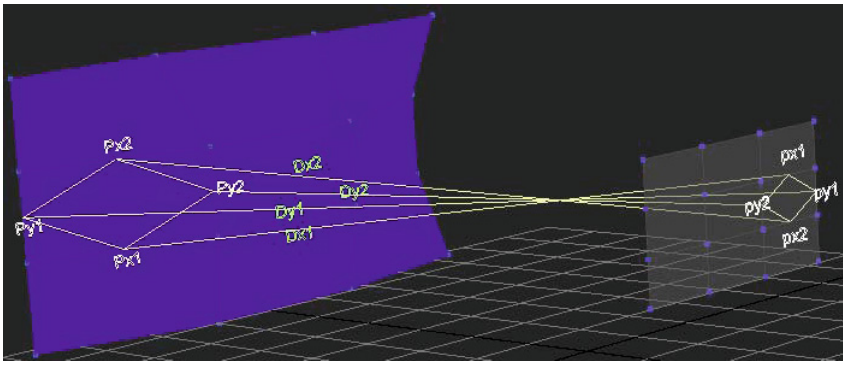


Fig. 3. representation of the surface

The projection lines of px1, px2, py1, py2 correspond to the perspective lines. They are noted (DX1, DX2, DY1 and DY2). The general form of a parametric equation of a line is:

$$P(i,j)\begin{cases} x = a \times t + x_0 \\ y = b \times t + y_0 \\ z = c \times t + z_0 \end{cases}$$

The lines (Dx1, Dx2, Dy1, Dy2) pass by the points (px1,px2,py1,py2) (respectively) and the optical center(0, 0, 0)

The equation of the line which passes by $(p_{x1} = (P_{x1}x, P_{x1}y, P_{x1}z))$ and the optical center(Dx) is:

$$P(i,j)\begin{cases} x = P_{x1}x \times t \\ y = P_{x1}x \times t \\ z = P_{x1}x \times t \end{cases}$$

The other perspective lines (Dx2,Dy1,Dy2) are computed in the same manner.

The aim of this method is to compute the four points (PX1, PX2, PY1, PY2) to generate the lozenge projection onto the scene. The equation of each surface plan (lozenge) is of the form:

$ax + by + cz + d = 0$.

The reconstruction of the scene is done by the orientation of each surface plan. The orientation is defined by the normal vector of the surface plan and an adjacent point (creation of a plan from a vector and a point). Initially we assume that we know at some initial data (like boundary conditions or a singular point), the intersection of the surface plan with the perspective lines generates the four points in the scene (see Figure 4)

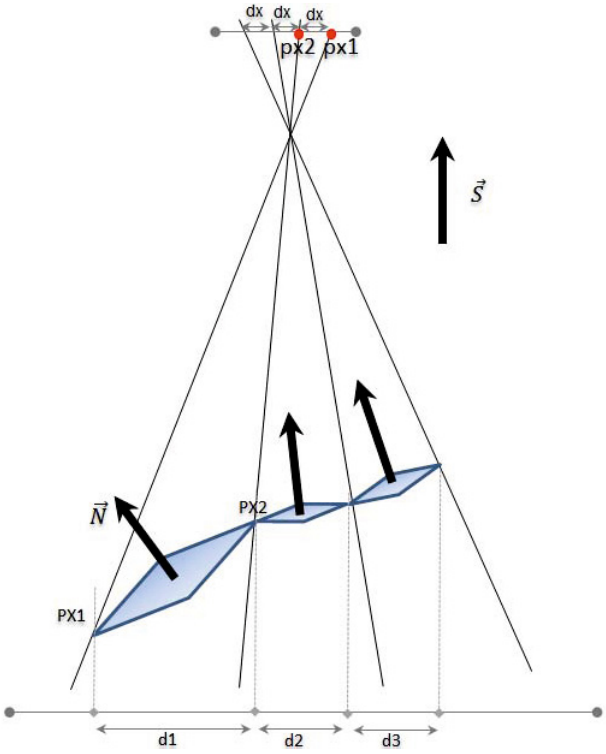


Fig. 4. Determination of the lozenges in the 3D scene

The Scene becomes a set of surface plans (3D lozenges) oriented by the normal vectors and delimited by four perspective lines (Dx1, Dx2, Dy1, Dy2).

Figure 5 shows the result of applying the proposed method on an image of a silt. The projection of each pixel on the scene is represented by a little lozenge.

To generate the 3D shape of the scene, we will calculate for each pixel the corresponding lozenge using the equations of the previous sections. Algorithm 1 is proposed to compute the lozenges. The complexity of this algorithm is equal to the time t_l (to calculate a single lozenge) multiplied by the number of lozenges

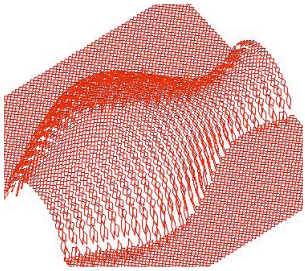


Fig. 5. representation of the surface by a set of lozenges

(n x m). In the case of a square image of size (n x n), the complexity of the worst case is

$$c = t_l * n * n = o(n^2) \tag{11}$$

The graph in the Figure 6 represents the processing time of the 3D reconstruction applied to the normal fields of Figure 9.b with respect to the size n (square matrix n x n)

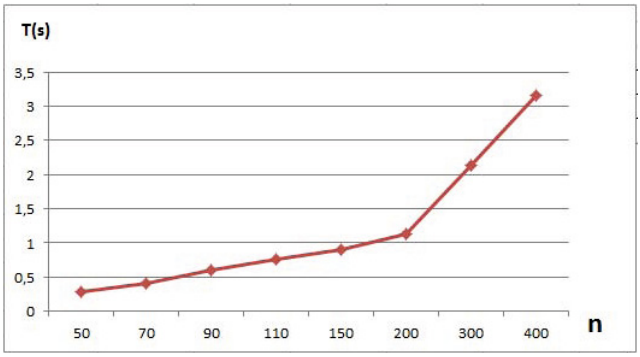


Fig. 6. processing time with respect to size n

The application of the proposed method requires initial data.

4 Reconstruction Using Boundary Conditions

The boundary conditions (BC) is the information on the boundary around the object which we want to reconstruct. Among the most used BC there is Dirichlet BC in which the information used is the depth (Z) of each point of the boundary [19]. There is also the Neumann BC that used the gradient of each point of the storyteller of the object. In this section we will use the Neumann BC because we need the normal of the BC (the normal can be computed from the gradient of the depth).

4.1 Reconstruction Steps

Algorithm 1 shows the different steps of reconstruction using the Boundary Conditions.

Algorithm 1. The needle-Map integration using Boundary Conditions

Initialization:

- The list of pixels already computed (**LCLOSE**) are initially the pixels of the boundary conditions.
- The list of pixels to compute (**LOPEN**) contains the pixels having one or more neighbors that belong to (**LCLOSE**).
- **LOPEN** is sorted from the pixel that contains the largest number of neighbors to the pixel that contains the smallest number of neighbors. This sort helps us to reduce the problem of non-integrability of the solution.

Lozenge Determination:

while (**LOPEN** is not empty) **do**
  P = the first element of **LOPEN**
  Compute the surface plan from the normal and the neighbors of **P**.
  Generate the points of the intersection between the surface plan with the perspective lines of the pixel P and then compute the lozenge.
  for each **PVi** neighbor of **P do**
    if (**PVi** is not in ***LOPEN*** and **LCLOSE**) **then**
      add **PVi** in **LOPEN**
    end if
  end for
end while

4.2 Experiments

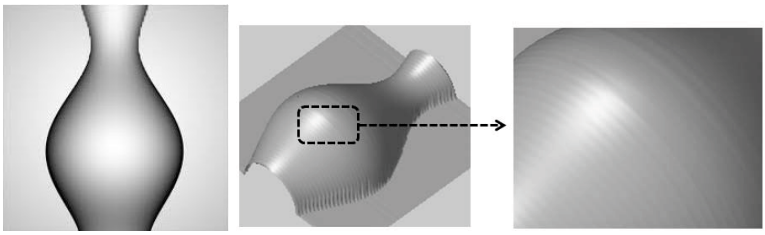
In this section we provide some experimental evaluation of the new integration method. We apply our method on some function and synthetic images. Some of them are generated by mathematics functions [1]

In order to test our approach, some treatment is applied to the image before generating the normal field:

First the boundaries of the scene are defined by a mask. It is essential to determine the area of the image that contains the object to be reconstructed.

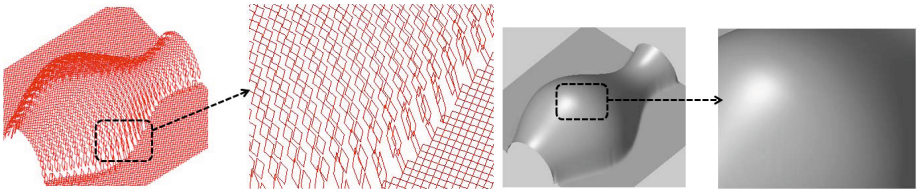
Then smoothing operations and the normalization must be applied to the image (we use the Gaussian filter in the tests). The image must verify the equation (eq. 4). In the experiments we take K equal to 1, so the gray level of the image is normalized between 0 and 1.

It is difficult to know the boundary conditions of the surface. During the tests, we assume that the background (the mask) of the image is perpendicular



(a) (b)

Fig. 7. (a) A synthetic image of a silt. (b) 3D reconstruction without any smoothing



(a) (b)

Fig. 8. (a)The problem of non-integrability of the solution applied to (Fig. 7.a). (b)Results of the proposed method on the image of Figure 7.a with smoothing using the Gaussian filter

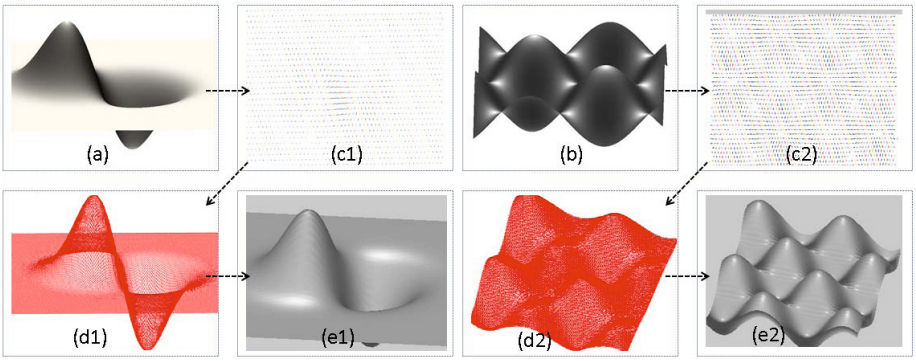


Fig. 9. Figure a (resp b) is generated using the function $f(x, y) = x*exp(-x^2-y^2)$ (resp $f(x, y) = sin(x)*sin(y)$). Figure c1 (resp c2) represents the normal field corresponding to figure a(resp b). Figure d1 (resp d2) is the result of the set of lozenges generated from c1 (resp c2). Figure e1 (resp e2) is the result of depth generated from d1 (resp d2).

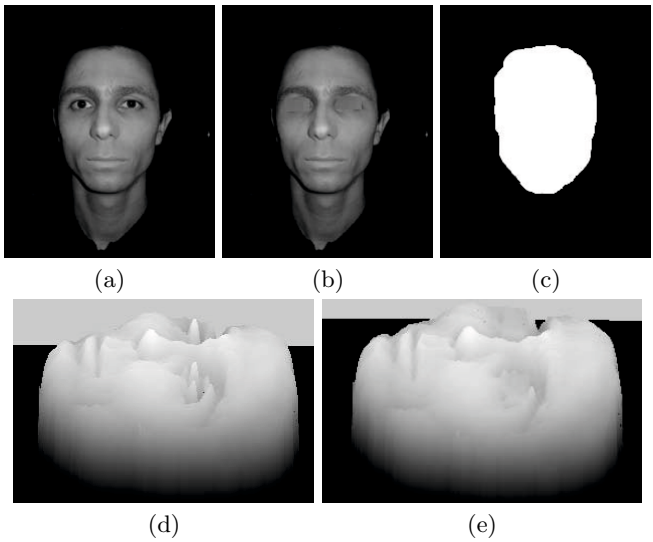


Fig. 10. (a) Real test facial image [21]. (b) real test facial image without the eyes. (c) the mask used to determine the area to construct. (d) Results of the proposed method on the image of Figure (a) using the mask of figure (c). (e) Results of the proposed method on the image of Figure (b) using the mask (c)

to the optical axis. From this supposition we can obtain the boundary conditions information.

Local resolution methods are very fast compared with global resolution methods because the majority of the global resolution methods are iterative. However the drawback of local resolution methods, is the non-integrability of the solution. Therefore we have proposed to solve this problem, an application of a smoothing process before the 3D reconstruction.

The synthetic image on which most researchers of Shape From Shading test their work is the image of a silt (see Figure 7.a). Figure 7.b shows the result of the silt constructed by the set of lozenges.

Figure 8.a shows the non-integrability of the solution. The lozenges are not stuck. Obtained results are not very good (see Fig. 7.b) but acceptable after image smoothing as shown in Figure 8.b.

We have also tested the proposed method on a real facial picture. The facial image does not check all constraints because the surface is not regular (for example eyes and brows). The reflection factor is not the same. Therefore the test on a facial image requires some modification. Figure 10.d shows the result given by the test on figure 10.a. There are problems in the eyes and eyebrows. Figure 10.b shows the face without eyes, the improved result of the application on this image is shown in Figure 10.e.

5 Reconstruction Using a Singular point

The majority of SFS methods are based on initial information about the object. In the previous section we used the Neumann boundary conditions for the 3D reconstruction of the object but this constraint is very limited. In order to facilitate the determination of the boundary conditions we assumed that the depth of all the pixels of the boundary is the same (the mask of the object is perpendicular to the optical axis). This constraint restricts the number of the used Images. Another disadvantage of the use of boundary conditions is the non-integrability of the solution. To solve these problems some improvements are proposed in the next section.

5.1 Reconstruction Steps

The reconstruction is done from one singular point (not many-points like the case of boundary conditions). Singular point are among the most important points in the 3D object. A singular point is a point of maximum gray level (the normal vector and the light source vector are parallel). The singular points can easily be determined by using a threshold. In our case we need only one singular point, so we will take the maximum gray level value of image.

The reconstruction is done following a circular movement around the singular point as shown in Figure 12.a. Lozenge 1 is a singular point. The numbers from 2 to 7 show the order of the reconstruction. The problem of the non-integrability of the solution appears in the case where we compute a new lozenge next to two neighbors already computed (Figure 12.c). To reduce this problem, the addition of a new lozenge is made such that the error is divided between all neighbors (the new lozenge absorbs the errors of the neighbors) like shown in Figure 12.b

Figure 13 shows an example of depth reconstructions using the method of a singular point. result are better than BC method and an example showing the improvement is illustrated in 14

The different steps of reconstruction using a singular point are mentioned in Algorithm 2.

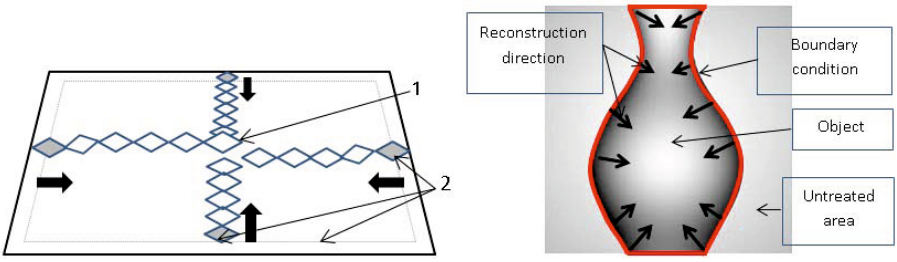


Fig. 11. direction of the reconstruction using boundary conditions

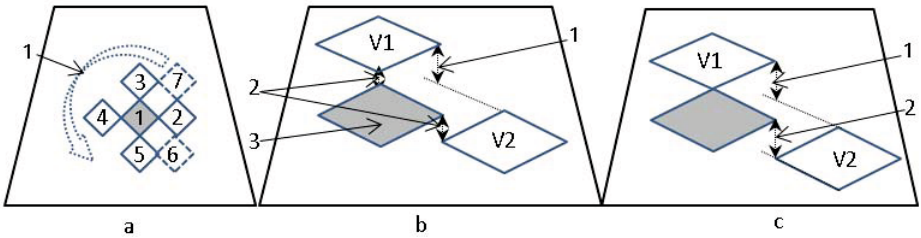


Fig. 12. (a) direction of reconstruction using a Singular point. (b)reduce the non-integrability problem. (c) non-integrability problem

Algorithm 2. The needle-Map integration using a Singular point

Initialization:

- The list of pixels already computed (**LCLOSE**) are Initialized by a pixel with a maximum gray level (singular point).
- The list of pixels to compute (**LOPEN**) contains the four adjacent points of the singular point with respect to Ox and Oy axis (left right top and bottom).
- **LOPEN** is sorted by FIFO method (first in first out) in order to start the reconstruction with the nearest pixels to the singular point.

Lozenge Determination:

while (**LOPEN** is not empty) **do**
  P = the first element of **LOPEN**
  Compute the surface plan from the normal and the neighbors of **P**.
  Generate the points of the intersection between the surface plan with the perspective lines of the pixel P and then compute the lozenge.
  for each **PVi** neighbor of **P do**
    if (**PVi** is not in *LOPEN* and **LCLOSE**) **then**
      add **PVi** in **LOPEN**
    end if
  end for
end while

5.2 Experiments

The advantage of the local resolution methods is the processing time but the disadvantage of these methods is on the error propagation. If there is an error in a pixel, this error influences on the other pixels during the reconstruction. The starting pixels of the reconstruction generate less errors than the pixels at the end of the process. In each image there is an important area. For example the important area of the image of Mozart (see figure 15.a) is the face. The beginning from the boundary pixels does not give good results with images of complex Boundary Conditions (15.a and 15.b) despite this area (BC) is not

important. If we start with a singular point, we can choose this point in the important area (on the nose of the face for example). For example Figure 15.c and 15.d show good results of Mozart and penny face despite the BC is not well reconstructed, Figures 14.a, 14.b 15.c, and 15.d are reconstructed by the proposed method without Boundary Conditions and without image smoothing.

Table 1. Comparison of the non-integrability of the solution between a reconstruction using Boundary Conditions and a Singular Point.

synthesis picture	f1	f2	Silt	Mozart	Penny
error BC method	0.0017	0.1153	0.1032	15.4782	39.1624
error Singular Point method	0.0040	0.0520	0.0493	0.1401	0.2428

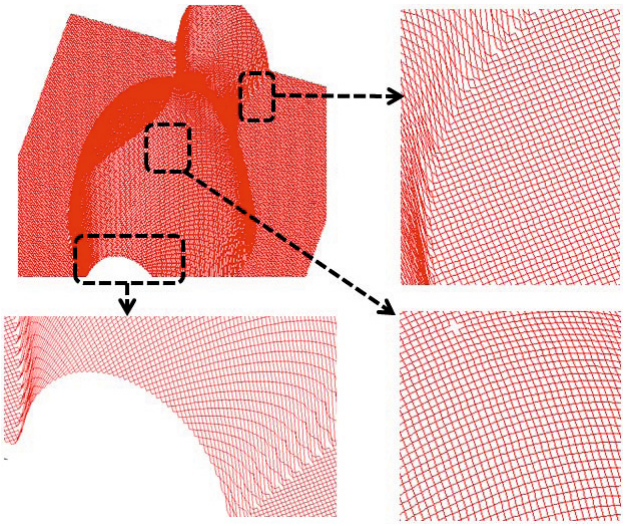
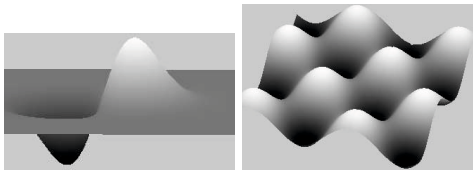


Fig. 13. Result of the set of lozenge generate from the figure (7.a) using a singular point



(a) (b)

Fig. 14. Figure (a) show the result of depth generated from needle-Map of figure 9.c1 and Figure (b) show the result of depth generated from needle-Map of figure 9.c2 using a singular point

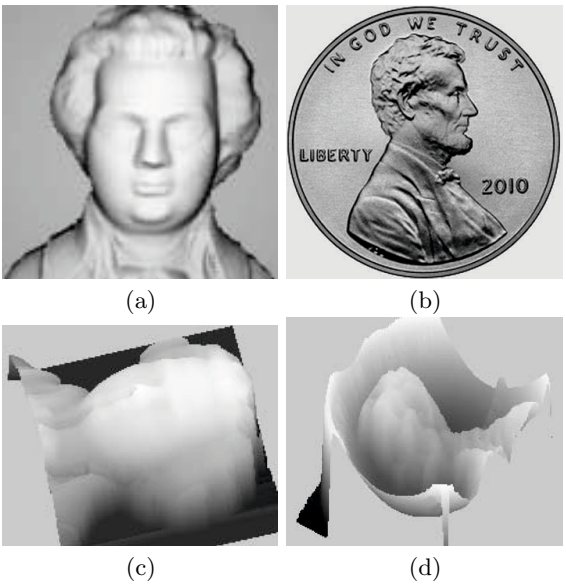


Fig. 15. Results (c) and (d) are generated from (a) and (b) using a singular point.

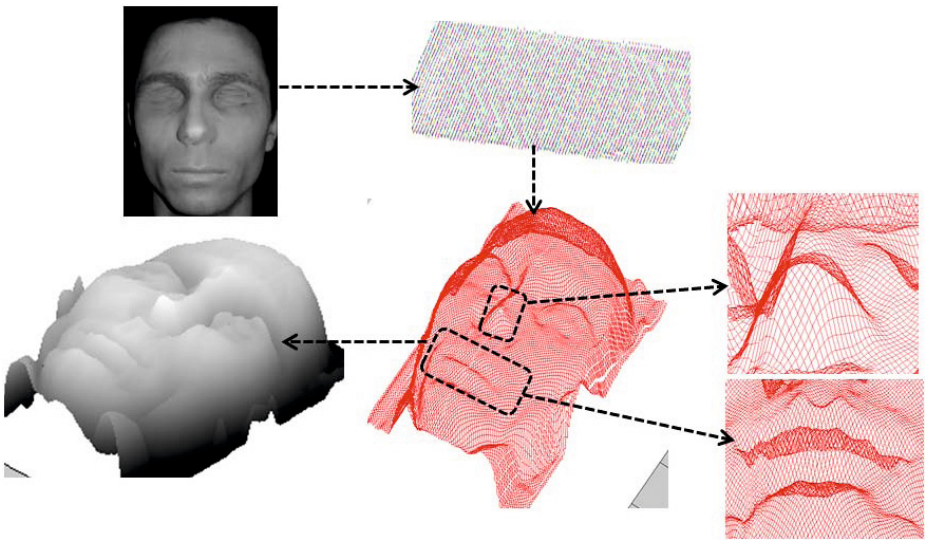


Fig. 16. The obtained results (the set of lozenge and 3D object) computed by the proposed method on the image of figure 10.b using a singular point, without BC and without mask

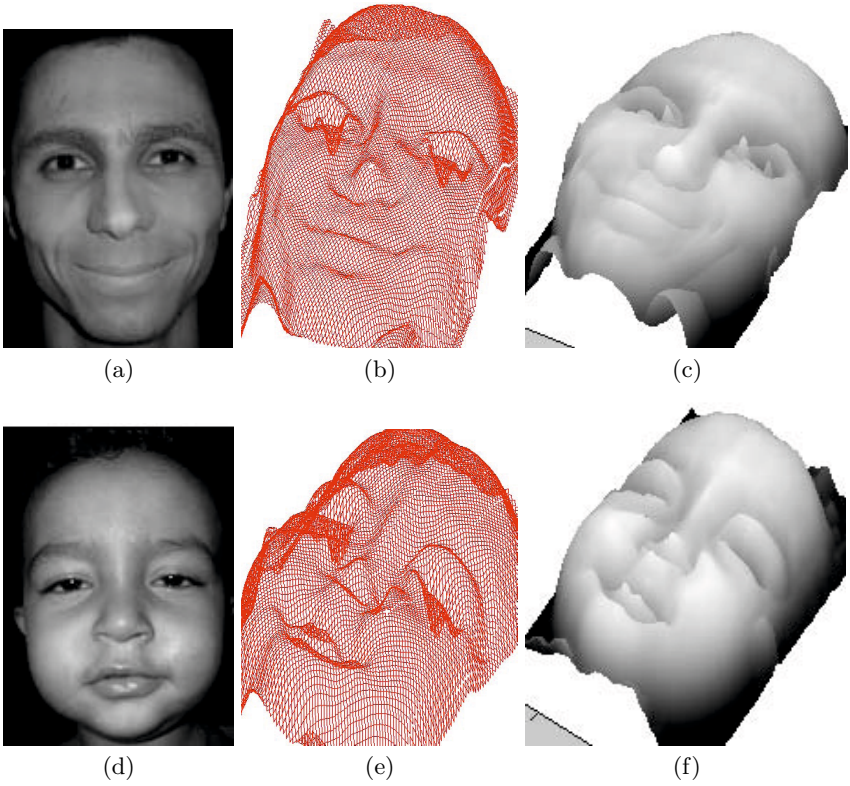


Fig. 17. Other results generated by the application on real Images

other examples on real image are given in figure 16 and 17. Obtained results are very interesting despite the non regularity of the images.

Table 1 shows the difference of the average error between the two approaches (BC and singular point). The error represents the displacement of a lozenge relative to its neighbors.

We can conclude that the second method is better because the problem of the non-integrability is less important.

6 Conclusion

In this chapter two non iterative methods of resolution based on the perspective model of the camera have been proposed. The first method uses boundary conditions. The second method is based on the utilization of a singular point of the image having the maximum gray level value. Both methods compute the 3D object by the intersection of a plan and perspective lines.

The use of a singular point as initial data facilitates the reconstruction and gives good results compared to the Boundary Conditions even with images that

have complex Boundary. We successfully applied the proposed method on some synthesis and simple real images. For the synthesis images we generated the needle-Map directly from the depth. However for real images we used the method of [11].

We will work later to reduce the number of constraints on which the techniques of Shape from Shading are often based on, and we will use more complex images.

References

1. Abada, L., Aouat, S.: Solving the perspective Shape From Shading problem Using a new integration method. In: IEEE Technically Co-Sponsored Science and Information Conference, SAI 2013, London, UK, October 7-9 (2013)
2. Courteille, F.: Prise en compte du modele stenope pour lextraction du relief en monovision. DEA Informatique de Image et du Langage (2002)
3. Denis Durou, J., Falcone, M., Sagona, M.: A Survey of Numerical Methods for Shape from Shading. Rapport de Recherche 2004-2-R, Institut de Recherche en Informatique. IRIT N 2004-2-R (January 2004)
4. Denis Durou, J., Falcone, M., Sagona, M.: Numerical methods for shape-from-shading: A new survey with benchmarks. Computer Vision and Image Understanding 109, 22–43 (2008)
5. Denis Durou, J.: Shape from shading Eclairages, rflexion et perspectives. Habilitation Searches, university of paul Sabatier toulouse (December 3, 2007)
6. Fan, W., Wang, K., Cayre, F., Zhang, X.: 3D lighting-based image forgery detection using shape-from-shading. In: 2012 Proceedings of the 20th European Signal Processing Conference (EUSIPCO), pp. 1777–1781 (August 2012)
7. Horn, B.K.P., Brooks, M.J.: The Variational Approach to Shape From Shading. Computer Vision, Graphics, and Image Processing 33(2): 174(208) (February 1986)
8. Huang, X., Gao, J., Wang, L., Yang, R.: Examplar-based Shape from Shading. In: Sixth International Conference on 3-D Digital Imaging and Modeling (2007)
9. Kunsberg, B., Zucker, S.W.: The differential geometry of shape from shading: Biology reveals curvature structure. In: 2012 IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), pp. 39–46 (June 2012)
10. Lee, K.M., Kuo, C.: Shape From Shading with a Linear Triangular Element Surface Model. IEEE Transactions on Pattern Analysis and Machine Intelligence 15, 815–822 (1993)
11. Li, J., Ren, A., Zhang, J.: 3D Reconstruction by Perspective Shape from Shading Using Linearized Triangular Element Surface Model. In: Proceedings of the 2006 IEEE International Conference on Mechatronics and Automation, pp. 1763–1768 (2006)
12. Li, J., Rosenfeld, A.: Improved methods of estimating shape from shading using the light source coordinate system. Artificial Intelligence 26(2), 125–143 (1985)
13. Ma, J., Zhao, P., Gong, B.: A shape-from-shading method based on surface reflectance component estimation. In: 2012 9th International Conference on Fuzzy Systems and Knowledge Discovery (FSKD), pp. 1690–1693 (May 2012)
14. Pentland, A.P.: Local Shading Analysis. IEEE Transactions on Pattern Analysis and Machine Intelligence PAMI-6(2), 170–187 (1982)

15. Prados, E., Faugeras, O.: Shape From Shading: a well-posed problem? In: Computer Vision and Pattern Recognition, CVPR 2005 (June 2005)
16. Prados, E., Faugeras, O.: Unifying approaches and removing unrealistic assumptions in Shape From Shading: Mathematics can help. In: Pajdla, T., Matas, J(G.) (eds.) ECCV 2004. LNCS, vol. 3024, pp. 141–154. Springer, Heidelberg (2004)
17. Prados, E., Faugeras, O.: Application of the theory of the viscosity solution to the Shape From Shading problem. Thse de doctorat, Universit de Nice-Sophia Antipolis (October 2004)
18. Prados, E., Faugeras, O.: A generic and provably convergent Shape-From-Shading Method for Orthographic and Pinhole Cameras. International Journal of Computer Vision 65(1/2), 97–125 (2005)
19. Prados, E., Faugeras, O.: Key aspect of the modeling in Shape From Shading. 15me Congrs de Reconnaissance des Formes et Intelligence Artificielle, INRIA 2005 (2005)
20. Zhang, R., Tsai, P.P., Cryer, J.E., Shah, M.: Shape from Shading: A Survey. IEEE Transactions on Pattern Analysis and Machine Intelligence 21(8) (August 1999)
21. INRIA DataBase, `ftp://ftp-sop.inria.fr/odyssee/Data/ShapeFromShading/realImages/`

Texture Segmentation and Matching Using LBP Operator and GLCM Matrix

Izem Hamouchene, Saliha Aouat, and Hadjer Lacheheb

Artificial Intelligence Laboratory (LRIA)
Computer science Department
University of sciences and technology(USTHB), Algiers, Algeria
{ihamouchene,saouat,hlacheheb}@usthb.dz

Abstract. Image processing is a dynamic research area. Recently, a lot of works have been made, efficient approaches have been developed and good results have been obtained. In this work, we propose a new texture matching and segmenting approach based on a new decomposing architecture. This method starts with one main window MW. For each iteration, the MW is reduced and all possible windows with the same size of the MW are generated. The Local Binary Pattern LBP operator, which is gray-scale invariant texture measure, and the Gray Level Co-occurrence Matrix (GLCM), which is a second order statistics measure, have been applied independently to extract the features from each generated window. Synthetic images and test images generated randomly from Brodatz album have been used in the experimentation. Good performances have been obtained and some results will be shown in the test section of this chapter.

Keywords: texture analysis, features extraction, texture segmentation, texture matching, LBP operator, decomposing architecture, GLCM matrix.

1 Introduction

Image processing is one of the main problems in computer vision. Since the last years, the automatic processing of images by their visual content have became an interesting and dynamic research area, such as indexing, segmentation and matching applications. It automatically processes and analyzes the image with its visual contents based on different features. These features can be regrouped into three groups: color, shape and texture. In this work, we are interested in texture features.

Texture image analysis is a fundamental problem in image processing and useful area in computer vision. Indeed, texture is present in most of objects and their surfaces are textured in real world. That makes it clearly an essential attribute in visual content based image analysis. Texture can be classified into different categories: coarse, micro, macro, regular, periodic, aperiodic, directional, random and stochastic [17]. Although, many researchers have widely

L. Chen, S. Kapoor, and R. Bhatia (eds.), *Intelligent Systems for Science and Information,* Studies in Computational Intelligence 542,
DOI: 10.1007/978-3-319-04702-7_22,

studied texture analysis and a lot of works have been published these last years [19]. Texture analysis keeps motivating the researchers particularly after the achieved success by different methods based on texture analyses.

There is not a strict and a clear definition of texture, but it can be considered as spatial repetition of a visual pattern in different directions in space, which consist of very small sub-patterns that have characteristics such as shape, color, size, luminosity The first works on texture image analysis were performed by Haralick [8]. Different approaches have been developed: statistical, structural, frequency domain (Gabor filter banks [2,11], wavelets transform). These approaches have been applied in a variety of application domains, especially in texture matching; such as printed documents, satellite and medical imagery. The goal of the matching texture is to recognize and to obtain the boundary map for one particular texture in an image that contains several different textures.

Each method belonging to the mentioned approaches requires the adjustment of its own parameters. This step is very important because the more parameters are adjusted, the better the results will be obtained. Almost all the methods proposed for texture matching are based on a classical fractional architecture, which consists of decomposing the image into 16x16 or 32x32 pixels or other fixed size blocks. After that, extract the features from each block and compare it with the researched texture feature. If the difference between a block and the sought texture (the researched texture) is greater than the threshold, this block is considered as relevant and added to the boundary map. It means that this block has the same researched texture. One weakness of this architecture is clearly the size of the decomposing blocks. Indeed, if the decomposing size is too important, one block may contain more than one texture. In the other case, if the decomposing size is too small, the researched texture could not be recognized. Almost all of the texture image analysis approaches use the decomposition architecture independently from the used feature extraction method. Therefore, this parameter (decomposition size) is clearly crucial and affects greatly the quality of the results. However, the choice of this parameter in texture analysis is an essential attribute, still a difficult problem to resolve and remains a research work.

In this study, we propose a new decomposing architecture for the texture matching task. This approach focuses essentially on the decomposition scale using a new image decomposing architecture. This architecture is mainly based on a dynamic decomposition size. In the features extraction step, any methods can be used, because the decomposition and the feature extraction steps can be used independently. To illustrate the interdependence between these two steps, we have used two different feature extraction methods. We have used the LBP (Local Binary Pattern) method, which is an adapted method for texture analysis and it is also invariant to rotation [16]. And for the second method, we have used the Gray Level Co-occurrence Matrix (GLCM) method [8]. These two methods are used to illustrate the proposed segmentation architecture process. Any other method for the feature extraction step can be used (the choice of the feature extraction method depends on the application domain).

The chapter is organized as follows: The next section illustrates the Local Binary Pattern (LBP) operator. Section 3 illustrates the Gray Level Co-occurrence Matrix (GLCM). In section 4, we present and explain in detail our decomposition architecture for texture matching and segmentation. Section 5 shows the application of the proposed method using different feature extraction method and circular shape of the analysis window. Section 6 gives the application of the proposed method. The experiments show the robustness and the good performances obtained by our method applied on many test images extracted from Brodatz album. The last section concludes this chapter.

2 The LBP Operator

The Local Binary Pattern LBP operator is a simple and powerful approach for texture analyses. It was mentioned by Harwood [9] and was introduced for the first time by Ojala and Pietikinen [15,14]. It has been widely exploited in various recent works [1,18,6,10]. Derived from a general definition of texture, LBP summarizes the local structure. It can be seen as a combination between the statistical and structural approach of texture analysis. The most important properties of the LBP operator are its monotonic gray-scale transformation and illumination invariance, rotation invariance [5,20] and its computational simplicity. This makes it possible to analyze images in very short time. These properties make it also attractive for many kinds of applications such as biomedical, face recognition, iris recognition, image and video retrieval

The original version of the LBP operator describes the texture by two measures: contrast and local spatial patterns. The contrast and local spatial measures are extracted by comparing each pixel with its circular eight neighbors in a 3x3 window. These measures are calculated using the value of the center pixel as a threshold on the eight neighbors around each pixel. A binary number (pattern) is obtained and converted to a decimal number (LBP code) to labels the pixels of the image. The contrast (Con) is the difference between the mean of higher neighbor values (which are greater than the value of the central pixel) and the mean of lower neighbor values (which are smaller than the value of the central pixel). The contrast is calculated by the Formula 1.

$$Con = \Sigma \frac{H_i}{N_i} - \Sigma \frac{L_j}{N_l} \tag{1}$$

Where H_i, L_j are the value of the i^{th} higher and the j^{th} lower neighbors respectively. N_h and N_l are the numbers of neighbors with higher and lower values respectively.

The local spatial patterns are obtained as follows: each pixel is encoded using its eight neighbors. First each neighbor is labeled by the values 1 and 0. 1 if the value of the neighbor is above than the value of the central pixel 0 otherwise.

Thus, the thresholded neighborhoods represent a binary code which is assigned to the central pixel. This binary code is converted to decimal (LBP) number; by multiplying the thresholded neighborhood by the weights matrix given to the corresponding pixels. The results obtained from each multiplication are summed, to obtain the LBP code for the central pixel.

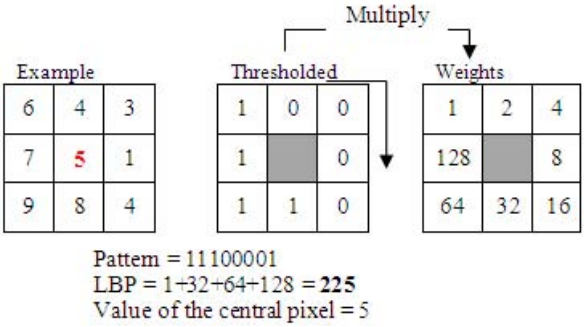


Fig. 1. Calculation of the LBP code

Figure 1 shows an example of the determination of the LBP code. The example matrix is a 3x3 window extracted from the image; all pixels around the central pixel (The value is 5 in red) are coded, by 1 or 0 in the thresholded matrix depending on if they have a higher or lower value than the central pixel. The pattern is extracted from the top left following the clockwise from the thresholded matrix. The resulted binary code is multiplied by the weights matrix to obtain a decimal corresponding code (225 in Figure 1). Figure 2 illustrates the resulted image by applying the LBP operator on an image.

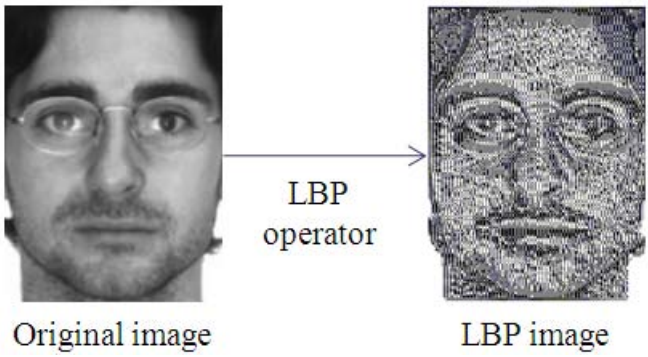


Fig. 2. LBP transform of an image

After the calculation of the LBP code at each pixel, the resulted image (LBP image in Figure 2) is characterized by the histogram of LBP values. The LBP histogram of the 256 possible patterns is created to collect up the occurrences of different binary patterns. The discrete histogram of the patterns in the whole image is used to describe the texture. Usually, histograms are used normalized to get a coherent description.

Let us denote the number of transitions, between 0 and 1 or 1 and 0, of each pattern by a number U as shown in Figure 3.

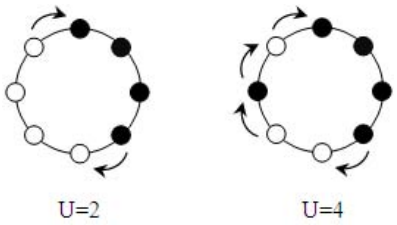


Fig. 3. Number of transitions

In fact, not all of the 256 possible patterns are important and used in the histogram; a specific set of patterns have been proven to be enough to perform a better recognition [13]. These patterns are called the uniform patterns. A pattern is uniform if their number of transitions (between 0 and 1 or 1 and 0) are equal to zero or two, U=0 or U=2. The uniform patterns histogram is used to describe the texture [5][18]. Menp et al. proposed to use a smaller number of spatial transitions in the pattern because it is more relevant to unwanted changes upon a geometric transformation such as rotation [6]. They proposed to use 58 uniform patterns that are composed of (00000000, 11111111), and seven patterns (00000001, 00000011, 00000111, 00001111, 00011111, 00111111, 01111111) with their eight circularly rotated versions. For example: 01111100 and 11011111 (two transitions) are uniform patterns, 111011011 (4 transitions) and 10100101 (6 transitions) are not a uniform patterns. Finally, a histogram which is composed of 59 bins (58 for the uniform pattern and one for all other patterns) is created to collect up the occurrences of different uniform and normalized patterns. This histogram is considered as a feature descriptor of the texture.

3 The Gray Level Co-occurrence Matrix

The Gray Level Co-occurrence Matrix (GLCM) is a second order statistics used to model the relationship between pixels. GLCM was firstly introduced by Haralick [8]. Haralick proposed to use the GLCM which has become one of the most famous, well-known and widely applied as features to describe the texture. The GLCM denoted by $C_{d,\theta}$ (i,j) is calculated by the occurrence frequency of two

pixels having a relationship. This relation is defined by a distance d and an angle direction θ.

For each pair of instance (d,θ) we calculate and obtain a different co-occurrence matrix. GLCM is an square matrix with LxL dimension, where L is the maximum gray level in the image. In case N=256 gray levels, the Co-occurrence matrix will be a matrix with the size of 255x255. However, usually a range of gray levels are used to summarize the calculation. This will reduce the size of the GLCM and the number of gray levels is reduced to 4, 8 or 16 [4]. One offset d ($\triangle x, \triangle y$) is used over the image I to calculate the co-occurrence matrix C, This matrix C is defined following Formula 2.

$$C_{\triangle x,\triangle y}(i,j) = \Sigma_{p=1}^{m}\Sigma_{q=1}^{n} \begin{cases} 1 \; if I(p,q) = i \wedge I(p+ \triangle x, q+ \triangle y) = j \\ 0 \; otherwise \end{cases} \tag{2}$$

Where $C_{(\triangle x,\triangle y)}$ is the Co-occurrence matrix associated to the distance d ($\triangle$x,$\triangle$y). Different distances and directions can be used to obtain different Co-occurrence matrices. However, only d=1,2 with θ = 0, 45, 90 and 135 are suggested in [8]. Figure 4 illustrates how the GLCM is calculated from an image, with a distance 1 and direction 0. It means that GLCM calculate the occurrence value each pixel value and its right neighbor over whole the image. The resulted GLCM matrix is illustrated in Fig 4.

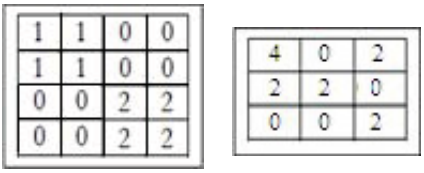


Fig. 4. GLCM calculation [8]

The entry (0,0) of the Co-occurrence matrix (Figure 4) is equal to four because there are four pixel pairs of (0,0) that have a relationship by offset (1,0).

The co-occurrence matrix reveals certain properties about the spatial distribution of the gray levels of the texture. This information cannot be analyzed in a short time. Haralick [8] has proposed a series of statistical indexes that characterize the distribution of the elements of the matrix. The most useful texture features that can be calculated from the co-occurrence matrix are listed below.

- Energy: it is an angular second moment

$$E = \Sigma_i \Sigma_j C(i,j)^2 \tag{3}$$

- Contrast: to extract the local variation of the gray levels. If this value is high, it means that there are few homogeneous regions. This parameter is used

especially to characterize the value dispersion of the GLCM compared to its main diagonal.

$$Con = \Sigma_i \Sigma_j (i-j)^2 * C(i,j) \tag{4}$$

- Homogeneity: This parameter has an opposite behavior of contrast. If this value is high, that means the texture is homogeneous

$$Hom = \Sigma_i \Sigma_j \frac{1}{1+(i-j^2)} C(i-j) \tag{5}$$

- Correlation: to determine if some of the columns of the matrix are equal. The correlation value is important if the values are distributed uniformly in the matrix.

$$Cor = \frac{\Sigma_i \Sigma_j (i-\mu_i)(j-\mu_j) C(i,j)}{\sigma_i \sigma_j} \tag{6}$$

$\mu_i, \mu_j, \sigma_i and \sigma_j$ denote the mean and standard deviations of the rows and columns sums of the matrix, respectively.

A set of co-occurrence matrices can be calculated from textured image due to the different values of (d,θ). Just one matrix is not enough to describe a texture. Thus, a set of matrices are calculated corresponding to each value (d,θ) to describe the texture. The number of the required values of (d,θ) to describe a texture is still a research problem.

4 The Decomposing Architecture

In this section, we will explain our proposed architecture and its different parameters. It is a dynamic segmentation architecture [7]. Some examples and one illustration which summarize all the process will be given at the end of this section. The dynamic segmentation architecture was proposed by [7]. The main idea of the proposed decomposing architecture is to choose one converging point α(x,y) from the image. The main window MW is extracted from α to the bottom-right of the image. The proposed architecture is illustrated in the Algorithm 1 [7]:

The window starting from the converging point α to the bottom-right corner of the image is considered as the main analysis window MW. For each iteration, we generate all possible windows having the same size of MW. The LBP operator is applied, as a feature extraction method, for each generated window to obtain the LBP normalized histogram. After that, the similarity measure between this histogram and the LBP histogram of the researched texture are calculated following this formula:

$$Sim(His_1, His_2) = \Sigma_{i=1}^{n} min(His_1[i], His_2[i]) \tag{7}$$

Where His_1 is the window's histogram and His_2 is the sought texture's histogram. If this similarity measure Sim is above a threshold λ, the window is

Algorithm 1 Square dynamic decomposition system

1. Choose the converging point α.
2. Consider the main window
3. Generate as many windows as possible(with the same size of MW)
4. **for** each window **do**
   (a) Apply the feature extraction method (LBP method)
   (b) Calculate the feature vector (LBP histogram)
   (c) Calculate the similarity *Sim* between the window's feature with the sought texture feature
   if *Sim* is above the threshold **then**
      Save the window's position and its feature vector V
   end if
end for
5. Reduce the size of MW by the distance d
if the size of MW is below the minimum size μ **then**
   Color the saved windows and terminate the process
else
   Goto step 3
end if

considered as relevant and classified as the same researched texture. the coordinates and histogram of the window are saved in one vector Vi=[X, Y, Height, Width, Hist]. Where X, Y, Height and Width are the position of the window, and Hist is the obtained histogram from applied the LBP operator on this window.

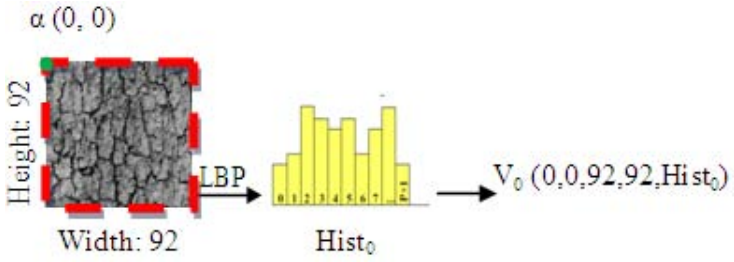


Fig. 5. Extraction of the feature vector

The same steps are repeated by reducing the main window's size by a distance *d*. During the iterations, many squares with the main window's size (same height and width) are selected and their vectors V_i are extracted. Thus, different windows are obtained with various sizes.

To summarize the whole proposed process, Figure 6 illustrations the proposed decomposing architecture. For each iteration the size of the main window is reduced by the distance d (green window in Figure6), and other windows with the same main window's size are generated (red windows), until a minimum size

μ is reached in the iteration n. For each window, a characterized vector V is extracted and saved if it is pertinent (its histogram is above the threshold). This process is stopped when the size of the main window is equal to a minimum size μ. All iterations and how the feature's vectors are extracted and saved for each pertinent window are shown in Figure 6.

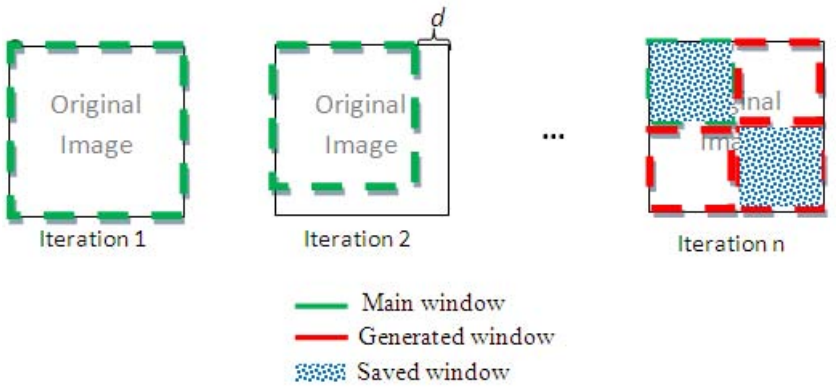


Fig. 6. Dynamic decomposition architecture using square blocks

Figure 6 illustrates the selection of the pertinent squares and the saving process. Only the windows having LBP histogram above the threshold were saved. Finally the saved squares were colored in red on the resulted image.

5 The Proposed Decomposing Architecture

In this section, we will apply the dynamic decomposition architecture using the circular geometric shapes to decompose the image. we will also use the GLCM method to extract the features from the generated squares. The adaptation of the dynamic decomposition process to circular shapes is presented in algorithm 2.

The boundary between the different textures can have a rounded shape. Thus, the square shape of analysis window cannot segment well this boundary. To resolve this problem, we propose to use a circular geometric shape for the segmenting process.

The circular geometric shape of the decomposing blocks segments better the textures than the square shape. This is due to the rounded shape. The circle decomposition process is illustrated in Figure 7.

In order to compare the application of the two geometric forms (square and circle), and to illustrates the improvement made with the proposed circular shape, we did some experiments to illustrate that improvement. we used the two geometric forms for the analyses windows. And we applied the dynamic decomposition architecture and the GLCM method for the feature extraction step. We can notice the weakness of the square decomposition to segment the rounded border.

Algorithm 2 Circular dynamic decomposition system

1. Choose the converging point α (the center of the image).
2. Consider the main circle MC
3. Generate as many circles as possible (with the same radius of MC)
4. **for** each window **do**
   (a) Apply the feature extraction method (LBP method)
   (b) Calculate the feature vector (LBP histogram)
   (c) Calculate the similarity *Sim* between the circle's feature with the sought texture feature
   if *Sim* is above the threshold **then**
   Save the circle's position (the center coordinate and the radius)and its feature vector V
   end if

end for

5. Reduce the radius of the MC by the distance d

if the radius of MC is below the minimum radius μ **then**
Color the saved circles and terminate the process
else
Goto step 3
end if

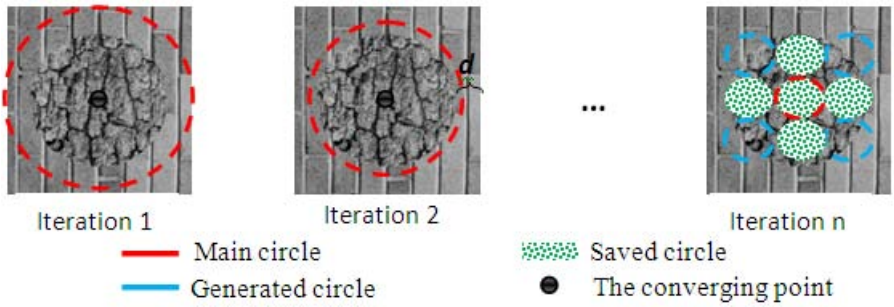
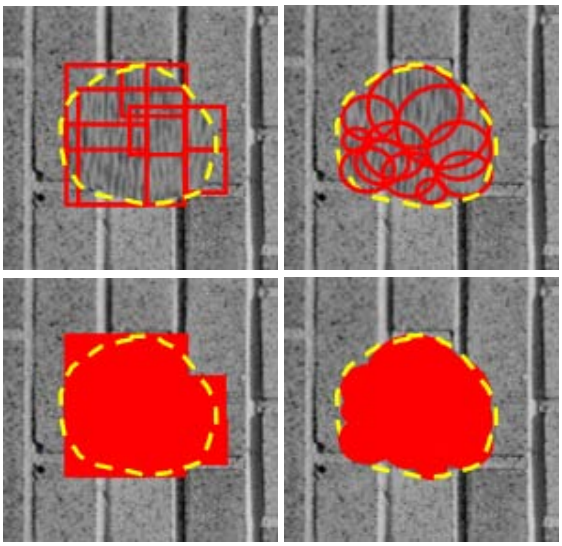


Fig. 7. Iterations of the circular decomposition process

We can also notice that the circle decomposition has solved the rounded border problem and has segmented better the researched texture that the square shape. This improvement is illustrated in Figure 8.

The proposed decomposition architecture is independent from the feature extraction step. So, any feature extraction step can be applied to extract the feature extraction vector. For example, we apply the GLCM as a feature extraction method instead the LBP method. The same mentioned algorithms can be applied except to change 4. (a) Apply the GLCM method, 4. (b) Calculate the four important measures from the GLCM (energy, contrast, homogeneity and correlation) and 4. (c) Calculate the Euclidean distance between the four window's extracted measures with the sought texture extracted measures. If another feature extraction method is used, the same algorithm is applied. Only three point will change: the feature extraction method, the feature vector and the formula of the similarity measure (depending on the feature extraction vector).



(a) (b)

Fig. 8. Comparison between square and circular decomposition shape. a) Results using square decomposition. b) Results using circular decomposition

In this study, many advantages can be noticed from this dynamic decomposing architecture.

- Windows of different size can be extracted and analyzed, thanks to the dynamic size of the MW. Thus, a rich and various set of windows can be extracted and compared with the researched texture.
- Different shapes of the analysis windows and converging points can be considered. Therefore, different configurations can be studied.
- The independence between the proposed architecture and the feature extraction step is a the main advantage of our approach. Thus, there are no requirements for feature extraction step. Which allows to use any features extraction method. Thus, the most adapted method to extract the feature should be used (depending on the application domain).

These advantages make this dynamic architecture a very robust and useful method for texture matching and segmentation.

6 Application and Evaluation

In order to evaluate the performance of the proposed architecture, many parts of the experiments have been done. The first experimenting part illustrates the obtained results using the proposed method and the LBP operator [7]. The Second part shows how to apply the proposed method using the GLCM as a feature descriptor. The third part illustrates the improved performance by

the proposed method compared to the classical methods [7]. The fourth part is about the application of our method on the test images generated randomly from Brodatz album, and using a circle shape of the decomposing window.

Part 1 : Six synthetic images that are composed of different textures (see Figure 9) have been used. From the top left to the bottom right, the numbers of textures within the images are 5, 5, 4, 2, 5, and 4 respectively.

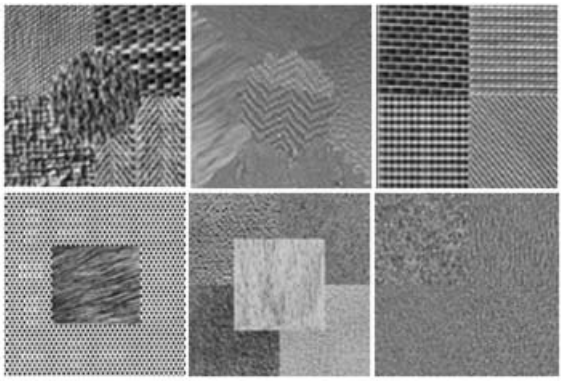


Fig. 9. Synthesis images used in the first experimenting part

In the experiments [7], the parameters of this method are fixed as follows: the reducing distance d was fixed to 10 pixels, the converging point was assigned to (0,0) (the top left pixel of the image), the minimum window's size μ was limited to 5 pixels (the height and width of the main window are greater than 5 pixels) and the threshold was fixed between 0.90 and 0.96 (depending of the researched texture). To evaluate the proposed method, the following steps are performed : First, one square was extracted from the researched texture; this square contains only one homogeneity texture and used as a query to be recognized in the test image. The most similar windows with the query texture (above the threshold λ) are colored with red (Figure 10). The result of this method is an image which contains the most relevant texture compared to the researched texture. The results are shown in Figure 10.

As shown in Figure 10, the query texture (a), which is extracted from the test image (b), is detected as shown in (c). We can notice that this method gives a better result in segmentation when the textures are situated in square areas in the image. This is caused by the choice of the geometric shape of the decomposing window.

Part 2 : Shows the obtained results using the proposed decomposition with a GLCM feature extraction method. The same test images (Figure 10) are used, the first query was changed and the two seconds were the same. The obtained

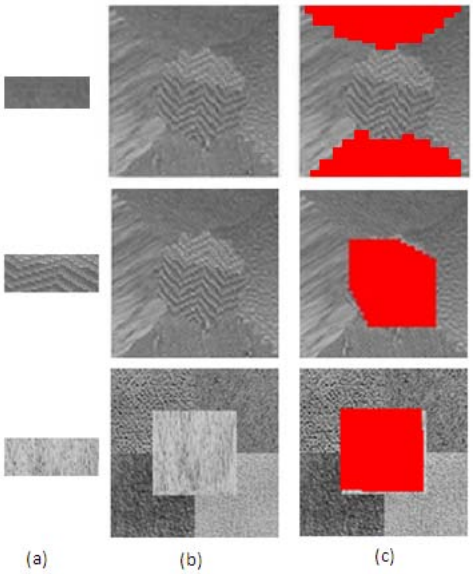


Fig. 10. Texture matching using square decomposing shape. (a) researched texture, (b) test image, (c) resulted image

results are shown in Figure 11. We can notice that obtained results using the proposed method and the LBP operator are a bit better than the GLCM method, because the LBP operator describe better the micro texture than the GLCM method.

Part 3 : The proposed method has been compared with the classical 32x32 decomposing method. The classical decomposition method has been widely used, it segments the image into a static blocks of 32x32 pixels (or other size) and calculate the similarity between the LBP histogram of each block and the sought texture's LBP histogram. Thus, some experiments are made with both the proposed method and the classic 32x32 decomposition. The results are shown in Figure 12 for the classical method and Figure 13 for the dynamic decomposition method. We can notice that this segmentation method can localize approximately the researched texture but some textures are not well recognized, this results have the problem of border detection and scale problem (fixed 32 pixels may not be the best scale) which is a no resolved disadvantage.

For the same test images, we can see that the results given by proposed method are remarkably better than the classical ones. Better result are obtained because the classical method's weaknesses (border detection, fixed scale) are improved in proposed method; the dynamic size of the decomposing windows allows us to analyze the image with different scales. This represents the strength of the dynamic decomposing method we have proposed.

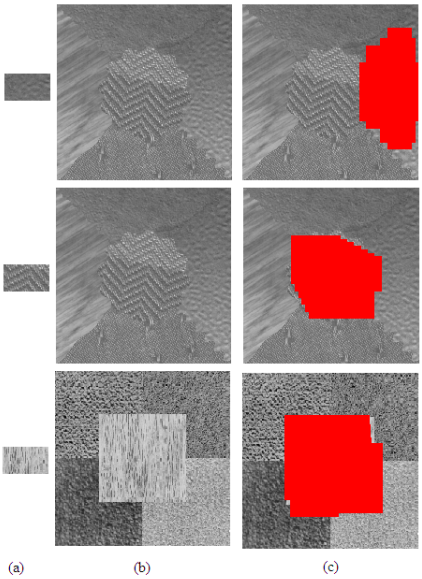


Fig. 11. Texture matching process using square decomposing shape and GLCM. (a) researched texture, (b) test image, (c) resulted image

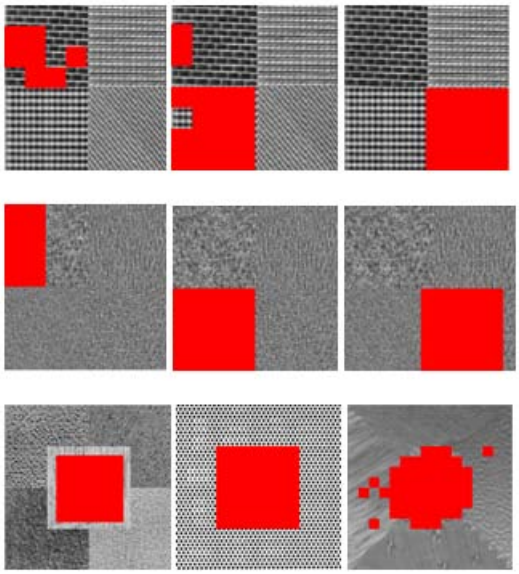


Fig. 12. Some experimental results using classical method and square decomposition shape

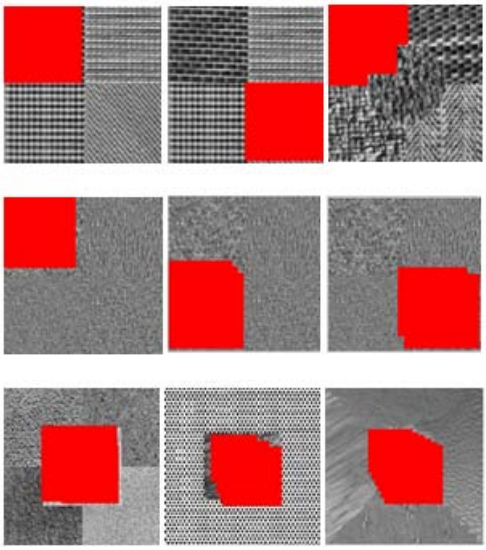


Fig. 13. Some experimental results using dynamic decomposition method and square decomposition shape

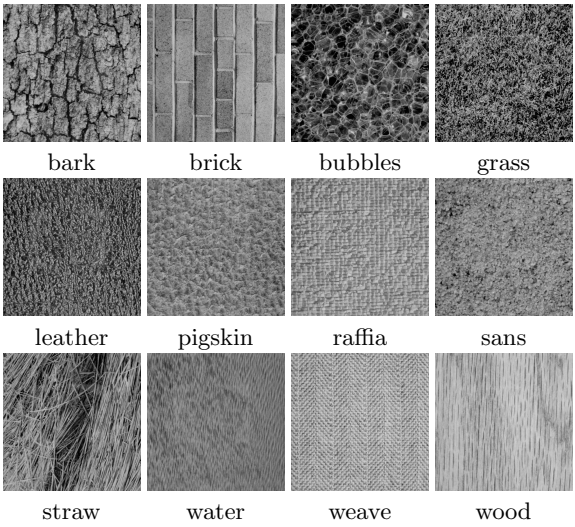


Fig. 14. Some textures from Brodatz album

Part 4 : We have used test images from Brodatz album [3] to evaluate the proposed system performances, different texture images are shown in Figure 14. The name of the Brodatz texture images from the top left to the bottom right in the next figure are: bark, brick, bubbles, grass, leather, pigskin, raffia, sand, straw, water, weave and wood.

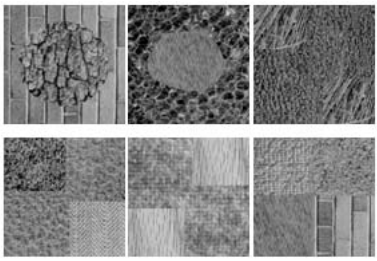


Fig. 15. Test images generated from the Brodatz album

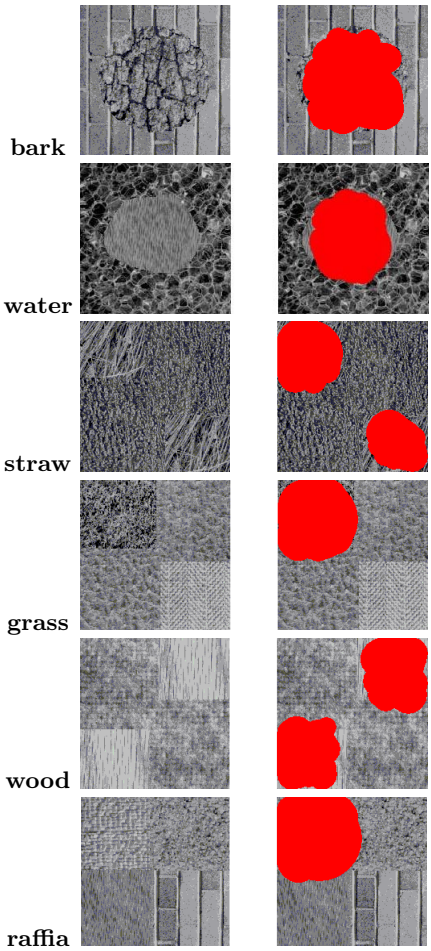


Fig. 16. Some experimental results on the random generated images from the Brodatz album

We have generated test images using a Brodatz texture images database Figure 14. Each image has been generated randomly using the Brodatz texture images as show in Figure 15. From the top left to the button right in the Figure 15, the images contain the following texture: brick and bark, bubbles and water, straw and leather, grass and pigskin and weave, wood and wool, raffia and sand and water and brick.

In the evaluation process, one texture named above has been taken as a query texture. The proposed system using the LBP operator and a circle geometric shape for the analyses window have been applied. The obtained results are shown in Figure 16.

The application of the proposed decomposing architecture on the Brodatz texture images, using LBP operator and circular decomposing blocks, illustrates that the query texture is well recognized in the test images. We can notice that this method gives a good result in segmentation even if the textures are situated in square areas in the image.

In order to resolve the boundary problem and to improve the recognized texture, the combination between the square and circular decomposing architecture has been applied. First the circular dynamic decomposing is used to detect the boundary of the texture. After that, the square dynamic decomposing architecture is used to recognize better the texture between the extracted circles. The result of the combination (square and circular decomposing architecture) is shown in Figure 17.

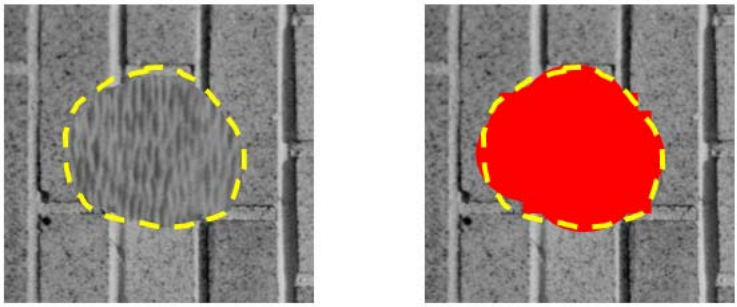


Fig. 17. Test images generated from the Brodatz album

7 Conclusion

In this chapter, we proposed a new dynamic architecture for texture matching and segmentation. This architecture uses dynamic and different window sizes, unlike other classical methods where the size of the window is fixed and static for the whole segmenting process. The main advantages of the proposed method are: different sizes of the analyses windows, which allows studying various size of windows. The independence between the proposed decomposition architecture

and the feature extraction method, which allows using any feature extraction method to describe the texture as well as possible. Different geometric shapes can be used for the analysis windows to obtain several segmentation configurations. For the illustrated process, the LBP operator (which is invariant to monotonic gray scale transformation) and GLCM method have been applied for each window to extract the features. These features are compared with the researched texture feature using a similarity measure. Several experiments have been done. First part shows the application of the proposed method using the LBP operator using synthetic images. Second part shows the application of the proposed method using GLCM method to illustrate the ease of using another feature extraction method. The third part compares the proposed dynamic decomposing architecture with the standard static decomposing method. The fourth part illustrates the application of the proposed system on the Brodatz image database, which is a very famous database of texture analyses, and using a circle geometric shape for the analysis windows. The query texture images have been well recognized and good segmentation results have been obtained with this segmentation architecture. In future works, we will study the behavior and the robustness of the dynamic decomposition approach applied on real textured images and the combination between different geometric shapes of the decomposing blocks.

References

1. Baohua, Y., Yuan, H., Jiuliang, C.: Combining Local Binary Pattern and Local Phase Quantization for Face Recognition. In: Biometrics and Security Technologies (ISBAST), pp. 51–53 (March 2012)
2. Bovik, A.C., Clark, M., Geisler, W.S.: Multichannel texture analysis using localized spatial filters. IEEE Transactions on Pattern Analysis and Machine Intelligence 12(1), 55–73 (1990)
3. Brodatz, P.: Textures: A Photographic Album for Artists and Designers. Dover Publications, New York (1966)
4. Clausi, D.A., Jernigan, M.E.: A fast method to determine co-occurrence texture features. IEEE Trans. on Geoscience & Rem. 36(1), 298–300 (1998)
5. Cuiyu, S., Fengjie, Y., Peijun, L.: Rotation Invariant Texture Measured by Local Binary Pattern for Remote Sensing Image Classification. In: Education Technology and Computer Science (ETCS), vol. 3, pp. 3–6 (2010)
6. Guo, G.Z., Zhang, L., Zhang, D.: A Completed Modeling of Local Binary Pattern Operator for Texture Classification. IEEE Transactions on Image Processing 19(6), 1657–1663 (2010)
7. Hamouchene, I., Aouat, S., Lacheheb, H.: New segmentation architecture for texture matching using the LBP method. In: IEEE Technically Co-Sponsored Science and Information Conference SAI, London, UK, October 7-9 (2013)
8. Haralick, R.: Statistical ans structural approaches to texture. Proc. of IEEE 67(5), 786–804 (1979)
9. Harwood, D., Ojala, T., Pietikinen, M., Kelman, S., Davis, S.: Texture classification by center-symmetric auto-correlation, using Kullback discrimination of distributions. Technical report, Computer Vision Laboratory, Center for Automation Research, University of Maryland, College Park, Maryland. CAR-TR-678 (1993)

10. Kellokumpu, V., Zhao, G., Pietikinen, M., Recognition, M.: Recognition of human actions using texture descriptors. Machine Vision and Applications 22(5), 767–780 (2011)
11. Levesque, V.: Texture segmentation using Gabor filters, Center For Intelligent Machines, McGill university (December 2000)
12. Liao, S., Chung, A.C.S.: Texture classification by using advanced local binary patterns and spatial distribution of dominant patterns. In: IEEE Conference of ICASSP, vol. 1, pp. 1221–1224 (2007)
13. Menp, T., Ojala, T., Pietikinen, M., Soriano, M.: Robust Texture Classification by Subsets of Local Binary Patterns. In: 15th International Conference on Pattern Recognition (ICPR 2000), vol. 3, pp. 39–47 (2000)
14. Ojala, T., Pietikinen, M.: Unsupervised Texture Segmentation Using Feature Distributions. Pattern Recognition 32, 477–486 (1999)
15. Ojala, T., Pietikinen, M., Harwood, D.A.: Comparative Study of Texture Measures with Classification Based on Feature Distributions. Pattern Recognition 19(3), 51–59 (1996)
16. Ojala, T., Pietikinen, M., Menp, T.: Multiresolution Gray-Scale and Rotation Invariant Texture Classification with Local Binary Patterns. IEEE Transactions on Pattern Analysis and Machine Intelligence 24, 971–987 (2002)
17. Richards, W., Polit, A.: Texture matching. Kybernatic 16, 155–162 (1974)
18. Xueming, Q., Xian-Sheng, H., Ping, C., Liangjun, K.: An effective local binary patterns texture descriptor with pyramid representation. Pattern Recognition 44(10-11), 2502–2515 (2011)
19. Zhang, J., Tan, T.: Brief review of invariant texture analysis methods. Pattern Recognit. 35, 735–747 (2002)
20. Zhenhua, G., Lei, Z., David, Z.: Rotation invariant texture classification using LBP variance (LBPV) with global matching. Pattern Recognition 43(3), 706–719 (2010)

Using Digital Image Processing and a Novelty Classifier for Detecting Natural Gas Leaks

Roberlânio de Oliveira Melo*,
Marly Guimarães Fernandes Costa, and Cícero Ferreira Fernandes Costa Filho

Technological and Information Center
Federal University of Amazonas, UFAM
Manaus, Brazil
roberlanio@yahoo.com.br,
marly.costa@uol.com.br,
cffcfilho@gmail.com
http://www.ceteli.ufam.edu.br

Abstract. This paper presents a pattern recognition system for detecting natural gas leaks in the oil and gas industry. More precisely, the detection is done in wellheads of industry installations. In the literature, other methods have been used previously but with some drawbacks. One technique detects gas leaks measuring the CH_4 concentration through the principle of catalytic combustion but suffers from reduced lifespan and a narrow detection range of sensors. Another technique that measures infrared spectrum absorption suffers from high false negative values in the presence of steam. The technique proposed in this study uses radiation in the visible range that can be captured through CCD cameras already present in Closed-Circuit Television systems used to monitor wells. The proposed method uses the novelty classifier concept to detect the leak and identify the region where it occurs. The proposed technique is a pioneering study of natural gas detection with CCD in visible range. The results presented are promising, showing sensitivity and specificity of 94% and 96%, respectively.

Keywords: detection of natural gas leak, novelty filter classifier, gas and oil industry.

1 Introduction

The oil and gas industry is one of the most complex and dangerous fields due to intrinsic characteristics of hydrocarbons, such as: toxicity, flammability and explosion velocity. The occurrence of gas leaks in oil installations generates undesirable financial and environmental consequences, and loss of human lives [1]. Constant monitoring is necessary to avoid these undesirable consequences and there is a great demand for the development of new systems for monitoring and controlling gas leaks. Several methods used to detect natural gas are based on

* Corresponding author.

L. Chen, S. Kapoor, and R. Bhatia (eds.), *Intelligent Systems for Science and Information,* Studies in Computational Intelligence 542,
DOI: 10.1007/978-3-319-04702-7_23,

detecting methane leaking to the atmosphere. In the sequence, we give some examples of them.

The Safety in Mines Research Establishment (SMRS) [2] proposed the principle of catalytic combustion to measure the CH_4 concentrations present in the environment. This principle is based on temperature increases resulting from the heat generated from methane combustion in a catalytic surface employing a palladium as a sensor element. Due to ease in manufacturing and low cost, this device has been used for many years, until presently. These sensors, nevertheless, have a reduced lifespan and a narrow detection range [3].

The analysis of infrared (IR) spectrum absorption has been used more frequently in methane detection. The main reasons are: the IR detector has a lifespan of more than five years, stability and reliability. An IR detection system is comprised of an IR transmitter and receptors with electromagnetic spectrum $\lambda_{IR} = 2 \sim 5\mu m$. When IR radiation interacts with methane gas ($\lambda_{CH_4} \approx 3.5\mu m$), a part of the energy is absorbed and the remaining energy is transmitted [4]. The energy absorbed increases the vibration of methane molecules and, consequently, increases the temperature of the gas. The gas concentration is obtained through the measure of the ratio between the incident and transmitted radiation [3]. The main drawbacks of this system include difficulties in installing and maintaining, as well as the high false negative values in the presence of steam, because the IR radiation is also absorbed by this substance.

Another technique used increasingly in detecting natural gas leaks is digital image processing. In 1997, the U.S. Department of Energy (DOE) together with Sandia National Laboratories for National Security Missions [5] proposed a system called Backscatter Absorption Gas Imaging (BAGI), whose basic principle was to illuminate a gas leak scenario, applying an IR laser, and then photograph this leakage using an IR camera. Systems employing this technology are very expensive [6], costing as much as US$ 80,000 when used for a single inspection, but not for continuous inspection.

Usually natural gas leaks appear to humans as a white cloud or patch of fog, because when the methane comes into contact with the atmosphere, its low temperature induces air condensation [7]. Considering this fact, this study proposes a natural gas detection method in wellheads of an onshore petroleum installation sites using a Closed-Circuit Television system with Charge-Coupled Device (CCD) cameras to monitor wells. As these systems are already available, no additional expense is necessary for hardware implementation.

In a previous work [8] we studied the use of the novelty filter, as described by Costa et al. [9], to investigate the presence of natural gas leaks in digital images captured by CCD cameras. In this paper we deepen this previous study. The main objectives of this paper are stated as follows:

- Test the system with different types of images such as daytime and nighttime images and noise images, with rain and human presence near the wellhead;
- Identify the component of the RGB and HSI space that better distinguishes images with a gas leak from images with no gas leak.

This study is organized according to the following sections: sections II and III describe the concepts of novelty filter and novelty classifier. Section IV presents the pattern recognition system developed for detecting natural gas leak. Section V presents the results and, finally, section 6 presents a discussion of these results.

2 Novelty Filter

A novelty filter is a kind of auto-associative memory, proposed by Kohonen [10]. Its workings can be understood through the following steps:

1. Store familiar patterns in a memory;
2. Apply a given input to the memory input and retrieve from the memory the pattern that best matches the input;
3. The novelty is defined as the difference between the given input and the retrieved pattern.

According to Kohonen [10], the most elementary form of associative memory is a physical network, where an input pattern is directly transformed into a corresponding output pattern. The scheme shown in Fig. 1 illustrates a novelty filter using this kind of associative memory.

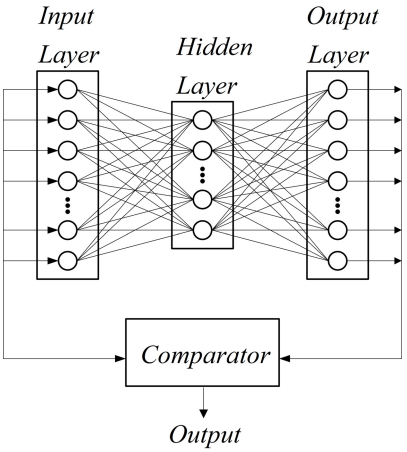


Fig. 1. Illustration of a Novelty Filter using a feedforward neural network

It is assumed that the "familiar" patterns are stored in a previously trained neural network. When presenting a pattern to the input of the neural network, one of the stored patterns that best matches the input is retrieved in the output. The input then compared with this best match output and a difference vector, the novelty, is presented in the output. If the norm of this vector is lower than a given threshold, it is assumed that the right pattern was retrieved.

In the literature, the scheme just described was used in two pattern recognition applications. Beh et al. [11] applied the novelty filter to the conventional dual channel Generalized Sidelobe Canceller (GSC) for improving the performance of speech enhancement. The idea presented by the authors is that the novelty filter is trained with noise patterns, so that the output reproduces the noise present in the input. When a noisy speech signal is presented in the input of the trained network, at the output, the noise signal is reproduced. So subtracting the input from the output, it is possible to recover the speech signal without noise. The authors representative experiments confirm the superior performance of the proposed method over the conventional ones.

Elsimary [12] also applied the novelty filter to detect shorted turns and mechanical failures in rotating machines. Measurement of the electrical current on a single phase of the power supply of an induction motor is used to estimate the motor condition. The authors store some spectra of this current signal for machines that present no problems. In the training phase the output of the neural network, trained with backpropagation algorithm, resembles the normal spectra of current signal presented at the input. It is supposed that when a machine presents a problem, the spectra of current signal is modified. When this modified signal is presented at the input of the neural network the output presents one of the normal current signals stored in the training phase that best resembles the new input. The difference between the input and the output of the neural network represents the novelty.

Another approach to calculate the novelty proposed by Kohonen [10] uses the concept of auto-associative memory as an orthogonal projection. In this case, the novelty filter is submitted to a supervised training that uses the Gram-Schmidt orthogonalization method to produce a set of orthogonal vectors.

Consider a group of vectors $\{x_1, x_2, \ldots, x_m\} \subset R^n$ forming a base that generates a subspace $L \subset R^n$, with $m < n$. An arbitrary vector $x \in R^n$ can be decomposed in two components, $\hat{x}$ and $\tilde{x}$, where $\hat{x}$ is a linear combination of vectors x_k and $\tilde{x}$ is the orthogonal projection of x on a subspace $L \perp$ (orthogonal complement of L). Fig. 2 illustrates the orthogonal projections of x in a tridimensional space. It can be shown, through the projection theorem, that $\tilde{x}$ is unique and has a minimum norm. So, $\hat{x}$ is the best representation of x on subspace L.

The $\tilde{x}$ component of the vector can be thought of as the result of an operation of information processing, with very interesting properties. It can be assumed that $\tilde{x}$ is the residue remaining when the best linear combination of the old patterns (base vectors x_k) is adjusted to express vector x. So it is possible to say that $\tilde{x}$ is the new part of x that is not explained by the old patterns. This component is named novelty and the system that extracts this component from x can be named the novelty filter. The vectors base x_k can be understood as the memory of the system, while x is a key through which information is associatively searched in the memory.

It can be shown that the decomposition of an arbitrary vector $x \in R^n$ in its orthogonal projections $\hat{x} \in L \subset R^n$ and $\tilde{x} \in L \perp$ can be obtained from a linear

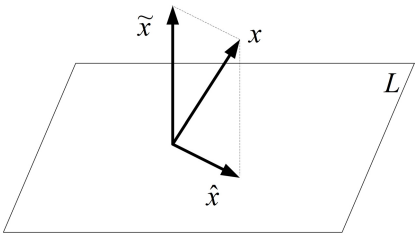


Fig. 2. Illustration of novelty filter concept using Gram-Schmidt Orthogonalization method

transformation, using a symmetric matrix P, so $\hat{x} = P \cdot x$ and $\tilde{x} = (I - P) \cdot x$. The matrix P is named orthogonal projector operator in L (P would be what is named as novelty filter), and $(I - P)$ orthogonal projector in subspace $L \perp$, as described by Kohonen [10].

Consider a matrix X with $x_1, x_2, \ldots, x_k$, with $k < n$, as its columns. Suppose that the vectors $x_i \in R^n$, $i = 1, 2, \ldots, k$ span the subspace L. As cited above, the decomposition of $x = \hat{x} + \tilde{x}$ is unique and $\tilde{x}$ can be determined through the condition that it is orthogonal to all columns of X. In other words:

$$\langle \tilde{x}^T \cdot X \rangle = 0 \tag{1}$$

The Penrose solution [13] to equation 1 is given by:

$$\tilde{x}^T = y^T \left(I - X \cdot X^+ \right) \tag{2}$$

Where:
y is an arbitrary vector with the same dimension of $\tilde{x}$;
X^+ is the pseudo-inverse matrix of X.
Using the properties of symmetry and idempotence of the pseudo-inverse matrix, it follows that:

$$\langle x^T \cdot \tilde{x} \rangle = x^T \cdot \left(I - X \cdot X^+ \right) \tag{3}$$

$$\langle x^T \cdot \tilde{x} \rangle = \langle \tilde{x}^T \cdot x \rangle = y^T \cdot \left(I - X \cdot X^+ \right)^2 \tag{4}$$

Comparing equations 3 and 4, it follows that $y = x$. So $\tilde{x}$ can be written as:

$$\tilde{x} = \left(I - X \cdot X^+ \right) \cdot x \tag{5}$$

As $\tilde{x}$ is unique, it follows that: $(I - P) = (I - X \cdot X^+)$ and $P = X \cdot X^+$.

When working with images, the calculation of projection matrix P, because of the dimensions involved, becomes an immense and time-consuming computational task. Each column of matrix X is a reference pattern or, in neural network terminology, a training vector. A vector like this is constructed from stacking the image columns. For example, with images of 128×128 pixels, the dimension of the column vector is $n = 16.384$ and the dimension of X matrix is $n \times N$,

where N is the number of training vectors (images). So, in this case, P results in a square matrix with dimension 16.384. Thus, it is preferable to obtain the novelty $\tilde{x}$ through an iterative technique based on the classical Gram-Schmidt orthogonalization method. This method results in creation of a base of vectors mutually orthogonal, $\{h_1, h_2, h_3, \ldots, h_n\} \in R^n$, from the training vectors $\{x_1, x_2, x_3, \ldots, x_n\} \in R^n$.

To build a base of mutually orthogonal vectors, a direction is first chosen, for example, the direction x_1, so:

$$h_1 = x_1 \tag{6}$$

In the sequence, this expression is used:

$$h_k = x_k - \sum_{j=1}^{k-1} \frac{\langle x_k, h_j \rangle}{\|h_j\|^2} \cdot h_j, \quad for \quad k = 2, 3, \ldots, m \tag{7}$$

Where:

$\langle x_k, h_j \rangle$ is the inner product of x_k and h_j.

The way that vectors h_j are constructed, it follows that the set $\{h_1, h_2, h_3, \ldots, h_n\}$ spans the same subspace as the set $\{x_1, x_2, x_3, \ldots, x_n\}$.

Given a sample x, to obtain the novelty $\tilde{x}$, it is necessary only to continue the process described by equation 7 one step more: $\tilde{x} = h_{n+1}$.

3 Novelty Classifier

The novelty classifier can be understood as a classifier that employs a supervised training. Two types of classifier can be built: the single class and the multiclass classifier. In this work we use only the single class one. Differing from neural networks, the training set of the novelty filter consists only of vectors that belong to a given class. No other kinds of vectors are used.

The training consists of generating the set $\{h_1, h_2, h_3, \ldots, h_n\}$ from the training set $\{x_1, x_2, x_3, \ldots, x_n\}$, according to equation 7.

The single class problem attempts to classify a given sample as belonging or not to a given class. In this problem, the first task is training the novelty filter as described above. The second task is choosing a test set that contains samples that belong to this class and samples that do not belong to this class. Each member of this set is applied to the trained novelty classifier and then the norm of the novelty is calculated. The norm of the members of the subset that belongs to the class is expected to have lower value than the norm of the members of the subset that does not belong to the class. So an optimum threshold is determined to separate these two subsets. For example, a Receiver-operating Characteristic (ROC) curve can be used to determine a threshold that maximizes the sensitivity and specificity of the novelty classifier. After, a sample x can be classified as belonging or not do this class as illustrated in block diagram of Fig. 3. If the norm $|\tilde{x}|$ is lower than the threshold, the output of block diagram is 0 and the

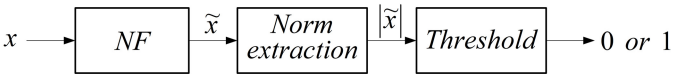


Fig. 3. Illustration of a single class novelty classifier

sample belongs to the class. On the other hand, if the norm $|\tilde{x}|$ is greater than the threshold, the output is 1 and the sample does not belong to the class.

In the literature the novelty classifier concept was used in some pattern recognition systems [9],[14]. The first one applies the novelty classifier to iris recognition and the second one applies the novelty classifier to diagnostic of cancer in scintmammographic images.

4 Pattern Recognition System

The Fig. 4 shows a block diagram of the pattern recognition system used for natural gas leaks. In the sequence we will explain all the operations shown.

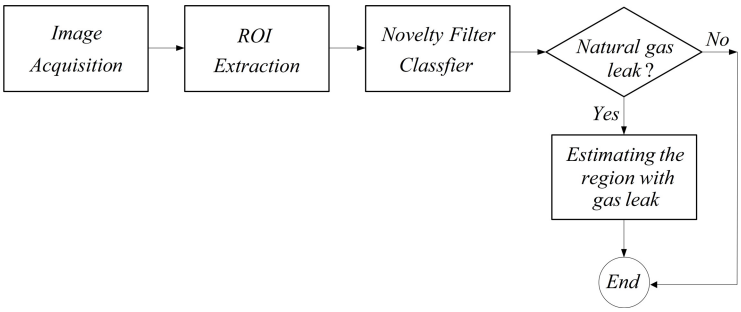


Fig. 4. Pattern recognition system for detecting natural gas leaks

4.1 Image Acquisition

The images of the wellhead area are obtained through a Yokogawa® image device. This equipment consists of a CCD camera in one side and an infrared illumination on the other side. This infrared illumination was used for night-time image capture. The spatial resolution of the CCD sensor is 320×240 pixels. This equipment was mounted $15m$ far from the wellhead, as shown if Fig. 5. The video capture card used was a MSI® VOX^{TM} USB one, with maximum resolution of 720×480 pixels. The videos were recorded in a notebook with the following characteristics: Intel® i5-520M Core^{TM}, 2 GB RAM and 320GB HD.

The Fig. 6a shows an example of an image extracted from a recorded video. Inside this image, a Region Of Interest (ROI) closer to the wellhead was selected, through a cropping operation, with dimensions 128×128. The objective was to reduce the noise from trees and the sky and to concentrate the novelty filter

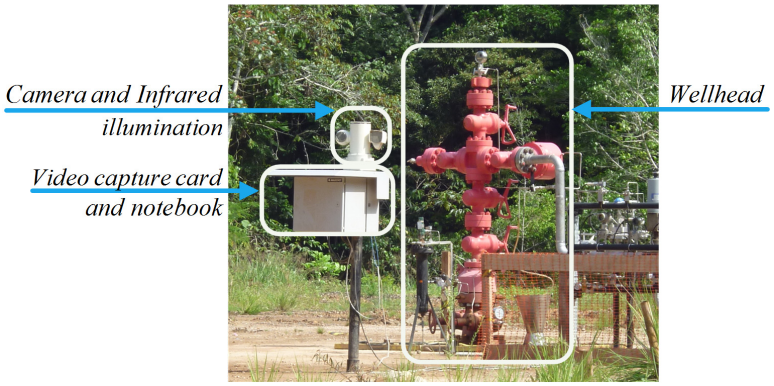


Fig. 5. Image equipment mounted

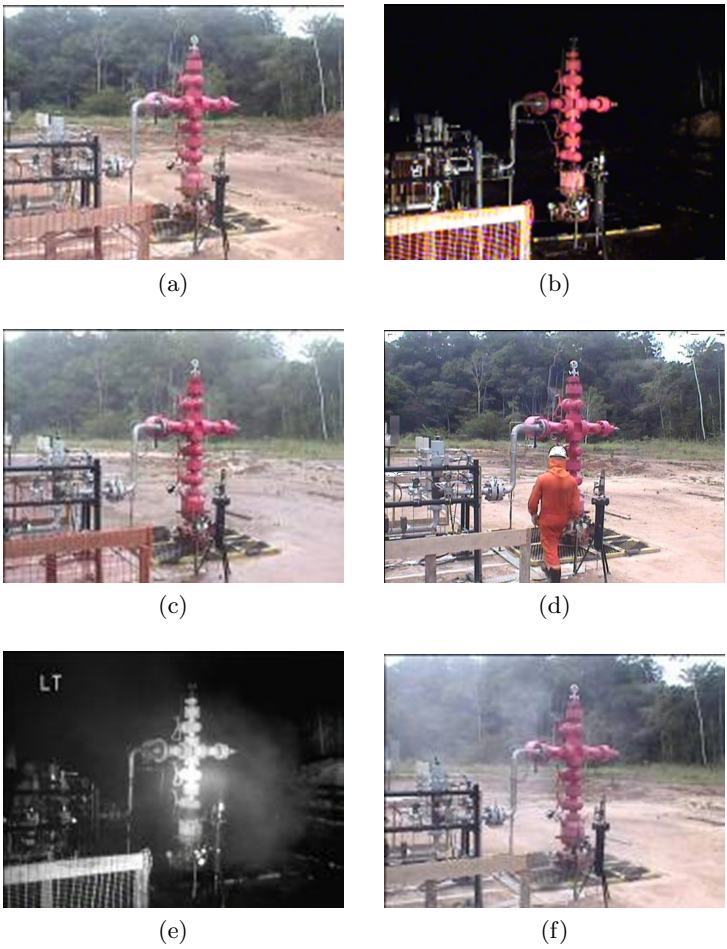
(a) (b)

Fig. 6. Images used in the study: (a) Original image; (b) ROI extracted close to the wellhead

focus on the Christmas tree of the wellhead, where the leaks are more frequent. An ROI with no natural gas leak is shown in Fig. 6b.

A set of 3060 images was extracted from recorded videos. A subset of 2000 images was comprised of images with no gas leaks and a subset of 1060 images was comprised of images with natural gas leaks. To achieve a robust pattern recognition system, wellhead images were captured in different conditions: daytime images with no noise, daytime images with human presence near the wellhead, daytime images with rain, daytime images with rain and human presence near the wellhead, night images with no IR illumination, nighttime images with IR illumination and no noise, nighttime images with infrared illumination and human presence near the wellhead. Fig. 7 shows some examples of captured images.

Table 1 shows the number of images with in each of these conditions, with gas leak and with no gas leak.



(a) (b) (c) (d) (e) (f)

Fig. 7. Examples of Captured Images: (a) daytime with no noise and no gas leak; (b) nighttime with no noise and no gas leak; (c) daytime with rain and no gas leak; (d) daytime with human presence near the wellhead; (e) nighttime with gas leak; (f) daytime with gas leak

Table 1. Number of images in different conditions

Time	Condition	No gas leak images	Gas leak images	Total
Daytime	No noise	500	250	750
	Human presence	250	200	450
	Rain	250	250	500
	Rain and human presence	250	60	310
Nightime	With IR and no noise	250	200	450
	With IR and human presence	250	100	350
	Total	1750	1060	2810

4.2 Trainig and Testing Methodologies

Six different novelty filter classifiers were designed, one for each color component of the RGB and HSI space. To use the novelty filter, each image of 128×128 pixels is converted in a vector of size 16384.

The size of the novelty filter basis depends on the training and test methodology adopted. Some methodologies, as "leave-one out", "holdout" and "cross-validation" [15] can be used for training and testing a classifier. In this study, due to the large amount of images, we used the holdout method, using training and testing sets of the same size, according to the scheme proposed in [15] and shown in Fig. 8.

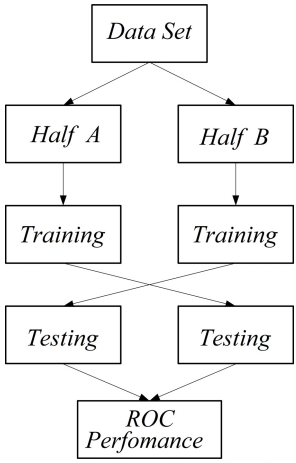


Fig. 8. Scheme used for training and testing the novelty filter classifier

The holdout method divides the data into two subsets, the training and the testing data set. Each novelty classifier is trained with images of one half and tested with images of another half. Each training set consists of 875 images with no gas leak. The testing set of one half consists of 875 images with no gas leak of the other half and of 1060 images with gas leak.

For each color component of RGB and HIS spaces, two ROC curves are obtained: one for half A and other for half B. For each half, one optimal threshold is obtained as the one that generates a ROC curve point nearest the ideal point of the ROC curve, the point with coordinates $(0, 1)$. A median threshold is calculated from these two previous values.

Two parameters were used to evaluate the ROC curve performance. The first one is the Area Under ROC Curve, AUC, and the other the Standard Error of the ROC curve, SE, given by equation equation 8 [16].

$$SE = \sqrt{\theta \cdot (1-\theta) + (n_A - 1) \cdot \left(\frac{\theta}{2-\theta} - \theta^2\right) + (n_N - 1) \cdot \left(2 \cdot \frac{\theta^2}{1+\theta} - \theta^2\right)} \tag{8}$$

Where:
θ-AUC;
n_N-Number of normal images (with no gas leak) used in test set;
n_A-number of abnormal images (with gas leak) used in test set.

5 Results

Fig. 9 shows two example of the applied methodology for detecting gas leaks. Fig. 9a shows an original image with a gas leak near the ground level, while Fig. 9b shows corresponding novelty image. Fig. 9c shows another image with a gas leak in left side, while Fig. 9d shows corresponding novelty image.

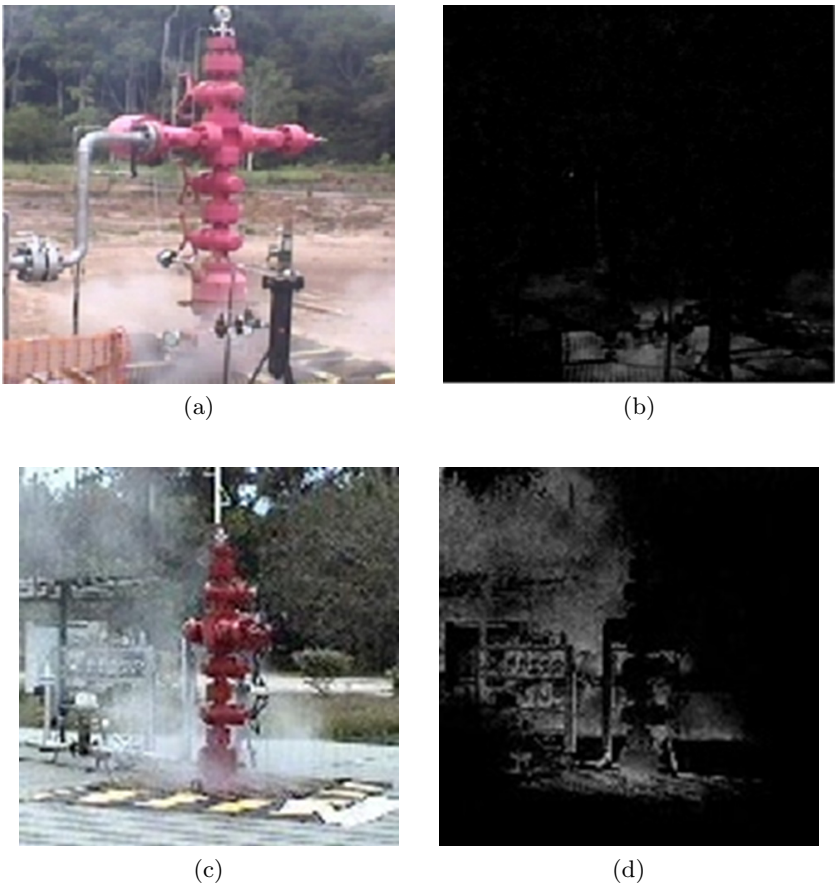
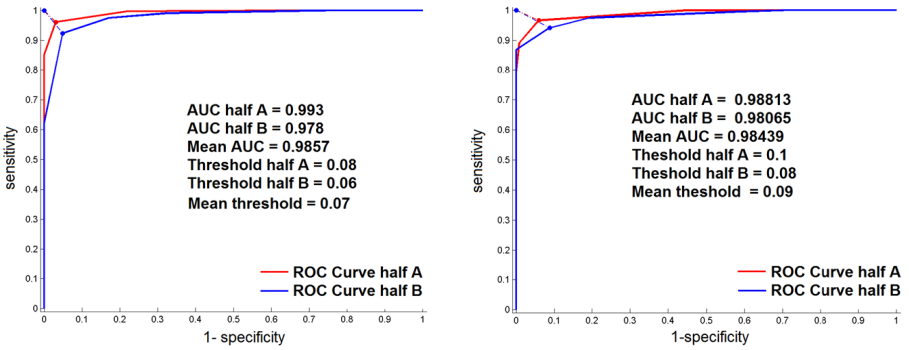


Fig. 9. (a) original image with gas leak near the ground level; (b) corresponding novelty image of (a); (c) original image with gas leak in left side; (d) corresponding novelty image of (c)

The Fig. 10a shows the ROC curves obtained for component R of RGB space, while Fig. 10b shows the ROC curves obtained for component I of HSI space. The distances of the optimum points from the ideal points of the ROC curve $(0, 1)$ are shown with traced segments. As shown, the performance of component R is much better than the performance of component I.



(a) (b)

Fig. 10. (a) ROC curves of component R of RGB space; (b)) ROC curves of component I of HSI space

Table 2 shows a comparison of the performance of all six color components used in this study, considering the AUC and SE parameters.

Table 2. Performance of components R, G, B, H, S and I

Color Component	Group				AUC	SE
	A		B		mean	mean
	AUC	SE	AUC	SE	valeu	value
R	99,25%	0,26%	97,80%	0,45%	98,53%	0,37%
G	97,83%	0,45%	97,27%	0,5%	97,55%	0,47%
B	77,11%	1,33%	78,63%	1,3%	77,87%	1,31%
H	66,92%	1,49%	64,26%	1,51%	65,59%	1,5%
S	28,57%	1,3%	69,96%	1,45%	49,26%	1,54%
I	98,81%	0,33%	98,07%	0,42%	98,44%	0,38%

6 Discussion and Conclusion

In the proposed method for detecting natural gas leaks, as cameras are already present in the Closed-Circuit Television system used to monitor the well, no additional expense is necessary for hardware implementation. This was a pioneering study of natural gas detection with CCD in the visible range. When

comparing the method just proposed with catalytic [3] and infrared [4] methods previously discussed in the introduction, we notice that the proposed method detects leaking in the region around the wellhead, covering an area of 360^o. The other two methods, however, detect leaking only in sensor position.

The following characteristic of the classifier used, a novelty filter classifier, are worth noting: the images must always be obtained in the same position and must be the same size.

From the five color components tested, the component R of RGB space and the component I of HSI space shown the best AUC values, with mean values above 98%. The worst mean value of AUC was obtained for S component of HIS space. The best pair of sensitivity and specificity mean values was obtained for component R, and was 94% and 96%, respectively. The R component shows also the best SE value.

The novelty classifier was successfully tested with some environmental changes, as rain and lack of daylight (night images). At night, the use of IR illumination, which is also already available in Closed-Circuit Television systems used to monitor wells, make it possible to detect leaks with little performance degradation. Nevertheless, for better performance, future works will address the question of using a specific novelty filter for low lighting conditions.

Acknowledgments. We would like to thank FAPEAM and FINEP (process 0329/08), for financial support. Jim Hesson of AcademicEnglishSolutions.com proofread the English.

References

1. Liu, H., Zhong, S., Rui, W., Keqiang, L.: Remote helicopter-borne laser detector for searching of methane leak of gas line. In: Prognostics and System Health Management Conference, pp. 1–5. IEEE, Shenzhen China (2011)
2. Lv, X.-Q., Zheng-Yong, Z., Kong, D.-Y.: A Catalytic Sensor using MEMS process for Methane Detection in Mines. In: International Conference on Information Acquisition, ICIA 2007, pp. 4–9 (2007)
3. Fan, Z., Taishan, L., Liping, Z.: BP Neural Network Modeling of Infrared Methane Detector for Temperature Compensation. In: Proceedings of 8th International Conference on Electronic Measurement & Instruments, ICEMI 2007, pp. 4123–4126. Xi 'An, China (2007)
4. Krier, A., Sherstnev, V.V.: Powerful interface light emitting diodes for methane gas detection. Journal of Physics D: Applied Physics 33(2), 101–106 (2000)
5. McRae, T.G., Kulp, T.J.: Backscatter absorption gas imaging: a new technique for gas visualization. Journal of the Optical Society of America, Applied Optical 32, 4037–4050 (1993)
6. Kastek, M., Sosnowski, T., Pitkowski, T., Polakowski, H.: Methane detection in far infrared using multispectral IR camera. In: 9th International Conference on Quantitative Infrared Thermography, pp. 1–4. Krakow, Poland (2008)
7. Ross, C.E.H., Solan, L.E.: Terra Incognita: A Navigation Aid for Energy Leaders. PennWell Corporation, Oklahoma (2007)

8. Filho, C.F.F.C., de Oliveira Melo, R., Costa, M.G.F.: Detecting natural gas leaks using digital images and novelty filters. In: Kamel, M., Karray, F., Hagras, H. (eds.) AIS 2012. LNCS, vol. 7326, pp. 242–249. Springer, Heidelberg (2012)
9. Costa, C.F.F.F., Pinheiro, C.F.M., Costa, M.G.F., Pereira, W.C.A.: Applying a novelty filter as a matching criterion to iris recognition for binary and real valued feature vectors. Journal Signal, Image and Video Processing 7(2), 287–296 (2013)
10. Kohonen, T.: Self-Organization and Associative Memory, 3rd edn. Springer, Heidelberg (1989)
11. Beh, J., Baran, R.H.: Hanseok Ko: Dual channel based speech enhancement using novelty filter for robust speech recognition in automobile environment. IEEE Transactions on Consumer Electronics 52(2), 583–589 (2006)
12. Elsimary, H.: Implementation of neural network and genetic algorithms for novelty filters for fault detection. In: IEEE 39th Midwest Symposium on Circuits and Systems, vol. 3, pp. 1432–1435 (1996)
13. Penrose, R.: A generalized inverse for matrices. Proceedings of the Cambridge Philosophical Society 51, 406–413 (1955)
14. Costa, M.G.F., Moura, L.: Automatic assessment of scintmammographic images using a novelty filter. In: Proceedings of the 19th Annual Symposium on Computer Applications in Medical Care, So Paulo, pp. 537–541 (1995)
15. Bowyer, K.W.: Validation of Medical Image Analysis Techniques. In: Sonka, M., Fitzpatrick, J.M. (eds.) Handbook of Medical Imaging. SPIE, vol. 2, pp. 567–607 (2000)
16. Hanley, J.A., McNeil, B.J.: The meaning and use of the area under a Receiver Operating Characteristic (ROC) curve. Radiology 143(1), 29–36 (1982)

Health Monitoring Systems Using Machine Learning Techniques

Fahmi Ben Rejab, Kaouther Nouira, and Abdelwahed Trabelsi

BESTMOD, Institut Supérieur de Gestion de Tunis
Université de Tunis
41 Avenue de la Liberté, 2000 Le Bardo, Tunisie
fahmi.benrejab@gmail.com, kaouther.nouira@planet.tn,
abdel.trabelsi@gmail.com

Abstract. This paper describes the steps of building two and efficient monitoring systems in intensive care unit (ICU). We propose two new systems that deal with large data sets and solves the main problems of the current monitoring system. In fact, the current monitoring system in ICU has many issues to detect real states of patients namely critical and normal states. It frequently generates a high number of false alarms having bad effects on the working conditions. Besides, these alarms can threat the patient life by misleading medical staff. Our aim, in this paper, is to avoid false alarms and keep a high level of sensitivity by improving the current monitoring system. In addition, our purpose is to generate groups of patients suffering from similar diseases and building a general model for similar patients. The obtained models will make the classification of new patients possible. To this end, we combine two incremental versions of support vector machines mainly the LASVM and ISVM techniques with the k-prototypes clustering method. The first proposed system is the KP-ISVM which is based on the ISVM technique and the k-prototypes. The second one is called KP-LASVM which takes profits of both the LASVM by reducing the false alarms and the k-prototypes by selecting the appropriate model to start classifying new patients. Both of our proposals are characterized by dealing with large amount of data streams and adding new patients. However, the system using LASVM and k-prototypes i.e. the KP-LASVM has produced the best results compared to the others monitoring systems and based on different evaluation criteria. All experimental results using real-medical databases have been analyzed and have proved the performance of the KP-LASVM.

Keywords: Intensive care unit, monitoring system, support vector machines, LASVM, classification, k-prototypes.

1 Introduction

Monitoring patients in intensive care unit (ICU) is a critical task made by nurses and doctors. In fact, the ICU is a special department in hospitals for patients

L. Chen, S. Kapoor, and R. Bhatia (eds.), *Intelligent Systems for Science and Information,* Studies in Computational Intelligence 542,
DOI: 10.1007/978-3-319-04702-7_24,

who suffer from dangerous diseases and have critical states. These patients need an intensive care from medical staff and they have to be always monitored using monitoring systems. As a result, monitoring system is considered as a fundamental tool. It uses monitoring devices to measure several medical parameters that indicate the state of patients. Each measured parameter has a threshold set by doctors. In case of having values that exceed their thresholds, an alarm is trigged. This alarm is considered as indication for nurses and doctors that the state of the patient is no more stable but very critical. As a result, this patient needs a particular treatment or his state will be more critical and he can even die. Unfortunately, there are many false alarms trigged by the current monitoring system. In fact, the monitoring system can trig alarms that does not indicate a real critical state but, in some cases they are due to a wrong setting of parameters, or a bad setting of monitoring devices. Besides, the monitoring system do not take into account of the relation between the measured parameters. It separately measures each parameter which can lead to false alarms.

Hence, false alarms present a real danger for the patient life. They do not report the real state of patients which can make the monitoring task more complicate. Furthermore, the working condition of the medical staff become more difficult since, such alarms disturb them and make patients under more pressure. As a result, avoiding these false alarms become more and more necessary.

Several researches have focused in this problem and many works have been proposed to avoid decrease the high number of false alarms in intensive care unit. We can mention the use of the digital signal processing in [9] where there is also a clinical validation study for two recently developed on-line signal filters, the use of trend extraction methodology based on the time evolution of signals in [13], and the use of the intelligent monitoring [21] detailed through the time series technology and multi-agent sub-systems. Moreover, there are several other studies [11], [26] that have reported and detailed this issue.

In [3], we have used the machine learning techniques namely the support vector machines (SVM) and the multilayer perceptron as attempt to reduce the high level of false alarms in intensive care. We have also proved that the support vector machines in batch mode provides better results than the multilayer perceptron when applied on medical databases. In our previous work [6], we have improved the results obtained in [3] by testing two incremental versions of the support vector machines namely the LASVM [8] and the ISVM [10]. These incremental techniques have been successfully applied on intensive care unit and both of them have provided interesting results. They have considerably improved the monitoring systems by reducing false alarms and keeping the high level of sensitivity.

However, these proposed systems based on machine learning techniques have successfully avoided the problem of the high level of false alarms but, how about the problem of monitoring new patients? we aim to monitor patients at real time and using information provided after monitoring similar patients. Our purpose is also to build models that allows the classification of new patient states. To this end, we propose two real time monitoring systems. The first one, called

KP-ISVM, combines the ISVM with the k-prototypes clustering technique. The second one, called KP-LASVM, is based on the LASVM and the k-prototypes techniques. The LASVM and ISVM techniques reduce false alarms and maintain a high level of sensitivity. In addition, the k-prototypes allows the classification of new patients. As a result, these novel systems can classify new patients without using the training set but, by using the most similar model to new patients. To select the best system, we compare the KP-ISVM and KP-LASVM to the current monitoring system and the incremental systems proposed in [6].

The rest of the paper is structured as follows: Section 2 and Section 3 present respectively an overview of the monitoring system in ICU and the incremental versions of SVM. Section 4 explains the clustering of new patients' states. Section 5 detail experiments of the new monitoring systems i.e. the KP-LASVM and KP-ISVM. Section 6 concludes the paper.

2 Monitoring System in Intensive Care Unit

Monitoring patients in intensive care unit (ICU) is a fundamental task. There is a special system in hospital called monitoring system which monitor patients with critical states. Monitoring system in ICU measures several medical parameters. Each measured parameter has a particular threshold indicating the state of the patient. Table 1 shows an example of some measured parameters with their thresholds.

Table 1. Some measured parameters

Medical parameters	Minimum value	Maximum value
Heart Rate	50	120
Respiratory rate	5	25
Pulse rate	65	115
Saturated percentage of Oxygen in the blood	90	130

If the value of the parameter exceeds its limits an alarm is trigged. Medical staff consider trigged alarms as indication of a critical state. As a result, this patient needs an immediate treatment.

However, not all trigged alarms correspond to a critical state. In fact, there are many false alarms, and monitoring system can mislead the medical staff. There are different alarms caused by the wrong setting of the thresholds, or the bad use of the devices. Besides, monitoring system does not take into account of the relation between medical parameters which can make the monitoring inaccurate.

Permanent false alarms have bad impact on nurses and patients. They cause a lot of stress and make the working condition more complicate. Besides, medical staff cannot take immediate decision but, they will delay their treatment which can threat the patient life. As a result, the current monitoring system looses

its main task which is monitoring patients. It is only considered as a measuring device.

In order to overcome the problems of false alarms in intensive care unit, many researchers have focused on this issue. They have proposed different works [1], [11], [24]. We can mention [23] where the author has shown in an US study that the response time of alarms can be up to 40 min. In addition, as described in [11], medical staff relay only on 10% of the total number of generated alarms. Furthermore, 50% of all relevant and true alarms are missed and are not taken into consideration by physicians.

Furthermore, to improve the current monitoring system in ICU, two monitoring systems have been proposed using two incremental versions of support vector machines namely the ISVM [4] and the LASVM [5]. Besides, in our previous work [6], we have made a comparison between the monitoring system based on the LASVM and the system using ISVM. we have proved that our proposals have successfully reduced the number of false alarms.

In this work, we take profits of the advantages of both ISVM and LASVM and we combine them with the k-prototypes method in order to only detect true alarms and to classify new patients.

3 Incremental SVM

In this section, we present the support vector machines (SVM) technique and the two incremental versions of the SVM namely the LASVM and the ISVM.

3.1 The SVM

Support vector machines (SVM), originally introduced to the literature in 1995 by Vapnik [15], are relatively new learning method that classifies any data into tow classes by finding the optimal hyperplane that separates the data. SVM has become one of the most widely used classification technique since its introduction and has been successfully applied in different fields [17], [19], [27]. In many medical decision support systems, the SVM has shown important results. We can mention [2] and [3] where it was used in batch mode and applied in ICU to better monitor the current system.

Generally, the used real-world data can be either linearly or nonlinearly separable. The SVM can easily and efficiency handle these two kinds of data.

1. **Case of linearly separable data:** where an optimal hyperplane can be drawn to separate the two classes as illustrated in Figure 1 using training data. This hyperplane can be defined as:

$$x_i w + b = 0 \tag{1}$$

In order to set w (a vector normal) and b (a scalar), we have to solve a convex quadratic programming (QP) problem:

$$\begin{cases} \min \frac{1}{2}||w||^2 + C, w \in R^d, b \in R. \\ subject\ \ to \\ y_i(x_i w + b) \geq 1 for\ i = 1, ..., m. \end{cases} \tag{2}$$

with the decision rule given by:

$$f_{w,b}(x) = sign(w^T x_i + b). \tag{3}$$

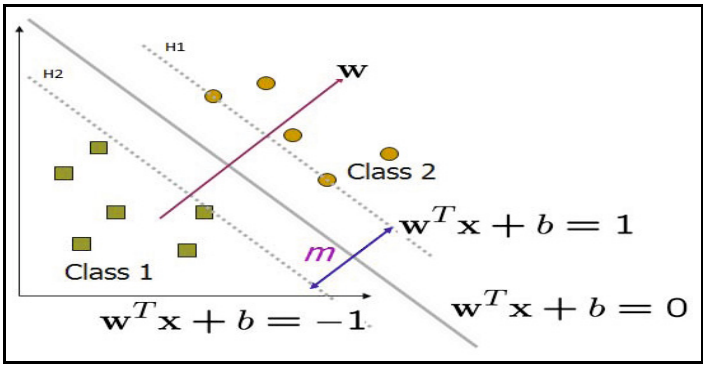


Fig. 1. The optimal hyperplane for linearly separable data

here w presents the weight vector, b is the bias (or $-b$ is the threshold), x_i presents an observation and P and N are respectively positive and negative data (the class of x_i), y_i denotes the class of the observation x_i, m the number of observations and R^d the number of dimension.

We can define two more hyperplanes, H_1 and H_2 parallel to the separating hyperplane. The distance between the tow planes are referred as SVM's margin. The goal is to find the optimal hyperplane that maximizes the margin while is equidistant from both H_1 and H_2.

2. **Case of non-linearly separable data:** A nonlinear classifier can be implemented by applying the kernel trick. We project the used data into a feature space in order to separate instances by an hyperplane. Figure 2 details the data mapping which generates a linear separation between positive and negative instances of the training set.

   The data mapping are defined through the function Φ defined by $R^d \rightarrow R^D (D \succ\succ d)$, with R^D is HILBERT space.

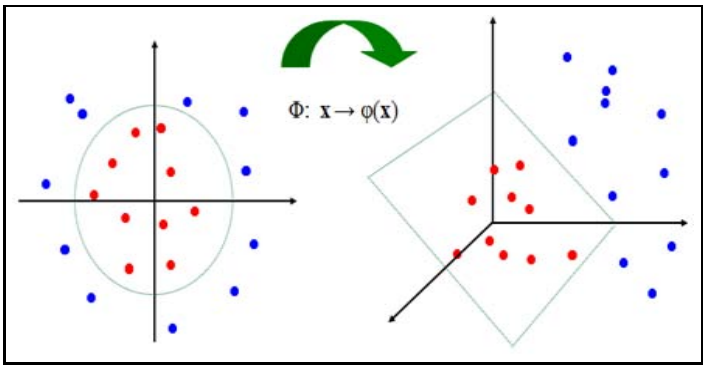


Fig. 2. The mapping of data for nonlinearly separable data

To detect the hyperplane, we have to solve the optimization problem defined as follows using the slack variable ξ_i .

$$\min \frac{1}{2}||w||^2 + C\sum_{i=1}^{m}\xi_i, w \in R^d, b \in R. \tag{4}$$

subject to

$$y_i(x_i w + b) \geq 1 - \xi_i, \xi_i \geq 1 for\ i = 1, ..., m. \tag{5}$$

Despite of providing efficient solution for linear and non-linear data, the SVM cannot take account of new information provided over time. To overcome its limit, two modified version of SVM have been proposed namely the LASVM [7] and the ISVM [10]. These latter are described with details in following subsections.

3.2 The Incremental Versions of SVM: The LASVM and ISVM

In this section, we focus on the incremental learning strategy that allows adding instances over time and hence, considering new information.

1. **Online and Active SVM: The LASVM**

   The LASVM proposed in [7], [8] presents a well-known incremental algorithm. It handles large amount of information (i.e. it can train large training sets) that are add over time.

   The LASVM is an incremental (or on-line) and a modified version of the SVM. It has been successfully applied in several works [22], [28].

   The LASVM algorithm is characterized by being faster than the SVM and uses less memory. In addition, it is able to handle noisy data sets and can avoid the over-fitting problem by discarding the useless support vectors.

The LASVM is based on the Sequential minimal optimization (SMO) implemented by the libsvm tool [12]. The SMO defines a τ which is a small positive tolerance where the $\tau - violating$ pair (i, j) is defined such that:

$$(i,j)\, is\ a\ \tau - violating\ pair \Leftrightarrow \begin{cases} \alpha_i < B_i \\ \alpha_j > A_j \\ g_i - g_j > \tau. \end{cases} \quad (6)$$

In fact, the LASVM selects data points then inserts them as support vectors. After that, it calculates the α and g using the process procedure.

Actually, the LASVM algorithm contains two main procedures mainly the reprocess procedures. They are described with details as follows.

The Process Procedure. It consists of selecting of data point as a new training instance and then, finding the corresponding support vector which forms with this new instance the $\tau - violating$ pair. The maximal gradient g is obtained through this pair and, the weight of each support vector is recomputed at the end.

The Reprocess Procedure. It is based on the remove of support vectors having α equals to zero.
actually, the $\tau - violating$ pair is determined through the maximal g then, the update of the α weight is made. After that, it is necessarily to verify if there is any support vector with $\alpha = 0$ to remove it. Finally, the update of the gradient g and the bias b are made.

2. **The Incremental SVM: ISVM** Incremental SVM offers different advantages mainly the possibility to be trained with huge data sets and the availability of data over time and not at a priori.

   The incremental and decremental SVM is a well-known online algorithm [10], based on the SVM, that has shown its efficiency in several works [18], [25].

   Incremental SVM learning algorithm details how to use the new data over time. In our work, The ISVM adds new samples at periodic time and ignores all previous data except their support vectors. As follows, the pseudo code of the incremental SVM detailed in [10].

(a) *Train the initial SVM on the initial training set.*
(b) *Classify the new sample using the initial model in step 1.*
(c) *Check if this new sample is a support vector or an error vector, if it is a support vector updates the initial model of SVM.*
(d) *Go to step (2).*

The main idea of the ISVM is to find whether the inputted data is in question and to submit these instances to the user to determine its right class. In the questing process, the machine quests the interrogative instance by the distance between the point and hyperplane.

The distance can be defined as:

$$DIS(x,w) = \left| \sum_{\in SV} \alpha_i y_i (\phi(x).\phi(x_i)) + b \right|, x_i. \tag{7}$$

with w the weight vector, b the bias, x_i is an observation, SV the set of support vectors, and ϕ the mapping function.

4 Clustering of Patients

4.1 The K-prototypes Method

The k-prototypes is a partitional clustering method proposed in [16] by Huang for mixture data i.e. categorical and numeric data. It is a modification version of the k-means method [20]. The k-prototypes approach uses two types of distances between attributes. The euclidean distance for numeric attributes and the simple matching dissimilarity measure applied between categorical attributes.

Assume we have two objects X and Y with mixture attributes' values i.e. numeric and categorical. The dissimilarity $d(X,Y)$ between these objects can be obtained using the following formula:

$$d(X,Y) = \sum_{j=1}^{p} (x_j - y_j)^2 + \gamma \sum_{j=p+1}^{m} \delta(x_j, y_j). \tag{8}$$

with p and m are respectively the number of numeric and categorical attributes and γ is a weight used to not favoring any type of attributes.

In our work, we propose to use the k-prototypes method to cluster patients having numeric and categorical measured values relative to medical parameters. The clustering results will be groups patients having the same diseases.

4.2 Monitoring System Based on Machine Learnings Techniques

In classification technique we have to dispose of a training set before starting the classification of the patient states, which is impossible in our medical case. To avoid this problem, we propose to build a general classification model based on incremental SVM for each group of patients. In order to obtain these groups, we use the k-prototypes clustering method detailed in the previous subsection. After building the models relative to different groups of patients, we can classify the observations of a new monitored patient. In fact, when we have a new patient, we try to find the appropriate cluster using the distance (see Equation (8)). We start classifying the states of the patient using the general model of this cluster.

Figure 3 explains the new approach.

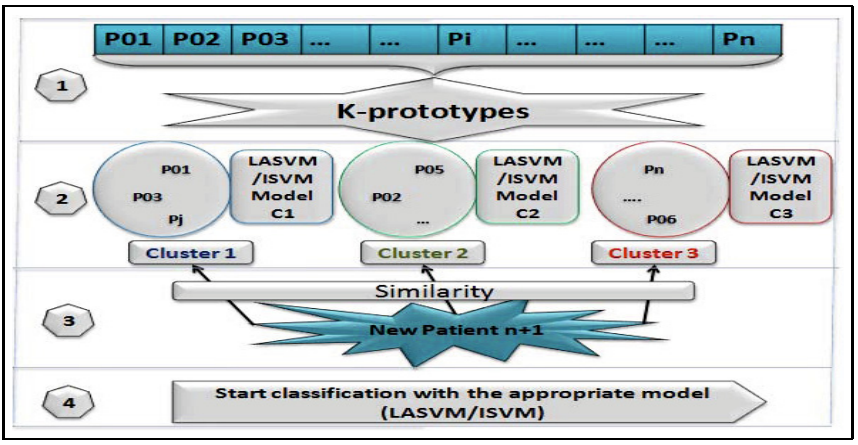


Fig. 3. The steps of the proposed approach

From this figure, we can detect the four steps provided by our new approach.

1. First, we use the k-prototypes clustering method to create groups of similar patients (patients who suffer from similar diseases).
2. Then, we use the incremental SVM method to generate a general model for each group of patients. This model will represent all patients belonging to the same group.
3. After that, we add a new monitored patient with new values of attributes.
4. Finally, we can easily and efficiency classify the values of this patient by using the most similar model to it. Note that this model is chosen after computing the similarity between the new patient and all groups and finding the appropriate group of this patient.

5 Experiments

In this section, we describe with details and compare the results of different monitoring systems (i.e. the current system, the systems based on SVM, ISVM and the LASVM and the new systems KP-ISVM and KP-LASVM).

5.1 Monitoring System Using Incremental Versions of SVM and K-prototypes

We detail as follows the main steps used to build the incremental systems.

1. *Cluster patients into several groups as explained in Figure 3.*

2. *Divide data set into training set and test set. Use the stratified sampling which assigns 70% of the data points to the training set and 30% of data points to the test set.*
3. *Use the grid search technique, proposed by Chen and Lin in [14] and based on the cross-validation, to set the appropriate parameters for the ISVM and the LASVM.*
4. *Build new incremental systems i.e. ISVM and LASVM from training data.*
5. *Add new patient after finding the most similar model to it based on distance.*
6. *Start the classification of new patient states.*
7. *Evaluate KP-ISVM and KP-LASVM using a test set.*

5.2 The Framework

To test and compare our proposed systems based on the ISVM and the LASVM and the k-prototypes, we used 14 data sets from MIMICII (Multiparameter Intelligent Monitoring for Intensive Care) database taken from Physiobank [21]. This database contains data from hemo-dynamically unstable patients hospitalized in 1996 in ICU of the cardiology division in the Teaching Hospital of Havard Medical School. There are 100 patients' records of continuous data recorded each second. Each recording is annotated by an expert in order to specify the state of the patient (critical or not). There are also many measured variables for each patient such as Heart Rate (HR), Oxygen Saturation (SpO_2), Non-Invasive Blood Pressure (NBP), Respiratory rate (Resp), Artery Blood Pressure (ABP), Pulmonary Artery Pressure (PAP) [21].

Note that, to avoid biased results, we have computed the average of the used evaluation criteria. Table 2 details real-world databases taken from MIMICII.

Table 2. Description of the used data sets

Databases	**#Attributes**	**#Instances**
Patient 01	6	4101
patient 02	8	42188
patient 03	8	42188
patient 04	7	42188
patient 05	9	42188
patient 06	9	5350
patient 07	7	11300
patient 08	7	10600
patient 09	12	5700
patient 10	5	42188
patient 11	7	42188
patient 12	7	42188
patient 13	9	42188
patient 14	7	42188

Where #Attributes and #Instances denote respectively the total number of measured parameters and the total number of instances for a specific database.

Table 3 shows the attributes of each patient used in this study. By using these different medical parameters (corresponding to the attributes), we can cluster patients having similar symptoms in the same group using k-Prototypes clustering method.

Table 3. Patients' attributes

Patients	Age	Sex	Diagnostic	Surgery	Rhythm	ART	PAP_{max}	PAP_{min}	RAP	RESP	Ventilation
P 01	80	F	Carotid	endartarec	Unifocal	95	56	16	10	16	Spontaneous
P 02	71	M	grafting	graft	Normal	85	32	8	4	14	Spontaneous
P 03	47	F	laporatomy	obstruction	Normal	110	48	28	9	36	Spontaneous
P 04	64	F	cholecystitis	Cholecys	Normal	90	32	16	5	18	Spontaneous
P 05	56	M	laporatomy	graft	Sinus	90	28	20	12	12	Controlled
P 06	72	M	Endocarditis	Sepsis	Ventricular	72	47	20	11	8	Intermittent
P 07	60	F	trauma	graft	Normal	92	32	18	12	16	Spontaneous
P 08	71	F	Endocarditis	regurgitation	Ventricular	70	34	16	10	16	Spontaneous
P 09	56	M	grafting	graft	Sinus	26	103	26	16	20	Intermittent
P 10	56	F	Endocarditis	regurgitation	sinus	86	80	35	16	17	Spontaneous
P 11	70	F	Nephrectomy	regurgitation	Unifocal	102	32	16	6	20	Spontaneous
P 12	77	F	trauma	regurgitation	multifocal	95	42	20	16	11	Intermittent
P 13	73	M	Angioplasty	graft	Ventricular	87	56	32	10	20	Spontaneous
P 14	58	M	Coronary	graft	Unifocal	84	41	13	13	18	Spontaneous

5.3 Evaluation Criteria

For the test and evaluation of our novel monitoring systems i.e. KP-LASVM and KP-ISVM, we use three evaluation criteria detailed as follows:

1. The false alarm reduction rate (FARR) [9]:

$$FARR = \frac{Suppressed \;\; false \;\; alarms}{Total \;\; number \;\; of \;\; false \;\; alarms}. \tag{9}$$

2. The rate of false alarms defined by the error rate (ER):

$$ER = \frac{FP + FN}{TP + TN + FP + FN}. \tag{10}$$

with FP, FN, TP and TN present respectively False Positive, False Negative, True Positive and True Negative alarms.

3. The sensitivity (S) defined by the ability of the system to detect positive results.

$$S = \frac{TP}{TP + FN}. \tag{11}$$

5.4 Experimental Results

In this section, we report and analyze all experimental results provided by all tested systems. We compare our proposed monitoring systems (KP-ISVM and KP-LASVM) to the current system (CS) and systems based on SVM, ISVM, and LASVM.

We will first apply the k-prototypes clustering method to real medical data sets in order to cluster patients with similar diseases to the same group. After that, we will build a general model for each group of patients using the incremental SVM (LASVM and ISVM).

The following table (Table 4) shows the obtained clusters after applying the k-prototypes on data sets (detailed before in Table 3).

Table 4. Cluster results

Clusters	Patients
C 01	Patient 01; Patient 03; Patient 07; Patient 12
C 02	Patient 02; Patient 05; Patient 09; Patient 13; Patient 14
C 03	Patient 04; Patient 06; Patient 08; Patient 10; Patient 11

The results illustrated in Table 4 are obtained as follows: when we have a new monitored patient with new values of attributes, we can easily classify this patient by using the most similar model to it. We have to compute the similarity between the new patient and all clusters. After that, we have to select the appropriate cluster. Finally, we choose the most similar model to it and we start the classification of patient states using this model.

As the current monitoring system trigged many false alarms. The main aim of our proposed systems i.e. KP-ISVM and KP-LASVM is the reduction of this high rate of false alarms.

Table 5 illustrates the FARR relative to different monitoring systems respectively based on the SVM, ISVM, LASVM, KP-ISVM, and KP-LASVM.

Table 5. Suppressed false alarm for different patients data sets

Patients	**SVM**	**ISVM**	**LASVM**	**KP-ISVM**	**KP-LASVM**
patient 01	98,74	99,98	99,41	**99,98**	**99,98**
patient 02	99,9	99,97	99,94	99,97	**99,98**
patient 03	99,96	99,98	99,94	**99,98**	**99,98**
patient 04	98,79	99,16	98,55	99,26	**99,85**
patient 05	99,15	99,86	99,77	99,86	**99,98**
patient 06	99,25	99,3	98,54	99,36	**99,76**
patient 07	99,56	99,82	99,64	99,82	**99,91**
patient 08	99	99,53	98,73	99,53	**99,85**
patient 09	99,71	99,8	99,22	99,8	**99,98**
patient 10	92,12	99,98	96,91	**99,98**	**99,98**
patient 11	99,76	99,92	99,54	99,92	**99,98**
patient 12	99,76	99,85	99,36	99,85	**99,91**
patient 13	99,76	99,94	99,2	99,94	**99,98**
patient 14	99,8	99,89	99,78	99,89	**99,98**

From Table 5, we can remark that the system using KP-ISVM and KP-LASVM have considerably improved the results of the monitoring systems using the SVM, ISVM, LASVM. Besides, we can notice that the KP-LASVM provides the best results for all patients. It has successfully removed almost all false alarms trigged by the current system. For example, for the first three patients, the KP-LASVM has reduced the rate of false alarms by 99.98%. These results proves the ability the the new systems to remove false alarms especially for the KP-LASVM monitoring system which improves the results of all systems. This improvement which corresponds to a reduction in the rate of false alarms is the result of the successful combination of the k-prototypes with the LASVM technique.

Table 6 shows the error rate of the KP-ISVM and KP-LASVM versus the ISVM, LASVM, SVM, and the current system.

Table 6. Error rate of the LASVM, ISVM, SVM, KP-ISVM, KP-LASVM and the current system

Patients	CS	SVM	ISVM	LASVM	KP-ISVM	KP-LASVM
patient 01	94,92	19,42	0,65	12,54	**0,64**	0,66
patient 02	98,93	17,36	4,91	6,83	4,9	**3,66**
patient 03	99,51	7,96	6,76	16	6,66	**5,78**
patient 04	88,5	11,7	8,32	12,31	8,24	**8,12**
patient 05	94,83	22,01	3,52	5,75	3,54	**3,15**
patient 06	100,33	13,91	12,93	25,86	12,89	**11,98**
patient 07	98,63	29,41	15,25	26,15	15,16	**14,05**
patient 08	102,86	20,51	13,08	30,17	13,08	**12,01**
patient 09	103,65	14,85	10,07	35,16	9,98	**9,1**
patient 10	72,99	17,84	0,35	7,14	0,33	**0,24**
patient 11	100,98	6,29	2,16	22,63	2,13	**2,1**
patient 12	95,18	7,76	4,3	21,81	4,25	**4,18**
patient 13	83,62	4,56	2,64	6,56	2,61	**2,46**
patient 14	98,35	17,37	11,82	20,19	**11,62**	11,74

Looking at Table 6, we can notice the high performance of the KP-ISVM and KP-LASVM by providing low error rates for all patients. It is also obvious that KP-LASVM has provided the best results compared to other systems. For example, for patient 10, the KP-LASVM produces an error rate equal to 0.24% compared to the current system which generated more than 70% of error rate for all patients.

The reduction of the rate of false alarms and the improvement of the current monitoring system prove again the performance gain of the KP-LASVM system.

In the following, Table 7 illustrates the sensitivity of the KP-ISVM and KP-LASVM compared to the incremental approaches (LASVM and ISVM) and the current system.

Table 7. Sensitivity of the LASVM, ISVM, SVM, KP-ISVM, KP-LASVM and the current system

Patients	CS	SVM	ISVM	LASVM	KP-ISVM	KP-LASVM
patient 01	100	99,68	99,68	97,41	98,54	**98,75**
patient 02	100	91,82	97,68	98,14	97,57	**98,14**
patient 03	100	99,19	96,62	95,41	96,28	**96,78**
patient 04	97,78	96,55	97,54	97,33	97,38	**97,72**
patient 05	97,34	91,38	98,8	98,1	98,19	**98,99**
patient 06	48,36	93,21	93,7	87,68	93,48	**95,12**
patient 07	100	94,44	96,3	94,44	96,25	**97,48**
patient 08	23,3	95,15	95,15	89,32	95,12	**96,59**
patient 09	4,6	92,5	94,89	83,3	94,61	**95,74**
patient 10	87,01	95,13	99,7	98,35	99,25	**99,98**
patient 11	36,28	100	100	89,54	99,85	**99,98**
patient 12	95,5	96,51	98,29	89,46	98,11	**99,98**
patient 13	100	96,76	97,7	97,41	97,64	**98,36**
patient 14	100	93,69	94,4	91,97	94,38	**95,55**

As shown in Table 7, the current system is characterized by a high level of sensitivity. This is due to the generation of many alarms, most of them are false the rest are true alarms. Besides, we can deduce that our proposals have also provided high sensitivity which proves its performance.

However, we can remark that the sensitivity of the current system is not stable as the sensitivity of the KP-LASVM (more than 95% for all patients). These instable results are obvious especially for patients 08 (23.3%) and 09 (4.6%).

Table 8 presents a comparison between all systems based on the number of true alarms. Note that true alarms consists of alarms generated by the monitoring system and indicating a real critical state of the patient.

Moving to the number of true alarms presented in Table 8, we remark that the new system based on the KP-LASVM has proved again its performance. In fact, the use of the LASVM with the k-prototypes techniques has improved the results by detecting relevant alarms i.e. true positive alarms. Besides, the KP-LASVM guarantees the classification of new patient states. Furthermore, we notice that the KP-LASVM gives very similar results to the expert who indicates the critical cases where alarms should be generated.

In addition, in contrast to the new monitoring system, the current system has a trouble to detect true positive alarms. This is obvious especially for patient six, eight and nine.

Table 8. Number of true positive alarms of the monitoring systems vs. the expert

Databases	CS	SVM	ISVM	LASVM	KP-ISVM	KP-LASVM	Expert
patient 01	309	308	308	301	308	309	309
patient 02	1076	988	1051	1056	1051	1066	1076
patient 03	740	734	715	706	717	736	740
patient 04	4533	4476	4522	4512	4523	4612	4636
patient 05	1942	1823	1971	1957	1971	1982	1995
patient 06	883	1702	1711	1601	1715	1798	1826
patient 07	54	51	52	51	52	52	54
patient 08	24	98	98	92	99	99	103
patient 09	27	543	557	489	557	582	587
patient 10	2002	2189	2294	2263	2298	2300	2301
patient 11	378	1042	1042	933	1042	1042	1042
patient 12	1232	1245	1268	1154	1272	1282	1290
patient 13	6909	6685	6750	6730	6801	6890	6909
patient 14	697	653	658	641	666	688	697

6 Conclusion

In this paper, we have focused on the problem of monitoring system in intensive care unit. We have proposed two new monitoring systems (KP-LASVM and KP-ISVM) combining the LASVM then the ISVM with the k-prototypes. Through our study, we have proved that the KP-LASVM has reduced the rate of false alarms, has detect true positive ones and has produced high sensitivity compared to other systems. In addition, using the KP-LASVM, the addition of new states of patients is possible and easy. All we need is finding the most similar model to the added patient. This model represent patients having similar states and it is generated using the k-prototypes clustering technique. All experimentations are based on real medical databases taken from MIMICII and different evaluation criteria.

References

1. Baxter, G.D., Monk, A.F., Tan, K., Dear, P.R., Newell, S.J.: Using cognitive task analysis to facilitate the integration of decision support systems into the neonatal intensive care unit. Artificial Intelligence in Medicine 35, 243–257 (2005)
2. Ben Rejab, F., Nouira, K.: Reducing False Alarms in Intensive Care Units Monitoring System Using Support Vector Machines. CCCM 2010 4, 106–109 (2010)
3. Ben Rejab, F., Nouira, K., Trabelsi, A.: Support Vector Machines versus Multilayer Perceptrons for Reducing False Alarms in Intensive Care Units. International Journal of Computer Applications, Foundation of Computer Science 49, 41–47 (2012)

4. Ben Rejab, F., Nouira, K., Trabelsi, A.: On the use of the incremental support vector machines for monitoring systems in intensive care unit. In: TAEECE 2013, pp. 266–270 (2013)
5. Ben Rejab, F., Nouira, K., Trabelsi, A.: Monitoring Systems in Intensive Care Units using Incremental Support Vector Machines. In: ICDIPC 2013, pp. 273–280 (2013)
6. Ben Rejab, F., Nouira, K., Trabelsi, A.: Incremental Support Vector Machines for Monitoring Systems in Intensive Care Unit. In: SAI 2013, pp. 496–501 (2013)
7. Bordes, A., Bottou, L.: The Huller: a simple and efficient online svm. In: Gama, J., Camacho, R., Brazdil, P.B., Jorge, A.M., Torgo, L. (eds.) ECML 2005. LNCS (LNAI), vol. 3720, pp. 505–512. Springer, Heidelberg (2005)
8. Bordes, A., Ertekin, S., Weston, J., Bottou, J.: Fast Kernel Classifiers With Online And Active Learning. Journal of Machine Learning Research 6, 1579–1619 (2005)
9. Borowski, M., Siebig, S., Wrede, C., Imhoff, M.: Reducing false alarms of intensive care online monitoring systems: An evaluation of two signal extraction algorithms. In: Computational and Mathematical Methods in Medicine, vol. 2011 (2011)
10. Cauwenberghs, G., Poggio, T.: Incremental and Decremental Support Vector Machine Learning, pp. 409–415 (2000)
11. Chambrin, M.C.: Alarms in the intensive care unit: how can the number of false alarms be reduced. Journal of Critical Care 5, 184–188 (2001)
12. Chang, C., Lin, C.: LIBSVM: A library for support vector machines. Transactions on Intelligent Systems and Technology 2, 27:1–27:27 (2011)
13. Charbonnie, S., Gentil, S.: A trend-based alarm system to improve patient monitoring in intensive care units. Control Engineering Practice 15, 1039–1050 (2007)
14. Chen, Y.W., Lin, C.J.: Combining SVMs with various feature selection strategies, `http://www.csie.ntu.edu.tw/cjlin/papers/features.pdf`
15. Cortes, C., Vapnik, V.: Support vector networks. Machine Learning 20, 273–297 (1995)
16. Huang, Z.: Clustering large data sets with mixed numeric and categorical values. In: Proceedings of the 1st Pacific-Asia Conference on Knowledge Discovery and Data Mining, pp. 21–34 (1997)
17. Ivanciuc, O.: Applications of Support Vector Machines in Chemistry. Rev. Comput. Chem. 23, 291–400 (2007)
18. Luo, J., Pronobis, A., Caputo, B., Jensfelt, P.: Incremental learning for place recognition in dynamic environments. In: Proc. IROS 2007 (2007)
19. Ming, H.T., Vojislav, K.: Gene extraction for cancer diagnosis by support vector machines an improvement. Artificial Intelligence in Medicine 35, 185–194 (2005)
20. MacQueen, J.B.: Some methods for classification and analysis of multivariate observations. In: Proceeding of the Fifth Berkeley Symposium on Math., Stat. and Prob., pp. 281–296 (1967)
21. Nouira, K., Trabelsi, A.: Intelligent monitoring system for intensive care units. Journal of Medical Systems 36, 2309–2318 (2011)
22. Passerini, A., Lippi, M., Frasconi, P.: Predicting Metal-Binding Sites from Protein Sequence. Transactions on Computational Biology and Bioinformatics 9, 203–213 (2012)
23. Reslan, Z.A.: Clinical alarm management and noise reduction in hospitals. University of Connecticut, Storrs (2007)
24. Siebig, S., Kuhls, S., Imhoff, M., Langgartner, J., Reng, M., Scholmerich, J., Gather, U., Wrede, C.E.: Collection of annotated data in a clinical validation study for alarm algorithms in intensive care-a methodologic framework. Journal of Critical Care 25, 128–135 (2010)

25. Tong, T., Koller, D.: Support Vector Machine Active Learning with Applications to Text Classification. Journal of Machine Learning Research, 45–66 (2001)
26. Tsien, C.: Reducing False Alarms in the Intensive Care Unit: A Systematic Comparison of Four Algorithms. In: Proceedings of the AMIA Annual Fall Symposium. American Medical Informatics Association (1997)
27. Zhang, S.W., Pan, Q., Zhang, H.C., Zhang, Y.L., Wang, H.Y.: Classification of protein quaternary structure with support vector machine. Bioinformatics 19, 2390–2396 (2003)
28. Zhang, S.W., Vucetic, S.: Online Training on a Budget of Support Vector Machines Using Twin Prototypes. Statistical Analysis and Data Mining 3, 149–169 (2010)

Author Index