Advances in Intelligent Systems and Computing 314

Aleksander Zgrzywa
Kazimierz Choroś
Andrzej Siemiński *Editors*

New Research in Multimedia and Internet Systems

Springer

Advances in Intelligent Systems and Computing

Volume 314

Series editor

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

About this Series

The series "Advances in Intelligent Systems and Computing" contains publications on theory, applications, and design methods of Intelligent Systems and Intelligent Computing. Virtually all disciplines such as engineering, natural sciences, computer and information science, ICT, economics, business, e-commerce, environment, healthcare, life science are covered. The list of topics spans all the areas of modern intelligent systems and computing.

The publications within "Advances in Intelligent Systems and Computing" are primarily textbooks and proceedings of important conferences, symposia and congresses. They cover significant recent developments in the field, both of a foundational and applicable character. An important characteristic feature of the series is the short publication time and world-wide distribution. This permits a rapid and broad dissemination of research results.

More information about this series at http://www.springer.com/series/11156

Aleksander Zgrzywa · Kazimierz Choroś
Andrzej Siemiński
Editors

New Research in Multimedia and Internet Systems

Springer

Editors
Aleksander Zgrzywa
Institute of Informatics
Division of Information Systems
Wroclaw University of Technology
Wrocław
Poland

Andrzej Siemiński
Institute of Informatics
Division of Information Systems
Wroclaw University of Technology
Wrocław
Poland

Kazimierz Choroś
Institute of Informatics
Division of Information Systems
Wroclaw University of Technology
Wrocław
Poland

ISSN 2194-5357 ISSN 2194-5365 (electronic)
ISBN 978-3-319-10382-2 ISBN 978-3-319-10383-9 (eBook)
DOI 10.1007/978-3-319-10383-9

Library of Congress Control Number: 2014948346

Springer Cham Heidelberg New York Dordrecht London

Printed on acid-free paper

Springer is part of Springer Science+Business Media (www.springer.com)

Preface

Multimedia and Internet technologies are more and more in the hands of the people. This trend calls for monographs that describe both theoretical and practical aspects of research work on these areas. Both aspects are closely related. Looking back we see that many ideas such as e.g. speech, face, or handwriting recognition have for long been deemed not feasible. For several years they were mainly research projects at universities or in research facilities. Now they form part of our everyday life. The pace of progress is so fast that we have decided to assemble a volume that covers recent advances on these areas that we have witnessed in the last few years.

The covered area is diverse but so are our needs. Therefore, the monograph is a multidisciplinary work. Some of the proposed solutions are so advanced that they may be used in practice while others are concentrated upon the study of more basic properties of Internet and multimedia data processing. In all cases the studies are original and were not published anywhere else before.

The content of the book has been divided into four parts:
I Multimedia Information Technology
II Information System Specification
III Information System Applications
IV Web Systems and Network Technologies

Part I consists of 7 chapters. It starts with three chapters on picture processing. They describe a document image segmentation algorithm based on the analysis of the tiles, content based indexing and retrieval of visual data using the principal component analysis (PCA) applied to spatial representation of object location, and finally the fast encoding of huge 3D data sets in lossless PNG format. The next two chapters are on the video data processing. The 4th chapter examines the most frequent cases of false and miss detections in temporal segmentation of TV sports news videos. The segmentation is the first step of the process for identification of sports disciplines in video. In the fifth chapter the trajectory clustering is used to discover different motion patterns and recognize event occurrences in videos. The last two chapters study the possibilities of automated human behavior recognition. One of them focuses on the important, from a practical point of view, task of identifying

the call for help behavior whereas the other on the facial emotion recognition using a cascade of neural networks.

Part II has 6 chapters and has a more theoretical flavor. The first chapter of this part is a report on research on the prediction of real estate sales transactions. The method analyses a data stream applying ensembles of regression models. Six self-adapting genetic algorithms with varying mutation, crossover, and selection were developed and tested using real-world datasets. The selection of an appropriate method for the pre-processing of information is covered by the next chapter. It applies the information about the level of signal interference. Based on an illustrative example the next chapter describes one of the conceptual methods in data mining area that relies on the one-sided concept lattices belonging to approaches known as Formal Concept Analysis (FCA). This Part contains also a report on an attempt to represent a textual event as a mixture of semantic stereotypes and factual information. The semantic prototypes are specific for a given event and to define them generic elements are used. The Part concludes with a schema of a new approach to process knowledge resources for learning and testing. The idea is to use structured facts, used by Google for its Knowledge Graph, in an e-learning environment to automate some parts of the education process.

Part III presents a handful of applications. It consists of six chapters. The first of them describes an adaptive application with an attention-getting name of Power Chalk. It was designed to resolve an important limitation of current design methods: adaptability. In doing so it uses the intelligent agent paradigm in order to support scaffolding activities and problem solving. Agents are also present in the next paper. Their extremely simple form, the ants of the ACO, are used to solve the Travelling Salesmen Problem. A number of experiments clearly indicate that the so called Hyper-populated Ant Colonies could be successfully applied to solve both the static and the dynamic version of the TSP. Heurist methods are also used in the next chapter. This time they are applied to the recognition of texts with various background and foreground colors even when their luminosity values are equal. The proposed solution may be adapted into current OCR systems or work as a standalone preprocessing system. The last chapter is devoted to the translation of medical texts. Having in mind the popularity of international travel and how much we value our health the importance of the subject could not be understated. The research follows the statistical approach to translation and uses the EMEA parallel text corpora from the OPUS project. The next chapter is a descriptive and predictive analyses of dataset from Federal Aviation Administration (FAA) Accident/Incident Data System describing aviation accidents. Data mining enabled the authors to generate models that could be used as a basis for aviation warning system or as a supporting method for different processes related to the aviation industry. In the last chapter of the Part III the authors propose visual identity system for employees that utilizes a information rich generative logo and generates graphic elements of identification.

The last fourth Part is made up of five chapters. The first two of them deal with an analysis of Web Service Retrieval Methods and with the application of learning algorithms to the detection of changes in group of services in SOA system respectively. The research reported in the former encompasses the Latent

Semantic Indexing and modified term-document matrix. This allows the authors to store scores for different service components separately. In the latter one three learning algorithms: Kohonen network, emerging patterns, and k-means clustering were used to detect the anomalies in a special model of SOA system. The system was designed and implemented for an experimental purpose. Then, a method of positioning search results in music information retrieval systems is described. It is based on a modified version of the PageRank. The next chapter deals with the problem of the configuration of complex interactive environments which are made up of various types of sensors such as touch, depth, or RFID. Sensors are also present in the last chapter on multimedia communication in Wireless Multimedia Sensor Networks (WMSN). It presents the design of low complexity scheme for object identification using WMSN. Its notable feature is low-power processing requirements at the source mote while unloading the network at the same time.

We hope that we have achieved our goal of providing the researcher community with up to date account on work going on the diverse field of multimedia and Internet data processing. We will be also pleased if the book will attract even more scholars to work on the area and it will be a source of inspiration for the research community already working on the domain.

This would be our greatest award for our efforts.

Aleksander Zgrzywa
Kazimierz Choroś
Andrzej Siemiński

Contents

Part I: Multimedia Information Technology

Part III: Information Systems Applications

Part IV: Web Systems and Network Technologies

List of Contributors

Abdullah Al-Dhelaan
King Saud University
College of Computer and Information Sciences
Riyadh
Saudi Arabia
dhelaan@ksu.edu.s

Mohsin S Alhilal
King Saud University
College of Computer and Information Sciences
Riyadh
Saudi Arabia
mhilal@student.ksu.edu.sa

Jarosław Andrzejczak
Lodz University of Technology
Institute of Information Technology
ul. Wólczańska 215, 90-924 Łódź
Poland
jaroslaw.andrzejczak@p.lodz.pl

Jędrzej Anisiewicz
Aduma SA
ul. Duńska 9, 54-427 Wrocław
Poland
j.anisiewicz@aduma.pl

František Babič
Technical University of Kosice
Department of Cybernetics and Artificial Intelligence
Letńa 9, 04200 Košice, Slovakia
frantisek.babic@tuke.sk

Ilona Bluemke
Warsaw University of Technology
Institute of Computer Science,
Nowowiejska 15/19, 00-665 Warszawa
Poland
i.bluemke@ii.pw.edu.pl

Peter Butka
Technical University of Košice
Faculty of Electrical Engineering and Informatics, Department of Cybernetics and Artifcial Intelligence
Letńa 9, 04200 Košice, Slovakia
peter.butka@tuke.sk

Kazimierz Choroś
Wrocław University of Technology
Institute of Informatics
Wybrzeże Wyspiańskiego 27,
50-370 Wrocław
Poland
kazimierz.choros@pwr.edu.pl

Adam Czyszczoń
Wrocław University of Technology
Institute of Informatics
Wybrzeże Wyspiańskiego 27,
50-370 Wrocław
Poland
adam.czyszczon@pwr.edu.pl

Daniel Dworak
Lodz University of Technology
Institute of Information Technology
ul. Żeromskiego 116, 90-924 Łódź
Poland
daniel.dworak@p.lodz.pl

Margarita Esponda-Argüero
Freie Universitt Berlin
Department of Mathematics and
Computer Science
Takustrasse 9, 14195 Berlin
Germany
esponda@inf.fu-berlin.de

Paweł Gardziński
Poznań University of Technology
Chair of Multimedia
Telecommunications and
Microelectronics
ul. Polanka 3, 60-965 Poznań
Poland
pgardzinski@multimedia.
edu.pl

Kinga Glinka
Lodz University of Technology
Institute of Information Technology
ul. Wólczańska 215, 90-924 Łódź
Poland
800559@edu.p.lodz.pl

Radosław Hofman
Smart Monitor sp. z o.o.,
ul. Niemierzyńska 17, 71-441 Szczecin
Poland
radekh@smartmonitor.pl

Mihai Bogdan Ilie
University of Galati “Dunarea de Jos"
Faculty of Automatic Control,
Computers, Electrical and
Electronics Engineering
Stiintei Street 2nd, 800146 Galati
Romania
mihai.ilie@ugal.ro

Bartosz Jakubicki
The Eugeniusz Geppert Academy of Art
and Design
ul. Plac Polski 3/4, 50-156 Wrocław
Poland
bjak@asp.wroc.pl

Tatiana Jaworska
Polish Academy of Sciences
Systems Research Institute
ul. Newelska 6, 01-447 Warszawa
Poland
tatiana.jaworska@ibspan.
waw.pl

Łukasz Kamiński
Poznań University of Technology
Chair of Multimedia
Telecommunications and
Microelectronics
ul. Polanka 3, 60-965 Poznań, Poland
lkaminski@multimedia.edu.pl

Marek Kopel
Wrocław University of Technology
Institute of Informatics
Wybrzeże Wyspiańskiego 27,
50-370 Wrocław, Poland
marek.kopel@pwr.edu.pl

Wojciech Korczyński
AGH University of Science and
Technology
Departmet of Informatics
al. Mickiewicza 30, 30-962, Kraków
Poland
wojciech.korczynski@agh.
edu.pl

Michał Korzycki
AGH University of Science and Technology
Departmet of Informatics
al. Mickiewicza 30, 30-962, Kraków
Poland
korzycki@agh.edu.pl

Krzysztof Kowalak
Poznań University of Technology
Chair of Multimedia Telecommunications and Microelectronics
ul. Polanka 3, 60-965 Poznań
Poland
fkkowalak@multimedia.edu.pl

Janusz Paweł Kowalski
Pomeranian Medical University
Faculty of Medicine
ul. Rybacka 1, 70-204 Szczecin
Poland

Elżbieta Kukla
Wrocław University of Technology
Institute of Informatics
Wybrzeże Wyspiańskiego 27,
50-370 Wrocław
Poland
elzbieta.kukla@pwr.edu.pl

Tadeusz Lasota
Wrocław University of Environmental and Life Sciences, Department of Spatial Management
ul. C.K. Norwida 25, 50-375 Wrocław
Poland
tadeusz.lasota@up.wroc.pl

Alexandra Lukáčová
Technical University of Kosice
Department of Cybernetics and Artificial Intelligence
Letńa 9, 04200 Košice, Slovakia
alexandra.lukacva@tuke.sk

Sławomir Maćkowiak
Poznań University of Technology
Chair of Multimedia Telecommunications and Microelectronics
ul. Polanka 3, 60-965 Poznań
Poland
smackg@multimedia.edu.pl

Krzysztof Marasek
Polish Japanese Institute of Information Technology, Department of Multimedia
ul. Koszykowa 86, 02-008 Warszawa
Poland
kmarasek@pjwstk.edu.pl

Zygmunt Mazur
Wrocław University of Technology
Institute of Informatics
Wybrzeże Wyspiańskiego 27,
50-370 Wrocław
Poland
zygmunt.mazur@pwr.edu.pl

Grzegorz Mikolajczak
West Pomeranian University of Technology
Faculty of Electrical Engineering
ul. Sikorskiego 37, 70-313 Szczecin
Poland
gmikolajczak@zut.edu.pl

Adam Musiał
Lodz University of Technology
Institute of Information Technology
ul. Wólczańska 215, 90-924 Łódź
Poland
800334@edu.p.lodz.pl

Paweł Nowak
Wrocław University of Technology
Institute of Informatics
Wybrzeże Wyspiańskiego 27,
50-370 Wrocław
Poland
pawel.nowak@pwr.edu.pl

Ján Paralič
Technical University of Kosice
Department of Cybernetics and Artificial Intelligence
Letńa 9, 04200 Košice, Slovakia
jan.paralic@tuke.sk

Jakub Peksinski
West Pomeranian University of Technology
Faculty of Electrical Engineering
ul. Sikorskiego 37, 70-313 Szczecin
Poland
jpeksinski@zut.edu.pl

Maria Pietruszka
Lodz University of Technology
Institute of Information Technology
ul. Żeromskiego 116, 90-924 Łódź
Poland
maria.pietruszka@p.lodz.pl

Jozef Pócs
Slovak Academy of Sciences
Mathematical Institute
Gresakova 6, 040 01 Kosice
Slovakia
pocs@saske.sk

Jana Pócsová
Technical University of Kosice
BERG Faculty, Institute of Control and Informatization of Production Processes
Bozeny Nemcovej 3, 043 84 Kosice
Slovakia
jana.pocsova@tuke.sk

Raúl Rojas
Freie Universitt Berlin
Department of Mathematics and Computer Science
Takustrasse 9, 14195 Berlin
Germany
raul.rojas@fu-berlin.de

Dan-El Neil Vila Rosado
Freie Universitt Berlin
Department of Mathematics and Computer Science
Takustrasse 9, 14195 Berlin
Germany
vila80@inf.fu-berlin.de

Andrzej Siemiński
Wrocław University of Technology
Institute of Informatics
Wybrzeże Wyspiańskiego 27,
50-370 Wrocław, Poland
andrzej.sieminski@pwr.edu.pl

Magdalena Smętek
Wrocław University of Technology
Institute of Informatics
Wybrzeże Wyspiańskiego 27,
50-370 Wrocław, Poland
Magdalena.smetek@pwr.edu.pl

Janusz Sobecki
Wrocław University of Technology
Institute of Informatics
Wybrzeże Wyspiańskiego 27,
50-370 Wrocław, Poland
janusz.sobecki@pwr.edu.pl

Adel Soudani
King Saud University
College of Computer and Information Sciences
Riyadh
Saudi Arabia
asoudani@ksu.edu.sa

Zhanhu Sun
East China Normal University
Department of Computer Science and Technology
500 Dongchuan Rd., Shanghai 200241
China
zhhsun@ica.stc.sh.cn

Marcin Tarka
Warsaw University of Technology
Institute of Computer Science
ul. Nowowiejska 15/19, 00-665 Warsaw
Poland
m.tarka@ii.pw.edu.pl

Bogdan Trawiński
Wrocław University of Technology
Institute of Informatics
Wybrzeże Wyspiańskiego 27,
50-370 Wrocław
Poland
ogdan.trawinski@pwr.edu.pl

Grzegorz Trawiński
Wrocław University of Technology
Faculty of Electronics
Wybrzeże Wyspiańskiego 27,
50-370 Wrocław
Poland
grzegorz.trawinski@pwr.edu.pl

Feng Wang
East China Normal University
Department of Computer Science and Technology
500 Dongchuan Rd., Shanghai 200241
China
fwang@cs.ecnu.edu.cn

Zbigniew Wantuła
Aduma SA
ul. Duńska 9, 54-427 Wrocław
Poland
z.wantula@aduma.pl

Konrad Wiklak
Wrocław University of Technology
Institute of Informatics
Wybrzeże Wyspiańskiego 27,
50-370 Wrocław
Poland
konrad.wiklak@gmail.com

Krzysztof Wołk
Polish Japanese Institute of Information Technology, Department of Multimedia
ul. Koszykowa 86, 02-008 Warszawa
Poland
kwolk@pjwstk.edu.pl

Aleksander Zgrzywa
Wrocław University of Technology
Institute of Informatics
Wybrzeże Wyspiańskiego 27,
50-370 Wrocław
Poland
aleksander.zgrzywa@pwr.edu.pl

Part I
Multimedia Information Technology

Chapter 1
Document Image Segmentation through Clustering and Connectivity Analysis

Mihai Bogdan Ilie

Abstract. This chapter presents a new document image segmentation algorithm, called Cluster Variance Segmentation (CVSEG). The method is based on the analysis of the tiles suspected to be part of an image and filtering them subsequently. In the end, the results are enhanced through a reconstruction stage. I present the design of the algorithm as well as the test results on various document images. The experiments validate the efficacy and efficiency of the proposed approach when compared with other algorithms.

1.1 Introduction

In the context of the current stage of document generation and recording, there is very much interest shown in the Document Analysis and Retrieval (DAR) field, which is a viable solution to automatically process and classify the documents in a specific area. There are many problems that have been addressed by DAR researchers all over the world, like:

- extracting the text from documents written in different languages with different characters [1];
- document sorting according to keywords [2];
- document sorting according to the layout;
- web crawling [3];
- automatically processing official forms [4];
- signature and stamp detection [5];

Mihai Bogdan Ilie
"Dunarea de Jos" University of Galati,
Faculty of Automatic Control, Computers,
Electrical and Electronics Engineering,
Stiintei Street 2nd, Galati, Romania
e-mail: mihai.ilie@ugal.ro

A. Zgrzywa et al. (eds.), *New Research in Multimedia and Internet Systems*,
Advances in Intelligent Systems and Computing 314, DOI: 10.1007/978-3-319-10383-9_1

- distinguishing between two different languages [6];
- many others.

The sub-problems encountered in the document processing area are complex and are originated in different areas:

- scanning conditions;
- noise determined by the page curvature;
- poor page illumination;
- degraded physical support;
- stroke size;
- different languages and characters etc.

The implementations vary from solving the basic problems of document processing up to complete, sophisticated software solutions, completely connected. A common approach (especially in the areas which require a classifier, but not only) is to use artificial intelligence techniques. Among these, the most common during the classification stage are the neural networks and the support vector machines. Besides these, there are implementations that make use of genetic algorithms [7], unsupervised learning [8], swarm intelligence, Markov random fields [9], fuzzy modules [10] or self organizing feature maps [11].

One of the problems addressed by the DAR area is the automated extraction of images from documents. Next, this information can be used for multiple purposes, like plagiarism detection, document classification according to the image content or according to the logo.

1.2 Related Work

The main purpose of the DAR area is to recognize the text and the graphical elements in a document scan, as a human would. The DAR field includes multiple research directions, like:

- binarization,
- noise reduction,
- segmentation,
- OCR,
- skew estimation,
- many others.

From all of the above I am mostly interested in the segmentation area, especially in image segmentation.

There are many possibilities to classify the current DAR approaches but one of the most complete studies establishes the below algorithm taxonomy [12]:

- based on image characteristics – extracting local and global descriptors for colour, shape, texture, gradient etc.,
- based on the physical structure – establishing the document geometrical hierarchy,

- based on logical characteristics – establishing the document logical hierarchy,
- based on text characteristics – extracting keywords, based on an OCR algorithm.

In what regards the classification strategy, the approaches can be classified as below:

- bottom-up – which start by analysing pixels, regions or connected tiles; the resulted objects are then merged and classified as document areas;
- top-down – which start analysing the document from the encompassing scan image and then split it in unit regions.

Regardless of the techniques described above, the authors agree on the below stages involved in the process of obtaining the desired results:

- binarization,
- noise reduction,
- segmentation,
- thinning,
- chain coding and vectorization.

In the document segmentation area, most of the researchers target text segmentation in all its variations - block, paragraph, line, word and character segmentation. There are not many image segmentation algorithms; among them, probably the best known is the one implemented by Bloomberg [13], based on eroding the image in order to eliminate the text and then dilating it in order to enhance the image features. Subsequently, this algorithm got improved by Syed Saqib Bukharia [14], who introduced some additional steps, refining the two stages.

I propose a bottom-up image segmentation approach, based on image characteristics, targeting the document image segmentation.

1.3 New Approach

The algorithm (CVSEG) is based on decomposing the document in tiles, clustering and filtering them based on their inner variance and connectivity.

The image segmentation process is split into several phases. First of all, the algorithm assumes that the image has been previously binarized. In order to achieve its goal, the algorithm extracts the pixel grid, splits the image in tiles of a certain size and goes through the below stages:

1) text filtering, in order to facilitate processing the remaining areas. Currently I am using a simple algorithm, based on XY axis projection [15]. I looked for a simple method which would be computational fast and that would manage to eliminate the text up to some extent. This step works well on documents which are correctly aligned with the axis; the results are shown in the image below (Fig. 1.1). However, if the text is not completely filtered, it will be excluded from the final result due to its poor variance score.

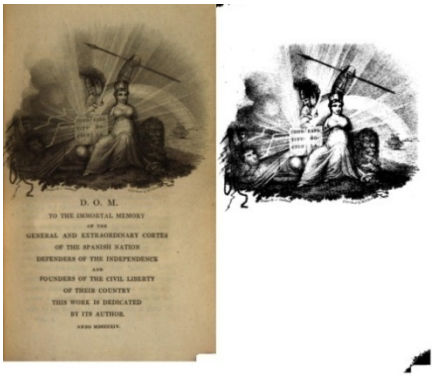


Fig. 1.1 Text filtering

2) calculating the average pixel intensity for each tile.

$$G_L = \frac{1}{w*h}\sum_{i,j} p_{i,j}\,, the\ average\ gray\ level \tag{1.1}$$

$$\Delta_G = \frac{1}{w*h}\sum_{i,j} |p_{i,j} - G_L|, w\ and\ h\ represent\ the\ tile\ width\ and\ height$$

$$\Delta_G = \frac{1}{w*h}\sum_{i,j} |p_{i,j} - G_L|, w\ and\ h\ represent\ the\ tile\ width\ and\ height$$

3) calculating the average variance in each tile. Based on this score, I am establishing whether a certain tile may be part of an image or not ($\Delta_G > T_G$ and $\Delta_C > T_C$). The decision criteria is that in a particular tile, an image tends to be more uniform than the text. The thresholds have been set through experiments at $T_G = 0.1$ and $T_C = 0.37$.

$$C_L(p_1, p_2) = \begin{cases} 0, if\ p1 = p2 \\ 1, if\ p1! = p2 \end{cases} \tag{1.2}$$

$$\Delta_C = \frac{1}{w*h}\sum_{i,j}\frac{1}{N_{i,j}}\sum_{m,n} C_L\left(p_{i,j}, p_{m,n}\right), m! = i, n! = j$$

$$N_{i,j} - \ total\ neighbours\ of\ the\ p_{i,j}\ pixel$$

4) eliminating singular tiles, which are not connected to any other validated sub-windows. At this stage I am calculating a score based on the neighbours and filtering out the isolated tiles. Generally, at this step I am discarding the tiles which include noise or any remaining text areas. The results are shown in the image below (Fig. 1.2).

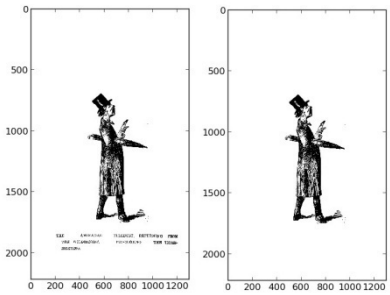


Fig. 1.2 Filtering singular tiles

5) on the resulted tile set I am applying a K-Means clustering algorithm, which uses as a decision metric the Euclidean distance between the elements. At this step I am building the tile sets which represent the images from the document, or parts of a larger image.
6) for each of the above clusters I am computing a score based on their scarcity and connectivity coefficients. The thresholds for these 2 scores have been determined experimentally as T_{sparse}=0.33 and T_{con}=0.5.

$$sparsity_{\mathrm{k}} = \frac{\mathrm{w}*\mathrm{h}*\mathrm{S_k}}{\mathrm{A_k}} \tag{3.3}$$

w and h represent the tile size,
S_k represents the number of tiles in the k cluster,
A_k represents the smallest rectangular window which includes all the tiles

$$conn_k = \frac{K_c}{S_k} \tag{3.4}$$

K_c represents the tile count with at least 2 valid neighbours,
S_k represents the number of tiles in the k cluster

7) the clusters are then filtered in order to eliminate tiles containing text areas with different fonts, affected by noise/poor illumination or by page curvature. The example in the figure below (Fig. 1.3) shows the effect of cluster filtering.
8) the resulting tiles are then merged and exposed to a reconstruction stage, which adds connected pixels, up to the distance D, where D is the diagonal of the tile. Sometimes, the image boundaries may be mistaken as text areas, which means that the final result may not include them. In order to avoid this, I am adding the pixels which are connected to the ones identified as being part of the image.

The algorithm's logical schema is presented in the below figure (Fig. 1.4).

Fig. 1.3 Filtering clusters

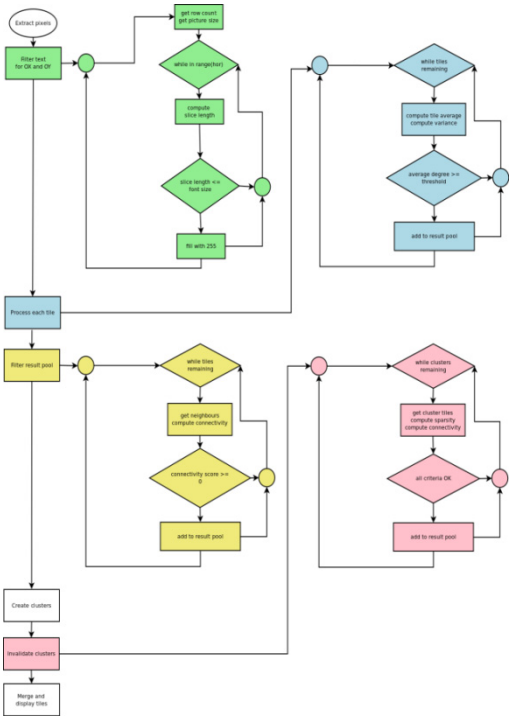


Fig. 1.4 CVSEG logical schema

1.4 Experimental Setup

The experiments have been conducted on a set of 1380 images, obtained from 2 sources:

- scans of old, degraded documents, used as a benchmark in the ICDAR 2007 conference [16];

- high quality copies, containing mostly manuals and documentation for the Ubuntu 12.04 operating system. In order to be able to use them, I have previously converted them from the pdf format to the jpeg one.

Since the CVSEG algorithm requires a pre-binarized document, I have used the below binarization algorithms:

- average binarization, obtained by splitting the image in multiple tiles and applying a normalized threshold on each of them;
- Sauvolabinarization, developed by [17];
- NLBIN, a non linearbinarization algorithm, recommended by the authors of the Ocropus library [18], the default document processing tool used on the Android operating system.

The chosen programming language was python, especially due to the facilities provided for the clustering algorithms and for the matrix handling API. I have used a 32 bit Ubuntu 12.04 operating system, running on a machine based on an Intel I5 dual core processor and 4GB of RAM. Due to the software's modular structure I could easily switch between different tile sizes and binarization algorithms.

The test images have been tagged with bounding boxes, saved in corresponding text files, containing the upper left and lower right coordinates.

The best results have been obtained when using a tile size of 20 pixels. If the tile size is increased too much, it is hard to distinguish between text and images based on the variance, as the analyzed surface would be too big and this indicator will always be high - this translates into disregarding most of the sub-windows. Choosing a tile which is too small translates into a very small variance score, which basically means that the algorithm will validate much more sub-windows.

The results are described in the Table 1.1. The total computing time is basically the sum of the CVSEG algorithm and the binarization execution times.

Table 1.1 CVSEG segmentation results

Segmentation / Binarization	Accuracy	Total computing time
CVSEG avebin	81.2 %	4.614 s
CVSEG Sauvola	84.0 %	14.500 s
CVSEG NLBIN	84.1 %	28.762 s

The figure below (Fig. 1.5) shows an example of how the algorithm behaved on the sample images, originating in both areas - poor quality scans and high quality images.

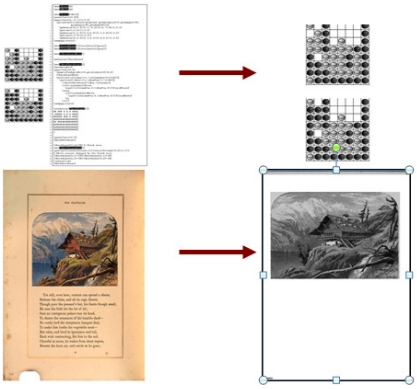


Fig. 1.5 CVSEG results

I have used for the test environment Bloomberg's document image segmentation algorithm, implemented by the author in the Leptonica library [19], available in the Ubuntu 12.04 repositories. The library contains a large set of utilities and is used by many DAR applications and OCR engines, including tesseract [20], an engine maintained and sponsored currently by Google. As this implementation requires a specific type of input images, I have used a GIMP CLI script in order to convert them to the indexed mode.

Besides the above configuration, the software and hardware environment has been the same as the one used for testing the CVSEG algorithm.

The results are described in the Table 1.2.

Table 1.2 Bloomberg segmentation results

Segmentation / Binarization	Accuracy	Total computing time
Bloomberg GIMP	72.2 %	5.205 s
Bloomberg GIMP ave bin	72.3 %	8.012 s
Bloomberg GIMP Sauvola	74.1 %	13.205 s
Bloomberg GIMP NLBIN	74.5 %	15.429 s

1.5 Conclusions

The CVSEG algorithm obtained results up to 9% better than the Bloomberg one. The most frequent errors in the Bloomberg algorithm have been:

- no image in the result - this was caused by documents containing images with very thin lines; the images have been filtered out during the eroding stage;
- the complete document in its negative form is included provided a result - this was caused by documents affected by noise due to the page transparency;
- incomplete results - usually caused by low contrast or by images being tightly connected to the text blocks (Fig. 1.6).

Fig. 1.6 Bloomberg incomplete segmentation

In what regards the CVSEG algorithm, most of the problems have been caused by the below:

- the result included additional areas which were not part of the image - this was caused by the text filtering stage, malfunctioning for text written in different font types (Fig. 1.7);

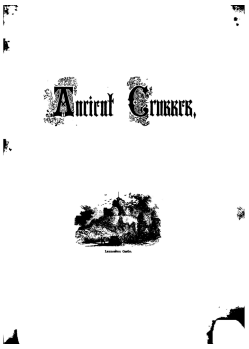


Fig. 1.7 CVSEG erroneous segmentation

- the result includes areas adjacent to the image - this is caused by two factors: the text filtering algorithm and the clustering algorithm have not succeeded to eliminate the text located next to the image (Fig. 1.8);

Fig. 1.8 CVSEG erroneous segmentation – clustering

- when using the Sauvola algorithm, the binarized output included an image rotated at a certain angle, which affects to some extent the segmentation algorithm, at the text filtering stage (Fig. 1.9);

Fig. 1.9 Sauvolabinarization output

- the result included black areas - this was caused by the binarization algorithms which failed to filter out the noise, especially in the images scanned under poor illumination conditions (Fig. 1.10 Binarization affected by noise;
- the high quality documents have been mostly segmented correctly. The only problems I have met have been related to incomplete diagrams and schemes, especially when using thin lines, as can be seen below.

The CVSEG execution times are higher than the Bloomberg ones, mostly due to the chosen programming language, which introduces a significant overhead when compared to native compiled C code.

We can notice that the binarization algorithms do not affect with much the accuracy. There is also a large increase in the computing time when switching to the NLBIN binarization (two times bigger than the Sauvola measurements), which produces just 0.1% accuracy improvement.

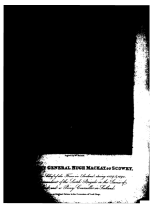


Fig. 1.10 Binarization affected by noise

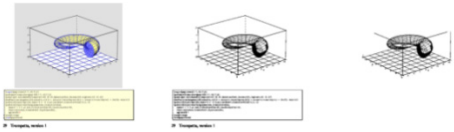


Fig. 1.11 CVSEG affected by thin lines

1.6 Future Research

In the future I would like to continue the research conducted so far. The CVSEG performances can be improved by refining the text filtering algorithm and the clustering algorithm.

Also, I want to add an OCR module based on a neural network combined with a variable size sliding window technique, in order to achieve 2 goals:

- detecting the character scale and automatically establishing the tile size;
- ensuring the absence of the text in the final result.

Another problem I have ran into is the result comparison stage, due to the lack of document image segmentation algorithms. Therefore, I intend to develop a classifier which assigns a specific score to any segmentation algorithm (image or text), according to its results. This will allow me to compare my results with text segmentation algorithms as well.

Acknowledgements. The author would like to thank the Project SOP HRD /107/1.5/S/76822 – TOP ACADEMIC, of University "Dunarea de Jos" of Galati, Romania.

References

1. Pujari, A.K., Dhanunjaya Naidu, C., Sreenivasa Rao, M., Jinaga, B.C.: An intelligent character recognizer for Telugu scripts using multiresolution analysis and associative memory. Image and Vision Computing 22(14), 1221–1227 (2004)
2. Cai, K., Bu, J., Chen, C., Huang, P.: An automatic approach for efficient text segmentation. In: Gabrys, B., Howlett, R.J., Jain, L.C. (eds.) KES 2006. LNCS (LNAI), vol. 4251, pp. 417–424. Springer, Heidelberg (2006)
3. Yang, J., Kang, J., Choi, J.: A focused crawler with document segmentation. In: Gallagher, M., Hogan, J.P., Maire, F. (eds.) IDEAL 2005. LNCS, vol. 3578, pp. 94–101. Springer, Heidelberg (2005)
4. Zhanga, X., Lyu, M.R., Dai, G.-Z.: Extraction and segmentation of tables from Chinese ink documents based on a matrix model. Pattern Recognition 40(7), 1855–1867 (2007)
5. Roy, P.P., Pal, U., Lladós, J.: Document seal detection using GHT and character proximity graphs. Pattern Recognition 44(6), 1282–1295 (2011)
6. Xia, Y., Xiao, B.H., Wang, C.H., Li, Y.D.: Segmentation of mixed Chinese/English documents based on Chinese Radicals recognition and complexity analysis in local segment pattern. In: Huang, D.-S., Li, K., Irwin, G.W. (eds.) Intelligent Computing in Signal Processing and Pattern Recognition. LNCIS, vol. 345, pp. 497–506. Springer, Heidelberg (2006)

7. Sas, J., Markowska-Kaczmar, U.: Similarity-based training set acquisition for continuous handwriting recognition. Information Sciences 191, 226–244 (2012)
8. Tsai, C.M.: Intelligent region-based thresholding for color document images with highlighted regions. Pattern Recognition 45(4), 1341–1362 (2012)
9. Tonazzini, A., Vezzosi, S., Bedini, L.: Analysis and recognition of highly degraded printed characters. Document Analysis and Recognition 6(4), 236–247 (2003)
10. Fonseca, M.J., Pimentel, C., Jorge, J.A.: CALI: An online scribble recognizer for calligraphic interfaces. In: AAAI Spring Symposium on Sketch Understanding, pp. 51–58 (2002)
11. Papamarkos, N.: A neuro-fuzzy technique for document binarisation. Neural Computing & Applications 12(3-4), 190–199 (2003)
12. Chen, N., Blostein, D.: A survey of document image classification: problem statement, classifier architecture and performance evaluation. International Journal of Document Analysis and Recognition (IJDAR) 10(1), 1–16 (2007)
13. Bloomberg, D.S.: Multiresolution morphological approach to document image analysis. In: Proc. of the International Conference on Document Analysis and Recognition, Saint-Malo, France (1991)
14. Bukhari, S.S., Shafait, F., Breuel, T.M.: Improved document image segmentation algorithm using multiresolution morphology. In: IS&T/SPIE Electronic Imaging, pp. 78740D-78740D. International Society for Optics and Photonics (2011)
15. Ha, J., Haralick, R., Phillips, I.T.: Recursive XY cut using bounding boxes of connected components. In: Proceedings of the Third International Conference on Document Analysis and Recognition, vol. 2, pp. 952–955. IEEE (1995)
16. Antonacopoulos, A., Bridson, D., Papadopoulos, C.: Page Segmentation Competition. In: ICDAR 2007 (2007), http://www.primaresearch.org/ICDAR2007_competition/
17. Sauvola, J., Seppanen, T., Haapakoski, S., Pietikainen, M.: Adaptive document binarization. In: Proceedings of the Fourth International Conference on Document Analysis and Recognition, vol. 1, pp. 147–152. IEEE (1997)
18. Google. Ocropus, http://code.google.com/p/ocropus (April 21, 2013)
19. Bloomberg, D.S.: libleptonica, http://www.ubuntuupdates.org/package/core/precise/universe/base/libleptonica (March 6, 2012)
20. Google, Inc. Tesseract (2013), http://en.wikipedia.org/wiki/Tesseract_%28software%29

Chapter 2
Fast Encoding of Huge 3D Data Sets in Lossless PNG Format

Daniel Dworak and Maria Pietruszka

Abstract. This chapter focuses on manners of filling the gaps in existing standards that are used in tridimensional web technologies. One of the aspects is coding and uploading the 3D huge data sets using alternative formats with an emphasis on architectural models. We have developed a technique for storing data about 3D geometry saved in PNG file, because of small size and lossless compression.

2.1 Introduction

Now tridimensional graphics is a popular and attractive form of data's visualization on the Web because of their possible interactive form. There are also other multimedia forms like computer animations or rendered pictures, although they are not as interactive as they are supposed to be. Due to development of standards that we use on the Internet, there appeared some new technologies and capabilities of visualization of 3D data (for example WebGL, Stage 3D or Away3D). Most of them offer interactive and easy to use virtual travel in virtual World.

On the other hand, accumulation of many models of buildings, trees etc. causes the increase of vertices and faces quickly. Most of current technologies for tridimensional graphics are limited to draw only 65,536 vertices per frame. The technique of splitting models with huge vertices number is used frequently to reduce file's size of each one and then portaling method [5] can be applied. This method loads models, with smaller number of vertices than the whole model, every time it is required, for example when a user is passing through the doors to another roomThere is also the

Daniel Dworak
Instytut Informatyki, Technical University of Lodz, Poland
Center for Media and Interactivity, Justus Liebig University Giessen, Germany
e-mail: 150859@edu.p.lodz.pl

Maria Pietruszka
Instytut Informatyki, Technical University of Lodz, Poland
e-mail: maria.pietruszka@p.lodz.pl

A. Zgrzywa et al. (eds.), *New Research in Multimedia and Internet Systems*,
Advances in Intelligent Systems and Computing 314, DOI: 10.1007/978-3-319-10383-9_2

LOD method (Level of Detail) [7] which requires few representations of the same object with different number of vertices. If user is far away from this model, algorithm loads the object with the smallest number of vertices. If distance is decreasing, there should be loaded a model with more vertices.

There are also problems with diversification of users' computers, platforms, browsers and computing power. Since there are special formats for transmission of audio (MP3), video (H.264) or images (JPEG) optimized for Web appliances, but untill that time were not created formats for 3D graphics transmission. Lack of standards for 3D data transmission caused a development of new format glTF (graphics library Transmission Format) by Khronos Group. Like author noticed [8], COLLADA is not a transport format for the Web. glTF format uses converted COLLADA assets to JSON format for easier use in WebGL and smaller files for transfer over the Internet.

The idea of encoding and decoding 3D data from two-dimensional graphic's files for reducing time of transfer files via Internet, what has been described in [4]. However, there was an attempt to store the data in the formats with lossy compression, what can produce a noise in decoded data. What is more, the authors were trying to save pre-converted data for methods like Sequential Image Geometry (SIG) and Progressive Binary Geometry (PBG).

The combination of those solutions is promising in order to increase the performance of the whole Web project. In fact, those techniques do not reduce size or complexity of files with a 3D geometry. This chapter focuses on those problems and proposes a way of optimization for storing 3D geometry.

2.2 File Formats for Storing 3D Data

The process of creation a tridimensional geometry is not complicated, but requires many informations about stored data. Basically, there are three types of information: vertices, faces and normals. Single vertex is a point in 3D space, therefore, it demands three numbers with x, y and z position. A face, according to adopted type, is a three (a triangle) or four (a quad) set of vertices' numbers which creates single surface. A normal is a vector that is perpendicular to the tangent plane to that surface at some point (Fig. 2.1). Normals determine a surface's orientation for shading and lightning this surface. What is more, textures require texture coordinates (called UV's) – those are pairs of numbers from [0.0;1.0] interval.

2.2.1 Popular File Formats for Storing 3D Data

The popular software for 3D modelling offers many file formats for storing 3D data. A part of them allows to save even animations of geometry or continuous surfaces. Very popular is an OBJ format – cross platform, open, human readable format, which can store all the informations about geometry and materials. The OBJ files store data as follows:

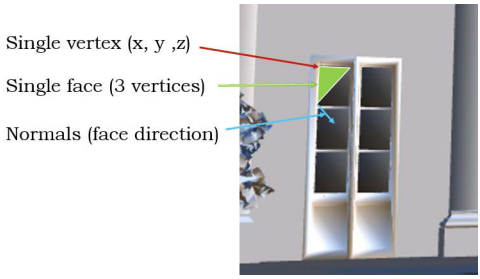


Fig. 2.1 Elements of geometry – vertices, faces and normals

1. Vertices (x, y, z):
   v -5.0000 0.0000 5.0000
   v -5.0000 0.0000 -5.0000
   v 5.0000 0.0000 -5.0000
   ...
   v -5.0000 10.0000 -5.0000

2. Normals (x, y, z):
   vn 0.0000 -1.0000 -0.0000
   vn 0.0000 1.0000 -0.0000
   vn 0.0000 0.0000 1.0000
   ...
   vn -1.0000 0.0000 -0.0000

3. Texture coordinates [u, v]:
   vt 1.0000 0.0000 0.0000
   vt 1.0000 1.0000 0.0000
   vt 0.0000 1.0000 0.0000
   ...
   vt 0.0000 0.0000 0.0000

4. Face definitions:
   f 1/1/1 2/2/1 3/3/1
   f 3/3/1 4/4/1 1/1/1
   f 2/4/6 1/1/6 5/2/6
   ...
   f 5/2/6 8/3/6 2/4/6

Faces are given in a form of lists of vertex, texture and normal indices. A triangle face is defined by three vertices, texture and normals indices; but a quad face is defined by four indices. According to that, OBJ files also support free form curved surface objects like the NURBS surfaces, what can reduce a file's size, but OBJ file cannot save data about hierarchy and animations.

More advanced is a COLLADA format (.DAE) which, besides geometry, allows to store data about shaders, effects, physics, animation, kinematics and even multiple versions' representations of the same asset. COLLADA file is based on XML schema:

1. Vertices:
   $< float\_arrayid = "Box001 - POSITION - array" count = "24" >$
   $-5.000000\ -5.000000\ 0.000000$
   $5.000000\ -5.000000\ 0.000000$
   $-5.000000\ 5.000000\ 0.000000$
   ...
   5.000000 5.000000 10.000000
   $< /float\_array >$
2. Normals:
   $< float\_arrayid = "Box001 - Normal0 - array" count = "108" >$
   $0.000000\ 0.000000\ -1.000000$
   $0.000000\ 0.000000\ -1.000000$
   $0.000000\ 0.000000\ -1.000000$

...
$-1.0000000.0000000.000000$
$</float\_array>$

3. Texture coordinates:
$<float\_arrayid="Box001-UV0-array"count="48">$
1.0000000.000000
0.0000000.000000
1.0000001.000000
...
1.0000001.000000
$</float\_array>$
4. Faces: $<trianglescount="12">$ ... $<p>$ 1 0 1 0 2 3 0 3 0 2 4 ... 21 $</p><$ $/triangles>$

Another format for 3D data storing is FBX by Autodesk, which is free, platform-independent. The data can be saved in binary or in text based version. Many applications, among others 3ds Max, Form-Z, Blender, Bryce, SketchUP, can save 3D models to a 3DS format which stores data about geometry, lights, materials, cameras, hierarchic objects and animations, but this format is limited to only 65,536 vertices per object.

There is also a VRML (.wrl) format, which has been created as the first one for storing tridimensional models optimized for the Web. The VRML is based on a scene graph, which saves informations in the nodes about geometry, transformations, materials, properties, etc. The geometry can be identified as a mesh or a parametric surface. The information about animations and interactions with the user might be triggered by external events. The VRML scene is saved in a text file with WRL extension:

```
DEF Box001-3 Transform {
translation 0 0 0
    children [
        Transform {
            translation 0 5 0
                children [
                    Shape {
                        appearance Appearance {
                            material Material {
                                diffuseColor 0.8824 0.3451 0.7804
                            }
                        }
                        geometry Box { size 10 10 10 }
                }
            ] } ]
        }
```

For testing purposes, we have created a model with 1.5M and 76,625 vertices to compare sizes of files saved in different formats Table 2.2.

Our tests showed how every format deals with huge data sets (Fig. 2.2). The Table 2.2 presented that FBX file is the smallest one, what can be justified by the

Table 2.1 Size comparison for different file formats with 1.5M and 76,625 vertices geometry

	DAE	FBX	OBJ	WRL
1.5M vertices	480MB	41.3MB	267MB	176MB
76,625 vertices	19.80MB	3.10MB	6.95MB	8.28MB

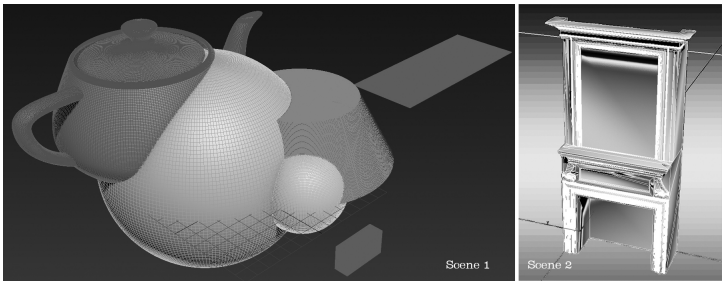


Fig. 2.2 Scene 1 – testing model with 1.5M vertices. Scene 2 – fireplace testing model with 76,625 vertices.

fact that this format can save parameric surfaces (like sphere, box or plane) without triangulation and is saved binary. What is more, the WRL gives also good results and files are smaller than OBJ, but only when objects have parametric surfaces. We have also experimented with architectural model of fireplace with 76,625 vertices (Fig. 2.2), created in Form-Z software, what has been showed on Table 2.2 in a third row. There is a change with OBJ and WRL files, what can be explained by fact, that the model is not a parametric object, so WRL file is bigger than OBJ.

2.2.2 File Formats for the Web

There are formats that are optimized and supposed to be more efficient in a Web appliance. Many Internet technologies supports COLLADA format because of variety of information that can be stored in one file. COLLADA is also supported by various software like the Maya, the 3ds Max, the SketchUP or the Blender 3D. This format can be sufficient for the models with small number of vertices, but according to our test model, .DAE was the biggest one with 480MB file's size.

JSON format (JavaScript Object Notation) is an open standard format with user-friendly notation (e.g. readable text) and is used to transmit data from server to client easily, because of it's specific way of saving geometry. It allows to store huge data sets like vertices, normals, texture coordinates, faces and textures' names. The size of JSON file is similar to OBJ one or in many cases even smaller. Although the Web technologies decode JSON files faster than OBJ. JSON file was 6.47MB for our testing model of fireplace, what confirms that OBJ and JSON files are similar size in many cases. JSON files have a structure like follows:

1. List of materials:
   ”$materials$” : [{”$DbgIndex$” : 0,”$DbgName$” : ”$dummy$”,
   ”$colorDiffuse$” : [1.0000, 0.0000, 0.0000],”$vertexColors$” : $false$}]
2. Vertices:
   ”vertices”: [-5.0,0.0,5.0,5.0,0.0,5.0,-5.0,0.0,-5.0,5.0,0.0,-5.0,-5.0,10.0,5.0,5.0,10.0,5.0,
   -5.0,10.0,-5.0,5.0,10.0,-5.0],
3. Normals coordinates:
   ”normals”: [0.0,-1.0,0.0,0.0,-1.0,0.0,0.0,1.0,0.0,0.0,1.0,0.0,0.0,0.0,0.0,1.0,0.0,0.0,
   1.0,1.0,0.0,0.0,1.0,0.0,0.0,0.0,0.0,0.0,-1.0,0.0,0.0,-1.0,-1.0,0.0,0.0,-1.0,0.0,0.0]
4. Colors:
   ”colors”: [],
5. Texture coordinates:
   ”uvs”: [[0.0,0.0,1.0,0.0,0.0,1.0,1.0,1.0,0.0,0.0,1.0,0.0,0.0,0.0,1.0,1.0,1.0,0.0,0.0,0.0,1.0,
   0.0,0.0,1.0,1.0,1.0]],
6. Faces:
   ”faces”: [42,0,2,3,0,9,11,10,0,0,0,42,3,1,0,0,...,11,11,11]

2.3 Proposal Way of Storing 3D Data

We have decided to improve many things due to some restrictions associated with any existing Web technology. Optimization and users hardware differentiation was worth considering. First steps were aimed to the way of saving and storing 3D models data. It was decided to save all 3D models as OBJ format and then parse them to JSON format. However, large 3D objects are too big to transfer throughout the Internet in a real time.

2.3.1 Conversion from Text Plain OBJ to Two-Dimensional PNG

We have developed an idea for optimization of transferring huge 3D data sets in form of two-dimensional format. This conception is based on saving any data to RGB channels of two-dimensional graphic’s file. Informations about the materials are stored in text file (e.g. JSON based format) and uploaded separately. First of all, geometry has to be placed on positive parts of XYZ axes. Then, every coordination of vertex is splitted into integer and fractional parts. The first part is stored in a red channel (R). Further, fractional part is divided by 256 and the floor of this product is stored in a green channel (G) and fractional part is stored in a blue channel (B), an alpha channel value (A) is set to 0.

$$R = \lfloor x_{position} \rfloor \tag{2.1}$$

$$t = \frac{(x_{position} - R) * 10000}{256}, G = \lfloor t \rfloor \tag{2.2}$$

$$B = t - G \tag{2.3}$$

$$A = 0 \tag{2.4}$$

where $\lfloor a \rfloor$ is a floor of a.

Information about faces are slightly unusually encoded. The face flag and indices are encoded one by one for vertices, material, texture UV's, normals and color. The flag is encoded as follows [10]:

1. bit – face type (0 if quad, 1 if triangle)
2. bit – 1 if face has material, 0 otherwise
3. bit – 1 if face has UV's, 0 otherwise
4. bit – 1 if vertices have UV's, 0 otherwise
5. bit – 1 if face has normals, 0 otherwise
6. bit – 1 if vertices have normals, 0 otherwise
7. bit – 1 if face has colors, 0 otherwise
8. bit – 1 if vertices have colors, 0 otherwise

Every face value is an integer number and requires only two channels – red and green. If face value is greater than 256, we divide it by 256 and save integer value in R channel and the fractional part in G channel. Otherwise, face value is stored only in R channel.

$$R = \lfloor \frac{face_{value}}{256} \rfloor \tag{2.5}$$

$$G = \frac{face_{value}}{256} - \lfloor \frac{face_{value}}{256} \rfloor \tag{2.6}$$

$$B = 0, A = 0 \tag{2.7}$$

where $face_{value}$ is every integer number for faces set.

Storing of normals is similar to faces, but we also store a sign of the value in blue channel.

$$t = \frac{normal_{value} * 10000}{256}, R = \lfloor t \rfloor \tag{2.8}$$

$$G = t - R, A = 0 \tag{2.9}$$

If $normal_{value} \geq 0$, then

$$B = 255, \tag{2.10}$$

otherwise

$$B = 0 \tag{2.11}$$

And finally, to store a UV's texture value is similar to a normal value, but UV's are not signed.

$$t = \frac{uv_{value} * 10000}{256}, R = \lfloor t \rfloor \tag{2.12}$$

$$G = t - R \tag{2.13}$$

$$B = 0, A = 0 \tag{2.14}$$

Alpha channel of the PNG file is used for separation of those data – firstly, we save vertices, then transparent pixel, faces, transparent pixel, normals, transparent pixel and UV's. Data are stored with accuracy to four decimal places.

This process is reversible. Therefore, a data from PNG file can be decoded easily. We have to add the value from an R channel to the sum of a G channel (multiplied by 256) and a B channel to determine a single vertex.

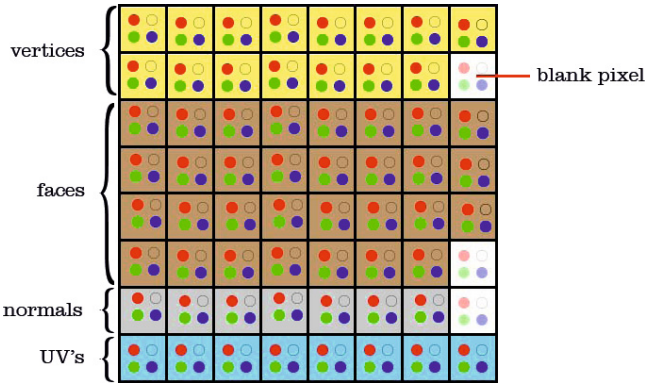


Fig. 2.3 Structure of PNG file with geometry – vertices (yellow background), faces (light brown), normals (gray) and UV's (light blue)

$$Vertex_{value} = R + \frac{G * 256 + B}{10000} \tag{2.15}$$

To determine a face value we have to calculate a product of an R channel (if any) and 256 value and add it to the G channel value.

$$Face_{value} = R + G * 256 \tag{2.16}$$

To determine a normal value we define if a value is negative, when B channel value equals 0, otherwise this value is positive. Then, we calculate a product of R channel (if any) and 256 value and finally add it to G channel value.
If $B = 255$, then

$$Normal_{value} = \frac{(R + G * 256)}{10000}, \tag{2.17}$$

otherwise

$$Normal_{value} = -\frac{R + G * 256}{10000}. \tag{2.18}$$

At first, our idea focused on saving those data to any of two-dimensional graphics format. We concluded, that using a format with lossy compression is incorrigible, because every stored information is improper relative to original data. In fact, we also wanted to use an alpha channel to save there additional information. Among to that there are left only PNG or TIFF file format. We have compared those two files to save models that we have mentioned earlier (with 1.5M and 76,625 vertices), but TIFF was about two times bigger than PNG file. What is more, the PNG file supports lossless data compression and was designed for transferring it throughout the Internet. The compression algorithm implemented for PNG file is called LZW (Lempel-Ziv-Welch) and it builds a data dictionary of information occuring in an original stream. Algorithm identifies patterns of data and matches them to entries in a dictionary. If the pattern does not match, a new phrase is created and added to the dictionary. Then, a new phrase is written to compressed output stream, what

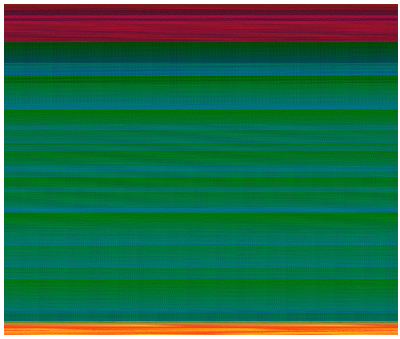


Fig. 2.4 PNG file with whole geometry – vertices, faces, normals and UV's for 550,000 vertices testing model of sphere

happens when pattern matches to entries from dictionary as well. This method eliminates redundant data, because, when any part of data occures once, then can be used many times. It is to be also an important point in our researches, because many of architectural models are symmetrical and have a lot of common vertices, faces and many others.

2.3.2 OBJ and PNG Benchmarks

The authors' idea is likely to decrease the size of files. For example, in case of raw 3D data with 550,000 vertices stored in OBJ we have 79MB file, when – after saving it to PNG – we have approximately only 11.5MB. There is also an improvement for small objects (35,000 vertices) – OBJ file was about 3.5MB, when our PNG file was only 0.76MB (780KB). Additionally, Internet technologies are well optimized for streaming transferring of pictures or movies, what in fact reduces time for uploading the PNG file against to OBJ file.

The time of decoding data from OBJ and PNG files are quite similar (about 100ms for testing model), therefore, there is no change. Decreasing of file's size affects the time of transferring the data via Web as well. Table 2.1 presents the approximate time of uploading our testing model of sphere (with 550,000 vertices) taking into account the diversity of Internet speed.

Table 2.2 Uploading time comparison for different file formats of sphere (550,000 vertices)

	1Mb/s	2Mb/s	4Mb/s	8Mb/s	12Mb/s
OBJ	15min 26s	7min 43s	3min 52s	1min 56s	1min 18s
PNG	2min 15s	1min 8s	0min 34s	0min 17s	0min 15s

2.4 Conclusions

The virtual reconstructions with a usage of portable and crossplatform technologies are burdened with many restrictions. Now, those technologies are not standards and there are many aspects to improve them. Our proposed solutions are in development stage, but so far they are promising. This fact can be confirmed by achieved results – reduction of file's size over seven times. There are many reasons to continue researches in this way, but Internet technologies (like WebGL) are demanding, needing many techniques of optimization to reach the main aim. What is more, WebGL lets us render only 65,536 vertices per draw, because vertex index buffers are limited to 16bit. Techniques described in this chapter, in combination with solutions applied in computer games [like normal mapping [9], portaling [5], GPU processing (shaders [1])] are promising to solve many restrictions.

Acknowledgements. The research is partially performed within the project "Virtual reconstructions in transitional research environments – the Web portal: Palaces and Parks in former East Prussia" supported by Liebnitz Gemeinschaft in the years 2013-2016.

References

1. Hoetzlein, R.C.: Graphics Performance in Rich Internet Applications in Computer Graphics and Applications. IEEE 32, 98–104 (2012)
2. Ksiazek, M., Pietruszka, M.: Interactive 3D architectural visualization with semantics in web browsers. Journal of Applied Computer Science 20, 59–70 (2012)
3. Limper, M., Jung, Y., Behr, J., Sturm, T., Franke, T., Schwenk, K., Kuijper, A.: Fast, Progressive Loading of Binary-Encoded Declarative-3D Web Content. IEEE Computer Graphics and Applications 33, 26–36 (2013)
4. Lowe, N., Datta, A.: A technique for rendering complex portals. IEEE Transactions on IEEE Visualization and Computer Graphics 11, 81–90 (2005)
5. Ortiz Jr., S.: Is 3D Finally Ready for the Web? IEEE Computer Magazine 43(1), 14–16 (2010)
6. Heok, T.K., Daman, D.: A review on level of detail. In: Computer Graphics, Imaging and Visualization, CGIV 2004, pp. 70–75 (2004)
7. Trevett, N.: 3D TransmissionFormat. NVIDIA (June 2013)
8. Yang, B., Pan, Z.: Hybrid Adaptive Normal Map Texture Compression Algorithm. In: 16th International Conference on Artificial Reality and Telexistence-Workshops, ICAT 2006, pp. 349–354 (2006)
9. (2014), `https://github.com/mrdoob/three.js/blob/master/utils/converters/obj/convert_obj_three.py` (April 10, 2014)

Chapter 3
Spatial Representation of Object Location for Image Matching in CBIR

Tatiana Jaworska

Abstract. At present a great deal of research is being done in different aspects of Content-Based Image Retrieval (CBIR). Representation of graphical object location in an image is one of the important tasks that must be dealt with in image DB as an intermediate stage prior to further image retrieval. The issue we address is the principal component analysis (PCA) applied to spatial representation of object location. We propose how to describe the object's spatial location to use it later in the search engine for image comparison. In this paper, we present the promising results of image retrieval based on the number of objects in images, object spatial location and object similarity.

3.1 Introduction

In recent years, the availability of image resources and large image datasets has increased tremendously. This has created a demand for effective and flexible techniques for automatic image classification and retrieval. Although attempts to construct Content-Based Image Retrieval (CBIR) in an efficient way have been made before [6], the extraction of semantically rich metadata from computationally accessible low-level features, is still considered a major scientific challenge because images and graphical data are complex in terms of visual and semantic contents. Depending on the application, images are modelled using their:

- visual properties (or a set of relevant visual features) [3],
- semantic properties [2], [15],
- spatial or temporal relationships of graphical objects [5].

Tatiana Jaworska
Systems Research Institute, Polish Academy of Sciences
ul. Newelska 6, 01-447 Warszawa, Poland
e-mail: Tatiana.Jaworska@ibspan.waw.pl

A. Zgrzywa et al. (eds.), *New Research in Multimedia and Internet Systems*,
Advances in Intelligent Systems and Computing 314, DOI: 10.1007/978-3-319-10383-9_3

The spatial relationships of graphical objects together with the classification problem are crucial for multimedia information retrieval in general, and for image retrieval in particular.

Object classification is mentioned here as an important issue in the context of CBIR because it is used for several purposes in the system, for example [13]:

- to compare whole images. Specifically, an algorithm which describes a spatial object location needs classified objects;
- to help the user form a query in the GUI. The user forms a query choosing graphical objects semantically collected in groups;
- to compare image objects coming from the same class as a stage in the image retrieval process.

3.1.1 CBIR Concept Overview

In general, our system consists of five main blocks (Fig. 3.1):

- the image preprocessing block, responsible for image segmentation and extraction of image object features, implemented in Matlab, (cf. [12]);
- the database, which is implemented in the Oracle Database (DB), stores information about whole images, their segments (here referred to as graphical objects), segment attributes, object location, pattern types and object identification, (cf. [14]);
- the classification module, used by the search engine and the GUI, is implemented in Matlab. Classification helps in the transition from rough graphical objects to human semantic elements. [12]
- the search engine, responsible for the searching procedure and retrieval process, based on feature vectors of objects and spatial relationship of these objects in an image, implemented in Matlab. The algorithms applied in this module will be described in general in section 3 and in detail in [12].
- the graphical user's interface (GUI) also implemented in Matlab, which allows users to compose their own image, consisting of separate graphical objects as a query. We have had to create a user-friendly semantic system.

3.1.2 Representation of Graphical Data

In our system, a new image is segmented, yielding as a result a collection of objects. Both the image and the extracted objects are stored in the database. Each object, selected according to the algorithm presented in detail in [15], is described by some certain low-level features. The features describing each object include: average colour k_{av}, texture parameters T_p, area A, convex area A_c, filled area A_f, centroid $C=\{x_c, y_c\}$, eccentricity e, orientation α, moments of inertia m_{11}, bounding box $\{bb_1(x,y), ..., bb_s(x,y)\}$ (s – number of vertices), major axis length m_{long}, minor axis length m_{short}, solidity s and Euler number E and Zernike moments $Z_{00},\ldots,Z_{33}$. All features, as well as extracted images of graphical objects, are stored in the DB.

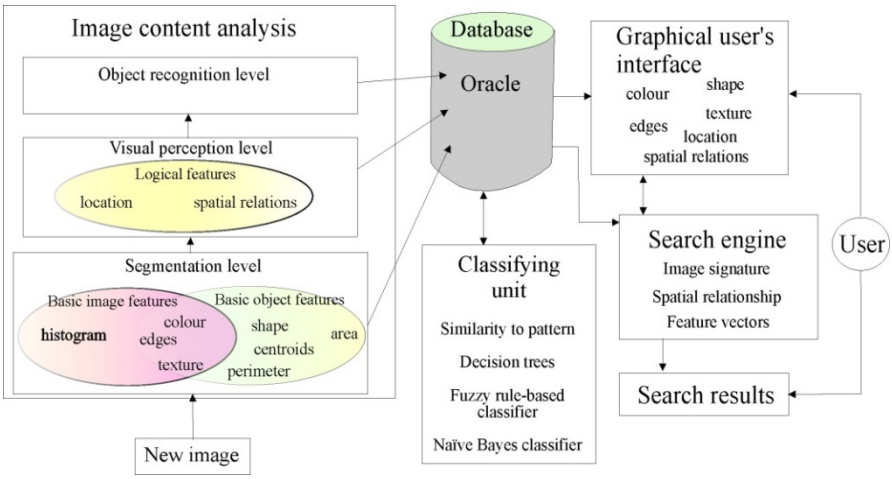


Fig. 3.1 Block diagram of our content-based image retrieval system

Let *Fo* be a set of features where:

$$F_O = \{k_{av}, T_p, A, A_c, \ldots, E\} \tag{3.1}$$

Hence, for an object, we construct a feature vector: $\mathbf{x} = [x_1, x_2, \ldots, x_r]$, where n is the number of the above-mentioned features, in our system $r = 45$. All the information is fed into the database.

3.1.3 Classification Methods Used in the CBIR System

Thus, the feature vector *Fo* (cf. (1)) is used for object classification. We have to classify objects in order to assign them to a particular class and to compare objects belonging to the same class.

So far, four kinds of classifiers have been implemented in our system. Firstly, there is the most intuitive one based on a comparison of features of the classified object with a class pattern. The problem of finding adequate weights, especially in the case of comparing complex values of some features, is also solved [1].

Secondly, decision trees as another option in a great number of classifying methods are used [7]. In order to avoid high error rates resulting from as many as 28 classes we use the hierarchical method. The more general division is created by dividing the whole data set into four clusters applying *k*-means clustering. The most numerous classes of each cluster constituting a meta-class are assigned to four decision trees, which results in 7 classes for each one.

Thirdly, to identify the most ambiguous objects we have built fuzzy rule-based classifiers (FRBC). There the ranges of membership functions for linguistic values for fuzzy rule-based classifiers according to crisp attributes are calculated [9, 10, 11].

Additionally, the Naïve Bayes classifier [17] has been implemented and now it seems to be as good as FRBC.

3.2 Spatial Relationship of Graphical Objects

It is easy for the user to recognize visually spatial location but the system supports full automatic identification based on rules for location of graphical elements which is a challenging task.

Let us assume that we analyse a house image. Then, for instance, an object which is categorized as a window cannot be located over an object which is categorized as a chimney. For this example, rules of location mean that all architectural objects must be inside the bounding box of a house. For the image of a Caribbean beach, an object which is categorized as a palm cannot grow in the middle of the sea, and so on.

In our system, spatial object location in an image is used as the global feature. For this purpose, the mutual position of all objects is checked. The location rules are also stored in the pattern library [13]. Moreover, object location reduces the differences between high-level semantic concepts perceived by humans and low-level features interpreted by computers.

For the comparison of the spatial features of two images an image I_i is interpreted as a set of n objects composing it:

$$I_i = \{o_{i1}, o_{i2}, ..., o_{in}\} \tag{3.2}$$

Each object o_{ij} is characterized by a unique identifier and a set of features discussed earlier. This set of features includes a centroid $C_{ij} = (x_{ij}, y_{ij})$ and a label L_{ij} indicating the class of an object o_{ij} (such as window, door, etc.), identified in the process described in [13]. Let us assume that there are, in total, M classes of the objects recognized in the database. For convenience, we number the classes of the objects and thus L_k's are just IDs of classes.

Formally, let I be an image consisting of n objects and k be the number of different classes of these objects, $k \leq M$, because usually there are some objects of the same type in the image, for example, there can be four windows in a house.

Then, by the signature of an image I_i (2) we mean the following vector:

$$\text{Signature}(I_i) = [\text{nobc}_{i1}, \text{nobc}_{i2}, \ldots, \text{nobc}_{iM}] \tag{3.3}$$

where: nobc_{ik} denotes the number of objects of class L_k present in the representation of an image I_i, i.e. such objects o_{ij}.

Additionally, for an image I_i we consider a representation of spatial relationships of the image objects. The o_{ij} objects' mutual spatial relationship is calculated based on the algorithm below.

```
Algorithm: PCV for an image
Input: ID_image object, object centroids, object classes
Output: PCV
Method:
    1. No_class = Binomial class combinations
    2. Com = Binomial centroid combinations
    3. For 1:object numbers
```

```
    4.    V_max = max Euclid distance
    5.    β = angle between V_max and the horizontal axis
    6.    θ = 90-atan(V_max, L_ij)
    7.    if L_p>L_q
    8.       θ = θ+360
    9.    S = Set all (L_p, L_q, θ)
    10.  End{for}
    11.  V = princomp(S)
End{Method}
```

Now, we consider one image; let C_p and C_q be two object centroids with $L_p<Lq$, located at the maximum distance from each other in the image, i.e.,

$$\text{dist}\,(C_p, C_q) = \max\,\{\text{dist}\,(C_i, C_j)\ \forall i,j \in \{1,2,\ldots,k\} \text{ and } L_i \neq L_j\} \tag{3.4}$$

where: dist(•) is the Euclidean distance between two centroids (see Fig. 2 middle subplot). The line joining the most distant centroids is the line of reference and its direction from centroid C_p to C_q is the direction of reference for computed angles θ_{ij} between other centroids. This way of computing angles makes the method invariant to image rotation.

Thus, we received triples (L_i, L_j, θ_{ij}) where the mutual location of two objects in the image is described in relation to the line of reference (see Fig. 2 middle subplot). Thus, there are $T=m(m-1)/2$ numbers of triples, generated to logically represent an image consisting of m objects. Let S be a set of all triples, then we apply the concept of principal component analysis (PCA) proposed by Chang and Wu [4] and later modified by Guru and Punitha [8] to determine the first principal component vectors (PCVs). For further analysis we use the first nine coefficients of the PCV (some example are shown in Table 3.1), which are the "spatial components" of the representation of an image I_i, and are denoted PCV_i.

Fig. 3.2 presents the most important stages in the determination of the spatial object location: from the presentation of the original image (top), through the object centroid locations (colours indicate particular classes) (middle subplot), to the 3D subplot of the principal components (bottom).

Table 3.1 Representative principal component vectors for the images shown in Fig. 3.2

Image name	First component	Second component	Third component
House-front	-0,001786	-0,003713	0,999992
Domy-banino-1	0,000206	0,003988	0,999992
Houselawn I_1	0,000388	0,001869	0,999998
Houselandscape I_2	0,004109	0,001557	0,999990

3.3 Construction of the Search Engine

Now, we will describe how the similarity between two images is determined and used to answer a query. Let a query be an image I_q, such as $I_q = \{o_{q1}, o_{q2},\ldots, o_{qn}\}$ (cf. (2)). An image in the database is denoted as I_b, $I_b = \{o_{b1}, o_{b2},\ldots, o_{bm}\}$.

In order to answer the query, represented by I_q, we compare it with each image I_b from the database in the following way.

A query image will be obtained from the GUI, where the user will construct their own image from selected DB objects. Now the GUI is under construction.

First of all, we determine a similarity measure $\mathrm{sim}_{\mathrm{sgn}}$ between query I_q and image I_b:

$$\mathrm{sim}_{\mathrm{sgn}}(I_q, I_b) = d_H(\mathrm{sgn}(I_q), \mathrm{sgn}(I_b)) \tag{3.5}$$

computing the Hamming distance between two vectors of their signatures (cf. (3)).

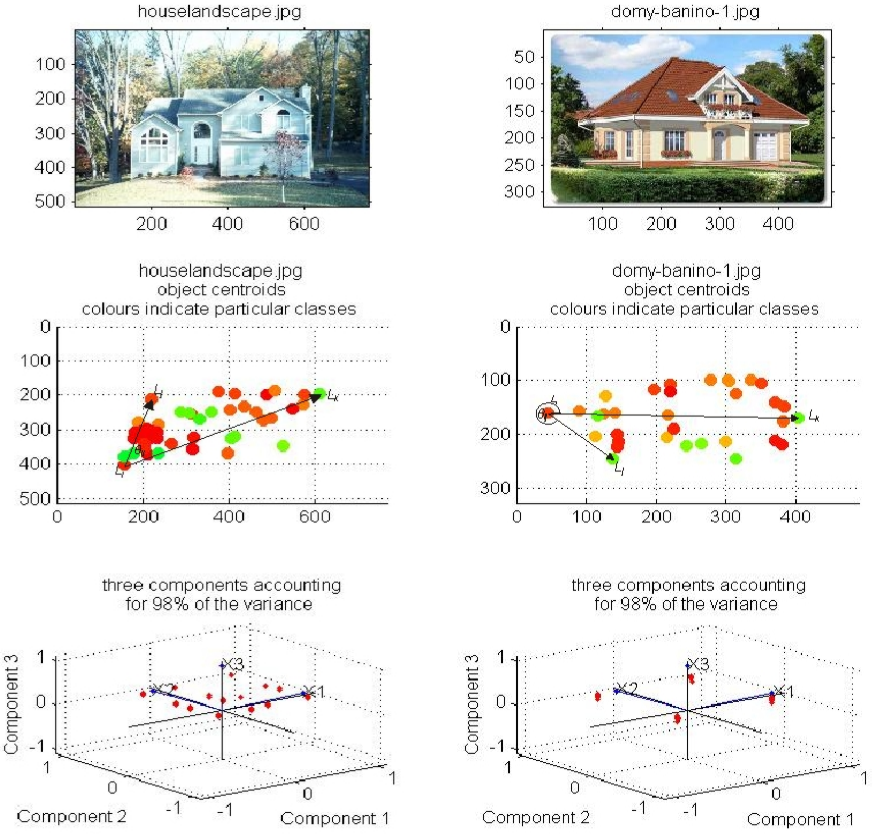


Fig. 3.2 The main stages of the PCV applied to determine the unique object spatial location in an image

If the similarity (6) is smaller than a threshold (a parameter of the query), then image I_b is rejected, i.e., not considered further in the process of answering query I_q. Otherwise, we proceed to the next step and we find the spatial similarity sim_{PCV} of images I_q and I_b, computing the Euclidean distance between their PCVs as:

$$\text{sim}_{\text{PCV}}(I_q, I_b) = 1 - \sqrt{\sum_{i=1}^{3} (PCV_{bi} - PCV_{qi})^2} \tag{3.6}$$

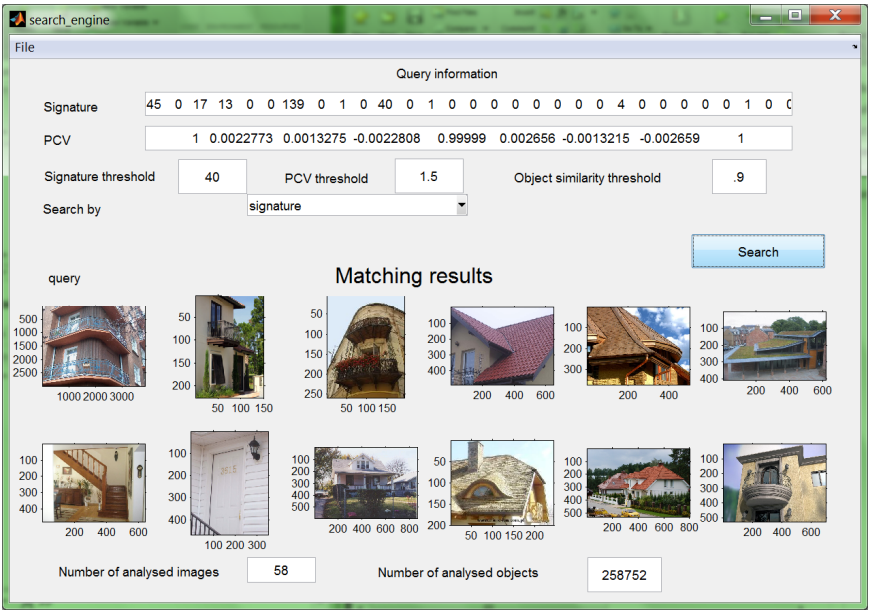


Fig. 3.3 An example of the search engine's operation. Matching images are found as an answer to the query (left image). The first step in the process of comparison is signature.

If the similarity (7) is smaller than the threshold (a parameter of the query), then image I_b is rejected, i.e., not considered further in the process of answering query I_q. The order of steps 6 and 7 can be reversed because they are the global parameters and hence can be selected by the user.

Next, we proceed to the final step, namely, we compare the similarity of the objects representing both images I_q and I_b. For each object o_{qi} present in the representation of the query I_q, we find the most similar object o_{bj} of the same class, i.e., $L_{qi} = L_{bj}$. If there is no object o_{bj} of the class L_{qi}, then $\text{sim}_{\text{ob}}(o_{qi}, o_b)$ is equal to 0. Otherwise, similarity $\text{sim}_{ob}(o_{qi}, o_b)$ between objects of the same class is computed as follows:

$$\text{sim}_{\text{ob}}(o_{qi}, o_{bj}) = 1 - \sqrt{\sum_{l} (Fo_{qil} - Fo_{bjl})^2} \tag{3.7}$$

where l indexes the set of features F_O used to represent an object, as described in (1).

When we find highly similar objects (for instance, $\text{sim}_{ob} > 0.9$), we eliminate these two objects from the process of comparison described by Mucha and Sankowski [16]. This process is realized according to the Hungarian algorithm for the assignment problem implemented by Munkres. Thus, we obtain the vector of similarities between query I_q and image I_b.

$$\text{sim}(I_q, I_b) = \begin{bmatrix} \text{sim}_{\text{ob}}(o_{q1}, o_{b1}) \\ \vdots \\ \text{sim}_{\text{ob}}(o_{qn}, o_{bn}) \end{bmatrix} \tag{3.8}$$

where n is the number of objects present in the representation of I_q. In order to compare images I_b with the query I_q, we compute the sum of $\text{sim}_{ob}(o_{qi}, o_b)$ and then use the natural order of the numbers. Therefore, the image I_b is listed as the first in the answer to the query I_q, for which the sum of similarities is the highest.

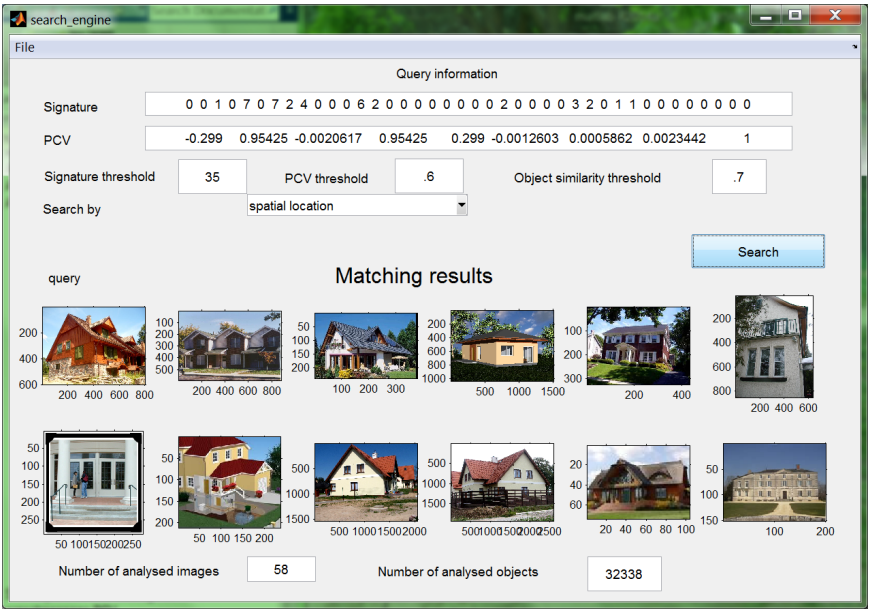


Fig. 3.4 An example of the search engine's operation. Matching images are found as an answer to the query (top left image). The first step in the process of comparison is the spatial location.

3.4 Results

Fig. 3.4 and 3.4 present tentative results obtained by having applied the above-mentioned procedure. We set all the thresholds for the search engine and the similarities are calculated. The first 11 most similar images to the query are displayed.

Now, we are using the whole images as a set of objects because the GUI is being constructed. Each image consists of a different number of elements, hence is divided into a different number of segments but on average there are 41 ones per image. In the next step, a measure or a ranking of image matching quality should be determined.

So far , the program code has not been optimised in terms of the time efficiency when confronted with thousands of images. Many experiments have to be carried out and different options have to be examined before taking a decision about the code optimization.

3.5 Conclusions

The methods already implemented will be also evaluated in terms of the addition of new classes to the system. GUI development will also enforce the introduction of subclasses to some of the most numerous classes.

Intensive computational experiments are under way in order to draw some conclusions regarding the choice of parameters for the search engine. However, the results we have obtained so far, using the simplest configuration, are quite promising.

As for the prospects for future work, image onthology should be prepared as a more powerful method for representation of objects in images. Image onthology will represent the semantic connections among images which are now impossible to add to image retrieval by the search engine.

References

1. Albert, A., Zhang, L.: A novel definition of the multivariate coefficient of variation. Biomedical J. 52(5), 667–675 (2010)
2. Berzal, F., Cubero, J.C., Kacprzyk, J., Marin, N., Vila, M.A., Zadrożny, S.: A General Framework for Computing with Words in Object-Oriented Programming. In: Bouchon-Meunier, B. (ed.) International Journal of Uncertainty. Fuzziness and Knowledge-Based Systems, vol. 15(suppl.), pp. 111–131. World Scientific Publishing Company, Singapore (2007)
3. Candan, K.S., Li, W.-S.: On Similarity Measures for Multimedia Database Applications. Knowledge and Information Systems 3, 30–51 (2001)
4. Chang, C.C., Wu, T.C.: An exact match retrieval scheme based upon principal component analysis. Pattern Recognition Letters 16, 465–470 (1995)

5. Cubero, J.C., Mari, N.N., Medina, J.M., Pons, O., Vila, M.A.: Fuzzy Object Management in an Object-Relational Framework. In: Proceedings of the 10th International Conference IPMU, Perugia, Italy, pp. 1775–1782 (2004)
6. Deb, S. (ed.): Multimedia Systems and Content-Based Image Retrieval, ch. VII and XI. IDEA Group Publishing, Melbourne (2004)
7. Fayyad, U.M., Irani, K.P.: The attribute selection problem in decision tree generation. In: Proceedings of the 10th National Conference on Artificial Intelligence, vol. 7, pp. 104–110. AAAI (1992)
8. Guru, D.S., Punitha, P.: An invariant scheme for exact match retrieval of symbolic images based upon principal component analysis. Pattern Recognition Letters 25, 73–86 (2004)
9. Hamilton-Wright, A., Stashuk, D.W.: Constructing a Fuzzy Rule Based Classification System Using Pattern Discovery. In: NAFIPS 2005 Annual Meeting of the Narth American Fuzzy Information Processing Society, pp. 460–465. IEEE (2005)
10. Ishibuchi, H., Nojima, Y.: Toward Quantitative Definition of Explanation Ability of fuzzy rule-based classifiers. In: IEEE International Conference on Fuzzy Systems, Taipei, Taiwan, June 27-39, pp. 549–556 (2011)
11. Jaworska, T.: Towards Fuzzy Classification in CBIR. In: Borzemski, L., Grzech, A., Świątek, J. (eds.) Information Systems Architecture and Technology. Knowledge Based Approach to the Design, Control and Decision Support, pp. 53–62. Oficyna Wydawnicza Politechniki Wrocławskiej, Wrocław (2013)
12. Jaworska, T.: A Search-Engine Concept Based on Multi-feature Vectors and Spatial Relationship. In: Christiansen, H., De Tré, G., Yazici, A., Zadrozny, S., Andreasen, T., Larsen, H.L. (eds.) FQAS 2011. LNCS, vol. 7022, pp. 137–148. Springer, Heidelberg (2011)
13. Jaworska, T.: Database as a Crucial Element for CBIR Systems. In: Proceedings of the 2nd International Symposium on Test Automation and Instrumentation, vol. 4, pp. 1983–1986. World Publishing Corporation, Beijing (2008)
14. Jaworska, T.: Object extraction as a basic process for content-based image retrieval (CBIR) system. Opto-Electronics Review, Association of Polish Electrical Engineers (SEP) 15(4), 184–195 (2007)
15. Liu, Y., Zhang, D., Lu, G., Ma, W.-Y.: A survey of content-based image retrieval with high-level semantics. Pattern Recognition 282, 262 (2007)
16. Mucha, M., Sankowski, P.: Maximum Matchings via Gaussian Elimination. In: Proceedings of the 45th Annual Symposium on Foundations of Computer Science (FOCS 2004), pp. 248–255 (2004)
17. Rish, I.: An empirical study of the naive Bayes classifier. In: Proceedings of IJCAI 2001 Workshop on Empirical Methods in AI, pp. 41–46 (2001)

Chapter 4
False and Miss Detections in Temporal Segmentation of TV Sports News Videos – Causes and Remedies

Kazimierz Choroś

Abstract. Categorization of sports videos, i.e. the automatic recognition of sports disciplines in videos, mainly in TV sports news, is one of the most important process in content-based video indexing. It may be achieved using different strategies such as player scenes analyses leading to the detection of playing fields, recognition of superimposed text like player or team names, identification of player faces, detection of lines typical for a given playing field and for a given sports discipline, detection of sports objects specific for a given sports category, and also recognition of player and audience emotions. The sports video indexing usually starts with the automatic temporal segmentation. Unfortunately, it could happen that two or even more consecutive shots of two different sports events, although most frequently of the same sport discipline, are falsely identified as one shot. The strong similarity mainly of color of playing fields does not sometimes allow the detection of a video transition. On the other hand, very short shots of several frames are detected in case of dissolve effects or they are simply false detections. Most often it is due to very dynamic movements of players or a camera during the game, changes in advertising banners on playing fields, as well as due to light flashes. False as well as miss detections diminish the efficacy of the temporal aggregation method applied to video indexing. The chapter examines the cases of false and miss detections in temporal segmentation of TV sports news videos observed in the Automatic Video Indexer AVI. The causes and remedies of false and miss detections are discussed.

Kazimierz Choroś
Institute of Informatics,
Wrocław University of Technology
Wyb. Wyspiańskiego 27, 50-370 Wrocław, Poland
e-mail: kazimierz.choros@pwr.edu.pl

A. Zgrzywa et al. (eds.), *New Research in Multimedia and Internet Systems*,
Advances in Intelligent Systems and Computing 314, DOI: 10.1007/978-3-319-10383-9_4

4.1 Introduction

A digital video is structured into a strict hierarchy. It is composed of structural units such as: acts, episodes (sequences), scenes, camera shots and finally, single frames. An act is the most general unit of a video. A video is composed of one or more acts. Then, acts include one or more sequences, sequences comprise one or more scenes, and finally, scenes are built out of camera shots. A shot is a basic unit. A shot is usually defined as a continuous video acquisition with the same camera, so, it is a sequence of interrelated consecutive frames recorded contiguously and representing a continuous action in time or space. The length of shots affects the dynamics of a video. The average shot length is generally several seconds or more. Whereas in the case of TV sports news video shots with player individual parts can be very short, they could last only a few seconds.

A scene usually corresponds to some logical event in a video such as a sequence of shots making up a dialogue scene, an action scene in a movie, or a sequence of shots recorded during a given sports event. A shot change occurs when a video acquisition is done with another camera. The cut is the simplest and the most frequent way to perform a change between two shots. In this case, the last frame of the first video sequence is directly followed by the first frame of the second video sequence. Cuts are the easiest shot changes to be detected. Software used for digital movie editing is more and more complex and other shot changes are now available. They include effects or transitions such as wipes, fades, or dissolves. A wipe effect is obtained by progressively replacing the old image by the new one, using a spatial basis. A dissolve is a transition where all the images inserted between two video sequences contain pixels whose values are computed as linear combination of the final frame of the first video sequence and the initial frame of the second video sequence. Cross dissolve describes the cross fading of two scenes. Over a certain period of time (usually several frames or several seconds) the images of two scenes overlay, and then the current scene dissolves into a new one. Fades are special cases of dissolve effects, where a most frequently black frame replaces the last frame of the first shot (fade in) or the first frame of the second shot (fade out). There are also many other kinds of effects combining for example wipe and zoom, etc.

A temporal segmentation of a video is a process of partitioning a sequence of frames into shots and scenes. It is the first and indispensable step towards video-content analysis and content-based video browsing and retrieval.

Although very effective temporal segmentation methods were and are developed [1], we could observe many false detections and miss detections of cuts or gradual transitions in the results of the segmentation process.

The causes may be diverse. Most frequently it is due to very dynamic movements of players or of a camera during the game, as well as due to light flashes during the interviews. We observe then extremely short shots.

Unfortunately, it also happens that two shots are detected as only one shot (cut is not detected) most frequently if these shots of different scenes are of the same sports disciplines. The main difference between these two shots easy observed by humans is the colors of player's clothing or the difference of small objects in the background.

Undetected cuts diminish the efficacy of the temporal aggregation [2] because shots become longer, and therefore may be treated as non-informative part of a video after its temporal aggregation. On the other hand false detections may suggest that this part of a sports video is more dynamic than it is and this part of a video can be incorrectly detected as a player scene part. It is highly desirable to eliminate such cuts.

The chapter is organized as follows. The next section describes the main related work in the area of temporal segmentation of sports videos. Some attempts already undertaken to solve the problem of false and miss detections in temporal segmentation process will be also presented. The section 4.3 reminds the basics of the measures used for the evaluation of temporal segmentation methods. The forth section explains why false and miss detections are not desirable in temporal aggregation process implemented in the Automatic Video Indexer AVI. The section 4.5 discusses the main causes of false and miss detections observed in practice and considers some remedies that could be undertaken to avoid this problem. The examples of such cases are also presented in this section. The final conclusions and the future research work are discussed in the last sixth section of this chapter.

4.2 Related Work

Investigations in automatic recognition of a content of a video have been carried out for many years, many of proposed methods have been tested on and applied for sport videos. An automatic summarization of TV sport videos has become the most popular application because sport videos are extremely popular in all video databases and Web archives. Due to the huge commercial appeal sports videos became a dominant application area for video automatic indexing and retrieval.

Segmentation of a video leads to the identification of the standard basic video units, such as shots and scenes. There are nowadays many methods of temporal segmentation. Many experiments have been also performed on the categorization of sports videos and many approaches and schemes have been developed. In the literature we can find review papers presenting the main and more and more effective approaches (see for example [3–9]).

The temporal segmentation methods has been tested using different genre of videos. Moreover, the segmentation methods can be adapted to the specificity of a given video category. The nature of movies, mainly the style of video editing, action dynamism, richness of movie storyboard and editing, dialog (audio) speed or pace of the narration, camera motion, light changes, and also special effects have a great influence on the effectiveness of temporal segmentation methods. In the experiments and tests performed in [10, 11] the effectiveness of segmentation

methods was analyzed for five different categories of movie: TV talk-show, documentary movie, animal video, action & adventure, and pop music video. TV talk-show is generally video shot realized in the TV studio, with a static scene, almost without camera movements and without object movements, without color changes of the scene, without light effects and without special editing effects. Documentary video is also static in nature, also without dynamic changes on the scene, but some fragments of such a video could be clearly different. In documentary videos many effects such as fades and dissolves are usually used. Animal videos are totally different. Objects of the scene (mainly animals) are moving, but rather in a slaw manner, also camera is moving also in a slow manner, rather constant but very rich variety of color used in every episode. The dissolve effects are rather long. Adventure video is a video composed of dynamic, short clips. We observe in such movies dynamic camera movements, also dynamic object movements on the scene, rich colors, contrast quickly changing, changing light, and many effects. In pop music videos the editing style is extremely dynamic, many images are rapidly juxtaposed together, many very rapid movements of cameras, lights, as well as objects of the scene. The shots of such a video are very short, even less then 1 second, cuts are very frequent, dissolve effects are relatively rare, contrary to light effects which are very often used.

These tests have shown that the specific nature of videos has an important influence on the effectiveness of temporal segmentation methods, fundamental methods of video indexing.

In turn, in [12, 13] seven of the most common video genres have been selected, i.e.: animated, commercial, documentary, movie, music, news, and sports video. Video materials were recorded from several TV channels. These seven different genres of videos have been used in the experiments. The strategy applied in this research was motivated by the observation that different genres of videos have different global color signatures, e.g. animated movies have specific color palettes and color contrasts (light-dark, cold-warm), music videos and movies tend to have darker colors (mainly due to the use of special effects), sports usually show a predominant hue (e.g. green for soccer, white for ice hockey), and so on.

Nevertheless, some error cases are observed using segmentation methods. These errors are discussed for example in [14, 15]. Changes in small regions may be missed or may be detected as a scene change, then the fast dissolve may be detected as a cut by a mistake. Additionally, the image blurs may lead to false positive. Also the problem of the detection of flashes or light changes have been analyzed to avoid the potential false positive detections. I has been observed that the flashes or light changes will lead to great changes in frame similarity or difference.

In [16] a new method has been proposed of an automatic shot based keyframe extraction for video indexing and retrieval applications. Initially, the shots are detected by using feature extraction, continuity value construction steps of shot boundary detection process and the shot frame clustering technique. The segmentation process has been tested on several ecological videos containing fast object/camera movements inside single shot, zoom-ins/outs and illumination effects.

4.3 Evaluation of Temporal Segmentation

How to evaluate the performance of a temporal segmentation process? Two main traditional measures: precision and recall are usually applied. Precision (P) is the fraction of detected shots that are correctly detected, while (R) recall is the fraction of shots that are correctly detected.

$$P = \textit{number of correctly detected shot / number of detected shots,}$$

$$R = \textit{number of correctly detected shot / number of all shots.}$$

To rank the performance of different algorithms, F-measure is often used, which is a harmonic average of recall and precision and defined as:

$$\textit{F-measure} = 2 \bullet P \bullet R / (P + R).$$

These measures have been used in different tests and experiments [1, 16]. Nevertheless, other measures are also useful: inserted transition count, deleted transition count, correction rate, deletion rate, insertion rate, error rate, quality index, correction probability [1]. In some evaluations the detection criteria require only a single frame overlap between the real and detected transition to make the detection independent of the accuracy of the detected boundaries. In the case of very short gradual transitions (5 frames or less), in some tests they are treated as cuts. The performance of detecting gradual shot transitions can be evaluated in terms of the numbers of frames overlapping in the detected and the real gradual transitions.

Because we observe false and miss detections some measures have been proposed to evaluate these cases. In [17] two indices have been used: the probability of miss detection defined as:

$$\textit{pmiss} = \textit{number of miss transitions / number of real transition}$$

and the probability of false detection:

$$\textit{pfalse} = \textit{number of false detections / (number of correctly detected transitions} + \textit{number of false detections).}$$

4.4 Temporal Segmentation and Aggregation in the AVI Indexer

The Automatic Video Indexer AVI [18] is a research project developed to investigate tools and techniques of automatic video content-based indexing for retrieval systems, mainly based on the video structure analyses [19] and using the temporal aggregation method [2]. The standard process of automatic content-based analysis and video indexing is composed of several stages starting with a temporal segmentation leading to the segmentation of a movie into small parts called video shots. Detected shots can be grouped to make scenes, and then content of shots and scenes is analyzed. Then a sports category can be identified using such strategies as: detection of playing fields, of superimposed text like player or team names, identification of player faces, detection of lines typical for a given playing field and for a given sports discipline, recognition of player and audience emotions, and also detection of sports objects specific for a given sports category.

Content-based indexing of TV sports news also starts by the automatic segmentation, then recognition and classification of scenes reporting sports events. The detection of video structure of sports news, not only of the simplest structural elements like frames or shots but also of video scenes, facilities the optimization of indexing process. The automatic identification of sports disciplines in TV sports news will be less time consuming if the analyzed video material is limited only to player scenes. The temporal aggregation method is applied for a significant reduction of video material analyzed in content-based indexing of TV sports news. The method detects and aggregates two kinds of shots: sequences of long studio shots unsuitable for content-based indexing and player scene shots adequate for sports categorization. The shots are grouped in scenes basing on the length of the shot as a sufficient sole criterion. The segmentation of video should be precise mainly for player scenes whereas studio scenes usually not useful for sport categorization should be grouped and neglected in further content-based analyses. The temporal segmentation of studio scenes is practically totally useless.

Player scenes represent only about third of the TV sports news. The temporal aggregation method leads to the easy rejection of these remaining two thirds of TV sports news with non-player scenes before starting content analyses of shots and in consequence to the reduction of computing time and to the improvement of the efficacy of content analyses and sports discipline recognition. The shot grouping is particularly desirable when content analyses are performed not for all frames of a shot or scene but only for their representatives called keyframes, i.e. the most representative frames, one frame for every given shot or scene.

Although, it can be presumed that the best segmentation systems have already achieved a very good and satisfactory performance on cuts as well as on most gradual transitions [1] any case of false or miss shot detection in temporal segmentation process unfortunately diminishes the usefulness of the temporal aggregation approach. Because wrong cases influence the results of temporal aggregation, it is obvious that better efficiency of temporal segmentation better usefulness of temporal aggregation.

False detections as well as undetected cuts diminish the efficacy of the temporal aggregation because shots become longer, and therefore may be treated as non-informative part of a video after its temporal aggregation.

4.5 False and Miss Detections

Very short shots including single frames are relatively very frequent. Generally very short shots of one or several frames are detected in case of dissolve effects or they are simply wrong detections.

4.5.1 False Detections

The causes of false detections may be different. Most frequently it is due to:

- very dynamic movements of players during the game (Fig. 4.1),
- very dynamic movements of objects just in front of a camera (Fig. 4.2),

- changes (lights, content) in advertising banners near the player fields (Fig. 4.3),
- very dynamic movements of a camera during the game (Fig. 4.4),
- light flashes for example during the interviews (Fig. 4.5).

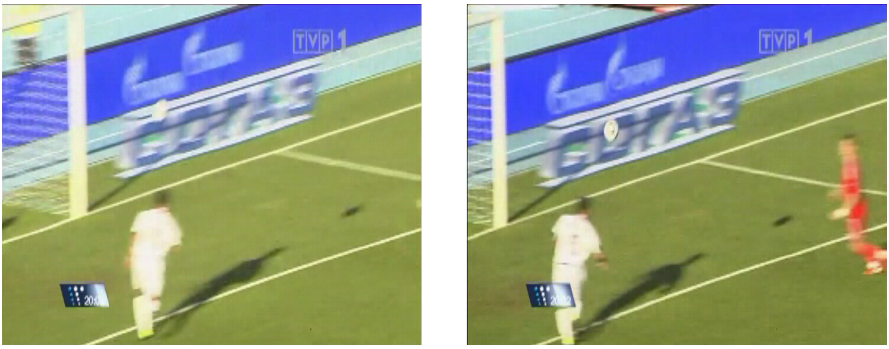


Fig. 4.1 Dynamic appearance/movement of a player in a camera

Fig. 4.2 Dynamic movements of an object just in front of a camera

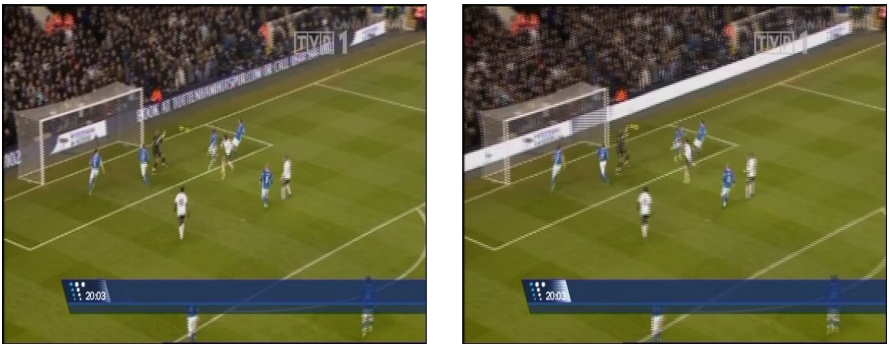


Fig. 4.3 Changes on a board and banner advertising near the player field

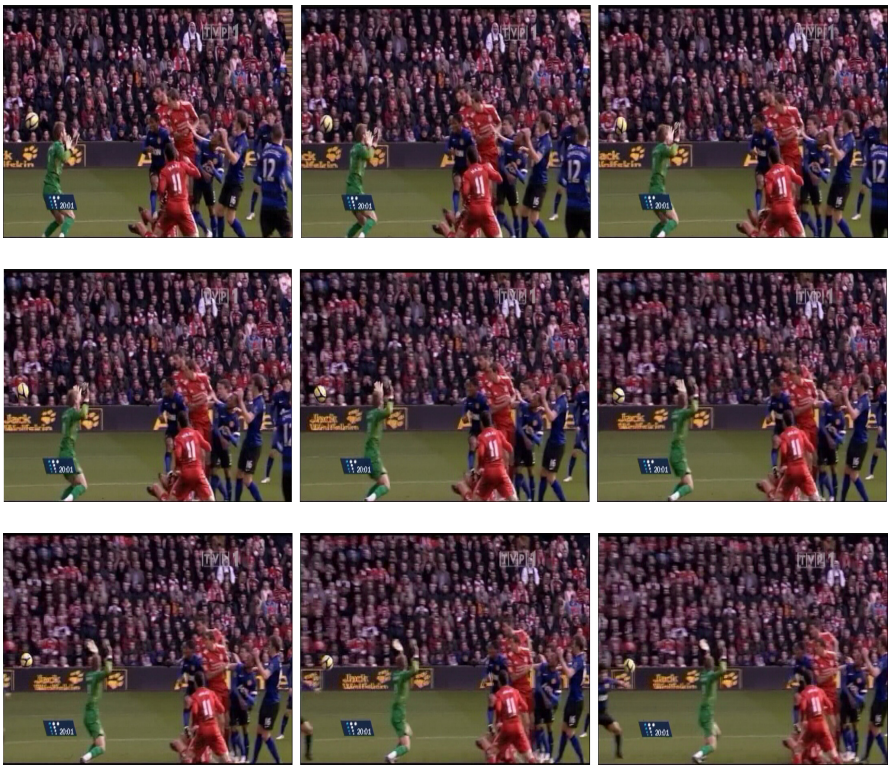


Fig. 4.4 Dynamic movement of a camera (and players) resulting in the significant changes in image background and in consequence resulting in erroneous detection of a cross dissolve effect

Fig. 4.5 Light flashes during giving away autographs

4.5.2 Miss Detections

Unfortunately, it happens that two shots usually of different scenes but of the same sports disciplines are detected as only one shot (cut is not detected). Most frequently it is due to the facts that:

- in the case of very wide views the backgrounds, so the most significant parts of frames, are very similar, it is obvious because these two shots are of the same sports disciplines (Fig. 4.6),
- for different winter sports disciplines the dominant color is frequently white color,
- when these two shots are close-up views of players like faces the specificity of the sports category does not sufficiently differ the frames.

Fig. 4.6 Two clearly different consecutive frames of two different shots from different sports events but, however, of very similar color histograms

4.5.3 Possible Remedies

Unfortunately, the methods of temporal segmentation are not perfect. New methods are still being developed and the well-known methods are being improved.

The cases of the wrong interpretation of sequences of gradual transitions (dissolves) are very frequent. All new solutions in this matter are desirable. Therefore, some possible remedies have been suggested,

The algorithm proposed in [20] can detect fades and dissolves and avoid the false detection caused by the flashlight.

Reduction of faulty detected shot cuts and cross dissolve effects in video segmentation process can be also achieved by analyzing the temporal relations of shots taking into account the specificity of a given category of digital videos [21].

A video shot segmentation scheme based on dual-detection model is proposed in [22], which includes the pre-detection process and the re-detection process. In the re-detection round, the scale invariant feature transform is applied to re-detect boundaries so as to improve the detection precision rate. In the case of an abrupt illumination or sudden changes of lights caused by camera flash effects the pixel values and histogram of a frame would suddenly change, so it leads to false detections in temporal segmentation based on pixel or histogram differences. Also it was observed that movements of large objects/camera would cause gradual changes in adjacent frames and they are similar to gradual transitions, so they are usually falsely detected as gradual shot boundaries.

Undetected cuts are mainly observed between the wide view shots. The main difference between two shots detected as one shot and easy observed by humans is the colors of player's clothing. The detection of players and then the analysis of colors of their clothes may diminish the number of miss cut detections [23].

4.6 Conclusions and Further Research

TV sports news videos are relatively short. The standard duration is several minutes. Many sports shots mainly those of player scenes included in headlines of TV sports news are very dynamic and very short. Furthermore, it happens that these shots are of the same sports category. The shots of the same sports category are very similar and they may be detected as one shot. The strong similarity mainly of color of playing fields and in consequence the similarity of frame histograms does not allow the detection of a video transition. On the other hand in case of dissolve effects very short shots of several frames are detected and they are simply false detections. So, we observe false and miss detections.

There are many reason of these wrong cases: very dynamic movements of players during the game, very dynamic movements of objects just in front of a camera or simply very dynamic movements of a camera itself during the game, changes (lights, content) in advertising banner near the player fields, light changes and light flashes.

Two shots of different scenes but usually of the same sports discipline are detected as only one shot most often in the case of very wide views. The backgrounds, so the most significant parts of frames, are very similar, for example for different winter sports disciplines when the dominant color is usually white color, or when these two shots are close-up views of players and these views does not sufficiently differ frames adequately to the specificity of the sports category.

Although, it is often said that the best temporal segmentation methods already achieve very good and satisfactory performance on cuts as well as on most gradual transitions it would be better to reduce these wrong cases. Any case of false or miss shot boundary detection in temporal segmentation process unfortunately diminishes the usefulness of the temporal aggregation approach. Wrong cases influences the results of temporal aggregation. False detections as well as undetected cuts diminish the efficacy of the temporal aggregation because shots become longer, and therefore may be treated as non-informative part of a video after its temporal aggregation. So, it is obvious that better efficiency of temporal segmentation better usefulness of temporal aggregation.

In further research and experiments the advanced methods of content analyses such as ball or player detections and audience analyses will be also applied in the Automatic Video Indexer to improve temporal segmentation and in consequence to improve temporal aggregation.

References

1. Smeaton, A.F., Over, P., Doherty, A.R.: Video shot boundary detection: Seven years of TRECVid activity. Computer Vision and Image Understanding 114(4), 411–418 (2010)
2. Choroś, K.: Temporal aggregation of video shots in TV sports news for detection and categorization of player scenes. In: Bădică, C., Nguyen, N.T., Brezovan, M. (eds.) ICCCI 2013. LNCS, vol. 8083, pp. 487–497. Springer, Heidelberg (2013)
3. Geetha, P., Narayanan, V.: A survey of content-based video retrieval. Journal of Computer Science 4(6), 474–486 (2008)
4. Money, A.G., Agius, H.: Video summarisation: A conceptual framework and survey of the state of the art. Journal of Visual Communication and Image Representation 19, 121–143 (2008)
5. Hu, W., Xie, N., Li, L., Zeng, X., Maybank, S.: A survey on visual content-based video indexing and retrieval. IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews 41(6), 797–819 (2011)
6. Del Fabro, M., Böszörmenyi, L.: State-of-the-art and future challenges in video scene detection: A survey. Multimedia Systems 19(5), 427–454 (2013)
7. Priya, R., Shanmugam, T.N.: A comprehensive review of significant researches on content based indexing and retrieval of visual information. Frontiers of Computer Science 7(5), 782–799 (2013)
8. Thounaojam, D.M., Trivedi, A., Singh, K.M., Roy, S.: A survey on video segmentation. In: Mohapatra, D.P., Patnaik, S. (eds.) Intelligent Computing, Networking, and Informatics. AISC, vol. 243, pp. 903–912. Springer, Heidelberg (2014)
9. Asghar, M.N., Hussain, F., Manton, R.: Video indexing: a survey. International Journal of Computer and Information Technology 3(1), 148–169 (2014)
10. Choroś, K., Gonet, M.: Effectiveness of video segmentation techniques for different categories of videos. In: New Trends in Multimedia and Network Information Systems, pp. 34–45. IOS Press, Amsterdam (2008)
11. Choroś, K.: Video shot selection and content-based scene detection for automatic classification of TV sports news. In: Internet – Technical Development and Applications. Advances in Soft Computing, vol. 64, pp. 73–80. Springer, Heidelberg (2009)
12. Ionescu, B., Seyerlehner, K., Rasche, C., Vertan, C., Lambert, P.: Content-based video description for automatic video genre categorization. In: Schoeffmann, K., Merialdo, B., Hauptmann, A.G., Ngo, C.-W., Andreopoulos, Y., Breiteneder, C. (eds.) MMM 2012. LNCS, vol. 7131, pp. 51–62. Springer, Heidelberg (2012)
13. Ionescu, B.E., Seyerlehner, K., Mironică, I., Vertan, C., Lambert, P.: An audio-visual approach to web video categorization. Multimedia Tools and Applications 70(2), 1007–1032 (2014)
14. Lian, S., Dong, Y., Wang, H.: Efficient temporal segmentation for sports programs with special cases. In: Qiu, G., Lam, K.M., Kiya, H., Xue, X.-Y., Kuo, C.-C.J., Lew, M.S. (eds.) PCM 2010, Part I. LNCS, vol. 6297, pp. 381–391. Springer, Heidelberg (2010)
15. Lian, S.: Automatic video temporal segmentation based on multiple features. Soft Computing 15(3), 469–482 (2011)
16. Smeaton, A.F., Over, P., Kraaij, W.: Evaluation campaigns and TRECVid. In: Proceedings of the 8th ACM International Workshop on Multimedia Information Retrieval, pp. 321–330. ACM (2006)

17. Adami, N., Corvaglia, M., Leonardi, R.: Comparing the quality of multiple descriptions of multimedia documents. In: Proceedings of the Workshop on Multimedia Signal Processing, pp. 241–244. IEEE (2002)
18. Choroś, K.: Video structure analysis and content-based indexing in the Automatic Video Indexer AVI. In: Nguyen, N.T., Zgrzywa, A., Czyżewski, A., et al. (eds.) Adv. in Multimed. and Netw. Inf. Syst. Technol. AISC, vol. 80, pp. 79–90. Springer, Heidelberg (2010)
19. Choroś, K.: Video structure analysis for content-based indexing and categorisation of TV sports news. International Journal of Intelligent Information and Database Systems 6(5), 451–465 (2012)
20. Ji, Q.G., Feng, J.W., Zhao, J., Lu, Z.M.: Effective dissolve detection based on accumulating histogram difference and the support point. In: Proceedings of the International Conference on Pervasive Computing Signal Processing and Applications (PCSPA), pp. 273–276. IEEE (2010)
21. Choroś, K.: Reduction of faulty detected shot cuts and cross dissolve effects in video segmentation process of different categories of digital videos. In: Nguyen, N.T. (ed.) Transactions on CCIV. LNCS, vol. 6910, pp. 124–139. Springer, Heidelberg (2011)
22. Jiang, X., Sun, T., Liu, J., Chao, J., Zhang, W.: An adaptive video shot segmentation scheme based on dual-detection model. Neurocomputing 116, 102–111 (2013)
23. Choroś, K.: Improved video scene detection using player detection methods in temporally aggregated TV sports news. In: Hwang, D., et al. (eds.) ICCCI 2014. LNCS (LNAI), vol. 8733, pp. 633–643. Springer, Heidelberg (2014)

Chapter 5
A Comparative Study of Features and Distance Metrics for Trajectory Clustering in Open Video Domains

Zhanhu Sun and Feng Wang

Abstract. Spatio-temporal trajectory is one of the most important features for understanding activities of objects. Trajectory clustering can thus be used to discover different motion patterns and recognize event occurrences in videos. Similarity measure plays the key role in trajectory clustering. In this chapter, we conduct a comparative study on different features and distance metrics for measuring similarities of trajectories from open video domains. The features include the location of each point on the trajectory, velocity and direction (curvature and angle) of motion along the timeline. The distance metrics include Euclidean distance, DTW (Dynamic Time Warping), LCSS (Longest Common Subsequence), and Hausdorff distance. Besides, we also investigate the combination of different features for trajectory similarity measure. In our experiments, we compare the performances of different approaches in clustering trajectories with various lengths and cluster numbers.

5.1 Introduction

With more and more video data captured to record real-world event occurrences and widely available from different sources such as web, automatic event recognition has attracted lots of research attentions. Motion trajectory is one of the most important cues in describing video event. A lot of interesting applications based on the analysis of trajectories have been developed such as detecting accidents or abnormal behaviors in surveillance video by analyzing trajectories of moving

Zhanhu Sun · Feng Wang
Department of Computer Science and Technology,
East China Normal University, Shanghai, China
e-mail: zhhsun@ica.stc.sh.cn, fwang@cs.ecnu.edu.cn

A. Zgrzywa et al. (eds.), *New Research in Multimedia and Internet Systems*,
Advances in Intelligent Systems and Computing 314, DOI: 10.1007/978-3-319-10383-9_5

vehicles or people, and tracking animals on earth by using GPS [6]. Trajectory of dynamic object in video, which is one of the most important features, will be used in analyzing semantic information in video. Same events or actions usually share similar motion patterns. Thus, trajectory analysis can be used to recognize different events in videos.

Trajectory is a sequence which is composed by points of multi-dimension location according to time sequence. In this chapter, trajectory is still presented as discrete point series:

$$T = \{P_1, P_2, P_3, ..., P_i, ..., P_n\}, i = 1,2,3,...,n,$$

where n is the total number of point in the trajectory. And the raw trajectories originate from trajectory generator [8]. For each point in the trajectory, we can present it as $P_i = (x_i, y_i)$. This means that the location of moving object is (x_i, y_i) at time i.

The remainder of this chapter is organized as follows. Section 5.2 describes the existing features and distance metrics for trajectory similarity measure. Section 5.3 presents our approach for trajectory clustering. Experimental results and discussions are presented in Section 5.4. Finally, Section 5.5 concludes this chapter.

5.2 Trajectory Similarity Measure

Similarity research is a crucial and fundamental task in multimedia mining. There are mass of works in this filed aiming to improve the performance of the similarity measuring method. We define length of trajectory as the number of point in trajectory.

There are existing several methods for measuring the similarity: algorithm based on Euclidean distance [4], algorithm based on DTW (Dynamic Time Warping) [1, 4, 5, 10], algorithm based on LCSS (Longest Common Subsequence) [3, 4, 5, 7] and algorithm based on Hausdorff distance.

In this section, we describe the features and distance metrics to be evaluated in this chapter. Besides, we also attempt to fuse different similarity measures to achieve better performance.

5.2.1 Features

Features to describe an object's moving trajectory include its location, moving velocity, the curvature and the corner of the moving path at each observation moment.

5.2.1.1 Location

Location is the basic and raw feature of trajectory for depicting spatial properties. It only contains the position information at each isolated moment. However an

object moves along the timeline still needs to be mined from the raw trajectory for describing its motion pattern.

5.2.1.2 Velocity

Velocity is an important feature to describe object's motion. The absolute speed of moving object and the change of speed along timeline are both informative to distinguish different motion patterns and event classes. Velocity is computed as the distance between two adjacent points in same intervals and we present it as a vector which has 2 components.

5.2.1.3 Curvature

Besides velocity, the direction is another feature to describe motion. Here we employ curvatures [7] to capture the direction information in the trajectory which is defined as:

$$\kappa[i] = \frac{x'[i]y''[i] - y'[i]x''[i]}{[x'[i]^2 + y'[i]^2]^{3/2}}, \tag{5.1}$$

where x' and x'' are the first and the second order derivation, respectively, and

$$x'[i] = x[i] - x[i-1], x''[i] = x'[i] - x'[i-1]$$

Curvature shows the curve degree at each point in sequence. Then there produces a curvature trajectory which has same length with original trajectory.

5.2.1.4 Corner

Two objects moving in different direction may present the same curvature although their trajectories are different. To cope with this problem, we further employ the absolute direction of the moving path with reference to x-coordinate. For this purpose, we calculate the corner angle formed by line between P_i and P_{i+1}, and the positive direction of x-coordinate. This gives a sequence:

$$T_{corner} = \{\theta_1, \theta_2, \theta_3, \cdots, \theta_n\},$$

where θ_i is the angle of velocity of the i-th point in trajectory related to x-axis. It can be computed by following formula:

$$\theta_i = \begin{cases} -\pi/2 & if\ x_{i+1} - x_i = 0, y_{i+1} - y_i < 0 \\ \pi/2 & if\ x_{i+1} - x_i = 0, y_{i+1} - y_i > 0 \\ \arctan\left((y_{i+1} - y_i)/(x_{i+1} - x_i)\right) & if\ x_{i+1} - x_i > 0 \\ \arctan\left((y_{i+1} - y_i)/(x_{i+1} - x_i)\right) - \pi & if\ x_{i+1} - x_i < 0, y_{i+1} - y_i < 0 \\ \arctan\left((y_{i+1} - y_i)/(x_{i+1} - x_i)\right) + \pi & if\ x_{i+1} - x_i < 0, y_{i+1} - y_i > 0 \\ 0 & if\ y_{i+1} - y_i = 0 \end{cases} \tag{5.2}$$

Corner angle is a feature which also illustrates the shape of trajectory. From formula (2) we can see that, comparing with curvature described in previous part, each component of corner sequence is the angle related to coordinate axes. So this give out an absolute direction of trajectory, and is more adequate for surveillance video. Corner angle can depict the direction of trajectory.

5.2.2 Distance Metric

In this section, we describe the distance metrics used to measure the similarity of different trajectories.

5.2.2.1 P-Norm Distance

P-Norm distance is the simplest method to calculate the distance between two vectors. Given two trajectories $T = (t_1, t_2, ..., t_n)$ and $R = (r_1, r_2, ..., r_n)$, the P-Norm distance between them is defined as:

$$D_{p-norm}(T,R) = \frac{1}{N}\sum_{i=1}^{N}\left((t_i.x - r_i.x)^p + (t_i.y - r_i.y)^p\right)^{1/p},$$

where N is the length of T and R. The time complexity of P-Norm distance is $O(n)$. When $p = 1$, it is Manhattan distance and when $p = 2$, it is Euclidean distance. In this chapter, we mainly discuss Euclidean distance which is widely used in distance measure. P-Norm distance suffers from the fact that it is very sensitive to distortions in the time axis. As can be seen in the definition, it requires that the two trajectories are with the same length and the corresponding points are well aligned.

5.2.2.2 Dynamic Time Warping (DTW) Distance

DTW distance is widely used to measure the distance of time series data. The calculation of DTW distance is based on dynamic programming. Given two trajectories $T = (t_1, t_2, ..., t_n)$ and $R = (r_1, r_2, ..., r_n)$, the DTW distance between them is defined as [5]:

$$DTW(T,R) = \begin{cases} 0 & if\ m=n=0 \\ \infty & if\ m=0 or n=0 \\ dist(t_1, r_1) + \min\{DTW(\text{Rest}(R), \text{Rest}(S)), DTW(\text{Rest}(R), S), DTW((R, \text{Rest}(S)))\} & otherwise \end{cases}$$

where Rest(T) is a trajectory by removing the first point from T. However, it can only address the problem of local scaling [7] and relatively sensitive to noise. The time complexity of DTW is $O(mn)$. When DTW is applied to very long trajectories in large datasets, time cost would be a limitation.

5.2.2.3 Longest Common Subsequence (LCSS) Distance

Similar to DTW distance, LCSS distance is also used to measure the distance between serial data, which is defined as [5]:

$$LCSS(T,R) = \begin{cases} 0 & if\ m = 0 or n = 0 \\ LCSS(\text{Rest}(T), \text{Rest}(R)) & \begin{array}{l} if\ |t_1.x - s_1.x| \le \delta \\ and\ |t_1.y - s_1.y| \le \varepsilon \end{array} \\ \max\{LCSS(\text{Rest}(T), R), LCSS((T, \text{Rest}(R)))\} & otherwise \end{cases}$$

LCSS distance is also based on dynamic programming and time complexity is $O(mn)$. In LCSS distance, two thresholds need to be determined which are used to decide whether the corresponding points in two trajectories are close enough and the values of them are dependent on the dataset. Just like DTW, LCSS is time expensive when it is applied to long trajectories in large datasets.

5.2.2.4 Hausdorff Distance and an Improved Version

Hausdorff distance [2] is defined as:

$$D_{Hausdorff} = \max\{d(T,R), d(R,T)\},$$

where $d(T,R) = \max_{t\in T} \min_{r\in R} \|t-r\|$ and $\|\cdot\|$ is Euclidean distance. Hausdorff distance could be used to measure the similarity between trajectories with various lengths. However, it doesn't reflect any information of direction. For example, it can't distinguish trajectories with same shape but different direction, so it is hard to achieve a better result.

In [9], an Improved Hausdorff distance measure is proposed. In this method, each point in trajectories is presented as a quadruple: $P_i = (x_i, y_i, v_i^x, v_i^y)$, where x and y are the positions in *x*-coordinate and *y*-coordinate, v_x and v_y are velocities in the direction of *x*-coordinate and *y*-coordinate, respectively. Velocity is a changing rate, and we assume that the interval between two adjacent points in same trajectory is a unit time. $v_i^x = x_{i+1} - x_i$, $v_i^y = y_{i+1} - y_i$. The problem, just as mentioned above, that Hausdorff distance can't distinguish direction of trajectory is solved when it employs velocity here.

The distance between two trajectories is:

$$h(T,R) = \frac{1}{N_T}\sum_{t\in T}[\|(x_i^t - x_{\varphi(i)}^r, y_i^t - y_{\varphi(i)}^r)\| + kD_v(v_i^t, v_{\varphi(i)}^r)] \tag{5.3}$$

$$h(R,T) = \frac{1}{N_R}\sum_{r\in R}[\|(x_i^r - x_{\varphi(i)}^t, y_i^r - y_{\varphi(i)}^t)\| + kD_v(v_i^r, v_{\varphi(i)}^t)] \tag{5.4}$$

$$D_{improvedhausdorff} = \min\{h(T,R), h(R,T)\}, \tag{5.5}$$

where

$$\varphi(i) = \arg\min_{j\in R} \|(x_i^T - x_j^R, y_i^T - y_j^R)\| \tag{5.6}$$

In Equation 6, $\varphi(i)$ is the point which belongs to trajectory R which has the smallest Euclidean distance to P_i in T, and the time complexity is $O(mn)$. As illustrated in Equation 7, D_v is cosine distance which is used to compute distance between 2 velocities, and is presented as follow:

$$D_v(v_i, v_j) = 1 - \cos\theta = 1 - \frac{v_i \cdot v_j}{|v_i| \cdot |v_j|} = 1 - \frac{v_x^i v_x^j + v_y^i v_y^j}{\sqrt{(v_x^i)^2 + (v_y^i)^2}\sqrt{(v_x^j)^2 + (v_y^j)^2}} \tag{5.7}$$

In Equation 3 and 4, k is a weighting parameter. The value of improved Hausdorff distance is the minimum in $h(T,R)$ and $h(R,T)$ according to Equation 5.

5.2.3 Combination of Different Features

Besides evaluating the performance of each feature, in this section, we combine different features for trajectory similarity measure. The four features described in Section 2.1 are considered. For distance metric, the Improved Hausdorff distance is employed due to its good performance according to our experiments. The ultimate distance is derived by linearly combining the above features with the Improved Hausdorff distance metric (LCIH) as illustrated below

$$D = \alpha \times D_{improvedHausdorff}\left(T_{original}, R_{original}\right) + \beta \times D_{improvedHausdorff}\left(T_{cur}, R_{cur}\right) + \gamma \times D_{improvedHausdorff}\left(T_{corner}, R_{corner}\right) \tag{5.8}$$

where $T_{original}$ and $R_{original}$ are original trajectories with velocity feature, T_{cur} and R_{cur} are the vectors of curvatures, T_{corner} and R_{corner} are the vectors of corners calculated along the trajectories, and α, β, γ $(\alpha + \beta + \gamma = 1)$ are the weights of three features which are empirically determined in our experiment.

5.3 Trajectory Clustering

Trajectory clustering is used for unsupervised learning of different motion patterns to describe event occurrences in videos. Here we employ K-means algorithm for trajectory clustering. The cluster centers are initialized in the way that all centers are distributed as diverse as possible. The first center is selected as the center of all trajectories. The trajectory which is the farthest from the first center is then selected as the second center. The remaining centers are selected one by one in the same way by maximizing the distances among all centers. Based on the initialization, the cluster centers are then iteratively updated with k-means algorithm until all the cluster centers do not change.

5.4 Experiments

In this section, we conduct experiments to evaluate the performances of different features, distance metrics and their combinations presented in Section 2 for trajectory similarity measure and clustering. A ThinkPad notebook with a processor Intel(R) Core(TM) i5-2500K 3.30 GHz, 16GB memory and 4TB hard disk is used for computation.

5.4.1 Dataset

Due to the difficulty in annotating the trajectories extracted from real world videos for evaluation, we employ a trajectory generator [8] to produce different trajectory clusters by setting different motion parameters. To simulate the moving properties

of real world objects, we adapt the generator's parameters including the length and the curve complexity to get different kinds of trajectories. In total we generate 4000 trajectories, and the length of each trajectory ranges from 100 to 500. Each cluster contains 20 trajectories generated with the same parameter setting. Originally the lengths of all trajectories in the same cluster are the same. To evaluate the performance of different approaches for measuring similarities between trajectories with varying lengths, we further randomly delete some points from the trajectories. Thus, the resulting trajectories in the same cluster have similar motion pattern and different lengths.

5.4.2 Evaluation Metric

We employ CCR (Correct Clustering Rate) [2] to measure the performance of different similarity measures. Assume that G is the ground truth set which has M clusters, and C is the clustering result. Given a cluster $c_i \in C$, we can find out the corresponding cluster $g_m \in G$ by using $\arg\max_i |c_i \cap g_i|$. Then CCR is defined as:

$$CRR = \frac{1}{N}\sum_i^M p_i \ , \tag{5.9}$$

where N is the total number of trajectories, P_i denote the number of the cluster trajectories in i-th resulting cluster, which is presented as follow:

$$p_i = \begin{cases} 0 & if\ \exists c_k \in C, |c_k \cap g_m| > |c_i \cap g_m| \\ |c_i \cap g_m| & otherwise \end{cases} \tag{5.10}$$

5.4.3 Results and Discussion

First, we evaluate the performances of four distance metrics: Euclidean distance, DTW distance, LCSS distance and improved Hausdorff distance. Fig. 5.1 illustrates the performances of these metrics for measuring similarities of trajectories with equal lengths. Fig. 5.1(a) employs curvature feature and Fig. 5.1(b) employs corners feature. Overall, corner feature performs significantly better then curvature feature. Both two features capture the moving direction of objects. The curvature employs the relative direction between neighboring points while corner employs the direction with reference to the x-axis. As discussed in Section 2, trajectories with different shapes may result in the same curvatures and this will eventually reduce the accuracy of similarity measure. In term of distance metrics, we can see that Euclidean distance performs relatively well when the lengths of trajectories are equal especially when curvature feature is employed. Overall the improved Hausdorff distance with corner feature achieves the best result in Fig. 5.1.

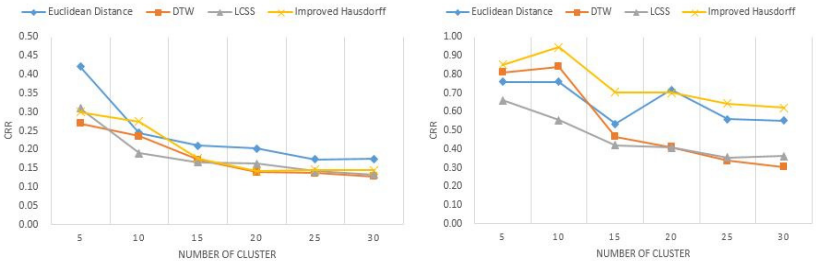


Fig. 5.1 (a) Using curvature to measure the similarity for equal-length trajectories (b) Using corner to measure the similarity for equal-length trajectories

In Fig. 5.2, we evaluate different metrics for measuring similarity of trajectories with different lengths. Euclidean distance is not used here since it cannot cope with trajectories of various lengths. As can be seen from Fig. 5.2, the improved Hausdorff distance gets the best performance for two different features. This demonstrates its capability of matching trajectories of different lengths in the same cluster. Furthermore, in our experiment, the improved Hausdorff distance can reach convergence more rapidly compared with DTW and LCSS.

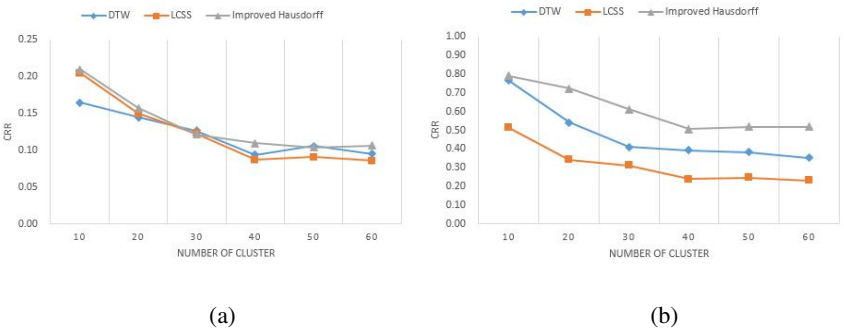
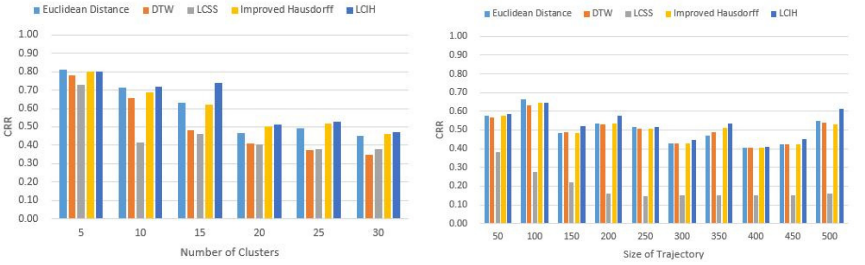


Fig. 5.2 (a) Using curvature to measure the similarity for different length trajectories (b) Using corner to measure the similarity for different length trajectories

From Fig. 5.3 we can see that Euclidean distance dose better when the number of cluster and the length of trajectories are small. When the number of cluster is 5 and the length of trajectory is 20, the CCR of Euclidean distance can reach 81%. It is clear that, no matter doing increment on number of cluster or length of trajectory, the improved Hausdorff distance provides a better performance comparing with other methods. LCSS always has a worse result and need to iterate much more times than other method.



(a) (b)

Fig. 5.3 (a) CCR of trajectories with same length as increasing the number of clusters (b) CCR of trajectories with same length as increasing the length of trajectories

Next, we use trajectories of different lengths to do another evaluation, and this time Euclidean distance is excluded. Fig. 5.4 shows the result of table 5.1, and our approach improves the accuracy by 1% to 4%. For all the trajectories in our dataset, we can see that LCIH distance could reach the best result compared with other similarity algorithms. Furthermore, LCIH is less time-consuming.

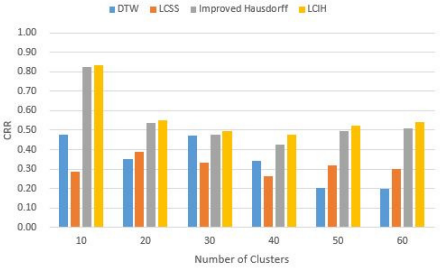


Fig. 5.4 Methods applied in trajectories with different length

Table 5.1 Methods apply in different length trajectory by increasing number of cluster

	10	20	30	40	50	60
DTW	0.475	0.350	0.470	0.340	0.205	0.198
LCSS	0.285	0.390	0.332	0.261	0.319	0.298
Improved Hausdorff	0.825	0.538	0.477	0.428	0.493	0.508
LCIH	**0.835**	**0.550**	**0.493**	**0.478**	**0.523**	**0.541**

5.5 Conclusion

This chapter compares existing methods for trajectory similarity measuring and clustering. Although there has been a lot of works on measuring trajectory similarity, these methods cannot be directly applied to open video domains. So there still

need further work to improve the performance of the algorithms. Compared with other methods mentioned in this chapter, our approach by linearly combining different features is more stable and achieves higher accuracy for clustering trajectories with various lengths and cluster numbers. In future work, we will address the noises in trajectories to derive more robust similarity measures and apply it to video event detection.

Acknowledgement. The work described in this chapter is sponsored by the National Natural Science Foundation of China (No. 61103127, No. 61375016), Shanghai Pujiang Program (No. 12PJ1402700), SRF for ROCS, SEM and the Fundamental Research Funds for the Central Universities.

References

1. Keogh, E.J., Pazzani, M.J.: Scaling up Dynamic Time Warping for Data mining Application. In: Proceedings of the Sixth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD 2000), pp. 285–289 (2000)
2. Zhang, Z., Huang, K., Tian, T.: Comparison of similarity measures for trajectory clustering in outdoor surveillance scenes. In: Proceedings of the 18th International Conference on Pattern Recognition, ICPR 2006, vol. 3, pp. 1135–1138. IEEE (2006)
3. Vlachos, M., Kollios, G., Gunopulos, D.: Discovering similarity multidimensional trajectories. In: Proceedings of the International Conference on Date of Conference, pp. 673–684 (2002)
4. Piciarelli, C., Micheloni, C., Foresti, G.L.: Trajectory-based anomalous event detection. IEEE Transactions on Circuits and Systems for Video Technology 18(11), 1544–1554 (2008)
5. Chen, L., Özsu, M.T., Oria, V.: Robust and fast similarity search for moving object trajectories. In: Proceedings of the ACM SIGMOD International Conference on Management of Data, pp. 491–502 (2005)
6. Ossama, O., Mokhtar, H.M.: Similarity search in moving object trajectories. In: Proceedings of the 15th International Conference on Management of Data, pp. 1–6 (2009)
7. Zhu, G., Huang, Q., Xu, C., Rui, Y., Jiang, S., Gao, W., Yao, H.: Trajectory based event tactics analysis in broadcast sports video. In: Proceedings of the 15th International Conference on Multimedia, pp. 58–67. ACM (2007), http://avires.dimi.uniud.it/papers/trclust/
8. Wang, X., Tieu, K., Grimson, W.E.L.: Learning semantic scene models by trajectory analysis. In: Leonardis, A., Bischof, H., Pinz, A. (eds.) ECCV 2006. Part III. LNCS, vol. 3953, pp. 110–123. Springer, Heidelberg (2006)
9. Salvador, S., Chan, P.: Toward accurate dynamic time warping in linear time and space. Intelligent Data Analysis 11(5), 561–580 (2007)

Chapter 6
Human Behavior Recognition Using Negative Curvature Minima and Positive Curvature Maxima Points

Krzysztof Kowalak, Łukasz Kamiński, Paweł Gardziński,
Sławomir Maćkowiak, and Radosław Hofman

Abstract. Recently, automated human behavior recognition are studied in the context of many new applications such as content-based video annotation and retrieval, highlight extraction, video summarization and video surveillance. In this chapter a novel description of human pose – a combination of negative curvature minima (NCM) and positive curvature maxima (PCM) points are proposed. Experimental results are provided in the chapter in order to demonstrate precision of the human activity recognition versus size of the descriptor (a temporal interval durations between the nodes of the model). The experimental results are focused on recognition of call for help behavior. The results prove high score of recognition of the proposed method.

6.1 Introduction

The past decade has witnessed a rapid proliferation of video cameras in all walks of life and has resulted in a tremendous explosion of video content. Several applications such as content-based video annotation and retrieval, highlight extraction and video summarization require recognition of the activities occurring in the video. The analysis of human activities in videos is an area with increasingly important consequences from security and surveillance to entertainment and personal archiving. In the area of surveillance, automated systems to observe pedestrian traffic areas and detect dangerous action are becoming important. This type of observation task is not well suited to humans, as it requires careful concentration over long periods of

Krzysztof Kowalak · Łukasz Kamiński · Paweł Gardziński · Sławomir Maćkowiak
Poznań University of Technology, Poland
e-mail: {kkowalak,lkaminski,pgardzinski,smack}@multimedia.edu.pl

Radosław Hofman
Smart Monitor sp. z o.o., Szczecin, Poland
e-mail: radekh@smartmonitor.pl

A. Zgrzywa et al. (eds.), *New Research in Multimedia and Internet Systems*,
Advances in Intelligent Systems and Computing 314, DOI: 10.1007/978-3-319-10383-9_6

time. Therefore, there is clear motivation to develop automated, intelligent, vision-based monitoring systems that can aid a human user in the process of risk detection and analysis. The system developed at Poznań University of Technology is able to perform fully automatic analysis of human behavior and recognition such behaviors as fainting, a fight or a call for help.

The scope of review is limited to some well-known graphical models that have been successfully applied on complex human activity modeling and behavior description in crowded public scenes. Activity modeling using single camera can be realized by many different methods, including: probabilistic graphical models (e.g. Bayesian networks [1,2], dynamic Bayesian networks [3-5], propagation nets [6]), probabilistic topic models (e.g. probabilistic latent semantic analysis models [7], latent Dirichlet allocation model [8, 9]), Petri nets [10], syntactic approaches [11] or rule-based approaches [12]. Bayesian networks are discussed in more detail because the approach proposed in this chapter belongs to them. Detailed reviews on other approaches are such as Petri nets, neural networks, synthetic approaches can be found in survey by Turaga et al. [13].

A Bayesian network or belief network is a probabilistic directed acyclic graphical model. Model consist of nodes which represent set of random variables (e.g. consecutive states of an event) and links which represent conditional dependencies between nodes. The strength of a dependency is parameterized by conditional probabilities that are attached to each cluster of parent-child nodes in the network. Such network has powerful capabilities in representing and reasoning state-oriented visual observations, so Bayesian network has been a popular tool for activity modeling. Bayesian network can be extended by introducing temporal dependencies between random variables. This extended network is called a dynamic Bayesian network.

Many different graphical models have been proposed for activity modeling. For instance, propagation nets [14], a subset of dynamic Bayesian networks with ability to explicitly model temporal interval durations, have been employed to capture the duration of temporal subintervals of multiple parallel streams of events.

This chapter is organized into 4 main sections. Section 2 presents the whole human activity recognition system and explained required blocks of video processing. Proposed approach uses a directed graphical model based on propagation nets, a subset of dynamic Bayesian networks approaches, to model the behaviors. Section 3 provides detailed explanations on the novel description of human pose – a combination of NCM and PCM points which are the main topics of this chapter. Section 4 presents the assumptions of the experiments and achieved results for exemplary behavior that is a $Callforhelp$. Section 5 provides conclusions and suggests a number of areas to be pursued as further work.

6.2 Human Activity Recognition System

The behavior of a person can be understood as a set of poses described by characteristic points of the person. A set of characteristic points at a given time defines a pose. A set of poses defined for consecutive time points or a set of time vectors for individual points forms a descriptor.

The set of points to define a pose may have a different configuration for different types of behavior to be detected. In other words, for at least one type of behavior, a set of points is generated having a configuration different than a set of points for another type of behavior. For consecutive frames, the positions of points belonging to the set are traced and form trajectories of points.

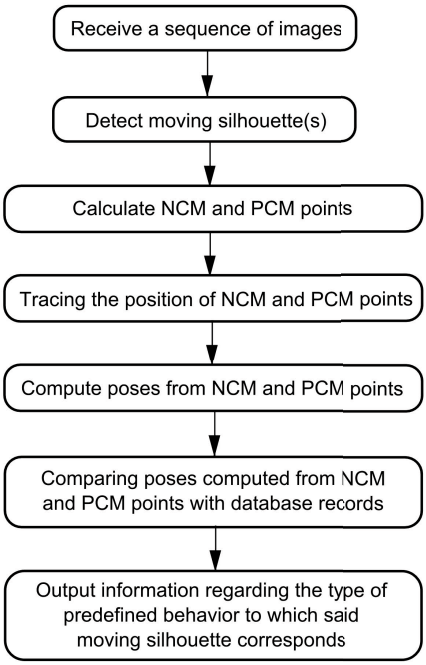


Fig. 6.1 General diagram of human activity recognition system

Figure 6.1 presents in general the human activity recognition system according to the proposed solution. Presented system used single stationary camera.

The main idea of the solution is to track the position of the NCM and PCM points on detected moving silhouette of the human. Position of points in the sets of points on consecutive images are traced in order to generate poses and next, the poses are compared with the predefined descriptors

corresponding to the behavior. The comparison is performed by calculating the Euclidean distance for pairs of corresponding points. Each pose shall fit within a predetermined range.

6.3 Negative Curvature Minima and Positive Curvature Maxima Points

There are different ways of describing shapes in images. The invention presented in this chapter is based on the contours of objects in the scene, which are well characterized using the so-called concavity minima or negative curvature minima (NCM) points and convexity maxima or positive curvature maxima (PCM). The NCM points alone may be used, inter alia, to recognize persons in video sequences, as described in [15]. The definition of minimum concavity is as follows: an NCM point is a point of the contour between the points ($P1$, $P2$ in Fig. 6.2) of a convex hull, for which the distance from the segment $|P1P2|$ is largest. The points $P1$ and $P2$ are suitably distant from each other which will be described in details in the subsequent sections of the present description. Fig. 6.1 shows an example of an NCM point.

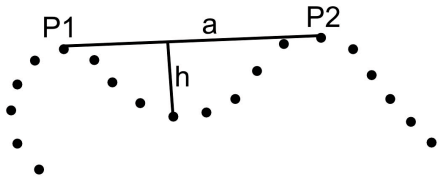


Fig. 6.2 Negative Curvature Minimum found based on hull points $P1$, $P2$

Like the concavity minima, contour convexity may be used to describe the curvature. The method utilizing contour convexity is called positive curvature maxima (PCM) points. This time, the extreme points $P1$ and $P2$ in Fig. 6.2 are selected from the convex hull so that $P1$ is the point closing the $P1$ is the point closing the i-th concavity and $P2$ is a point opening the $i + 1$-th concavity. Fig. 6.3 shows an example of a PCM point. Among so selected pair of points, from the contour a PCM point is selected so that the distance from the segment $|P1P2|$ is as high as possible.

6.3.1 Points Selection

This method is based on NCM points and PCM points and has been depicted in Fig. 6.4. Because of such combination it is possible to describe silhouettes with data defining curvature of the hull NCM for negative curvature minima and PCM for positive curvature maxima. This combination contains more information than a standalone NCM point description.

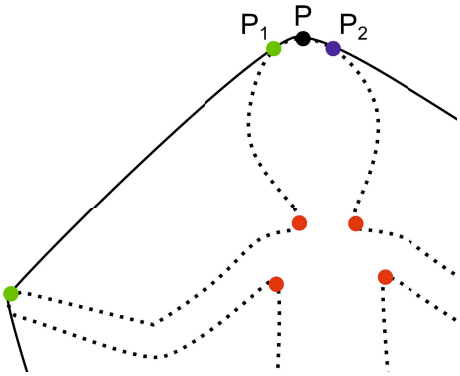


Fig. 6.3 Positive Curvature Maximum P (yellow)

A procedure (Fig. 6.4) determining NCM points starts from step S1 with selecting a pair of consecutive points $\{h_i, h_{(i+1)}\}$ in a vector of a convex hull. If a complete vector of the convex hull has been analyzed, the procedure proceeds from step S2 to step S9. If not, then in step S3 there is determined a length of a segment a between the points $\{h_i, h_{(i+1)}\}$. Next, at step S4, it is verified whether the length a is greater than a threshold γ. In case the length a is greater than the threshold, the procedure proceeds to step S5. Otherwise the procedure returns to step S1 in order to select another pair of points. At step S5, there is selected a point C from the contour vector such that point C is between points h_i and $h_{(i+1)}$ such that its distance h from the segment a is the greatest.

At step S6 it is verified whether an update hull condition is fulfilled, so that point C may be added to the convex hull. The parameters of the condition are as follows: AH_{th} is a concavity depth threshold, $cncArea$ is an area of concavity defined by the section of the hull between h_i and $h_{(i+1)}$ points, $cntArea$ is an inner area of the currently analyzed hull and the $Area_{th}$ is a threshold defining minimum ratio of concavity area to the inner area of the currently analyzed hull.

If the condition is fulfilled the procedure moves to step S7 where the update hull algorithm is executed and then moves to step S8 where point C is added to the NCM output vector.

Lastly, at step S9, the number of iterations is being checked as an end condition. In case it has not reached a required count, the process returns to step S1 and selects another pair of points from the vector of convex hull. The process is repeated from the beginning. The aforementioned update hull method utilizes an implementation of a known algorithm called "Gift wrapping" or "Jarvis march". Its task is to include in the convex hull a previously selected NCM point so that its definition is maintained. During execution of this method there is added a minimum number of contour points to the hull such that the hull vector maintains its continuity.

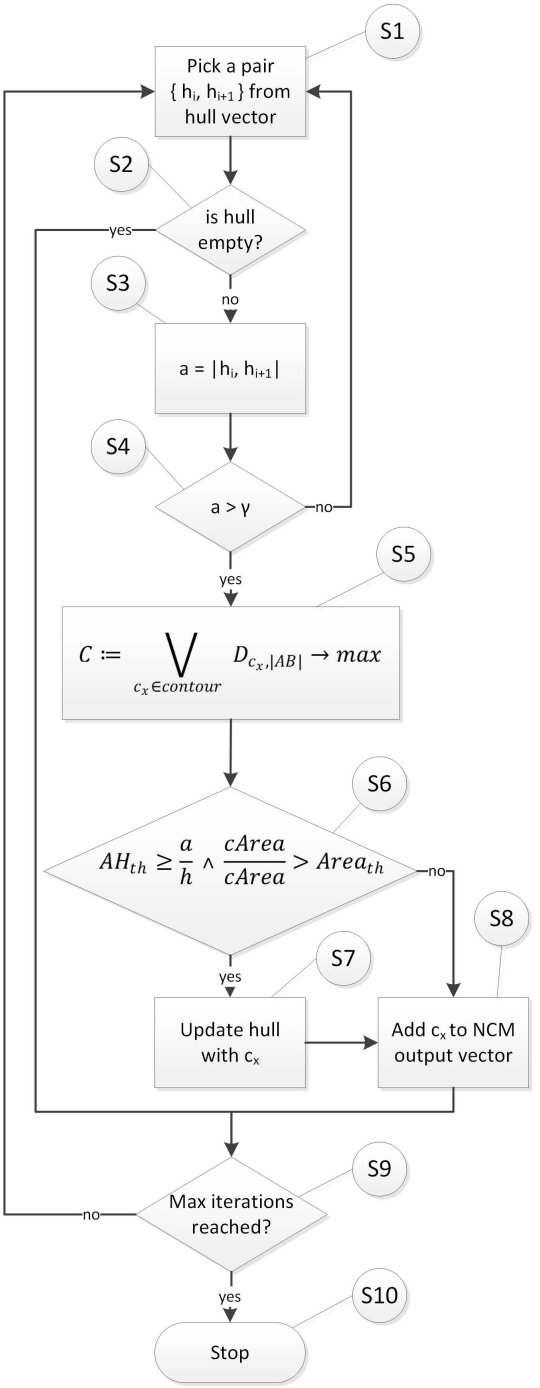


Fig. 6.4 NCM point selection algorithm

The other part of the fourth method relates to the PCM points that are determined similarly to the NCM points. The method may also be applied to pairs of points of a convex hull and is executed as follows:

1. First there are selected pairs of points $\{left_i, right_i\}$ until a pair fulfilling the condition for the length of a segment $D_{(|left_i right_i|)} > \gamma$ is obtained, thereby obtaining $\{left_0, right_0\}$ pair shown in Fig. 6.5
2. The second step is to move the $\{left_i, right_i\}$ until a next pair is found that fulfills the NCM condition thereby arriving at $\{left_1, right_1\}$ shown in Fig. 6.4 variant (b). The index of $left_1$ in the hull vector is stored as $idxRight$.
3. The third step of the procedure is to select a point K_0 from the convex hull between $idxLeft$ and $idxRight$, for which the distance h_0 from the segment $|right_0 left_1|$ is the greatest.
4. Lastly, as the fourth step set the $idxLeft = idx(right_1)$ and continue from the second step.

The subsequent K_i points are computed in an analogous way by maximizing their corresponding distances h_i from the segments $|right_i left_{(i+1)}|$. The process executes its last iteration when $left_n = left_0$. The vector of calculated points is added to previously determined NCM points thereby creating a silhouette descriptor.

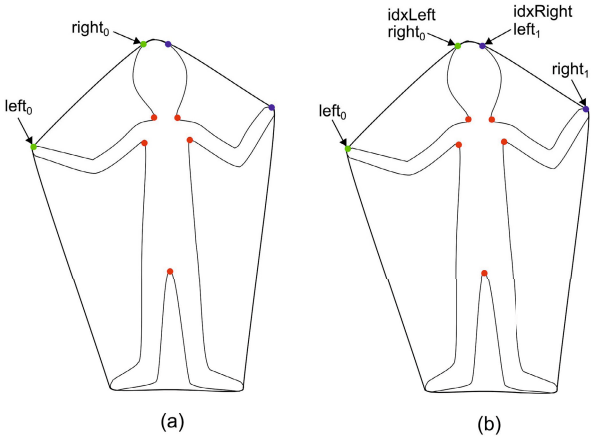


Fig. 6.5 PCM point selection example

6.4 Experimental Results

In our experiments all sequences were divided with respect to the subjects into a training set (9 persons), a validation set (8 persons) and a test set (9 persons)[1].

[1] The test sequence used in the experiments is available at `http://www.multimedia.edu.pl/missi2014-human-behavior`

The classifiers were trained on the training set while the validation set was used to optimize the parameters of each method (γ parameter mentioned in Section 3). The recognition results were obtained on the test set. This chapter contains results concerning $Call for help$ behavior.

Efficiency of recognition was analyzed and for evaluating classification algorithms the *precision* and *recall* performance metrics were used. The metrics are defined as follows:

$$precision = \frac{TP}{TP+FP} \quad (1)$$

$$recall = \frac{TP}{TP+FN} \quad (2)$$

where TP is the set of true positives, FP is the set of false positives and FN is the set of false negatives.

The effectiveness of the proposed approach was validated using four descriptors that consisted of varied number of states (10, 18, 24 and 36 states). Each of them was checked for effectiveness depending on the varied threshold.

The best results were obtained for 36 states descriptor that reached over 90% recall ratio with nearly 80% precision while second promising descriptor achieved a result of roughly 90% recall ratio at the cost of 60% precision. Those experiments show that more complicated descriptors require higher threshold of acceptance to achieve good results. Additionally, it has been observed that the points distribution over human silhouette is important. Some configurations prove better results than others. The recognition result is presented on Fig. 6.6 and Fig. 6.7.

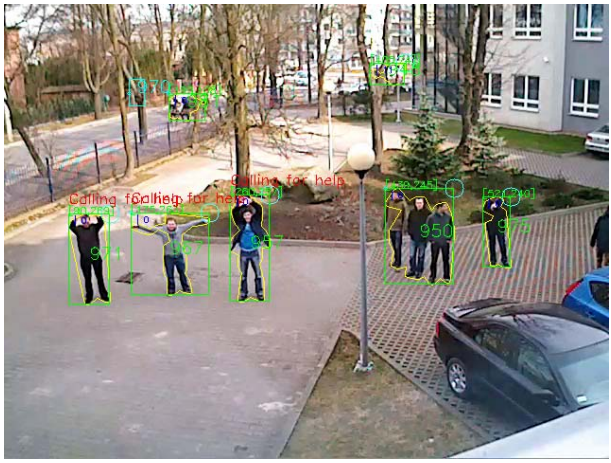


Fig. 6.6 Visual evaluation results

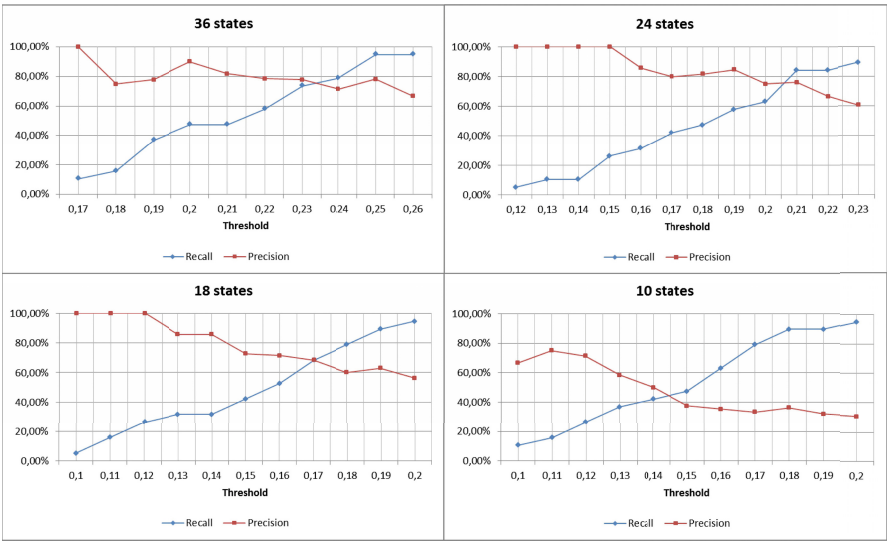


Fig. 6.7 Human activity recognition system evaluation results

6.5 Conclusions

In this chapter, a system for human activity recognition from monocular video is proposed. This system uses a novel description of human pose – a combination of NCM and PCM points on human contour. The points are used in classification process of the behaviour. Results prove that proposed solution seems to achieve a high detection efficiency. Moreover, the experiment results show that some work regarding characteristic points distribution and their relation to the specific behaviors is worth further research in the proposed approach.

Acknowledgements. The project *Innovative security system based on image analysis – SmartMonitor prototype construction (original title: Budowa prototypu innowacyjnego systemu bezpieczeństwa opartego o analize obrazu – SmartMonitor)* is the project co-founded by European Union (project number: UDA-POIG.01.04.00-32-008/10-02, value: 9.996.604 PLN, EU contribution: 5.848.560 PLN, realization period: 07.2011-04.2013). European funds for the development of innovative economy (Fundusze Europejskie – dla rozwoju innowacyjnej gospodarki).

References

1. Buxton, H., Gong, S.: Visual surveillance in a dynamic and uncertain world. Artificial Intelligence 78(1-2), 431–459 (1995)

2. Intille, S.S., Bobick, A.F.: A framework for recognizing multi-agent action from visual evidence. In: AAAI Conference on Artificial Intelligence, pp. 518–525 (1999)
3. Du, Y., Chen, F., Xu, W., Li, Y.: Recognizing interaction activities using dynamic Bayesian network. In: International Conference on Pattern Recogntion, pp. 618–621 (2006)
4. Duong, T., Bui, H., Phung, D., Venkatesh, S.: Activity recognition and abnormality detection with the switching hidden semi-Markov model. In: IEEE Conference on Computer Vision and Pattern Recognition, pp. 838–845 (2005)
5. Gong, S., Xiang, T.: Recognition of group activities using dynamic probabilistic networks. In: IEEE International Conference on Computer Vision, pp. 742–749 (2003)
6. Shi, Y., Bobick, A., Essa, I.: Learning temporal sequence model from partially labeled data. In: IEEE Conference on Computer Vision and Pattern Recognition, pp. 1631–1638 (2006)
7. Li, J., Gong, S., Xiang, T.: Global behaviour inference using probabilistic latent semantic analysis. In: British Machine Vision Conference, pp. 193–202 (2008)
8. Kuettel, D., Breitenstein, M.D., Gool, L.V., Ferrari, V.: Whats going on? Discovering spatio-temporal dependencies in dynamic scenes. In: IEEE Conference on Computer Vision and Pattern Recognition, pp. 1951–1958 (2010)
9. Wang, X., Ma, X., Grimson, W.E.L.: Unsupervised activity perception in crowded and complicated scenes using hierarchical Bayesian models. IEEE Transactions on Pattern Analysis and Machine Intelligence 48(2), 31(3), 539–555 (2009)
10. Albanese, M., Chellappa, R., Moscato, V., Picariello, A., Subrahmanian, V.S., Turaga, P., Udrea, O.: A constrained probabilistic Petri net framework for human activity detection in video. IEEE Transactions on Multimedia 10(6), 982–996 (2008)
11. Ivanov, Y.A., Bobick, A.F.: Recognition of visual activities and interactions by stochastic parsing. IEEE Transactions on Pattern Analysis and Machine Intelligence 22(8), 852–872 (2000)
12. Dee, H., Hogg, D.: Detecting inexplicable behaviour. In: British Machine Vision Conference, pp. 477–486 (2004)
13. Turaga, P., Chellappa, R., Subrahmanian, V.S., Udrea, O.: Machine recognition of human activities – a survey. IEEE Transactions on Circuits and Systems for Video Technology 18(11), 1473–1488 (2008)
14. Shi, Y., Bobick, A., Essa, I.: Learning temporal sequence model from partially labeled data. In: IEEE Conference on Computer Vision and Pattern Recognition, pp. 1631–1638 (2006)
15. Zhao, L.: Dressed Human Modeling, Detection, and Parts Localization. The Robotics Institute, Carnegie Mellon University, Pittsburgh (2001)

Chapter 7
Facial Emotion Recognition Based on Cascade of Neural Networks

Elżbieta Kukla and Paweł Nowak

Abstract. The chapter presents a method that uses the cascade of neural networks for facial expression recognition. As an input the algorithm receives a normalized image of a face and returns the emotion that the face expresses. To determine the best classifiers for recognizing particular emotions one- and multilayered networks were tested. Experiments covered different resolutions of the images presenting faces as well as the images including regions of mouths and eyes. On the basis of the tests results a cascade of the neural networks was proposed. The cascade recognizes six basic emotions and neutral expression.

7.1 Introduction

Human emotions expressed by mimics play fundamental role in everyday communication. Nonverbal information conveyed during conversation permits akes it possible to properly interpret and understand meaning of an utterance as well as the intentions of an interlocutor. Human brain recognizes mimics in a split second. For this reason emotion recognition became an important element of "natural" dialog between user and computer system.

Psychologists separated six basic emotions that are universal and occur in every culture. These are: happiness, sadness, fear, anger, disgust and surprise [2]. Although, the task of facial emotion recognition seems to be simple and intuitive for most of the people it is not easy for computer systems. One of the reasons is that every emotion can be expressed in many different ways, e.g. for fear there are about 60 various face expressions that have some common features [1]. For remaining emotions this number is similar. So, it is necessary to distinguish them somehow even if the differences are sometimes really subtle.

Elżbieta Kukla · Paweł Nowak
Institute of Informatics, Wrocław University of Technology
Wyb. Wyspiańskiego 27, 50-370 Wrocław, Poland
e-mail: {elzbieta.kukla,pawel.nowak}@pwr.edu.pl

A. Zgrzywa et al. (eds.), *New Research in Multimedia and Internet Systems*,
Advances in Intelligent Systems and Computing 314, DOI: 10.1007/978-3-319-10383-9_7

The chapter presents an approach to solve the problem of facial emotion recognition using a cascade of the neural networks trained to recognize individual emotions. The solution was tested on three different sets of photographs. Results obtained confirm intuitions and earlier observations made by other authors.

The chapter is organized as follows. Second part presents facial emotion recognition problem and some of the solutions reported in the literature. Part three describes in detail the way a cascade of neural networks is constructed as well as the tests that were carried out to verify its performance. Last section of this part reports and discusses results achieved for different sets of photos. Fourth part contains general conclusions and presents plans for future works.

7.2 Facial Emotion Recognition

Modern facial emotion recognition systems receive as input both pictures and videos. They all have a similar structure (Fig. 7.1) and consist of three main modules [11]: face capturing, extraction of the face features and emotion recognition.

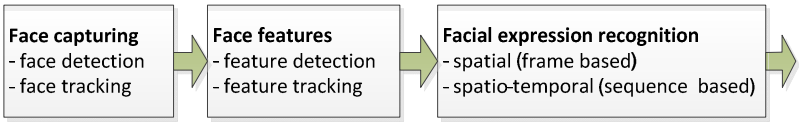


Fig. 7.1 General structure of the systems analyzing face expression [11]

Face capturing includes face detection and tracking that is connected with estimation of the face position. In the space of years many methods and algorithms concerning face detection have been developed. Their exhaustive review is presented among others by Yang, Kriegman and Ahuja in [13]. Separate category of problems is face tracking. In this field position [10] and especially chapter "Image and Video Processing Tools for HCI" gives comprehensive literature review.

Extraction of face features comprises feature detection and tracking. Generally, the approaches used to solve this problem can be categorized into local feature detection, global model detection and hybrid systems [10].

Facial emotion recognition systems are based on two approaches. The first of them classifies facial expression to one of the categories representing emotions. The second approach relies on identifying and measuring facial muscles motions (FACS) that delivers more detailed information about facial expressions.

Classification approach uses vectors of features that are characteristic for face appearance as well as for their movement. Features are often gathered during the process of their detecting and tracing. Depending on the solution categorization can be based on spatial- (referring to a single frame) or spatio-temporal (referring to a sequence of video frames) classifiers [4]. While spatial classifiers commonly use neural networks, spatio-temporal ones utilize Hidden Markov Models [9], [7].

The facial Action Coding System (FACS) [2] is based on human observations. It detects changes in particular parts of face [11]. FACS contains all the Action Units (AUs) of a face that cause facial movement. Part of them is closely connected with specific face muscle. The majority of Action Units permit both symmetric and asymmetric coding of facial actions, e.g. closing one eye. For AU that can be more or less intensive it is possible to use three- or five-level scales. Individual AU may form more complex actions that can be found in real situations. FACS system alone is designed for the actions descriptions and does not offer the possibilities of an inference about the emotions expressed. For this purpose distinct systems were developed. One of the examples is EMFACS (Emotional Facial Action System) that uses combinations of AC from FACS to determine the emotions expressed [Friesen 1983]. The other instance is FACSAID (Facial Action Coding System Affect Interpretation Dictionary) [3] that describes the meanings of particular behaviors represented by the combinations of AU and in this way gives the systems of facial expression recognition an ability of an interpretation of their results. The experts (psychologists) have assigned an emotion to every combination of AU.

7.3 Cascade of Neural Networks Applied to Facial Expressions Recognition

The experiment, presented in the subsequent part of this section, investigates in what way a cascade of neural networks will recognize six basic facial emotion expressions depending on image size, color, lighting and focus. This kind of recognition has been proposed by Golomb [5] and used for the first time in a system for female/male sex recognition.

At the first stage, various (one- and multilayer) neural networks were considered and compared to find out the six the most suitable networks for facial emotion recognition, one network for a particular emotion. Basing on this research, at the second stage, a cascade of neural networks has been proposed and tested to determine the influence of image representation on classification results.

All the experiments used three sets of photographs presenting six basic emotions: fear, anger, disgust, happiness, sadness, surprise and additionally neutral face expression. The photographs originated from The Karolinska Directed Emotional Faces (KDEF) data base [8], John Kanade [6] data base and a set of photographs of the students gathered by the authors.

The images chosen from three databases mentioned above were preprocessed to obtain the pictures that composed an input to cascade of classifiers.

7.3.1 Sets of Photographs

The Karolinska Directed Emotional Faces (KDEF) data base is one of the biggest set of photographs available in Internet. KDEF contains images of 70 faces: 35 men and 35 women that are 20–30 years old and none of them wears bread or glasses. The photographs present full-faces that express six basic emotions: fear,

anger, disgust, happiness, sadness and surprise. All the pictures are colorful and have the same dimensions 562x762 pixels. The data base comprises two series of pictures that present actors expressing the same emotion twice (Fig. 7.2). One of the series was used for training classifier and the other – for testing it.

Fig. 7.2 An example of KDEF subsets used for (a) training and (b) testing classifier [8]

The John Kanade data base of photographs is often used in scientific research related to facial emotion recognition. It contains grey scale, 640x490 pixels images. For the purpose of this work 16 photographs of different persons were chosen for every emotion to be recognized. Fig. 7.3 presents exemplary pictures from Kanade's data base.

Third set of photographs consists of the color pictures in size of 800x600 pixels. The photo present six basic emotions expressed by students, from 14 to 20 snaps for one emotion. The set was collected by one of the authors.

Fig. 7.3 Exemplary test photographs from John Kanade database [6]

7.3.2 Selection of Neural Networks for Emotion Recognition

Main idea of the research reported in this chapter was the application of a cascade of the classifiers to facial emotion recognition. Each of the classifiers was a single neural network or a configuration of two neural networks that was able to recognize the most fitting, single emotion. Configuration of two neural networks was used when an emotion was expressed in two different ways by different "actors". Then, the final result of the recognition was maximum of the results given by both of the networks. Such a situation occurred in case of anger and surprise that were expressed with closed or opened mouth, disgust that was expressed with closed mouth or opened mouth and visible teeth and sadness that was expressed by lowered mouth corners or in any other way. For fear and happiness recognition the classifier consisted of a single neural network.

The elements of the cascade were determined experimentally. Two kinds of neural networks were considered: a network consisted of one perceptron (PERC) and a three-layered network (MLP) with one exit neuron and five hidden neurons in the intermediate layer. The numbers of the networks entries depended on the size of the images that were recognized. For the purpose of the experiment, the authors took under consideration three kinds of pictures. Two of them consisted of 38x38 pixels (1444 entries) and 50x50 pixels (2500 entries). Third input image composed of two regions including mouth (30x30 pixels) and eyes (15x16 pixels), what gives total 1700 entries.

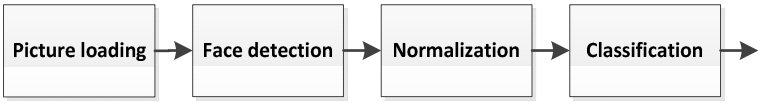


Fig. 7.4 Schema of the experiment

The six networks: 1444PERC, 1444MLP, 2500PERC, 2500MLP, 1700PRC and 1700MLP were examined according to the schema presented on Fig. 7.4. First, the images were loaded to test system. Then, Viola and Jones [12] algorithm detected faces on pictures. On every image a region containing face was reduced to the size of 300x300 pixels. If original region was smaller than 300x300 pixels it was enlarged to this dimension. Normalization that was accomplished at next step consisted in pruning peripheries, i.e. cutting background that occupies about 1/8 of image, and reducing its size to the three dimensions mentioned above. During this stage the colorful pictures were converted into the grey scale images and their histograms were smoothed. The normalized pictures were converted into the vectors that composed entrances for the networks bequeathed. The pixels of the images were the coefficients of the equivalent vectors. The last step was classification that aimed at appointing the best neural network for recognizing given emotion. All the networks were first trained. For training multilayered networks backward propagation algorithm was used with learning coefficient 0.01 and momentum 0.7

until max error level 0.00005 or number of iteration equal to 10000. Perceptron networks were trained with learning coefficient 0.2 and max error level 0.00005. Every network was tested then three times using KDEF database, Kanade database and the third set of photographs made by the author. Results obtained in particular tests are presented in Tables 7.1, 7.2 and 7.3. Symbols TP and FP means true positive and false positive recognition respectively. The best results of recognition for each emotion are bolded.

Table 7.1 Results of the tests for KDEF database

		Fear	Anger	Disgust	Happiness	Sadness	Surprise
1444PERC	TP	0.57	0.53	**0.84**	0.94	0.73	0.81
	FP	0.03	0.02	0.06	0.01	0.10	0.02
1444MLP	TP	0.46	0.57	0.77	0.96	0.76	0.86
	FP	0.01	0.01	0.03	0.02	0.06	0.01
2500PERC	TP	0.54	**0.61**	0.76	**0.97**	0.70	0.76
	FP	0.04	0.02	0.05	0.02	0.08	0.01
2500MLP	TP	0.49	0.56	0.76	0.96	**0.83**	**0.94**
	FP	0.01	0.01	0.04	0.02	0.09	0.02
1700PERC	TP	**0.63**	0.59	0.79	**0.97**	0.60	0.81
	FP	0.03	0.02	0.04	0.02	0.05	0.02
1700MLP	TP	0.49	0.60	0.77	0.94	0.67	0.89
	FP	0.01	0.02	0.04	0.02	0.04	0.02

The KDEF database contains two subsets of the pictures that present actors expressing the same emotion twice and therefore one of them was used for networks training, the other – for testing them. In this case (Table 7.1), happiness expressed by a smile achieved the best results of recognition 0.97 for 2500PERC and 1700PERC networks. Next was surprise with recognition 0,94 and 2500MLP network. Disgust gained recognition 0,84 for 1444PERC network. Only a little bit worse results 0,83 were obtained for sadness and 2500MLP network. Fear was recognized in 63% by 1700PERC network. The worst effects of recognition 0.61 were achieved for anger and 2500PERC network. False positive results in all the tests are not greater than 0.1. This value was obtained for sadness and disgust. Generally, better results were gained for perceptron networks. Multilayered network was better only in the case of surprise recognition.

The tests based on Kanade database and set of author's photographs were constructed somewhat differently. In these two cases the sets contained one unique photo per actor and per emotion. So, for every emotion two subsets were isolated basing on the Kanade database and two subsets from author's photo set respectively. Every subset contained photos of different persons expressing the same emotion.

Table 7.2 Results of the tests for Kanade database

		Fear	Anger	Disgust	Happiness	Sadness	Surprise
1444PERC	TP	0.00	0.19	**0.38**	0.75	**0.25**	**0.81**
	FP	0.02	0.06	0.04	0.03	0.23	0.09
1444MLP	TP	0.00	0.19	0.13	**0.81**	0.06	0.75
	FP	0.00	0.03	0.00	0.07	0.06	0.05
2500PERC	TP	0.00	**0.50**	0.25	**0.81**	0.31	0.75
	FP	0.02	0.11	0.03	0.07	0.24	0.03
2500MLP	TP	0.00	0.19	0.19	**0.81**	0.06	0.75
	FP	0.00	0.05	0.01	0.07	0.06	0.10
1700PERC	TP	**0.06**	0.00	0.19	0.75	0.00	0.69
	FP	0.03	0.01	0.02	0.08	0.06	0.06
1700MLP	TP	**0.06**	0.00	0.19	**0.81**	0.00	0.69
	FP	0.00	0.01	0.01	0.08	0.05	0.05

For Kanade database (Table 7.2) the best results of recognition 0.81 were achieved for happiness and surprise. The best recognitions of happiness was achieved by 1444MLP, 2500PERC and 2500MLP networks. For surprise the best identification effects were obtained by 1444PERC network. Remaining recognition results presented as follows: for anger 0.50 and 2500PERC network, disgust 0.38 and sadness 0.25 by 1444PERC network, and at last fear 0.06 by 1700PERC and 1700MPL networks. It is easy to notice that the results achieved in this test were inferior than in the case of photos from KDEF database. Also false positive results were inferior and reached even up to 0.24.

Table 7.3 Results of the tests for author's set of photo

		Fear	Anger	Disgust	Happiness	Sadness	Surprise
1444PERC	TP	0.00	0.06	**0.15**	0.48	**0.29**	0.50
	FP	0.01	0.19	0.12	0.02	0.28	01.
1444MLP	TP	0.00	0.19	0.00	0.52	**0.29**	0.56
	FP	0.00	0.13	0.01	0.04	0.11	0.03
2500PERC	TP	0.00	**0.44**	0.08	**0.57**	**0.29**	0.28
	FP	0.00	0.19	0.11	0.05	0.24	0.04
2500MLP	TP	0.00	0.06	0.00	**0.57**	**0.29**	0.56
	FP	0.00	0.14	0.05	0.05	0.12	0.14
1700PERC	TP	0.00	0.00	0.08	0.52	0.21	**0.67**
	FP	0.01	0.11	0.09	0.06	0.10	0.06
1700MLP	TP	0.00	0.06	0.00	0.48	0.21	0.67
	FP	0.00	0.11	0,02	0.03	0.04	0.04

The tests performed on author's photos set (Table 7.3) returned the best results for surprise (0.67) and 1700PERC network. Next was happiness with recognition ratio 0.57 gained by 2500MLP and 2500 network. Anger was recognized the best (0.44) by 2500PERC network. Recognition of sadness reached value 0.29 by 1444PERC, 1444MLP, 2500PERC and 2500MLP networks. Disgust was recognized by 1444PERC network at level of 0.15. The worst result of recognition (0.00) was achieved for fear. False positive results reached maximum value 0.28 for sadness.

The tests revealed that the best recognition results were obtained for the photos from the KDEF database. Therefore, they served to determine the elements of a cascade of neural networks. Figure 7.5 presents the cascade where every emotion is recognized by a single classifier (the best for this emotion in KDEF test). The neural networks in the cascade are arranged in ascending order of false positive results.

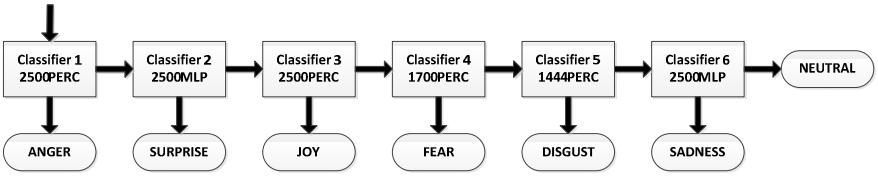


Fig. 7.5 The cascade of classifiers for facial emotion recognition

Every succeeding classifier assigns an input picture to one of the emotion class, for example the first classifier recognizes anger and if it does not identify this emotion the picture passes to the next classifier. The procedure concludes whenever the picture is classified to one of the classes representing six basic emotions. If none of the classifiers recognizes the picture then it is supposed that face at this picture presents neutral expression.

Additionally, particular networks composing the cascade were visualized. The images arisen as the visualization effects permitted to discover the features of the faces that decided about the results of the classifications.

Neural networks containing one perceptron and recognizing anger were trained to identify two mimic expressions. For anger expressed by closed mouth the pixels from the regions of eyebrows (their middle part), lips and chin had the greatest influence on the recognition results. For picture where anger is expressed by opened mouth the most important appeared the pixels that represented eyebrows, open mouth with visible teeth as well as the lines running from nose to mouth corners.

In the multilayered network that recognized surprise at the pictures with widely opened mouth it was possible to distinguish several groups of "deciding" neurons. One of them are concentrated on the region of mouth, the others – on the region of eyebrows, nose and middle part of brow. Surprise expressed by faces with closed mouth was recognized the best by multilayered neural network with the most important neurons focused around eyebrows and nose.

The perceptron network for happiness recognition took into account mainly the region of mouth with visible corners and wrinkles accompanying smile. In the case of fear recognition the most important are terrified eyes and opened mouth with and opened mouth with lines running from nose to mouth corners. The visualization of neutron that recognized disgust expressed by closed mouth revealed the most important regions around eyebrows and beneath nose. Disgust expressed by open mouth was recognized the best basing on wrinkles around nose, teeth and wrinkled eyebrows. In sadness recognition the most important neutrons from multilayered network are focused around lowered mouth corners and region of eyebrows.

7.3.3 Studies of the Facial Emotion Recognition by the Cascade of Classifiers

Facial emotion recognition based on the cascade of classifiers was tested using three sets of pictures described in section 7.3.2. All the experiments were carried out according to the schema presented at Fig. 7.4.

Results obtained for KDEF database are presented in Table 7.4. The best effect of recognition was achieved for happiness (0.97) and the worst for anger (0.51). Fear was often classified as surprise (0.11) what might be caused by widely opened eyes and half-opened mouth that were characteristic both for fear and surprise. Similarly, anger was classified as disgust (0.09) with respect to wrinkled eyebrows and as sadness (0.1) when lowered mouth corners were taken into account. Sadness in turn was confused with disgust (0.1) what might be caused by the fact that some actors wrinkled their noses and eyebrows.

Table 7.4 Emotion recognition for KDEF database

In\Out	Fear	Anger	Disgust	Happiness	Sadness	Surprise	Neutral
Fear	**0.54**	0.00	0.00	0.03	0.06	0.11	0.26
Anger	0.00	**0.51**	0.09	0.00	0.10	0.03	0.26
Disgust	0.00	0.00	**0.79**	0.06	0.06	0.00	0.10
Happiness	0.00	0.00	0.00	**0.97**	0.01	0.00	0.01
Sadness	0.01	0.00	0.10	0.00	**0.76**	0.00	0.13
Surprise	0.03	0.00	0.00	0.00	0.00	**0.94**	0.03
Neutral	0.01	0.07	0.03	0.00	0.00	0.04	**0.84**

The next test referred to identification of facial expressions presented at the photos from Kanade database. Recognition results achieved by the cascade in this case are presented in Table 7.5. Like in the previous experiment the best recognized emotion was happiness (0.81). The worst results were obtained for sadness (0.00) – none of pictures presenting sad face was identified. Sadness was relatively often (0.19) confused with anger and surprise what could be caused by significant differences between the photos used in training phase and the photos from Kanade's database.

Table 7.5 Emotion recognition for Kanade database

In\Out	Fear	Anger	Disgust	Happiness	Sadness	Surprise	Neutral
Fear	**0.06**	0.00	0.06	0.44	0.00	0.06	0.38
Anger	0.00	**0.50**	0.00	0.00	0.06	0.06	0.38
Disgust	0.00	0.38	**0.25**	0.00	0.00	0.06	0.31
Happiness	0.06	0.06	0.00	**0.81**	0.06	0.00	0.00
Sadness	0.00	0.19	0.00	0.00	**0.00**	0.19	0.63
Surprise	0.00	0.00	0.00	0.00	0.00	**0.75**	0.25
Neutral	0.06	0.06	0.00	0.00	0.06	0.06	**0.75**

The last experiment dealt with recognition of the emotions expressed by the faces from the author's collection of the photos. Table 7.6 presents the results achieved in this case. Fear was not recognized at all (0.00). Identification of the remaining emotions varied from 0.14 for sadness up to 0.57 for happiness.

Generally the results obtained in this test were worse than in previous tests. This can result from the fact that people presented at the pictures were not professional actors and the emotions presented by them were expressed intuitively. Additionally, photos from the third set were taken in unprofessional manner with respect to the equipment, conditions, lighting etc.

Figure 7.6 presents positive results of facial emotion recognition based on the cascade of neural networks for three sets of photographs presented in details and discussed in previous paragraphs.

Table 7.6 Emotion recognition for the author's set of the photographs

In\Out	Fear	Anger	Disgust	Happiness	Sadness	Surprise	Neutral
Fear	**0.00**	0.07	0.13	0.33	0.00	0.00	0.47
Anger	0.00	**0.44**	0.00	0.00	0.06	0.06	0.44
Disgust	0.00	0.23	**0.08**	0.00	0.15	0.00	0.54
Happiness	0.05	0.00	0.05	**0.57**	0.05	0.00	0.29
Sadness	0.00	0.07	0.00	0.00	**0.14**	0.14	0.64
Surprise	0.00	0.00	0.00	0.00	0.06	**0.56**	0.39
Neutral	0.00	0.06	0.06	0.00	0.12	0.35	**0.41**

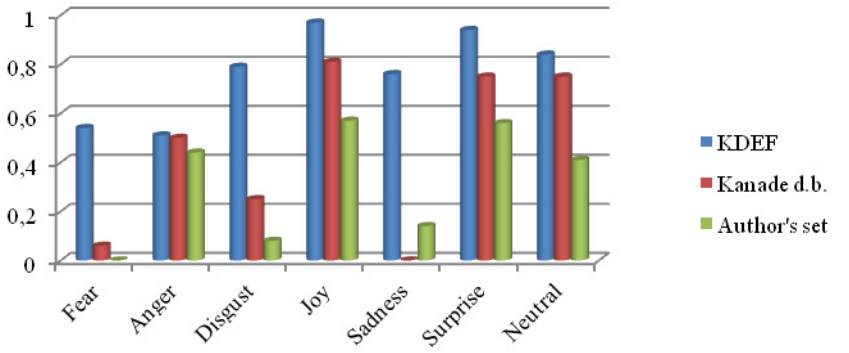


Fig. 7.6 Positive results of six basic recognition for three sets of photos

7.4 Conclusions and Future Works

The cascade of the neural networks developed in these studies returned satisfactory results (on average 76% for KDEF database) but it did not achieved the efficiency close to that of recent facial emotion recognition systems (>95%). The results obtained are different for particular emotions. The worst recognized emotion is fear, the best is happiness and surprise. It is worth to notice that exact comparison of the cascade with the other systems described in the literature is difficult because the authors could not have an access to the photos used in testing them.

Future investigations will concentrate on separate recognition of the two parts of faces that are the most important in emotion identification, i.e. regions of eyes, eyebrows and forehead (upper parts) and regions of mouth, nose and chin (lower parts). Particular networks could be trained and used to recognize separately particular action units from FACS coding system. Final recognition results is this case could be a fusion of the results obtained by individual classifiers.

References

1. Ekman, P.: Facial expression and emotion. American Psychologist 48(4), 384, 384–392 (1993)
2. Ekman, P., Friesen, W., Hager, J.: Facial action coding system. Consulting Psychologists Press, Palo Alto (1978)
3. Ekman, P., Hager, J., Rosenberg, E.: FACSAID: A computer database for predicting affective phenomena from facial movement (2003), http://face-and-emotion.com/dataface/facsaid/description.jsp, http://face-and-emotion.com/dataface/nsfrept/psychology.html (visited April 4, 2014)
4. Fasel, B., Luettin, J.: Automatic facial expression analysis: A survey. Pattern Recognition Society 36(1), 259–275 (2003)
5. Golomb, B.A., Lawrence, D.T., Sejnowski, T.J.: Sexnet: A neural net identifies sex from human faces. In: Lippman, R.P., Moody, J., Touretzky, D.S. (eds.) NIPS, vol. 3, pp. 572–577. Morgan Kaufmann, San Francisco (1991)
6. Kanade, T., Cohn, J., Tian, Y.: Comprehensive database for facial expression analysis. In: Proceedings of the Fourth IEEE International Conference on Automatic Face Gesture Recognition (FG 2000), Grenoble, France, pp. 46–53 (2000)
7. Lien, J., Kanade, T., Cohn, J., Li, C.: Detection, tracking and classification of action units in facial expressions. Robotics and Autonomous Systems 31(3), 131–146 (2000)
8. Lundqvist, D., Flykt, A., Öhman, A.: The Karolinska directed emotional faces (KDEF). CD ROM from Department of Clinical Neuroscience, Psychology section, Karolinska Institutet, pp. 91–630 (1998)
9. Pardàs, M., Bonafonte, A.: Facial animation parameters extraction and expression recognition using Hidden Markov Models. Signal Processing: Image Communication 17(9), 675–688 (2002)
10. Thiran, J.P., Marques, F., Bourlard, H. (eds.): Multimodal Signal Processing: Theory and Applications for Human-Computer Interaction. Elsevier, San Diego (2010)

11. Tian, Y., Kanade, T., Cohn, J.: Facial expression analysis. In: Handbook of Face Recognition, pp. 247–276 (2005)
12. Viola, P., Jones, M.: Rapid object detection using a boosted cascade of simple features. In: Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, CVPR 2001, vol. 1, pp. I-511–I-518 (2001)
13. Yang, M.H., Kriegman, D., Ahuja, N.: Detecting faces in images: A survey. IEEE Transactions on Pattern Analysis and Machine Intelligence 24(1), 34–58 (2002)

Part II
Information Systems Specification

Chapter 8
An Attempt to Use Self-Adapting Genetic Algorithms to Optimize Fuzzy Systems for Predicting from a Data Stream

Tadeusz Lasota, Magdalena Smętek, Bogdan Trawiński, and Grzegorz Trawiński

Abstract. In this chapter we present the continuation of our research into prediction from a data stream of real estate sales transactions using ensembles of regression models. The method consists in building models over the chunks of a data stream determined by a sliding time window and incrementally expanding an ensemble by systematically generated models in the course of time. The aged models are utilized to compose ensembles and their output is updated with trend functions reflecting the changes of prices in the market. In the study reported we attempted to incorporate self-adapting techniques into genetic fuzzy systems aimed to construct base models for property valuation. Six self-adapting genetic algorithms with varying mutation, crossover, and selection were developed and tested using real-world datasets. The analysis of experimental results was made employing non-parametric statistical techniques devised for multiple N×N comparisons.

Tadeusz Lasota
Wrocław University of Environmental and Life Sciences,
Department of Spatial Management, Wrocław, Poland
e-mail: tadeusz.lasota@up.wroc.pl

Magdalena Smętek · Bogdan Trawiński
Institute of Informatics, Wrocław University of Technology
Wyb. Wyspiańskiego 27, 50-370 Wrocław, Poland
e-mail: {magdalena.smetek,bogdan.trawinski}@pwr.edu.pl

Grzegorz Trawiński
Wrocław University of Technology, Faculty of Electronics,
Wyb. Wyspiańskiego 27, 50-370 Wrocław, Poland
e-mail: grzegorz.trawinsky@gmail.com

A. Zgrzywa et al. (eds.), *New Research in Multimedia and Internet Systems*,
Advances in Intelligent Systems and Computing 314, DOI: 10.1007/978-3-319-10383-9_8

8.1 Introduction

Mining data streams has attracted many researchers during the last decade. Processing data streams is a demanding question because it requires considering memory limitations, short processing times, and single scans of incoming data. Gaber in his overview paper distinguishes four categories of data stream mining methods: two-phase techniques, Hoeffding bound-based, symbolic approximation-based, and granularity-based ones [1]. The issue of concept drift which occurs when data distributions and definitions of target classes change over time has drawn a considerable interest of the data mining community [2], [3], [4]. Thorough reviews of ensemble based methods for dealing with concept drift in data streams are presented in [5], [6].

For a few years we have been working out and testing methods for generating regression models to assist with real estate appraisal based on fuzzy and neural approaches: i.e. genetic fuzzy systems and artificial neural networks as both single models [7] and ensembles built using various resampling techniques [8], [9], [10], [11], [12], [13]. An especially good performance revealed evolving fuzzy models applied to cadastral data [14], [15]. In this chapter we present the results of our further study on the methods to predict from a data stream of real estate sales transactions based on ensembles of genetic fuzzy systems [16], [17], [18]. Our former investigations on the use of evolutionary algorithms to optimize the architecture of fuzzy systems showed it is a laborious and time consuming process. Therefore we attempted to incorporate self-adapting techniques into genetic fuzzy systems aimed to generate models for property valuation.

Studies presented here are also a continuation of our research reported in [19] and [20]. We developed then genetic algorithms with self-adaptive mutation and crossover based on an idea developed by Maruo et al. [21] and tested them using some selected multimodal benchmark functions. The algorithms employing self-adaptive mutation and crossover revealed better performance than a traditional genetic one. We also employed successfully the self-adapting genetic algorithms for model selection to compose heterogeneous bagging ensembles [22].

8.2 Ensemble Approach to Predict from a Data Stream

Our ensemble approach to predict from a data stream lies in systematic building models over chunks of data and utilizing aged models to compose ensembles. The output produced by component models is corrected by means of trend functions reflecting the changes of prices in the market over time. The outline our approach to is illustrated in Fig. 8.1. The data stream is partitioned into data chunks of a constant length t_c. The sliding window, which length is a multiple of a data chunk, delineates training sets; in Fig. 8.1 it is double the chunk. We consider a point of time t_0 at which the current model was built over data that came in between time t_0–$2t_c$ and t_0. The models created earlier that have aged gradually are utilized to compose an ensemble so that the current test set is applied to each

component model. However, in order to compensate ageing, their output produced for the current test set is updated using trend functions determined over all data since the beginning of the stream; we denote them as *BegTrends*. As the functions to model the trends of price changes the polynomials of the degree from one to five were employed: *Ti(t)*, where *i* stands for the degree. The method of updating the prices of premises with the trends is based on the difference between a price and a trend value in a given time point. More detailed description of the approach presented in the chapter can be found in [23].

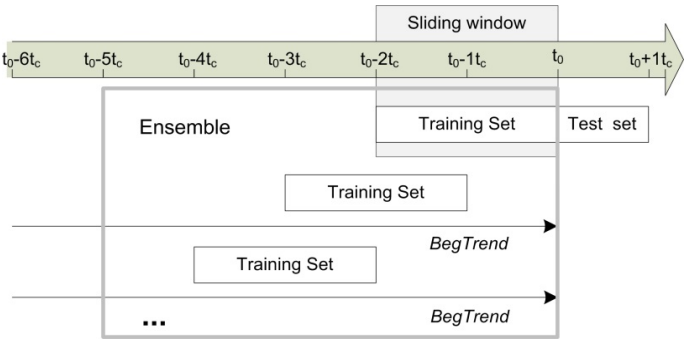


Fig. 8.1 Outline of ensemble approach to predict from a data stream

8.3 SAGA Methods Used in Experiments

We developed six self-adapting genetic algorithms (SAGA) with varying mutation (M), crossover (C), and selection (T) called respectively SAM, SAC, SAMC, SACT, SAMT, and SAMCT. In all variants of SAGAs constant length chromosomes were used and structures of their chromosomes are illustrated in Fig. 8.2.

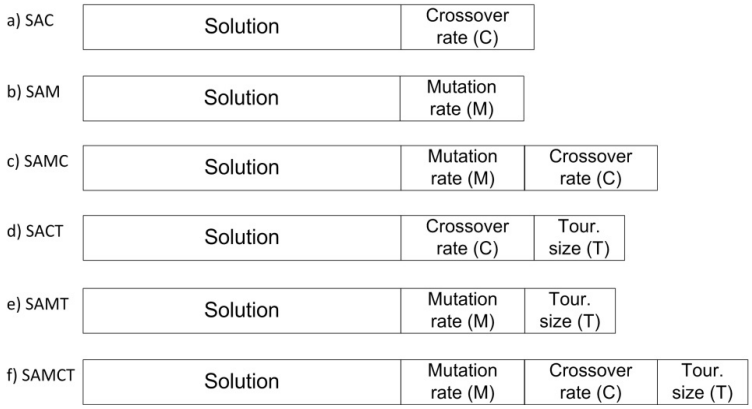


Fig. 8.2 Chromosome structures of individual self-adaptive genetic algorithms

To implement self-adaptive methods the main part of a chromosome which comprised the solution was extended to include mutation rates, crossover rates, and/or tournament size. The solution part comprised the representation of a fuzzy system. For each input variable three triangular and trapezoidal membership functions, and for output - five functions, were automatically determined by the symmetric division of the individual attribute domains. The evolutionary optimization process combined both learning the rule base and tuning the membership functions using real-coded chromosomes. Similar designs are described in [24], [25]. In order to implement the proposed self-adapting methods binary coding was employed. The mutation rate could be set to values from the bracket 0 to 0.3, and crossover rate from the range 0.5 to 1.0. Therefore, to encode the mutation rate 5 genes and crossover rate 7 genes were used. In turn, the tournament size was encoded to represent the range from 1 to 7. The detailed description of self-adapting mutation, crossover and tournament was presented in our paper [22]. The self-adapting parameters encoded in SAMCT chromosomes were depicted in Fig. 8.3.

solution	mutation rate					crossover rate							tournament size		
	0	1	1	0	1	0	0	0	1	1	0	1	1	0	1

Fig. 8.3 Self-adaptive parameters encoded in SAMCT chromosomes

8.4 Experimental Setup

The experiments were conducted with our system implemented in Matlab. The system was designed to carry out research into machine learning algorithms using various resampling methods and constructing and evaluating ensemble models for regression problems. We have extended our system to include functions of building ensembles over a data stream. The trends were modelled using the Matlab function *polyfit.*

Real-world dataset used in experiments was derived from a cadastral system and included records referring to residential premises transactions accomplished in one Polish big city within 14 years from 1998 to 2011. After selection and cleansing the final dataset counted 9795 samples. Due to the fact we had the exact date of each transaction we were able to order all instances in the dataset by time, forming a sort of a data stream. Four following attributes were pointed out as main price drivers by professional appraisers: usable area of a flat (*Area*), age of a building construction (*Age*), number of storeys in the building (Storeys), the distance of the building from the city centre (*Centre*), in turn, price of premises (*Price*) was the output variable.

The evaluating experiments were conducted for 36 points of time from 2001-01-01 to 2010-10-01, with the step of 3 months. Component models were built over training data delineated by the sliding windows of constant length of 12 months .The sliding window was shifted by one month along the data stream. The test datasets, current for a given time point, determined by the interval of 3 months

were applied to each ensemble. As the accuracy measure the root mean squared error (*RMSE*) was employed. The resulting output of the ensemble for a given time point was computed as the arithmetic mean of the results produced by the component models and corrected by corresponding trend functions.

We determined following parameters of our experiments including two phases:

1) Generating single SAMC models

- Set the length of the sliding window to 12 months, $t_w = 12$.
- Set the starting point of the sliding window, i.e. its right edge, to 2000-01-01 and the terminating point to 2010-10-01.
- Set the shift of the sliding window to 1 month, $t_s = 1$.
- Move the window from starting point to terminating point with the step $t_s = 1$.
- At each stage generate a *SAMC* from scratch over a training set delineated by the window. In total 108 single models were built for each *SAMC*.

2) Building SAMC ensembles

- Select a period to investigate the real estate market in Poland, i.e. 2002-2010.
- At the beginning of each quarter (t_0) build ensembles composed of 1, 6, 12, 18, and 24 ageing *SAMCs*. An ensemble is created in the way described in Section 8.2 with the shift equal to one month, $t_s = 1$.
- Take test sets actual for each t_0 over a period of 3 months, $t_t = 3$.
- Compute the output of individual *SAMCs* and update it using trend functions of degree from one to five determined for *BegTrends*.
- As the aggregation function of ensembles use the arithmetic mean.

The analysis of the results was performed using statistical methodology including nonparametric tests followed by post-hoc procedures designed especially for multiple $N \times N$ comparisons [26], [27], [28]. The routine starts with the nonparametric Friedman test, which detect the presence of differences among all algorithms compared. After the rejection of the null-hypotheses following nonparametric post-hoc procedures are applied in order to point out the particular pairs of algorithms which reveal significant differences: Nemenyi's, Holm's, Shaffer's, and Bergmann-Hommel's ones.

8.5 Experimental Results

To illustrate our extensive experiments, the performance of *SAC* ensembles comprehending 1, 6, 12, 18 and 24 models for *BegT4* trend functions is shown in Fig. 8.4. In turn, the accuracy of *SAC* ensembles comprising 24 models with corrected output using *BegT1, BegT2, BegT3, BegT4*, and *BegT5* trend functions and without output correction (*noT*) is depicted in Fig. 8.5. The values of *RMSE* are given in thousand PLN. However, the differences among the models are not visually apparent, therefore one should refer to statistical tests of significance.

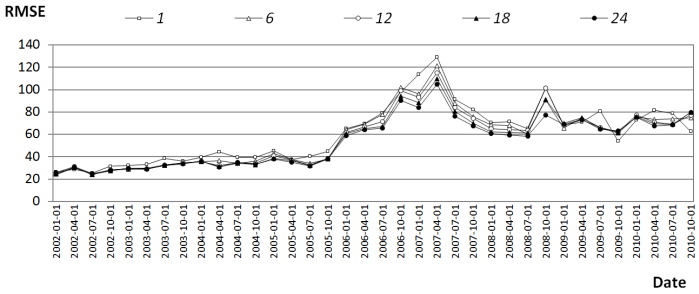


Fig. 8.4 Performance of *SAC* ensembles of different size for correction with Beg*T4*

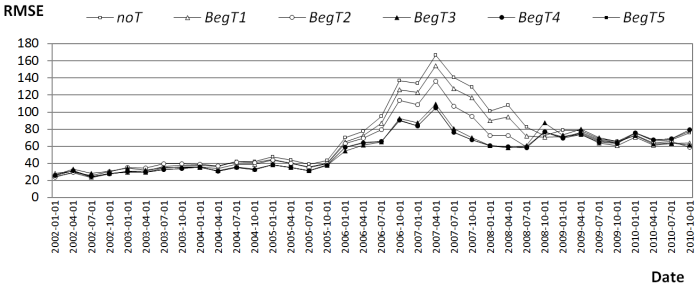


Fig. 8.5 Performance of *SAC* ensembles with correction using trend functions for Size=24

The Friedman test performed for individual *SAGA* methods over 36 observation points showed that there were significant differences among ensembles in each case. The test ranked the ensembles in respect of *RMSE* error measure and the lower position the better model. Average ranks of individual ensemble sizes for correction with *BegT4* trend function and for individual *SAGA* methods are placed in the columns of Table 8.1. Further statistical tests showed that the biggest ensembles composed of 24 models outperformed significantly the others. In turn, average ranks of individual ensembles of size 24 for polynomial trend functions of degrees from 1 to 5 and with no output correction and for individual *SAGA* methods are shown in the columns of Table 8.2. The lowest score achieved ensembles with *BegT4* correction. However, further statistical tests showed that the ensembles with *BegT3*, *BegT4*, and *BegT5* correction provided significantly better accuracy than the others. However, no significant differences were observed among ensembles with *BegT3*, *BegT4*, and *BegT5* correction. The models with no output correction reveal statistically worse performance than the ones corrected.

Table 8.1 Average rank positions of ensembles of different size for correction with *BegT4* trend function determined during Friedman test

Size	SAC	SAM	SAMC	SACT	SAMT	SAMCT
1	4.31	4.22	4.39	4.56	4.64	4.64
6	3.17	3.39	3.33	3.08	3.39	3.36
12	3.03	2.92	2.69	2.56	2.69	2.67
18	2.47	2.44	2.39	2.50	2.31	2.25
24	2.03	2.03	2.19	2.31	1.97	2.08

Table 8.2 Average rank positions of ensembles with correction using different trend functions for Size=24 determined during Friedman test

Trend	SAC	SAM	SAMC	SACT	SAMT	SAMCT
noT	4.92	5.14	4.97	5.17	5.03	5.08
BegT1	3.58	3.86	3.83	3.92	3.72	3.81
BegT2	4.08	3.89	3.83	3.81	3.97	3.89
BegT3	3.22	3.22	3.11	3.00	3.17	3.08
BegT4	2.50	2.31	2.53	2.39	2.42	2.44
BegT5	2.69	2.58	2.72	2.72	2.69	2.69

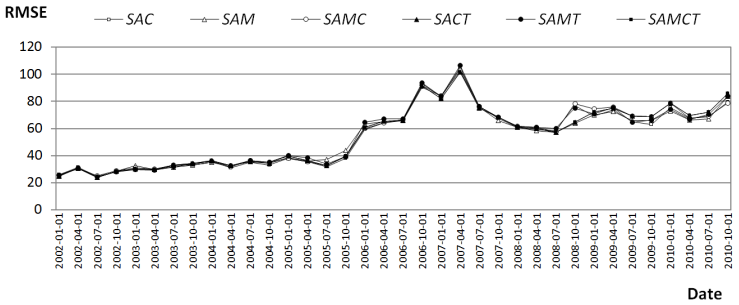


Fig. 8.6 Performance of ensembles of Size=24 for correction with BegT4 trend function

Based on the aforementioned results we compared all the *SAGA* methods for ensembles composed of 24 component models with output correction using *BegT4* trend functions. The performance of individual *SAGAs* is illustrated in Fig. 8.6 and again we employed the statistical tests of significance. Average rank positions of ensembles of Size=24 for correction with *BegT4* trend function determined during Friedman test are shown in Table 8.3. Adjusted p-values for Nemenyi's, Holm's, Shaffer's, and Bergmann-Hommel's post-hoc procedures for $N \times N$ comparisons for all possible pairs of *SAGA* methods are shown in Table 8.4. The p-values indicating the statistically significant differences between given pairs of algorithms are marked with italics. The significance level considered for the null hypothesis rejection was 0.05. Significant differences were observed only for three pairs of *SAGAs*. *SAC* ensembles surpassed the *SAMC* and *SAMT* ones and *SAM* ensembles outperformed the *SAMT* ones.

Table 8.3 Average rank positions of ensembles of Size=24 for correction with *BegT4* trend function determined during Friedman test

1st	2nd	3rd	4th	5th	6th
SAC (2.58)	SAM (2.92)	SACT (3.64)	SAMCT (3.75)	SAMC (3.94)	SAMT (4.17)

Table 8.4 Adjusted p-values for $N \times N$ comparisons of ensembles of Size=24 for correction with *BegT4* trend function for all 15 hypotheses

Method vs Method	pNeme	pHolm	pShaf	pBerg
SAC vs SAMT	*0.0049*	*0.0049*	*0.0049*	*0.0049*
SAC vs SAMC	*0.0304*	*0.0283*	*0.0202*	*0.0202*
SAM vs SAMT	0.0688	0.0596	*0.0459*	*0.0459*
SAC vs SAMCT	0.1223	0.0978	0.0815	0.0571
SAC vs SACT	0.2501	0.1834	0.1668	0.1001
SAM vs SAMC	0.2965	0.1976	0.1976	0.1186
SAM vs SAMCT	0.8817	0.5290	0.4115	0.2351
SAM vs SACT	1.0000	0.8116	0.7102	0.4058
SACT vs SAMT	1.0000	1.0000	1.0000	1.0000
SAMCT vs SAMT	1.0000	1.0000	1.0000	1.0000
SAC vs SAM	1.0000	1.0000	1.0000	1.0000
SAMC vs SACT	1.0000	1.0000	1.0000	1.0000
SAMC vs SAMT	1.0000	1.0000	1.0000	1.0000
SAMC vs SAMCT	1.0000	1.0000	1.0000	1.0000
SACT vs SAMCT	1.0000	1.0000	1.0000	1.0000

8.6 Conclusions

The results of our further research into the method to predict from a data stream of real estate sales transactions based on ensembles of regression models are reported in the chapter. Our approach consists in incremental expanding an ensemble by models built from scratch over successive chunks of a data stream determined by a sliding window. In order to counterbalance ageing the results provided by component models for the current test dataset are updated with trend functions which reflect the market dynamics. As the base machine learning algorithms we employed fuzzy systems generated and tuned by self-adapting genetic algorithms.

Extensive evaluating experiments were conducted using real-world data of sales transactions taken from a cadastral system They aimed at examining the impact of the number of aged models used to compose an ensemble and the influence of degree of polynomial correction functions on the predictive accuracy. Moreover, we compared the performance of six different SAGA algorithms.

The results proved the usefulness of ensemble approach incorporating the correction of individual component model output. The application of SAGA fuzzy models was also successful. As for correcting the output of component models, the need to apply trend functions to update the results provided by ageing models is indisputable. However, the selection the most suitable trend function in terms of the polynomial degree has not been definitely resolved. In majority of cases the trend functions of higher degree, i.e. three, four, and five provided better accuracy. However, the differences were not statistically significant. Therefore, further study is needed into the selection of correcting functions dynamically depending on the nature of price changes. Moreover, we plan to tune empirically the parameters of self-adaptive crossover, mutation, and selection as well as to compare the performance of SAGA fuzzy models with the ones produced by other machine learning methods such as artificial neural networks, support vector machines, and classical genetic fuzzy systems.

Acknowledgments. This work was partially supported by the National Science Centre under grant no. N N516 483840 and the "Młoda Kadra" funds of Wrocław University of Technology.

References

1. Gaber, M.M.: Advances in data stream mining. Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery 2(1), 79–85 (2012)
2. Tsymbal, A.: The problem of concept drift: Definitions and related work. Technical Report. Department of Computer Science, Trinity College, Dublin (2004)
3. Sobolewski, P., Woźniak, M.: Concept Drift Detection and Model Selection with Simulated Recurrence and Ensembles of Statistical Detectors. Journal for Universal Computer Science 19(4), 462–483 (2013)
4. Brzeziński, D., Stefanowski, J.: Reacting to Different Types of Concept Drift: The Accuracy Updated Ensemble Algorithm. IEEE Transactions on Neural Networks and Learning Systems 25(1), 81–94 (2014)
5. Kuncheva, L.I.: Classifier ensembles for changing environments. In: Roli, F., Kittler, J., Windeatt, T. (eds.) MCS 2004. LNCS, vol. 3077, pp. 1–15. Springer, Heidelberg (2004)
6. Minku, L.L., White, A.P., Yao, X.: The Impact of Diversity on Online Ensemble Learning in the Presence of Concept Drift. IEEE Transactions on Knowledge and Data Engineering 22(5), 730–742 (2010)
7. Król, D., Lasota, T., Trawiński, B., Trawiński, K.: Comparison of Mamdani and TSK Fuzzy Models for Real Estate Appraisal. In: Apolloni, B., Howlett, R.J., Jain, L. (eds.) KES 2007/ WIRN 2007, Part III. LNCS (LNAI), vol. 4694, pp. 1008–1015. Springer, Heidelberg (2007)
8. Lasota, T., Telec, Z., Trawiński, B., Trawiński, K.: Exploration of Bagging Ensembles Comprising Genetic Fuzzy Models to Assist with Real Estate Appraisals. In: Corchado, E., Yin, H. (eds.) IDEAL 2009. LNCS, vol. 5788, pp. 554–561. Springer, Heidelberg (2009)
9. Lasota, T., Telec, Z., Trawiński, B., Trawiński, K.: A Multi-agent System to Assist with Real Estate Appraisals Using Bagging Ensembles. In: Nguyen, N.T., Kowalczyk, R., Chen, S.-M. (eds.) ICCCI 2009. LNCS (LNAI), vol. 5796, pp. 813–824. Springer, Heidelberg (2009)
10. Graczyk, M., Lasota, T., Trawiński, B., Trawiński, K.: Comparison of Bagging, Boosting and Stacking Ensembles Applied to Real Estate Appraisal. In: Nguyen, N.T., Le, M.T., Świątek, J. (eds.) ACIIDS 2010, Part II. LNCS (LNAI), vol. 5991, pp. 340–350. Springer, Heidelberg (2010)
11. Krzystanek, M., Lasota, T., Telec, Z., Trawiński, B.: Analysis of Bagging Ensembles of Fuzzy Models for Premises Valuation. In: Nguyen, N.T., Le, M.T., Świątek, J. (eds.) ACIIDS 2010, Part II. LNCS (LNAI), vol. 5991, pp. 330–339. Springer, Heidelberg (2010)
12. Kempa, O., Lasota, T., Telec, Z., Trawiński, B.: Investigation of bagging ensembles of genetic neural networks and fuzzy systems for real estate appraisal. In: Nguyen, N.T., Kim, C.-G., Janiak, A. (eds.) ACIIDS 2011, Part II. LNCS (LNAI), vol. 6592, pp. 323–332. Springer, Heidelberg (2011)
13. Lasota, T., Telec, Z., Trawiński, G., Trawiński, B.: Empirical Comparison of Resampling Methods Using Genetic Fuzzy Systems for a Regression Problem. In: Yin, H., Wang, W., Rayward-Smith, V. (eds.) IDEAL 2011. LNCS, vol. 6936, pp. 17–24. Springer, Heidelberg (2011)

14. Lasota, T., Telec, Z., Trawiński, B., Trawiński, K.: Investigation of the eTS Evolving Fuzzy Systems Applied to Real Estate Appraisal. Journal of Multiple-Valued Logic and Soft Computing 17(2-3), 229–253 (2011)
15. Lughofer, E., Trawiński, B., Trawiński, K., Kempa, O., Lasota, T.: On Employing Fuzzy Modeling Algorithms for the Valuation of Residential Premises. Information Sciences 181, 5123–5142 (2011)
16. Trawiński, B., Lasota, T., Smętek, M., Trawiński, G.: An Analysis of Change Trends by Predicting from a Data Stream Using Genetic Fuzzy Systems. In: Nguyen, N.-T., Hoang, K., Jędrzejowicz, P. (eds.) ICCCI 2012, Part I. LNCS (LNAI), vol. 7653, pp. 220–229. Springer, Heidelberg (2012)
17. Trawiński, B., Lasota, T., Smętek, M., Trawiński, G.: Weighting Component Models by Predicting from Data Streams Using Ensembles of Genetic Fuzzy Systems. In: Larsen, H.L., Martin-Bautista, M.J., Vila, M.A., Andreasen, T., Christiansen, H. (eds.) FQAS 2013. LNCS (LNAI), vol. 8132, pp. 567–578. Springer, Heidelberg (2013)
18. Trawiński, B., Smętek, M., Lasota, T., Trawiński, G.: Evaluation of Fuzzy System Ensemble Approach to Predict from a Data Stream. In: Nguyen, N.T., Attachoo, B., Trawiński, B., Somboonviwat, K. (eds.) ACIIDS 2014, Part II. LNCS (LNAI), vol. 8398, pp. 137–146. Springer, Heidelberg (2014)
19. Smętek, M., Trawiński, B.: Investigation of Genetic Algorithms with Self-adaptive Crossover, Mutation, and Selection. In: Corchado, E., Kurzyński, M., Woźniak, M. (eds.) HAIS 2011, Part I. LNCS (LNAI), vol. 6678, pp. 116–123. Springer, Heidelberg (2011)
20. Smętek, M., Trawiński, B.: Investigation of Self-adapting Genetic Algorithms using Some Multimodal Benchmark Functions. In: Jędrzejowicz, P., Nguyen, N.T., Hoang, K. (eds.) ICCCI 2011, Part I. LNCS, vol. 6922, pp. 213–223. Springer, Heidelberg (2011)
21. Maruo, M.H., Lopes, H.S., Delgado, M.R.: Self-Adapting Evolutionary Parameters Encoding Aspects for Combinatorial Optimization Problems. In: Raidl, G.R., Gottlieb, J. (eds.) EvoCOP 2005. LNCS, vol. 3448, pp. 154–165. Springer, Heidelberg (2005)
22. Smętek, M., Trawiński, B.: Selection of Heterogeneous Fuzzy Model Ensembles Using Self-adaptive Genetic Algorithms. New Generation Computing 29(3), 309–327 (2011)
23. Trawiński, B.: Evolutionary Fuzzy System Ensemble Approach to Model Real Estate Market based on Data Stream Exploration. Journal of Universal Computer Science 19(4), 539–562 (2013)
24. Cordón, O., Herrera, F.: A Two-Stage Evolutionary Process for Designing TSK Fuzzy Rule-Based Systems. IEEE Tr. on Sys., Man and Cyber., Part B 29(6), 703–715 (1999)
25. Król, D., Lasota, T., Trawiński, B., Trawiński, K.: Investigation of evolutionary optimization methods of TSK fuzzy model for real estate appraisal. International Journal of Hybrid Intelligent Systems 5(3), 111–128 (2008)
26. Demšar, J.: Statistical comparisons of classifiers over multiple data sets. Journal of Machine Learning Research 7, 1–30 (2006)
27. García, S., Herrera, F.: An Extension on "Statistical Comparisons of Classifiers over Multiple Data Sets" for all Pairwise Comparisons. Journal of Machine Learning Research 9, 2677–2694 (2008)
28. Trawiński, B., Smętek, M., Telec, Z., Lasota, T.: Nonparametric Statistical Analysis for Multiple Comparison of Machine Learning Regression Algorithms. International Journal of Applied Mathematics and Computer Science 22(4), 867–881 (2012)

Chapter 9
Estimation of the Level of Disturbance in Time Series Using a Median Filter

Jakub Peksinski, Grzegorz Mikolajczak, and Janusz Pawel Kowalski

Abstract. Information about the level of signal interference, allows you to select the appropriate method pre-processing information. Assuming that the disturbance is a process additive, a normal distribution can do this using the smoothing filters, and in particular the median filter. This chapter presents a method of estimating the level of disturbance, based on median filtration and the assumption that the smoothing process applies to noise, exclusively. The knowledge of a noise reduction coefficient enables the determining of an estimated quantity.

9.1 Introduction

The analysis of time series follows two primary purposes: a) discovering the nature of the phenomenon represented by a sequence of observations, and b) forecasting (predicting) future quantities of a time series. In the context of both purposes, it is required to identify and describe the elements of a time series, in a more or less formal manner – it is possible to distinguish the following components of a time series: a tendency for development (trend), seasonal fluctuations, cyclical fluctuations (business fluctuations), and random fluctuations (random component, noise) [1].

Jakub Peksinski · Grzegorz Mikolajczak
Faculty of Electrical Engineering,
West Pomeranian University of Technology
ul. Sikorskiego 37, 70-313 Szczecin, Poland
e-mail: jpeksinski@zut.edu.pl

Janusz Pawel Kowalski
Faculty of Medicine,
Pomeranian Medical University
ul. Rybacka 1, 70-204 Szczecin, Poland

A. Zgrzywa et al. (eds.), *New Research in Multimedia and Internet Systems*,
Advances in Intelligent Systems and Computing 314, DOI: 10.1007/978-3-319-10383-9_9

Reduction of undesirable disturbances is one of the problems encountered in the analysis of time series. It is quite difficult to choose the most effective method, because of the multitude of methods and algorithms that feature varied complexity and efficiency of eliminating noise, which in turn are based on the nature and level of disturbance. Provided that we have certain knowledge or have grounds to make certain assumptions, as to the nature and form of disturbances, we are able to select an appropriate method, which would ensure an optimum quality for the analysis of a time series [2].

The problem of noise estimation most often consists in the determining of a standard disturbance deflection σ_n, or variance σ_n^2, in a given time series, assuming that the random component is an additive and stationary process not correlated with a signal, and has an average value of zero, and normal distribution [3, 4].

$$x_k = s_k + n_k \tag{9.1}$$

where: s_k – usable signal, x_k – disturbed signal, n_k – disturbances with normal distribution, average value of zero $E(n)=0$, and variance $V(n)=\sigma_n^2$.

The methods of estimating the level of disturbances may be generally divided into two groups. The first method is based on the filtration of a disturbed signal and the assumption that the filtered signal is original. That leads to the determining of a standard disturbance deflection σ_n, based on the difference between the disturbed and filtered signals. The other method of estimating disturbance consists in finding a place in a time series, in which there is no usable signal present, so as to enable the determining the level of disturbance.

9.2 Idea Behind an Estimation Method

It is advised to use median filtration to evaluate a disturbance, and its principle of operation has been described below. A standard median filter is obtained, as the result of arranging input samples x_i, in the ascending order, and selecting the median value as the input one, if the number of samples is odd. Otherwise, the input sample for a median filter will be any value placed between the two median values (it is usually an arithmetic average). If we mark $x=(x_1, x_2,..., x_N)$ as a set of observations, and its median as $med(x)$, then the afore-mentioned relationship can be represented by the following equation [4]:

$$med\,(x) = \begin{cases} x_{v+1}, & for \quad N = 2 \cdot v + 1 \\ \dfrac{x_v + x_{v+1}}{2}, & for \quad N = 2 \cdot v \end{cases} \tag{9.2}$$

Table 9.1 Boundary values of disturbance variance, at the output for moving average and median filters, using different models of disturbance and maintaining consistency of suppression; N – the number of elements in a window

Disturbance model	Moving average	Median
Uniform distribution $f(x)=\begin{cases}\frac{1}{2\sigma\sqrt{3}}, & -\sigma\sqrt{3}\le x\le\sigma\sqrt{3}\\ 0\end{cases}$	$\frac{\sigma^2}{N}$	$\frac{3\sigma^2}{N+2}$
Gaussian distribution (normal) $f(x)=\frac{1}{\sigma\sqrt{2\pi}}e^{-x^2/2\sigma^2}$	$\frac{\sigma^2}{N}$	$\frac{\pi\sigma^2}{2N}$
Laplace distribution (biexpotential) $f(x)=\frac{1}{\sigma\sqrt{2}}e^{-\sqrt{2}\lvert x\rvert/\sigma}$	$\frac{\sigma^2}{N}$	$\frac{\sigma^2}{N}$

The process of smoothing leads to the reduction of disturbance variances to the values defined by the relationships demonstrated in Table 9.1 [5]. The knowledge of such relationships is a basis for the developing of a method of estimating the level of disturbance.

It is assumed that the smoothing process leads only to the reduction of noise variance, according to the relationship demonstrated in Table 9.1, while the variance of usable signal $V(s)=\sigma_s^2$ remains intact.

Assuming that the observed time series has form (9.1) and taking into account that noise and usable signal are not correlated, the variance of an input signal is:

$$V(x)=\sigma_s^2+\sigma_n^2 \tag{9.3}$$

Assuming that the smoothing only reduces the variance of disturbance, we can present the relationship concerning the variance of an input signal, as:

$$V(y)=\sigma_s^2+\frac{\pi}{2N}\sigma_n^2 \tag{9.4}$$

If we denote: $V(y)=V_{med}$ – variance after median filtration, and $V(x)=V_0$ – signal variance without smoothing, the result is:

$$V_{med}=\sigma_s^2+\frac{\pi}{2N}\sigma_n^2 \tag{9.5}$$

$$V_0=\sigma_s^2+\sigma_n^2 \Rightarrow \sigma_s^2=V_0-\sigma_n^2 \tag{9.6}$$

If we substitute (9.6) with (9.5), then, after transformations, it is possible to obtain a relationship for disturbance variance (9.7), determined based on the knowledge of the variance of a disturbed signal and the variance of a signal after median filtration:

$$\sigma_n^2 = \frac{2N}{2N - \pi}(V_0 - V_{med}) \tag{9.7}$$

The method of determining a variance of disturbance (9.7), as presented above, requires the knowledge of variances of input and output signals, exclusively, the values of which can be calculated [6]. This enables a simple determining of the level of disturbances.

9.3 Test Results

The suggested method (9.7) has been tested based on the smoothing of a series of samples $\{x_k\}$, generated from a harmonic signal with a trend $\{s_k\}$, demonstrated in Fig. 9.1, disturbed with noise $\{n_k\}$, having a normal distribution (Gaussian), average value of zero $E(n)=0$, and variance $V(n)=\sigma_n^2$, where the value of standard deflection σ_n was changed, according to $\sigma_n=0\text{-}20$ range.

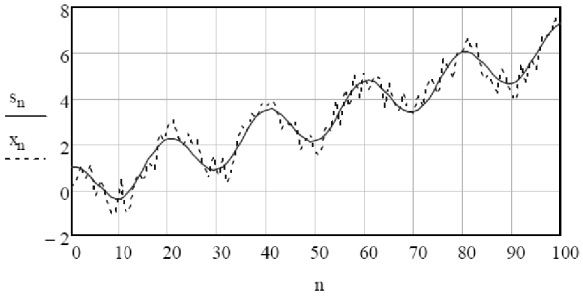


Fig. 9.1 Series of samples $\{x_k\}$, generated from a harmonic signal with a trend $\{s_k\}$

Estimations of disturbance have been determined for different numbers of elements in a series (K), and for a changeable number of elements in a window (N) of a median filter (9.2).

$$x_k = s_k + n_k \;\; where \;\; s_k = \cos\left(\tfrac{10\cdot\pi}{K}\cdot k\right) + \tfrac{2\cdot\pi}{K}\cdot k \qquad k \in (0; K) \tag{9.8}$$

For a single signal $\{s_k\}$, for each value σ_n generated $M = 10$ independent disturbances for which the estimated value obtained $\overline{\sigma_n}$:

$$\overline{\sigma}_n = \frac{1}{M}\cdot\sum_m \sigma_{n,m} \quad for \quad m = 0 \ldots M - 1 \tag{9.9}$$

Then, for each value of σ_n determined μ_n load estimator μ_n and standard deviation of the estimator SD_n;

$$\mu_n = \overline{\sigma}_n - \sigma_n \quad ; \quad SD_n = \sqrt{\frac{1}{M}\cdot\sum_m \left(\sigma_{n,m} - \overline{\sigma}_n\right)^2} \tag{9.10}$$

Determined the mean square error MSE estimation of the form:

$$MSE = \sqrt{\frac{1}{M} \cdot \sum_{m} \left(\sigma_{n,m} - \sigma_n\right)^2} \tag{9.11}$$

For the generation of noise and calculations used a computer program Mathcad 14. Calculations were made for a number of *K = 100*, *K = 500* and *K = 1000* samples. The results are shown in Tables 9.2 – 9.4.

Table 9.2 The values of the individual coefficients (9.9) – (9.11) obtained in the estimation of noise the proposed method (9.7) for K = 100 samples

σ_n	N=3				N=5				N=7			
	$\overline{\sigma_n}$	μ_n	SD_n	MSE	$\overline{\sigma_n}$	μ_n	SD_n	MSE	$\overline{\sigma_n}$	μ_n	SD_n	MSE
0	0.15	0.15	0	0.15	0.27	0.27	0	0.27	0.36	0.36	0	0.36
1	1.16	0.16	0.12	0.2	1.05	0.05	0.1	0.12	1.11	0.11	0.13	0.17
2	2.03	0.03	0.22	0.22	2.02	0.02	0.14	0.14	2.05	0.05	0.12	0.13
3	3.26	0.26	0.24	0.35	3.15	0.15	0.14	0.2	3.17	0.17	0.15	0.22
4	4.39	0.39	0.41	0.57	3.98	-0.02	0.26	0.26	4.09	0.09	0.12	0.15
5	5.32	0.32	0.32	0.45	5.14	0.14	0.2	0.25	5.16	0.16	0.2	0.26
6	6.3	0.3	0.35	0.46	6.09	0.09	0.26	0.27	6.15	0.15	0.2	0.25
7	7.45	0.45	0.38	0.59	7.15	0.15	0.32	0.36	7.11	0.11	0.16	0.2
8	8.22	0.22	0.8	0.83	8.21	0.21	0.27	0.34	8.11	0.11	0.36	0.38
9	9.47	0.47	0.75	0.89	9.36	0.36	0.25	0.44	9.27	0.27	0.2	0.34
10	11	1	0.43	1.09	10.15	0.15	0.39	0.42	10.09	0.09	0.32	0.33
11	11.98	0.98	0.34	1.04	11.04	0.04	0.61	0.62	11.11	0.11	0.44	0.45
12	13.19	1.19	0.56	1.32	12.53	0.53	0.4	0.66	11.9	-0.1	0.53	0.54
13	14.18	1.18	1.18	1.67	13.46	0.46	0.46	0.65	13.23	0.23	0.41	0.47
14	15.3	1.3	1.09	1.7	14.33	0.33	0.37	0.5	14.35	0.35	0.21	0.41
15	15.68	0.68	0.98	1.19	15.35	0.35	0.82	0.89	15.2	0.2	0.63	0.66
16	17.18	1.18	0.9	1.49	16.43	0.43	0.61	0.75	15.94	-0.06	0.66	0.66
17	18.21	1.21	1.06	1.61	17.31	0.31	0.96	1.01	17.37	0.37	0.62	0.72
18	19.7	1.7	0.95	1.95	18.45	0.45	0.42	0.61	18.16	0.16	0.57	0.59
19	20.04	1.04	1.7	1.99	19.45	0.45	0.69	0.83	19.29	0.29	0.57	0.64
20	20.97	0.97	1.29	1.61	20.73	0.73	0.76	1.06	19.99	-0.01	0.49	0.49

The presented method was compared with noise estimation method called Delta Test (DT). The Delta Test (DT), firstly introduced by Pi and Peterson [7] for time series and proposed for variable selection in [8], is a technique to estimate the variance of the noise, or the mean squared error (MSE), that can be achieved without over fitting. Given N input-output pairs $(x_i, y_i) \in R^d \times R$, the relationship between x_i and y_i can be expressed as: $y_i = f(x_i)+r_i$, $i=1,...,N$, where f is the unknown function and r is the noise.

The DT estimates the variance of the noise r. The DT can be interpreted as a particularization of the Gamma Test [9] considering only the first nearest neighbor. Let us denote the first nearest neighbor of a point x_i in the R^d space as $x_{NN(i)}$. The nearest neighbor formulation of the DT estimates $Var[r]$ by

$$Var[r] \approx \delta = \frac{1}{2N} \sum_{i=1}^{N} \left(y_i - y_{NN(i)}\right)^2 \; with \quad Var[\delta] \to 0 \; for \; N \to \infty \tag{9.12}$$

where $y_{NN(i)}$ is the output of $x_{NN(i)}$.

The estimation results for the interference by DT for $K = 500$ are shown in Table 9.5.

Table 9.3 The values of the individual coefficients (9.9) – (9.11) obtained in the estimation of noise the proposed method (9.7) for K = 500 samples

σ_n	N=3				N=5				N=7			
	$\overline{\sigma_n}$	μ_n	SD_n	MSE	$\overline{\sigma_n}$	μ_n	SD_n	MSE	$\overline{\sigma_n}$	μ_n	SD_n	MSE
0	0.02	0.02	0	0.02	0.03	0.03	0	0.03	0.05	0.05	0	0.05
1	1.09	0.09	0.06	0.1	1.02	0.02	0.05	0.06	0.98	-0.02	0.07	0.07
2	2.17	0.17	0.11	0.2	2.04	0.04	0.06	0.07	2.04	0.04	0.07	0.08
3	3.21	0.21	0.14	0.25	3.07	0.07	0.09	0.11	3.01	0.01	0.11	0.11
4	4.39	0.39	0.09	0.4	4.05	0.05	0.11	0.12	4.02	0.02	0.08	0.08
5	5.39	0.39	0.11	0.4	5.16	0.16	0.11	0.2	5.04	0.04	0.09	0.1
6	6.39	0.39	0.19	0.43	6.17	0.17	0.1	0.2	6.02	0.02	0.14	0.14
7	7.64	0.64	0.17	0.66	7.14	0.14	0.12	0.18	7.07	0.07	0.14	0.15
8	8.7	0.7	0.19	0.72	8.08	0.08	0.2	0.22	8.09	0.09	0.09	0.13
9	9.65	0.65	0.28	0.71	9.2	0.2	0.17	0.26	9.1	0.1	0.16	0.19
10	10.66	0.66	0.27	0.71	10.32	0.32	0.17	0.36	10.13	0.13	0.17	0.21
11	11.87	0.87	0.27	0.91	11.21	0.21	0.25	0.33	11.17	0.17	0.25	0.3
12	12.89	0.89	0.37	0.96	12.19	0.19	0.18	0.26	12.13	0.13	0.16	0.21
13	13.93	0.93	0.69	1.15	13.3	0.3	0.32	0.44	13.06	0.06	0.23	0.24
14	15.28	1.28	0.34	1.32	14.43	0.43	0.26	0.51	14.12	0.12	0.21	0.24
15	16.3	1.3	0.31	1.34	15.31	0.31	0.29	0.43	15.07	0.07	0.21	0.22
16	17.58	1.58	0.3	1.61	16.16	0.16	0.2	0.26	16.22	0.22	0.19	0.29
17	18.12	1.12	0.51	1.24	17.34	0.34	0.41	0.53	17.05	0.05	0.3	0.31
18	19.14	1.14	0.24	1.16	18.48	0.48	0.18	0.52	18.16	0.16	0.22	0.27
19	20.47	1.47	0.55	1.57	19.48	0.48	0.38	0.62	19.12	0.12	0.23	0.26
20	21.07	1.07	0.63	1.24	20.3	0.3	0.39	0.5	20.14	0.14	0.28	0.31

Table 9.4 The values of the individual coefficients (9.9) – (9.11) obtained in the estimation of noise the proposed method (9.7) for K = 1000 samples

σ_n	N=3				N=5				N=7			
	$\overline{\sigma_n}$	μ_n	SD_n	MSE	$\overline{\sigma_n}$	μ_n	SD_n	MSE	$\overline{\sigma_n}$	μ_n	SD_n	MSE
0	0.01	0.01	0	0.01	0.02	0.02	0	0.02	0.02	0.02	0	0.02
1	1.05	0.05	0.05	0.08	1.01	0.01	0.04	0.04	1.03	0.03	0.04	0.05
2	2.18	0.18	0.06	0.19	2.02	0.02	0.03	0.04	2.01	0.01	0.03	0.04
3	3.24	0.24	0.08	0.25	3.04	0.04	0.05	0.07	3.05	0.05	0.08	0.09
4	4.32	0.32	0.1	0.34	4.04	0.04	0.08	0.09	4.03	0.03	0.07	0.07
5	5.33	0.33	0.13	0.35	5.11	0.11	0.09	0.14	5.03	0.03	0.08	0.08
6	6.59	0.59	0.14	0.6	6.11	0.11	0.05	0.12	6.03	0.03	0.06	0.07
7	7.56	0.56	0.17	0.59	7.14	0.14	0.05	0.15	7.05	0.05	0.08	0.1
8	8.68	0.68	0.17	0.7	8.18	0.18	0.09	0.2	8.07	0.07	0.1	0.12
9	9.66	0.66	0.09	0.67	9.22	0.22	0.17	0.28	9.03	0.03	0.06	0.07
10	10.72	0.72	0.15	0.74	10.19	0.19	0.1	0.22	10.08	0.08	0.09	0.12
11	11.87	0.87	0.17	0.89	11.27	0.27	0.17	0.32	11.1	0.1	0.16	0.19
12	12.77	0.77	0.26	0.81	12.31	0.31	0.15	0.34	12.07	0.07	0.14	0.16
13	14	1	0.28	1.04	13.22	0.22	0.23	0.32	13.12	0.12	0.16	0.2
14	15.1	1.1	0.26	1.13	14.28	0.28	0.11	0.3	14.2	0.2	0.23	0.31
15	16.13	1.13	0.29	1.17	15.21	0.21	0.22	0.3	15.16	0.16	0.16	0.23
16	17.29	1.29	0.27	1.32	16.26	0.26	0.18	0.32	16.13	0.13	0.15	0.2
17	18.26	1.26	0.33	1.31	17.32	0.32	0.2	0.38	17.12	0.12	0.22	0.25
18	19.31	1.31	0.21	1.33	18.17	0.17	0.22	0.28	18.35	0.35	0.21	0.41
19	20.62	1.62	0.43	1.67	19.36	0.36	0.22	0.42	19.18	0.18	0.25	0.31
20	21.44	1.44	0.43	1.5	20.41	0.41	0.2	0.46	20.29	0.29	0.27	0.4

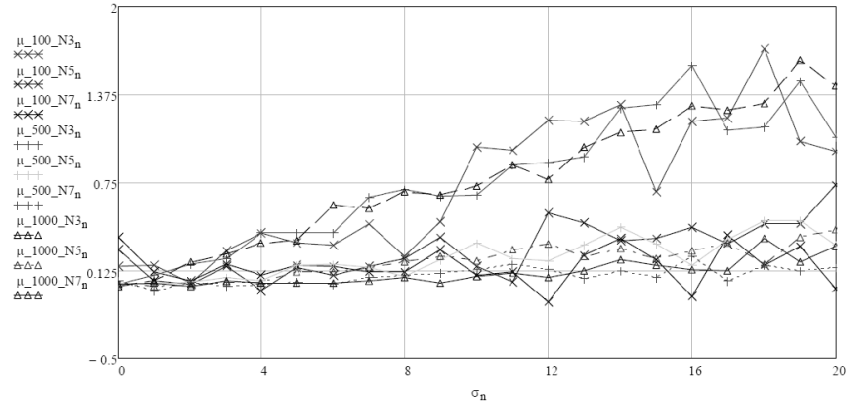


Fig. 9.2 The values of the load factor estimator μ_n for the proposed method (9.7) with different number of samples K and the size of the window N

Having analyzed the obtained results of disturbance estimation (Tables 9.2–9.4), it can be stated that all cases have produced satisfactory results. The mean values of the estimated disturbance $\overline{\sigma_n}$ (9.9), were burdened with an error not exceeding 10%. Only for the level of interference σ_n*=1-2,* was larger. However for the level of noise σ_n*>10,* was lower than 5%. The values obtained independently of the quantity of samples are comparable. The proposed noise estimation method (9.7) is characterized by a small sniffing obtained results, especially for the larger number of samples of a number of K = 500 and K = 1000.

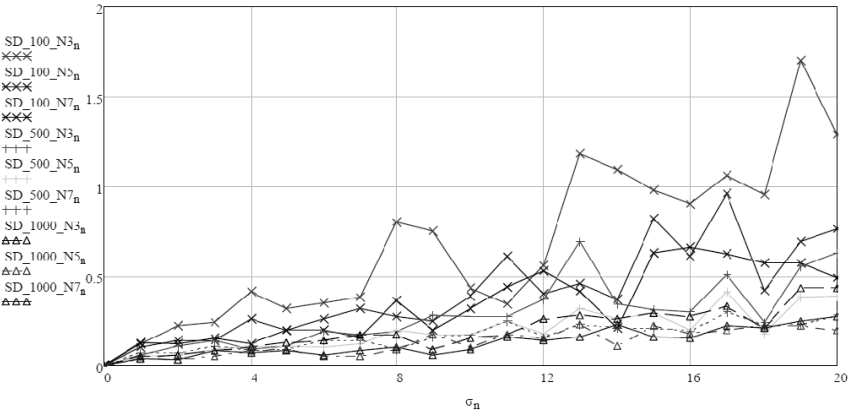


Fig. 9.3 A standard deviation of the estimator SD_n for the proposed method (9.7) with different number of samples K and the size of the window N

This can be better seen by analyzing the graph of the load estimator μ_n in Fig. 2 and the standard deviation of the estimator of SD_n in Fig. 3 in the functions is interference σ_n. Analyzing the results obtained by the DT can be seen that in each case the results were less than the level of distortion.

Table 9.5 The values of the individual coefficients (9.9) – (9.11) obtained in the estimation of noise DT method (9.12) for K = 500 samples

σ_n	N=3				N=5				N=7			
	$\sqrt{\delta}$	μ_n	SD_n	MSE	$\sqrt{\delta}$	μ_n	SD_n	MSE	$\sqrt{\delta}$	μ_n	SD_n	MSE
0	0.13	0.13	0	0.13	0.23	0.23	0	0.23	0.32	0.32	0	0.32
1	0.79	-0.21	0.03	0.21	0.91	-0.09	0.02	0.09	0.96	-0.04	0.02	0.04
2	1.64	-0.36	0.09	0.37	1.77	-0.23	0.05	0.23	1.82	-0.18	0.07	0.19
3	2.39	-0.61	0.08	0.61	2.66	-0.34	0.07	0.34	2.7	-0.3	0.1	0.32
4	3.23	-0.77	0.13	0.78	3.49	-0.51	0.13	0.53	3.58	-0.42	0.07	0.42
5	4.12	-0.88	0.21	0.9	4.49	-0.51	0.13	0.53	4.58	-0.42	0.11	0.44
6	4.98	-1.02	0.25	1.05	5.41	-0.59	0.13	0.61	5.47	-0.53	0.19	0.56
7	5.76	-1.24	0.26	1.27	6.06	-0.94	0.17	0.95	6.37	-0.63	0.19	0.66
8	6.41	-1.59	0.34	1.62	7.12	-0.88	0.24	0.91	7.28	-0.72	0.17	0.74
9	7.34	-1.66	0.44	1.72	7.84	-1.16	0.23	1.19	8.37	-0.63	0.21	0.67
10	7.97	-2.03	0.26	2.05	8.8	-1.2	0.32	1.24	9.26	-0.74	0.33	0.81
11	9.09	-1.91	0.25	1.92	9.71	-1.29	0.3	1.32	10.08	-0.92	0.42	1.01
12	10	-2	0.59	2.09	10.51	-1.49	0.49	1.56	10.82	-1.18	0.19	1.2
13	10.27	-2.73	0.41	2.76	11.61	-1.39	0.43	1.45	11.98	-1.02	0.27	1.06
14	11.3	-2.7	0.4	2.73	12.4	-1.6	0.29	1.62	12.72	-1.28	0.37	1.33
15	12.15	-2.85	0.84	2.97	13.6	-1.4	0.39	1.46	13.68	-1.32	0.45	1.4
16	12.96	-3.04	0.57	3.09	14.16	-1.84	0.49	1.91	14.76	-1.24	0.35	1.28
17	13.86	-3.14	0.66	3.21	14.72	-2.28	0.6	2.36	15.59	-1.41	0.35	1.45
18	14.49	-3.51	0.47	3.54	15.56	-2.44	0.65	2.53	16.67	-1.33	0.53	1.43
19	15.77	-3.23	0.66	3.29	16.91	-2.09	0.78	2.23	17.16	-1.84	0.79	2.01
20	16.07	-3.93	0.97	4.04	17.53	-2.47	0.43	2.51	18.18	-1.82	0.85	2

Only in the range of σ_n=*0-10* results were satisfactory, particularly for N = 7 The graph of Fig. 9.4 shows the MSE for the proposed method (9.7) and DT (9.12) based on the distortion value.

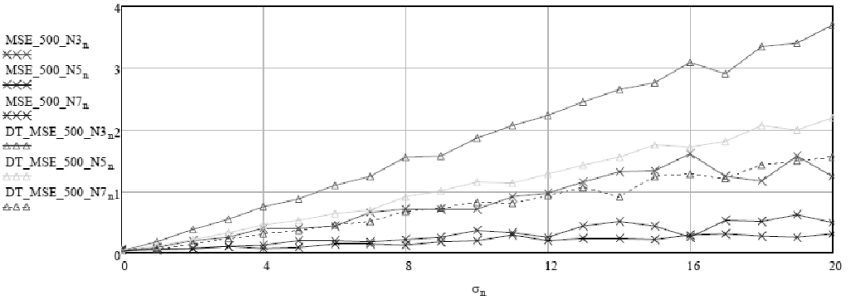


Fig. 9.4 MSE error values as a function of interference to the K=500 samples

9.4 Conclusions

The more a disturbance increased, the more accurate was estimation of the level of a random component. It can be noticed that nearly all of the obtained values of the load estimation μ_n ratio have exceeded zero, which leads to the suggestion that the method in question has a tendency to inflate values of disturbance variances.

Summing up, it can be assumed that the assumptions made were correct, as to the fact that median filtration has no effect on a usable signal, and in relation to the knowledge of the filter's suppression coefficient and its impact on a disturbance with normal distribution.

References

1. Ljung, L.: System Identification. Theory for User. Prentice-Hall, Englewood Cliffs (1987)
2. Poor, H.: An Introduction to Signal Detection and Estimation. Springer, New York (1985)
3. Ferrario, P.G.: Local Variance Estimation for Uncensored and Censored Observations. Springer (2013)
4. Mitra, S.K., Kaiser, J.F.: Handbook for Digital Signal Processing. John Willey & Sons (1993)
5. Kowalski, J.P., Peksinski, J., Mikolajczak, G.: Detection of noise in digital images by using the averaging filter name COV. In: Selamat, A., Nguyen, N.T., Haron, H. (eds.) ACIIDS 2013, Part II. LNCS (LNAI), vol. 7803, pp. 1–8. Springer, Heidelberg (2013)
6. Peksinski, J., Stefanowski, S., Mikolajczak, G.: Estimating the level of noise in digital images. In: Kanellopoulos, D.N. (ed.) Intelligent Multimedia Technologies for Networking Applications: Techniques and Tools. Information Science Reference, pp. 409–433 (2013)
7. Pi, H., Peterson, C.: Finding the embedding dimension and variable dependencies in time series. Neural Computation 6(3), 509–520 (1994)
8. Eirola, E., Liitiainen, E., Lendasse, A., et al.: Using the delta test for variable selection. In: Proceedings of the European Symposium on Artificial Neural Networks, ESANN 2008, Bruges, Belgium, pp. 25–30 (2008)
9. Jones, A.: New tools in non-linear modelling and prediction. Computational Management Science 1(2), 109–149 (2004)

Chapter 10
Reduction of Concepts from Generalized One-Sided Concept Lattice Based on Subsets Quality Measure

Peter Butka, Jozef Pócs, and Jana Pócsová

Abstract. One of the conceptual methods in data mining area is based on the one-sided concept lattices, which belongs to approaches known as Formal Concept Analysis (FCA). It provides an analysis of objects clusters according to the set of fuzzy attributes. The specific problem of such approaches is sometimes large number of concepts created by the method, which can be crucial for the interpretation of the results and their usage in practice. In this chapter we describe the method for evaluation of concepts from generalized one-sided concept lattice based on the quality measure of objects subsets. Consequently, this method is able to select most relevant concepts according to their quality, which can lead to useful reduction of information from concept lattice. The usage of this approach is described by the illustrative example.

10.1 Introduction

There are several approaches for identification of conceptual models from the datasets. One of them is called Formal Concept Analysis (FCA, [7]) and has

Peter Butka
Technical University of Košice, Faculty of Electrical Engineering and Informatics, Department of Cybernetics and Artificial Intelligence, Letná 9, 04200 Košice, Slovakia
e-mail: peter.butka@tuke.sk

Jozef Pócs
Palacký University Olomouc, Faculty of Science, Department of Algebra and Geometry, 17. listopadu 12, 771 46 Olomouc, Czech Republic
and
Mathematical Institute, Slovak Academy of Sciences, Grešákova 6, 040 01 Košice, Slovakia
e-mail: pocs@saske.sk

Jana Pócsová
Technical University of Košice, BERG Faculty, Institute of Control and Informatization of Production Processes, Boženy Němcovej 3, 043 84 Košice, Slovakia
e-mail: jana.pocsova@tuke.sk

A. Zgrzywa et al. (eds.), *New Research in Multimedia and Internet Systems*,
Advances in Intelligent Systems and Computing 314, DOI: 10.1007/978-3-319-10383-9_10

been applied in different domains like conceptual data analysis, data/text mining, information retrieval, or in other areas of artificial intelligence and machine learning. Standard (or classic) approach to FCA is based on 'crisp' case, where object-attribute model is represented by the binary relation (object has/has-not the attribute). In practice, there are many examples of object-attribute models where relationship between objects and their attributes is based on the fuzzy relations. In order to process such type of data using FCA-based methods, several approaches to fuzzy FCA have been designed. Here we can mention an approach of Bělohlávek [4, 5], an approach of Krajči [11], Popescu [15], generalized approach based on Galois connections [14], or other approaches [1, 12].

Important role for practical data mining tasks in fuzzy FCA has models known as one-sided concept lattices, where objects are considered as it is usual for crisp case (classic subsets) and their attributes obtain fuzzy values, i.e., on the side of attributes we have fuzzy sets. From existing one-sided approaches we could mention work of Krajči [10], Yahia and Jaoua [3], or paper of Jaoua and Elloumi on Galois lattices of real relations [9]. All these approaches are not able to process data table with different types of attributes, i.e., entries of each data tables are values from different truth value structure corresponding to given set of attributes. It is natural for practical data mining tools that data tables contain different types of attributes, e.g., binary (with possible values $0, 1$), quantitative (with values from interval of reals), ordinal (scale-based), or nominal (particular values from some domain). Therefore, in [6] we have introduced model called Generalized One-Sided Concept Lattice (GOSCL), which is able to work with data tables containing different types of attributes, i.e., different truth value structures for every attribute.

In case of one-sided concept lattices there is strong connection with clustering (cf. [8]). As it is known, clustering methods produce subsets of a given set of objects, which are closed under intersection, i.e., closure system on the set of objects. Since one-sided concept lattice approach produces also closure system on the set of objects, one-sided concept lattice approaches can be seen as a special case of hierarchical clustering. One of the problems in hierarchical clustering, and also in FCA, is how to select suitable and useful clusters of objects, i.e., relevant clusters. It is specially important issue in FCA domain, where there is relatively high number of concepts produced for input data tables.

The aim of this chapter is to introduce the method for reduction of number of concepts from FCA-based analysis by the selection of most relevant concepts from the lattice, where relevance is based on the quality measure on subsets of objects and it is computed using frequency of subsets presence in concept lattices for binary contexts created by the system of threshold-based transformations of original generalized one-sided concept context (input data table). This transformations are based on so-called h-cuts, which data analyst can define according to the combination of expected threshold values from the attributes truth value structures. We introduce the approach together with equation for the computation of quality for subsets of objects. The approach is then described by the illustrative example.

In the following section we provide the information on generalized one-sided concept lattices. Section 10.3 is devoted to defined quality measure on subsets of

objects, which can be used for the selection of most relevant concepts from the created concept lattice. The last section is related to the illustrative example which shows the applicability of our approach on the selected data table.

10.2 Generalized One-Sided Concept Lattices

In this section we shortly describe some necessary details regarding the generalized one-sided concept lattices and provide the example of such model, which will be then used also in another section. For more details on GOSCL model, its mathematical preliminaries (like Galois connections, closure operators, closure systems, etc.), related theorems and proofs, detailed algorithm for the creation of lattice and more examples, you are able to read our paper on this topic [6], or any other of our papers which used this model in our research.

A 4-tuple $\big(B,A,\mathsf{L},R\big)$ is said to be a *generalized one-sided formal context* if the following conditions are fulfilled:

(1) B is a non-empty set of objects and A is a non-empty set of attributes.
(2) $\mathsf{L} : A \to \mathsf{CL}$ is a mapping from the set of attributes to the class of all complete lattices. Hence, for any attribute a, $\mathsf{L}(a)$ denotes the complete lattice, which represents structure of truth values for attribute a.
(3) R is generalized incidence relation (input data table), i.e., $R(b,a) \in \mathsf{L}(a)$ for all $b \in B$ and $a \in A$. Thus, $R(b,a)$ represents a degree from the structure $\mathsf{L}(a)$ in which the element $b \in B$ has the attribute a.

Then the main aim is to introduce a Galois connection between classical subsets of the set of all objects $\mathbf{P}(B)$ and the direct products of complete lattices $\prod_{a\in A}\mathsf{L}(a)$ which represents a generalization of fuzzy subsets of the attribute set A, i.e., to the each element of the attribute set A there is assigned the different structure of truth values represented by complete lattice $\mathsf{L}(a)$.

Now, for any generalized one-sided formal context $\big(B,A,\mathsf{L},R\big)$ we are able to define a pair of mapping $^{\perp} : \mathbf{P}(B) \to \prod_{a\in A}\mathsf{L}(a)$ and $^{\top} : \prod_{a\in A}\mathsf{L}(a) \to \mathbf{P}(B)$ as follows:

$$X^{\perp}(a) = \bigwedge_{b\in X} R(b,a), \tag{10.1}$$

$$g^{\top} = \{b \in B : \forall a \in A,\ g(a) \le R(b,a)\}. \tag{10.2}$$

Such pair $(^{\perp},^{\top})$ forms a Galois connection between $\mathbf{P}(B)$ and $\prod_{a\in A}\mathsf{L}(a)$. Then $\mathscr{C}\big(B,A,\mathsf{L},R\big)$ defines the set of all pairs (X,g), where $X \subseteq B$, $g \in \prod_{a\in A}\mathsf{L}(a)$, satisfying

$$X^{\perp} = g \quad \text{and} \quad g^{\top} = X.$$

Set X is usually referred as *extent* and g as *intent* of the concept (X,g). Moreover, we can define the partial order on $\mathscr{C}\big(B,A,\mathsf{L},R\big)$ as:

$$(X_1,g_1) \le (X_2,g_2) \ \text{ iff } \ X_1 \subseteq X_2 \text{ iff } g_1 \ge g_2.$$

Finally, if (B,A,L,R) is generalized one-sided formal context, then $\mathscr{C}(B,A,\mathsf{L},R)$ with the partial order defined above forms a complete lattice, which is also called generalized one-sided concept lattice. This approach generalizes one-sided approaches and crisp (binary) approach to FCA mentioned in introduction section. Also, we are able to use GOSCL directly for computation of concept lattices from binary contexts (with the same result), simply by the usage of binary attributes in the input data table. We have used it for our computation of lattices from binary contexts in section with example of quality measure computation.

At the end of this section we provide the example of generalized one-sided context and corresponding concept lattice. Let $B = \{b_1,b_2,b_3,b_4,b_5\}$ is the set of objects and $A = \{a_1,a_2,a_3,a_4,a_5,a_6\}$ is the set of attributes. Now, attributes a_1,a_4,a_6 are binary ($\mathsf{L}(a_1) = \mathsf{L}(a_4) = \mathsf{L}(a_6) = \mathbf{2}$), attribute a_2 is ordinal attribute (or scale-based) with four possible values $0,1,2,3$ ($\mathsf{L}(a_2) = \mathbf{4}$). The last two attributes a_3,a_5 are real-valued attributes ($\mathsf{L}(a_3) = \mathsf{L}(a_5) = [0,1]$). The values of attributes for particular objects are provided by the relation R, which represents the input data table and is presented in Table 10.2. The corresponding one-sided concept lattice is depicted in Figure 10.1.

Table 10.1 Input data table – relation R for our example of generalized one-sided context

R	a_1	a_2	a_3	a_4	a_5	a_6
b_1	1	2	0.20	0	0.43	1
b_2	0	3	0.50	1	0.20	0
b_3	0	1	0.35	0	0.85	0
b_4	1	1	0.20	0	0.55	1
b_5	1	2	0.75	1	0.45	0

10.3 Relevance of Concepts – Quality Measure on Subsets of Objects

In this section we provide the approach for measuring the quality of objects subsets based on the calculation of their presence in concept lattices specially created from generalized one-sided context. It means that we setup some threshold-based cuts in generalized one-sided contexts in order to define quality measure for particular attributes, then these cuts are used to transform generalized context into several binary contexts, which are used for calculation of frequency of measured subset of objects. Such calculation is then used as quality of subset of objects.

Now we present our approach more formally. In next section we will describe the example, which illustrates the approach in more practical way.

Let P and Q be partially ordered sets. A mapping $f\colon P \to Q$ is said to be order-preserving if $x \leq y$ implies $f(x) \leq f(y)$. Recall, that a subset $U \subseteq P$ is upward closed

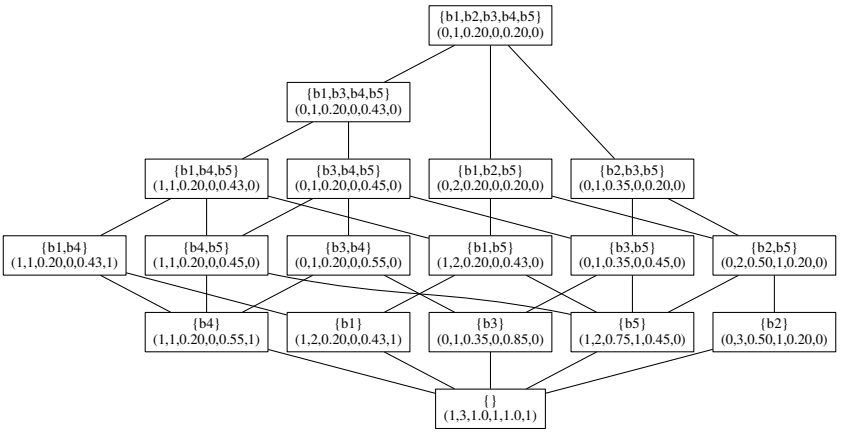


Fig. 10.1 One-sided concept lattice corresponding to R

if $x \leq y$, $x \in U$, $y \in P$ implies $y \in U$. Similarly, a subset $D \subseteq P$ is downward closed if $y \leq x$, $x \in D$, $y \in P$ yields $y \in D$. Upward and downward closed subsets are in one-to-one connection with order preserving mappings to the two element chain **2**.

Let $f\colon P \to \mathbf{2}$ be an order-preserving mapping. In this case $f^{-1}(\{1\})$ is an upward closed subset of P and $f^{-1}(\{0\})$ is a downward closed subset of P. This follows from the fact that if $x \in f^{-1}(\{1\})$ and $x \leq y$, then $1 = f(x) \leq f(y) = 1$, which yields that $y \in f^{-1}(\{1\})$. Fact that $f^{-1}(\{0\})$ forms a downward closed subset of P can be shown in similar way. Conversely, let U be an upward closed subset of P. Then $D = P \setminus U$ is downward closed and we can define $f\colon P \to \mathbf{2}$ as $f(x) = 1$ if $x \in U$ and $f(x) = 0$ if $x \in D$. As one can easily verify, such defined mapping f is order preserving.

Then, let $p \in P$ is an arbitrary element of partially ordered set P. According to the previous facts the principal filter $\uparrow(p) = \{x \in P : p \leq x\}$ determines an order-preserving mapping. Hence the system of all order-preserving mappings form P to **2** is relatively rich, it contains at least as many elements as the set P.

Now we are able to move into area of FCA, i.e., we have generalized one-sided context $\big(B, A, \mathsf{L}, R\big)$. It is possible to define for every attribute $a \in A$ an h-cut as an order-preserving mapping $h\colon \mathsf{L}(a) \to \mathbf{2}$. When it is useful for particular case, we are able to define several h-cuts for every attribute. For example, if we have scale-based attribute a^* defined by $\mathsf{L}(a^*) = \mathbf{4}$ (i.e., attribute values are 0,1,2 and 3), we can define some h-cut with the threshold on value 2, i.e., where $h_{a^*} = 1$ only if value for this attribute is at least 2, otherwise $h_{a^*} = 0$. Similarly, we can have another h-cut with threshold on other value. Another example can be real-valued attribute, where we are able to define h-cuts using some threshold represented by value from the interval of reals used for the attribute (e.g., 0.75 can be used as a threshold for some attribute with values from $[0,1]$).

If we know h-cuts for our attribute, we are able to combine them for calculation of frequency of some subset of objects in order to find most relevant subsets of objects

according to their presence in concept lattices created from binary contexts defined by particular h-cuts. In more details, let m be a positive integer and $\left(B,A,\mathsf{L},R,\mathscr{H}_m\right)$ be a generalized one-sided context with the system of h-cuts $\mathscr{H}_m = \left((h_a^n)_{a\in A}\right)_{n=1}^{m}$, where $h_a^n\colon \mathsf{L}(a) \to \mathbf{2}$ for all $a \in A$ and for all $1 \leq n \leq m$.

Now we are able to analyze every binary context created using particular h-cut, i.e., let $n \in \{1,\ldots,m\}$ be a positive integer (indexing n-th h-cut). From the system of h-cuts $(h_a^n)_{a\in A}$ we obtain a binary context $\left(B,A,R^n\right)$ with

$$(b,a) \in R^n \quad \text{iff} \quad h_a^n(R(b,a)) = 1 \tag{10.3}$$

for all objects $b \in B$ and all attributes $a \in A$.

After this step we have m binary contexts and we are able to create their corresponding crisp concept lattices using standard FCA framework. Then it is only needed to calculate frequency of any subset of objects in those concept lattices and we get the quality of subset with the value from $[0,1]$ interval. The formula for this quality measure for any subset X from set of objects B based on defined m h-cuts is ($\underline{\mathfrak{B}}(B,A,R^n)$ is concept lattice created from binary relation R^n):

$$Q_m(X) = \frac{\left|\left\{n \in \{1,\ldots,m\} : \exists Y \subseteq A, (X,Y) \in \underline{\mathfrak{B}}(B,A,R^n)\right\}\right|}{m} \tag{10.4}$$

In practice, we have some data table containing more attributes and we should select h-cuts for them particularly. Then all the combinations of different local h-cuts through the attributes create our system of h-cuts for whole data table (generalized one-sided context). Number of all these combinations is then equal to m, which is used in equation (10.4). We will show the details in more practical way in next section.

Then if we have all necessary h-cuts for the whole data table, it is possible to compute quality measure for every subset of objects in concepts presented in original generalized one-sided concept lattice (GOSCL). This allows us to find concepts from GOSCL containing subsets of objects with higher quality (we can see it as some ranking of concepts), which are then used as most relevant concepts for reduction of information from FCA-based analysis. The selection of h-cuts can be systematic (like selection of all possible thresholds for discrete attributes, or some proportional selection of thresholds for real-valued attributes) or it can be up to data analyst and his/her knowledge of domain (e.g., he/she knows which thresholds are really important for the domain and particular analysis).

10.4 Illustrative Example with Selected Data Table

Now we provide an example of the presented approach for subset quality measure computation. As generalized one-sided context we use the example from the end of section 10.2. For summary, we have set of objects $B = \{b_1,b_2,b_3,b_4,b_5\}$ and set of attributes $A = \{a_1,a_2,a_3,a_4,a_5,a_6\}$. Three attributes are binary attributes (a_1,a_4,a_6) with $\mathsf{L}(a_1) = \mathsf{L}(a_4) = \mathsf{L}(a_6) = \mathbf{2}$. One attribute is scale-based (ordinal) attribute

(a_2) with $\mathsf{L}(a_2) = \mathbf{4}$. The other two attributes are real-valued attributes (a_3, a_5) with $\mathsf{L}(a_3) = \mathsf{L}(a_5) = [0,1]$.

Now we want to measure the quality of concepts of generalized one-sided concept lattice depicted in Figure 10.1. It means that we will compute the quality (or relevance) of subsets representing the objects in particular concepts using equation (10.4). Of course, it is possible to select different h-cuts for every attribute, but for the simplicity of our example, we will use same h-cuts for every type of attribute. Of course, binary attributes will not be transformed to some other values, we use them as they are (h-cut for these attributes is identity function).

Then, for ordinal attribute a_2 we use two different h-cuts definitions. In first setting h-cut threshold is 2, i.e., the original values of attributes are transformed into 1 only for original value higher or equal to 2. Similarly, second h-cut setting has threshold value 3. Next, for real-valued attributes a_3, a_5 we use three thresholds 0.25, 0.5 and 0.75. Therefore, for our whole input data table we have 6 different combinations of h-cuts settings through all attributes (for every of three h-cuts used with real-valued attribute we can use two h-cuts for scale-based attribute, i.e., we have 2×3 combinations of h-cuts settings).

Table 10.4 provides relations for three binary contexts created using h-cut settings related to combination of threshold value 2 for scale-based attribute a_2 and three different threshold values for real-valued attributes a_3, a_5, where R^1, R^2, R^3 belongs to usage of thresholds $0.25, 0.5, 0.75$. Similarly, Table 10.4 provides relations for three binary contexts created using h-cut settings related to combination of threshold value 3 for scale-based attribute a_2 and three different threshold values for real-valued attributes, where R^4, R^5, R^6 belongs to usage of thresholds $0.25, 0.5, 0.75$.

Table 10.2 Relations of binary contexts transformed from original relation for threshold value 2 for a_2 combined with threshold values $0.25, 0.5, 0.75$ for a_3, a_5 (corresponding relations are in same order as threshold values for real-valued attributes – R^1, R^2, R^3)

R^1	a_1	a_2	a_3	a_4	a_5	a_6
b_1	1	1	0	0	1	1
b_2	0	1	1	1	0	0
b_3	0	0	1	0	1	0
b_4	1	0	0	0	1	1
b_5	1	1	1	1	1	0

R^2	a_1	a_2	a_3	a_4	a_5	a_6
b_1	1	1	0	0	0	1
b_2	0	1	1	1	0	0
b_3	0	0	0	0	1	0
b_4	1	0	0	0	1	1
b_5	1	1	1	1	0	0

R^3	a_1	a_2	a_3	a_4	a_5	a_6
b_1	1	1	0	0	0	1
b_2	0	1	0	1	0	0
b_3	0	0	0	0	1	0
b_4	1	0	0	0	0	1
b_5	1	1	1	1	0	0

After the creation of particular binary contexts, we are able to create concept lattices for every of them. In our example resulted concept lattices are provided in Figure 10.2 (for binary relations R^1, R^2 and R^3) and in Figure 10.3 (R^4, R^5 and R^6). Now we only need to get all subsets of objects (extents) of concepts from original generalized one-sided concept lattice (see Figure 10.1) and count the quality of them using equation (10.4). The result of the computation can be found in Table 10.4, where are all subsets ordered by their quality.

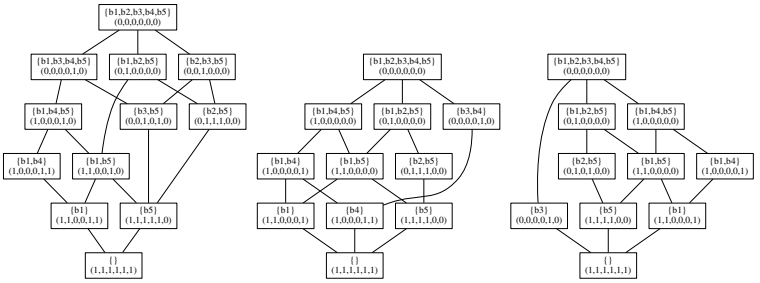


Fig. 10.2 Concept lattices corresponding to binary relations R^1 (left), R^2 (middle), R^3 (right)

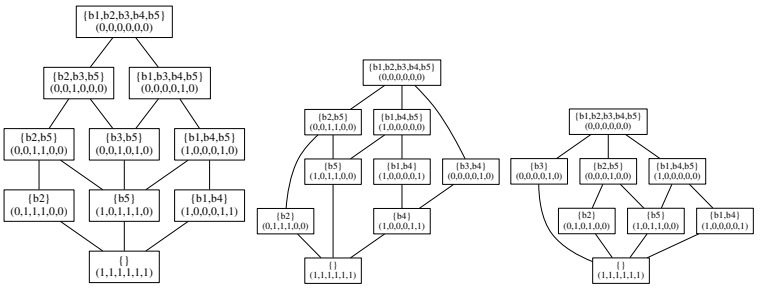


Fig. 10.3 Concept lattices corresponding to binary relations R^4 (left), R^5 (middle), R^6 (rights)

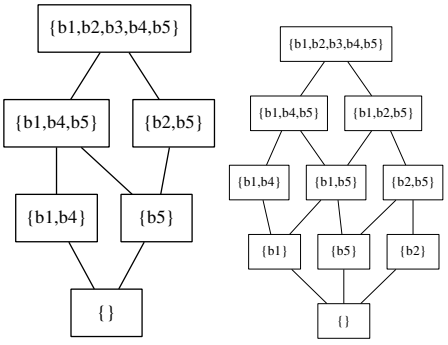


Fig. 10.4 Hierarchical structures of clusters for some quality threshold – left for 1.0, right for 0.5

Table 10.3 Relations of binary contexts transformed from original relation for threshold value 3 for a_2 combined with threshold values $0.25, 0.5, 0.75$ for a_3, a_5 (corresponding relations are in same order as threshold values for real-valued attributes – R^4, R^5, R^6)

R^4	a_1	a_2	a_3	a_4	a_5	a_6
b_1	1	0	0	0	1	1
b_2	0	1	1	1	0	0
b_3	0	0	1	0	1	0
b_4	1	0	0	0	1	1
b_5	1	0	1	1	1	0

R^5	a_1	a_2	a_3	a_4	a_5	a_6
b_1	1	0	0	0	0	1
b_2	0	1	1	1	0	0
b_3	0	0	0	0	1	0
b_4	1	0	0	0	1	1
b_5	1	0	1	1	0	0

R^6	a_1	a_2	a_3	a_4	a_5	a_6
b_1	1	0	0	0	0	1
b_2	0	1	0	1	0	0
b_3	0	0	0	0	1	0
b_4	1	0	0	0	0	1
b_5	1	0	1	1	0	0

Table 10.4 Qualities of subsets of objects (extents) for every concept from generalized one-sided concept lattice depicted in Figure 10.1 – ordered by the calculated quality

Extent	Q_m	Extent	Q_m	Extent	Q_m
$\{b_1, b_2, b_3, b_4, b_5\}$	1.0	$\{b_1, b_2, b_5\}$	0.5	$\{b_3, b_4\}$	0.333
$\{b_1, b_4, b_5\}$	1.0	$\{b_1, b_5\}$	0.5	$\{b_3, b_5\}$	0.333
$\{b_1, b_4\}$	1.0	$\{b_1\}$	0.5	$\{b_3\}$	0.333
$\{b_2, b_5\}$	1.0	$\{b_2\}$	0.5	$\{b_4\}$	0.333
$\{b_5\}$	1.0	$\{b_1, b_3, b_4, b_5\}$	0.333	$\{b_3, b_4, b_5\}$	0.0
$\{\}$	1.0	$\{b_2, b_3, b_5\}$	0.333	$\{b_4, b_5\}$	0.0

The presented results show the possibility to select more relevant concepts from the original model, i.e., it is possible to use it as method for reduction of information from generalized one-sided concept lattice. For example, if we choose quality threshold 1.0, we have 6 concepts (from original 18). Also we are able to select 11 best concepts for threshold 0.5. Of course, two concepts (one with all objects and second with empty set of objects) are somehow trivial and can be removed from the result in some analytical cases. Interesting result of the analysis is also the possibility to setup threshold and creates the reduced hierarchical structure of clusters on defined level of quality. Example is shown in Figure 10.4, where one can see such structures for thresholds 1.0 and 0.5 for quality. The important is the role of data analyst who is able to identify useful setup of h-cuts for particular problem and domain of the analysis. Then this quality measure become more successful tool for meaningful reduction of information from FCA-based analysis.

In the future we would like to test our approach as a selection method for relevant concepts on real datasets. We can imagine the information retrieval system based on the selection of relevant clusters (and their objects like documents, news, etc.) using quality measure, some data/text-mining tasks in different domains (e.g., classification and clustering of documents [16, 17], or data analysis of e-learning users groups [13] or meteorological data [2].

10.5 Conclusions

In this chapter we have presented the method for reduction of concepts from FCA-based model called generalized one-sided concept lattices by the calculation of their quality. This measure is based on the frequency of the presence of specified subset of objects in binary concept lattices, which are created by the transformation of original context to binary contexts using defined threshold-based h-cuts. The approach can be useful in selection of relevant clusters from the analysis of input data table. It can be used as a final tool, or as an input for another analysis in some hybrid analytical approach. Interesting fact is that skilled data analyst can setup the h-cuts specifically to problem and domain of the analysis and therefore achieve more precise selection and reduction of the information.

Acknowledgements. This work was supported by the Slovak Research and Development Agency under contracts APVV-0208-10, APVV-0035-10, and APVV-0482-11, by the Slovak VEGA Grants 1/1147/12, 1/0729/12, and 1/0497/11, by the Slovak KEGA grant No.040 TUKE-4/2014, and by the ESF Fund CZ.1.07/2.3.00/30.0041.

References

1. Antoni, L., Krajci, S., Kridlo, O., Macek, B., Piskova, L.: On heterogeneous formal contexts. Fuzzy Set. Syst. 234, 22–33 (2014)
2. Bartok, J., Babič, F., Bednár, P., Paralič, J., Kováč, J., Bartoková, I., Hluchý, L., Gera, M.: Data Mining for Fog Prediction and Low Clouds Detection. Comput. Informat. 31(6+), 1441–1464 (2012)
3. Ben Yahia, S., Jaoua, A.: Discovering knowledge from fuzzy concept lattice. In: Kandel, A., Last, M., Bunke, H. (eds.) Data Mining and Computational Intelligence. STUDFUZZ, vol. 68, pp. 167–190. Springer, Heidelberg (2001)
4. Bělohlávek, R.: Lattices generated by binary fuzzy relations. Tatra Mt. Math. Publ. 16, 11–19 (1999)
5. Bělohlávek, R.: Lattices of Fixed Points of Fuzzy Galois Connections. Math. Log. Quart. 47(1), 111–116 (2001)
6. Butka, P., Pócs, J.: Generalization of one-sided concept lattices. Comput. Informat. 32(2), 355–370 (2013)
7. Ganter, B., Wille, R.: Formal concept analysis. Mathematical foundations. Springer, Berlin (1999)
8. Janowitz, M.F.: Ordinal and relational clustering. World Scientific Publishing Company, Hackensack (2010)
9. Jaoua, A., Elloumi, S.: Galois connection, formal concepts and Galois lattice in real relations: application in a real classifier. J. Syst. Software 60, 149–163 (2002)
10. Krajči, S.: Cluster based efficient generation of fuzzy concepts. Neural Netw. World 13(5), 521–530 (2003)
11. Krajči, S.: A generalized concept lattice. Logic Journal of IGPL 13(5), 543–550 (2005)
12. Medina, J., Ojeda-Aciego, M., Ruiz-Calviño, J.: Formal concept analysis via multi-adjoint concept lattices. Fuzzy Set. Syst. 160, 130–144 (2009)
13. Paralič, J., Richter, C., Babič, F., Wagner, J., Raček, M.: Mirroring of Knowledge Practices based on User-defined Patterns. J. Univers. Comput. Sci. 17(10), 1474–1491 (2011)

14. Pócs, J.: Note on generating fuzzy concept lattices via Galois connections. Inform. Sci. 185(1), 128–136 (2012)
15. Popescu, A.: A general approach to fuzzy concepts. Math. Log. Quart. 50(3), 265–280 (2004)
16. Sarnovský, M., Kačur, T.: Cloud-based classification of text documents using the Gridgain platform. In: 7th IEEE International Symposium on Applied Computational Intelligence and Informatics (SACI 2012), pp. 241–245 (2012)
17. Sarnovský, M., Ulbrik, Z.: Cloud-based clustering of text documents using the GHSOM algorithm on the GridGain platform. In: 8th IEEE International Symposium on Applied Computational Intelligence and Informatics (SACI 2013), pp. 309–313 (2013)

Chapter 11
Does Topic Modelling Reflect Semantic Prototypes?

Michał Korzycki and Wojciech Korczyński

Abstract. The chapter introduces a representation of a textual event as a mixture of semantic stereotypes and factual information. We also present a method to distinguish semantic prototypes that are specific for a given event from generic elements that might provide cause and result information. Moreover, this chapter discusses the results of experiments of unsupervised topic extraction performed on documents from a large-scale corpus with an additional temporal structure. These experiments were realized as a comparison of the nature of information provided by Latent Dirichlet Allocation based on Log-Entropy weights and Vector Space modelling. The impact of different corpus time windows on this information is discussed. Finally, we try to answer if the unsupervised topic modelling may reflect deeper semantic information, such as elements describing given event or its causes and results, and discern it from factual data.

11.1 Introduction

Unsupervised probabilistic topic modelling is widely used in the information retrieval techniques and is a widely applied tool for analysis of large-scale corpora. It is based on the assumption that text documents are mixtures of topics. This mixture may be treated as a multinomial probability distribution over the words. A couple of methods have been developed that allow to create such distributions with various properties [15, 6, 2].

Although useful for information retrieval purposes, those topic extraction methods do not deal well with many linguistic issues such as polysemy or synonymy, since recent researches tend to show that deeper semantical relations found in texts cannot be directly retrieved using those unsupervised methods [5, 17].

Michał Korzycki · Wojciech Korczyński
AGH University of Science and Technology
Al. Mickiewicza 30, 30-962, Krakow, Poland
e-mail: {korzycki,wojciech.korczynski}@agh.edu.pl

A. Zgrzywa et al. (eds.), *New Research in Multimedia and Internet Systems*,
Advances in Intelligent Systems and Computing 314, DOI: 10.1007/978-3-319-10383-9_11

In this chapter we present the results of a preliminary work that allowed us to extract deeper semantic information from the text through postprocessing the results of unsupervised methods. We took advantage of the fact that we operated on a large-scale corpus (of the order of millions of documents) that possesses an additional temporal structure. The suggested approach permits to find in an analyzed text elements related to the specific semantic prototypes (e.g. mental models) of the events described within and to discern them from pure factual information, given that a sufficiently large corpus is available.

The framework of the experiments presented in this chapter is an integral part of a large scale project related to security and intelligence analysis.

11.2 Related Work

Latent Semantic Analysis (LSA) is an original word/document matrix rank reduction algorithm, which extracts word co-occurrences in the frame of a text. As a result, each word in the corpus is related to all co-occurring words and to all texts in which it occurs. This makes a base for an associative text comparison. The applicability of the LSA algorithm is a subject of various types of research – from a text content comparison [3] to an analysis of human association norm [12]. But there is still little interest in studying the linguistic significance of LSA-made associations.

Latent Dirichlet Allocation (LDA) is one of the most known generative models used for topic extraction. It was first presented by David Blei, Andrew Ng, and Michael Jordan in 2003. As the other generative models, it assumes that a collection of documents may be represented by a mixture of latent topics, however words creating each topic are chosen according to the per-document multinomial Dirichlet distribution of fixed dimensionality. LDA is a technique based on the „bag of words" paradigm and it can infer distributions of topics e.g. with the use of variational Bayes approximation [1, 16], Gibbs sampling [6] or expectation propagation [11].

Some recent research was focusing on finding if the relationships coming from the unsupervised topic extraction methods reflect semantic relationship reflected in human association norms. A comparison of human association norm and LSA-made association lists can be found in [5] and it should be the base of the study. Results of the other preliminary studies based on such a comparison: [17, 19, 18], show that the problem needs further investigation. It is worth noticing that all the types of research referred to, used a stimulus-response association strength to make a comparison. The results of the afformentioned research have shown that using unsupervised topic extraction methods one is able to create associations between words that are strongly divergent from the ones obtained by analysing the human generated associations.

Fully aware of the issues related to the automatic text classification mentioned above, by using different approaches and postprocessing methods, we show that it is possible to obtain results more in line with what could be called a semantic text classification.

11.3 Semantic Prototypes

The notion of a semantic prototype comes from cognitive theory [13] where a notion is represented by its elements with their features. So, according to this model, a notion of a "bird" would be "composed" of such elements and features as "feathers", "beak" and "ability to fly". Semantic prototypes can also be discussed in the context of event descriptions that occur in texts. Prototype theory has also been applied in linguistics for mapping from phonological structure to semantics.

In a domain of natural language processing, this approach is reflected in so-called content theories. A content theory is used to determine a meaning of phrases and information they carry. One of the classic and most known elements of content theory is the Conceptual Dependency theory that was invented and developed by Robert Schank [14]. His main goal was to create a conceptual theory consistent in every natural language. The theory's main assumptions were: the generalization of representation language and inference about implicitly stated information. The first assumption means that two synonymous sentences should have identical conceptual representations. The second one states that all the implicit information has to be stated explicitly by inferring it from explicit information. Each sentence can be then represented in a form of conceptual dependency graph built of three types of elements: primitives, states and dependencies. Primitives are predicates that represent a type of an action, states specify the preconditions and results of actions and dependencies define conceptual relations between primitives, states and other objects [10]. Accordingly to the Conceptual Dependency theory we may represent the event of "tea drinking" by a sequence of events: "tea making", "cup operating", "tea sipping" and so on, that are composed of action, object etc.

The described event model proved itself to be very useful in many applications [9] and we found it very suitable to quantifiable comparisons to unsupervised topic modelling methods [4].

From that theory we deduce our model of an event – its prototype – a compound structure of actions, actors, states, and dependencies but also composed of preconditions and results, being events themselves. This event model reflects also very well the semantic structure of a text. If a document describes an event, it is almost always presented in the context of the causes of the event and the resulting consequences. This will be reflected in various topic models that will tend to reflect that most texts are represented as a linear combination of multiple topics. Determining from such combination which topic (event) can be classified as a cause and which is an effect would be very interesting, but that issue is beyond the scope of this chapter.

On the other hand, the modelled text is composed not only of events, but also features of those events specific only to that instance of the event. As such, a text can be seen as having two aspects – the main event of the text intermingled with the elements of cause and result events and factual features that are specific to that single event. The latter aspect would relate to places, actors and contextual information. The former aspect would relate to generic elements that are common to similar events that occur in the corpus.

This chapter focuses on unsupervised identification and retrieval of those two different event model aspects of texts. Further experimental research is the first attempt at generating semantic prototypes automatically.

11.4 Corpus

The experiment was conducted on a subset of a 30-year worldwide news archive coming from the New York Times. That corpus has been chosen as it is interesting for a number of reasons:

- it is freely available,
- some interesting research results have been obtained based on it [8],
- it is quite comprehensive in terms of vocabulary and range of described event types,
- its relatively large size (approximately 90.000 documents per year, for a total of 2.092.828 documents spanning the years 1981-2012) gives an ample opportunity to experiment with various document time spans without impacting noticeably its event scope representiveness and lexicon balance.

After a set of trials with various time spans ranging from months to 15 years, the most pronounced effects of the experiment described below could be obtained by comparing two time spans – one covering 6 months from July 2000 to December 2000 and the other covering 4 years from 1997 to 2000. Those sub-corpora contain 45.815 and 350.499 documents, respectively.

11.5 Method

We based the data of our experiment on the term/document matrix populated with Log-Entropy weights [7]. More precisely, the value a_{ij} in the matrix corresponding to the i-th word and j-th document can be expressed as the usual ratio of a local and a global weight:

$$a_{ij} = e_i \log(t_{ij} + 1)$$

where:

$$e_i = 1 - \sum_j \frac{p_{ij} \log p_{ij}}{\log n}$$

for n – the total number of documents, t_{ij} is the number of times the i-th word occurs in the j-th document and $p_{ij} = \frac{t_{ij}}{g_i}$, with g_i the total number of times the term i occurs in the whole corpus.

After building an LDA model (with d dimensions), we obtain a v_{jk} matrix of size $n \times d$ describing how much the topic k impacts the document j. The v_{jk} matrix contains per design only non-negative values.

Each topic k is in turn represented by a vector of non negative weights describing how much a single word i participates in this topic, in the form of a $N \times d$ matrix w_{ik} (for N – the size of the lexicon).

For each document j and word i the word model matrix m_{ji} of size $n \times N$, obtained by multiplying the document weights with the topic model $m = vw^{\top}$, gives a general representation of a document j within a word space.

For a given document j we will now analyze the rank of words in the sorted vector m_{ji}, for each word i.

11.6 Experiment

We based our experiment on a subset of the New York Times corpus described above. The two compared subsets were spanning 6 months and 4 years, respectively. We focused on the representation of the news item related to the accident of Kursk, the Russian submarine that sank on 12th August 2000. Below is a fragment of the input text used for the experiments:

> *A Russian submarine plunged to the seabed in the Barents Sea on Sunday during a naval exercise, possibly after an explosion on board, officials said today. They said the submarine was badly damaged, and was trapped at least 450 feet below the surface. They said they did not know how many of the more than 100 crew members on board were alive or how long they could survive. Tonight the navy began preparing a desperate attempt to rescue the crew. But navy officials said the odds of saving the men were slim. The submarine, called the Kursk, was not carrying nuclear weapons, the navy said, but was powered by two nuclear reactors, raising concerns about possible radioactive contamination.*

The experiments were performed using 2 methods of topic modelling: LDA based on a term/document matrix populated with Log-Entropy weights and the Vector Space Model based on the mentioned matrix. These models were computed basing on texts from a 4-year article span. Additionally, the Log-Entropy model was built on texts from a 6-month range in order to observe the changes resulting in considering a different time windows. Finally, a ratio of Log-Entropy results from the 6-month and 4-year ranges was calculated so that we could better analyze changes that took place in models built in different time windows.

In order to properly understand the results of LDA-based modelling, one has to look at the analyzed event – the sinking of the Kursk submarine – in a more general way as an accident of a naval vehicle that happened in Russia.

After analysing which words are the highest ranked in our model, it may be observed that LDA model distinguished words that may be somehow connected with:

- vehicles: ship (ranked 2nd with score 0.00417), vessel (7th, 0.00246), plane (8th, 0.00237), boat (11th, 0.00228) words associated with transport: port (1st, 0.00434), airline (3rd, 0.00304), flight (4th, 0.00275), airport (5th 0.00271), tunnel (15th, 0.00200), pilot (26th, 0.00169), passenger (27th, 0.00167)
- Russia: Russia (6th, 0.00251), Russian (9th, 0.00236), Moscow (30th, 0.00156)

- sea (except for the already mentioned port, ship, vessel, boat): navy (10th, 0.00231), sea (13th, 0.00208), harbor (14th, 0.00201), naval (28th, 0.00164), shipping (32nd, 0.00153), water (51st, 0.00121), sailor (56th, 0.00118)
- accidents: besides many of the words already mentioned, the word crash (35th, 0.00138) is significant.

These words are very general and are common terms used while describing some event. It has to be emphasized that there is no word specific for a given event. They were filtered out in accordance with the nature of LDA that rejects words characteristic for just a narrow set of documents and promotes words that are specific to extracted topics. Therefore, we cannot expect highly ranked terms that would be strictly connected with the accident of the Kursk submarine but rather words related generally to accidents or vehicles, transport, sea and Russia.

These words are very general and descriptive. Using them, it is not possible to state anything specific ("factual") about the nature of a given event, its causes or consequences.

Analysing the results of the Log-Entropy model calculations, we are able to see that the highest ranked words are more specific than in the case of the LDA model.

Among the highest ranked words are ones that are related to the causes of the main event:

- *reactor* (6th, 0.13189), *nuclear* (13th, 0.10489), *radioactive* (29th, 0.07876): despite the fact that in case of Kursk accident reactors shutting down is rather a consequence, many news described also some previous submarine accidents caused by malfunction of nuclear reactors
- *accident* (17th, 0.09974): some "accident" as a reason of submarine sinking
- *pressurized* (19th, 0.09732): media reported that the lack of pressurized escape chambers was the reason why the crew was not able to get out of a submarine
- *sank* (25th, 0.08470): "the submarine sank" as a the central event
- *stricken* (34th, 0.07292): "submarine was stricken" as a reason of the accident

Words can also be found related to the consequences of the discussed accident:

- *minisub* (ranked 3rd with score 0.20723): a minisub was sent with a rescue mission.
- *Thresher* (7th, 0.12576): USS Thresher was a submarine, which sinking was frequently compared to the accident of Kursk.
- *rescue* (10th, 0.11231) and crew (12th, 0.10625): rescue crew was sent in order to help sailors
- *Kuroyedov* (24th, 0.08620): Fleet Admiral Vladimir Kuroyedov was in charge of navy when Kursk sank and therefore after the accident spoke with the media very often
- *Lockhart* (28th, 0.08162): Joe Lockhart was the White House spokesman that talked to the media after the accident of Kursk and informed about the American president's offer of help

- *Nilsen* (31st, 0.07615): Thomas Nilsen is a Norwegian researcher that wrote a report on Russian fleet. He was also interviewed by media after the accident
- *Komsomolet* (33rd, 0.07436): K-278 Komsomolet was a Soviet nuclear-power submarine that was mentioned frequently in many reports on Soviet/Russian fleet after the accident of Kursk
- *stricken* (34th, 0.07292): "stricken submarine" as a consequence of the accident

These words are much more specific than in the case of those extracted by the LDA model. They strictly concern this event and describe its causes and consequences.

At first sight, the results of Log-Entropy model calculations in 6-month time window are very similar to the previous ones. We can see the same elements that we identified as the cause of the accident (*sank, stricken, reactor, nuclear*) and its consequences (eg.: *minisub, Thresher, rescue, crew, Kuroyedov*). However, the results yielded in these two time windows differ in scores. In order to analyze how the rank of particular words changed, we calculated a ratio of each word's score in two time span windows – spanning 4 years and 6 months. Having in mind that changing the time window practically does not change the local weight of a given term but changes its global weight, this ratio would emphasize these changes as a comparison of each word's global weights while similar local weights would become irrelevant.

Table 11.1 Top 20 words based on Log-Entropy models ratio

Word	Log-Entropy ratio
al	1.28062742713
Kursk	1.24871629725
Barents	1.19631168918
Kuroyedov	1.17117438796
minisub	1.16238976560
aug	1.15070033007
squalu	1.11170935342
noaa	1.10874462017
iii	1.09844052384
32f	1.08708935266
seabed	1.08585777559
Nilsen	1.07438727067
Vladimir	1.06498173260
torpedo	1.06238346439
flotilla	1.06017391338
Thresher	1.05610904592
photo	1.05231801755
certified	1.05071293879
outcome	1.04815882266
avalon	1.03809885010

As Table 11.1 presents, it turned out that the words that could be used in describing causes and consequences of the Kursk sinking are now much more emphasized. Moreover, the most specific for this particular event words are stressed, while terms that could be used in descriptions of other, similar accidents (e.g. reactor, nuclear, radioactive, rescue, crew) have lower rank. Besides, new interesting words appeared when considering a ratio-based ranked list of words:

- *Kuroyedov* (ranked 4th), *minisub* (ranked 5th), *Nilsen* (ranked 12th): they are still high ranked as the most specific words for this particular event
- *seabed* (ranked 11th): as a consequence of the accident, the submarine was plunged to the seabed
- *torpedo* (ranked 14th), *torpedoes* (ranked 29th): an explosion of one of torpedoes that the Kursk was carrying, has been recognized as the main reason of the accident
- *Ivanov* (ranked 31st): in time of the Kursk sinking Sergei Ivanov was the head of the Russian Security Council, therefore was highly involved in this case, so his name was often mentioned as a consequence of this accident
- *hull* (ranked 33rd): after the accident, the rescue crew tried to get into the submarine through its hull
- *slim* (ranked 43rd): day after day the chances of saving sailors were slimmer

As one can see, Table 11.1 also contains words like *al*, *aug*, *noaa*, *iii*, *32f*. They may be treated as a noise since they do not provide any significant information, even though the algorithm classified them as important data. In future work we intend to exclude this type of unnecessary terms.

It seems very interesting how calculating of the ratio helped with finding new words describing causes and consequences and how it distinguished terms that are specific for a given event. It also emphasized changes that occurred in different time windows.

11.7 Conclusion

In this chapter we introduced the concept of the text being a structure consisting of a mixture of event descriptions and factual information. Our main goal was to verify if unsupervised methods were able to model an event in a semantically meaningful way, reflecting its semantic structure, understood in the context of the event model described above. Relevant experiments on large-scale corpora have been performed. They included techniques of topic modelling with the use of a Latent Dirichlet Allocation (LDA) model and a Vector Space Model, both based on the Log-Entropy weighting scheme. The nature of information provided by these two models has been analysed. It turned out, that according to its nature, LDA rejected all the information specific for the given event and distinguished very descriptive and general terms. On the other hand, a Vector Space Model was able to emphasize words that were more specific for the given event, that describe its causes and consequences.

Additionally, we examined changes in the Vector Space Model depending on a different time window. Calculating a ratio of scores obtained in a 4-year and

6-month articles time span has allowed us to identify the most specific and meaningful words that provide the most information about the given event.

This preliminary research shows that there are possibilities of extracting information of real semantic value from the postprocessing of results of unsupervised topic modelling methods, given the availability of a sufficiently large corpus.

Presented research may be applied in a security and intelligence analysis. From point of view of such an analysis, full-text sources (testimonies, reports, notes, etc.) are essential and carry great importance, however an identification of objects and information in these contents is often difficult. Therefore, automatic mechanisms, such as the one discussed in this chapter, are highly necessary.

We suppose that comparing the results of topic modelling and vector modelling may enable to grade them by the level of generality or specificity. This research is a step towards our long term goal which is the creation of a method of automatic semantic prototype identification.

Acknowledgements. The research reported in the chapter was partially supported by grants "Advanced IT techniques supporting data processing in criminal analysis" (No. 0008/R/ID1/2011/01) and "Information management and decision support system for the Government Protection Bureau" (No. DOBR-BIO4/060/13423/2013) from the Polish National Centre for Research and Development.

References

1. Blei, D.M., Ng, A.Y., Jordan, M.I.: Latent dirichlet allocation. J. Mach. Learn. Res. 3, 993–1022 (2003), http://dl.acm.org/citation.cfm?id=944919.944937
2. Boyd-Graber, J., Chang, J., Gerrish, S., Wang, C., Blei, D.: Reading tea leaves: How humans interpret topic models. In: Neural Information Processing Systems (NIPS) (2009)
3. Deerwester, S., Dumais, S.T., Furnas, G.W., Landauer, T.K., Harshman, R.: Indexing by latent semantic analysis. Journal of the American Society for Information Science 41(6), 391–407 (1990)
4. Dorosz, K., Korzycki, M.: Latent semantic analysis evaluation of conceptual dependency driven focused crawling. In: Dziech, A., Czyżewski, A. (eds.) MCSS 2012. CCIS, vol. 287, pp. 77–84. Springer, Heidelberg (2012)
5. Gatkowska, I., Korzycki, M., Lubaszewski, W.: Can human association norm evaluate latent semantic analysis? In: Proceedings of the 10th NLPCS Workshop, pp. 92–104 (2013)
6. Griffiths, T.L., Steyvers, M.: Finding scientific topics. Proceedings of the National Academy of Sciences 101(suppl. 1), 5228–5235 (2004)
7. Landauer, T.K.: Handbook of Latent Semantic Analysis. University of Colorado Institute of Cognitive Science Series. Lawrence Erlbaum Associates (2007), http://books.google.pl/books?id=jgVWCuFXePEC
8. Leetaru, K.: Culturomics 2.0: Forecasting large-scale human behavior using global news media tone in time and space. First Monday 16(9) (2011)
9. Lubaszewski, W., Dorosz, K., Korzycki, M.: System for web information monitoring. In: 2013 International Conference on Computer Applications Technology (ICCAT), pp. 1–6 (2013)

10. Lytinen, S.L.: Conceptual dependency and its descendants. Computers and Mathematics with Applications 23, 51–73 (1992)
11. Minka, T., Lafferty, J.: Expectation-propagation for the generative aspect model. In: Proceedings of the Eighteenth Conference on Uncertainty in Artificial Intelligence, UAI 2002, pp. 352–359. Morgan Kaufmann Publishers Inc., San Francisco (2002), http://dl.acm.org/citation.cfm?id=2073876.2073918
12. Ortega-Pacheco, D., Arias-Trejo, N., Martinez, J.B.B.: Latent semantic analysis model as a representation of free-association word norms. In: MICAI (Special Sessions), pp. 21–25. IEEE (2012)
13. Rosch, E.: Principles of categorization. In: Rosch, E., Lloyd, B. (eds.) Cognition and Categorization, pp. 27–48. Erlbaum, Hillsdale (1978)
14. Schank, R.C.: Conceptual dependency: A theory of natural language understanding. Cognitive Psychology 3(4), 532–631 (1972)
15. Steyvers, M., Griffiths, T.: Probabilistic topic models. In: Latent Semantic Analysis: A Road to Meaning. Lawrence Erlbaum (2005), http://psiexp.ss.uci.edu/research/papers/SteyversGriffithsLSABookFormatted.pdf
16. Řehůřek, R., Sojka, P.: Software framework for topic modelling with large corpora. In: Proceedings of the LREC 2010 Workshop on New Challenges for NLP Frameworks, pp. 45–50. ELRA, Valletta (2010), http://is.muni.cz/publication/884893/en
17. Wandmacher, T.: How semantic is latent semantic analysis? In: Proceedings of TALN/RECITAL (2005)
18. Wandmacher, T., Ovchinnikova, E., Alexandrov, T.: Does latent semantic analysis reflect human associations? In: Proceedings of the Lexical Semantics Workshop at ESSLLI 2008 (2008)
19. Wettler, M., Rapp, R., Sedlmeier, P.: Free word associations correspond to contiguities between words in texts. Journal of Quantitative Linguistics 12(2-3), 111–122 (2005)

Chapter 12
Moodle and Freebase: A Semantically Sound Couple

Marek Kopel

Abstract. Today's ease of access and the volume of knowledge available at the tips of our fingers completely change rules of the game for learning and teaching. Preparing the Web for its 3.0 version i.e. the machine readable version, means publishing facts in a standardized format. It gives the opportunity for using the facts and the knowledge that can be derived not only to teach machines. This chapter proposes a new approach for preparing knowledge resources for learning and testing. The idea is to use structured facts, that Google uses for its Knowledge Graph, in e-learning environment to automate some parts of the education process.

12.1 Background

12.1.1 One Day Expert

Let's say I want to consciously buy a new computer or a smartphone. Consciously meaning I have enough information to make the decision "which one?" by myself. Before online knowledge I could only gather opinions and experiences of people I meet. Or I could rely on the recommendation of a man selling at the store. Now I am able to access expert opinions, professional reviews, global ratings and more. After going through all of the information, forming my own opinion I become and expert myself. I am ready to choose consciously.

But after a few months my friend asks me to help him choose a new computer he needs to buy. Since I haven't kept interest in the field of new computer hardware. I have already done my purchase, so why would I. So my knowledge, my expertise became obsolete. I am not an expert any more in this field. But I can become one again. With some extra effort. This kind of behaviour is becoming more and more

Marek Kopel
Wroclaw University of Technology,
Wybrzeze Wyspianskiego 27, 50-370 Wroclaw, Poland
e-mail: marek.kopel@pwr.edu.pl

A. Zgrzywa et al. (eds.), *New Research in Multimedia and Internet Systems*,
Advances in Intelligent Systems and Computing 314, DOI: 10.1007/978-3-319-10383-9_12

common with every field of knowledge. Even the fields where the facts and expert opinions don't change or become outdated so fast. For example: while visiting Rome, I might be an expert on Italian painters, architecture, food, local customs, and so on. But after a few month after coming back and not being involved in discussion on Italy, memory of those aspects fades away.

12.1.2 Semantic Web

Most usual the information found and used for becoming an expert in a topic comes from online sources. And today still most of the online data can be found as a free text in natural language. But efforts are made to allow machines become experts in a topic – the same way humans do, when needed. The so called intelligent user agents' are software that can consume online data and process it to be a useful source of knowledge for user. The intelligence' also concerns the aspect of inferencing new facts based on given data, but that is what knowledge is usually about. User agents can deliver the expertise that is needed, but the learning would be much quicker and more persistent than in case of human.

In order to make the user agents work, a special structure of data is needed for the machine to understand' it. Free text data in natural language may be comprehensible and even prefered by human, but machine has time trying to extract data from text without a proper ontology[1]. For example, with semantics of a word leaves' in context he leaves for work', human would have no doubt it has nothing to do with a word leaf'. But for computer it may not seem so obvious, hence the problems of web text search or the DBpedia Spotlight experiment[2]. That is where the idea of Semantic Web or Web 3.0 comes with help. The Semantic Web standard give structures and formats in which data on the Web shall be machine readable. This may mean the that there has to be a second redundant source of data for each human readable document or that an HTML document would hold single instance of data in both: the human and the machine readable formats. The first approach is based on the classic Semantic Web standard like RDF[3] with RDF/XML[4] or other triple serialization[5][6][7][8][9]. The second, more recent, approach is using semantic annotation to nest metadata inside an HTML document with standards like: RDFa[10], Microformats[11] or Microdata[12].

12.1.3 Freebase

The Web 3.0 idea provides a nice vision: the Web is a global, open database, where information may be retrieved using a structured, SQL-like, query language (e.g. SPARQL[13]). But in practice, an agent can make a better job while operating on a single, consistent, fail-safe database. Of course, the metadata may come from various sources over the Web and be presented in any of the Semantic Web formats, but a single instance, which is as trustable, complete, error-free as possible is always a safer solution. This is how Freebase[14] was designed. Freebase is an online database of semantic metadata, available on Creative Commons and maintained

mainly by its community members. It is the same to machines, which Wikipedia is to humans. Freebase is now the main part of Google's Knowledge Graph, usually seen as a structured data column on the right of the Google search results.

12.1.4 Moodle

Moodle is a Learning Management System used in education usually for e-learning purposes[15]. It's open source licence and modular architecture make it a useful platform for customizing. It comes with a number of built-in modules for creating learning resources, communicating and collaborating, interactive learning and testing. It also has a huge base of plug-in modules created by community for many custom e-learning scenarios. The modularity of the structure is recursive. It means that, for example: one may create it's own, custom module for making quizzes, but she may also use the preinstalled quiz module and only create a custom question type module. This way the whole quiz functionality is granted and the missing part, i.e. a custom question type (e.g. "listen and mark the key signature" type), is integrated into the platform.

12.2 Concept of Instant Learning and Testing

The main idea in this chapter is to use structured data, which in many cases may be used as knowledge, for creation of learning content. The slogan "tap into the knowledge of Google" may be – in this case – taken literally and made possible by combining ontologies and facts provided by Freebase and functionality for managing learning and testing activities provided by Moodle. As extreme as this may seem it is not far from basing business decisions on an expert system outcome. Of course, many of aspects of the process need human supervision, but still there is a lot of room for the automatization. The the machine stick to what it is good at, i.e. processing simple facts and let human think of the context in which the facts become knowledge.

12.2.1 Metaquestions

Online facts and resources may be used in any form of learning activity, but let us focus for the sake of this chapter on quiz type questions and knowledge testing. As a proof-of-concept a custom moodle module has been developed. The module allows creating multiple choice question while automatically filling the possible (right and wrong) answers from the Freebase query answer. This way only a metaquestion needs to be created by human and then the actual question plus answers instance is generated. Some examples of metaquestions, human needs to come up, are:

1. "Choose the actual members of the music band [parameter]",
2. "Choose movies in which [parameter] contributed as an actor",
3. "Choose the titles of books written by [parameter]",

4. ”Choose the flags of countries where [parameter] is the official language”,
5. ”Choose the visual artwork created by [parameter]”,
6. ”Choose organisms that belong to the class of [parameter]”.

The parameter may be a single value (e.g. mammals’ in case of the example 6.) or a list of values, which also may be retrieved from Freebase (e.g. list of all bands inducted into Rock and Roll Hall of Fame’ in case of example 1.). Still in the case of a single value, another value is needed for retrieving the incorrect answers for creating a question instance. Of course the generated instance may and usually need to be curated by human.

There are two more benefits for using Freebase as a source. First one is the aspect of internationalization. Since the machine readable facts need to be independent of the natural language, for each topic (addressed by identificator, called *mid*’) there exists a set of labels and descriptions for any specific language (usually coming from different language versions of Wikipedia). This way, if only is the metaquiesion formulated in many languages, the corresponding names in parameters and answers would automatically be translated and used in proper, localized version.

The other benefit is that each topic, when accessed by *mid*, may have a picture. And using pictures along with labels is not only a better experience, which is proved to allow faster learning. It is usually also a necessary form of information (see above example metaquestions 4. and 5.)

12.3 Prototype

A quick prototyping was made by modifying a standard Moodle question type module, e.i. */moodle/question/type/multichoice*. A simple hacking allowed installing the module again as a new instance called *multichoice_fb*. Than the module was changed to use Freebase API for displaying answer labels and images. At all times while using Freebase topics, they are referenced by their *mid* (machine id). Knowing its *mid*, the topic’s image can always be accessed using URI:
https://usercontent.googleapis.com/freebase/v1/image/mid. The label for the topic is available in JSON object returned by API at specific URI, e.g.:
https://www.googleapis.com/freebase/v1/topic/m/01mkq?filter=/type/object/name&lang=pl, where */m/01mkq* is the *mid* and parameter *lang* specifies the required language for the label. Output for this URI looks as follows:

```
{
 "id": "/m/01mkq",
 "property": {
  "/type/object/name": {
   "valuetype": "string",
   "values": [
    {
     "text": "Informatyka",
     "lang": "pl",
     "value": "Informatyka",
```

```
      "creator": "/user/mwcl_wikipedia_en",
      "timestamp": "2007-09-28T00:55:27.000Z"
     }
    ],
    "count": 45.0
   }
  }
}
```

12.3.1 MQL

The metaqueries, examples of which were given above, need the parameters and potential answers (correct and incorrect). Beside the natural language version a structured query is needed retrieve the required mids. In case of Freebase the query language is called MQL. In MQL, just like in the responses JSON syntax is used. This way, knowing the corresponding ontology, a query is most intuitive. The MQL query requesting the correct answers for the first example metaquestion with parameter set to ”Queen” looks as follows:

```
{
  "member": [{
    "member": [{
      "mid": null,
      "name": null
    }]
  }],
  "name": "Queen",
  "type": "/music/musical_group"
}
```

This query shows that in Freebase ontology node of type: */music/musical_group* contains child nodes *member*, specifying the role (instrument and period) and the actual person in another *member* subnode. Setting null as values for its fields, the query should respond with all possible values stored in those child nodes. The corresponding response to the query would look like this:

```
{
  "result": {
    "member": [
      {
        "member": [{
          "mid": "/m/01vn0t_",
          "name": "Freddie Mercury"
        }]
      },
      {
```

```
          "member": [{
            "mid": "/m/018y81",
            "name": "Roger Meddows Taylor"
          }]
        },

        ...

      ],
      "type": "/music/musical_group",
      "name": "Queen"
    }
  }
```

As stated earlier, a query for incorrect answers is also needed. Since the incorrect answers should be of the same type as the correct ones, the best query would be the one that gives a superset of the results with correct answers. In context of the example above, the following query would return members of all the bands inducted to the ”Rock and Roll Hall of Fame” in this century:

```
{
  "name": "Rock and Roll Hall of Fame",
  "type": "/award/hall_of_fame",
  "inductees": [{
    "date>=": "2000-01-01",
    "inductee": [{
      "member": [{
        "member": [{
          "mid": null
        }]
      }],
      "type": "/music/musical_group"
    }]
  }]
}
```

The query show that the ontology branch, from the *hall_of_fame* node to the *mid* of a band member, needs to be traversed as follows: *hall_of_fame –>inductees –> inductee –>member –>member –>mid*. A real instance of a question in moodle quiz using the example metaquestion is presented in Fig.12.1.

12.4 Conclusions, Problems and Dangers

Quiz-based learning, a concept similar to flashcards[16], may be used in traditional education, but in this case it would rather support becoming an expert in a narrow topic or lifelong learning[17].

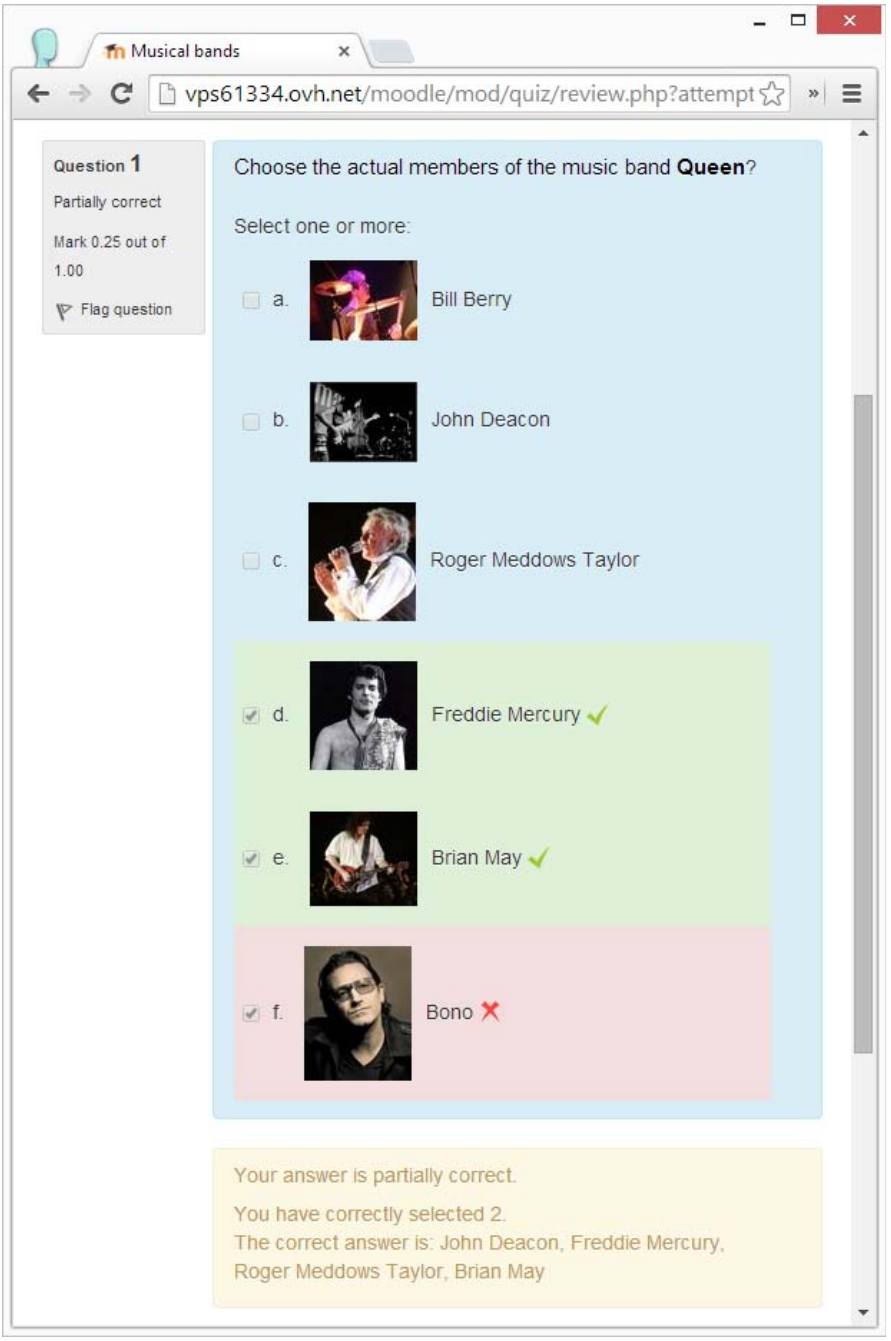


Fig. 12.1 Screenshot of an Freebase facts-based question in a moodle quiz result preview

Even though, the presented solution is meant mainly for self-directed learning, the role of a teacher or a supervisor is still crucial. First aspect, in which help from already an expert may be needed, is stating and testing the metaquestions. The metaquestions allow for generating automatically instances of actual quiz questions. But still there is a need for reviewing the instances before using them for the educational process. In case of an intermediate learner, this may not be a problem. However, a beginner would not be able to assess the educational value of generated instances.

Second, far more important, aspect is the authoritativeness of a data source. Since Freebase is maintained mainly by its community members and anyone can become a member, which is the same model that Wikipedia uses – the same potential risks must be considered. There are many examples showing that ease of access beats any other aspect for choosing an information source.

In 2009, as an experiment, a university student added a poetic, but phony quote to Wikipedia article on Maurice Jarre, hours after the French composer's death[18].The experiment was to test how globalized, increasingly Internet-dependent media is upholding accuracy and accountability in an age of instant news. Majority of the media, while reporting the composer's death, used the quote, without checking the sources. This is just one example showing the problem, which is not new for media, but with Internet it becomes a major one. Shane Fitzgerald, the student who run the experiment, commented later: "I am 100 percent convinced that if I hadn't come forward, that quote would have gone down in history as something Maurice Jarre said, instead of something I made up. It would have become another example where, once anything is printed enough times in the media without challenge, it becomes fact."

There are many cases where authoritative sources are updated with false data for less noble causes. Wikipedia holds a list of hoaxes, i.e. the "clear or blatant attempts to make up something, as opposed to libel, vandalism or a factual error. A hoax is considered notable if it evaded detection for more than one month"[19]. Considering the time and the number of cases in which the community maintained source of data may be corrupted, involvement of an expert reviewer in the process of generating quiz questions seems inevitable.

References

1. Guarino, N., Oberle, D., Staab, S.: What Is an Ontology? In: Staab, S., Studer, R. (eds.) International Handbooks on Information Systems, pp. 1–17. Springer, Heidelberg (2009)
2. Mendes, P.N., Jakob, M., García-Silva, A., Bizer, C.: DBpedia Spotlight: Shedding Light on the Web of Documents. In: Proceedings of the 7th International Conference on Semantic Systems, pp. 1–8. ACM (2011)
3. Miller, E., Manola, F.: RDF Primer, W3C Recommendation. W3C (February 2004)
4. Beckett, D.: RDF/XML Syntax Specification. W3C Recommendation. W3C (February 2004) (revised)
5. Gandon, F., Schreiber, G.: RDF 1.1 XML Syntax, W3C Recommendation. W3C (February 2014)

6. Seaborne, A., Carothers, G.: RDF 1.1 N-Triples. W3C Recommendation. W3C (February 2014)
7. Carothers, G.: RDF 1.1 N-Quads. W3C Recommendation. W3C (February 2014)
8. Carothers, G., Prud'hommeaux, E.: RDF 1.1 Turtle. W3C Recommendation. W3C (February 2014)
9. Lanthaler, M., Sporny, M., Kellogg, G.: JSON-LD 1.0. W3C Recommendation. W3C (January 2014)
10. Adida, B., Herman, I., Sporny, M., Birbeck, M.: RDFa 1.1 Primer, 2nd edn. W3C (August 2013)
11. Khare, R.: Microformats: The next (small) Thing on the Semantic Web? IEEE Internet Computing 10(1) (January 2006)
12. Hickson, I.: HTML Microdata, W3C Working Draft (March 29, 2012)
13. Harris, S., Seaborne, A.: SPARQL 1.1 Query Language. W3C Recommendation. W3C (March 2013)
14. Bollacker, K., Evans, C., Paritosh, P., Sturge, T., Taylor, J.: Freebase: A Collaboratively Created Graph Database for Structuring Human Knowledge. In: Proceedings of the 2008 ACM SIGMOD International Conference on Management of Data, pp. 1247–1250. ACM (2008)
15. Dougiamas, M., Taylor, P.: Moodle: Using Learning Communities to Create an Open Source Course Management System. In: World Conference on Educational Multimedia, Hypermedia and Telecommunications, pp. 171–178 (2003)
16. Kornell, N.: Optimising Learning Using Flashcards: Spacing Is More Effective than Cramming. Applied Cognitive Psychology 23(9), 1297–1317 (2009)
17. Knapper, C., Cropley, A.J.: Lifelong Learning in Higher Education. Psychology Press (2000)
18. Pogatchnik, S.: Student Hoaxes World's Media on Wikipedia, Msnbc.com (2014), http://www.nbcnews.com/id/30699302/ns/technology_and_science-tech_and_gadgets/t/student-hoaxes-worlds-media-wikipedia/ (accessed June 4, 2014)
19. Wikipedia: List of Hoaxes on Wikipedia. Wikipedia, the Free Encyclopedia (2014), http://en.wikipedia.org/w/index.php?titleWikipedia:List_of_hoaxes_on_Wikipedia (June 4, 2014)

Part III
Information Systems Applications

Chapter 13
Developing a Multi-Agent System for a Blended Learning Application

Dan-El Neil Vila Rosado, Margarita Esponda-Argüero, and Raúl Rojas

Abstract. Blended learning systems have become more popular than e-learning systems or even more than conventional educational methodologies. On Blended learning systems, the learners can view teaching materials asynchronously from different sources and collaborate with their peers and also they get the necessary face to face interaction with the instructor in the classroom. PowerChalk has arisen as the result from the analysis of several systems for E-Learning; it was designed to resolve an important limitation of current design methods: adaptability. In terms of adaptability, we have to consider that parallel to the evolution of e-learning methodologies, the intelligent agent paradigm has generated interest in many applications specially in order to support scaffolding activities and problem solving. It is for this reason that in this chapter presents the develop of a Multi-Agent System (MAS) on the PowerChalk system, a blended learning application that provide a robust, reliable, usable and sustainable multimedia technology for collaborative learning.

13.1 Introduction

Actually, teaching is supported by multimedia technologies at different levels; videlicet, the technology used in some educational organizations could range from just a simple personal computer up to a complete intelligent environment. In a technological sense, we can say that the gap between traditional and e-learning is narrowing but we need to take in account that e-learning does not replace obsolete existing pedagogical theories and approaches.

The technology-based learning is just an alternative to the traditional classroom model of a teaching-learning environment. The excuse for implementing Educational Technology, also called Learning Technology is based on reducing employee

Dan-El Neil Vila Rosado · Margarita Esponda-Argüero · Raúl Rojas
Freie Universität Berlin, Department of Mathematics and Computer Science,
Takustrasse 9. 14195 Berlin
e-mail: {vila80,esponda}@inf.fu-berlin.de,
Raul.Rojas@fu-berlin.de

A. Zgrzywa et al. (eds.), *New Research in Multimedia and Internet Systems*,
Advances in Intelligent Systems and Computing 314, DOI: 10.1007/978-3-319-10383-9_13

time away from the job of develop educational contents and shortening the amount of time students spend in their own learning; however, this technology has the drawback of the little consideration to delivering effective instruction in accordance with the medium, and also the problem to provide effective social transactions for learners and teachers. Unfortunately, in several educational technologies, the learning interaction is a one on one relationship between the student and the educational content.

Blended learning emerge like the convergence of on-line and face to face education. Josh Bersin gives a definition of blended learning as the combination of different training media (technologies, activities, and types of events) to create an optimum training program for a specific audience [2]. From this point of view, blended learning is a solution to integrate the different kinds of technological advances with the interaction offered in the traditional learning. Among the technologies to use in a blended learning session we can find blogs, wikis, on-line tools and on-line material, e-books, podcast and digital ink.

From the perspective of merchandising, several companies have entered to the development of learning technology (E-learning systems) which has generated scientific challenges and therefore a huge expansion of scientific groups focusing on E-learning systems. The different scientific groups have trying to develop systems with high quality, low cost and capable of revenue growth but above all solve the challenge of customer retention; however, research shows that up to 60% of any kind of implementations failed, despite improvements in technology, content development, increased knowledge and awareness [7, 16]. This situation does not mean that existing market products and applications are completely wrong, but shows that the developers have lead to the generation of new categories of systems with new capabilities and product configurations. Particularly, adaptability, the reuse of existing technologies, components and functionalities and the integration of different systems is an important issue that will lead the development of any Educational technology.

In general, a formative application should be interactive and provide feedback, have specific goals, motivate, communicate a continuous sensation of challenge, provide suitable tools, and finally avoid distractions and factors of nuisance interrupting the learning stream [11]. On the other hand a set of features specific for educational learning systems interfaces are: they have to provide a comprehensive idea of content organization and of systems functionalities, simple and efficient navigation, advanced personalization of contents and clear exit. Following these motivations, we developed PowerChalk as an interactive media tool that provides an adaptive, modern, flexible, technology-friendly and pedagogical approach suitable to any intelligent environment; specially those focused on blended learning [13].

The modular structure of PowerChalk let us amend quickly any problem in the system and evolve to new features or modules to improve any E-learning process [14]. As evidence of the adaptability of ”PowerChalk” in this chapter we propose a modular multilayered architecture for integrating agents and computational intelligence techniques in PowerChalk that could be used in a blended learning session for the purpose of enhancing learning and teaching.

The concept of multi-agent systems are broadly used in the development of large, complex systems by including attributes such as autonomy, cooperation, reactivity, adaptability and mobility [15]. Using intelligent and reactive agent in an e-learning architecture enables researchers to obtain a personalized e-learning system that adapts to the goals and characteristic of each learner [6]. With the modular multilayered architecture we want to extend the adaptability to the goals and characteristics of learners, teachers and developers.

This chapter is structured as follows. We review the related work in section 13.2. Then, in section 13.3 we describe the PowerChalk architecture, and in section 13.4 we describe the multilayered architecture for integrating agents in PowerChalk. Finally, in section 13.5, the implications of our findings and further research are discussed.

13.2 Related Work

Nowadays there are few electronic systems and projects that offer a combination of collaboration platforms, interactive chalkboards and displays that enhance any discussion session. Some representative examples of electronic systems and applications to give support to education and sharing information are:

- K-Sketch. Interface design for creating informal animations from sketches [5].
- E-Chalk. Electronic chalkboard developed by the Freie Universität Berlin [9].
- Cabri software. Interactive media software to create content faster to accompany any learning session with mathematical and physical objects or to provide activities as resources in 2 or 3 dimensions [12].

The systems above are specialized to a very specific task and have different limitations, among which we mention: cost, hardware or software limitations, inefficient software architecture, they can only work with certain types of data, lack proper software engineering, and present difficulty to evolve or update for developers or end-users. PowerChalk structure let us to get a sustainable application because resolve the difficulty to evolve or update as long as the capability to work with different kinds of data. These systems and many others do not use the agent technology.

On the other hand, several project implement learning systems based on multi-agents architectures. The use of intelligent agents makes it possible to achieve a powerful system adapted to the needs and characteristic of every learner and to provide adaptability and intelligence to the learning process through the introduction of agents. However, each of these systems use the multi-agent technology from a particular point of view, and in most cases the focus is on a specific agent type [8].

ABITS is composed by different kinds of agents (evaluation, affective and pedagogical agents) which extend a course management system to an automatic curricula generation [4]. The Intelligent Tutoring System (ITS) developed in [6] take in account the perspective of the students and the teachers. In this work, they have introduced a Student Model, a Domain model, a Pedagogical model and an Educational model. In this manner the ITS proposed gets all needed data, obtained fruit of

the interaction of the students with the system, to adapt the rhythm of introducing the contents of the matter to the learning rhythm of each student and the Education module obtains measures that permit to get recommendations to enhance the course. MASCE is a multi-agent systems for collaborative e-learning [10] that consider two types of users; namely students and instructors. This system consist of three types of agents, the Student, Instructor and Assistant agents.

The work of [8] offers a good list of previous and related studies based on intelligent agents. They justified that e-learning systems suffer from a lack of integration, the lack of re-use support and the lack of adaptation and also claim that the majority of web-based educational systems are currently from the class of Learning Management Systems, wich provide a large variety of support services to both learners and teachers but lack adaptability. In this sense, they develop an agent-based adaptive architecture to overcome these problems. The proposed architecture is based on Intelligent Blackboard agents (that provide preferences for designing and delivering educational content) and a formal design model (object Petri Net) used to verify and validate the model before the implementation.

In general, interesting results have been achieved by systems where pedagogical agents are developed regarding the student motivation and companion. Also, tutor agents are usually related to student modelling and educational decision taking. However, there is no systems that consider the characteristics of a blended learning approach and the respective interaction of a complete e-learning process.

13.3 PowerChalk Architecture

This blended learning application is composed of smaller, separated chunks of code that are well isolated. They can then be developed in separated teams with their own life cycle, and their own schedule. The results can then be assembled together by a separate entity. PowerChalk has a modularized architecture for distributed development based in Java-NetBeans technology.

Such a modular architecture has the followings advantages [3]:

- It simplifies the creation of new features.
- It makes it easy for users to add and remove features.
- It makes it easy to update existing features even to code level.
- Fast application development.

With these benefits PowerChalk becomes a modern, configurable approach to any blended-learning situation, lesson-planning or intelligent environment. The modular architecture of PowerChalk is shown in figure 13.1.

13.4 Agent Based Multilayered Architecture

In this chapter we start from the idea of having a tool for learning sessions. So we have to consider the development and structure of an E-learning process. On the other hand, different architectures have been developed to design Educational

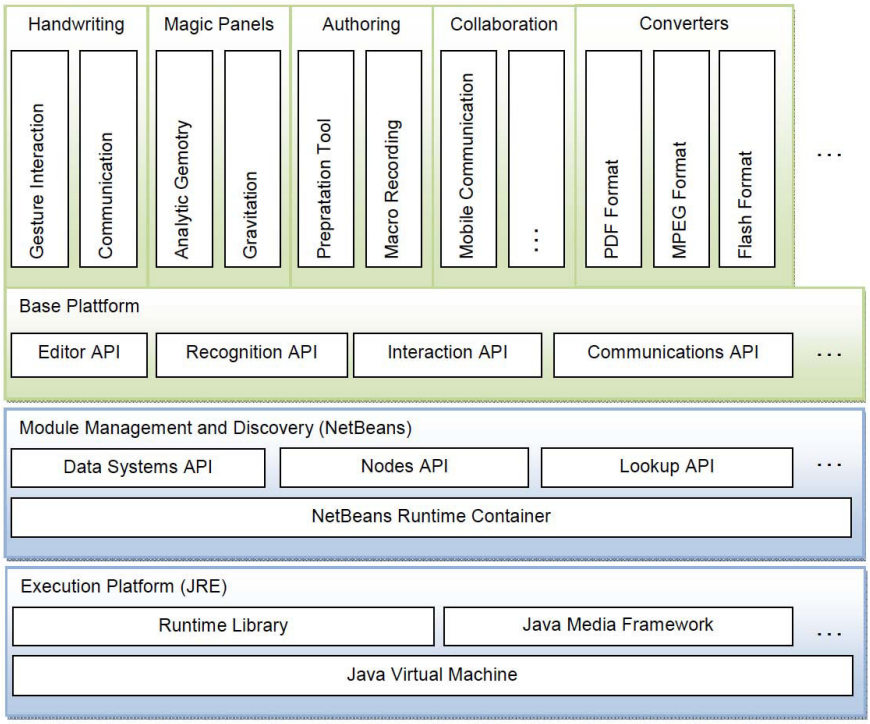


Fig. 13.1 Modular structure of PowerChalk

content. Some interesting architectures take in account just one-level of development and implement direct agents to the users based on student's personal profile [10, 7, 6]. Agent based multilayered architectures are developed in [8, 1]. The advantage of a multilayer architecture are: Makes a logical separation of tasks and services for the end-users (teachers and students), developers and agents; easy to define the respective tasks of the agents in the development process; any change to the elements on the layer can be made in one place and be available to the whole application; it is possible to change the contents of any layer without having to make corresponding changes of the others and enables parallel development of the different layers of the application.

In this chapter we focus in the development of a multilayer architecture based on agents for the different stages of any e-learning process.

13.4.1 E-learning Process

The e-learning process can be divided into two major phases: content development and content delivery and maintenance (figure 13.2). A typical e-learning process has planning, design, development, evaluation delivery and maintenance stages [7]. On these schema the participating roles are: learner, teacher and administrator. The

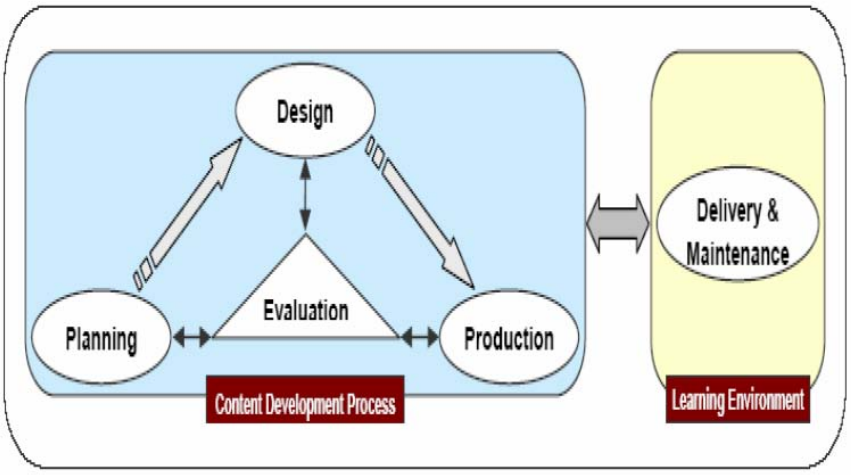


Fig. 13.2 The iterative process of E-learning

number of individuals involved in the stages of a learning session may change. Even some roles ad responsabilities may overlap.

From this schema we can see that the e-learning process is iterative in nature and the individuals involved in the stages should be in contact with each other. In this chapter, we propose a set of agents to every phase, namely, a set of agents to the content development process and another one to the learning environment.

The modular architecture of PowerChalk let us to identify the modules focussed on every stage of the e-learning process and develop several kind of agents. In the general case, agents will be acting on behalf of the stage or some specified user with different goals and motivations. We build agents in order to achieve task for the different roles involved. The attributes of an agent in the PowerChalk system are: persistence, autonomy and reactivity.

13.4.2 Content Development Based on Agents

We propose the following architecture 13.3. User agent is used for identification in order to get individual user profiles. The content agent delivered content need request to data layer and the user multiagent system (user MAS) communicate between a user (normally a teacher) and system, to create educational content. In the interface layer we can found agents whose activities work in the editor module of PowerChalk [13].

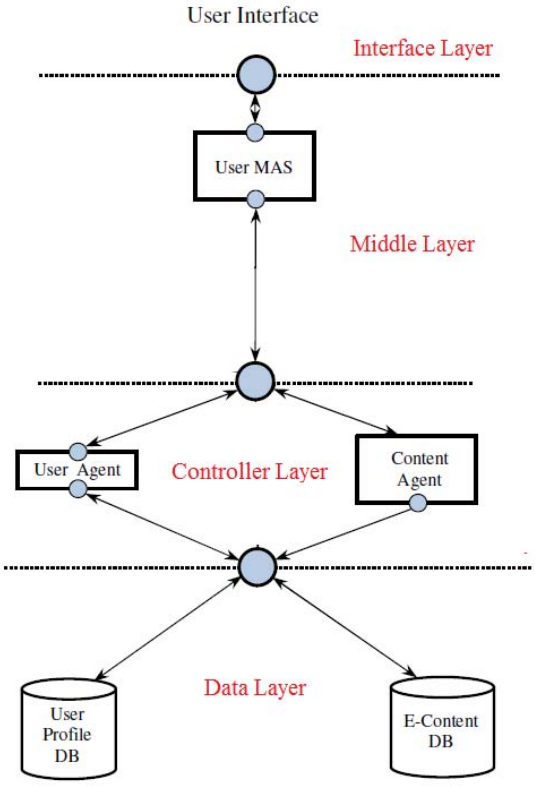


Fig. 13.3 Content development multilayered architecture based on agents

13.4.3 Content Delivery Based on Agents

In terms of the content delivery, we focused in the development of agents to interact with the different modules that work with the editor module of PowerChalk. So that the multilayered architecture is the same like the PowerChalk structure (See section 13.4).

Examples of this agents are: development of agents for the improvement of the visualization of the strokes (Handwriting module), agents for the magic panels (improvement of rendered animations), imple mentation of agents that help tothe visualization of the information on the canvas editor in another formats (converters module), etc.

13.5 Conclusions and Future Work

The consensus is that PowerChalk it is a collaborative, robust, reliable, usable and sustainable interactive multimedia system for blended learning and also a friendly tool to review and share information on-line or off-line. Supply PowerChalk with a set of agents preserves the pedagogical benefits of the traditional chalkboard and provides the possibility to present educational content in a learning session in different ways. Combining the advantages of an interactive multimedia tool and the faculty to be easily updated and adapted, the modular architecture allows it to be an useful and efficient method to embed agents to improve planing, design and production of a learning session as well as enhance the delivery of educational content on this blended learning application. It must be pointed out that the modular structure of PowerChalk let us amend quickly any problem in the system.

A complete usability test it is in progress, in order to assess in a measurable way the advantages of modular programming in PowerChalk and the implementation of the different agents. Also, we are increasing the efficiency of every module and the functionalities of the agents. In this sense, a short-term objective is the development of agents to support an Intelligent Tutoring System for the learners. It has considered the development of agents with data-mining techniques to generate personalized Student and Pedagogic Models. The goal of this set of agents is that the alumni learn more and better, taking in account the specific rhythm and learning style of the students.

Otherwise, teachers and students have been proposing new PowerChalk modules to design and develop. Among the future modules we can find: Algortihms animations, virtual labs sketched, etc. In this manner the development of agents focused on the performance of these modules always will be taken in consideration.

References

1. Ali, A., Dehghan, H., Gholampour, J.: An agent based multilayered architecture for e-learning system. In: 2010 Second International Conference on E-Learning and E-Teaching (ICELET), pp. 22–26. IEEE (2010)
2. Bersin, J.: The blended learning book: Best practices, proven methodologies, and lessons learned. John Wiley & Sons (2004)
3. Boudreau, T., Tulach, J., Wielenga, G.: Rich client programming: Plugging into the netbeans platform. Prentice Hall Press (2007)
4. Capuano, N., Marsella, M., Salerno, S.: Abits: An agent based intelligent tutoring system for distance learning. In: Proceedings of the International Workshop on Adaptive and Intelligent Web-Based Education Systems, ITS (2000)
5. Davis, R.C., Colwell, B., Landay, J.A.: K-sketch: a'kinetic'sketch pad for novice animators. In: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, pp. 413–422. ACM (2008)
6. Gascueña, J.M., Fernández-Caballero, A.: An agent-based intelligent tutoring system for enhancing e-learning/e-teaching. International Journal of Instructional Technology and Distance Learning 2(11), 11–24 (2005)

7. Giotopoulos, K.C., Alexakos, C.E., Beligiannis, G.N., Likothanassis, S.D.: Integrating agents and computational intelligence techniques in e-learning environments. In: IEC, Prague, pp. 231–238 (2005)
8. Hammami, S., Mathkour, H.: Adaptive e-learning system based on agents and object petri nets (aels-a/opn). Computer Applications in Engineering Education (2013)
9. Jantz, K., Friedland, G., Rojas, R.: Ubiquitous pointing and drawing. International Journal of Emerging Technologies in Learning 2(1) (2007)
10. Mahdi, H., Attia, S.S.: Developing and implementing a multi-agent system for collaborative e-learning
11. Norman, D.A.: Things that make us smart: Defending human attributes in the age of the machine. Basic Books (1993)
12. Straesser, R.: Cabri-geometre: Does dynamic geometry software (dgs) change geometry and its teaching and learning? International Journal of Computers for Mathematical Learning 6(3), 319–333 (2002)
13. Vila Rosado, D.E.N., Esponda-Argüero, M., Rojas, R.: An adaptive interactive multimedia system for intelligent environments. International Journal of Information & Education Technology 4(1) (2014)
14. Vila Rosado, D.E.N., Esponda-Argüero, M., Rojas, R.: Modular architecture for pen-based digital ink on blended learning applications. International Journal of Information & Education Technology 4(2), 189–193 (2014)
15. Wooldridge, M., Jennings, N.R.: Intelligent agents: Theory and practice. Knowledge Engineering Review 10, 115–152 (1995)
16. Zemsky, R., Massy, W.F.: Thwarted innovation: What happened to e-learning and why, a final report for the weatherstation project of the learning alliance at the university of pennsylvania in cooperation with the thomson corporation (June 2004)

Chapter 14
Using ACS for Dynamic Traveling Salesman Problem

Andrzej Siemiński

Abstract. The chapter addresses the problem of optimizing the performance of the Ant Colony Optimization (ACO) technique. The area of study is the Travelling Salesmen Problem (TSP) in its static and dynamic version. Although the individual ants making an Ant Colony are remarkably simple their interaction makes the process so complex that an analytical approach to optimize the parameters that control the Ants Colony is not yet possible. Therefore its performance is analyzed optimized on an experimental basis. The experiments strongly suggest that the observed performance is mostly effected by the colony size. In particular the usefulness of the colonies with a very large number of ants, the so called Hyper-Populated Ant Colonies is stressed.

14.1 Introduction

The aim of the chapter is to discuss the problem of optimizing the performance of Ant Colony Optimization (ACO) used for both static and dynamic environments of Travelling Salesman Problem. The operation of the ACO is controlled by a number of parameters with values that are usually chosen in an experimental manner. This is due to the complexity of its operation that defies an analytical solution. A detailed analysis of the operation of the ACO for traditional static graphs has reviled that of all of the parameters of the ACO it is the population sized size is probably the most significant one. The reported in the chapter experiments clearly indicate that it also remains true for dynamic environments. This is a promising result as it offers a relatively simple way of optimizing the ACO performance.

Andrzej Siemiński
Institute of Informatics, Wrocław University of Technology
Wyb. Wyspiańskiego 27, 50-370 Wrocław, Poland
e-mail: andrzej.sieminski@pwr.wroc.pl

A. Zgrzywa et al. (eds.), *New Research in Multimedia and Internet Systems*,
Advances in Intelligent Systems and Computing 314, DOI: 10.1007/978-3-319-10383-9_14

The chapter is organized as follows. The second section presents the Travelling Salesman Problem. It still remains one of the most challenging problems in the AI area. The 3rd section is devoted to the introduction to Ant Colony Optimization and specifies measures for measuring its performance for both static and dynamic graphs. Related work in the area is presented in the next section. The next 5th section discusses the proposed solution and presents the results of experiments that confirm its usefulness. Plans for further work conclude the chapter .

14.2 Traveling Salesman Problem

The travelling salesman problem (TSP) is one of the classical problems of Artificial Intelligence. It could be stated in a remarkably simple way: given a list of cities and the distances between each pair of cities, what is the shortest possible route that visits each city exactly once and returns to the origin city? The problem was stated for the first time as early as in 1800's. At that time it had some practical implications but was treated mainly as a recreational puzzle.

The number of all possible different routes for a graph with n nodes is equal to n!. The number is estimated by:

$$\sqrt{2\pi n} \quad \left(\frac{n}{e}\right)^n \tag{14.1}$$

and even for relatively small values of n like 50 it is equal approximately to 3,04141E+64. This calls for heuristic solutions as the complete search is not feasible. The TSP has received considerable amount of interest in the scientific circles in the 1950's and in 1970's it was proved to be a NP-hard problem. Since then it is one of the most intensively studied problems in optimization and is also used for other application areas besides route planning. It is also used as a benchmark for many optimization methods.

In a classical statement of problem the distances between nodes are symmetric and do not change. This seems to match the real life where the road structure remains relatively static. Having said that we should bear in mind that minimizing the distance is not what we have really interest in. In more practical objectives include e.g. the travelling time which is subject to the ever changing road conditions and therefore is inherently dynamic. In what follows we will continue to use the word distance to describe the time of moving from one node to another rather than the physical distance that separates them. This calls for dynamic graphs with distances changing all the time.

14.3 Ant Colony Optimization

The Ant Colony Optimization is an example of the Swarm Intelligence concept. The concept includes a wide variety of techniques which solve problems by a

collective behavior of decentralized, self-organized agents which may be natural or artificial. Although the individual agents are simple their interaction with each other and their environment make them capable to solve complex problems. The ACO was described for the first time by M. Dorigo in his PhD thesis [1] in 1992. Since then the researcher remains one of the key scientists active on the field. His extensive account of the state of art of ACO is presented in [2].

An ant could be regarded as an extremely simple agent. All it could do is to move from one node to another laying a pheromone trail on its way. It is also capable of detecting its current position, remembering the nodes it has visited so far and sensing the direct distances from its current position to other nodes and also the amount of pheromone laid on them. A set of ants forms an Ant Colony. The colony works in iterations. At the start or each iteration the ants are placed randomly on the graph. In each step of an iteration an ant selects most valuable node that was not visited so far. The iteration stops when all cities are included in an ants' route. The Ant Colony is not just a set of ants but it also harvests the collective intelligence of individual ants. The intelligence takes form of the pheromone levels. A colony remembers the performance of the best ant in the current iteration, the best so far ant and it globally updates the pheromone levels.

Table 14.1 The ACO recommended parameter values for N nodes according to [3]

Name	Description	Suggested Value
N	Number of Ants	N
Q0	Probability of selecting exploitation over exploration	0.8
α	Aging factor used in the global updating rule	0.1
β	Moderating factor for the cost measure function	2.0
ρ	Aging factor in the local updating rule	0.1

The basic operations defining ACO are defined by three rules:

- State transition rule that specifies in what manner an ant selects the next node.
- Local updating rule that describes the way of updating pheromone values by an ant as it finds its route.
- Global updating rule which is initialized at the end of each iteration by a colony and effects the pheromone values of the whole graph.

The Travelling Agent Problem (TSP) was the first application area of the ACO. The performance measures include:

- BsF_j (Best so Far) - the best (shortest) route found in up to j iterations
- IB_j (Iteration Best) - the best (shortest) route found the in j-th iteration.
- CCj (computational complexity) number of ants * number of nodes * number of iterations

14.4 Related Work

We believe that to suggest solutions for the dynamic TSP an analysis of its static version is necessary.

14.4.1 Static TSP

The parameter valued presented in the Table 14.1 were identified in a series of experiments. An attempt to make the search more extensive was described in [4] and [5]. The former paper attempted to automate the process of parameter optimization by using an algorithm that was inspired by a combination of Evolutionally Programming (EP) [6] and Simulated Annealing (SA) [7]. Each of the genes that make up a chromosome represented a single parameter of the ACO algorithm. The population of chromosomes evolved using the typical for EP selection and mutation mechanisms. The scope of mutation was controlled by an algorithm adopted from Simulated Annealing. Some of the reported results were slightly better than the reference results obtained for the recommended values. However the parameter values were either not correlated. It should be stressed that in the study the number of ants was in the range from 10 to 50 so it has not changed much.

14.4.2 Dynamic TSP

The Dynamic TSP was introduced for the first time by Psaraftis in [8]. His work however was discussed the general properties of dynamic TSP, discussing performance measures and not on proposing any solutions. For the study the DTSP the basic factor is the graph modification scheme.

The first attempts to adopt the standard operation of ACO to the dynamic case concentrated upon introducing global and local reset strategies [9] and [10]. A change in the distances invalidates part of accumulated pheromone levels. Global reset starts the optimization process once more again and is used mainly for reference purpose. Local reset changes only the pheromone levels for modified segments of graph. It enables the Colony to exploit at least part of data gathered so far but requires a precise data on where the change had occurred. let us note also that both of the reset strategies are less efficient or even entirely not useful in the case of constantly changing environments.

In an another approach the ant population diversity necessary to adopt to changing environment was achieved by the implementation of the so called immigrant schemes. There are three types of immigrants: randomly generated, elitism-based, and hybrid immigrants. The approach could use a long-term memory as in P-ACO [10]. In a more recent paper a short-term memory is used [11]. The study reviled that different immigrants schemes are advantageous under different environmental conditions.

The crucial element of a dynamic environment is the way it changes. In the studies mentioned above the modifications ware simple. They included only a single

node deletion or introduction. More elaborate schemas for graph modifications were considered in [11]. Modifying a graph was a process that consisted of a sequence of node insertions and deletions performed at certain intervals. Still these approaches make two rather unrealistic assumptions:

1. The majority of distances do change in a single burst of graph modification activity. What is changed is the structure of the graph and not the distances.
2. It is assumed that the Ant Colony knows instantly where the changes have taken place and what is their scope. This may look innocent at first but it really breaches the very nature of ACO. The colony consists of simple ants having very limited knowledge about their surroundings and there is no room for omniscient Colony Manager.

An earlier attempt to study more realistic dynamic graphs was introduced in [12]. The graphs are replaced by graph generators that potentially could change in a parameterized manner any distance in the graph. The approach unfortunately looks promising but it has one deficiency: the more complex graphs based on Markov sources could not guarantee that the average distance length will not change. This in itself could happen in real life but it makes it difficult to evaluate the performance of tested algorithms.

The described below graph generators preserve the average distance length so it is possible to relate the observed performance and the graph change scope to the reference performance of the well studied static version of the graph.

14.5 Proposed Solution and Its Verification

The described above attempts to identify near optimal values for the ACO parameters have not concluded with a specification of a clear set of rules even for static graphs. The standard set of parameters have proved to provide more than acceptable and consistent performance. In what follows we try to study the influence that the changes in an ant colony population size have on the way ACO operates. In doing so we first analyze more deeply the static graphs case. The gained in that way insight is applied to the more challenging case of dynamic graphs. As far as we can say the impact of ant population size was not studied extensively before. This is probably due to the notion that increase in population size results in a direct increase of processing time what is generally considered to be a serious disadvantage e.g. in the paper [4] the fluctuation of population was in the range 30 to 50.

In all experiments the graphs consisted of 50 nodes. The distances between them were in the range from 0.0 to 1.0. The parameters we set to the values recommended by [3] that is: $\alpha = 0.8$, $\beta = 0.1$ and ρ=0,1. As in other papers [4] and [12] the experiment used the JACSF – Java Ant Colony System Framework for TSP developed and made available to the research community by U. Chirico [3]. The modifications to the basic framework included the replacement of static graph by a graph generator and the inclusion of various monitoring tools.

In the subsequent experiments the number of ants were in the range from 30 up to as high as 1000. The usual value is equal to the number of nodes. The number of ants in the more numerous Colonies deviates very much from the usually used valued and therefore they are referred to as Hyper-Populated Ant Colonies.

14.5.1 Static Graphs

Let us start with an analysis of the performance of the standard algorithm. We belief that the insight into to operation of static graph is a prerequisite for suggesting solutions for the more complex, dynamic graph. Therefore, we have interest not only in the BsF value but also in the way it was obtained.

There are two performance measures for the static graphs:

BsFj - Best so Far route length
BIN - Best iteration number.

The Figure 14.1 depicts for each iteration the currant best path, the best so far path and the gains. A gain is defined as follows:

$$Gain_i = (BsF_i - BsF_{i-1}) * 10. \tag{14.2}$$

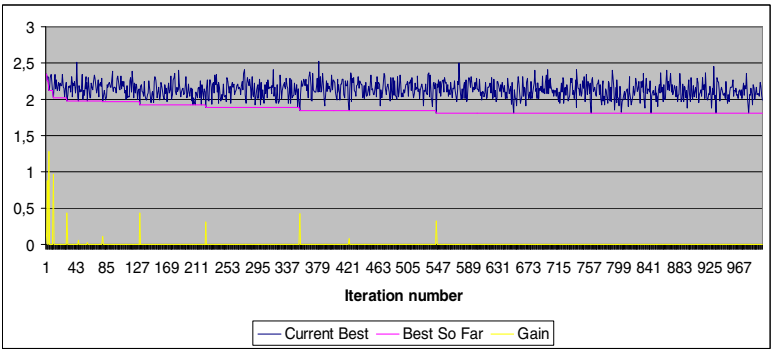


Fig. 14.1 The performance of a standard version of ACO

The criterion for stopping the optimization process is the value of $Gain_i$. The yellow gain peaks mark iterations that contribute to shortening the BsF(i) . The picture is not promising. As the value of j increases the values of the $Gain_j$ <> from 0 are far and far apart from each other. Their size remains relatively the same. This means that although we can expect some decrease in route length but it is not guaranteed by any means. This is illustrated by the data presented in the Table 14.2. It shows the BsF iteration numbers and their respective route lengths for 10 runs with the maximum number of iteration set to 5000.

Table 14.2 Observed Best Iteration Number for standard set of parameters

Route Length	Best Iteration Number
1,783418336	40
1,783418336	106
1,783418336	356
1,781135524	639
1,805144111	965
1,783418336	1956
1,783418336	2709
1,800436635	3827
1,730822148	3951
1,783418336	4982

The route lengths fluctuate not more then 35% but the lowest Best Iteration Number (BIN) is equal to 40 and the highest to 4982. To make things even worse the route length in both cases is identical. This means that the computational effort in the former case is more then two order of magnitude lower than in the latter case. In exactly half of the cases the BIN was <1000. We are lured into increasing the iteration number but the computational effort becomes less and less productive.

In the next step the influence of the Ant Colony size on the overall performance of optimization process. The cumulative number of gain peaks is observed in 10 runs is presented on the Figure 14.2. The size of Ant Colony varies from 30 to 150.

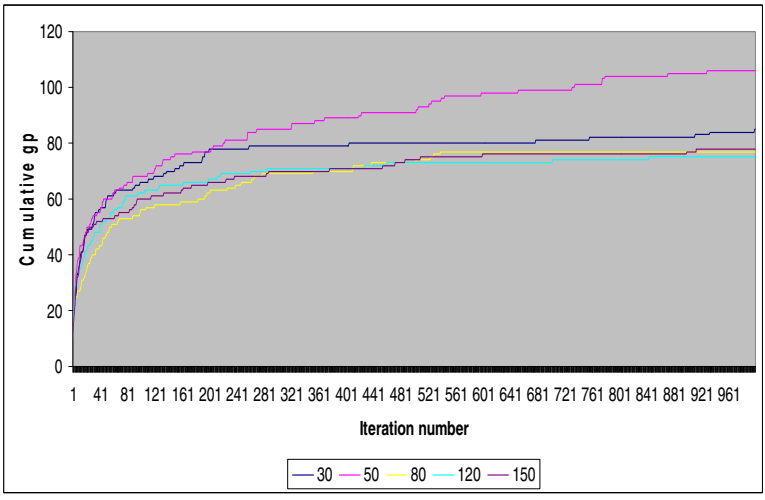


Fig. 14.2 The cumulative gain number for different number of Ants; number of runs is 10

The highest line represents is for the usually used number of ants. It means that the improvement is steady over a long range of iterations. This is a welcome result for a static environment but is a clear disadvantage for dynamic environments. The next shows the BsF routes found by Ant Colonies of different sizes. To compensate for the larger computational effort necessary to complete an iteration the maximal number of iterations has been decreased, see Table 14.3. As a result the test run time was the same for each Ant Colony size.

Table 14.3 The average BSF for different number of ants and fixed and normalized number of iterations (NI)

Ant Colony Size	NI	BSF for 1000 itereations	BSF for normalized NI
30	1666	1,824843861	1,806816153
50	1000	1,812277212	1,812277212
80	625	1,799868455	1,80
120	417	1,80206432	1,85
150	333	1,787258566	1,798
1000	50	1,783418336	1,78

For the number of iterations set to 1000 we observe a small but noticeable decrease in BsF length. The more numerous Ant colonies require more computations to complete an iteration. To compensate the effect during the experiments the number of iterations have been reduced the more numerous ant colonies so to preserve CC. The results are given in the last column of the table. The results obtained by the colonies with large population of ants remain on the same level as the for 1000 iterations. This is a very promising result as it shows that increasing the population size reduces the route length even when the computational complexity is taken into account. The reason for that phenomena is probably the small number of peaks observed for iteration numbers > 100 that is observed on the Figure 14.2.

14.5.2 Dynamic Graphs

The dynamics of standard graphs used in the majority of papers comes down to adding and deleting a controlled number of nodes. This clearly does not reflect the real life cases where the road structure itself remains stable for a considerable period of time. Unlike the distances the travel times vary all the time due to the ever changing road conditions. In our opinion the tour time is more useful as performance measure than the tour length.

Graphs undergoing permanent changes were firstly introduced in [12]. In the chapter a modification is used of the graph generators defined there. The modifications insure, that the average distance between nodes remains stable. This enables to relate the performance to the reference performance achieved for a static graph.

Two parameters control the variability of a graph generator:

sc - the scope of changes, that is the share of distances that are modified
rg - the range of changes , that is the maximal value of change.

The following Java code used to generate a new distance between nodes:

```
public synchronized double newDistance(double oldDist) {
double res=oldDist;
double x= randGen.nextDouble();
if (x>sc)
        return(res);
x= s_randGen.nextDouble()*rg
if (randGen.nextDouble()>0.5) { // increasing distance
        res=oldDist+x;
        if (res>1.0)  res=res-1.0;  // distance normalization
} else { // decreasing distance
        res=oldDist-x;
        if (res<0.0)  res=1.0+res;  // distance normalization      }
return (res);
}
```

The properties of the two dynamic graph generators discussed in the chapter are in the Table 14.4.

Table 14.4 Graph generators used in the experiments

Name	Parameters	Interpretation
CCc - constant changes	Sc= 0.1 Rg= 1.0	Every tenth distance is subject to a random change.
DCd - damped changes	Sc =0.9 Rg = 0.5	Almost all distances change but the range of changes is damped by the 0.5 factor

The CCc graph is moderately changeable, the range of changes is not limited but the changes effect one tenth of all segments. The changes in the graph DCd effect the overwhelming majority of segment lengths but the their scope is damped by the factor of 0.5. The conducted experiments included 10 test runs for each generator with the number of ants ranging from 30 to 1000. In the case of dynamic graphs the BsF routs could be make obsolete after each iteration and therefore the value of IB is used as a performance measure. The results are shown in the Table 14.5.

Table 14.5 The average IB for different number of ants, nymber of runs =10, number of iterations 1000

Number of ants	Graph Generator CCc	Graph Generator DCd
30	2.40	3.83
50	2.34	3.31
80	2.32	2.98
120	2.31	2.80
150	2.29	2.71
1000	2.17	2.44

As expected the with the increase of the ant number the impact of the graph changeability decreases. For 1000 ants the IB does nor diverge much from the performance of ant colony of standard size for the much less changeable generator CCc. It is true that the computational effort is much higher for the former case but even the moderately sized colony with 150 ants performs much better than the colony with standard size.

14.6 Conclusions and Future Work

The ACO version used for the TSP exhibits a remarkable power to adopt itself to diverse set of parameters values and produce acceptable results for a variety of input data. This is a welcome feature but it makes hard to identify guidelines for the selection of near-optimal parameter values. The studied reported in [4, 12] have produces sets of parameters slightly better than the recommended values but they are obtained in an experimental and not a analytic manner. This make them hard to use in dynamic environments.

The chapter concentrated upon the analyzing of the impact that the changing of the Ant Colony population. The experiments indicate that the increase in this number has is clearly connected with the shortening of the found routes. The correlation is noticeable for the traditional static TSP and becomes even more apparent for dynamic environments. For the most dynamic graphs very numerous Ant Colonies, called HP (Hyper-Populated) Ant Colonies performed the best. What should be stressed they perform better than regular colonies even utilizing the same level of computational complexity.

The correlation between the colony population size and its performance is clear but the optimizing its size with respect to the computational complexity and graph changeability remains to be studied.

We can not expect dramatic improvement of processing speed in a traditional environment. There is not much space left for the increasing of clock frequency. On the other hand the ACO is well suited for parallel processing. Each ant or a group of ants could work as a process on a separate computer. In the experiments

described in the chapter each ant was implemented as a Java thread so all of them run within one process. Therefore the processing time was the same for all runs with the same computational complexity no matter what the number of ants was. The full potential of HPAC could be fully utilized in truly parallel processing environments offered e.g. by Hadoop [13] or cloud processing. One could expect in that case not only better results but also the shortening of the processing time. The next planned step of the research is the porting the ACO to the Hadoop environment.

References

1. Dorigo, M.: Optimization, Learning and Natural Algorithms, PhD thesis, Politecnico di Milano, Italie (1992)
2. Dorigo, M., Stuetzle, T.: Ant Colony Optimization: Overview and Recent Advances, IRIDIA – Technical Report Series, Technical Report No. TR/IRIDIA/2009-013 (May 2009)
3. Chirico, U.: A Java framework for ant colony systems. In: Ants 2004: Forth International Workshop on Ant Colony Optimization and Swarm Intelligence, Brussels (2004)
4. Siemiński, A.: TSP/ACO Parameter Optimization; Information Systems Architecture and Technology; System Analysis Approach to the Design, Control and Decision Support, pp. 151–161. Oficyna Wydawnicza Politechniki Wrocławskiej (2011)
5. Gaertner, D., Clark, K.L.: On optimal parameters for ant colony optimization algorithms. In: IC-AI, pp. 83–89 (2005)
6. Eiben, A.E., Smith, J.E.: Introduction to Evolutionary Computing. Springer (2003)
7. Busetti, F.: Simulated Annealing Overview, Report (2003)
8. Psarafits, H.N.: Dynamic vehicle routing: status and prospects. National Technical Annals of Operations Research. University of Athens, Greece (1995)
9. Guntsch, M., Middendorf, M.: Pheromone modification strategies for ant algorithms applied to dynamic TSP. In: Boers, E.J.W. (ed.) EvoWorkshops 2001. LNCS, vol. 2037, pp. 213–222. Springer, Heidelberg (2001)
10. Guntsch, M., Middendorf, M.: A population based approach for ACO. In: Cagnoni, S., Gottlieb, J., Hart, E., Middendorf, M., Raidl, G.R. (eds.) EvoWorkshops 2002. LNCS, vol. 2279, pp. 72–81. Springer, Heidelberg (2002)
11. Mavrovouniotis, M., Yang, S.: Ant colony optimization with immigrants schemes in dynamic environments. In: Schaefer, R., Cotta, C., Kołodziej, J., Rudolph, G. (eds.) PPSN XI. LNCS, vol. 6239, pp. 371–380. Springer, Heidelberg (2010)
12. Siemiński, A.: Ant colony optimization parameter evaluation. In: Zgrzywa, A., Choroś, K., Siemiński, A. (eds.) Multimedia and Internet Systems: Theory and Practice. AISC, vol. 183, pp. 143–153. Springer, Heidelberg (2013)
13. Tom, W.: Hadoop the Definite Guide. O'Reilly (2011)

Chapter 15
Adaptive Heuristic Colorful Text Image Segmentation Using Soft Computing, Enhanced Density-Based Scan Algorithm and HSV Color Model

Adam Musiał

Abstract. This chapter describes intelligent multilayer image segmentation algorithm for Optical Character Recognition system. The additional algorithm brings new ability of recognizing colorful texts. Presented solution allows to successfully recognize texts with various background and foreground colors even when their luminosity values are equal. It may be adapted into current OCR systems or work as a standalone pre-processing system.

15.1 Introduction

Optical Character Recognition is one of the most famous example association of Artificial Neural Network (ANN). Many introductory academic descriptions of ANN capabilities present an example with one character recognition problem. Usually, in such examples the character is normalized and there is no need of pre-processing. Moreover, there is often one more assumption about color set present. Most plain OCR applications assume that there is no need to process colors and they envision that every paper contains black text and white background. This is the very good assumption for most cases, but it will not work with inverted or colorful text images.

In this chapter author wanted to go beyond that limit and to create an intelligent system which is able to successfully recognize text images with uncommon color set. This chapter describes how it was done and what example results are.

Adam Musiał
Institute of Information Technology,
Lodz University of Technology,
ul. Wólczańska 215, 90-924 Łódź, Poland
e-mail: 800334@edu.p.lodz.pl

A. Zgrzywa et al. (eds.), *New Research in Multimedia and Internet Systems*,
Advances in Intelligent Systems and Computing 314, DOI: 10.1007/978-3-319-10383-9_15

15.2 Example Base OCR System

15.2.1 Before Improvement

Let the example base system for the current research will be a working academic Optical Character Recognition system described in [2]. The system was optimized in many ways and used some innovative solutions to bring the accuracy ratio even higher. It was able to successfully process text images with significant noise and visible luminosity distortion. There were a lot of achieved image improvement techniques and system worked even for contrastive colorful texts.

15.2.2 Example Image That Exceeds Grayscale Algorithms Capabilities

Unfortunately, the grayscale conversion could be one of the biggest potential drawbacks which remained in the algorithm as a constraint. The reason is luminance calculation. Most common luminance expressions are as follow:

- plain RGB average:
$$\mathrm{l}_1(P) = \tfrac{1}{3}r + \tfrac{1}{3}g + \tfrac{1}{3}b; \tag{15.1}$$
- standard, objective luminance [4, p.21]:
$$\mathrm{l}_2(P) = 0.2125r + 0.7154g + 0.0721b; \tag{15.2}$$
- luminance, perceived [4, p.24]:
$$\mathrm{l}_3(P) = 0.299r + 0.587g + 0.114b. \tag{15.3}$$

where:

- $r \in [0;255]$– red subchannel value; $g \in [0;255]$– green subchannel value;
- $b \in [0;255]$– blue subchannel value; $P = \begin{bmatrix} r & g & b \end{bmatrix}$ – pixel descriptor;
- $l_n \in [0;255]$– luminosity value computed using n-th method.

One may easily see, that the three values are converted into one value. It allows existing of two RGB pixel descriptors:

$$\mathrm{P}_1 = \begin{bmatrix} r_1 & g_1 & b_1 \end{bmatrix}, P_2 = \begin{bmatrix} r_2 & g_2 & b_2 \end{bmatrix}, \tag{15.4}$$

which satisfy the following statement:

$$\mathrm{l}(P_1) = l(P_2) \wedge (r_1 \neq r_2 \vee g_1 \neq g_2 \vee b_1 \neq b_2) \tag{15.5}$$

In other words there may exist an example image of colorful text with dark red foreground and bright blue background and those colors may be of the same or very similar luminance. Such luminance values are ambiguous. Such example text image is easy to be read by human but is also impossible to be read by the grayscale OCR

system. Example RGB subchannel values of such example text image are shown on Figure 15.1. It is easy to see that the image contains many different colors with the same luminance. This is why the grayscale converter may interpret completely different colors as the same value. After grayscale conversion such image become a simple solid gray box.

Fig. 15.1 Example ambiguous image – RGB channel values and desaturated version are shown

Succeeding sections present an additional system which is able to convert colorful text image with various colors into known indexed model.

15.3 The Improvement

15.3.1 Colorful Image Segmentation Overview

The image presented on Figure 15.1 was analyzed in terms of pixel luminosity. The new approach adds a completely new layer to the pre-processing stage. The new layer analyses all three RGB color values using completely different techniques when comparing to the plain grayscale conversion algorithms.

15.3.2 New Color Model

RGB color model is one of the most important color models in the world. Computers, common compact digital cameras and mobile phone digital cameras use RGB as the main or even the only color model. Unfortunately, RGB is not the best tool when analyzing colors or colorful images. When using RGB color model we have three different values and none of them alone gives us the pure information about the color. Even simple exposure change affects all three RGB values. Because of that HSB color model was used. It allows for easier feature extraction. A very good example of using different color model for better feature extraction purposes is described in [1].

HSV color model [5] also uses three values for every pixel, but their meaning is different when comparing to RGB. First value (h) defines *hue*. Second value (s) defines *saturation*. Third value (v) defines *value*. In HSB model – which is HSV model synonym – that third value is named as *brightness* (b). Using HSB model allows to decompose information about color and brightness much more easily rather than using plain RGB model. The conversion between the RGB and HSV/HSB color model

is possible using special algorithm described below. HSV values are represented by equations (15.6) and (15.8).

$$\mathrm{h}' = \begin{cases} \text{undefined} & \text{if } c = 0 \\ \frac{g-b}{c} \bmod 6 & \text{if } m = r \\ \frac{b-r}{c} + 2 & \text{if } m = g \\ \frac{r-g}{c} + 4 & \text{if } m = b \end{cases}; h = 60^\circ \times h', \tag{15.6}$$

where:

$$\mathrm{m} = max(r,g,b);\ n = min(r,g,b);\ c = m - n \tag{15.7}$$

and

$$\mathrm{s} = \begin{cases} 0 & \text{if } c = 0 \\ \frac{c}{v} & \text{otherwise} \end{cases}; v = m \tag{15.8}$$

15.3.3 Approach for Intense Colorful Texts

When considering intense colorful text only hue component carries important information. It allows to decompose colorful text image using hue histogram. Important thing is that hue value wraps up near range edges. Saturation comes with another information, but will be considered as a less important value for colorful text image. Every color will be described using one value which is described by (15.9).

$$\mathrm{a}_c(h,s,v) = h \cdot \frac{256}{360^\circ} \tag{15.9}$$

where: h – hue, s – saturation, v – value.

15.3.4 Cyclic HSV Histogram

When comparing values of RGB vs. HSV, we need to consider that hue value is cyclic. In this case the color with hue value $h \to 360^\circ$ is very close to the one with hue value set to $h = 0^\circ$. When using RGB model minimum and maximum values represent opposite color properties. There is an easy way to make RGB values compatible with cyclic space by mirroring them or dividing by two.

15.3.5 Approach for Grayscale Texts

Parts of grayscale text are interpreted as any color with low *saturation* or *value*(*brightness*). Then luminosity gives most important data. Due to lack of determinism *hue* value may be omitted in such case. Because of that, such color may be described using value described in (15.10).

$$\mathrm{a}_g = \frac{v}{2} \tag{15.10}$$

where: v – value. Shown value is divided by 2 to make it compatible with cyclic histogram space.

15.3.6 Approach for Both Cases or for a Mixed Case

Joining two presented approaches is the key of making the system all-purpose.

Images may be categorized into three cases. First contains intense colorful text images. Second contains greyscale images. Third contains intermediate ones. Due to fuzzy nature of picture features classification using continuous range is helpful. Additional two threshold values are present. Upper boundary determine the case with intense color. The lower boundary determine the case with grayscale. Classification value ($b \in [0;9]$]) is computed using following expressions:

$$\mathrm{b}' = h \cdot s \; ; b = \begin{cases} 0 & , \text{if } b' \leq b_l \\ 9 & , \text{if } b' \geq b_h \\ 9 \cdot \frac{b' - \cdot b_l \cdot b_l}{b_u \cdot b_u - b_l \cdot b_l} & , \text{if } b' \in (b_l; b_h) \end{cases} \tag{15.11}$$

where:

- h – hue; s – saturation; b_l – lower boundary (colorful pixel);
- b_u – upper boundary (desaturated pixel).

Value of b represents degree of pixel desaturation. It is the second pixel descriptor value. The first pixel descriptor value a is chosen depending of b value using the following expression:

$$\mathrm{a} = \begin{cases} a_c & , \text{if } b = 0 \\ a_g & , \text{if } b = 9 \\ \frac{a_g \cdot b}{9} + \frac{a_c \cdot (9-b)}{9} & , \text{if } b \in (0;9) \end{cases} \tag{15.12}$$

15.3.7 New Pixel Descriptor

A new pixel descriptor may be expressed as:

$$\mathrm{D} = \left[a \; b \cdot y_t \right], \text{ where: } y_t \text{– aspect factor constant.} \tag{15.13}$$

Main features of such descriptor are:

- ability to describe feature (in case of colorful pixel) or brightness (in case of desaturated or dark pixel);
- ability to distinguish between grayscale and colorful pixels.

Such descriptor allows to create a new two-dimensional histogram which is useful for further statistical and processing purposes. Example histogram is shown on Figure 15.2.

Let define following values:

$$\mathrm{d}_c \text{ – sum of all histogram values for the colorful half;} \tag{15.14}$$

$$\mathrm{d}_g \text{ – sum of all histogram values for the desaturated half.} \tag{15.15}$$

The first half describes colorful pixels while the second half describes desaturated pixels.

Let's assume that the text image contains approximately $s_f = 11\%$ of foreground and $s_f = 89\%$ of background. Let's define a threshold value less than the lowest class $t = 5\% < s_f$. There may be five cases:

1. $d_c < t$ – text is desaturated;
2. $d_g < t$ – text is colorful;
3. $d_g \geq t \wedge d_c \geq t \wedge d_c > d_g$ – colorful background and desaturated foreground;
4. $d_g \geq t \wedge d_c \geq t \wedge d_c < d_g$ – colorful foreground and desaturated background;
5. $d_c = d_g$ – very unlikely result, ambiguous case.

Another useful factor is histogram value sharpness which may be described as a fraction of maximum histogram value divided by sum of all histogram values.

15.3.8 Clusterization Using Modified DBSCAN Algorithm

When considering proper clusterization methods size constrains must be accommodated. One of the best known clusterization algorithm is the k-means algorithm. Unfortunately, such algorithm tends to group data vectors into classes of similar size which is unacceptable. It is unacceptable behavior when clustering into classes of significantly different sizes like text image foreground and background.

Because of that variation of cardinalities density-based algorithm (called DBSCAN) was used [6]. The algorithm was customized to meet requirements of this special clusterization case. The main customization is that the algorithm tends to engage clusterization starting from points with the highest count value.

Moreover, during one recursive processing only points with count value higher than specified dynamic threshold may be joined. Such dynamic threshold depends on the count value of the base point and is expressed as follow: $j_{th} = d j_f$, where: j_{th} – threshold value, $j_f \in [0;1)$ – threshold factor, $d \in [0;1)$ – count value of base point. All points with lower value are omitted. Also points further than e are omitted.

After iterating through whole image all clusters are sorted by their count value. First cluster with the highest value is considered as a background cluster. Second cluster is considered as a foreground. All other clusters are considered as a noise.

Fig. 15.2 Histogram for example ambiguous image (darker – bigger value count)

Fig. 15.3 Clustered histogram for example ambiguous image (black – foreground, white – background, gray – not classified)

Example histogram after clusterization is shown on Figure 15.3.

15.3.9 Progressive Interpolation of Unambiguous Image Pixels

After clustering by histogram values the text image pixels are clustered into three classes: *background*, *foreground* and *unknown*. All pixels with the *unknown* status must be classified into other groups. It is done using progressive growing interpolation method. The algorithm is as follow:

1. Mark all *unknown* pixels as non-visited.
2. Mark all *background* pixels as visited and set their value to -1.0.
3. Mark all *foreground* pixels as visited and set their value to +1.0.
4. For every non-visited pixel which has at least one visited adjacent pixel (including hexagonal ones):
   - set pixel's value as the sum of all known or visited pixels divided by maximum number of adjacent pixels (including hexagonal ones, in this case it is divided by 8);
   - mark pixel as computed.
5. For every pixel marked as computed: set pixel as visited.
6. Go back to step 4 until all pixels are marked as visited.
7. Classify every pixel with value greater than zero as a *foreground* pixel.
8. Classify every pixel with value equal or lower than zero as a *background* pixel.

The main advantage of shown algorithm is that the value is computed for every pixel basing on near and far adjacent pixels. Further pixels causes less impact on final value when comparing to nearby pixels. On the other hand such change exists so the method is able to properly fill tiny and huge regions. Some example images are included in the next section.

15.3.10 Multicase Selector Using Soft Computing

The new algorithm is widely adaptive, but still contains some constants like thresholds and coefficients. Unfortunately optimal for desaturated onscreen text images may not be optimal for colorful text images retrieved from digital camera and vice-versa.

To overcome possible limitations a set of threshold values may be used. Each set is considered optimal for each representative case (e.g., intense colorful on-screen, colorful from digital sensors, etc.). Input text image is described by some statistical values such as mentioned in subsection 15.3.7. Some representative text images are needed to gather computed statistical values for each representative case (e.g., intense colorful on-screen image, colorful image retrieved from digital sensors or mixed case).

One of the most important part is a good quality classifier. ANN is one example, but Support Vector Machines (SVM) [3] technology may be also used. SVM finds optimal hyperplane for classification purposes so it is very good solution for extracting unambiguous knowledge using few data.

Table 15.1 Accuracy for certain pixels before and after interpolation

	T	F	U	TP	TN	FN	FP	A	E	SE	SP
Before interpol.	10063	92750	4311	10015	92599	151	48	99.8%	0.19%	99.52%	99.84%
After interpol.	11162	95962	0	10971	95683	279	191	99.56%	0.43%	98.29%	99.71%

15.4 Example Results

15.4.1 Prepared Image with Intense Computer Added Noise

This subsection describes processing of text image with computer added noise. The text image is shown on Figure 15.4. There are two histograms. The first (Figure 15.5) shows cardinality. The second (Figure 15.6) shows histogram clusterization results. Image segmentation using histogram clusterization results is shown on Figure 15.7. Ambiguous parts are processed using progressive interpolation and the result is shown on Figure 15.8.

Using prepared test image pairs allows to evaluate accuracy. Accuracy for certain pixels before interpolation is shown in Table 15.1 (second row). For the first output image uncertainty value was $u = \frac{U}{P+N} = \frac{4311}{102813} \approx 4.02\%$ where: T – answer „*true*", F – answer „*false*", U – answer „*unambiguous*", TP – true positive,TN – true negative,FN – false negative,FP – false positive,$A = (TP+TN)/(P+N)$ – accuracy, $E = 1-A$ – error, $SE = TP/P$ – sensitivity, $SP = TN/N$ – specificity [7, p. 12].

Summing up, error rate before interpolation is $e_1 = e_{U,1} + e_{E,1} \approx 4.02\% + 0.19\% = 4.21\%$ (for second row). Of course, after the interpolation uncertainty value $u = \frac{U}{P+N} = \frac{0}{102813} = 0\%$. Final error rate after interpolation is better: $e_2 = e_{U,2} + e_{E,2} \approx 0\% + 0.43\% = 0.43\%$ (for third row), so final accuracy value is at least $a_2 \geq 99.56\%$. Letters are reproduced very well before and after final interpolation.

15.4.2 Real Images with Real Noise

This subsection describes processing an image of text made using digital camera with flash enabled. It contains environmental noises. It is not prepared image, so no corresponding ideal template is present.

The text image is shown on Figure 15.9. Histograms before and after clusterization are shown on Figure 15.10 and on Figure 15.11 respectively. Output images before and after interpolation are shown on Figure 15.12 and on Figure 15.13 respectively. Before interpolation there are solid snippets of uncertainty. This is because there were used the same algorithm parameters as in the previous example. Despite of that, the final accuracy is more than satisfactional.

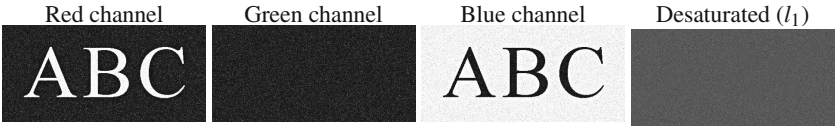


Fig. 15.4 Example ambiguous image – RGB channel values and desaturated version are shown

Fig. 15.5 Histogram for example ambiguous image (darker – bigger value count)

Fig. 15.6 Clustered histogram for example ambiguous image (black – foreground, white – background, gray – not classified)

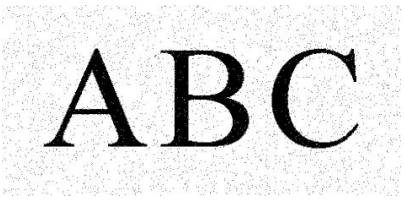


Fig. 15.7 Example image clustered (before clusters interpolation)

Fig. 15.8 Example image clustered and interpolated

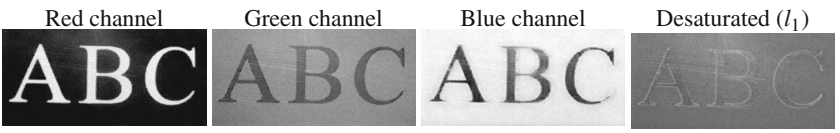


Fig. 15.9 Example ambiguous image – RGB channel values and desaturated version are shown

Fig. 15.10 Histogram for example ambiguous image (darker – bigger value count)

Fig. 15.11 Clustered histogram for example ambiguous image (black – foreground, white – background, gray – not classified)

Fig. 15.12 Example image clustered (before clusters interpolation)

Fig. 15.13 Example image clustered and interpolated

Table 15.2 Results of using intelligent multicase parameters selector

	Set 1	Set 2	Best possible	$ANN_{50\%}$	$ANN_{67\%}$	$ANN_{75\%}$
A	89,202%	95,931%	96,443%	96,254%	96,264%	96,273%

15.4.3 Optimized Approach for Set of Example Images

Clustering algorithm uses two threshold parameters. Unfortunately, optimal parameter values depend on image characteristics. ANN was used as a multicase parameter set selector and two optimal parameter sets were proposed. Following constraints apply. *Input text images* – set of 1592 example input text images; set covers all example variations of different foreground/background color set; two subclasses of text images: sharp text images and text images which were soften and altered with digital camera noise and computer generated noise; all cases uses background and foreground colors of the same luminosity. *Classifier input values* – relative values of two maximum histogram values and their locations, average histogram value & color configuration. *Parameter sets* – set 1 is $d_1 = \frac{1}{7}$, $e_1 = 3$, set 2 is $d_2 = \frac{1}{60}$, $e_2 = 20$. Results of achieved accuracy (A) after clustering phase are shown in Table 15.2. Columns with $ANN_{x\%}$ show accuracy when using ANN as a selector and the value $x\%$ shows training set to whole set ratio. Using intelligent multicase selector allowed to achieve 96,273% accuracy which was close to the global best possible accuracy. The ratio of training set to whole set was 3:4, but even 1:2 ratio gives very good results.

15.5 Summary

Presented system works very well for various text images and allows to successfully cluster inverted or colorful text images. Shown examples prove that it is able to process non-trivial text images with many different noise sources. Many layers of self-adaptation allow reliable result. The software may be used in as a standalone system or in conjunction with existing OCR systems.

Acknowledgements. Author would like to express his gratitude to Prof. Piotr Szczepaniak, Eng., M.Sc., Ph.D. for proposing the very interesting and inspirational field of expertise and for his significant additional help during researches. He would also thank to Jagoda Lazarek, Eng., M.Sc. for scientific consultation and to all people who were helpful in the achievement of the goal.

Publication of this chapter and participation in a corresponding conference (MISSI'14) was partially funded by Dean of Faculty Research Grant – "Grant dla młodych naukowców", Faculty of Technical Physics, Information Technology and Applied Mathematics, Lodz University of Technology, 2014.

References

1. Lazarek, J., Szczepaniak, P.S.: Detection of Semantically Significant Image Elements Using Neural Networks. In: Burduk, R., Kurzyński, M., Woźniak, M., Żołnierek, A. (eds.) Computer Recognition Systems 4. AISC, vol. 95, pp. 357–364. Springer, Heidelberg (2011)
2. Musiał, A.: Optical Character Recognition using Artificial Intelligence, Master's thesis, Lodz University of Technology, Lodz (2011)
3. Szczepaniak, P.S.: Obliczenia inteligentne, szybkie przekształcenia i klasyfikatory, Akademicka Oficyna Wydawnicza EXIT, Warsaw (2004)
4. Plataniotis, K.N., Venetsanopoulos, A.N.: Color image processing and applications, pp. 16–35. Springer, Heidelberg (2001)
5. Agoston, M.K.: Computer Graphics and Geometric Modelling, pp. 299–307. Springer, London (2005)
6. Ester, M., Kriegel, H., Sander, J., Xu, X.: A density-based algorithm for discovering clusters in large spatial databases with noise. In: Proceedings of the SICKDDM (KDD 1996). AAAI Press (1996)
7. Lazarek, J.: Metody analizy obrazu – analiza obrazu mammograficznego na podstawie cech wyznaczonych z tekstury, p. 12, IAPGOŚ 4/2013. CITT LPNT, Lublin (2013)

Chapter 16
Polish-English Statistical Machine Translation of Medical Texts

Krzysztof Wołk and Krzysztof Marasek

Abstract. This new research explores the effects of various training methods on a Polish to English Statistical Machine Translation system for medical texts. Various elements of the EMEA parallel text corpora from the OPUS project were used as the basis for training of phrase tables and language models and for development, tuning and testing of the translation system. The BLEU, NIST, METEOR, RIBES and TER metrics have been used to evaluate the effects of various system and data preparations on translation results. Our experiments included systems that used POS tagging, factored phrase models, hierarchical models, syntactic taggers, and many different alignment methods. We also conducted a deep analysis of Polish data as preparatory work for automatic data correction such as true casing and punctuation normalization phase.

16.1 Introduction

Obtaining and providing medical information in an understandable language is of vital importance to patients and physicians [1–3]. For example, as noted by GCH [4], a traveler in a foreign country may need to communicate their medical history and understand potential medical treatments. In addition, diverse continents (e.g., Europe) and countries (e.g., the U.S.) have many residents and immigrants who speak languages other than the official language of the place in which they require medical treatment.

Karliner [5] discusses how human translators could improve access to health care, as well as improve its quality. However, human translators training in

Krzysztof Wołk · Krzysztof Marasek
Department of Multimedia, Polish-Japanese Institute of Information Technology,
ul. Koszykowa 86, 02-008 Warszawa, Poland
e-mail: kwolk@pjwstk.edu.pl

A. Zgrzywa et al. (eds.), *New Research in Multimedia and Internet Systems*,
Advances in Intelligent Systems and Computing 314, DOI: 10.1007/978-3-319-10383-9_16

medical information are often not available to patients or medical professionals [6]. While existing machine translation capabilities are imperfect [6], machine translation has the promise of reducing the cost of medical translation, while increasing its availability [7] in the future. The growing technologies of mobile devices hold great promise as platforms for machine translation aids for medical information.

Medical professionals and researchers, as well as patients, have a need to access the wealth of medical information on the Internet [1, 8]. Access to this information has the potential to greatly improve health and well-being. Sharing medical information could accelerate medical research. English is the prevalent language used in the scientific community, including for medical information, as well as on the Internet, where a vast amount of medical information may be found.

Polish is one of the most complex West-Slavic languages, which represents a serious challenge to any SMT system. The grammar of the Polish language, with its complicated rules and elements, together with a large vocabulary (due to complex declension) are the main reasons for its complexity. Furthermore, Polish has 7 cases and 15 gender forms for nouns and adjectives, with additional dimensions for other word classes.

This greatly affects the data and data structure required for statistical models of translation. The lack of available and appropriate resources required for data input to SMT systems presents another problem. SMT systems should work best in specific, narrow text domains and will not perform well for a general usage. Good quality parallel data, especially in a required domain, has low availability. In general, Polish and English also differ in syntax. English is a positional language, which means that the syntactic order (the order of words in a sentence) plays a very important role, particularly due to limited inflection of words (e.g., lack of declension endings). Sometimes, the position of a word in a sentence is the only indicator of the sentence meaning. In the English sentence, the subject group comes before the predicate, so the sentence is ordered according to the Subject-Verb-Object (SVO) schema. In Polish, however, there is no specific word order imposed and the word order has no decisive influence on the meaning of the sentence. One can express the same thought in several ways, which is not possible in English. For example, the sentence "I bought myself a new car." can be written in Polish as "Kupiłem sobie nowy samochód", or "Nowy samochód sobie kupiłem.", or "Sobie kupiłem nowy samochód.", or "Samochód nowy sobie kupiłem." Differences in potential sentence word order make the translation process more complex, especially when working on a phrase-model with no additional lexical information.

As a result, progress in the development of SMT systems for Polish is substantially slower as compared to other languages. The aim of this work is to create an SMT system for translation from Polish to English, and vice versa, for medical data. This chapter is structured as follows: Section 16.2 explains the Polish data preparation. Section 16.3 presents the English language issues. Section 16.4 describes the translation evaluation methods. Section 16.5 discusses the results. Sections 16.6 and 16.7 summarize potential implications and future work.

16.2 Preparation of the Polish Data

The Polish data we used was a corpora created by the European Medical Agency (EMEA). Its size was about 80 MB, and it included 1,044,764 sentences built from 11.67M words that were not tokenized. The data was provided as pure text encoded with UTF-8. In addition, texts are separated into sentences (one per line) and aligned in language pairs.

Some discrepancies in the text parallelism could not be avoided. These are mainly repetitions of the Polish text not included in the English text, some additional remarks in the Polish data and poor translations made it appear that volunteers translated at least part of the data. It made the training material a bit noisy.

The size of the vocabulary is 148,170 unique Polish words forms and 109,326 unique English word forms. The disproportionate vocabulary sizes are also a challenge, especially in translation from English to Polish.

Before the use of a training translation model, preprocessing that included removal of long sentences (set to 80 tokens) had to be performed. The Moses toolkit scripts [9] were used for this purpose. Moses is an open-source toolkit for statistical machine translation, which supports linguistically motivated factors, confusion network decoding, and efficient data formats for translation models and language models. In addition to the SMT decoder, the toolkit also includes a wide variety of tools for training, tuning and applying the system to many translation tasks.

16.3 English Data Preparation

The preparation of the English data was definitively less complicated than that for Polish. We developed a tool to clean the English data by removing foreign words, strange symbols, etc. Compare to Polish, the English data contained significantly fewer errors. Nonetheless, some problems needed to be corrected. Most problematic were translations into languages other than English and strange UTF-8 symbols. We also found a few duplications and insertions inside a single segment.

16.4 Evaluation Methods

Metrics are necessary to measure the quality of translations produced by the SMT systems. For this purpose, various automated metrics are available to compare SMT translations to high quality human translations. Since each human translator produces a translation with different word choices and orders, the best metrics measure SMT output against multiple reference human translations. Among the commonly used SMT metrics are: Bilingual Evaluation Understudy (BLEU), the U.S. National Institute of Standards & Technology (NIST) metric, the Metric for Evaluation of Translation with Explicit Ordering (METEOR), Translation Error Rate (TER), and the Rank-based Intuitive Bilingual Evaluation Score (RIBES). These metrics will now be briefly discussed [10].

BLEU was one of the first metrics to demonstrate high correlation with reference human translations. The general approach for BLEU, as described in [11], is to attempt to match variable length phrases to reference translations. Weighted averages of the matches are then used to calculate the metric. The use of different weighting schemes leads to a family of BLEU metrics, such as the standard BLEU, Multi-BLEU, and BLEU-C [12].

The NIST metric seeks to improve the BLEU metric by valuing information content in several ways. It takes the arithmetic versus geometric mean of the *n*-gram matches to reward good translation of rare words. The NIST metric also gives heavier weights to rare words. Lastly, it reduces the brevity penalty when there is a smaller variation in translation length. This metric has demonstrated improvements over the baseline BLEU metric [13].

The METEOR metric, developed by the Language Technologies Institute of Carnegie Mellon University, is also intended to improve the BLEU metric. We used it without synonym and paraphrase matches for polish. METEOR rewards recall by modifying the BLEU brevity penalty, takes into account higher order *n*-grams to reward matches in word order, and uses arithmetic vice geometric averaging. For multiple reference translations, METEOR reports the best score for word-to-word matches. Banerjee and Lavie [14] describe this metric in detail.

TER is one of the most recent and intuitive SMT metrics developed. This metric determines the minimum number of human edits required for an SMT translation to match a reference translation in meaning and fluency. Required human edits might include inserting words, deleting words, substituting words, and changing the order or words or phrases [15].

The focus of the RIBES metric is word order. It uses rank correlation coefficients based on word order to compare SMT and reference translations. The primary rank correlation coefficients used are Spearman's ρ, which measures the distance of differences in rank, and Kendall's τ, which measures the direction of differences in rank [16].

16.5 Experimental Results

A number of experiments have been performed to evaluate various versions of our SMT systems. The experiments involved a number of steps. Processing of the corpora was accomplished, including tokenization, cleaning, factorization, conversion to lower case, splitting, and a final cleaning after splitting. Training data was processed, and a language model was developed. Tuning was performed for each experiment. Lastly, the experiments were conducted.

The baseline system testing was done using the Moses open source SMT toolkit with its Experiment Management System (EMS) [17]. The SRI Language Modeling Toolkit (SRILM) [18] with an interpolated version of the Kneser-Ney discounting (interpolate –unk –kndiscount) was used for 5-gram language model training. We used the MGIZA++ tool for word and phrase alignment. KenLM [19] was used to binarize the language model, with a lexical reordering set using the

msd-bidirectional-fe model. Reordering probabilities of phrases are conditioned on lexical values of a phrase. It considers three different orientation types on source and target phrases: monotone(M), swap(S), and discontinuous(D). The bi-directional reordering model adds probabilities of possible mutual positions of source counterparts to current and subsequent phrases. Probability distribution to a foreign phrase is determined by "f" and to the English phrase by "e" [20, 21]. MGIZA++ is a multi-threaded version of the well-known GIZA++ tool [22]. The symmetrization method was set to grow-diag-final-and for word alignment processing. First, two-way direction alignments obtained from GIZA++ were intersected, so only the alignment points that occurred in both alignments remained. In the second phase, additional alignment points existing in their union were added. The growing step adds potential alignment points of unaligned words and neighbors. Neighborhood can be set directly to left, right, top or bottom, as well as to diagonal (grow-diag). In the final step, alignment points between words from which at least one is unaligned are added (grow-diag-final). If the grow-diag-final-and method is used, an alignment point between two unaligned words appears [23].

We conducted many experiments to determine the best possible translation method from Polish to English, and vice versa. For experiments we used Moses SMT with Experiment Management System (EMS) [24].

The experiments were conducted with the use of the test and development data randomly selected and removed from the corpora. We generated 1000 segments for each purpose, for Polish-to-English and English-to-Polish translations. The experiments were measured by the BLEU, NIST, TER, RIBES and METEOR metrics. Note that a lower value of the TER metric is better, while the other metrics are better when their values are higher. Table 16.1 and 16.2 represent the results of experiments. Experiment 00 in the tables indicates the baseline system. Each of the following experiments is a separate modification to the baseline. Experiment 01 additionally uses truecasing and punctuation normalization.

Table 16.1 Polish-to-English translation

System	BLEU	NIST	METEOR	RIBES	TER
00	70.15	10.53	82.19	83.12	29.38
01	64.58	9.77	76.04	72.23	35.62
02	71.04	10.61	82.54	82.88	28.33
03	71.22	10.58	82.39	83.47	28.51
04	76.34	10.99	85.17	85.12	24.77
05	70.33	10.55	82.28	82.89	29.27
06	71.43	10.60	82.89	83.19	28.73
07	71.91	10.76	83.63	84.64	26.60
08	71.12	10.37	84.55	76.29	29.95
09	71.32	10.70	83.31	83.72	27.68
10	71.35	10.40	81.52	77.12	29.74
11	70.34	10.64	82.65	83.39	28.22
12	72.51	10.70	82.81	80.08	28.19

Table 16.2 English-to-Polish translation

System	BLEU	NIST	METEOR	RIBES	TER
00	69.18	10.14	79.21	82.92	30.39
01	61.15	9.19	71.91	71.39	39.45
02	69.41	10.14	78.98	82.44	30.90
03	68.45	10.06	78.63	82.70	31.62
04	73.32	10.48	81.72	84.59	27.05
05	69.21	10.15	79.26	82.24	30.88
06	69.27	10.16	79.30	82.99	31.27
07	68.43	10.07	78.95	83.26	33.05
08	67.61	9.87	77.82	77.77	29.95
09	68.98	10.11	78.90	82.38	31.13
10	68.67	10.02	78.55	79.10	31.92
11	69.01	10.14	79.13	82.93	30.84
12	67.47	9.89	77.65	75.19	33.32

Experiment 02 is enriched with Operation Sequence Model (OSM). The motivation for OSM is that it provides phrase-based SMT models the ability to memorize dependencies and lexical triggers, it can search for any possible reordering, and it has a robust search mechanism. Additionally, OSM takes source and target context into account, and it does not have the spurious phrasal segmentation problem. The OSM is values especially for the strong reordering mechanism. It couples translation and reordering, handles both short and long distance reordering, and does not require a hard reordering limit [25].

Experiment 03 is based on a factored model that allows additional annotation at the word level, which may be exploited in various models. Here we facilitate the part of speech tagged data on English language side as basis for the factored phrase model [26].

Hierarchical phrase-based translation combines the strengths of phrase-based and syntax-based translation. It uses phrases (segments or blocks of words) as units for translation and uses synchronous context-free grammars as rules (syntax-based translation). Hierarchical phrase models allow for rules with gaps. Since these are represented by non-terminals and such rules are best processed with a search algorithm that is similar to syntactic chart parsing, such models fall into the class of tree-based or grammar-based models. We used such a model in Experiment 04.

The Target Syntax model implies the use of linguistic annotation for non-terminals in hierarchical models. This requires running a syntactic parser. For this purpose, we used the "Collins" [27] statistical natural language parser in Experiment 05.

Experiment 06 was conducted using stemmed word alignment. The factored translation model training makes it very easy to set up word alignment based on word properties other than the surface forms of words. One relatively popular method is to use stemmed words for word alignment. There are two main reasons for

this. For morphologically rich languages, stemming overcomes data sparsity problem. Secondly, GIZA++ may have difficulties with very large vocabulary sizes, and stemming reduces the number of unique words.

Experiment 07 uses Dyer's Fast Align [28], which is another alternative to GIZA++. It runs much faster, and often gives better results, especially for language pairs that do not require large-scale reordering.

In Experiment 08 we used settings recommended by Koehn in his Statistical Machine Translation system from WMT'13 [29].

In Experiment 09 we changed the language model discounting to Witten-Bell. This discounting method considers diversity of predicted words. It was developed for text compression and can be considered to be an instance of Jelinek-Mercer smoothing. The n-th order smoothed model is defined recursively as a linear interpolation between the n-th order maximum likelihood model and the (n-1)th order smooth model [30].

Lexical reordering was set to hier-mslr-bidirectional-fe in Experiment 10. It is a hierarchical reordering method that considers different orientations: monotone, swap, discontinuous-left, and discontinuous-right. The reordering is modeled bidirectionally, based on the previous or next phrase, conditioned on both the source and target languages.

Compounding is a method of word formation consisting of a combination of two (or more) autonomous lexical elements that form a unit of meaning. This phenomenon is common in German, Dutch, Greek, Swedish, Danish, Finnish, and many other languages. For example, the word "flowerpot" is a closed or open compound in English texts. This results in a lot of unknown words in any text, so splitting up these compounds is a common method when translating from such languages. Moses offers a support tool that splits up words if the geometric average of the frequency of its parts is higher than the frequency of a word. In Experiment 11 we used the compound splitting feature. Lastly, for Experiment 12 we used the same settings for out of domain corpora we used in IWSLT'13 [31].

16.6 Discussion and Conclusions

Several conclusions can be drawn from the experimental results presented here. It was surprising that truecasing and punctuation normalization decreased the scores by a significant factor. We suppose that the text was already properly cased and punctuated. In Experiment 02 we observed that, quite strangely, OSM decreased some metrics results. It usually increases the translation quality. However, in the PL->EN experiment the BLEU score increased just slightly, but RIBES metrics decreased at the same time. The similar results can be seen in the EN->PL experiments. Here, the BLEU score increased, but other metrics decreased.

Most of the other experiments worked as anticipated. Almost all of them raised the score a little bit or were at least confirmed witch each metric it the same manner. Unfortunately, Experiment 12, which was based on settings that provided best system score on IWSLT 2013 evaluation campaign, did not improve quality on

this data as much as it did previously. The most likely reason is that the data used in IWSLT did not come from any specific text domain, while here we dealt with a very narrow domain. It may also mean that training and tuning parameter adjustment may be required separately for each text domain if improvements cannot be simply replicated.

On the other hand, improvements obtained by training the hierarchical based model were surprising. In comparison to other experiments, Experiment 04 increased the BLEU score by the highest factor. The same, significant improvements can be observed in both the PL->EN and EN->PL translations, which most likely provide a very good starting point for future experiments.

Translation from EN to PL is more difficult, what is shown in the experiment, results are simply worse. Most likely the reasons for this, of course, are the complicated Polish grammar as well as the larger Polish vocabulary.

One of the solutions to this problem (according to work of Bojar [32]) is to use stems instead of surface forms that will reduce the Polish vocabulary size. Such a solution also requires a creation of an SMT system from Polish stems to plain Polish. Subsequently, morphosyntactic tagging, using the Wroclaw Natural Language Processing (NLP) tools (or other similar tool) [33], can be used as an additional information source for the SMT system preparation. It can be also used as a first step for creating a factored translation system that would incorporate more linguistic information than simply POS tags [4].

The analysis of our experiments led us to conclude that the results of the translations, in which the BLEU measure is greater than 70, can be considered only satisfactory within the selected text domain. The high evaluation scores indicate that the translations should be understandable by a human and good enough to help him in his work, but not good enough for professional usage or for implementation in translation systems e.g., for patients in hospitals abroad. We strongly believe that improving the BLEU score to a threshold over 80 or even 85 would produce systems that could be used in practical applications, when it comes to PL-EN translations. It may be particularly helpful with the Polish language, which is one of the most complex in terms of its structure, grammar and spelling. Additionally it must be emphasized that the experiments were conducted on texts obtained from the PDF documents shared by the European Medicines Agency. That is the reason why the data is more complex and has more sophisticated vocabulary that casual human speech. Speech usually is less complicated and easier to process by SMT systems. It is here, where we see other opportunity to increase output translation quality.

16.7 Future Work

Applying machine translation to medical texts holds great promise to be of benefit to patients, including travelers and those who do not speak the language of the country in which they need medical help. Improving access to the vast array of medical information on the Internet would be useful to patients, medical professionals, and medical researchers.

Human interpreters with medical training are too rare. Machine translation could also help in the communication of medical history, diagnoses, proposed medical treatments, general health information, and the results of medical research. Mobile devices and web applications may be able to boost delivery of machine translation services for medical information.

Several potential opportunities for future work are of interest, to extend our research in this critical area. Additional experiments using extended language models are warranted to improve the SMT scores. Much of the literature [22] confirms that interpolation of out of domain language models and adaptation of other bilingual corpora would improve translation quality. We intend to use linear interpolation as well as Modified Moore Levis Filtering for these purposes. We are also interested in developing some web crawlers in order to obtain additional data that would most likely prove useful. Good quality parallel data, especially in a required domain, has low availability.

In English sentences, the subject group comes before the predicate, so the sentence is ordered according to the Subject-Verb-Object (SVO) schema. Changing the Polish data to be in SVO order is of interest for the future as well.

Acknowledgements. This work is supported by the European Community from the European Social Fund within the Interkadra project UDA-POKL-04.01.01-00-014/10-00 and Eu-Bridge 7th FR EU project (Grant Agreement No. 287658).

References

1. Goeuriot, L., Jones, G., Kelly, L., Kriewel, S., Pecina, P.: Report on and prototype of the translation support. Khresmoi Public Deliverable 3 (2012)
2. Pletneva, N., Vargas, A., Boyer, C.: Requirements for the general public health search. Khresmoi Public Deliverable, D, 8 (2011)
3. Gschwandtner, M., Kritz, M., Boyer, C.: Requirements of the health professional search. Khresmoi Project Public Deliverable, D8. 1.2 (2011)
4. GCH Benefits, Medical Phrases and Terms Translation Demo, n.d. (accessed February 28, 2014)
5. Karliner, L.S., Jacobs, E.A., Chen, A.H., Mutha, S.: Do professional interpreters improve clinical care for patients with limited English proficiency? A systematic review of the literature. Health Services Research 42(2), 727–754 (2007)
6. Randhawa, G., Ferreyra, M., Ahmed, R., Ezzat, O., Pottie, K.: Using machine translation in clinical practice. Canadian Family Physician 59(4), 382–383 (2013)
7. Deschenes, S.: 5 benefits of healthcare translation technology. Healthcare Finance News (October 16, 2012)
8. Zadon, C.: Man vs machine: the benefits of medical translation services. Ezine Articles: Healthcare Systems (2013)
9. Koehn, P., Hoang, H., Birch, A., Callison-Burch, C., Federico, M., Bertoldi, N., Cowan, B., Shen, W., Moran, C., Zen, S.R., Dyer, C., Bojar, R., Constantin, A., Herbst, E.: Moses: open source toolkit for statistical machine translation. In: Proceedings of the ACL 2007 Demo and Poster Sessions, pp. 177–180 (2007)

10. Radziszewski, A.: A tiered CRF tagger for Polish. In: Bembenik, R., Skonieczny, Ł., Rybiński, H., Kryszkiewicz, M., Niezgódka, M. (eds.) Intell. Tools for Building a Scientific Information. SCI, vol. 467, pp. 215–230. Springer, Heidelberg (2013)
11. Axelrod, A.E.: Factored Language Models for Statistical Machine Translation, University of Edinburgh, Master of Science Thesis (2006)
12. Koehn, P.: What is a better translation? Reflections on six years of running evaluation campaigns. Auditorium du CNRS, Paris (2011)
13. Papineni, K., Rouskos, S., Ward, T., Zhu, W.J.: BLEU: a method for automatic evaluation of machine translation. In: Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics, Philadelphia, pp. 311–318 (2002)
14. Banerjee, S., Lavie, A.: METEOR: An automatic metric for MT evaluation with improved correlation with human judgments. In: Proceedings of the ACL Workshop on Intrinsic and Extrinsic Evaluation Measures for Machine Translation and/or Summarization, Ann Arbor, pp. 65–72 (2005)
15. Doddington, G.: Automatic evaluation of machine translation quality using n-gram co-occurrence statistics. In: Proceedings of the Second International Conference on Human Language Technology (HLT) Research, San Diego, pp. 138–145 (2002)
16. Isozaki, H., Hirao, T., Duh, K., Sudoh, K., Tsukada, H.: Automatic evaluation of translation quality for distant language pairs. In: Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing, pp. 944–952 (2010)
17. Snover, M., Dorr, B., Schwartz, R., Micciulla, L., Makhoul, J.: A study of translation edit rate with targeted human annotation. In: Proceedings of the 7th Conference of the Association for Machine Translation in the Americas, Cambridge (2006)
18. Koehn, P., et al.: Moses: open source toolkit for statistical machine translation. In: Annual Meeting of the Association for Computational Linguistics (ACL) Demonstration Session, Prague (June 2007)
19. Koehn, P., Axelrod, A., Birch, A., Callison-Burch, C., Osborne, M., Talbot, D., White, M.: Edinburgh system description for the 2005 IWSLT speech translation evaluation. In: IWSLT, pp. 68–75 (2005), http://www.cs.jhu.edu/~ccb/publications/iwslt05-report.pdf
20. Heafield, K.: KenLM: faster and smaller language model queries. In: Proceedings of the Sixth Workshop on Statistical Machine Translation, Association for Computational Linguistics, pp. 187–197 (2011)
21. Ruiz Costa-Jussà, M., Rodríguez Fonollosa, J.A.: Using linear interpolation and weighted reordering hypotheses in the Moses system, Barcelona, Spain (2010)
22. Stolcke, A.: SRILM – an extensible language modeling toolkit. In: INTERSPEECH (2002)
23. Gao, Q., Vogel, S.: Parallel implementations of word alignment tool. In: Software Engineering, Testing, and Quality Assurance for Natural Language Processing, pp. 49–57 (2008)
24. Moses Factored Training Tutorial, http://www.statmt.org/moses/?n=FactoredTraining.EMS
25. Durrani, N., Schmid, H., Fraser, A., Sajjad, H., Farkas, R.: Munich-Edinburgh-Stuttgart Submissions of OSM Systems at WMT13. In: ACL 2013 Eight Workshop on Statistical Machine Translation, Sofia, Bulgaria (2013)
26. Koehn, P., Hoang, H.: Factored translation models. In: Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning, Prague, pp. 868–876 (2007)

27. Bikel, D.: Intricacies of Collins' parsing model. Computational Linguistics 30(4), 479–511 (2004)
28. Dyer, C., Chahuneau, V., Smith, N.: A simple, fast and effective reparametrization of IMB Model 2. In: Proceedings of NAACL (2013)
29. Bojar, O.: Rich morphology and what can we expect from hybrid approaches to MT. Invited talk at International Workshop on Using Linguistic Information for Hybrid Machine Translation (LIHMT 2011) (2011), http://ufal.mff.cuni.cz/~bojar/publications/2011-FILEbojar_lihmt_2011_pres-PRESENTED.pdf
30. Hasan, A., Islam, S., Rahman, M.: A comparative study of Witten Bell and Kneser-Ney smoothing methods for statistical machine translation. Journal of Information Technology 1, 1–6 (2012)
31. Wolk, K., Marasek, K.: Polish-English speech statistical machine translation systems for the IWSLT 2013. In: Proceedings of the 10th International Workshop on Spoken Language Translation, Heidelberg, Germany (2013)
32. Bojar, O., Buck, C., Callison-Burch, C., Federmann, C., Haddow, B., Koehn, P., Monz, C., Post, M., Soricut, R., Specia, L.: Findings of the 2013 Workshop on Statistical Machine Translation. In: Proceedings of the Eight Workshop on Statistical Machine Translation. Association for Computational Linguistics, Sofia (2013)
33. Radziszewski, A., Śniatowski, T.: Maca – a configurable tool to integrate Polish morphological data. In: Proceedings of the Second International Workshop on Free/OpenSource Rule-Based Machine Translation, FreeRBMT 2011, Barcelona (2011)

Chapter 17
Descriptive and Predictive Analyses of Data Representing Aviation Accidents

František Babič, Alexandra Lukáčová, and Ján Paralič

Abstract. The aim of this chapter is to evaluate a potential of suitable data mining methods for analyses of aviation historical data. In our case, we used public aviation dataset from Federal Aviation Administration (FAA) Accident/Incident Data System containing information about civil aviation accidents or incidents within United States of America. This dataset represents interesting source of data for analytical purposes, e.g. it is possible to evaluate an influence of various factors on both types of events and based on identified hidden relations to generate a prediction model for specified target attribute. We compared our approach with some other existing works in this domain and obtained results are plausible and inspiring. Generated models could be used as a basis for aviation warning system or as a supporting method for different processes related to the aviation industry.

17.1 Introduction

The main motivation behind performed analytical experiments was to evaluate suitability of various data mining methods for effective processing of collected aviation data about accidents and incidents. We defined several particular analytical tasks in order to provide various views of available data. At first, most general level, we extracted some interesting statistics about different relations included in the dataset, e.g. which days of week are characterized by higher number of accidents? Or which construction details of the aircraft most often cause an accident in the air traffic? At the second level, we focused on identification of most valuable attributes for two target categories (i.e. accidents with serious consequences vs. accidents with light consequences). At this level we performed traditional data

František Babič · Alexandra Lukáčová · Ján Paralič
Department of Cybernetics and Artificial Intelligence,
Technical University of Kosice, Slovakia
e-mail: {frantisek.babic,alexandra.lukacova,jan.paralic}@tuke.sk

A. Zgrzywa et al. (eds.), *New Research in Multimedia and Internet Systems*,
Advances in Intelligent Systems and Computing 314, DOI: 10.1007/978-3-319-10383-9_17

mining experiments oriented to classification using suitable methods from machine learning theory. Last level dealt with generation of association rules in order to identify important combinations of reasons for negative events in air traffic. Summary, these approaches can provide important knowledge for different application purposes, e.g. part of aviation warning system in order to proactively alert the pilots, aircraft dispatcher or plane operators about potential risks. Also, it is possible to use this information for more effective management of insurance policies in the aviation industry.

Difference between aviation incident and accident is specified in Convention on International Civil Aviation Annex 13[1]. Accident is defined as an occurrence associated with the operation of an aircraft, which takes place between the time any person boards the aircraft with the intention of flight until such time as all such persons have disembarked, where a person is fatally or seriously injured, the aircraft sustains damage or structural failure or the aircraft is missing or is completely inaccessible. Incident is defined as an occurrence, other than an accident, associated with the operation of an aircraft that affects or could affect the safety of operations[2].

The content of this chapter consists of three main sections and starts with introduction, which briefly presents motivation and describes related works. The second chapter deals with all performed preprocessing operations, applied algorithms, generated models and obtained results. The last one summarizes the whole paper and outlines possible directions for the future work.

17.1.1 Related Work

Data mining methods have been applied to various datasets relevant to aviation with the aim to extract useful information for different purposes. We focused on the works devoted to processing and analysis of data from air traffic like safety reports, accidents or incidents reports. One part of research work in this domain is oriented to detection of human errors within human factor analysis and classification system (HFACS). HFACS is a theoretically based framework [19] for investigating and analyzing human error associated with aviation accidents and incidents. This framework was designed to investigate only aircrew related accidents, not events caused by maintenance error, flight problems or weather conditions.

Analysis of data from database of Turkey airline is described in [7]. This data sample contained records between years 1970 – 2011. Data mining tool Weka was used to create classification models to predict if the result of the incident was safety or not. The best model was based on decision trees with achieved accuracy 87.39%. Authors extracted also decision rules with the aim to identify key attributes for initiation of the incidents warning level.

[1] http://www.iprr.org/manuals/Annex13.html
[2] http://www.airsafe.com/events/define.htm

Another study dealt with improvement of classification models precision through initial reduction of original set of attributes by means of rough sets theory [8]. Authors of this study started experiments with initial set of records representing incidents from FAA Accident/Incident Data system between 2000 and 2006 years. These records were described by original set of 25 attributes. In the next step, some reduction methods (tootle statistic view, link analysis, found dependencies and histograms) were applied to these attributes and resulted into a set of 8 selected attributes in strong relation to incidents with fatality. Second reduction was applied by rough sets within Rosseta toolkit[3]. The reduction produced by rough sets was used as input for decision trees algorithm provided by PolyAnalyst data mining tool[4]. Best model offered 99.4% accuracy and contained 8 selected attributes as aircraft damage, type of operation, number of engines, etc. Next possibility to solve this task includes Formal Concept Analyses (FCA) with its practical implementation through generalized one-sided concept lattices [5] that allow exploitation of FCA method on data with different types of attributes [6].

Combination of Support Vector Machine and Simulated Annealing algorithm was used in [15] to analyze text incident reports. Obtained results showed that 89% were classified correctly.

Modification of the Apriori and C4.5 algorithms with existing domain knowledge in order to reduce the search space and to generate more interesting rules is described in [14]. In the first case, internal data structure was extended within domain knowledge and the second one used modified attribute selection method: lower scoring attributes can be chosen if the domain knowledge indicates such a preference. Both extensions resulted into reduced number of uninteresting rules and improved stability and decreased running time, e.g. comparison between traditional C4.5 and improved C4.5: number of decision rules from 41.6 to 4.6 (avg); the error rate was quite the same, but much shorter was the average running time (114.8 sec vs. 1.4 sec).

All presented studies confirmed a big potential of using data mining in aviation to extract important causes and reasons of occurred events; to identify hidden relations and dependencies or to simplify the work of persons responsible for flights operations. These facts motivated our research activities with publicly available database of civil aviation accidents. We had some previous experiences with experimental analysis of data sample containing records representing aviation incidents [12]. This sample is different from a data processed in this paper (accidents), mainly in quality characterized by a higher number of missing values in actual data sample. Our activities described in this paper were devoted to: suitable preprocessing methods to reduce the missing values, selection of key attributes for accidents classification, generation of classification models based on selected machine learning algorithms and finally extraction of interesting rules representing combination of input attributes causing relevant level of accidents.

[3] http://www.lcb.uu.se/tools/rosetta/

[4] http://www.megaputer.com/site/polyanalyst.php

17.1.2 Methods

Data mining is a multidisciplinary field, using work from areas including database technology, artificial intelligence, machine learning, neural networks, statistics, pattern recognition, knowledge based systems, knowledge acquisition, information retrieval, high performance computing and data visualization [10]. The aim is to extract potentially useful knowledge represented in form of various types of patterns from large set of data. In our case we used following methods to accomplish specified goals.

Decision tree is a flow-chart-like tree structure, where each internal node denotes a test on an attribute, each branch represents an outcome of the test, and leaf nodes represent target classes or class distributions [13]. Decision trees are popular classification method for their simple understandability and interpretability. The most popular algorithms are ID3 [16], C4.5 [17] and CART [4]. Random forest is an ensemble classifier which is using multiple decision tree models. The generalization error of a forest depends on the strength of the individual trees in the forest and the correlation between them [3]. Generated models will be evaluated within F-measure [18] that use for calculation both the precision (number of correct positive results divided by the number of all returned positive results) and the sensitivity (number of correct positive results divided by the number of results that should have been returned) related to these models.

Association rules are a popular and well researched method for discovering interesting relations between variables in large databases [1, 9]. The most used algorithm to mine association rules is Apriori [2].

17.2 Analytical Levels

In introduction, we outlined three analytical levels with the aim to give the answers to different questions arising from the data and possible application cases. In this chapter we will describe used dataset, performed operations, algorithms and methods to meet the expectations.

Data used in experiments contained accidents from FAA Accident/Incident Data System between years 2000 – 2013. We used this time period, because progress in aviation industry brings new design practices and technologies, new safety approaches and procedures every year. This initial dataset contained more than 25 thousand records described by 23 attributes. At first we eliminated attributes with little or no relevance to the target task, such as *Event ID*, *Accident number*, *Registration number* and *Report status*. Remaining attributes are described in Table 17.1.

First analytical level was oriented to exploratory data analysis with the aim to extract some basic interesting knowledge from available dataset. At first we provided some cleaning operation to eliminate records with high level of missing values. Also, we selected accident records only for location = USA, because they represented more than 92% of all available examples. Afterward we derived two

new attributes from variable *Event Date*; *Number of Day*, as we were interested if there are differences between Monday's and Friday's accidents; and *Season*, which had similar reason for creation. We created a new target attribute (fatal and non-fatal) as transformation and aggregation of four previous ones (Total fatal injuries, serious injuries, minor injuries and uninjured).

Also, we continued with evaluation of relation between these two new attributes (Number of Day, Season) and target attribute's values (Fatal or NonFatal) through the Pearson's chi-squared test of independence. This test resulted into chi-squared probability less than 0.05 which rejected the null hypothesis (attributes are independent) and confirmed the association in both cases.

These two examples show the possibilities how to visualize interesting relations extracted from data, e.g. which aircraft model is the most common object of occurred accidents; phase of flight with the highest probability of accident; aircraft company with the highest occurrence of various accidents; relations between level of aircraft damage and level of accident consequence and many others. Requirements for such type of information can motivate and start a deployment of suitable Business intelligence (BI) solution, see Fig. 17.1. But, important conditions for successful implementation of such a project are cooperation of all relevant data providers, data quality and low number of missing values. Meet these assumptions requires good discipline in collecting data on aviation accidents and incidents, whether by manual or automated means.

Table 17.1 Attributes description for data understanding

Name	Description	Missing values (%)	Unique values
Event date	Date of accident mm/dd/yyyy	0.00	between 01/01/2000 and 03/30/2014
Location	The closest town to the place where the accident occurred	0.10	9 889 locations
Country	Country where the accident occurred	0.15	146 countries
Aircraft damage	Severity of aircraft damage	1.83	3 (destroyed, minor, substantial)
Aircraft category	Aircraft category	62.59	8 (e.g. airplane, balloon, helicopter, ultralight, etc.)
Make	Manufacturer of the aircraft	0.20	3 690
Model	Aircraft model	0.24	5 357 different models of aircrafts
Amateur	Amateur-built aircraft	1.41	2 (yes, no)
Number of engines	Number of aircraft engines	6.84	7 (0 - 18)
Engine type	Type of aircraft engine	8.82	14 (e.g. turbo jet, turbo shaft, reciprocating)
FAR description	FAA flight code at the time of the accident	61.49	13 (e.g. Non-U.S. Commercial, Ultralight, Air Carrier, Air taxi, etc.)

Table 17.1 (*continued*)

Schedule	Scheduled or non-scheduled flight	91.18	3 (scheduled, non-scheduled, unknown)
Purpose of flight	Purpose of the flight	8.09	22 (e.g. personal, instructional, business, air show, etc.)
Air carrier	Name of the company to which the aircraft belongs to	96.72	695 companies
Total fatal injuries	The number of fatally injured persons	70.08	79 (min. 0 – max. 228)
Total serious injuries	The number of seriously injured persons	76.99	30 (0 - 265)
Total minor injuries	The number of minor injured persons	73.57	37 (0 - 67)
Total uninjured	The number of uninjured persons	37.49	230 (0 - 576)
Weather		4.07	3 (VMC – visual meteo conditions, IMC – instrument meteo conditions, unknown)
Broad phase of flight		17.68	12 (approach, climb, cruise, landing, etc.)

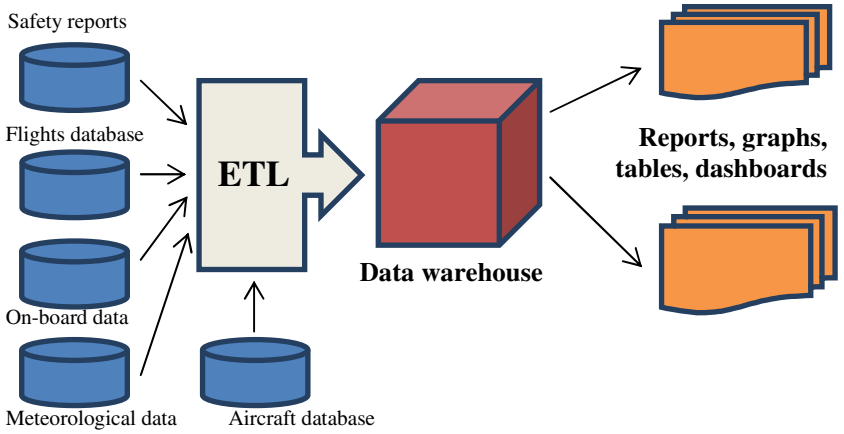


Fig. 17.1 Rough sketch of possible BI solution architecture for processing of various aviation data

Architecture presented on Fig. 17.1 can be understood only as proposal that emerged from our experiences with performed experiments. When we started with data understanding in combination with specified mining goals, we found several gaps, e.g. not only missing values, but missing crucial information about accident that was not included in FAA dataset. But, we tried to find a way how to start with this type of experiments based on our previous experiences with different data mining tasks.

Second level was devoted to necessary preprocessing task in order to ensure sufficient quality dataset for selected classification algorithms. We selected decision trees (C5.0) as most suitable method for this purpose, mainly due to its simple interpretability of generated models. Also, we performed experiments with a method called Random Forests too. Quality of generated classification models was evaluated by means of traditional metrics as accuracy, sensitivity, F measure, specificity; geometric mean [11] and such a joined metrics that take into account both sensitivity and specificity. Obtained results are presented below.

Initial experiments with selected classification algorithms showed the need for further preprocessing mainly oriented to reduce the still high level of missing values. We applied several filters and sorting techniques with the aim to improve the quality of processed dataset, e.g. we solved the typos in the values of two attributes: *Manufacturer of the aircraft* and *Air carrier* by manual identification and correction. Also, we used algorithm k-nearest neighbor to fill in remaining missing values. Next, we used a logistic regression to identify significant variables for specified target attribute and based on it we excluded only attribute E*ngine type* from further modeling. Finally, we addressed unbalanced character of the target attribute. It has a ratio 1:4.37 between minority and majority examples of target attribute, which means that if were classified all examples as negative; the accuracy would be 81.3%. This status motivated our next operations to deal with unbalanced dataset. As it is known, the random sampling in such a case can cause overfitting, we tried also Synthetic Minority Over-sampling Technique (SMOTE), but as you can see in Tab. 3 below, it did not produce significant improvements against the default classifier. Tab. 2 summarizes all performed experiments with three types of algorithms (C5.0, CART, Random Forest) and their different settings of relevant tuning parameters.

Table 17.2 Summary of the experiments

	Algorithm	Solution for unbalanced dataset	Settings of tuning parameters
1	C5.0	random over-sampling	pruning severity 75, minimum records per child branch 20, cross-validate with 10 folds in training set
2	C5.0	random over-sampling	pruning severity 75, minimum records per child branch 20, cross-validate with 10 folds in training set, boosting with 10 trials
3	CART	random over-sampling	maximum tree depth 5
4	Random Forest	random over-sampling	node size 20, number of trees 200
5	C5.0	SMOTE	pruning severity 75, minimum records per child branch 20, cross-validate with 10 folds in training set

Tab. 3 presents a summary of our experiments. It was important to decide which type of incorrect classification is the worse one. Based on the potential cost of wrong prediction we wanted mainly to minimize the false negative cases (fatal accidents classified as non-fatal). Final evaluation was therefore based on two selected metrics: sensitivity and geometric mean which compares the classification performance between the majority and minority class.

Table 17.3 The summary of the results of the experiments

	Accuracy (%)	Sensitivity (%)	Specificity (%)	Geometric mean (%)	F measure (%)
1	82.98	**76.97**	84.37	80.59	62.85
2	84.24	75.30	86.30	**80.61**	64.12
3	84.24	71.74	87.12	79.06	63.00
4	**84.76**	73.18	**87.41**	79.98	**64.23**
5	82.36	73.71	84.35	78.85	60.98

Third level's experiments were oriented on generation of association rules with the aim to identify interesting combination of reasons for different types of accidents. Association rules provide similar format of knowledge as decision rules extracted from classification models. Unfortunately, this direction did not provide any significant rules because of too specific records in used dataset. Available accidents and related values of attributes are very different, which did not allow the Apriori algorithm to mine any frequent item sets and relevant association rules. We will continue our work in this direction with enlargement of input data sample or with evaluation of similar characteristics between records. As output rules for flight operators we used the decision rules extracted from generated classification models, see Table 17.4.

Table 17.4 Decision rules extracted from classification models

Segment	Result	Cover	Frequency	Probability (%)
Make="B" and Broad Phase of Flight ="Maneuvering"	Fatal	106	85	80,19
Weather Condition="IMC" and Purpose of Flight = „Personal" and Broad Phase of Flight ="Maneuvering"	Fatal	118	95	80,51
Weather Condition="IMC" and Purpose of Flight = „Personal" and Broad Phase of Flight ="Cruise"	Fatal	263	205	77,95
Broad Phase of Flight ="Maneuvering" and Number of Engines >1 and Aircraft Category = „Airplane"	Fatal	99	74	74,75

17.3 Conclusions

In general, use of data mining in aviation represents an interesting way how to optimize the air traffic, to prevent a possible dangerous situation with negative consequences or to improve the customer services based on collected historical data. In our case we were focused on accidents during air traffic with the aim to evaluate the reasons of their occurrence with possibility to generate an easily understandable and cost effective prediction model. Our experiments resulted into several models that the best one had 84.24% accuracy and 80.61% geometric mean. In comparison with studies presented in related work, our experiments were devoted to the different dataset (free available) that was characterized by high number of missing values. In comparison with mentioned accuracy of baseline model we have achieved a small progress, but our best model was trained with the aim to classify the minority class as accurately, i.e. to prevent occurrence of accidents with fatality.

Obtained results are plausible, but also show the need for improvement of input data and additional preprocessing operations to increase the data quality. On the other hand, extension of this dataset with meteorological data relevant to the dates of accidents or with more detailed flight data, can contribute to the quality and exploitation level of generated models. This approach requires an effective integration of data from different sources as is mentioned on Fig. 17.1. Our future work will be devoted to the evaluation of its reality in our conditions.

Acknowledgment. This work was supported by the Slovak VEGA grant no. 1/1147/12 (30%) and the Slovak Research and Development Agency under the contract APVV-0208-10 (35%). It is also the result of the Project implementation: University Science Park TECHNICOM for Innovation Applications Supported by Knowledge Technology, ITMS: 26220220182, supported by the Research & Development Operational Programme funded by the ERDF (35%).

References

1. Agrawal, R.T., Imilienski, T., Swami, A.: Mining association rules between sets of items in large databases. In: Proceedings of the ACM SIGMOD International Conference on Management of Data, pp. 207–216 (1993)
2. Agrawal, R., Srikant, R.: Fast algorithms for mining association rules in large databases. In: Proceedings of the 20th International Conference on Very Large Data Bases (VLDB), Santiago, Chile, pp. 487–499 (1994)
3. Breiman, L.: Random forests. Machine Learning 45, 5–32 (2001)
4. Breiman, L., Friedman, J.H., Olshen, R.A., Stone, C.J.: Classification and regression trees. Wadsworth & Brooks/Cole Advanced Books & Software, Monterey (1984)
5. Butka, P., Pócsová, J., Pócs, J.: Design and implementation of incremental algorithm for creation of generalized one-sided concept lattices. In: 12th IEEE International Symposium on Computational Intelligence and Informatics, CINTI 2011, Budapest, Hungary, pp. 373–378 (2011)

6. Butka, P., Pócs, J., Pócsová, J.: Use of concept lattices for data tables with different types of attributes. Journal of Information and Organizational Sciences 36(1), 1–12 (2012)
7. Christopher, A.B.A., Appavu, S.: Data mining approaches for aircraft accidents prediction: An empirical study on Turkey Airline. In: 2013 IEEE International Conference on Emerging Trends in Computing, Communication and Nanotechnology, India, pp. 739–745. IEEE (2013)
8. Gürbüz, F., Özbakir, L., Yapici, H.: Classification rule discovery for the aviation incidents resulted in fatality. Knowledge-Based System 22(8), 622–632 (2009)
9. Hipp, J., Untzer, U.G., Nakhaeizadeh, G.: Algorithms for association rule mining – a general survey and comparison. ACM SIGKDD Explorations Newsletter 2(1), 58–64 (2000)
10. Jiawei, H., Kamber, M.: Data Mining: Concepts and Techniques, University of Simon Fraser (2001)
11. Kubat, M., Matwin, S.: Addressing the curse of imbalanced training sets: One-sided selection. In: Proceedings of the Fourteenth International Conference on Machine Learning, pp. 179–186 (1997)
12. Lukáčová, A., Babič, F., Paralič, J.: Building the prediction model from the aviation incident data. In: 12th International Symposium on Applied Machine Intelligence and Informatics SAMI 2014, Slovakia, pp. 365–369. IEEE (2014)
13. Murthy, K.S.: Automatic construction of decision tress from data: A multidisciplinary survey. In: Data Mining and Knowledge Discovery, pp. 345–389 (1997)
14. Nazeri, Z., Bloedorn, E.: Exploiting Available Domain Knowledge to Improve Mining Aviation Safety and Network Security Data. The MITRE Corporation, McLean, Virginia (2004)
15. Oza, N., Castle, J.P., Stutz, J.: Classification of aeronautics system health and safety documents. IEEE Transactions on Systems, Man, and Cybernetics, Part C 39(6), 670–680 (2009)
16. Quinlan, J.R.: Induction of Decision Trees. Machine Learning 1(1), 81–106 (1986)
17. Quinlan, J.R.: C4.5: Programs for Machine Learning. Morgan Kaufmann, San Mateo (1993)
18. Van Rijsbergen, C.J.: Information Retrieval, 2nd edn. Butterworth (1979)
19. Wiegmann, D., Shappell, S.: Human error analysis of commercial aviation accidents: Application of the Human Factors Analysis and Classification System (HFACS). Aviation, Space and Environmental Medicine 72(11), 1006–1016 (2001)

Chapter 18
Generative Visual Identity System

Jarosław Andrzejczak and Kinga Glinka

Abstract. This work presents how generative systems and computer-based algorithms can be used in visual identity. Generative mechanisms and special systems (defining rules and constraints) become an alternative for this part of graphic design, automating all process of creating identity. The chapter shows new solutions: information rich generative logo and generating not only mark but also graphic elements of identification. The authors propose generative visual identity system for employees of the Lodz University of Technology. It is able to generate over 1.5 million unique logos with other identity elements for employees from their features. The system and usability tests of generated marks are presented in this work.

18.1 Introduction

Nowadays, along with rapid technological changes and growing customer needs we observe a continuous search for new forms of visual identity. People no longer expect only visual identity in itself but also uniqueness, freshness and above all effectiveness. Flexibility and dynamism of the identification are increasingly often required, which show the ongoing transformation and development of the company, as well as its adaptation to the continuously changing world and the specific needs of customers [14]. Dynamic forms are expected to be more effective in attracting attention and more memorable. Therefore, modern visual identity ceases to be only designed, but its graphics begin to be created by means of programming possibilities. The most commonly used tool are generative systems, derived from generative art [7, 16]. The operation of a generative system can be based not only on the established rules and restrictions but also input data, thus becoming form of information visualization [10, 11].

Jarosław Andrzejczak · Kinga Glinka
Institute of Information Technology, Lodz University of Technology, Poland
e-mail: jaroslaw.andrzejczak@p.lodz.pl, 800559@edu.p.lodz.pl

A. Zgrzywa et al. (eds.), *New Research in Multimedia and Internet Systems*,
Advances in Intelligent Systems and Computing 314, DOI: 10.1007/978-3-319-10383-9_18

The existing generative visual identities were created mostly after 2009. Against the standard approach they are still rarely used. The base and only generated element of the identification in those systems is the logo [1], and the whole idea of generating focuses only on it. The existing generative marks can be divided into two groups: randomly[1] generated and created on the basis of specific information or data, including avatars [8, 14]. Most of these solutions use Processing[2] as a tool for the implementation of a generative system. However, the complexity of the implemented systems and even individual algorithms can vary. Most of them are based on random generation of the mark. In contrast, those whose starting point is data or information are usually limited to a small number of visualized features (two or three), with no possibility of reading them accurately from the generated mark. References do not mention the generation of other identity elements by the system (except the mark). Most likely they are created manually by the designers[3].

18.2 The Problem to Solve

In the previous section, the authors highlighted the main problems of the existing solutions, namely:

- a small number of data-based systems,
- a small number of visualized features,
- no possibility of reading back information from the mark,
- no systems for generating most of the identity elements (not only the mark itself).

Based on the conclusions of the analysis, the authors have designed and implemented their own system offering solution to the problems mentioned above. This approach extends the functionality of visual identity created on a standard basis and fully uses the capability of the generative system in the creation process. The authors made a few assumptions: the mark is generated by the system, being an information rich, fully bidirectional[4], and visualizing as big number of features[5] as possible. The problem analysed by the authors is the selection of appropriate means and methods of visualization which will allow to read the proposed number of features from the mark. In the authors' system the number of features is nine. On the other hand, the mark should meet the criteria for the logo [1] to become a base of designed visual identity (for example good scalability without losing readability, even for small size

[1] *Randomness* is understood to create elements based on some rules and constraints, but with no specific content and information.

[2] `http://processing.org/`

[3] In contrast to the automatic process using generative system.

[4] Bidirectional mark is created based on specific data or information, that later can be read from the same mark.

[5] According to the authors, *big number of features* is the number significantly exceeding that visualized in the analyzed generative visual identities, but also limited by human capacity for processing information, according to Miller's Law (magical number 7, plus or minus 2) [13].

on a business card). Its graphic elements are also generated by a separated module of the system.

18.3 Proposed Generative Visual Identity System

The authors created generative visual identity for research and education employees of the Lodz University of Technology. Each employee can be characterized by a number of features which determine their position and affiliation to the specific units. Of these, nine features were selected that form a coherent and complete data source to visualize them in a logo project. They were grouped into six categories:

- sex,
- degrees (MSc, PhD, Prof., etc.),
- faculty and functions on it (dean, vice-dean),
- institute and functions at it (heads of institute),
- team and function on it (head of team),
- position at the university (assistant, adjunct, professor, etc.).

The shape of the mark depends on the employee's sex, referring to the symbol of Venus or Mars [6]. It is divided into four main parts (inspired by the Socrates square [17]) and every one of them is also again divided into even smaller square elements (Fig. 18.1, A). Logo colors reflect each of the nine faculties of the Lodz University of Technology (Fig. 18.1, B). They are based on currently used faculty colors and were chosen to maximize the differences between them (and consequently being easily distinguishable by recipients). Because more than one employee might have all the same features (mentioned above), it was necessary to introduce an element allowing their distinction. This was solved by calculating the color from employee's name and it filled the sex quarter (quarter 1 in Fig. 18.1, C). The result is the base form of the logo, on which next features are marked. The logo quarters are numbered clockwise from 1 to 4, starting from the sex quarter (Fig. 18.1, C). Every quarter is responsible for different features:

1. Degrees and functions on the faculty – the sex quarter.
2. Institute and functions at it – the darkest quarter from the color range.
3. Team and function on it – diagonally from the sex quarter.
4. Position – the brightest quarter from the color range.

According to the authors' general rule, further information connected with affiliation to the individual units are indicated by painting the appropriate small squares in the color range from the previous quarter. In contrast, the idea of labelling functions (to distinguish them from the information about the affiliation) is to remove relevant elements from the basic shape or add new ones in the particular places.

The degrees (like PhD, MSc, etc.) are marked in the sex quarter (number 1), according to Fig. 18.2. Painted element means that the employee has this degree. The further from the centre of the mark (depending on sex: men up, and down for women), the higher degrees are. The same degrees in both sexes are a mirror image relative to the horizontal axis, which facilitates comparing the marks.

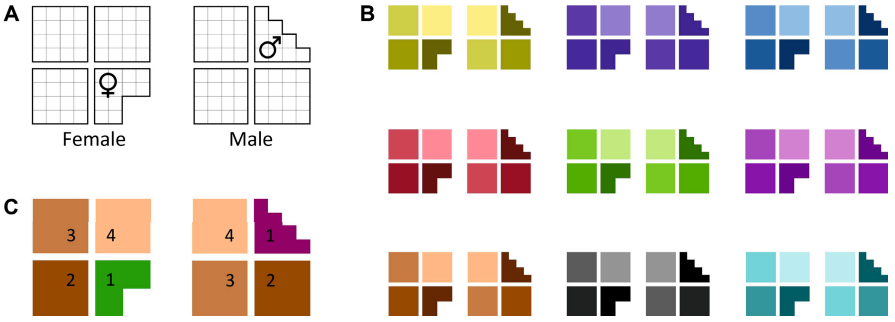


Fig. 18.1 Logo construction: A – shape and division into smaller components, B – nine basic faculty colors, C – an example of color computed from employee's name (put on the one of basic faculty colors), quarters numeration

In the next quarter in a clockwise direction (number 2 which is also the darkest from the color range, Fig. 18.2) there is information about the affiliation to the institute. Each institute is represented by the number of the painted squares on the non-white background. They are filled rows (so in certain order) in accordance with the number of the unit, therefore the brain faster estimates their number [5]. No painted small squares in this quarter show that employee does not belong to any institute. Possible functions are four heads of the institute – marked by cutting relevant elements with the white background, numbered from 1 to 4 in Fig. 18.2.

The team is marked in the quarter located diagonally from the sex quarter (number 3). The different number of painted small squares (from 1 to 14 in Fig. 18.2, without the one on the white background) represents different teams, same as the previous information about the institute. No painted squares mean that employee does not belong to any team. In this unit only one function is available, the head of the team – presented by cutting out the square marked number 1 on the white background in the team quarter. Against to the vertical axis of the logo, functions in the institute quarter are cut outside of it for both sexes and in the team quarter – inside.

The brightest quarter from the color range (number 4) is responsible for the employee's position. There are 9 possible positions – the more small squares are painted (numbered as in Fig. 18.2), the higher position is. All nine painted is the highest position and it means a professor. The lowest position, the service of learning and teaching, is represented by one square.

Dean and vice-dean, as the most important[6] functions at the university, are marked by a new small square placed in the quarter number 1 (Fig. 18.2). The same functions for sexes are mirror image relative to the horizontal axis, like the degrees.

In Fig. 18.2 besides the rules of the logo construction, there are two examples of marks generated by the system (part named *Examples*). In accordance with the number of all units at the Lodz University of Technology, degrees, positions and

[6] Function on the faculty is more important than the function at the institute or on the team.

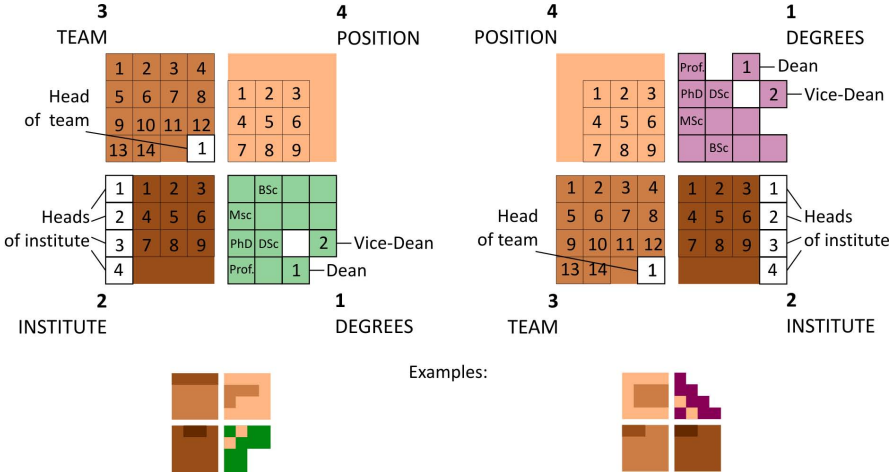


Fig. 18.2 Logo construction by quarters: 1 – degrees and functions on faculty, 2 – institute and functions at it, 3 – team and function on it, 4 – position

functions, the proposed system is able to generate over 1.5 million unique logos (formula (1), according to the Polish degrees naming).

$$CoS = 2D\,(F_F + 1)\,(I_F + 1)\,P \sum_{n=0}^{f-1} \sum_{m=0}^{i_n - 1} ((T_{nm} + 1)\,(T_F + 1)) \tag{18.1}$$

where CoS is the capability of the system, D – number of degrees, F_F[7] – number of functions on faculty, I_F – number of functions at institute, P – number of positions, f – number of faculties at the university, i_n – number of institutes on the particular faculty number n, T_{nm} – number of teams at the particular institute (institute number m on the faculty number n), T_F – number of functions on team.

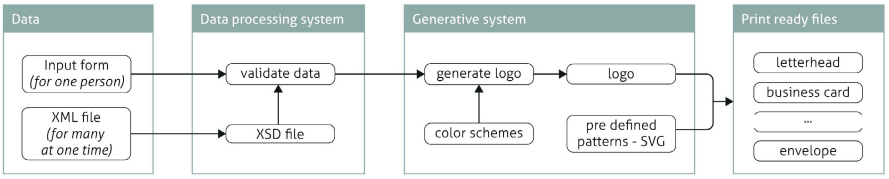


Fig. 18.3 Process of generation a whole visual identity

The system was expanded with the ability to automatically generate files of several personalized identity elements. The whole process of generation can be seen in

[7] For all features where it can occur, it is also included its absence (for example no team which employee belongs to or no functions in the specific units of the university), so "+1" in the formula.

Fig. 18.3. The generated elements are: a business card, a letterhead, a DL envelope, a presentation template, a CD print and a paper folder. The generation of all these elements for the entire staff of the Lodz University of Technology[8] would take less than 9.5 minutes[9] and not require a designer's work.

18.4 Usability Tests With Users

The aim of the performed tests was to examine if the number of features used in generated logo and their visual representations allow easy recognition of the information included in the sign. As described before, we use nine features, which is the upper limit of human capacity for processing information, according to Miller's Law (magical number 7, plus or minus 2) [13]. The test group consisted of potential future users of the generated logo – 7 employees and 20 students of the Lodz University of Technology. The survey was divided into 3 stages spread over time, and measures were based on Jakob Nielsen's quality components [15]:

- Learnability: How easy is it for users to start using the generated logo and to read information (employee's characteristics) from it?
- Memorability: How much information the user can recognize in the generated logo during the first and subsequent tests? Which of the employee's features are easiest to remember?
- Errors: How many errors do users make during tasks and how does the number of them vary between stages of the survey? Which of the employee's features causes the most problems?
- Satisfaction: What are the users' ratings of the system? Are they satisfied with such information rich visual identity?

At each stage participants answered the questions with the same level of difficulty and complexity. During the tests, the Lodz University of Technology employees' logos generated using the system were used. Every stage consisted of four parts, each focused on specific topic regarding interpretation of the generated sign:

1. Interpretation of logo division into four main areas along with features associated with them.
2. Interpretation of two compared logos and their relation.
3. Interpretation of a few logos and choosing those, which satisfy certain criteria.
4. Interpretation of a single logo (without others to compare).

At the beginning of the first stage participants were provided with an instruction sheet which shows the mapping of information onto a generated logo. Those rules were also explained to the test group by the person conducting the usability test. The first and the second stage took place immediately after each other. In the first

[8] 2971 employees at the university in 2013, according to data from Personnel Department of the Lodz University of Technology.

[9] Tested on Sony Vaio laptop with a dual-core processor Intel Core i5 2.27 GHz, 4 GB RAM and 64-bit operating system Windows 7 Professional.

stage the participants were allowed to use the prepared instruction sheet with the previously explained rules of how to use visual identity. The second stage was based only on the knowledge acquired in the previous one. The last stage was performed after two weeks, also without a possibility of using the instruction sheet. At the end of the survey the additional polls were performed according to the rules of System Usability Scale [2, 4]. An attempt to evaluate the system's usability was based on the tasks performed by the test group as well as their opinions and comments about the system gathered using questionnaires and polls [12, 15].

18.5 Tests Results and Conclusions

Each of the three survey stages consisted of 25 questions, after which the opinions of the participants were examined. At the first stage the understanding of logo generating rules has been studied (due to the possibility of using the instructions sheet). On average, respondents correctly interpreted 92% (σ = 8%) of the information included in the generated logo. Next two stages tested how much information can be read immediately after getting to know the rules, and after a two-week break. In both cases, the participants were able to correctly interpret over 75% of the information (Table 18.1) – despite the natural decline in the number of correct answers.

Table 18.1 Percentage of correct answers at each stage[a]

	Stage 1		Stage 2		Stage 3	
	mean [%]	σ [%]	mean [%]	σ [%]	mean [%]	σ [%]
Percentage of correct answers	92	8	88	11	76	15
Percentage of correct answers (relative to the first stage)			92	9	79	13

[a] σ – standard deviation.

If we treat the first stage as a reference for the other two[10], the result will be even higher (an increase of 4% in the second stage, and 3% in the third, with reducing the standard deviation of 2% in both cases). Considering the high number of features included in one sign, the obtained level of errors is satisfactory. Similar to the percentage of correct answers, mean user satisfaction from the use of the system in order to obtain information about a specific employee from the generated logo is set[11]. It is at 72.59 (σ = 12.64) percentile, which after normalization gives an evaluation of "B" in the traditional school grading scale (according to [4]). This

[10] In the first stage, participants were allowed to use the instructions sheet, so mistakes made in this step can be interpreted as the result of misunderstanding of the logo generating rules. Thus, if the error made in the first stage occurs successively in subsequent test, it should not be interpreted as an error.

[11] According to the results of the SUS study.

score also means that the our system's usability is acceptable and gives adjective rating of "good" [2]. The idea of generating logo that shows each employee's personal characteristics is ranked even higher – at 87.41 (σ = 15.84) percentile, which gives respectively an "A" and "excellent" grades (the overall result underestimates the difficulties in the interpretation of some of the features, described later in this section). At the same time, there was no correlation between the number of correct answers and the participant's level of satisfaction (Table 18.2). Some users despite the many errors made ranked the system positively, while others gave neutral rank while answering mostly correctly to the survey's questions.

Table 18.2 Correlation between mean number of correct answers and satisfaction[b]

	r	df	*t*-value	critical *t*-value	p≤0.05
Correlation between mean number of correct answers and satisfaction	0.3277	25	1.734	2.052	

[b] r – Pearson's correlation coefficient, df – degrees of freedom, *t-value* – t-test result (2-tailed), *critical t-value* – critical value of t for given df, p – significance level.

Tests have shown that there is a significant difference between errors made by participants when they analyse a single logo and many of them at one time ($p \leq 0.05$, Table 18.3). Authors have assumed that created sign should function independently as well as compared to the other, so there is still work to do in that field.

Table 18.3 Difference between number of errors made with and without comparison to another logo[c]

	df	*t*-value	critical *t*-value	p≤0.05
Difference between number of errors made with and without comparison to another logo	26	2.174	2.055	0.039

[c] df – degrees of freedom, *t-value* – t-test result (2-tailed), *critical t-value* – critical value of t for given df, p – significance level.

Let's look at the number of errors made in each of the twenty five questions in all of the three stages. We can group these questions according to the features they refer to (discussed in Section 18.3), which will give us seven categories (Table 18.4). Notice that fourteen questions from three categories represents about 76% of all errors (the remaining 24% relates to eleven questions from the four categories). The study has shown that the following features were easiest to interpret by the participants: faculty, degree and sex. While hardest to remember and read from logo were: position, institute and team with functions in them.

Table 18.4 Questions grouped into seven categories that represents features[12]

Feature	Errors percentage [%]	Mean number of errors per question
Faculty	1	1.5
Degree	5	3.5
Sex	8	3
Functions on faculty	10	14.5
Team and function in it	23	14.2
Position	23	17.5
Institute and functions at it	30	18.4

The above results overlap with the opinions and participant satisfaction measurements, i.e. the more errors made in questions about a specific feature, the more likely it was ranked worse or mentioned as difficult to interpret. For example, 89% of the participants indicated sex as a feature very easy to find in a generated sign, and satisfaction for this feature was at 99.26 (σ = 3.84) percentile ("A" grade after normalization). Best remembered and rated were the features presented in the logo through color, shape and position (in accordance with the hierarchical layout). Most difficulties were caused by those employee characteristics that were marked by filling smaller squares in the specific quarter (where significant was the number of them, not their layout). It has also turned out that counting the quarter representing each feature from the quarter representing sex is also problematic (in case of remembering and comparing employees of various sexes). Although, such solution probably caused a low error level in the interpretation of gender features.

The participants pointed out in surveys the difficulty in memorizing the order of features (which quarter contains information about the institute, the team and position). This was directly reflected in the results of the tests. The best identified features lie in the first quarter, while those which constituted 76% of all errors lie in the consecutive quarters. Also, if we look at the data, we will notice that the answers to the questions about gender composed solely of correct subset or the subset complementary to it[13]. This indicates that the participants have correctly found that feature in generated logo, but mistakenly interpreted its value. Such errors can be probably eliminated by re-explanation of the logo generating principles.

Finally, it is worth mentioning that the extremely low mean values of correct answers (within 60-65%) were mostly the result of answers such "I do not know" or "I do not remember", rather than simply wrong answer[14]. At the same time,

[12] Features: faculty and function on it has been separated into two categories because there was a significant difference between the number of errors made in questions concerning those features.

[13] For example, from the given set of [A, B, C, D, E] where the correct answer was [A, C, E] participants answered even correct subset or [B, D] .

[14] Participants were asked to make such annotation, if they were not sure about the answer. This approach allowed us to minimize false positive answers.

only complete answers were considered as correct (if in any of the questions the participant has pointed out only part of correct answer, it was considered an error).

18.6 Summary and Further Development

The tests of the created system have shown the potential of information rich generative logo as well as the strengths and weaknesses of such a solution. Even with such a large number of features included in the generated sign, users were able to read about 75% information provided with the mean satisfaction level at 72 percentile ("B" grade). At the same time, the study described in the article is only the prelude to the next tests, which will focus on improving the readability of individual features by changing the ways of their presentation. It is also possible to develop a system for automated verification of feature readability (similar to one proposed in [3]) or verify chosen features and their representation using an automated visualization assistant (like ViA [9]).

References

1. Adams, S., Morioka, N., Stone, T.: Logo Design Workbook: A Hands-On Guide to Creating Logos. Rockport Publishers, Massachusetts (2004)
2. Bangor, A., Kortum, P.T., Miller, J.T.: An empirical evaluation of the system usability scale. International Journal of Human-Computer Interaction 24(6), 574–594 (2008)
3. Bednarski, R., Pietruszka, M.: The Computer-Aided Estimate of the Text Readability on the Web Pages. In: Zgrzywa, A., Choroś, K., Siemiński, A. (eds.) Multimedia and Internet Systems: Theory and Practice. AISC, vol. 183, pp. 211–220. Springer, Heidelberg (2013)
4. Brooke, J.: SUS: A Retrospective. Journal of Usability Studies 8(2), 29–40 (2013)
5. De Cruz, H.: Why humans can count large quantities accuratel. Philosophica 74(2), 63–83 (2004)
6. Frutiger, A., Heiderhoff, H.: Der Mensch und seine Zeichen. Marix Verlag, Wiesbaden (2004)
7. Galanter, P.: What is generative art? Complexity theory as a context for art theory. In: Proceedings of the 6th Generative Art International Conference (GA 2003), pp. 225–245. AleaDesign Publisher, Milan (2003)
8. Glinka, K.: Generative visual identity. Master thesis, pp. 33–60. Lodz University of Technology (2013) (in Polish)
9. Healey, C.G., Kocherlakota, S., Rao, V., Mehta, R., St. Amant, R.: Visual Perception and Mixed-Initiative Interaction for Assisted Visualization Design. IEEE Transactions on Visualization and Computer Graphics 14(2), 396–411 (2008)
10. Klanten, R., Ehmann, S., Bourquin, N., Tissot, T.: Data Flow: Visualizing Information in Graphic Design. Die Gestelen Verlag, Berlin (2008)
11. Lankow, J., Ritchie, J., Crooks, R.: Infographics: The Power Of Visual Storytelling. Wiley Publishing, New York (2012)
12. Lazar, J., Feng, J.H., Hochheiser, H.: Research Methods in Human-Computer Interaction. Wiley Publishing, New York (2010)

13. Miller, G.A.: The Magical Number Seven, Plus or Minus Two: Some Limits on Our Capacity for Processing Information. Psychological Review 63(2), 81–97 (1956)
14. Van Nes, I.: Dynamic Identities: How to create a living brand. BIS Publishers, Amsterdam (2012)
15. Nielsen, J.: Usability Engineering. Morgan Kaufmann Publishers Inc., San Francisco (1993)
16. Pearson, M.: Generative Art: A practical guide using Processing. Manning Publications Co., New York (2011)
17. Zubrzycki, J.S.: Cieślictwo Polskie, p. 128. Grafika, Warsaw (2010)

Part IV
Web Systems and Network Technologies

Chapter 19
Analysis of Web Service Retrieval Methods

Adam Czyszczoń and Aleksander Zgrzywa

Abstract. The purpose of this chapter is to investigate different Vector Space Model approaches for Web Service Retrieval in a manner that allows the retrieval of both SOAP and RESTful Web Services. The research also includes Latent Semantic Indexing and modified term-document matrix that allows to store scores for different service components separately. All indexing models are also investigated against different weighting schemes. Additionally, this chapter introduces three test collections that were used to test all approaches in terms of effectiveness and performance. The collections can also be used for many other benchmarks on Web Service Retrieval.

19.1 Introduction

Web services are application components that are interoperable and platform independent. There exist two classes of Web Services – commonly referred to in the literature as SOAP Web Services and RESTful Web Services [1, 3, 8, 11]. Services of the first class are used mainly in the industry and lightweight RESTful Web Services are used mainly in Web applications. Developers have access to a variety of services that are published on the Internet. However, finding most suitable services from the huge collection available on the Web is still the key problem. To solve it, Web Service Retrieval techniques are used that are based on Information Retrieval models and use web crawlers to collect services that are publicly accessible on the Internet. In contrast to ineffective keyword-based service discovery methods, service retrieval can be applied to both SOAP and RESTful Web Services. The most effective and commonly used model is the Vector Space Model (VSM) that allows to model services as vectors and compare their relevancy. Another common model

Adam Czyszczoń · Aleksander Zgrzywa
Wrocław University of Technology, Institute of Informatics,
Wybrzeże Wyspiańskiego 27, 50-370 Wrocław, Poland
e-mail: {adam.czyszczon,aleksander.zgrzywa}@pwr.edu.pl

A. Zgrzywa et al. (eds.), *New Research in Multimedia and Internet Systems*,
Advances in Intelligent Systems and Computing 314, DOI: 10.1007/978-3-319-10383-9_19

is the Latent Semantic Indexing (LSI) that is based on VSM but allows to capture the higher-order association between services.

In this chapter we investigate current approaches to Web Service Retrieval in a model that allows the retrieval of both SOAP and RESTful Web Services. We evaluate the VSM and LSI indexing methods against different variants of *TF-IDF* (Term Frequency – Inverted Document Frequency) weighting schemes. Additionally, we present modified term-document matrix that allows to store scores for different service components separately. Such a "extended" indexing method is applied for both VSM and LSI, that gives in total four different index structures to be tested. Presented evaluation results concern effectiveness and performance. Additionally, we introduce our Web Service test collections used in evaluation and that can also be used for many other Web Service Retrieval benchmarks.

19.2 Related Work

One of the earliest research on Web Service Retrieval that used Information Retrieval methodology and utilized the VSM was presented in [10]. Authors used standard *TF-IDF* weighting scheme, inverted index file (term-document matrix) and cosine coefficient to calculate the relevance. However, authors indexed only service descriptions and depended on service repositories. The same model was followed by many other authors but they extended it with indexing more than just descriptions [12, 4], introduced grouping services into concepts [9] or proposed different indexing and ranking methods [5].

In contrast to VSM, the LSI approach allows to find hidden semantic association between terms even though they do not appear in any service. One of the earliest papers on LSI for Web Services was proposed in [7]. Introduced method used standard VSM model with *TF-IDF* inverted index file. The method, however, relied on public service repositories and concerned service descriptions only. In [13] authors presented LSI approach that also utilized standard VSM model but indexed more information than just descriptions. In further research [14] authors presented effectiveness evaluation but only for single term queries.

To sum up, any of the above research tested different weighting schemes and all of the mentioned studies were applied for SOAP Web Services only. Additionally, all the research also lacked common evaluation test collection. In many cases authors gave only the information about destination URLs of public Web Service directories but such a information quickly becomes outdated and thus not usable.

19.3 Standard Web Service Retrieval

In Vector Space Model services are represented as vectors. Every element in the document vector is calculated as weight that reflects the importance of a particular term that occurs in a particular service. The weights are computed using *TF-IDF* scheme. Weights are stored in data structure called inverted index – a term-document matrix where document vectors represent the columns and term vectors represent rows.

User query is also converted to vectors in the same manner as services. To calculate the similarity between a query and document the cosine value between this two vectors is calculated. Because vectors have different lengths they have to be normalized to the unit equal to one. Additionally, there are many different *TF-IDF* weighting variants. According to the SMART notation [6], the weighting options are denoted in a form of *ddd.qqq* syntax where first three letters represent document vector weighting and second three query vector weighting. The first letter in each triplet specifies the term frequency component of the weighting, the second the document frequency component, and the third the form of normalization used [6].

The LSI is a variant of the VSM in which the original VSM matrix is replaced by a low-rank approximation matrix using the Singular Value Decomposition. Assuming that A is a $m \times n$ matrix where $a_{ij} \in A$ represents *TF-IDF* weight of $i-th$ term in $j-th$ service, the result of the decomposition is following: $A = U\Sigma V^T$, where A is the initial VSM term-service matrix, U is the matrix which columns represent term vectors, Σ is a diagonal matrix of size $R^{terms \times services}$ containing singular values, and V is the matrix which columns represent the service vectors. By reducing matrix Σ into Σ_K as $K \times K$ matrix and projecting it on term matrix U and service matrix V^T we obtain the reduced matrix $A_K = U_K \Sigma_K V_K^T$. After reduction, the terms are represented by the row vectors of the $m \times K$ matrix $U_K \Sigma_K$, and services are represented by the column vectors the of the $K \times n$ matrix $\Sigma_K V_K^T$.

In LSI the query vectors are extracted from term matrix based on query terms. The term matrix is created during the indexing process, and thus the SMART notation *ddd.qqq* for LSI is equal to *ddd.ddd*.

19.4 Extended Web Service Retrieval

The definition of Web Service structure was elaborated in our previous study [2]. We consider Web Service to be composed of quadruple of elements α where the first three represent service *parameters* which correspond to service *name*, *description* and *version*, and the fourth element represents service *components* which are composed of six-tuple of following elements: *name*, *input value*, *output value*, *description*. To simplify we join all components as one bag-of-words. The biggest advantage of presented structure is that it conforms to both SOAP and RESTful Web Services.

All current retrieval approaches treat all service elements as single bag-of-words. The extended retrieval approach means indexing each service element α separately. Based on our research carried out in [2] Web Services are indexed in the following manner: the extended index A is a $m \times n$ matrix where $a_{ij} \in A$ is a four-tuple $< \alpha_{ij1}, \alpha_{ij2}, \alpha_{ij3}, \alpha_{ij4} >$ where α_{ijx} represents weights of $i-th$ term for $j-th$ service's $x-th$ element. Weights are calculated using the modified *TF-IDF* scheme presented in Equation 19.1.

$$TFIDF_{extended} = (1 + log(\mathrm{tf}(t,\alpha))) \cdot log \frac{N_\alpha}{|\{\alpha \in WS : t \in \alpha\}|} \tag{19.1}$$

where $\mathrm{tf}(t,\alpha)$ is term frequency of term t in element α, N_α is the total number of α elements, and $|\{\alpha \in WS : t \in \alpha\}|$ is the number of service elements where the term t appears. Because service element counters are stored separately, there is substantial difference if weights between standard VSM indexing model and the extended one. Afterwards the extended index is reduced to the size of the basic VSM index structure. This is achieved using the Mean Weight Vector (MWV) method presented in [2] that is computed as average weight of all service elements. The final index structure is following: A' is a $m \times n$ matrix where $a_{ij} \in A'$ represent the MWV weight $\sigma_{ij} = \frac{1}{4}\sum_{x=1}^{4} \alpha_{ijx}$.

19.5 Test Collections

In this section we present test collections that will be used in our evaluation[1]: *SOAP_WS_12*, *SOAP_WS_14*, *SOAP_WS_12-14*. Collections are prepared using web crawlers. First collection contains 267 Web Services collected from `xmethods.net` – a directory of publicly available Web Services, used by many researchers in various benchmarks. Second collection contains 662 services collected from: `xmethods.net`, `service-repository.com`, `webservicex.net`, `venus.eas.asu.edu`, `visualwebservice.com` and `programmableweb.com` – popular repository of public Web Services. Third collection represents merged collections one and two, including elimination of duplicate services. In Table 19.1 we present summary data of the above collections.

Table 19.1 Test collections summary

	SOAP_WS_12	SOAP_WS_14	SOAP_WS_12-14
Services	267	662	889
Parameters	432	886	1254
Components	5140	18353	22660
Total elements	5572	19239	23914

At present, introduced in this chapter test collections do not include any RESTful Web Services because our methods of their identification on the Web are still being improved and currently developed collection is too small. This, however, does not influence experimental results presented in this chapter.

19.6 Evaluation

Based on the approach presented in this chapter we implemented indexing system that allowed us to conduct evaluation experiments. The evaluation concerns performance and effectiveness analysis of basic and extended LSI and VSM indexes for *ltc.ltc*, *ltc.lnc*, *lnc.lnc*, *lnc.ltc*, *mtc.ltc*, and *mtc.lnc TF-IDF* schemes, according

[1] All test collections are available to download at:
`http://www.ii.pwr.edu.pl/~czyszczon/WebServiceRetrieval`

to the SMART notation. The *lnc* variant for term weighting is a desired one, since it allows to skip some computation during the indexing process and reduce indexing time. In order to evaluate the effectiveness, we used the *WS_SOAP_12* test collection and the following classical information retrieval metrics: *Precision*, *Recall*, *F-measure* ($\beta = 1$) and *Mean Average Precision (MAP)*. In order to evaluate performance we checked the retrieval time, indexing time and index size for *WS_SOAP_12*, *WS_SOAP_14*, *WS_SOAP_12-14* test collections of every index structure. The experiment was carried out for following queries: *"temperature conversion"*, *"email validation"*, *"weather forecast"*, *"weather forecast service"*, *"currency exchange"*, *"demographics"*, *"new york"* and *"send sms"*. The MAP is computed as the average precision for all mentioned queries, whereas the average precision value is calculated for each of the *k* top relevant services in search results list.

19.6.1 Effectiveness Analysis

Based on our preliminary studies on LSI for Web Service Retrieval, we established that dimension reduction $K = 41$ gives the best effectiveness at possibly low indexing time and index size. Therefore, evaluated in this chapter LSI index is reduced to 41 dimensions. Selecting the best indexing model is evaluated using MAP at top 100 positions.

In Figure 19.1 we present the evaluation of potentially most effective *ltc.ltc* scheme against the *ltc.lnc*. Since for LSI the SMART natation is *ddd.ddd*, presented figure does not contain the LSI index for *ltc.lnc* (the LSI of *ltc.ltc* is the same as LSI of *ltc.lnc*). Both basic and extended LSI indexes were much less effective than the VSM, where for the VSM variants the worst one was the *VSM Extended ltc.lnc*. The basic version of *VSM ltc.lnc* returned less relevant services than *ltc.ltc* variants within the top 2 results, however, later it returned slightly more relevant results. Both *VSM ltc.ltc* variants performed very close to each other, with tiny difference at 19-th position on the favor of the basic structure. In general, the extended LSI was much more effective than the basic version, and the *VSM Extended ltc.lnc* was the worst method among the VSM.

In Figure 19.2 we present evaluation of *lnc.lnc* and *lns.ltc* schemes. Again LSI methods have much lower precision than VSM methods. All LSI methods performed almost at the same level. Both *VSM lnc.ltc* had the same results and had higher effectiveness than *VSM lnc.lnc*. In the case of *VSM lnc.ltc* indexes, the extended version was slightly more effective. The worst methods among the VSM were the *VSM Basic lnc.lnc* and *VSM Extended lnc.lnc*.

In Figure 19.2 we present evaluation of *mtc.ltc* and *mtc.lnc* schemes. One more time LSI methods have much lower precision than VSM methods. All VSM indexes performed very close to each other, with small advantage of the extended indexes. However, *mtc.lnc* approaches had higher overall MAP, in the beginning at top 2 positions, they resulted in lower recall. The worst method among VSM was *VSM Basic mtc.ltc*, then *VSM Extended mtc.ltc*.

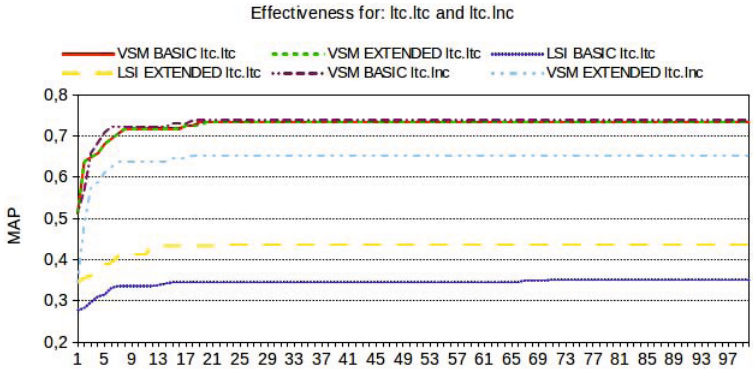


Fig. 19.1 MAP at top 100 positions for *ltc.ltc* and *ltc.lnc* schemes

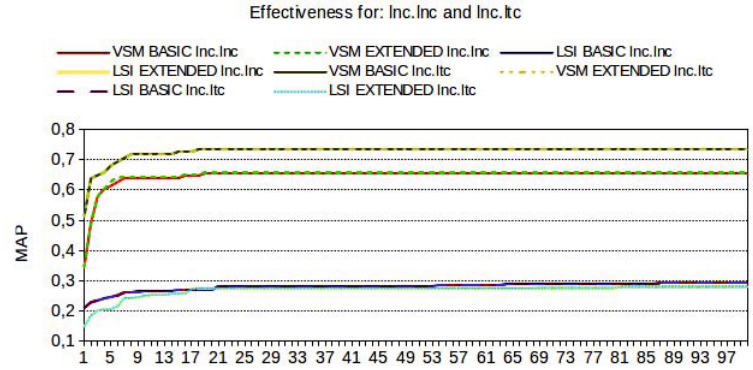


Fig. 19.2 MAP at top 100 positions for *lnc.lnc* and *lnc.ltc* schemes

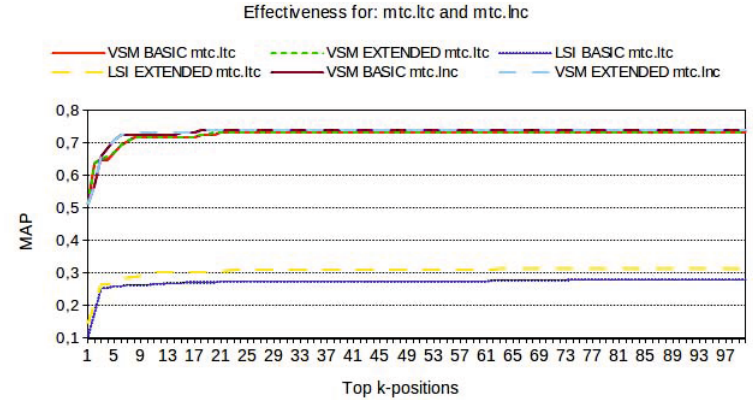


Fig. 19.3 MAP at top 100 positions for *mtc.ltc* and *mtc.lnc* schemes

To sum up, the best results for LSI index were obtained for *ltc.ltc* scheme where the extended index achieved much higher effectiveness than the basic one. In the case of VSM index, the best results were obtained by *VSM Basic lnc.ltc* and *VSM Extended lnc.ltc* equally, and *VSM Extended mtc.lnc*. In Figure 19.4 we illustrate 7 methods that obtained best results in terms of MAP. Those methods, together with both *LSI ltc.ltc* will be considered in further analysis.

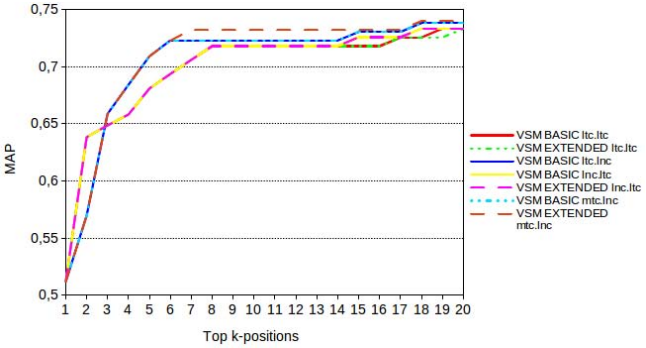


Fig. 19.4 Best seven VSM methods at top 20 positions

19.6.2 Performance Analysis

In order to investigate the performance of selected above most effective methods, we tested them against indexing time, index size, and retrieval time, for three test collections: *WS_SOAP_12*, *WS_SOAP_14*, *WS_SOAP_12-14*. In Figure 19.5 we illustrate the indexing time and index size. Both LSI methods are characterized by the longest indexing time (except for a *WS_SOAP_12* dataset) and very small index size. The indexing time of extended approach was much higher than in basic approach. However, the index size of extended approach was a little smaller than in the basic one. In the case of the VSM methods, the indexing time of extended approach was also much bigger than in the basic one. The index size was almost the same. The *lnc* and *mtc* performed much better than *ltc*. The shortest indexing time was obtained by the basic methods, where the *VSM BASIC mtc.lnc* was the best one and the *VSM Basic lnc.ltc* was the second one. Extended versions of these indexes performed 300%-425% slower, however indexing using *VSM Extended lnc.ltc* was around 10%-15% faster than *VSM Extended mtc.lnc*.

In Figure 19.6 we present the average indexing time. Both LSI methods had the fastest retrieval time for every test collection, however the extended approach was around 4%-7% faster than the basic approach. In the case of VSM methods, the retrieval time for such a small dataset as *WS_SOAP_12* was almost the same. The difference can be noticed for bigger datasets. The 3 best results were archieved by the *VSM Extended mtc.lnc*, *VSM Basic ltc.lnc*, *VSM Extended ltc.ltc* methods. The worst results were obtained by *VSM Basic mtc.lnc* and *VSM Extended lnc.ltc*. On the other hand, the differences are very small.

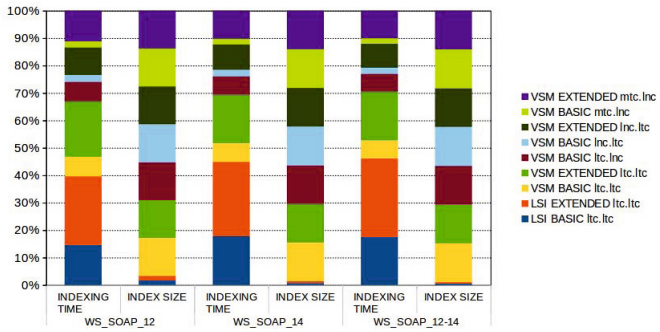


Fig. 19.5 Average indexing time and average index size of nine best indexing methods

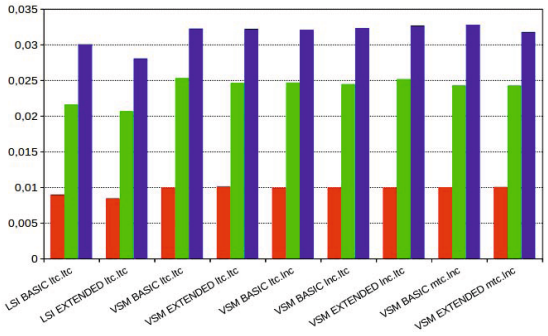


Fig. 19.6 Average retrieval time of nine best indexing methods

19.6.3 Effectiveness Comparison of Best Retrieval Methods

Based on the above effectiveness analysis and performance analysis, from selected best nine methods, we excluded from further research following approaches: *VSM Extended ltc.ltc* – it had lowest effectiveness among VSM methods and the longest indexing time, whereas, its retrieval time was comparable to other approaches; *VSM Basic ltc.ltc* – second smallest effectiveness, indexing and retrieval time worse than in other methods with higher effectiveness; *VSM Basic ltc.lnc* – effectiveness the same as in the *VSM Basic mtc.lnc* but significantly higher indexing time; *VSM Extended lnc.ltc* – effectiveness the same as in *VSM Basic lnc.ltc* but much higher indexing time. In Figure 19.7 we present five remaining methods plotted on a precision-recall curve.

In the beginning the *VSM Basic lnc.ltc* outperforms other indexing methods, but at some level (after 2nd position – see Figure 19.4) its effectiveness drops down. In other words, this method returns most relevant results on the very top of the search results list, but after second position the search results are worse than in other VSM approaches. Moreover, at some point (after 6th position – see Figure 19.4) the *VSM Extended mtc.lnc* outperforms the basic *mtc.lnc* approach. In the case of LSI, the extended approach gives more relevant results at each point.

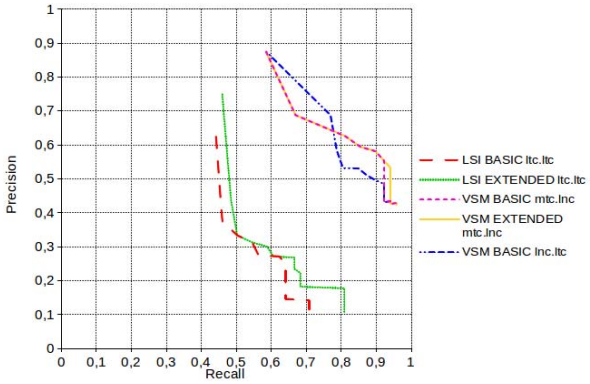


Fig. 19.7 Precision-recall curve of four best indexing methods

19.7 Conclusions and Future Work

The goal of presented research was to investigate different VSM approaches for Web Service Retrieval, including alternative approach that used modified index structure. The modified approach included extended matrix that allowed to count scores for different service components separately. Moreover, we presented three datasets that allowed us to test all approaches in terms of effectiveness and performance.

The experimental results indicate that best retrieval results can be achieved by basic *lnc.ltc* index or basic and extended *mtc.lnc* indexes, depending if we want to focus on returning most relevant results at top two positions or at top six positions. In terms of indexing time all methods, except the extended one, obtained best results. On the other hand, the extended approach turned out to be more effective than the basic one after 6th position, which is a very good result. The best *TF-IDF* scheme for the LSI methods was the *ltc.ltc*. However, the extended index proved to be much more effective than the basic LSI. Despite the fact that LSI approach return less relevant results, it captures the latent semantics between services and thus allows to return results where nothing would be found using VSM method. Therefore, it can be used as an alternative approach, when there are no results after using VSM index.

In further research we plan to check effectiveness for more queries and on all three test collections. We also plan to investigate more indexing methods and *TF-IDF* variants. For example, some researchers on Web Service Retrieval suggested that skipping length-normalization for LSI may bring better results. Our goal is to find or propose indexing method that gives highest effectiveness explicitly and that has possibly lowest indexing time and index size.

References

1. Booth, D., Haas, H., McCabe, F., Newcomer, E., Champion, M., Ferris, C., Orchard, D.: Web services architecture. W3C Working Group Note 11 February 2004, World Wide Web Consortium (2004), `www.w3.org/TR/ws-arch` (accessed April 2014)

2. Czyszczoń, A., Zgrzywa, A.: The concept of parametric index for ranked web service retrieval. In: Zgrzywa, A., Choroś, K., Siemiński, A. (eds.) Multimedia and Internet Systems: Theory and Practice. AISC, vol. 183, pp. 229–238. Springer, Heidelberg (2013)
3. Erl, T., Bennett, S., Schneider, R., Gee, C., Laird, R., Carlyle, B., Manes, A.: Soa Governance: Governing Shared Services On-Premise and in the Cloud. The Prentice Hall Service-oriented Computing Series from Thomas Erl. Prentice Hall (2011)
4. Hao, Y., Cao, J., Zhang, Y.: Efficient ir-style search over web services. In: van Eck, P., Gordijn, J., Wieringa, R. (eds.) CAiSE 2009. LNCS, vol. 5565, pp. 305–318. Springer, Heidelberg (2009)
5. Hao, Y., Zhang, Y., Cao, J.: Web services discovery and rank: An information retrieval approach. In: Future Generation Computer Systems, vol. 26, pp. 1053–1062 (2010)
6. Manning, C.D., Raghavan, P., Schütze, H.: Introduction to Information Retrieval. Cambridge University Press, New York (2008)
7. Paliwal, A.V., Adam, N.R., Bornhövd, C.: Web service discovery: Adding semantics through service request expansion and latent semantic indexing. In: IEEE SCC, pp. 106–113. IEEE Computer Society (2007)
8. Pautasso, C., Zimmermann, O., Leymann, F.: Restful web services vs. big web services: Making the right architectural decision. In: 17th International WWW Conference, Beijing (2008)
9. Peng, D.: Automatic conceptual indexing of web services and its application to service retrieval. In: Jin, H., Rana, O.F., Pan, Y., Prasanna, V.K. (eds.) ICA3PP 2007. LNCS, vol. 4494, pp. 290–301. Springer, Heidelberg (2007)
10. Platzer, C., Dustdar, S.: A vector space search engine for web services. In: Proceedings of the 3rd European IEEE Conference on Web Services (ECOWS 2005), pp. 14–16. IEEE Computer Society Press (2005)
11. Richardson, L., Ruby, S.: RESTful Web Services: Web Services for the Real World. O'Reilly Media, Inc., Sebastopol (2007)
12. Wu, C., Chang, E.: Searching services "on the web": A public web services discovery approach. In: Eighth International Conference on Signal Image Technology and Internet Based Systems, pp. 321–328 (2007)
13. Wu, C., Chang, E., Aitken, A.: An empirical approach for semantic web services discovery. In: Australian Software Engineering Conference, pp. 412–421. IEEE Computer Society (2008)
14. Wu, C., Potdar, V., Chang, E.: Latent semantic analysis – the dynamics of semantics web services discovery. In: Dillon, T.S., Chang, E., Meersman, R., Sycara, K. (eds.) Advances in Web Semantics I. LNCS, vol. 4891, pp. 346–373. Springer, Heidelberg (2009)

Chapter 20
Detection of Changes in Group of Services in SOA System by Learning Algorithms

Ilona Bluemke and Marcin Tarka

Abstract. The objective of this chapter is to present the detection of some anomalies in SOA system by learning algorithms. Special model of SOA system was designed and implemented for the experimental purpose. In this systems several anomalies were introduced. The detection of one of them i.e. changes in the frequency of a group of services is presented. Three learning algorithms: Kohonen network, emerging patterns and k-means clustering were used in the anomalies detector. The effectiveness of algorithms in detecting anomaly were measured. The results of experiments may be used to select an efficient algorithm for the detection of anomaly.

20.1 Introduction

Nowadays many system are based on SOA [1, 2] idea. A system based on a SOA provides functionality as a suite of interoperable services that can be used within multiple, separate systems from several business domains. SOA also provides a way for consumers of services, such as web-based applications, to find a service.

The objective of this chapter is to present the detection of changes in frequency of group of services in SOA system by learning algorithms. Our goal was to find the most appropriate algorithm to detect this kind of anomaly. Changes in frequency of a group of services often happen in real systems. It may happen if usability of some system service changes e.g. caused by changes of input data or routing polices. Information about changes in the frequency of group services may be very useful in optimization e.g. if services are operating on a cluster of processors. If this anomaly concerns the part of the system communicating with external system it may be also useful for the cost estimation and for the maintenance team.

Ilona Bluemke · Marcin Tarka
Institute of Computer Science, Warsaw University of Technology,
ul. Nowowiejska 15/19, 00-665 Warszawa, Poland
e-mail: i.bluemke@ii.pw.edu.pl

A. Zgrzywa et al. (eds.), *New Research in Multimedia and Internet Systems*,
Advances in Intelligent Systems and Computing 314, DOI: 10.1007/978-3-319-10383-9_20

The most appropriate algorithm should efficiently detect anomaly and generate output containing information about the services taking part in the anomaly.

The detection of another anomaly – a change in the frequency of single service call, we presented in [3] and in [4] we described the abilities of learning algorithms to detect unused functionality in a SOA system.

Related work is presented in section 20.2. Special model of SOA system was built for experiments. In this systems several anomalies were injected and three algorithms: k-means clustering [5], emerging patterns [6,7] and Kohonen networks [8] were used to detect them. In sections 20.3 and 20.4 the detection of changes in the frequency of group of services in SOA system is presented and some conclusions are given in section 20.5.

20.2 Related Work

There are many anomaly detection algorithms proposed in the literature that differ according to the information used for analysis and according to techniques that are employed to detect deviations from normal behavior. Lim and Jones in [9] proposed two types of anomaly detection techniques based on employed techniques: the learning and the specification model.

The *learning* approach is based on the application of machine learning techniques, to automatically obtain a representation of normal behaviors from the analysis of system activity. The *specification-based* approach requires that someone manually provides specifications of the correct behavior. Approaches that concern the model construction are presented in Fig. 20.1.

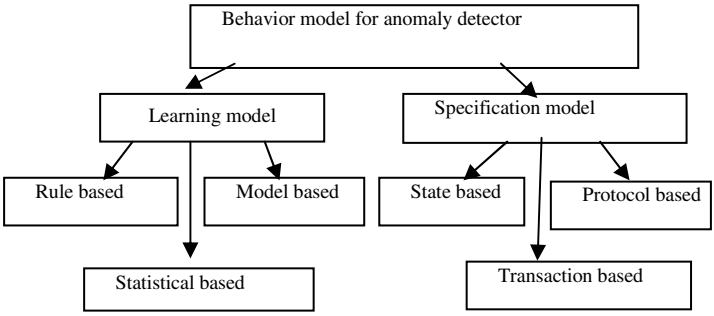


Fig. 20.1 Taxonomy of anomaly detection behavioral model (based on [9])

The *specification* approach depends more on human observation and expertise than on mathematical models. It was first proposed by C. Ko et. al. [10] and uses a logic based description of expected behavior to construct a base model. This specification-based anomaly detector monitors multiple system elements, ranging from application to network traffic.

In the *protocol based* approach the model of a normal use is built from the protocol specification e.g. TCP/IP. Lemonnier [11] proposed a protocol anomaly filter able to specifically analyze a protocol and model the normal usage of a specific protocol. This technique can be seen as a filter looking for protocol misuse. Protocol could be interpreted as any official set of rules describing the interaction between the elements of a computer system.

In networks often some events must take place at specified times so many protocol anomaly detectors are built as *state* machines [12]. Each state corresponds to a part of the connection, such as a server waiting for a response from client. The transitions between states describe the legal and expected change between states. State based model may be also used to describe intrusion attacks e.g. Z. Shan et. al. [13].

In the *transaction* based approach the regular behavior is formally described. The desired action and sequence of actions are specified by the definition of transactions. Such explicit definition makes the transaction an integral part of security policy. In the research proposed by R. Buschkes et. al. [14] the detection of anomalies is based on the definition of correct transactional behavior.

The *learning* model must be trained on the specific network. In the training phase, the behavior of the system is observed and logged, machine learning techniques are used to create a profile of normal behaviors. In the process of creating an effective anomaly detection model: *rule-based*, *model-based*, and *statistical-based* approaches have been adopted to create the baseline profile.

Rule-based systems used in anomaly detection describe the normal behavior of users, networks and/or computer systems by a set of rules. These predefined rules typically look for the high-level state change patterns observed in the audit data.

SRI International's Next-generation Intrusion Detection Expert System (NIDES) [15] is a statistical algorithm for the anomaly detection, and an expert system that encodes known intrusion scenarios. A variant of incorporating expert system in anomaly detection is presented in [16]. Owens et. al. presented an adaptive expert system for the intrusion detection based on fuzzy sets.

Model based anomaly detector models intrusions at a higher level of abstraction than audit records in the rule based approach. It restricts execution to a precomputed model of the expected behavior. Many researchers used different types of models to characterize the normal behavior of the monitored system like *data mining*, *neural networks*, *pattern matching, etc.* to build predictive models.

In the *data mining* approach, models are automatically extracted from a large amount of data. Current intrusion detection system requires the frequent adaptation to resist to new attacks so data mining techniques, that can adaptively build new detection models, are very useful. An example of data mining system was proposed by Lee et. al. and is presented in [17]. *Neural network* is trained on a sequence of information, which may be more abstract than an audit record. Once the neural net is trained on a set of representative command sequences of a user, the net constitutes the profile of the user, and the fraction of incorrectly predicted events then measures the variance of the user behavior from his profile. Kohonen networks used to detect anomalies in information system are described in [18].

There are several libraries supporting building neural networks e.g. [19–21]. Other neural networks based systems for intrusion detection are described in [18, 22–24]. In the *pattern matching* approach learning is used to build a traffic profile for a given network. Traffic profiles are built using features such as e.g. the packet loss, the link utilization and the number of collisions. Normal behavior is captured as templates and tolerance limits are set, based on the different levels of the standard deviation. The usage of emerging patterns in anomaly detection is described in [6].

The first *statistical based* anomaly detection was proposed by Denning and Neumann [25] in 1985. The anomaly detector observes subjects and generates profiles for them that represent their behavior. Widely known techniques in statistics can often be applied; e.g. data points that lie beyond a multiple of the standard deviation on either side of the mean might be considered as anomalous. There are many statistical techniques like *Bayesian statistics*, *covariance matrices and Chi-square statistics* [26] for the profiling of anomaly detection. Example of *Chi-square statistic* in anomaly detection is described in [27].

Commercial products applying rule based, statistical based, model based and neural networks approach to detected anomalies (known till 2008) are briefly described in [9].

20.3 Experiment

We were not able to detect anomalies in real SOA system so a special system (called VTV - virtual television) was implemented. The system simulates a real SOA system and enables to inject typical anomalies into its regular operation. More details describing the design and the implementation of this system can be found in [28]. Three learning algorithms were used in the anomalies detector: *Kohonen network*, *emerging patterns* and *k-means clustering*. These algorithms are using text input files prepared from logs of the VTV system. In this log file information like the number of request, the name of process, the name of service called and the execution time are written. These logs files are then processed by scripts, implemented in R [29] environment into transactions (for emerging patterns algorithm) or summarized reports (for *k-means* and *Kohonen* networks). The implementation of *k-means* and *Kohonen* networks [8] algorithms was made in R environment while the *emerging patterns* algorithm, which is more complicated, was implemented in C++. *k-means* algorithm was prepared based on [5], for *emerging patterns* algorithm ideas from [6,7] were applied.

Anomalies detection is a kind of clustering with two types of clusters grouping normal and abnormal behaviors. Correctly identified anomaly is an abnormal event which was introduced by purpose and was assigned to the abnormal cluster. Identified as anomalies other events or not recognized anomalies are treated as errors. The values measured in experiments are:

- FP (false positive) – number of incorrectly identified anomalies,
- FN (false negative) – number of not recognized anomalies,
- TP (true positive) – number of correctly identified normal behaviors,
- TN (true negative) – number of incorrectly identified normal behaviors.

Good anomalies detector should generate small number of errors. To compare the quality of detectors often sensitivity and specificity measures are used. The sensitivity is defined as the ratio of TP to the sum of TP and FN. Specificity is defined as ratio of TN to the sum of TN and FP.

The relation between specificity and sensitivity – ROC (Receiver Operating Characteristics) curve [30] can be used to compare the quality of models.

The experiment was conducted as follows. Firstly, data for regular behavior were created. Next, for each of the test case scenario, data for the abnormal behavior in this scenario were created. For each algorithm the learning phase, using regular data, was performed and the detection phase on data for abnormal behavior was executed. The quality of detection was calculated and the algorithms were compared.

In [3] the results of detecting the *change in frequencies of a service calls* was described. The worst in the detection of this kind of anomaly was the *k-means* clustering algorithm and the best was *emerging pattern* algorithm. Kohonen algorithm [5] also produced quite good results. In [4] the results for the scenario "*unused functionality*" detected by *k-means* clustering and *Kohonen* network were presented. *k-means* clustering and *Kohonen* networks were able to satisfactorily detect unused functionality. Below the results of detection of *changes in the frequency of group of services* is presented.

20.4 Results of Experiment

Changes in the frequency of several services often happen in real SOA systems. They may be caused e.g. by changes of the usability of some system service, changes of routing polices. Information about such change may be very useful in the optimization e.g. if services are operating on a cluster of processors it allows to evenly distribute them. If this anomaly concerns the part of the system communicating with external system it may be also useful for the cost estimation and for the maintenance team.

The goal of our experiment was to check if learning algorithms are able to detect changes in frequencies of several services. In our experiment the frequency of one service (A courier delivery) was increased and the second (delivery by B courier) one was decreased. Other goal of the experiment was to evaluate the output generated by algorithm, it was important if from this output the services taking part in the anomaly can be identified.

k-means Algorithm

k-means clustering [7] is a clustering analysis algorithm that groups objects based on theirs feature values into *k* disjoint clusters. Objects that are classified into the same cluster have similar feature' values. *k* is a positive integer number specifying

the number of clusters, and has to be given in advance. All object are assigned to their closest cluster according to the ordinary Euclidean distance metric. The results for *k-means* algorithm are presented in Table 20.1 and the ROC curve for k=3 and the number of training logs=150, are shown in Fig. 20.2. In point A (Fig. 20.2) the number of false alarms was zero but the specificity was only 0.61. Point B is optimal, in it the costs of false qualification of anomaly and false qualification of normal behavior are equal, sensitivity= 0.83 and specificity= 0.87. In point C the sensitivity was maximal (0.89 for *maxL*=2.5) but the specificity= 0.6 was low. The quality of detection is similar to the detection of changes of frequency of one service described in [4].

Table 20.1 Results for *k-means* algorithm

k	logsize	maxL	TP	TN	FP	FN	Sensitivity	Specificity
4	100	2.5	20	15	7	6	0.77	0.68
4	100	3	18	19	3	8	0.69	0.86
4	100	3.5	14	19	3	12	0.54	0.86
4	100	4	4	22	1	22	0.15	0.96
4	150	3	15	9	6	3	0.83	0.6
4	150	3.5	13	12	3	5	0.72	0.8
4	150	4	12	15	0	6	0.67	1
4	150	2.5	11	7	8	7	0.61	0.47
3	150	1.5	16	1	14	2	0.89	0.07
3	150	2	16	6	9	2	0.89	0.4
3	150	2.5	16	9	6	2	0.89	0.6
3	150	3	15	13	2	3	0.83	0.87
3	150	3.5	14	12	3	4	0.78	0.8
3	150	4	14	14	1	4	0.78	0.93
3	150	4.5	11	15	0	7	0.61	1
4	200	2.5	12	5	6	1	0.92	0.45
4	200	3	10	5	6	3	0.77	0.45
4	200	3.5	12	8	3	1	0.92	0.73

The output information generated by k-means algorithm is insufficient in the detection which services were taking part in the anomaly. *k-means* algorithm can detect changes in the frequency of group of services but can not identify processes with changed behavior. If the summarization is prepared for the whole system not for a process, the result from this algorithm “that the system does not behave normally” will not satisfy the maintenance staff.

Kohonen Networks

The results for Kohonen networks are shown in Table 20.2. High numbers of incorrectly identified normal behaviors (TN true negative) can be seen, which results also in high values of the specificity. Numbers of incorrectly identified anomalies (FP) for Kohonen network (Table 20.2) are significantly greater than for *k-means* algorithm (Table 20.1). In Kohonen network algorithm normal distribution is assumed but the probability of service used in the experiment was not normal.

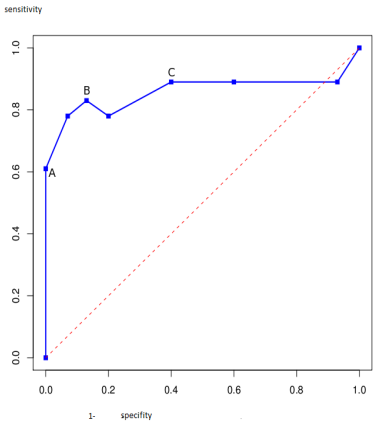


Fig. 20.2 ROC curve for *k-means* algorithm

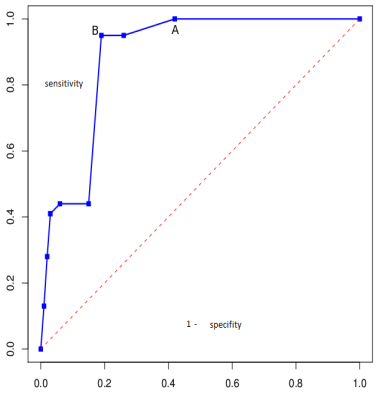


Fig. 20.3 ROC curve for Kohonen networks algorithm

Kohonen network algorithm examines each service independently. Others algorithms used in our experiment, are able to detect dependency between several entities (in *k-means* algorithm by distance function, in emerging patterns by association rules).

In Fig. 20.3 the ROC curve obtained for 150 training logs is shown. In point A the sensitivity is one but the specificity equals only 0.58. This result is worse than the result obtained in the detection of changes of frequency of single service we described in [3]. Point B is optimal, with the sensitivity= 0.95 and specificity= 0.81, the costs of false qualification of anomaly and false qualification of normal behavior are equal.

Table 20.2 Results for Kohonen networks

N	lambda	logsize	TP	TN	FP	FN	Sensitivity	Specificity
25	1	100	43	587	229	5	0.9	0.719
25	2	100	17	783	33	31	0.35	0.960
25	3	100	10	802	14	38	0.21	0.983
25	4	100	5	809	7	43	0.1	0.991
25	0.3	150	22	317	227	0	1	0.58
25	0.5	150	21	405	139	1	0.95	0.74
25	0.7	150	21	443	101	1	0.95	0.81
25	1	150	14	465	79	18	0.44	0.85
25	1.5	150	14	513	31	18	0.44	0.943
25	2	150	13	529	15	19	0.41	0.972
25	2.5	150	13	5529	15	19	0.41	0.972
25	3	150	9	534	10	23	0.28	0.982
25	3.5	150	4	537	7	28	0.13	0.99
20	2	150	13	528	16	19	0.41	0.971
15	2	150	13	529	15	19	0.41	0.972
25	1	200	19	320	54	3	0.86	0.856
25	2	200	12	361	13	10	0.55	0.965
25	3	200	9	364	10	13	0.41	0.973

The output from Kohonen network algorithm is shown in Fig. 20.4. Information is associated with each service, so if anomaly is detected the list of services taking part in this anomaly can be produced.

```
Anomaly by service 33: 0.01+−0.006765634<>0.015
Anomaly by service 34: 0.068+−0.02138631<>0.075
 service 35: 0.07+−0.02207476 IN 0.075
Anomaly by service 36: 0.012+−0.007174009<>0.015
Anomaly by service 1: 0.005+−0.008849663<>0.01
Anomaly by service 2: 0.04+−0.0126581<>0.05
```

Fig. 4 Output from Kohonen network algorithm

Emerging Patterns

Emerging patterns (EP) are defined as item-sets whose supports increase significantly from one dataset to another. EP can capture emerging trends over time or useful contrasts in data sets. EP are item-sets whose growth rates the ratios of the two supports are larger than a given threshold. EP algorithm is working on transactions obtained from the monitored system and containing names of services.

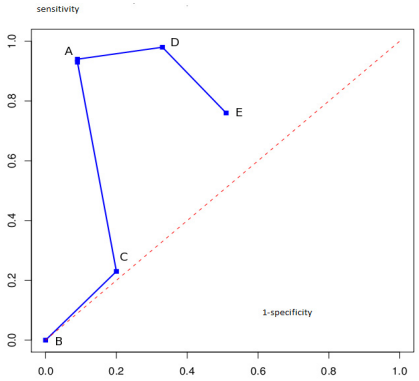


Fig. 20.5 ROC curve for emerging patterns algorithm

In Fig. 20.5 the ROC curve for emerging patterns algorithm is shown. In point D the sensitivity is maximal, undetected anomalies were only 2 but the specificity was only 0.67. This values were obtained for parameter *cSup*=0.1. Further decrease of this parameter will result in the decrease of the specificity. Points B and C were obtained for high values of the pattern support *cSup*, which caused small number of generated patterns and the decrease of sensitivity. Points D and E were obtained for low values of pattern support so many patterns were generated and the specificity decreased, "overtraining" was observed, patterns were too close to the training data. Exemplary output produced by the emerging pattern algorithm in our system is given in Fig. 20.6. Such detailed information, containing names of services, may be very useful for the maintenance team.

```
Anomaly by pattern: {processTechTask}{processError}
 with score 0.15 IN transaction 1

No anomaly in transaction 2

Anomaly by pattern: {courierASystemOrder}{courierASystemResponse}
with score 0.37 IN transaction 3

Anomaly by pattern: {courierASystemOrder}{courierASystemResponse}
{courierASystemFail}
with score 0.11 IN transaction 3
```

Fig. 20.6 Output from emerging patterns algorithm –patterns with anomalies

20.5 Conclusions

In this chapter some results of the detection of one anomaly – *changes in the frequency of group of services* in SOA system are presented. The experiment was conducted in special environment built to introduce several types of anomalies, detect them and measure. Experiment shows that all three algorithms were successful in the detection of this anomaly. *k-means* algorithm, the worst one in the detection of anomaly in single service [3], significantly improved its results. However emerging pattern algorithm, the best in the detection of anomaly in single service [3], did not improved its results but still was very successful and showed the best settings (specificity, sensitivity).

The goal of our experiment was also to examine the suitability of the output produced by each algorithm in identifying services taking part in the anomaly. Very useful is the information generated by emerging pattern algorithm and Kohonen networks. The output from *k-means* algorithm contains only alert that the anomaly was detected but does not provide the names of services in the anomaly.

The results presented in this chapter and in [3, 4] show that in anomaly detection in SOA systems different algorithms may appear most suitable for different type of anomaly so further research in this field should be conducted.

References

1. BPEL Standard, `http://docs.oasis-open.org/wsbpel/2.0/wsbpel-v2.0.html` (access July 2011)
2. SOA manifesto, `http://www.soa-manifesto.org` (access July 2011)
3. Bluemke, I., Tarka, M.: Detection of anomalies in a SOA system by learning algorithms. In: Zamojski, W., Mazurkiewicz, J., Sugier, J., Walkowiak, T., Kacprzyk, J. (eds.) Complex Systems and Dependability. AISC, vol. 170, pp. 69–85. Springer, Heidelberg (2012)
4. Bluemke, I., Tarka, M.: Learning algorithms in the detection of unused functionalities in SOA systems. In: Saeed, K., Chaki, R., Cortesi, A., Wierzchoń, S. (eds.) CISIM 2013. LNCS, vol. 8104, pp. 389–400. Springer, Heidelberg (2013)
5. Munz, G., Li, S., Carle, G.: Traffic anomaly detection using k-means clustering, Wilhelm Schickard Institute for Computer Science. University of Tuebingen (2007)

6. Ceci, M., Appice, A., Caruso, C., Malerba, D.: Discovering emerging patterns for anomaly detection in network connection data. In: An, A., Matwin, S., Raś, Z.W., Ślęzak, D. (eds.) ISMIS 2008. LNCS (LNAI), vol. 4994, pp. 179–188. Springer, Heidelberg (2008)
7. Guozhu, D., Jinyan, L.: Efficient mining of emerging patterns: discovering trends and differences. Wright State University, The University of Melbourne (2007)
8. Kohonen, T.: The self-organizing map. Proc. IEEE 78(9), 1464–1480 (1990)
9. Lim, S.Y., Jones, A.: Network anomaly detection system: the state of art of network behaviour analysis. In: Proceedings of the International Conference on Convergence and Hybrid Information Technology, pp. 459–465 (2008), doi:10.1109/ICHIT2008.249
10. Ko, C., Ruschitzka, M., Levitt, K.: Execution monitoring of security-critical programs in distributed systems: a specification-based approach. In: Proceedings of the IEEE Symposium on Security and Privacy, Oakland, CA, USA (1997)
11. Lemonnier, E.: Protocol Anomaly Detection in Network-based IDSs. Defcom white paper (2001)
12. Sekar, R., Gupta, A., Frullo, J., Shanbag, T., Tiwari, A., Yang, H., Zhou, S.: Specification-based anomaly detection: A new approach for detecting network intrusions. In: ACM Computer and Communication Security Conference, Washington, DC, USA (2002)
13. Shan, Z., Chen, P., Xu, Y., Xu, K.: A network state based intrusion detection model. In: Proceedings of the 2001 International Conference on Computer Networks and Mobile Computing, ICCNMC 2001 (2001)
14. Buschkes, R., Borning, M., Kesdogan, D.: Transaction-based anomaly detection. In: Proceedings of the Workshop on Intrusion Detection and Network Monitoring, Santa Clara, California, USA (1999)
15. Anderson, D., Frivold, T.: Valdes: a next-generation intrusion detection expert system (NIDES) (2005)
16. Owens, S., Levary, R.: An adaptive expert system approach for intrusion detection. International Journal of Security and Networks 1(3-4), 206–217 (2006)
17. Lee, W., Stolfo, S.J.: Data mining approaches for intrusion detection. In: Proceedings of the 7th USENIX Security Symposium (1998)
18. Bivens, A., Palagrini, C., Smith, R., Szymański, B., Embrechts, M.: Network-based intrusion detection using neural networks. In: Proceedings Intelligent Eng. Systems through Neural Networks, St. Louis, MO, vol. 12, pp. 579–584. ASME Press, NY (2002)
19. C Neural network library, `http://franck.fleurey.free.fr/NeuralNetwork/`
20. NeuroBox, `http://www.cdrnet.net/projects/neuro/`
21. Fast Artificial Neural Network Library, `http://sourceforge.net/projects/fann/`
22. Ryan, J., Lin, M., Miikkulainen, M.: Intrusion detection with neural networks. In: Advances in Neural Information Processing Systems, vol. 10, pp. 943–949 (1998)
23. Ghosh, A.K., Schwartzbard, A.: A study in using neural networks for anomaly and misuse detection. In: Proceedings of the 8th USENIX Security Symposium, Washington, D.C., USA (1999)
24. Han, S.J., Cho, S.B.: Evolutionary neural networks for anomaly detection based on the behavior of a program. IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics 36(3), 559–570 (2005)

25. Denning, D., Neumann, P.: Requirements and Model for IDES-A Real-Time Intrusion-Detection Expert System. SRI Project 6169, SRI International, Menlo Park, CA (1985)
26. Masum, S., Ye, E.M., Chen, Q., Noh, K.: Chi-square statistical profiling for anomaly detection. In: Proceedings of the 2000 IEEE Workshop on Information Assurance and Security (2000)
27. Ye, N., Chen, Q.: An anomaly detection technique based on a chi-square statistic for detecting intrusions into information systems. Quality and Reliability Engineering International 17(2), 105–112 (2001)
28. Tarka, M.: Anomaly detection in SOA systems. Msc Thesis, Institute of Computer Science, Warsaw University of Technology (2011) (in Polish)
29. The R Project for Statistical Computing, `http://gcc.gnu.org/`(access September 2011)
30. Hanley, J.A.: Receiver operating characteristic (ROC) methodology: the state of the art. Crit. Rev. Diagn. Imaging 29(3), 307–335 (1989)

Chapter 21
Modification of Page Rank Algorithm for Music Information Retrieval Systems

Zygmunt Mazur and Konrad Wiklak

Abstract. The chapter describes modifications of the Google PageRank as a ranking algorithm for music search engines. An assessment of its suitability for a music information retrieval systems has been made. The authors present new MusicPageRank algorithm which is based on existing connections – links – between websites and music files in similar way like PageRank. By using these connections, the new method of creating the music search engine results ranking has been developed.

21.1 Introduction

Although music information retrieval is currently a well-known domain, it is still a subject of much research. The popular text-based MP3 search engines are now replaced by systems that provide many different ways of creating queries, like query by humming, writing a sequence of notes or providing the melodic contour of a song.

Sound files are a specific data source. Both audio signal frequencies, stored in a sound file, and an additional textual information, stored in metadata, allow performing different types of queries. The term metadata means textual information extracted from a sound file, like song name, composer, name of the album etc. By using one of the online available music databases, e.g. GraceNote or MusicBrainz, the correctness of the audio metadata can be verified. Creating identification system of text information, that is stored in music files, by ourselves is a relatively complex and difficult task. It is also not a subject of this chapter. However, it is

Zygmunt Mazur · Konrad Wiklak
Institute of Informatics,
Wrocław University of Technology
Wyb. Wyspiańskiego 27, 50-370 Wrocław, Poland
e-mail: {Zygmunt.Mazur,Konrad.Wiklak}@pwr.wroc.pl

A. Zgrzywa et al. (eds.), *New Research in Multimedia and Internet Systems*,
Advances in Intelligent Systems and Computing 314, DOI: 10.1007/978-3-319-10383-9_21

possible to relatively simple to check the completeness of basic information about song, like title, artist name, album, year of album release, or copyrights. The correctness and completeness of data, that is stored in an audio file as metadata, is now the major criterion of sorting query results, returned by a music search engine. Such an approach of positioning search results in modern music search engines has significant advantages – it gives certainty that file returned as a result contains a song corresponding to metadata, which is stored in that file. However, it is not difficult to see disadvantages of using only music files itself in results ranking algorithm. Each music file is treated as a single entity that is not related with any other of music files, documents and other contents in the Internet. Unfortunately such an approach does not correspond to the real state and the information about the file's overall popularity in the Web is permanently lost. The popularity of a music file depends not only on the individual preferences of an search engine user, but also on the popularity of websites that host that file. The first possible approach of calculating music file popularity is the number of websites that contain at least single outgoing link, pointing to specified music file. However each website has an equal importance, which means that any website from the Internet is equally popular. Therefore, a better solution is to use the music websites popularity based on classic links-based algorithms for text documents. Such algorithms are used by modern search engines like Google. The overall sum of weights – or the maximum weight – of websites return by the text documents ranking algorithm is a rank of the music file context.

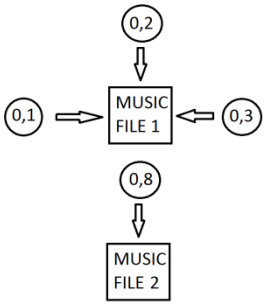


Fig. 21.1 The music file popularity depends on overall websites popularity

This chapter assesses the usability of selected link-based results ranking algorithms, which are used in textual search engines, for the possibility of using them in the Internet music information retrieval systems. It also presents a modification of the original PageRank algorithm due to the specificity of music files as a source of data. The most important part of the chapter is the presentation of the authors concept algorithm – which is based on the PageRank – the MusicPageRank.

21.2 Modifications of the Websites Ranking Algorithms in the Scope of Music Information Retrieval

At the end of the nineties – the time when HITS and PageRank algorithms were invented – users used to navigate through the Internet structure by using mainly links between websites. Contemporary search engines, were only basic help in finding websites related with the user-specified subject. The main navigation information source were hyperlinks between websites. The original versions of algorithms, based on an adjacency matrix of websites – like HITS, SALSA and PageRank – are a practical application of this concept. Now, after several years, the behaviour of the Internet users has changed. Today user, who is searching for specified information, still uses a search engine, but if the user notices that a website does not contain the interesting subject and does not contain any links to other websites related to the interesting topic, the user visits next website from the search engine's results list. That illustrates decreasing importance of the hyperlinks and the main role of search engine in modern Web navigation.

Therefore a question arises: how the behaviour of modern user can be applied in the modification of the classic link-based result ranking algorithms? How can it be applied into music search engines? Intuitively, it can be assumed that the rank for a music file is the maximum value of the weight function values returned by the ranking algorithm. The weights values are collected from all of the websites containing at least one link to this specified music file. An example of this idea application is illustrated in Fig. 21.1. In Fig. 21.1 MUSIC FILE 2 has greater value than MUSIC FILE 1, because $max\{0,8\} > max\{0,3;\ 0,2;\ 0,1\}$. Another important issue is that many websites contain audio files, it is necessary to define another criterion, that determines whether a website is related with the music topic. One of such criterion could be the lower Tolerance Threshold (*TL*) for the number of music files that are referenced from the specified website. Usually it will be a value from 1 to 5 files. If a number of music files is lower than the tolerance threshold, the website that contains the outgoing links to these files is omitted in the ranking algorithm. That causes that the overall rank value of a music file which is referenced from that website decreases. Excluding websites that are not related with music subject allows to decrease the scale of the problem to solve. That is very important because the fact that in 2011 year the total number of websites indexed by Google exceeded 36 billion. Despite the technological hardware and software progress, the calculation of the dominant eigenvalue for a such large sparse matrix by using the power method is still a difficult task.

The structure of connections – links – between music websites can be described with a direct oriented graph of connections between websites, denoted as GM(V,E). Where GM – Music Graph, V – set of vertices (Websites), and E – set of directed edges (connections – links between Websites).

The process of creating the graph GM(V,E), containing the number of music files links higher than the tolerance threshold *TL,* can be described as follows:

$$For\ i = 1\ to\ Number\ of\ Websites\ in\ search\ engine$$

$$If\ card(O_i) > TL\ Then\ add\ i$$
$$-th\ Website\ to\ set\ of\ vertices\ V\ of\ graph\ GM(V,E)$$
$$End\ For$$
$$\forall_{i,j\in V}\ If\ \big(link(i,j)\big)\ Then\ add\ edge(i,j)\ to\ set\ of\ edges\ E\ graph\ GM(V,E)$$

The expression link(i, j) returns true if there exists a connection – link – from the i-th to j-th website and corresponds to the edge in the GM graph. The expression $card(O_i)$ describes the number of outgoing links from i-th website to different music files. In other words, $card(O_i)$ is the cardinality the set of outgoing links from i-th website to music files.

The GM(V,E) graph is used in creating Reduced Adjacency Matrix RAM for websites that contain outgoing links to music files. The direct creation of RAM matrix, which corresponds to the graph GM, can be described with the following code:

$$For\ i = 1\ To\ Number\ of\ Websites\ in\ search\ engine$$
$$If\ card(O_i) > TL\ Then$$
$$RAMIndex := RAMIndex + 1 \quad \text{//increase matrix size}$$
$$RAM[1..RAMIndex, RAMIndex] := 0 \quad \text{//add row containing zeros}$$
$$RAM[RAMIndex, 1..RAMIndex] := 0 \quad \text{//add column containing zeros}$$
$$Add\ i-th\ Website\ to\ RAM\ matrix$$
$$End\ If$$
$$End\ For$$
$$For\ i = 1\ To\ Number\ of\ Websites\ in\ RAM\ matrix$$
$$For\ j = 1\ To\ Number\ of\ Websites\ in\ RAM\ matrix$$
$$If\ \big(link(i,j)\big)\ Then\ RAM[i,j] := 1$$
$$End\ For$$
$$End\ For$$

Example 1

The Fig. 21.2 shows a directed graph of connections between websites. Only websites denoted with numbers 1,3,5,6 contain links to music files. In Fig 21.2 the nodes corresponding to these websites are denoted with letter M.

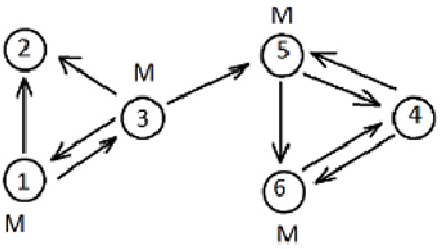


Fig. 21.2 Directed graph of connections between websites

The Adjacency Matrix AM for this graph is defined as follows:

$$AM = \begin{array}{c} \\ s1 \\ s2 \\ s3 \\ s4 \\ s5 \\ s6 \end{array} \begin{array}{c} \begin{array}{cccccc} s1 & s2 & s3 & s4 & s5 & s6 \end{array} \\ \begin{pmatrix} 0 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 & 0 \end{pmatrix} \end{array} \tag{21.1}$$

Let's assume that website s1 contains 5 links to music files, *s3* – 2 links, *s6* – 50 links and *s5* – 30 links. Assume that lower Tolerance Threshold is TL=3. During the construction process of the RAM matrix only *s1*, *s5* and *s6* websites are included, because website *s3* contains only 2 outgoing links to music files (TL>2) and websites *s2*, *s4* don't contain any links to music files. The matrix RAM (21.2) is presented as follows:

$$RAM = \begin{array}{c} \\ s1 \\ s5 \\ s6 \end{array} \begin{array}{c} \begin{array}{ccc} s1 & s5 & s6 \end{array} \\ \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{pmatrix} \end{array} \tag{21.2}$$

In case of user's search for music files with using the original algorithms (HITS, PageRank, SALSA) the RAM matrix better describes the real link structure between music Websites than AM matrix. Costs for music websites subject verification and hyperlinks structure analysis will quickly generate the profit as a reduction of the Adjacency Matrix size. The matrix (21.2), constructed from the websites from Example 1, corresponds to the graph presented in the Fig. 21.3.

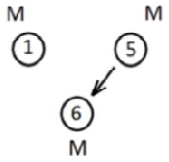


Fig. 21.3 Reduced directed graph of connections between websites

The modifications presented in this chapter can be used not only with websites that contain outgoing links to music files but also in rating websites that contain links to graphic or video files.

21.3 MusicPageRank

The Reduced Adjacency Matrix (2) can be used in the result ranking vector calculation using the same power method as the Google PageRank algorithm. However, the information how many music files are referenced from a website and how many other websites contain links to the same music file is completely lost. It is

only possible to verify that given website contains not less links to music files than the lower Tolerance Threshold TL.

In our proposed MusicPageRank (MPR) algorithm every element of the RAM matrix is replaced with the probability of the occurrence outgoing link on the *j*-th website to a music file:

$$MRAM_{i,j} = \frac{card(O_j)}{\sum_{k=1}^{n} card(O_k)} \cdot RAM_{i,j}$$

where $card(O_j)$ is the number of outgoing links from the j-th website to different music files. Many links to the same music file are treated as a single link. That number can be interpreted as the cardinality of the set outgoing links from the *j*-th website. The expression describes the number of links to different files from all websites that are included in the RAM matrix. In other words, this is the sum of cardinalities of the sets music file links that are located on all websites from the RAM matrix. The RAM matrix element that corresponds to the website in the *j*-th column is equal to the probability of occurrence a link to music file during the transition from the *i*-th to *j*-th website. Because the matrix RAM_{3x3} from Example 1 is too simple to show the transformations used in the MusicPageRank algorithm, in next example a matrix RAM_{6x6} (3) is used to present the MPR algorithm.

Example 2

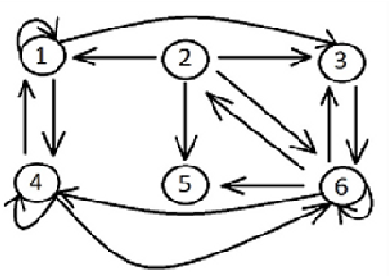


Fig. 21.4 An example of reduced directed graph of connections between websites

$$RAM = \begin{array}{c} \\ 1 \\ 2 \\ 3 \\ 4 \\ 5 \\ 6 \end{array} \begin{array}{c} \begin{array}{cccccc} 1 & 2 & 3 & 4 & 5 & 6 \end{array} \\ \begin{pmatrix} 1 & 0 & 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 & 1 & 1 \end{pmatrix} \end{array} \tag{21.3}$$

The cardinalities of links to different music files located on websites from the matrix (3) are equal as follows: $card(O_1)$=52, $card(O_2)$=38, $card(O_3)$=69, $card(O_4)$=66, $card(O_5)$=95, $card(O_6)$=91. Hence is calculated the sum of cardinalities of links to different music files located on all websites in the RAM matrix.

$$\sum_{k=1}^{6} card(O_k) = card(O_1) + card(O_2) + card(O_3) + card(O_4) + card(O_5) + card(O_6) = 52 + 38 + 69 + 66 + 95 + 91 = 411$$

The nonzero elements of the RAM matrix are replaced as follows:

$$MRAM_{1,1} = \frac{card(O_1)}{\sum_{k=1}^{6} card(O_k)} \cdot RAM_{1,1} = \frac{52}{411}$$
$$MRAM_{2,1} = \frac{card(O_1)}{\sum_{k=1}^{6} card(O_k)} \cdot RAM_{2,1} = \frac{52}{411}$$
$$MRAM_{2,3} = \frac{card(O_3)}{\sum_{k=1}^{6} card(O_k)} \cdot RAM_{2,3} = \frac{69}{411}$$

Performing similar transformations of all elements from the RAM matrix, the MRAM matrix is obtained:

$$MRAM = \begin{pmatrix} \frac{52}{411} & 0 & \frac{69}{411} & \frac{66}{411} & 0 & 0 \\ \frac{52}{411} & 0 & \frac{69}{411} & 0 & \frac{95}{411} & \frac{91}{411} \\ 0 & 0 & 0 & 0 & 0 & \frac{91}{411} \\ \frac{52}{411} & 0 & 0 & \frac{66}{411} & 0 & \frac{91}{411} \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & \frac{38}{411} & \frac{69}{411} & \frac{66}{411} & \frac{95}{411} & \frac{91}{411} \end{pmatrix}$$

For each row of the MRAM matrix we calculate the probability of occurrence of a link to a music file on websites that are not referenced by any other website from the MRAM matrix:

$$z_i = \frac{1 - \sum_{k=1}^{n} MRAM_{i,k}}{n}$$

The expression $\sum_{k=1}^{n} MRAM_{i,k}$ represents the sum of all elements in the *i*-th row of the MRAM matrix. The calculations z_i for the MRAM matrix from Example 2 are performed as follows:

$$z_1 = \frac{1 - (\frac{52}{411} + \frac{69}{411} + \frac{66}{411})}{6} = \frac{112}{1233}$$
$$z_2 = \frac{1 - (\frac{52}{411} + \frac{69}{411} + \frac{95}{411} + \frac{91}{411})}{6} = \frac{52}{1233}$$
$$z_3 = \frac{1 - \frac{91}{411}}{6} = \frac{160}{1233}$$
$$z_4 = \frac{1 - (\frac{52}{411} + \frac{66}{411} + \frac{91}{411})}{6} = \frac{101}{1233}$$

$$z_5 = \frac{1-0}{6} = \frac{1}{6}$$

$$z_6 = \frac{1-(\frac{38}{411}+\frac{69}{411}+\frac{66}{411}+\frac{95}{411}+\frac{91}{411})}{6} = \frac{26}{1233}$$

Every zero row, as we can observe in case z_5, is replaced with elements:

$$r_{zero} = \frac{\frac{\sum_{k=1}^{n} card(O_k)}{n}}{\sum_{k=1}^{n} card(O_k)} = \frac{1}{n}$$

where n is the size of the MRAM matrix (the number of all websites in the MRAM matrix).

Finally, the MPR matrix is constructed by adding probabilities z_i to every element of the i-th row of the MRAM matrix:

$$MPR_{i,j} = MRAM_{i,j} + z_i$$

The final form of the MPR matrix, constructed from the graph in Example 2, looks as follows:

$$MPR = \begin{pmatrix} \frac{268}{1233} & \frac{112}{1233} & \frac{319}{1233} & \frac{310}{1233} & \frac{112}{1233} & \frac{112}{1233} \\ \frac{208}{1233} & \frac{52}{1233} & \frac{259}{1233} & \frac{52}{1233} & \frac{337}{1233} & \frac{325}{1233} \\ \frac{160}{1233} & \frac{160}{1233} & \frac{160}{1233} & \frac{160}{1233} & \frac{160}{1233} & \frac{433}{1233} \\ \frac{257}{1233} & \frac{101}{1233} & \frac{101}{1233} & \frac{299}{1233} & \frac{101}{1233} & \frac{374}{1233} \\ \frac{1}{6} & \frac{1}{6} & \frac{1}{6} & \frac{1}{6} & \frac{1}{6} & \frac{1}{6} \\ \frac{26}{1233} & \frac{140}{1233} & \frac{233}{1233} & \frac{224}{1233} & \frac{311}{1233} & \frac{299}{1233} \end{pmatrix}$$

$$MPR = \begin{pmatrix} \frac{268}{1233} & \frac{112}{1233} & \frac{319}{1233} & \frac{310}{1233} & \frac{112}{1233} & \frac{112}{1233} \\ \frac{208}{1233} & \frac{52}{1233} & \frac{259}{1233} & \frac{52}{1233} & \frac{337}{1233} & \frac{325}{1233} \\ \frac{160}{1233} & \frac{160}{1233} & \frac{160}{1233} & \frac{160}{1233} & \frac{160}{1233} & \frac{433}{1233} \\ \frac{257}{1233} & \frac{101}{1233} & \frac{101}{1233} & \frac{299}{1233} & \frac{101}{1233} & \frac{374}{1233} \\ \frac{1}{6} & \frac{1}{6} & \frac{1}{6} & \frac{1}{6} & \frac{1}{6} & \frac{1}{6} \\ \frac{26}{1233} & \frac{140}{1233} & \frac{233}{1233} & \frac{224}{1233} & \frac{311}{1233} & \frac{299}{1233} \end{pmatrix}$$

Using z_i is possible to determine the exact value of teleportation factor α known from the classic Google PageRank algorithm version [3,4,10]. Using information about the number of unique outgoing links to music files from websites, α factor is simply defined individually for every row of the MPR matrix and the value of α doesn't have to be obtained experimentally.

To calculate the result MusicPageRank vector MPRV the iterative power method [2,9] is used. An initial MPRV vector is set with size *1* x *n* and every initial vector coordinate value is set from range [0,1]. The sum of initial vector coordinates has to be equal 1 (the start vector is a stochastic vector). Next, until the convergence of the MPRV vector, the following iterations are performed:

$$LastMPRV = MPRV$$
$$MPRV = MPRV \cdot MPR$$
$$IF \ \|MPRV - LastMPRV\|_1 > delta \ THEN \ continue \ ELSE \ end.$$

The result MusicPageRank vector (MPRV) is a vector of weights that are assigned to websites for calculating the overall music file rank. For the MPR matrix from Example 2 MPRV looks as follows:

$$MPRV = (0{,}14019490;\ 0{,}10865438;\ 0{,}16852394;\ 0{,}17572757;\ 0{,}16703498;\ 0{,}23986421)$$

This MPRV means that the highest weight value is assigned to sixth website, and lowest to second.

As comparison the result of PageRank for the same matrix presents as follows:

$$PageRank = (0{,}15185364;\ 0{,}09088946;\ 0{,}15512212;\ 0{,}18949969;\ 0{,}11020348;\ 0{,}29574886)$$

The main advantage of the MPR algorithm is omitted transformation of the MRAM matrix with the fixed α factor and using information about searched websites content – in this case music files. Because the structure of links between websites is important, the MPR algorithm gives better results than a simple vector of occurrence of a link to a music file on a website probabilities. It is also possible to use the MPR algorithm with an adjacency matrix of websites AM – in this case all links to websites that don't contain outgoing links to music files are replaced with zeros in the AM matrix. An advantage of using all websites is the original structure of links between websites. A disadvantage is assigning nonzero probability of occurrence outgoing links to music file also to websites that don't really contain such links. That causes decreasing the probability for websites that contain outgoing links to music files. Because of the bigger size of the matrix AM than RAM, computational and memory complexities also increase.

21.4 MusicPageRank – Tests

For the MPR algorithm tests have been performed for the number of iterations necessary to achieve the convergence of the result vector in the power method with specified calculation precision of the MPRV vector. The calculation precision means the computation accuracy set to specified number of decimal places. Tests have been performed on a single computer with 8GB of RAM memory. The tests included both original PageRank and MusicPageRank algorithms for the same sample matrices. The test program has been written in Java with using BigDecimal class to obtain higher calculation precision. Matrix for each test was filled randomly by using Random method.

The tests proved satisfactory convergence speed of the result MPRV vector for sparse matrices. The most important fact is that no method of result vector acceleration convergence [1], like quadric extrapolation, has been used. Results prove that the MPR algorithm can be applied into real music information retrieval

systems. Detailed results for 3 calculation precisions – 8, 30 and 100 decimal places are presented in the Fig.21.5 – 21.7.

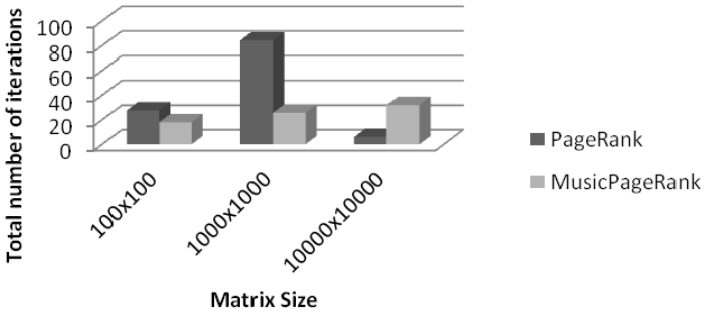


Fig. 21.5 Experimental results of the MPR and PageRank algorithms for fixed calculation precision 8 decimal places: number of iterations in power method for different matrix sizes and calculation precision set to 8 decimal places

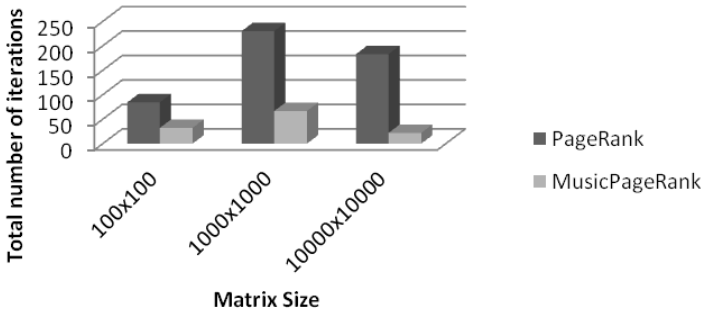


Fig. 21.6 Experimental results of the MPR and PageRank algorithms for fixed calculation precision 30 decimal places: : number of iterations in power method for different matrix sizes and calculation precision set to 30 decimal places

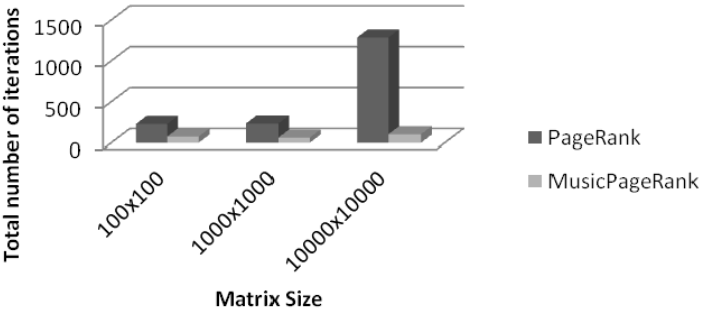


Fig. 21.7 Experimental results of the MPR and PageRank algorithms for fixed calculation precision 100 decimal places: : number of iterations in power method for different matrix sizes and calculation precision set to 100 decimal places

21.5 Summary

The chapter presented algorithm that provide context information about the music files. Information, that is based not on the music file itself, but on the structure of links between the websites. Results proved that existing and well tested link-based ranking algorithms, like PageRank, can be applied to another sources of information than text documents. The performed tests proved that the proposed in the chapter algorithm MusicPageRank is ready for practical applications in the domain music information retrieval. Using information about the number of outgoing links to unique music files on websites provides better projection of reality in the World Wide Web.

References

1. Czyżowicz, M.: Comparative study of intelligent navigation methods and adaptation of optimization algorithms to navigation tasks in text documents collections. Master Thesis, Warsaw University of Technology, Warsaw (2003) (in Polish)
2. Langville, A., Meyer, C.: A survey of eigenvector methods for Web Information Retrieval. SIAM (Society for Industrial and Applied Mathematics) Review 47(1), 135–161 (2005)
3. Langville, A., Meyer, D.: Deeper Inside Page Rank. Internet Mathematics 1(3), 335–380 (2004)
4. Langville, A., Meyer, C.: Google's PageRank and Beyond. The Science of Search Engine Rankings. Princeton University Press (2006)
5. Manning, Ch., Raghavan, P., Schutze, H.: Introduction to Information Retrieval. Cambridge University Press (2008)
6. Mazur, Z.: Properties of a model of distributed homogeneous information retrieval system based on weighted models. Information Processing & Management 24(5), 525–540 (1988)
7. Mazur, Z.: Models of a Distributed Information Retrieval System Based on Thesauri with Weights. Information Processing & Management 30(1), 61–78 (1994)
8. Mazur, Z., Wiklak, K.: Music information retrieval on the internet. In: Nguyen, N.T., Zgrzywa, A., Czyżewski, A. (eds.) Advances in Multimedia and Network Information System Technologies. AISC, vol. 80, pp. 229–243. Springer, Heidelberg (2010)
9. Meyer, C.: Matrix Analysis and Applied Linear Algebra. SIAM Society for Industrial and Applied Mathematics (2000)
10. Prystowsky, J., Gill, L.: Calculating Web Page Authority Using the Page Rank Algorithm. Math. 45 (2005)
11. Wiklak, K.: Internet Music Information Retrieval Systems. Master Thesis, Wrocław University of Technology, Wrocław (2010) (in Polish)

Chapter 22
Configuration of Complex Interactive Environments

Jędrzej Anisiewicz, Bartosz Jakubicki, Janusz Sobecki, and Zbigniew Wantuła

Abstract. In this chapter the problem of complex interactive environments configuration is presented. These systems are using various types of sensors such as touch, depth or RFID. First we present a few examples of digital signage applications delivered and then we propose their taxonomy. The presented taxonomy is used to prepare set of default configuration settings. Then we describe a proprietary solution called Interactive Control Environment that enables to configure complex interactive systems.

22.1 Introduction

Today modern digital interactive media are to be met everywhere. They are applied in many different areas, such as: entertainment, educational and advertisement applications and use technologies such as touch screens, kinetic interfaces, interactive walls and floors, augmenter reality, virtual reality and many others [2,16].

In this chapter we present the solutions that are partially based on the experiences of ADUMA, that since 2009 has delivered over 1000 solutions for many

Jędrzej Anisiewicz · Zbigniew Wantuła
Aduma SA, ul. Duńska 9, 54-427 Wrocław, Poland
e-mail: {j.anisiewicz,z.wantula}@aduma.pl

Bartosz Jakubicki
The Eugeniusz Geppert Academy of Art and Design,
ul. Plac Polski 3/4, 50-156 Wrocław, Poland
e-mail: bjak@asp.wroc.pl

Janusz Sobecki
Institute of Informatics, Wrocław University of Technology
Wyb. Wyspiańskiego 27, 50-370 Wrocław, Poland
e-mail: janusz.sobecki@pwr.wroc.pl

A. Zgrzywa et al. (eds.), *New Research in Multimedia and Internet Systems*,
Advances in Intelligent Systems and Computing 314, DOI: 10.1007/978-3-319-10383-9_22

different customers such as shopping malls, cinemas, museums and fair exhibitions. ADUMA developed several tools to make the implementations more efficient, such as: Designer to easily design interactive applications, Desktop for running the applications created in Designer and Interactive Control Environment (ICE) to configure sensors in the interactive environment.

These tools and technologies help to reduce costs of the installations by providing functionalities that enable easy and fast configuration of the installation in many differentiated environments. The localization of the installations may be quite distant from the provider's headquarters, so it is usually quite difficult and expensive to send there specialists. One way of the simplifying the configuration process is using specific templates.

The content of the chapter is the following. In the second paragraph we present several selected digital signage realizations that were delivered by ADUMA, we tried to show not only the interesting and latest but also other applications. The third paragraph presents the taxonomy of the ambient hybrid installations in architectural objects. In the fourth paragraph we describe ADUMA's ICE technology, its goals and general functionalities. The final paragraph summarizes the chapter and presents future works.

22.2 Overview of Applications

The examples presented in this chapter were selected from a wider set of projects and realizations delivered by ADUMA. We shall first describe an interactive installation set up in the Central Museum of Textiles in Lodz. This is the first installation where all elements of our proprietary system called ADS were used to design content and manage multiple interactive terminals. The second example is the most technologically advanced Polish multimedia installation – a 15 meters wide kinetic video wall. The installation was the first in the Central European region to use a complex network of depth sensors as an input.

In 2013, Central Museum of Textiles in Lodz (Poland) opened an interactive exhibition, in which, thanks to innovative multimedia, it's possible to move around in a virtual world of textiles and the methods of their production. The exhibition consists of more than 40 multimedia devices, including interactive floors, interactive 3D mappings, a virtual mannequin, multi touch tables and multimedia information totems. All multimedia are controlled by the ADS system, a proprietary exhibition control system created and owned entirely by ADUMA. The system ensures the correct functioning of each of the devices. A built-in mechanism sets up the schedule and enables for the devices to be automatically switched on and off at any given time. The entire set up is controlled from one computer via ADUMA Admin management component. Through this software, an administrator of the exhibition can freely configure each element of the multimedia exhibition. Moreover, thanks to a tracking mechanism, ADUMA Admin also provides information about the state of each of the devices controlled by the system and informs about any failures in real time, such as e.g. a damaged projector lamp or a

disrupted connection with one of the computers. Each of the devices is activated only when the motion sensors detect visitors approaching designated zones. The same mechanism informs the visitors that a multimedia show is about to start. When the visitors gather in a certain place, the show starts automatically. The show combines elements of classical projection, 3D mapping, a light show, music and other special effects. Once the show ends the system points the visitors in the direction of another attraction. The show is interrupted when no users are in the room. This system allows for smooth control of the flow of visitors and decreases the costs associated with the use of electricity and replacing parts. The Desktop software is installed on all touch terminals. Content designed in the Designer content creation software is seamlessly distributed to all touch terminals via ADUMA Admin. The Designer tool allows exhibitors to easily change the presented content, for example by occasionally adding new presentation material or by organizing special shows. For the first time, museums have full control and flexibility when it comes to the presented interactive exhibition content without incurring any additional costs.

a) b)

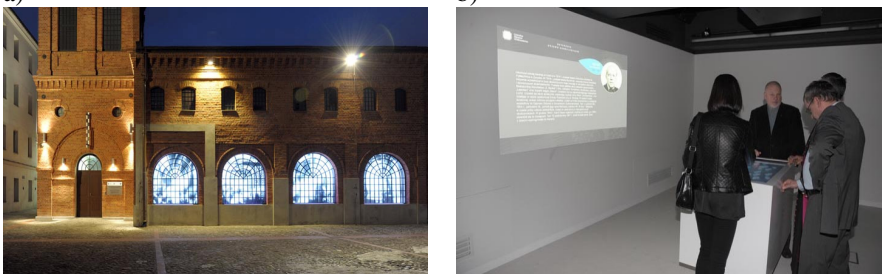


Fig. 22.1 a) Central Museum of Textiles in Łódź (Poland) b) Kinetic systems tracking in the office of a former owner of the factory

One of the biggest attractions of the described museum is a visit to the office of a former owner of the factory, where kinetic systems track the behavior of the visitors and adjusts the behavior of a projected figure of the factory owner accordingly by switching video sequences. The visitors fall under the impression that the projected figure interacts with them by gazing in the right direction, greeting and throwing occasional remarks.

The next attraction is a specially designed room where walls are used as projection surfaces for moving projectors while the entire floor is covered with projected images reacting to the movement of visitors passing through. Applications presented on the floor are integrated with a special sequence projected on the walls and this creates an environment where visitors witness a multimedia show, for example about textile manufacturing processes. The audio content transmitted from loudspeakers combined with the subject-specific video content presented in a non-standard way on the walls is further enhanced through special interactive scenarios

prepared for the visitors on the interactive floor. For example, while the role of water in the production process is being explained, the whole floor surface is covered with virtual water, on which one can walk producing waves and ripples. This effect was made possible by connecting 4 interactive floor systems and 3 projectors installed on mobile mounting solutions. The real challenge was to prepare such applications and templates as to best show the content to certain audience groups. Some of the devices can recognize if they are being used by children or adults and, depending on the user, display content prepared specifically for that age group. Novel multimedia solutions are not just used as additional eye-catchers, but are designed to educate in an interesting way and make a visit in the museum unforgettable. More than one hundred diverse applications were prepared for the purpose of this installation, including quizzes and games which deepen the understanding of the presented topic even further.

The most advanced installation by ADUMA company can be found in the Sky Tower shopping centre in Wroclaw, Poland (Fig. 22.2). To our knowledge, the installation constitutes the largest kinetic-controlled interactive video wall in the world. The content presented on the video wall reacts to human movements, can be used for playing games, learning about the building, as well as for interactive advertising and animations. The installation includes two separate walls, consisting of 45 and 15 seamless screens respectively. The entire installation is controlled by twelve computers. The larger wall, which reacts both to full body motion and to particular gestures, has a resolution of 9600 x 1080 pixels, that is five times higher than Full HD. The shopping mall visitors can use the interactive wall to play football, discover the latest fashion trends in a virtual fitting room or to interact with any other interactive scenario. In total, more than 30 engaging applications were prepared for the video wall. One of the biggest advantages of this system is the inherent flexibility in terms of content, as new applications can be easily added on regular basis to keep surprising the visitors. Every customer, no matter what age or interest, can find a suitable application to match their interests.

The dedicated interactive system is based on 45 integrated screens in a 15 x 3 arrangement using 46" LCDs, creating a large format video wall measuring 15.385 x 1.739 m. The system integrates two functionalities, offering the users interaction on two separate layers. The first layer covers the entire screen surface and creates a coherent interactive panorama. This layer reacts to the movement of the people passing in front of the screen within a designated 'interaction area', which is a 2 meter wide strip stretching from 1 meter away from the screen to three meters away. The second layer of interaction divides the screen into 5 separate modules of augmented interaction, each being 3 meters wide and 1,8 meters tall (9 screens in an a 3x3 arrangement). The modules use gesture detection to enable advanced interaction between an individual user and a particular segment of the screen. A person who stops on a designated spot within the interaction area activates the selected module and is then introduced to the application via a short animation serving as a user manual [11].

Fig. 22.2 Interactive wall in the SkyTower shopping centre, Wrocław

The system uses 5 sets of motion detectors, each of which is responsible for providing interaction for a different 3x3 screen module. Each set includes a single sensor installed directly under the screens (a special rectangular opening was cut out in a glass overlay to ensure full sensor effectiveness), while another sensor is placed in an aluminum and acrylic glass housing situated on the ceiling, directly above the interaction area. The ADUMA Designer and ADUMA Admin software described above were used to manage the entire installation. Both systems were adjusted so as to enable creating kinetic effects using templates and to allow for remote management and remote updating of the described interactive wall systems. The kinetic video wall in Sky Tower shopping center presented a serious of technological challenges in terms of understanding the environment and translating input from a whole network of sensors into interactive scenarios driven by stimuli provided by end users in real time.

22.3 Digital Signage Applications Taxonomy

Digital Signage Installations is the concept of information and advertising screens, as well as other digital communication solutions in public places, expanded by spatial actions and search for new forms and interaction with a user. At the base of that object category lie:

- optimal communicativeness,
- utilitarianism of solutions,
- synergic links of physical form with software,

- hybrid character of structure,
- localisation in architectonic spaces,
- phenomenon of attracting spectators,
- ambient character.

In case of that typology by the hybrid character the mixing of building materials of the works made of the information technology and the real space components is understood, with the purpose of creating a complementary entirety. This is some kind of augmented reality, though it does not precisely correspond to popular understanding of the said technology. It creates architectural and interior design components, as well as material objects with shared domains of computer equipment or the displayed virtual pictoriality.

Instead, the ambient specificity of that type of implementations manifests itself in their unique character, inventiveness, non-standard way, contextuality, mediality, as well as transiency. Variability happens to be one of the features of the modern design, not only the ambient one, both in product design with determined durability and in architectural design where stability, long life and strength ceased to be the leading virtues.

The zone of multimedia interior furnishing seems to lead to the extreme the ideas of temporary or current short-term solutions. The hybrids of digital equipment with architectonic infrastructure create almost incidental solutions. Each designed contact of digital technology with a particular place creates new contexts, opportunities and unique ideas of mutual support of those spheres. At the same time their life is only that long as the timeliness of the used applications and equipment and the attractiveness of a given multimedia technology.

Quite often, Digital Signage Installations engage architectural components such as walls, floors, ceilings, niches, pillars, doors, and sometimes activate only fragments of them [4]. They also commonly use the already established standards of interior furnishing. They only enrich the traditional forms of furniture, lighting and other components of arrangements with the new functionalities and new aesthetics. Such succession seems only natural because of the centuries-old legacy of anthropomorphic and humanist design. However, it is hard to omit the potential of the computer equipment platform implanted into the structure of interiors, introducing the domain of information to our environment through: being on-line, interaction systems, touch and movement interfaces, augmented reality, virtual reality, hypermedia, and finally the different from natural sensoriality and perception of the known components of furnishing.

As examples of hybrid objects already realised that are copying the traditional patterns could be: various panels and counters, arm chairs, chairs, shelves, showcases, kiosks, displays, pylons and totems, but also moving platforms, simulators, and even mobile robots. Many new multimedia forms are created which combine the existing utility models into complex objects e.g.: the winding ribbon coming out from a floor and entering into a table, later into a vertical media wall and further into a hanging ceiling, and all the components are integrated with the cooperating interactive projections. Such collection of objects is being expanded by

various multimedia and interactive technologies, which in any possible configuration may connect with the mentioned forms depending on physical, technical, economic, functional and aesthetic factors.

Among the three most frequently used forms of equipment or hybrid objects definitely the leading one is the standard monitor window in a form of a touch screen and its smaller mobile variations. The fact seems to indicate for preferences of versatility and easy access, as well as portability of communication equipment although it is certain that the mobile technology is not going to provide all of the functions and the need for contact with the environment. The supreme value of the group of hybrid objects is the variety, presence of mass solutions although received individually as in case of augmented reality, but also precisely dedicated to a specific place.

Unusual variability of solutions, dynamics of their impact and natural context referring to the place or the subject of presentation, make the hybrid objects noticed by spectators as novel, interesting attraction and diversification of existing forms of communication all the time. However, the utilitarian values and mass computerisation of our civilised environment will cause relatively fast shift of that ambient perception in the direction of serious technology of design [10].

One of the key issues differentiating hybrid products from traditional media is their integration with utilitarian forms, and the resultant change in communication types in the direction of more natural for humans than keyboard and mouse follows.

One of the required features of human environment is poly-sensory communication of user body and mind with the environment. In mixed objects a combination of traditional relations of environment and humans with multimedia activity of inanimate matter, telematics and the interactivity of technical equipment leading to digital communication took place. Implementation of digital data and non-physical content into the interior furnishing brings new phenomena and perception-psychic interactions.

The receptor relations provide information for human brain, due to which it can respond to stimuli coming from the environment. In the receptor relations the main receptors participate such as: sight, hearing, smell, and touch, but frequently equally important are complementary feelings e.g.: mechanoreceptors (gravity equilibrium, accelerations, low frequency vibrations), or the proprioceptors (joints movement and muscle tension) [5]. By suspending them between digital communication and physical objects, the hybrid objects give us chances of receiving both domains simultaneously. Whether they adopt the form of tactile furniture, interactive architecture components or the space supplemented by augmented reality, in each case they open new channels of perception for which the separate taxonomy of phenomena and utility will have to be created soon.

22.4 Interactive Control Environment

ICE is best described as a bridge between interactive applications and various sources of information about the environment in which those applications operate

(such sources as e.g. video captured by a camera or hand gestures made by an end user) [16]. Interactive systems include three major elements: sensors collecting data about the environment, algorithms which analyze this data and interactive applications, which make use of this data to provide interactive experiences (Fig. 22.3). Since there is a multitude of different sensors operating within different standards, and since it was necessary to use different libraries to analyze some of the data efficiently (e.g. data from RGB cameras) it was necessary to introduce an additional level of abstraction to the system.

ICE also have one additional output that provides overview data for the GUI that can be accessed by the person working with ICE. The graphic user interface allows the user to select and configure processors and select preview from particular processors. The example of the ICE application GUI together with sample process configuration is given in Fig. 22.4.

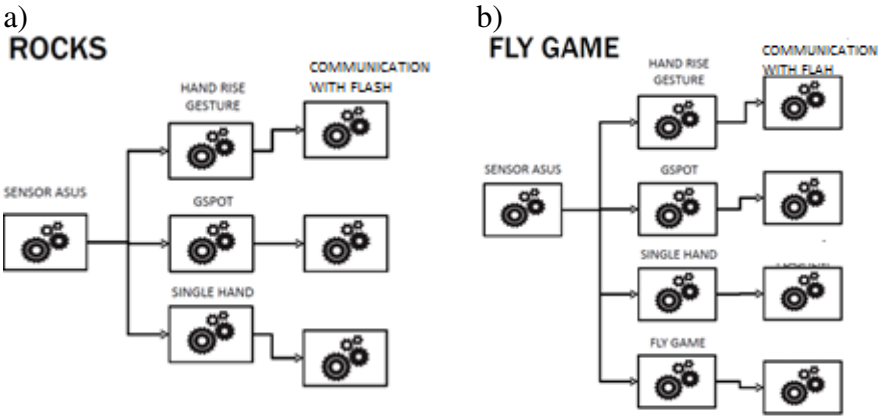


Fig. 22.3 ICE scenarios containing different processors for (a) Rocks application and (b) Fly game

It is also possible to start and stop selected processors. Different configurations may be saved as XML files. This provides unprecedented flexibility, e.g. if our interactive scenario is built using cameras from one OEM and we are forced to change the hardware to that provided by another OEM. In this case it is not necessary to rebuild the entire solution. Rather, only a single processor responsible for the video input needs to be switched.

ICE is an open environment and can be easily integrated with other solutions by creating appropriate plugins. At the moment we operate on input processors which work among others with such hardware solutions as Kinect, Asus Xtion, webcams, GigE cameras, as well as BlackMagic Design [12–15].

22.5 Discussion and Future Works

In this chapter we presented ICE that is the new sensor technology for interactive systems. The need of the development of this technology emerged from the practice of the development and implementation of more than 1000 solutions made by ADUMA. In the second paragraph we presented two selected digital signage realizations that were delivered by this firm. Then we have given the taxonomy of the ambient hybrid installations in architectural objects to present their diversity. We would like to show that we need also very different design and configuration approaches, as well we need different types of sensors to control such applications effectively. ICE is an open environment and can be easily integrated with other solutions by creating appropriate plugins. At the moment we operate on input processors which work among others with such hardware solutions as Kinect, Asus Xtion, webcams, GigE cameras, as well as BlackMagic Design.

The solution to the problems of configuration of interactive environments was the design and development of ICE technology together with specialized software tool for ICE configuration that enables to select the predefined sensors and set their configuration data values. However it has some limitations, one of which forms fill-in interaction style and the second the use of the specific inner format. To overcome this disadvantage an application of one of the BPM IT tools [7,9]

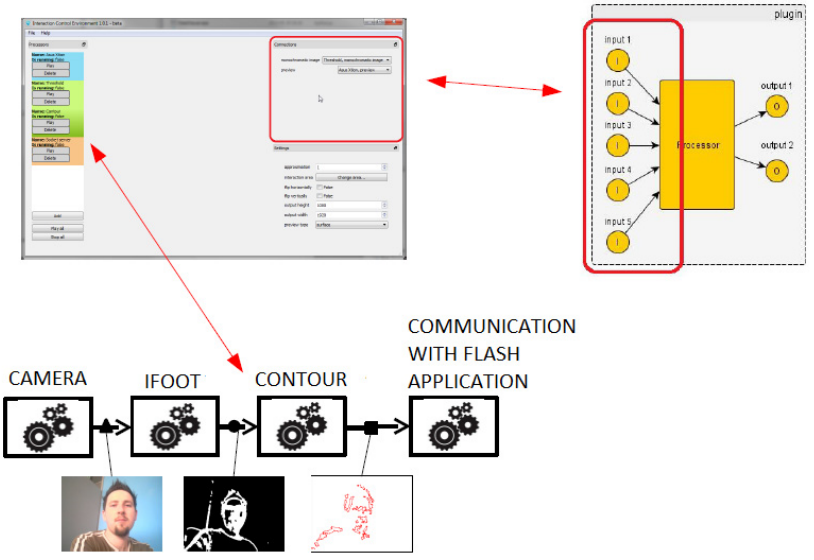


Fig. 22.4 Sample process configuration with ICE application GUI

that deliver full environment containing all of the system elements such as design, implementation, deployment, execution and control of the whole configured application, or only design and configuration of the ICE by means of graphical tool for composition of services. One example of the tool of the second type is Bizagi Process Modeler that enables to diagram, document and even simulate execution that is using standard Business Process Modelling Notation (BPMN) [1]. Using Bizagi we can not only to draw a diagram but also set the specific properties of the tasks (here processors). The task *properties* in Bizagi are divided into the following elements: *Basic*, *Extended*, *Advanced* and *Presentation Action*. In the *Basic* section we may specify: *Name*, *Description* and *Performers* (describes the resource that will perform the action). In the *Extended* section we give configuration values connected with each processor, for example by means of attributes and their values or extended files.

In the future we should decide whether to apply the existing solutions for the process modelling and execution or to implement our own solution based on the SOA paradigm [3,6]. That kind of more specialized tool will be maybe less flexible however more effective in the specified domain.

Acknowledgment. The research was partially supported by the European Commission under the 7th Framework Programme, Coordination and Support Action, Grant Agreement Number 316097, ENGINE – European research centre of Network intelliGence for INnovation Enhancement (http://engine.pwr.edu.pl/).

References

1. Cardoso, J., van der Aalst, W.: Handbook of Research on Business Process Modeling. Information Science Reference (2009)
2. Kopel, M., Sobecki, J., Wasilewski, A.: Automatic web-based user interface delivery for SOA-based systems. In: Bădică, C., Nguyen, N.T., Brezovan, M. (eds.) ICCCI 2013. LNCS, vol. 8083, pp. 110–119. Springer, Heidelberg (2013)
3. Grzech, A., Juszczyszyn, K., Kołaczek, G., Kwiatkowski, J., Sobecki, J., Świątek, P., Wasilewski, A.: Specifications and deployment of SOA business applications within a configurable framework provided as a service. In: Ambroszkiewicz, S., Brzeziński, J., Grzech, A., Zielinski, K. (eds.) Advanced SOA Tools and Applications. SCI, vol. 499, pp. 7–71. Springer, Heidelberg (2014)
4. Winkler, T.: Computer Aided Design of Antropotechnical Systems, pp. 11–14. Science and Technology Publishing House, Warsaw (2005)
5. Gallery-Dilworth, L.: How Digital Signage Technologies Are Transforming Interior Design and Architecture (2013), http://www.digitalsignageconnection.com/how-digital-signage-technologies-are-transforming-interior-design-and-architecture-jan-15-2013-654
6. Brodecki, B., Brzeziński, J., Dwornikowski, D., Kobusiński, M., Sasak, P., Szychowiak, M.: Selected aspects of management in SOA. In: Ambroszkiewicz, S., Brze-ziński, J., Cellary, W., Grzech, A., Zielinski, K. (eds.) SOA Infrastructure Tool – Concepts and Methods, pp. 83–116. Poznan University of Economy Press, Poznań (2010)

7. Business Process Management, http://www.gartner.com/it-glossary/business-process-management-bpm/ (downloaded November 2012)
8. Havey, M.: Essential Business Process Modeling. O'Reilly (2005)
9. Richardson, C., Miers, D.: The Forrester Wave™: BPM Suites, Q1 2013, http://public.dhe.ibm.com/software/solutions/soa/pdfs/forresterwave_bpmste.pdf (downloaded December 2013)
10. Austin, T., Doust, R.: New Media Design, pp. 128–132. Laurence King Publishing, London (2007)
11. MegaWall, K.: Wrocław, Poland (2013), http://www.ADUMA.pl/pl/produkty/kinetic-megawall
12. Kinect for Windows, http://www.microsoft.com/en-us/kinectforwindows/
13. Asus Xtion Pro, http://www.asus.com/Multimedia/Xtion_PRO/
14. Asus Xtion Pro LIVE, http://www.asus.com/Multimedia/Xtion_PRO_LIVE/
15. Leap Motion 3D Gesture Control for Windows & Mac, https://www.leapmotion.com/product
16. Cichoń, K., Sobecki, J., Szymański, J.M.: Gesture tracking and recognition in touchscreens usability testing. In: Marasek, K., Sikorski, M. (eds.) Proceedings of the International Conference on Multimedia, Interaction, Design and Innovation, MIDI 2013, pp. 1–8. ACM, New York (2013)

Chapter 23
A Shape-Based Object Identification Scheme in Wireless Multimedia Sensor Networks

Mohsin S. Alhilal, Adel Soudani, and Abdullah Al-Dhelaan

Abstract. Multimedia communication is highly attractive in Wireless Multimedia Sensor Networks (WMSN) due to their wealth of information's. However, the transmission of multimedia information such as image and video requires a specific scheme and an efficient communication protocol. In fact, the performances of multimedia based applications on WMSN are highly dependent on the capabilities of the designer to provide low-power data processing and energy-aware communication protocols. This chapter presents a contribution to the design of low complexity scheme for object identification using Wireless Multimedia Sensor Networks. The main idea behind the design of this scheme is to avoid useless multimedia data streaming on the network. In depth, it ensures the detection of the specific event (target) before sending image to notify the end user. The chapter discusses the capabilities of the proposed scheme to identify a target and to achieve low-power processing at the source mote while unloading the network. The power consumption and the time processing of this scheme were estimated for MICA2 and MICAZ motes and showed that it outperforms other methods for communication in WMSN such as the methods based on image compression.

23.1 Introduction

Recently, research in the area of WSN is more focusing on the idea of enhancing the capabilities of the WSN to provide the end user with useful information's gathered in a smart scheme instead of simply sending all the measurements to report

Mohsin S. Alhilal · Adel Soudani · Abdullah Al-Dhelaan
College of Computer and Information Sciences,
King Saud University
Saudi Arabia
e-mail: mhilal@student.ksu.edu.sa,
{asoudani,dhelaan}@ksu.edu.sa

A. Zgrzywa et al. (eds.), *New Research in Multimedia and Internet Systems*,
Advances in Intelligent Systems and Computing 314, DOI: 10.1007/978-3-319-10383-9_23

about a single event occurrence or information of interest. This idea becomes with big interest when talking about multimedia communication over Wireless Sensor Networks. In fact, a Wireless Multimedia Sensor Network (WMSN) is built using wireless sensor nodes that integrates multimedia devices such as image and audio sensors enabling to retrieve video or audio data streams. The use of these sensors provides the application with rich visual verification, in-depth awareness of the real scene and recognition with a lot of others interesting capabilities. As the multimedia applications are characterized of the production of relatively huge data streams, the power consumption appears as the major challenge to face when deploying the WSN for the transmission of multimedia information's. Since the power consumption is proportional to the number of bits to be transmitted that represents the multimedia information, then, as a first reflection, reducing this amount of data will help to reduce the power consumption. However, most of the research works concerning image compression in WMSN have noticed that classical compression methods for video and image are not suitable to be processed within weak hardware capabilities characterizing the wireless sensor nodes [11, 15].

From another approach, to minimize the amount of images transmitted through the WMSN, it will be recommended to first check if the captured image contains interesting information's about phenomena that interests the end user. Then, sending the minimum of bytes that represents the detected object or the physical phenomena will contribute to achieve low power consumption.
The efficiency of this method depends on the scheme used to extract the useful information from the video stream at the source node and its capabilities to detect the occurrence of a specific event. So, while this idea looks very attractive to keep a strategic balance between the local processing resources of the mote and power saving, we think that more focus is still required to design an efficient scheme that will be implemented in the multimedia sensor.

The main contribution of this chapter is to specify a low-complexity scheme to detect and to identify an object based on image processing and to notify it to the end user.

In the remaining part of the chapter we will first present the specified scheme for object detection and identification. Then we will discuss its performances to identify the target and it's invariance for different parameters. The end part of the chapter, will address the performances of this scheme when implemented on MICA2 and MICAZ motes.

23.2 Related Works and Motivation

WMSN(s) were deployed for remote object detection. Some research contributions were developed to detect a new object in the background of the video scene then to remotely notify that to the end user. The image of the detected object may be then transmitted through the network when demanded by the application. Shin-Chih Tu et al. In [1] presented a new scheme to detect change in the background of a video scene using WMSN. They have based their approach on the detection

of a significant difference in the background. The performance of this scheme was addressed to show a low complexity at the implementation level and a high accuracy in the detection of new objects. Nevertheless, this solution was not specified to ensure the identification of a specific object that can be applied to enable potential application related to different areas.

Biljana et al. In [3] were interested to develop a solution for motion detection over WMSN that can be used in the area of surveillance. They specified a new scheme where the captured images will be divided in a set of small size blocks. This subdivision helps to detect using a comparison process a change compared to the reference sub-block in the background of the scene. This approach helps to save energy and is very attractive for low-bandwidth communication. It can be well applied in the context of object detection; however it was not developed to identify a specific object target.

A shape features methods for matching to identify a target objects is one of the most interesting schemes. The shape context has been presented by Serge et al. in [14]. They developed a new solution for matching based on distance between shapes. This proposed scheme demonstrated a good results regarding to the simplicity and accuracy. An interesting survey in the image feature extraction is presented by Yang et al. in [9]. The authors divided the approaches to six main approaches (shape signature, polygonal approximation, space interrelation feature, moments, scale-space methods, and shape transform domains).

Teresa et al. in [10] was interested to the issue of low-power wireless image sensor. They developed an algorithm to minimize power consuming by reducing the data acquired for the images. This algorithm is based on the idea of minimizing the communication between nodes in the processing of the captured image and sharing the data result.

Another related research work has been presented in [8]. It concerns object motion detection in WMSN. The presented idea is based on background extraction. The authors divided the image into m*m blocks of pixels using the expectation and variance that will help in the extraction of the background. This method seems to be very useful in the context of WMSN and might fit well with the requirement of our application for object identification.

In [16] S. Vasuhi et al. presented a new method for object detection and tracking for multiple objects in WMSN. They used for object detection the following steps: background techniques, Haar Wavelet for the feature extraction method and the joint boosting algorithm for the classification. While this method looks to be very interesting, the authors did not addressed the complexity and the power consumption of this scheme in the context of WMSN to prove up the adequacy to the constraints of these systems.

In [4, 5, 6 and 7] the authors studied the problems of new object detection and tracking by the use of WMSN. Different methods were specified and designed to ensure this objective. But still the problem of specific target object identification was not studied. It requires more effort to define the efficient way to perform this application through wireless multimedia sensor networks.

23.3 The Proposed Scheme for Object Detection and Identification

The well-known schemes developed for object identification based on image processing cannot be directly applied in the area of WMSNs. Our contribution is to specify new scheme that inspires from algorithms developed basically for computer vision while adapting it to meet the constraints of WMSNs. The general scheme for object detection and identification is described by the following sequential steps (Fig. 23.1). The main consideration in the specification of this scheme is to reduce:

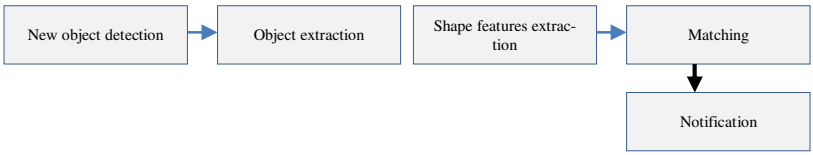


Fig. 23.1 General structure of the identification scheme

- The memory usage and mainly the size of feature's vector of the object.
- The number of arithmetic operations in order to achieve low complexity processing.

A. *New object detection:* The detection of a new object is based on the approach of background subtraction. The used approach divides the image on a set of blocks of 8*8 pixels, and then, the difference at the level of pixels is processed between corresponding blocks of the new image and the image of the background. If the whole difference is greater than a certain threshold (Tth) a new object is supposed detected.

B. *Object extraction:* when a new object is detected the set of image blocks containing the object will be isolated that reduces the useful image size. The new image is then transformed to the binary level that allows applying an edge-detection method for segmentation to extract the shape and then the signature.

C. *Features extraction:* The feature extraction process of the detected object is the main task on which depend the performances of the whole scheme for identification. The main keys considered on the design and the specification of this task are mainly:

- Low complexity method.
- To keep as short as possible the feature vector that represents the object descriptor.

In the literature review [6, 7] it was proved that the shape-based recognition methods are very attractive to achieve low-complexity and high efficiency that matches very well the previous indicated considerations. In this chapter we will address the efficiency of the centroid distance method as a shape-based feature to identify a target in WMSN [9]. The main motivation to use the centroid distance shape signature is related to its simplicity and accuracy for identification compared to other shape signature's methods such as area function. In addition, this method is invariant to translation and when normalized it is also invariant to the scale. If a shape is represented by its region function, its centroid (g_x, g_y) is given by (23.1) [9]

$$\begin{cases} g_x = \frac{1}{N}\sum_{i=1}^{N} x_i \\ g_y = \frac{1}{N}\sum_{i=1}^{N} y_i \end{cases} \tag{23.1}$$

The centroid distance signature of the object is, then, expressed by the distance of the boundary points from the centroid (g_x, g_y) of the shape as follows (23.2):

$$r(t) = ([x(t) - g_x]^2 + [y(t) - g_y]^2)^{\frac{1}{2}} \tag{23.2}$$

r(t) is defined for a set of number N, $t \in [1, N]$.

The feature vector represents the distance to the centroid in function of the angle θ that is calculated as follows (23.3):

$$\theta(n) = \arctan\frac{y(t) - y(t-w)}{x(t) - x(t-w)} \tag{23.3}$$

Where w is a small window to calculate the angle θ to be more accurate for efficient results.

D. *Matching:* Many methods can be used to compare the extracted features vector of the detected object with the reference one. One of the most interesting is the Spearman rank correlation or Spearman's Rho [13, 14] to match the similarity between the signatures of the two objects. We compared the signature of the new detected object to the reference one that is supposed to be saved on the memory of the mote. The Spearman's Rho (ρ) is defined for a given a two distances vectors, $D_1=\{d_0, d_1, d_3, ..., d_n\}$ and $D_2=\{d_0, d_1, d_2, ..., d_n\}$ and a two ranking vectors, R_1, R_2 for the two distances vectors, D_1, D_2, by (23.4)

$$\rho = 1 - \frac{6\sum_{i=0}^{n}(R_1(i)-R_2(i))^2}{n(n^2-1)} \tag{23.4}$$

$R_1(i)$ and $R_2(i)$ represent ranks of the two vectors D_1 and D_2 and n is the size of the vector. Spearman's ρ is normalized between -1 and 1. The Interpretation of the correlation coefficient values is given in [13, 14].

E. *End user notification*: When the object is detected, the event will be notified to the end user. At that level the mote has to process the notification according to the end user requirements. A big gain in time and power-consumption can be achieved at this step.

23.4 Performances Analysis the Proposed Scheme

We studied the capability of the proposed scheme to ensure object identification for a specific target under different conditions. The idea is to check out the invariance of the proposed method mainly for translation, rotation and for the scale. The following figures (Fig. 23.2 & Fig. 23.3) show the tested images and the capabilities of the proposed scheme to successfully identify the target. Grayscale images (128*128 pixels, 8 bpp) were used (as well as others sizes). The signature given by the centroid distance method is used for matching with the features.

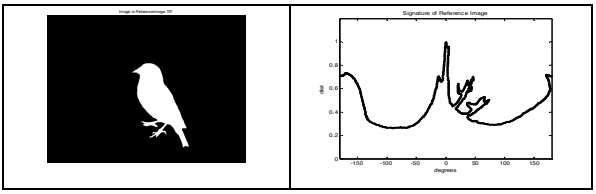


Fig. 23.2 Reference object with centroid distance signature

The proposed scheme was programmed with Matlab in order to check out its performances. Fig 23.2 shows the target object and its shape feature based on the centroid distance function. Fig 23.3 sums up the image of the bird (target object) taken in different positions and their signatures of the centroid distance compared to the reference one.

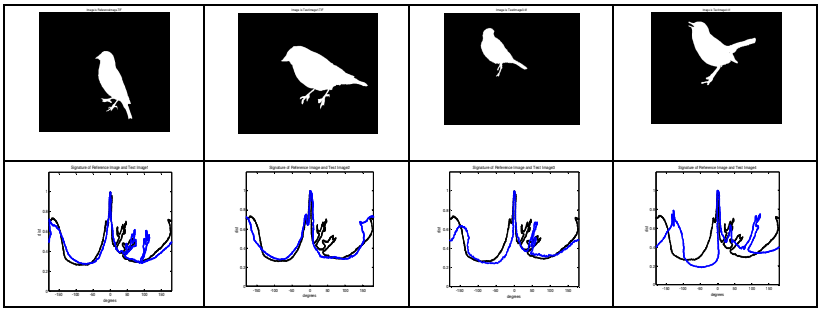


Fig. 23.3 Test images of a bird with centroid distance signature

The application of the matching method based on Spearman's Rho proved that this method is able to successfully identify the target. As shown in Fig. 23.4, even if the target object appears in different positions, scales and orientations compared to the reference one, the algorithm to extract the features of the detected object and to match it with the reference vector was able to identify the object with high percentage. In fact, this percentage for the different positions is always greater than 75% that expresses high correlation with the reference. This result reflects the capability of our scheme to achieve efficiently specific target object detection and identification.

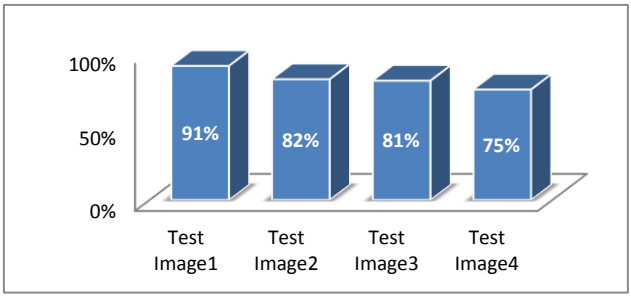


Fig. 23.4 Birds matching rate

23.5 Implementation of the Scheme on tinyOS Based Platforms

The proposed scheme was intentionally specified to meet low complexity for efficient power deployment in WMSN. To check the efficiency of this scheme we estimated the performances when implemented on crossbow motes (MICA2 and MICAZ) used to build up WSN. For that purpose, we have used an image at the grayscale level (64*64 pixels 8bpp) in order to evaluate the processing time and the required energy. The vector feature's was considered with a size of 128 values. This value allows an optimal ratio of matching with the reference vector. The scheme for object detection and identification was implemented in the emulator for microcontroller WinAVR that allows checking out the number of clock cycles for Atmel series. Then, using the characteristics of the microcontrollers for MICA2 and MICAZ, the power consumption and the processing time were calculated. Table 23.1 sums up the energy consumption and the processing time for the indicated platforms. It shows well that the main consuming part of the scheme is related to the segmentation and the object extraction from the image. The method of feature extraction based on centroid distance consumes low-power consumption and looks to the energy constraint of the mote. When compared to other schemes related to image transmission in WMSN, our proposed method outperforms them in power consumption and the time processing even if we consider the notification step (Table 23.2). It was shown in [11] that the energy cost of a (128 × 128, 8 bpp image is about 860 mJ and the time processing is around 13.5 s.

Table 23.1 Evaluation of the proposed scheme for MICA2 and MICAZ

	Processing		
	Clk cycles	Time(s)	Energy (mj)
New object detection	2505220	0, 34	7, 50
Extraction and segmentation	7926851	1,08	23 ,76
Features extraction	3540661	0,48	10,56
Whole scheme (without notification)	13972732	1,9	41,8

Table 23.2 shows the notification process and the required amount of energy. Globally, MicaZ consume less power than Mica2 while transmitting the notification. This is mainly related to the hardware characteristics of transceiver circuit. However the main note is that depending on the requirements of the end user, the source mote can smartly notify with less bit-stream which achieves low power processing and unloads the network.

Table 23.2 Power consumption for notification in the source mote for MICA2 and MICAZ

	Data stream for notification (bits)	Energy for Mica2 [mJ]	Energy for MicaZ [mJ]
Processing and transmission of 1byte	8	41,814	41,801
Processing and transmission of the feature's vector	1024	43.670	42,030
Extraction of useful information and transmission	2084	45,540	42,275

23.6 Conclusion

This chapter discussed an approach for image based object identification in wireless multimedia sensor network (WMSN). The main aim of the proposed scheme is to achieve low power processing at the source mote and to unload the network. This scheme is based on the idea of identifying the target object at the source mote before sending the useful data to the end user through the network. It uses the object shape features to abstract the target and the new detected object. In this chapter, we studied the application of the method of centroid distance to build up feature's vector for identification. The performances of the proposed scheme were analyzed for object identification. It showed high performances for specific target identification while keeping a low-power processing compared to image transmission with the application of compression scheme.

As future work, we think that the application of others shape schemes for object identification such as curvature function will ensure better performances and might reduce the power consumption.

Acknowledgement. This work is a part of the project RC1303102 that is supported by the Research Center of the College of Computer and Information Sciences - King Saud University.

References

1. Shin-Chih, T., Guey-Yun, C., Jang-Ping, S., Wei, L., Kun-Ying, H.: Scalable continuous object detection and tracking in sensor networks. Journal of Parallel and Distributed Computing 70(3) (2010)
2. Yelisetty, S., Namuduri, K.R.: Image change detection using wireless sensor networks. In: Third IEEE DCOSS 2007, USA, pp. 240–252 (2007)
3. Stojkoska, B.L., Davcev, D.P., Trajkovik, V.: N-Queens-based algorithm for moving object detection in distributed wireless sensor networks. CIT 16(4), 325–332 (2008)
4. Essaddi, N., Hamdi, M., Boudriga, N.: An image-based tracking algorithm for hybrid wireless sensor networks using epipolar geometry. In: IEEE ICME, New York, NY (2009)
5. Mauricio, C., Mehmet, V.C., Senem, V.: Design of a wireless vision sensor for object tracking in wireless vision sensor networks. In: Proceedings of the Second ACM/IEEE International Conference on Distributed Smart Cameras (2008)
6. Tu, G., Gao, S., Zhang, Z.: Object tracking and QoS control for wireless sensor networks. Chinese Journal of Electronics 18(4) (2009)
7. Ikeura, R., Niijima, K., Takano, S.: Fast object tracking by lifting wavelet filters. In: SPIT Conference (2003)
8. Peijiang, C.: Moving object detection based on background extraction. In: Computer Network and Multimedia Technology, pp. 1–4. IEEE (2009)
9. Yang, M., Kpalma, K., Ronsin, J.: A survey of shape feature extraction techniques. In: Pattern Recognition, pp. 43–90 (2008)
10. Ko, T.H., Berry, N.M.: On scaling distributed low-power wireless image sensors. In: Proceedings of the 39th Annual Hawaii International Conference on System Sciences (HICSS 2006). HICSS, vol. 9, p. 235c (2006)
11. Kaddachi, M.L., Soudani, A., Lecuire, V., Torki, K., Makkaoui, L., Moureaux, J.: Low power hardware-based image compression solution for wireless camera sensor networks. Computer Standards & Interfaces 34(1), 14–23 (2012)
12. Muselet, D., Trémeau, A.: Rank correlation as illumination invariant descriptor for color object recognition. In: Proceedings of the 15th International Conference on Image Processing, pp. 157–160 (2008)
13. Ayinde, O., Yang, Y.-H.: Face recognition approach based on rank correlation of Gabor filtered images. Pattern Recognition 35, 1275–1289 (2002)
14. Belongie, S., Malik, J., Puzicha, J.: Shape matching and object recognition using shape contexts. IEEE Transactions on Pattern Analysis and Machine Intelligence 24(4), 509–522 (2002)
15. Chefi, A., Soudani, A., Sicard, G.: Hardware compression scheme based on low complexity arithmetic encoding for low power image transmission over WSNs. International Journal of Electronics and Communications (AEÜ) 68(3), 193–200 (2014)
16. Vasuhi, S., Annis Fathima, A., Anand Shanmugam, S., Vaidehi, V.: Object detection and tracking in secured area with wireless and multimedia sensor network. In: Benlamri, R. (ed.) NDT 2012, Part II. CCIS, vol. 294, pp. 356–367. Springer, Heidelberg (2012)
17. Zuo, Z., Qin, L., Wusheng, L.: A two-hop clustered image transmission scheme for maximizing network lifetime in wireless multimedia sensor networks. Computer Communications 35(1), 100–108 (2012)

Author Index